Digital Object Identifier

Binary Social Mimic Optimization Algorithm with X-Shaped Transfer Function for Feature Selection

KUSHAL KANTI GHOSH [1], PAWAN KUMAR SINGH [2] (Member, IEEE), JUNHEE HONG [3], ZONG WOO GEEM [3], RAM SARKAR [1] (Senior Member, IEEE)
[1]Department of Computer Science and Engineering, Jadavpur University, Kolkata, India
[2]Department of Information Technology, Jadavpur University, Kolkata, India
[3]Department of Energy IT, Gachon University, Seongnam 13120, South Korea

Corresponding author: Zong Woo Geem (email: geem@gachon.ac.kr)

This work was supported by ...

ABSTRACT Definitive optimization algorithms are not able to solve high dimensional optimization problems when the search space grows exponentially with the problem size and an exhaustive search will be impractical. To encounter this problem, researchers use approximate algorithms. A category of approximate algorithms is meta-heuristic algorithms which have shown an acceptable efficiency to solve these kind of problems. Social Mimic Optimizer (SMO) is a recently proposed meta-heuristic algorithm used to optimize continuous problems. It is proposed by following the behavior of people in society. SMO can efficiently explore the solution space for optimal or near-optimal solution minimizing a given fitness function. To convert the continuous search space to a binary one, transfer functions are utilized. Two types of transfer functions are commonly used in literature: S-shaped and V-shaped. We have proposed eight different binary versions of SMO and applied these variants to select the optimal feature subset that maximizes classification accuracy of 18 standard UCI datasets. The effect a transfer function has on the binary variants is very important as selecting a particular subset of features based on the values attained by the algorithm in continuous search space depends on on the utilized transfer function. To this end, we have proposed a new transfer function, called X-shaped transfer function, to enhance the exploration and exploitation ability of binary SMO . The proposed X-shaped transfer function utilizes two components and crossover operation to obtain a new solution. Effect of the proposed X-shaped transfer function is compared with the effects of four S-shaped and four V-shaped transfer functions on SMO in terms of achieved classification accuracy, rate of convergence, and number of features selected over 18 standard UCI datasets. The proposed algorithm is also compared with five state-of-the-art meta-heuristic FS algorithms. Experimental results confirm the efficiency of the proposed approach in improving the classification accuracy compared to other meta-heuristic algorithms, and the superiority of X-shaped transfer function over commonly used S-shaped and V-shaped transfer functions.

INDEX TERMS Social Mimic Optimization, Transfer Function, Meta-heuristic, Feature Selection

I. INTRODUCTION

In this era of computer and technology, with every advancement in the field of image processing, pattern recognition, financial analysis, business management, medical studies [1, 2] and others, we are bound to deal with huge amount of data, whose dimensions are increasing everyday. Two most important categories of the methods in the field of data mining are classification and clustering, which work on the features or attributes representing the dataset to make some predictions or extract useful information from the datasets. However, when the dimensions *i.e.*, number of features of the datasets are increased then the performance of these methods gets affected considerably [3]. Again, high dimensional datasets have various disadvantages such as larger time requirement for model construction, possible existence of irrelevant and redundant data, and degraded performance due to redundancy of features which make analysis or classification of the data very difficult. Here comes the importance of feature selection (FS) methods. FS is a data pre-processing step which attempts to remove all possible irrelevant and redundant features [4] from the underlying dataset or feature vector, and thereby reduces the storage and time requirement to process the data.

FS is considered to be an NP-complete combinatorial optimization problem. Generating all possible subsets and evaluating those are not feasible for large datasets since, for a dataset containing n features, 2^n feature subsets will be generated and evaluating all of those requires a huge computational cost. There are randomized algorithms that attempt to search for the optimum feature subset in a randomized manner. On the other hand, a heuristic search strategy performs a guided search which may not always find the optimum solution rather tries to obtain a near-optimal solution in terms of computational time. Heuristic approaches are classified into two categories- specific heuristics which are designed for a particular problem, and general purposed meta-heuristics which are designed to solve a wide range of problems [5].

Based on the usage of learning algorithm, FS methods can broadly be divided into two categories [6]: filter and wrapper. Filter methods do not use any learning algorithm during elimination (selection) of the irrelevant (important) features, rather use different pre-defined scoring criteria to rank the features indicating their importance in terms of classification ability. Wrapper methods use learning algorithms (such as classifiers) as a part of the selection and evaluating the subset of the selected features in intermediate steps. Filter methods are faster but wrapper methods, in general, perform much better [6]. Meta-heuristic methods are wrapper based, since they require a classification algorithm for evaluation of a selected feature subset.

In the last decade, meta-heuristic algorithms have become quite popular in solving FS problems due to their ability to obtain an optimal or near-optimal solution in a reasonable time [7]. Two main characteristics of these algorithms are: exploration or diversification, which is the ability to search the whole solution space for new solution in each iteration to avoid local optima and exploitation or intensification, which implies finding a better solution in the neighborhood of the obtained solution, leading to faster convergence. A good meta-heuristic algorithm tries to find a proper balance between exploration and exploitation.

In this work, we have made an attempt to propose a meta-heuristic FS algorithm. Here, we have introduced a new transfer function and applied this transfer function to a recently proposed meta-heuristic optimization algorithm called Social Mimic Optimization (SMO) algorithm for the purpose of FS. Main contributions of this work are as follows:

- A novel X-shaped transfer function is introduced.
- A new FS technique is developed following a recently proposed optimization algorithm called SMO.
- The performance of the new transfer function in combination with SMO is compared with widely used four S-shaped and four V-shaped transfer functions.
- The proposed FS method is evaluated on 18 standard UCI datasets [8].
- It is also compared with 5 state-of-the-art meta-heuristic based FS methods.
- The performance of the proposed FS method is statistically validated using using Wilcoxon rank-sum test [9].

The rest of this paper is organized as follows: Section II provides a brief review about the FS methods and transfer functions present in the literature. Section III provides detailed description of the proposed FS method. The results obtained by the FS versions of SMO are explained in Section IV. Section V provides the comparison of the proposed model with five state-of-the-art FS methods. Lastly, Section VI concludes this work and provides directions for future extension of this work.

II. LITERATURE SURVEY

FS is an optimization problem where the aim is to simultaneously maximize the classification accuracy and minimize the number of features to be used for classification. The role of FS is crucial because it helps us to gauge the performance of different machine learning and data mining techniques.

In the past two decades, nature-inspired meta-heuristic algorithms are at the forefront due to number of important factors of these algorithms: easy to adopt, flexible, not requiring very complex mathematical derivation, their ability to avoid local optima. These algorithms have the ability to exploit the information of the population in order to find the optimal solutions. Meta-heuristic algorithms can be divided into different categories based on different criteria: single solution based and population based [10], nature inspired and non-nature inspired [11], metaphor based and non-metaphor based [12]. From the 'inspiration' point of view, these algorithms can roughly be divided into four categories [13]: Evolutionary, Swarm inspired, Physics based, and Human related.

- **Evolutionary algorithms** are basically inspired from biology. It utilizes crossover and mutation operators to evolve the initial population, usually selected in a random fashion, over the iterations and eliminates the worst solutions in order to obtain the improved solution. Genetic algorithm (GA) [14] is a well-known method of this category which follows the Darwin's theory of evolution. Co-evolving algorithm [15], Cultural algorithm [16], Genetic programming [17], Grammatical evolution [18], Bio-geography based optimizer [19], Stochastic fractal search [20] etc. are some well-known evolutionary algorithms.
- **Swarm inspired algorithms** imitate individual and social behavior of swarms, herds, schools, teams or any group of animals. Every individual has its own behavior, but the behavior of the accumulated individuals helps to solve complex optimization problems. One of the most popular algorithms of this category is Particle swarm optimization (PSO) [21], developed by following the behavior of flock of birds. Another notable method of this category is Ant colony optimizer (ACO) [22], inspired from the foraging method of some ant species. Some other methods belonging to this category are: Shuffled frog leaping algorithm

[23], Bacterial foraging [24], Artificial bee colony [25], Firefly algorithm [26], Grey Wolf optimizer (GWO) [7], Ant Lion optimizer (ALO) [27], Whale optimization algorithm [28], Grasshopper optimization algorithm (GOA) [29], Squirrel search algorithm [30], Harris Hawks optimization (HHO) [31] etc.

- **Physics based algorithms** are inspired by the rules governing a physical process. The inspiring physical process ranges from music, metallurgy to mathematics, physics, chemistry, and complex dynamic systems. One of the oldest algorithms of this category is: Simulated Annealing (SA) [32], developed by following the annealing [33] process of metals present in metallurgy and materials sciences. Another popular method of this category is Gravitational search algorithm (GSA) [34], developed by following gravity and mass interaction. Some other methods of this category are: Self propelled particles [35], Harmony search (HS) algorithm [36], Black hole optimization [37], Sine Cosine algorithm [38], Multi-verse optimizer [39], Find-Fix-Finish-Exploit-Analyze [40] etc.
- **Human related algorithms** searches for the global optima by following human behavior. Teaching-Learning-Based optimization [41] is one such popular method belonging to this category, developed by following the enhancing procedure of class grade. Some other methods of this category are: Society and civilization [42], League championship algorithm [43], Fireworks algorithm [44], Tug of war optimization [45], Volleyball Premier League algorithm [46].

FS is a binary optimization problem, and transfer functions are required to convert the search space of a continuous optimization algorithm to a binary one. Transfer function generates a probability value based on the position/velocity of a solution and with this probability value, real valued solution is converted to a binary one. Kennedy and Eberhart have proposed binary PSO (BPSO) algorithm, using a sigmoid transfer function [47]. GA is used in [48] for the selection of features in automatic pattern classifier. In [49], the authors have proposed V-shaped transfer function. In [34], binary GSA (BGSA) is proposed using V-shaped transfer function ($|\tanh(x)|$). In [50], the authors have proposed eight binary variants of PSO using four S-shaped and four V-shaped transfer functions. These transfer functions are given in Table 1.

TABLE 1: Popular S-Shaped and V-Shaped transfer functions (used for comparison with X-Shaped transfer function)

S-shaped transfer functions		V-shaped transfer functions	
S1	$T(x) = \frac{1}{1+e^{-x}}$	V1	$T(x) = \lvert tanh(x) \rvert$
S2	$T(x) = \frac{1}{1+e^{-2x}}$	V2	$T(x) = \left\lvert erf(\frac{\sqrt{\pi}}{2}x) \right\rvert$
S3	$T(x) = \frac{1}{1+e^{-3x}}$	V3	$T(x) = \left\lvert \frac{x}{\sqrt{1+x^2}} \right\rvert$
S4	$T(x) = \frac{1}{1+e^{-x/2}}$	V4	$T(x) = \lvert \frac{2}{\pi} \arctan(\frac{\pi}{2}x) \rvert$

In [51], the authors have proposed six binary variants of ALO using three S-shaped S2, S3, and S4 (as mentioned in Table 1) and three V-shaped V2, V3, V4 (as mentioned in Table 1) transfer functions. In [52], Dragonfly algorithm is used for FS by utilizing V3 transfer function and applied on 18 standard UCI datasets. In [53], binary variants of GOA is proposed using S1 and V1 transfer functions. HHO [54] is converted to its binary version using S1 and V1 transfer functions and applied on microarray datasets. In [55], the authors have proposed binary variants of Butterfly optimization algorithm using S1 and V2 transfer functions and applied on 21 UCI datasets. In [56], four V-shaped transfer functions V1, V2, V3, and V4 are used to convert GWO into its binary variant for solving FS problems.

Presence of such a significant number of meta-heuristic FS algorithms along with transfer functions, clearly raises the question about the need for (i) another meta-heuristic FS method, and (ii) another transfer function. However, as indicated by *No Free Lunch* [57] theorem for optimization, there cannot be any single algorithm which will be equally applicable for all the optimization problems desiring optimal solutions. With each new algorithm following any regular or natural phenomenon, researchers primarily aim to provide some new facet to the algorithm where both exploration and exploitation will have a superior trade-off, thereby trying to get away from the local optima and eventually compass to the global optima. Nevertheless, accomplishing these objectives are not straightforward, hence motivating researchers to propose new algorithm that can be applicable to different problem domains. In summary, this is the key reason to the researchers to make an attempt in order to formulate better methods in comparison with the past methods which, thus, keeps the research alive in this domain. For a specific problem, in order to discover the best algorithm, the NFL ought to guide researchers that they have to concentrate on the particular problem at hand, the hypotheses, the priors (additional data), the information and the cost.

For the complex optimization problems, the multi-modal functions are having huge number of dimensions and finding an ideal value for all those dimensions at the same time is almost next-to-impossible. This challenging aspect of the optimization problem prompts researchers to plunge into the field of meta-heuristic strategies where the aim is to get an optimal solution within a reasonable amount of time. FS is considered as an optimization problem - there may exist numerous optimal subsets *i.e.*, having same dimension and same precision. Here likewise, it would be extremely hard to discover an optimal feature set where burden of the extra storage space and running time alongside the performance of the machine learning algorithm would be lessen. In this way, research is still done to contribute algorithms which can meet these requirements. This has also inspired us to propose a new meta-heuristic FS methodology based on the SMO [58] algorithm.

III. PRESENT WORK

A. SOCIAL MIMIC OPTIMIZATION: AN OVERVIEW

SMO algorithm [58] is proposed by following human behavior. Each individual tries to 'mimic' or assimilate himself/herself to someone more esteemed, more intelligent or more powerful. Accordingly, each solution (analogous to individual) in an optimization problem moves towards the global optima reached so far by imitating the parameters of that global optima. $Follower$ represents the population, $Follower_i$ represents i^{th} solution in the population, $Leader$ represents the global optima obtained so far. During an iteration, each $Follower_i$ calculates the difference between its fitness value and the fitness value of the global optimal using Equation 1.

$$Difference = \frac{fitness(Leader) - fitness(Follower_i))}{fitness(Follower_i)} \tag{1}$$

$$if(Difference == 0) \;\; Difference = random(0, 1] \tag{2}$$

In next step, each $follower_i$ updates itself using Equation 3.

$$Follower_i = Follower_i + Difference \times Follower_i \tag{3}$$

Fitness value of each $Follower_i$ is calculated and $Leader$ is updated accordingly. The reason we have chosen this optimization method because SMO is simple to implement but can produce effective results. Besides, it does not require any initial parameter in contrary to other popular meta-heuristic algorithms, except only the population size and maximum number of iterations. As a result of this, no parameter tuning is required which itself requires exhaustive experiments.

B. PROPOSED METHOD

Let the original feature set be $\mathcal{F} = \{f_1, f_2, ..., f_D\}$, where D is the total number of features or the dimension of the feature set and let the class label be $\mathcal{C} = \{c_1, ..., c_l\}$, where l is the number of classes. FS method tries to find out a subset $\mathcal{S} = \{s_1, ..., s_m\}$, where $m < D$, $\mathcal{S} \subset \mathcal{F}$ and $\mathcal{S}$ has lower classification error than any other subset of same size or any proper subset of $\mathcal{S}$.

It has already been mentioned that FS is a binary optimization problem [59], where the solution is limited to binary values $\{0, 1\}$. Here, a solution is represented using a binary vector where 1 indicates that corresponding feature is selected and 0 indicates otherwise. The size of this vector is equal to number of features in the original dataset. SMO is proposed to solve continuous optimization problems where a solution consists of real values. To map the continuous search space of the standard SMO to a binary one, a transfer function is required [50]. In the literature, there are two types of transfer functions commonly used, which are S-shaped and V-shaped.

In case of S-shaped transfer functions, solutions are updated based on Equation 4.

$$F_i^d(t+1) = \begin{cases} 1 & \text{if } rnd < SFunction(F_i^d(t+1)) \\ 0 & \text{if } rnd \geq SFunction(F_i^d(t+1)) \end{cases} \tag{4}$$

where $rnd \in [0, 1]$ is a random number, $F_i^d(t+1)$ represents the d^{th} dimension of the i^{th} solution (follower) in $(t+1)^{th}$ iteration.

In case of V-shaped transfer functions, solutions are updated based on Equation 5.

$$F_i^d(t+1) = \begin{cases} F_i^d(t) & \text{if } rnd < VFunction(F_i^d(t+1)) \\ \sim F_i^d(t) & \text{if } rnd \geq VFunction(F_i^d(t+1)) \end{cases} \tag{5}$$

where $rnd \in [0, 1]$ is a random number, $F_i^d(t+1)$ represents the d^{th} dimension of the i^{th} solution in $(t+1)^{th}$ iteration, $F_i^d(t)$ represents the d^{th} dimension of the i^{th} solution in t^{th} iteration, $\sim F_i^d(t)$ represents the complement of $F_i^d(t)$, *i.e.,* if $F_i^d(t) = 0$, then $\sim F_i^d(t) = 1$ and *vice-versa.*

Now, in case of S-shaped transfer function, solution in the next ($(t+1)^{th}$) iteration is modified without considering the solution in the current (t^{th}) iteration. This may diverge the agents, leading to slower convergence of the algorithm. In swarm inspired algorithms, where the agents are updated based on their velocity values, a big value of velocity in the positive or negative direction shows that agents should have large movements to reach the optimum position. In contrast, a small value of the velocity indicates insignificant movement. Again, the zero velocity means that the new position should not be changed [49]. Now, these concepts are changed by using the S-shaped transfer function. The value of velocity in the negative and the positive directions creates different values for the new position. Moreover, the zero value of velocity generates different values of zero or one with probability 0.5 for the new position [49]. Whereas, with V-shaped transfer functions, the solution may get stuck in local optima since if low velocities are associated with a particular solution, in next iteration the solution remains the same with high probability. Transfer function performs key role in helping an optimization algorithm to find the optimum solution [50]. In early steps, the exploration is very important to search promising regions and avoid getting trapped in local optima but during the later steps, the exploitation is more essential so that the probability of finding better solutions gets increased. In other words, a balance between exploration and exploitation is essential to achieve a good result. In the literature, we have seen many such cases where the meta-heuristic strategies need to be enhanced by a local or global search in order to be able to find the optimal solution [60, 61, 62].

Considering the limitations of the commonly used transfer functions found in the literature, we have introduced a new transfer function which is X-shaped. Two components, as shown in Figure 1, are used to generate two different results. The best result is chosen and compared with the previous

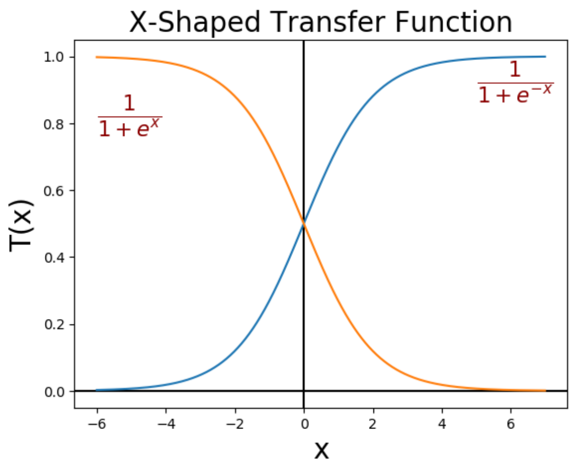


FIGURE 1: The proposed X-shaped transfer function

solution. If the new solution is better than the previous one, it will be selected as the next position; otherwise, a crossover operator is applied on the new and previous solution. In this case, the best result of crossover operator is chosen as the new position. Due to crossover, there is a chance for the new solution to retain the good characteristics of the solution of previous iteration.

To improve the exploration and exploitation abilities of the optimization algorithm (SMO here), two components are utilized Equation 6 and Equation 8, where Equation 8 is a mirror image of Equation 6 *w.r.t.* the line $y = 0$.

$$X_1(x) = \frac{1}{1+e^{-x}} \tag{6}$$

$$y_i^d(t+1) = \begin{cases} 1 & \text{if } rnd1 < X_1(y_i^d(t+1)) \\ 0 & \text{if } rnd1 \geq X_1(y_i^d(t+1)) \end{cases} \tag{7}$$

$$X_2(x) = \frac{1}{1+e^{x}} \tag{8}$$

$$z_i^d(t+1) = \begin{cases} 1 & \text{if } rnd2 > X_2(z_i^d(t+1)) \\ 0 & \text{if } rnd2 \leq X_2(z_i^d(t+1)) \end{cases} \tag{9}$$

where, y_i and z_i are the binary versions of $Follower_i$ generated by Equation 6 and Equation 8 respectively, $rnd1, rnd2 \in [0, 1]$ are random numbers.

$$F_i'(t+1) = \begin{cases} y_i & \text{if } fitness(y_i) < fitness(z_i) \\ z_i & \text{if } fitness(y_i) \geq fitness(z_i) \end{cases} \tag{10}$$

Now, if $fitness(F_i'(t+1)) < fitness(F_i(t))$, then $F_i(t+1) := F_i'(t+1)$. Otherwise, crossover operation is performed on $F_i'(t+1)$ and $F_i(t)$. The crossover results in two children, the best one is chosen as next solution. In this case, the child has a chance to retain the good qualities of the parent $F_i(t)$. Uniform crossover [63] has been chosen for crossover operation. This part is summed up in Equation 11.

$$\begin{aligned}
&\text{if } fitness(F_i'(t+1)) < fitness(F_i(t)) : \\
&\qquad F_i(t+1) := F_i'(t+1) \\
&\text{else:} \\
&\qquad [child1, child2] = crossover(F_i'(t+1), F_i(t)) \\
&\qquad \text{if } fitness(child1) < fitness(child2) : \\
&\qquad\qquad F_i(t+1) := child1 \\
&\qquad \text{else:} \\
&\qquad\qquad F_i(t+1) := child2 \\
&\qquad \text{endif} \\
&\text{endif}
\end{aligned} \tag{11}$$

In this work, we have compared the performance of the introduced X-shaped transfer function with the performance of eight different transfer functions when these are used with SMO. We have used eight popular transfer functions present in literature, four S-Shaped and four V-Shaped transfer functions. Table 1 shows the mathematical formulas of the eight transfer functions considered here and Figure 2 shows their corresponding graphs.

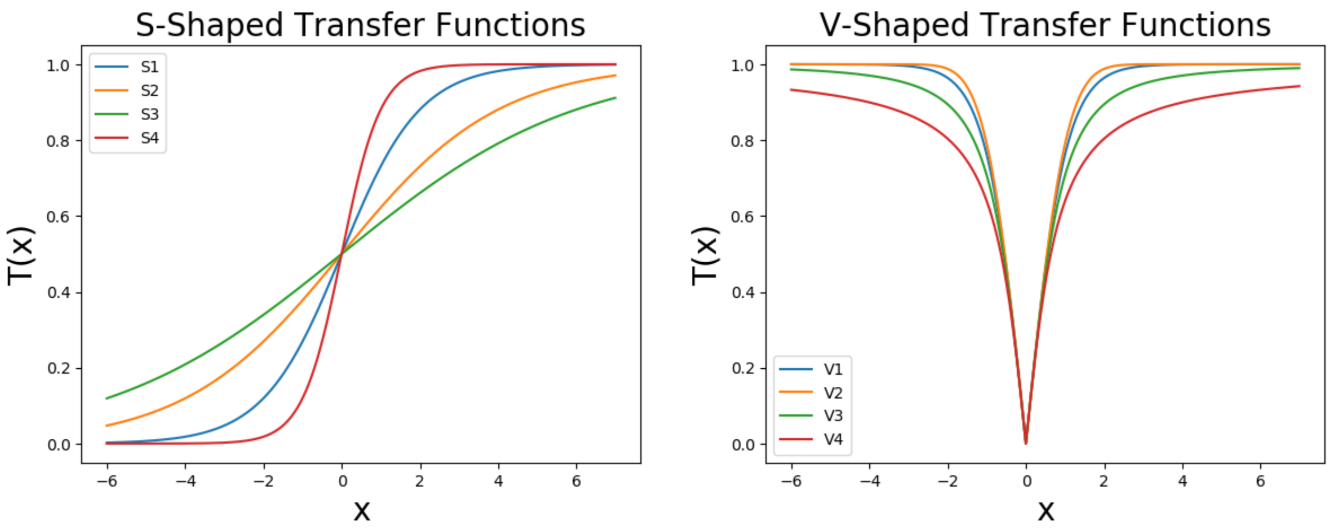
Now, FS is a multi-objective optimization problem with two main objectives: achieve maximum classification accuracy and select minimum number of features. Since these two goals are opposite in nature, we have considered classification error rate instead of accuracy. These two objectives are then combined into a single one by the proposed fitness function, given in Equation 12. Each follower (solution) is assessed by the proposed fitness function which relies on the performance of the K-Nearest Neighbor (KNN) classifier [64] in order to determine the classification error rate and on the number of features selected.

$$\downarrow Fitness = \omega\gamma(\mathcal{F}') + (1-\omega)\frac{|\mathcal{F}'|}{|\mathcal{F}|} \tag{12}$$

where $|\mathcal{F}|$ represents total number of features in the original dataset, $|\mathcal{F}'|$ represents the number of features in the selected subset, $\gamma(\mathcal{F}')$ denotes the classification error rate of $\mathcal{F}'$ using KNN classifier. $\omega \in [0, 1]$ denotes the importance of classification quality and selected subset dimension.

IV. RESULTS AND DISCUSSION

We have used KNN [64] classifier with Euclidean distance metric to measure classification accuracy of the selected optimal feature subset by SMO algorithm. As per the recommendations provided in the works described in [65, 66, 59], we have set $K = 5$. For each dataset, 5 fold cross-validation is used. Fundamentally, in k-fold cross-validation, the dataset is divided into k equal partitions (folds) where k-1 folds are utilized for training and the remaining fold is utilized for the testing purpose. This procedure is iterated for M times. We have applied the FS methods on the train fold and determined which features are to be included in the selected feature subset. From test fold, only those features are selected and



(a) (b)

FIGURE 2: (a) S-shaped and (b) V-shaped transfer functions used to compare the performance of proposed X-shaped transfer function

test classification accuracy is measured based on these using KNN classifier. Test fold is completely hidden from the FS method and used only for final evaluation purpose only. The proposed methods are implemented using Python3 [67] and graphs are plotted using Matplotlib [68]. The source code can be found in https://github.com/Rangerix/SocialMimic.

A. DATASET DESCRIPTION

For assessing the performance of the proposed FS method, 18 UCI datasets [8] are considered. The datasets are selected from various backgrounds. The underlying reason for selecting these datasets is that they contain a number of attributes and instances which represent a variety of issues on which the proposed binary approaches can be tested on [69]. The description of these datasets is presented in Table 2. It is clear from Table 2 that there are 15 bi-class and 3 multi-class datasets. The datasets are diverse in terms of both number of attributes (features) and instances. These variances help in establishing the robustness of the proposed methods.

B. PARAMETER TUNING

There are two parameters which are always very important for any multi-agent evolutionary algorithm: (a) population size and (b) number of maximum iterations. Population size characterizes how a single agent learns from other agents' experience, and iterations provide step-wise evolution of the agents. In order to find the optimal values for these two parameters, exhaustive experiments have been performed by varying one parameter *w.r.t.* the other.

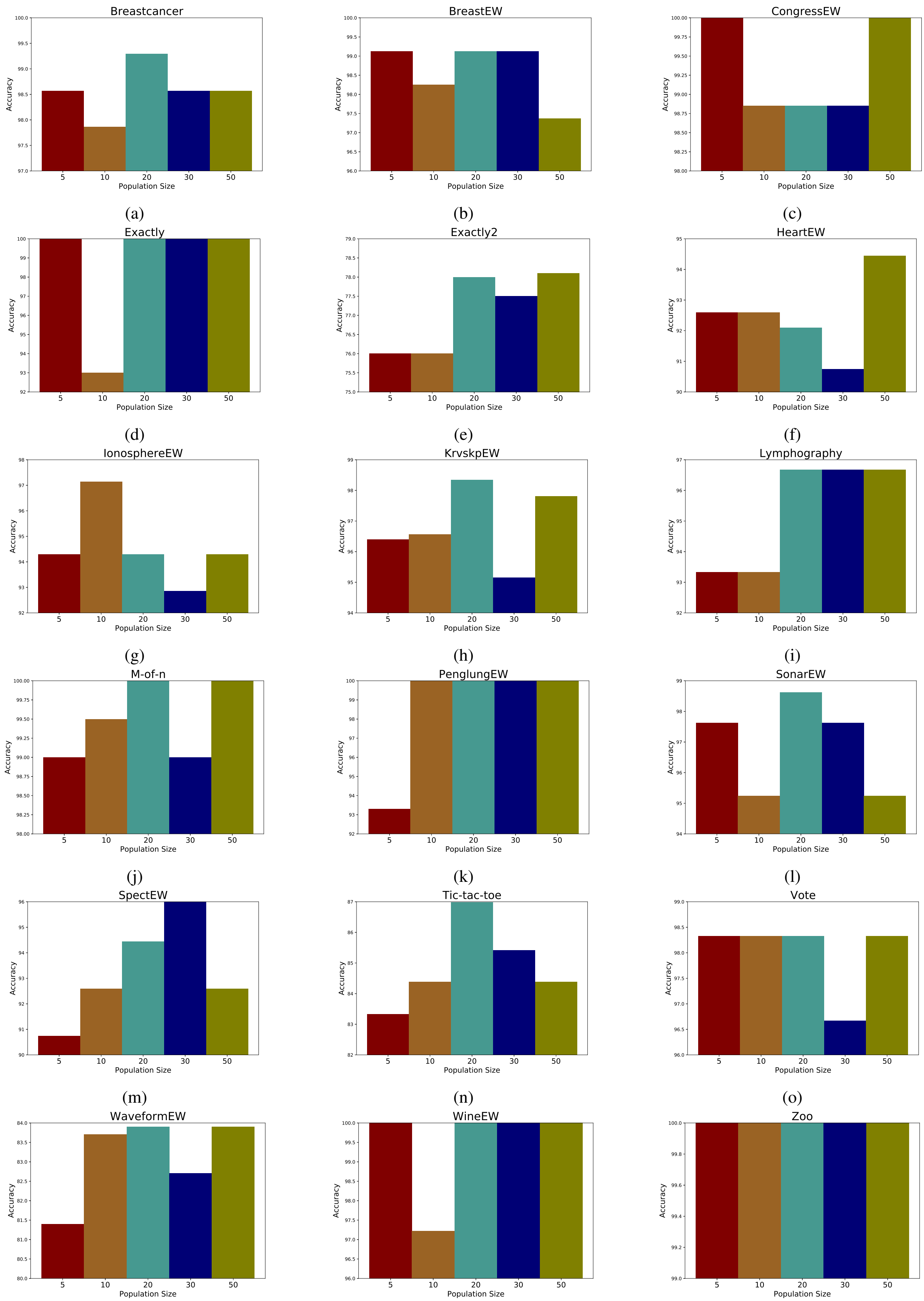
Figure 3 shows the effect of the size of the population on achieved classification accuracy using SMO algorithm with the proposed transfer function and considered S-shaped and V-shaped transfer functions. Figure 4 shows the values of the fitness function in each iteration using the proposed X-shaped and considered S-shaped and V-shaped transfer functions. Considering both Figure 3 and Figure 4, it has been decided to set the values of population size as 20 and the maximum number of iterations as 30 for all further experimentations.

C. EXPERIMENTAL RESULTS

In this section, we have discussed about the results achieved by binary SMO algorithm using the proposed X-shaped transfer function and four S-shaped and four V-shaped transfer functions. The detail results are mentioned in Table 1. We have denoted the binary SMO algorithm with i^{th} S-shaped and j^{th} V-shaped transfer function in Table 1, as SMOsi and SMOvj respectively. The proposed binary SMO with X-shaped transfer function is abbreviated as SMOX.

Now, from Table 3, it can be observed that the SMOX algorithm has achieved the highest accuracy for all the utilized 18 UCI datasets. The SMOX algorithm is able to achieve 100% classification accuracy for 9 cases (50%): Breastcancer, CongressEW, Exactly, M-of-n, PenglungEW, SonarEW, Vote, WineEW, and Zoo. For BreastEW, it has achieved the second best accuracy of 99.12% . For Exactly2, Tic-tac-toe, and WaveformEW, SMOX has achieved 80.5%, 82%, and 84.4% classification accuracy. From Table 4, it can be observed that the proposed SMOX algorithm has selected minimum number of features for exactly 8 datasets which are CongessEW, KrvskpEW, M-of-n, PenglungEW, Tic-Tac-Toe, Vote, WineEW and Zoo. However, the second best performing algorithm is found to be SMOs4 algorithm which selects the minimum number of features for almost 5 datasets such as BreastCancer, BreastEW, Exactly2, IonosphereEW and Zoo.

Figure 5 displays the average accuracies achieved by the



(p) (q) (r)

FIGURE 3: Effect of population size on classification accuracy for 18 UCI datasets using SMOX

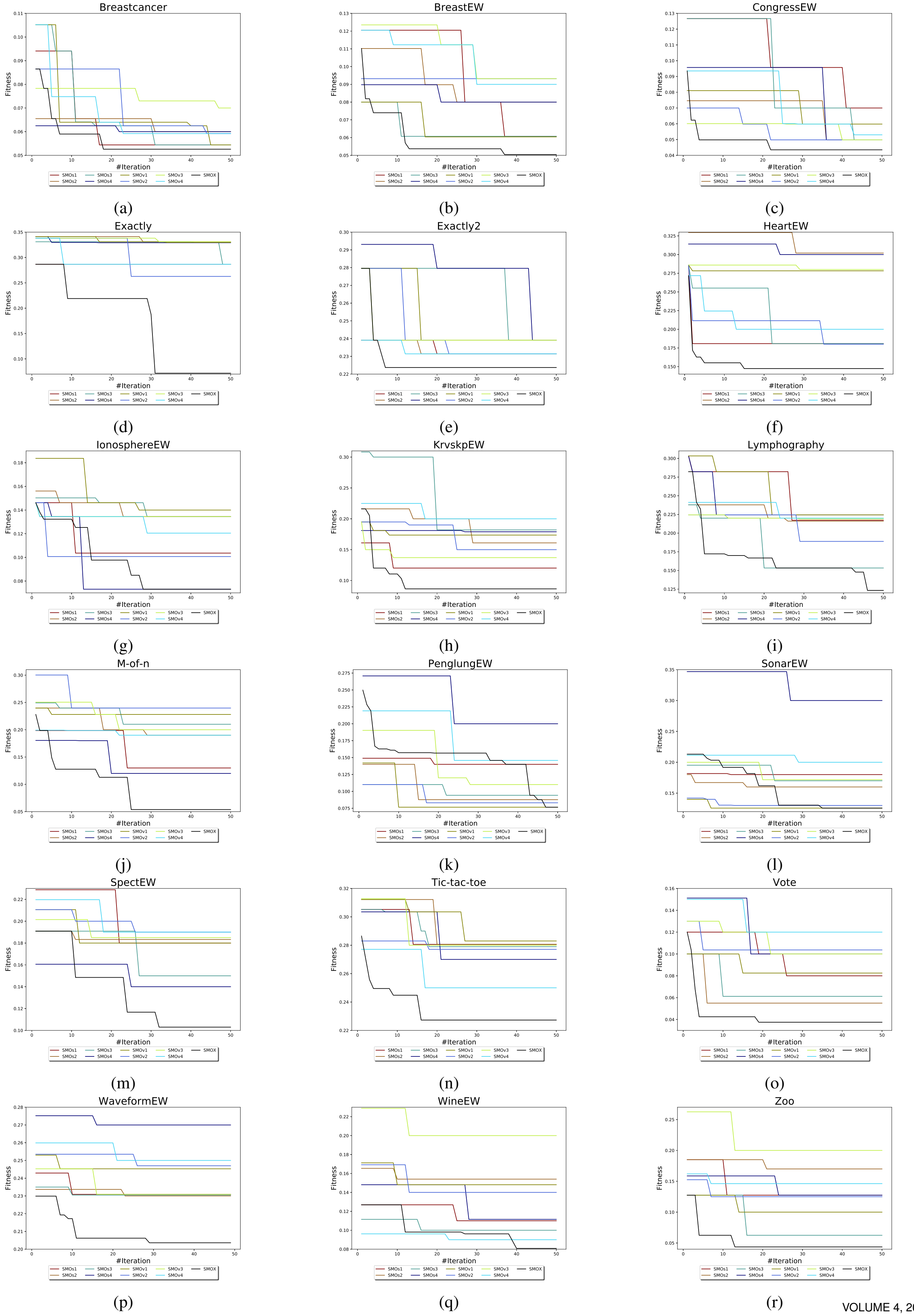


FIGURE 4: Best fitness values obtained in each iteration by SMO using X-shaped, four S-shaped and four V-shaped transfer functions for 18 UCI datasets

TABLE 2: Description of the 18 UCI datasets used in present work

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

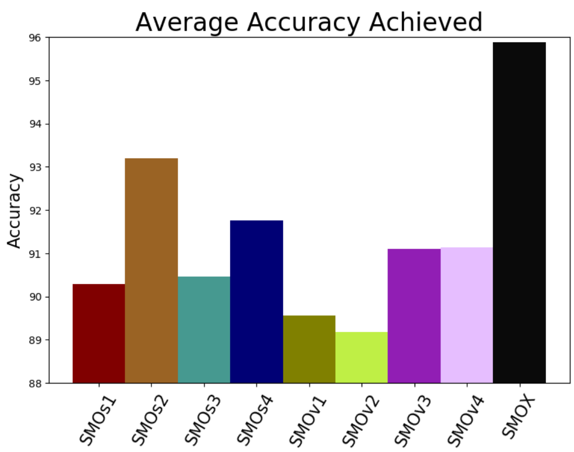


FIGURE 5: Average accuracies achieved by SMO algorithm using 4 S-shaped and 4 V-shaped and the proposed X-shaped transfer function on 18 UCI datasets

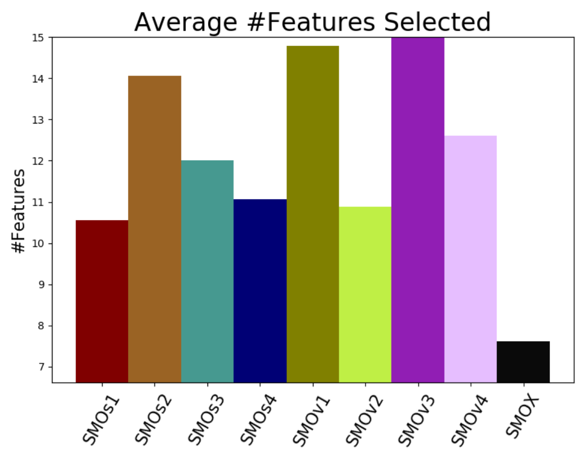


FIGURE 6: Average number of features selected by SMO algorithm using 4 S-shaped and 4 V-shaped and the proposed X-shaped transfer function on 18 UCI datasets

nine (four SMOsi, four SMOvj, SMOX algorithms) binary variants of SMO over the utilized 18 UCI datasets. It is clearly seen that our proposed SMOX algorithm has achieved the highest classification accuracy among other binary variants. Figure 6 shows the average number of features selected by the nine binary variants of SMO. From Figure 6, it can be observed that the SMOX algorithm has selected the lowest number of features in most of the cases. Upon averaging over the utilized 18 UCI datasets, the proposed SMOX algorithm has selected < 10 features.

D. STATISTICAL ANALYSIS

To determine the statistical significance of the proposed SMOX algorithm, a non-parametric statistical test, known as Wilcoxon rank-sum test [9], has been performed. This is done in order to check whether the results of proposed algorithm are statistically different from other algorithms [70]. The null hypothesis is that the two sets of results are from the same distribution, therefore any difference in the two mean ranks come only from sampling error. If the distributions of two results are statistically different, then the generated p-value from the test statistics will be < 0.05 (level of significance), as we have performed the test at 0.05% significance level, resulting in the rejection of the null hypothesis.

Here, we have deployed Wilcoxon test to prove that the obtained results by th proposed SMOX algorithm is statistically different from the obtained results by SMOsi and SMOvj methods. For each datasets, each of the binary variants has been made to run 20 times and the accuracies obtained by the SMOX algorithm is compared with each of the SMOsi and SMOvj methods via Wilcoxon test. The p-values obtained for pair-wise comparisons of SMOX, SMOsi or SMOvj algorithms for each of the 18 UCI datasets are provided in Table 5.

TABLE 3: Comparison of the classification accuracies achieved by SMO algorithm using 4 S-shaped, 4 V-shaped and the proposed X-shaped transfer functions (the highest classification accuracies are highlighted in bold)

Dataset	SMOs1	SMOs2	SMOs3	SMOs4	SMOv1	SMOv2	SMOv3	SMOv4	SMOX
Breastcancer	98.57	99.29	98.57	97.85	97.14	97.86	98.57	98.57	**100**
BreastEW	97.37	98.25	96.49	97.36	96.49	96.49	98.25	96.49	**99.12**
CongressEW	97.7	96.55	97.7	96.55	98.85	97.7	96.55	98.85	**100**
Exactly	69	87	82	75.5	84	75	70	84.5	**100**
Exactly2	76	77	76	76	76	76	76	76	**80.5**
HeartEW	88.89	83.33	85.19	81.48	77.78	79.63	81.48	85.19	**92.59**
IonosphereEW	92.86	**95.71**	91.43	92.85	94.29	91.43	92.86	94.29	**95.71**
KrvskpEW	96.09	96.87	96.71	97.183	94.68	93.58	96.87	93.58	**98.5**
Lymphography	90	90	80	86.66	86.67	93.33	83.33	83.33	**96.67**
M-of-n	92.5	97	95.5	98	88	84	**100**	99.5	**100**
PenglungEW	93.33	**100**	93.33	**100**	**100**	93.33	**100**	**100**	**100**
SonarEW	90.48	95.24	95.24	95.23	90.48	85.71	92.86	88.1	**100**
SpectEW	85.19	**96.3**	88.89	92.59	79.63	87.04	94.44	83.33	**96.3**
Tic-tac-toe	76.56	81.25	78.65	81.25	77.6	78.65	77.08	77.08	**82**
Vote	**100**	**100**	96.67	**100**	96.67	98.33	98.33	**100**	**100**
WaveformEW	83.6	83.8	81	83.2	81.7	80.1	83.3	81.7	**84.4**
WineEW	97.22	**100**	**100**	**100**	97.22	97.22	**100**	**100**	**100**
Zoo	**100**	**100**	95	**100**	95	**100**	**100**	**100**	**100**

TABLE 4: Comparison of the number of features selected by SMO algorithm using 4 S-shaped, 4 V-shaped and the proposed X-shaped transfer function (bold values signify minimum number of features selected)

Dataset	SMOs1	SMOs2	SMOs3	SMOs4	SMOv1	SMOv2	SMOv3	SMOv4	SMOX
Breastcancer	4	4	4	**2**	3	3	4	3	3
BreastEW	6	3	7	**4**	10	10	13	4	9
CongressEW	2	8	3	5	10	6	4	5	**2**
Exactly	**1**	9	9	10	9	10	5	**1**	6
Exactly2	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	6
HeartEW	4	**1**	6	6	9	2	7	9	4
IonosphereEW	7	13	9	**4**	10	12	9	10	8
KrvskpEW	27	15	27	28	26	21	24	25	**11**
Lymphography	11	10	9	7	5	12	**6**	9	7
M-of-n	10	10	10	8	8	8	7	9	**6**
PenglungEW	30	108	61	43	104	32	113	64	**20**
SonarEW	23	13	**9**	34	13	22	20	13	16
SpectEW	4	7	11	6	**1**	13	5	12	7
Tic-tac-toe	6	7	**5**	6	7	6	7	**5**	**5**
Vote	10	2	5	3	2	8	2	8	**1**
WaveformEW	31	27	26	24	30	**17**	30	31	19
WineEW	8	5	4	4	8	6	5	9	**3**
Zoo	5	10	10	**4**	10	7	8	9	**4**

V. COMPARISON

In this section, the proposed X-function has proved its superiority in compared to other transfer functions. Here, we have compared the results obtained by the SMOX algorithm with five state-of-the-art approaches which are widely applied to solve FS problems in the literature: GA, PSO, ALO, GSA, and HS. The values of the control parameters of these five methods are given in Table 6.

Table 7 shows the performance of the SMO algorithm in compared to the mentioned five methods in terms of classification accuracies achieved and number of features selected. SMOX has achieved better classification accuracy than BGA in 15 cases and achieved same accuracy in 3 cases. In compared to BGA, SMOX has selected lower number of features in 6 cases, same number of features 6 cases. SMOX has achieved better classification accuracy than BPSO in 16 cases and achieved same accuracy in 2 cases. Considering selected number of features, SMOX has 9 wins and 3 ties with BPSO. In compared to both BALO and BGSA, SMOX has achieved better accuracy in all the 18 cases. In terms of selected number of features, SMOX has 14 wins and 2 ties with BALO and 10 wins with BGSA. In terms of classification accuracy, SMOX outperforms Binary HS algorithm in 17 cases and achieved same classification accuracy for PenglungEW dataset.

Figure 7, we have provided the average accuracy achieved by SMOX and five state-of-the-art methods considered here. From Figure 7, it can be observed that SMOX has achieved the highest classification accuracy. Figure 8 provides the average number of features selected by SMOX and five state-of-the-art methods. SMOX has selected the lowest number of features of all the methods considered. This proves the

TABLE 5: p-values obtained by the Wilcoxon rank-sum test for 18 UCI datasets using the proposed X-shaped transfer function as compared to 4 S-shaped and 4 V-shaped transfer functions

Dataset	SMOs1	SMOs2	SMOs3	SMOs4	SMOv1	SMOv2	SMOv3	SMOv4
Breastcancer	0.001	0	0.001	0	0	0.002	0	0
BreastEW	0.001	0.001	0.002	0	0.003	0	0	0
CongressEW	0.041	0.038	0.02	0.007	0.017	0.084	0.017	0.01
Exactly	0	0	0	0	0	0	0	0
Exactly2	0.317	0.414	0.317	0.144	1	0.18	0.317	0.068
HeartEW	0	0	0	0	0	0	0	0
IonosphereEW	0	0	0	0	0	0	0	0
KrvskpEW	0	0.013	0	0	0	0.002	0.314	0.014
Lymphography	0	0	0	0	0	0	0	0
M-of-n	0	0	0	0	0	0	0	0
PenglungEW	0.005	0.008	0.025	0.157	0.046	0.046	0.014	0.011
SonarEW	0	0	0	0	0	0	0	0
SpectEW	0.001	0	0	0	0	0	0	0.001
Tic-tac-toe	0	0	0	0	0	0	0	0
Vote	0.002	0.001	0	0.001	0.001	0.001	0.001	0.001
WaveformEW	0.321	0.21	0.001	0	0.092	0	0.011	0
WineEW	0	0.001	0	0	0.001	0	0	0.001
Zoo	0.001	0.001	0.001	0	0	0	0.001	0

TABLE 6: Parameters setting of state-of-the-art methods used for comparison

Algorithm	Parameters
BGA	popSize = 20 maxIter = 30 Mutation and Crossover rate = 0.8
BPSO	popSize = 20 maxIter = 30 inertia factor = 0.1 individual-best acceleration factor = 0.1
BALO	popSize = 20 maxIter = 30
BGSA	popSize = 20 maxIter = 30 G=1 α = 20
BHS	popsize = 20 maxIter = 30 HMCR = 0.8 PAR=0.2

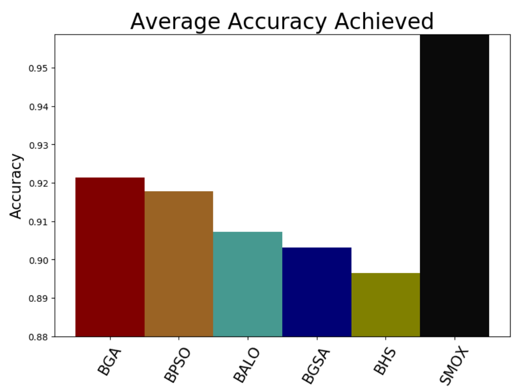


FIGURE 7: Average accuracies achieved by the proposed SMOX algorithm and 5 state-of-the-art methods over 18 UCI datasets

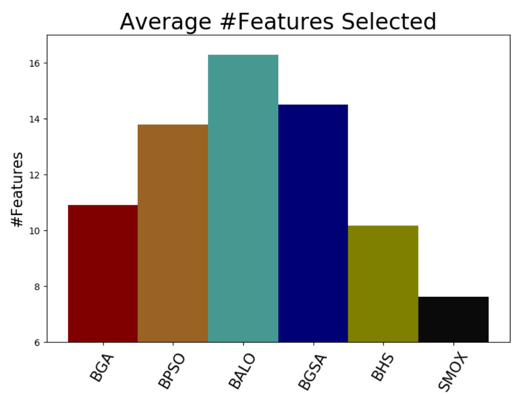


FIGURE 8: Average number of features selected by the proposed SMOX algorithm and 5 state-of-the-art methods over 18 UCI datasets

robustness of the proposed SMOX.

To prove the statistical significance of the results obtained by SMOX in compared to the state of the art methods, we have performed Wilcoxon rank-sum test for each pair of the methods. In Table 8, the p-values obtained for each pair of methods is provided, with $p < 0.05$ marked bold.

VI. CONCLUSION

In this work, we have proposed a new transfer function that inherently utilizes crossover operation which helps the optimization algorithm to properly find any region where the global optima may lie. We have chosen a competent meta-heuristic algorithm SMO, which is proposed recently by following the human behavior of mimicking/copying other more esteemed individuals. This SMO itself requires no parameter tuning, since the agents simply follow the best agent

TABLE 7: Comparison of classification accuracies achieved and number of features selected by the proposed SMOX algorithm with five state-of-the-art methods

Dataset	BGA		BPSO		BALO		BGSA		BHS		SMOX	
	Accuracy	#Features	Accuracy	#Features	Accuracy	#Features	Accuracy	#Features	Accuracy	#Features	Accuracy	#Features
Breastcancer	0.9743	4	0.9629	4	0.974	4.7	0.9686	4	0.9928	5	**1**	**3**
BreastEW	0.9754	8	0.9719	9	0.974	13.85	0.9544	10	0.9561	**6**	**0.9912**	9
CongressEW	0.9679	**2**	0.9633	3	0.981	6.65	0.9633	4	0.9885	4	**1**	**2**
Exactly	1	6	1	6	0.965	5.75	0.994	**4**	0.69	1	**1**	6
Exactly2	0.77	**1**	0.768	**1**	0.762	1.5	0.77	**1**	0.76	**1**	**0.805**	6
HeartEW	0.8741	5	0.837	**3**	0.838	8.6	0.8296	**3**	0.8889	5	**0.9259**	4
IonosphereEW	0.9489	7	0.9489	7	0.904	11.75	0.9432	9	0.9285	**3**	**0.9571**	8
KrvskpEW	**0.985**	11	0.9731	12	0.973	16.15	0.9549	14	0.8529	**10**	**0.985**	11
Lymphography	0.8378	**5**	0.8919	**5**	0.917	7.35	0.8649	6	0.9333	10	**0.9667**	7
M-of-n	**1**	6	**1**	6	0.967	6	0.994	**5**	0.845	6	**1**	6
PenglungEW	0.9189	84	0.9189	130	0.827	133.1	0.8333	140	**1**	77	**1**	**20**
SonarEW	0.9904	19	0.9423	22	0.845	26.6	0.9135	24	0.9523	**16**	**1**	**16**
SpectEW	0.8955	**5**	0.8881	6	0.899	7.65	0.8433	**5**	0.8703	8	**0.963**	7
Tic-tac-toe	0.7996	5	0.7996	6	0.783	5	0.7766	**4**	0.7656	3	**0.82**	5
Vote	0.9733	5	0.96	3	0.972	6.6	0.96	4	0.9833	3	**1**	**1**
WaveformEW	0.7836	15	0.756	15	0.797	20.5	0.7344	**14**	0.806	16	**0.844**	19
WineEW	0.9888	4	0.9775	5	0.972	5.4	0.9775	4	0.9722	**3**	**1**	**3**
Zoo	0.902	**4**	0.9608	5	0.98	5.7	0.9804	6	0.95	6	**1**	**4**

TABLE 8: Pairwise p-values of the Wilcoxon rank-sum test for the classification accuracy of the SMOX and 5 state-of-the-art FS methods considered here

	BGA	BPSO	BALO	BGSA	BHS	SMOX
BGA	-	0.152	0.122	**0.017**	0.446	**0.001**
BPSO	0.152	-	0.616	**0.008**	0.948	**0.000**
BALO	0.122	0.616	-	0.528	0.879	**0.000**
BGSA	**0.017**	**0.008**	0.528	-	0.586	**0.000**
BHS	0.446	0.948	0.879	0.586	-	**0.000**
SMOX	**0.001**	**0.000**	**0.000**	**0.000**	**0.000**	-

found so far. We have compared the effect of the proposed X-shaped transfer function with four S-shaped and four V-shaped transfer functions commonly used in the literature while converting the continuous search space of SMO to a binary one. 18 standard UCI datasets have been considered to assess the performance of these approaches. The comparison clearly displays the superiority of X-shaped transfer function both in terms of achieved classification accuracy and reduction of feature dimension. Hence, it can be concluded that X-shaped transfer function helps SMO to search for the possible region towards global optima. Finally, the proposed FS method, SMOX (SMO with X-shaped transfer function) is compared with five popular state-of-the-art FS methods. The experimental results show that SMOX is able to achieve higher classification accuracy with lower number of features, which in turn, indicates that SMOX is able to effectively search the feature space and find the optimal solution better than other methods. Statistical significance of the obtained results is also tested using Wilcoxon rank-sum test.

As future scope of the work, we can apply the proposed X-shaped transfer function on different state-of-the-art FS methods. We can also apply SMOX on different real world problems, like musical symbol recognition, facial emotion recognition, printed and handwritten script recognition, etc. It would be interesting to investigate the performance of SMOX on high-dimensional datasets such as microarray datasets. Enhanced initialization techniques can be utilized where the algorithm starts with an initial population closer to the global optima. We can hybridize this method with other population based meta-heuristic algorithms.

References

[1] Lu Xiong et al. “Multi-feature fusion and selection method for an improved particle swarm optimization”. In: *Journal of Ambient Intelligence and Humanized Computing* (Dec. 2019). DOI: 10.1007/s12652-019-01624-4.

[2] Hossam Faris et al. “Time-varying hierarchical chains of salps with random weight networks for feature selection”. In: *Expert Systems with Applications* 140 (Feb. 2020), p. 112898. DOI: 10.1016/j.eswa.2019.112898.

[3] Mohamed Bennasar, Yulia Hicks, and Rossitza Setchi. “Feature selection using Joint Mutual Information Maximisation”. In: *Expert Systems with Applications* 42.22 (Dec. 2015), pp. 8520–8532. DOI: 10.1016/j.eswa.2015.07.007.

[4] Heming Jia et al. “Spotted Hyena Optimization Algorithm With Simulated Annealing for Feature Selection”. In: *IEEE Access* 7 (2019), pp. 71943–71962. DOI: 10.1109/access.2019.2919991.

[5] Majdi M. Mafarja and Seyedali Mirjalili. “Hybrid binary ant lion optimizer with rough set and approximate entropy reducts for feature selection”. In: *Soft Computing* 23.15 (June 2018), pp. 6249–6265. DOI: 10.1007/s00500-018-3282-y.

[6] Manosij Ghosh et al. “Genetic algorithm based cancerous gene identification from microarray data using ensemble of filter methods”. In: *Medical & Biological Engineering & Computing* 57.1 (Aug. 2018), pp. 159–176. DOI: 10.1007/s11517-018-1874-4.

[7] Seyedali Mirjalili, Seyed Mohammad Mirjalili, and Andrew Lewis. “Grey Wolf Optimizer”. In: *Advances in Engineering Software* 69 (Mar. 2014), pp. 46–61. DOI: 10.1016/j.advengsoft.2013.12.007.

[8] Dheeru Dua and Casey Graff. *UCI Machine Learning Repository*. 2017. URL: http://archive.ics.uci.edu/ml.
[9] Frank Wilcoxon. "Individual Comparisons by Ranking Methods". In: *Springer Series in Statistics*. Springer New York, 1992, pp. 196–202. DOI: 10.1007/978-1-4612-4380-9_16.
[10] Michel Gendreau and Jean-Yves Potvin. "Metaheuristics in Combinatorial Optimization". In: *Annals of Operations Research* 140.1 (Nov. 2005), pp. 189–213. DOI: 10.1007/s10479-005-3971-7.
[11] Iztok Fister Jr. et al. *A Brief Review of Nature-Inspired Algorithms for Optimization*. 2013. arXiv: 1307.4186 [cs.NE].
[12] Mohamed Abdel-Basset, Laila Abdel-Fatah, and Arun Kumar Sangaiah. "Metaheuristic Algorithms: A Comprehensive Review". In: *Computational Intelligence for Multimedia Big Data on the Cloud with Engineering Applications*. Elsevier, 2018, pp. 185–231. DOI: 10.1016/b978-0-12-813314-9.00010-4.
[13] Amin Foroughi Nematollahi, Abolfazl Rahiminejad, and Behrooz Vahidi. "A novel meta-heuristic optimization method based on golden ratio in nature". In: *Soft Computing* 24.2 (Apr. 2019), pp. 1117–1151. DOI: 10.1007/s00500-019-03949-w.
[14] Lawrence Davis. "Handbook of genetic algorithms". In: (1991).
[15] W.Daniel Hillis. "Co-evolving parasites improve simulated evolution as an optimization procedure". In: *Physica D: Nonlinear Phenomena* 42.1-3 (June 1990), pp. 228–234. DOI: 10.1016/0167-2789(90)90076-2.
[16] Xiaowei Xue, Min Yao, and Ran Cheng. "A Novel Selection Operator of Cultural Algorithm". In: *Advances in Intelligent and Soft Computing*. Springer Berlin Heidelberg, 2011, pp. 71–77. DOI: 10.1007/978-3-642-25661-5_10.
[17] John R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. Cambridge, MA, USA: MIT Press, 1992. ISBN: 0262111705.
[18] Conor Ryan, JJ Collins, and Michael O Neill. "Grammatical evolution: Evolving programs for an arbitrary language". In: *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 1998, pp. 83–96. DOI: 10.1007/bfb0055930.
[19] D. Simon. "Biogeography-Based Optimization". In: *IEEE Transactions on Evolutionary Computation* 12.6 (Dec. 2008), pp. 702–713. DOI: 10.1109/tevc.2008.919004.
[20] Hamid Salimi. "Stochastic Fractal Search: A powerful metaheuristic algorithm". In: *Knowledge-Based Systems* 75 (Feb. 2015), pp. 1–18. DOI: 10.1016/j.knosys.2014.07.025.
[21] J. Kennedy and R. Eberhart. "Particle swarm optimization". In: *Proceedings of ICNN'95 - International Conference on Neural Networks*. Vol. 4. Nov. 1995, 1942–1948 vol.4. DOI: 10.1109/ICNN.1995.488968.
[22] Marco Dorigo, Mauro Birattari, and Thomas Stutzle. "Ant colony optimization". In: *IEEE Computational Intelligence Magazine* 1.4 (Nov. 2006), pp. 28–39. DOI: 10.1109/mci.2006.329691.
[23] Muzaffar Eusuff, Kevin Lansey, and Fayzul Pasha. "Shuffled frog-leaping algorithm: a memetic metaheuristic for discrete optimization". In: *Engineering Optimization* 38.2 (Mar. 2006), pp. 129–154. DOI: 10.1080/03052150500384759.
[24] K. M. Passino. "Biomimicry of bacterial foraging for distributed optimization and control". In: *IEEE Control Systems Magazine* 22.3 (June 2002), pp. 52–67. ISSN: 1941-000X. DOI: 10.1109/MCS.2002.1004010.
[25] Dervis Karaboga and Bahriye Basturk. "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm". In: *Journal of Global Optimization* 39.3 (Apr. 2007), pp. 459–471. DOI: 10.1007/s10898-007-9149-x.
[26] Xin-She Yang. "Firefly Algorithms for Multimodal Optimization". In: *Stochastic Algorithms: Foundations and Applications*. Springer Berlin Heidelberg, 2009, pp. 169–178. DOI: 10.1007/978-3-642-04944-6_14.
[27] Seyedali Mirjalili. "The Ant Lion Optimizer". In: *Advances in Engineering Software* 83 (May 2015), pp. 80–98. DOI: 10.1016/j.advengsoft.2015.01.010.
[28] Seyedali Mirjalili and Andrew Lewis. "The Whale Optimization Algorithm". In: *Advances in Engineering Software* 95 (May 2016), pp. 51–67. DOI: 10.1016/j.advengsoft.2016.01.008.
[29] Shahrzad Saremi, Seyedali Mirjalili, and Andrew Lewis. "Grasshopper Optimisation Algorithm: Theory and application". In: *Advances in Engineering Software* 105 (Mar. 2017), pp. 30–47. DOI: 10.1016/j.advengsoft.2017.01.004.
[30] Mohit Jain, Vijander Singh, and Asha Rani. "A novel nature-inspired algorithm for optimization: Squirrel search algorithm". In: *Swarm and Evolutionary Computation* 44 (Feb. 2019), pp. 148–175. DOI: 10.1016/j.swevo.2018.02.013.
[31] Ali Asghar Heidari et al. "Harris hawks optimization: Algorithm and applications". In: *Future Generation Computer Systems* 97 (Aug. 2019), pp. 849–872. DOI: 10.1016/j.future.2019.02.028.
[32] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. "Optimization by Simulated Annealing". In: *Science* 220.4598 (May 1983), pp. 671–680. DOI: 10.1126/science.220.4598.671.
[33] Peter J. M. van Laarhoven and Emile H. L. Aarts. "Simulated annealing". In: *Simulated Annealing: Theory and Applications*. Springer Netherlands, 1987, pp. 7–15. DOI: 10.1007/978-94-015-7744-1_2.
[34] Esmat Rashedi, Hossein Nezamabadi-pour, and Saeid Saryazdi. "BGSA: binary gravitational search algorithm". In: *Natural Computing* 9.3 (Dec. 2009), pp. 727–745. DOI: 10.1007/s11047-009-9175-3.

[35] Tamás Vicsek et al. "Novel Type of Phase Transition in a System of Self-Driven Particles". In: *Physical Review Letters* 75.6 (Aug. 1995), pp. 1226–1229. DOI: 10.1103/physrevlett.75.1226.

[36] Zong Woo Geem, Joong Hoon Kim, and G.V. Loganathan. "A New Heuristic Optimization Algorithm: Harmony Search". In: *SIMULATION* 76.2 (Feb. 2001), pp. 60–68. DOI: 10.1177/003754970107600201.

[37] Abdolreza Hatamlou. "Black hole: A new heuristic optimization approach for data clustering". In: *Information Sciences* 222 (Feb. 2013), pp. 175–184. DOI: 10.1016/j.ins.2012.08.023.

[38] Seyedali Mirjalili. "SCA: A Sine Cosine Algorithm for solving optimization problems". In: *Knowledge-Based Systems* 96 (Mar. 2016), pp. 120–133. DOI: 10.1016/j.knosys.2015.12.022.

[39] Seyedali Mirjalili, Seyed Mohammad Mirjalili, and Abdolreza Hatamlou. "Multi-Verse Optimizer: a nature-inspired algorithm for global optimization". In: *Neural Computing and Applications* 27.2 (Mar. 2015), pp. 495–513. DOI: 10.1007/s00521-015-1870-7.

[40] Ali Husseinzadeh Kashan, Reza Tavakkoli-Moghaddam, and Mitsuo Gen. "Find-Fix-Finish-Exploit-Analyze (F3EA) meta-heuristic algorithm: An effective algorithm with new evolutionary operators for global optimization". In: *Computers & Industrial Engineering* 128 (Feb. 2019), pp. 192–218. DOI: 10.1016/j.cie.2018.12.033.

[41] R.V. Rao, V.J. Savsani, and D.P. Vakharia. "Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems". In: *Computer-Aided Design* 43.3 (Mar. 2011), pp. 303–315. DOI: 10.1016/j.cad.2010.12.015.

[42] T. Ray and K.M. Liew. "Society and civilization: an optimization algorithm based on the simulation of social behavior". In: *IEEE Transactions on Evolutionary Computation* 7.4 (Aug. 2003), pp. 386–396. DOI: 10.1109/tevc.2003.814902.

[43] Ali Husseinzadeh Kashan. "League Championship Algorithm (LCA): An algorithm for global optimization inspired by sport championships". In: *Applied Soft Computing* 16 (Mar. 2014), pp. 171–200. DOI: 10.1016/j.asoc.2013.12.005.

[44] Ying Tan and Yuanchun Zhu. "Fireworks Algorithm for Optimization". In: *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2010, pp. 355–364. DOI: 10.1007/978-3-642-13495-1_44.

[45] A. Kaveh. "Tug of War Optimization". In: *Advances in Metaheuristic Algorithms for Optimal Design of Structures*. Springer International Publishing, Nov. 2016, pp. 451–487. DOI: 10.1007/978-3-319-46173-1_15.

[46] Reza Moghdani and Khodakaram Salimifard. "Volleyball Premier League Algorithm". In: *Applied Soft Computing* 64 (Mar. 2018), pp. 161–185. DOI: 10.1016/j.asoc.2017.11.043.

[47] J. Kennedy and R. C. Eberhart. "A discrete binary version of the particle swarm algorithm". In: *1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation*. Vol. 5. 1997, 4104–4108 vol.5.

[48] W. Siedledlecki and J. Skansky. "A Note on Genetic Algorithms for Large-Scale Feature Selection". In: *Handbook of Pattern Recognition and Computer Vision*. World Scientific, Aug. 1993, pp. 88–107. DOI: 10.1142/9789814343138_0005.

[49] Hossein Nezamabadi-pour, Majid Rostami-Shahrbabaki, and Malihe Farsangi. "Binary Particle Swarm Optimization: challenges and New Solutions". In: *The Journal of Computer Society of Iran (CSI) On Computer Science and Engineering (JCSE),* 6 (Apr. 2008), pp. 21–32.

[50] Seyedali Mirjalili and Andrew Lewis. "S-shaped versus V-shaped transfer functions for binary Particle Swarm Optimization". In: *Swarm and Evolutionary Computation* 9 (Apr. 2013), pp. 1–14. DOI: 10.1016/j.swevo.2012.09.002.

[51] Majdi Mafarja et al. "S-Shaped vs. V-Shaped Transfer Functions for Ant Lion Optimization Algorithm in Feature Selection Problem". In: *Proceedings of the International Conference on Future Networks and Distributed Systems - ICFNDS '17*. ACM Press, 2017. DOI: 10.1145/3102304.3102325.

[52] Majdi M. Mafarja et al. "Binary Dragonfly Algorithm for Feature Selection". In: *2017 International Conference on New Trends in Computing Sciences (ICTCS)*. IEEE, Oct. 2017. DOI: 10.1109/ictcs.2017.43.

[53] Majdi Mafarja et al. "Binary grasshopper optimisation algorithm approaches for feature selection problems". In: *Expert Systems with Applications* 117 (Mar. 2019), pp. 267–286. DOI: 10.1016/j.eswa.2018.09.015.

[54] Thaer Thaher et al. "Binary Harris Hawks Optimizer for High-Dimensional, Low Sample Size Feature Selection". In: *Algorithms for Intelligent Systems*. Springer Singapore, Nov. 2019, pp. 251–272. DOI: 10.1007/978-981-32-9990-0_12.

[55] Sankalap Arora and Priyanka Anand. "Binary butterfly optimization approaches for feature selection". In: *Expert Systems with Applications* 116 (Feb. 2019), pp. 147–160. DOI: 10.1016/j.eswa.2018.08.051.

[56] Pei Hu, Jeng-Shyang Pan, and Shu-Chuan Chu. "Improved Binary Grey Wolf Optimizer and Its application for feature selection". In: *Knowledge-Based Systems* 195 (May 2020), p. 105746. DOI: 10.1016/j.knosys.2020.105746.

[57] D.H. Wolpert and W.G. Macready. "No free lunch theorems for optimization". In: *IEEE Transactions on Evolutionary Computation* 1.1 (Apr. 1997), pp. 67–82. DOI: 10.1109/4235.585893.

[58] Saeed Balochian and Hossein Baloochian. "Social mimic optimization algorithm and engineering applications". In: *Expert Systems with Applications* 134 (Nov. 2019), pp. 178–191. DOI: 10.1016/j.eswa.2019.05.035.
[59] Majdi M. Mafarja and Seyedali Mirjalili. "Hybrid Whale Optimization Algorithm with simulated annealing for feature selection". In: *Neurocomputing* 260 (Oct. 2017), pp. 302–312. DOI: 10.1016/j.neucom.2017.04.053.
[60] Il-Seok Oh, Jin-Seon Lee, and Byung-Ro Moon. "Hybrid genetic algorithms for feature selection". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26.11 (Nov. 2004), pp. 1424–1437. DOI: 10.1109/tpami.2004.105.
[61] Mohammad Tubishat et al. "Improved Salp Swarm Algorithm based on opposition based learning and novel local search algorithm for feature selection". In: *Expert Systems with Applications* 145 (May 2020), p. 113122. DOI: 10.1016/j.eswa.2019.113122.
[62] Majdi Mafarja and Salwani Abdullah. "Investigating memetic algorithm in solving rough set attribute reduction". In: *International Journal of Computer Applications in Technology* 48.3 (2013), p. 195. DOI: 10.1504/ijcat.2013.056915.
[63] Gilbert Syswerda. "Simulated Crossover in Genetic Algorithms". In: *Foundations of Genetic Algorithms*. Elsevier, 1993, pp. 239–255. DOI: 10.1016/b978-0-08-094832-4.50021-0.
[64] N. S. Altman. "An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression". In: *The American Statistician* 46.3 (Aug. 1992), p. 175. DOI: 10.2307/2685209.
[65] Majdi Mafarja et al. "Efficient Hybrid Nature-Inspired Binary Optimizers for Feature Selection". In: *Cognitive Computation* 12.1 (July 2019), pp. 150–175. DOI: 10.1007/s12559-019-09668-6.
[66] E. Emary, Hossam M. Zawbaa, and Aboul Ella Hassanien. "Binary grey wolf optimization approaches for feature selection". In: *Neurocomputing* 172 (Jan. 2016), pp. 371–381. DOI: 10.1016/j.neucom.2015.06.083.
[67] Guido van Rossum and Fred L. Drake. *The Python Language Reference Manual*. Network Theory Ltd., 2011. ISBN: 1906966141.
[68] J. D. Hunter. "Matplotlib: A 2D graphics environment". In: *Computing in Science & Engineering* 9.3 (2007), pp. 90–95. DOI: 10.1109/MCSE.2007.55.
[69] E. Emary, Hossam M. Zawbaa, and Aboul Ella Hassanien. "Binary ant lion approaches for feature selection". In: *Neurocomputing* 213 (Nov. 2016), pp. 54–65. DOI: 10.1016/j.neucom.2016.03.101.
[70] Joaquín Derrac et al. "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms". In: *Swarm and Evolutionary Computation* 1.1 (2011), pp. 3–18. ISSN: 2210-6502. DOI: https://doi.org/10.1016/j.swevo.2011.02.002.

KUSHAL KANTI GHOSH (email: kushalkanti1999@gmail.com) is currently a senior year undergraduate student of Computer Science and Engineering at Jadavpur University, Kolkata, India. His areas of interest include Machine Learning, Optimization, Game Theory, and Image Processing.

PAWAN KUMAR SINGH (email: pksingh.it@jadavpuruniversity.in) received his B. Tech degree in Information Technology from West Bengal University of Technology in 2010. He received his M. Tech in Computer Science and Engineering and Ph.D. (Engineering) degrees from Jadavpur University (J.U.) in 2013 and 2018 respectively. He also received RUSA 2.0 fellowship for pursuing his post-doctoral research in J.U. in 2019. He is currently working as an Assistant Professor in the Department of Information Technology in J.U. He has published more than 50 research papers in peer-reviewed journals and international conferences. His areas of current research interest are Pattern Recognition, Handwritten Document Analysis, Image Processing, Machine Learning and Artificial Intelligence. He is a member of the IEEE, Computer Society of India (CSI).

JUNHEE HONG (email: hongpa@gachon.ac.kr) received the B.S., M.S., and Ph.D. degrees in electrical engineering from Seoul National University, Seoul, South Korea, in 1987, 1989, and 1995, respectively. Since 1995, he has been a Professor of the Department of Energy IT at Gachon University, Seongnam, South Korea. His research interests include smart grid, supergrid, renewable energy, digitalization of electric power, and energy policy.

ZONG WOO GEEM (email:geem@gachon.ac.kr) is an Associate Professor in Department of Energy IT at Gachon University, South Korea. He has obtained B.Eng. in Chung-Ang University, Ph.D. in Korea University, and M.Sc. in Johns Hopkins University, and researched at Virginia Tech, University of Maryland - College Park, and Johns Hopkins University. He invented a music-inspired optimization algorithm, Harmony Search, which has been applied to various scientific and engineering problems. His research interest includes phenomenon-mimicking algorithms and their applications to energy, environment and water fields. He has served for various journals as an editor (Associate Editor for Engineering Optimization; Guest Editor for Swarm & Evolutionary Computation, Int. Journal of Bio-Inspired Computation, Journal of Applied Mathematics, Applied Sciences, and Sustainability).

RAM SARKAR (email:rsarkar@ieee.org) received his B. Tech degree in Computer Science and Engineering from University of Calcutta in 2003. He received his M.E. degree in Computer Science and Engineering and PhD (Engineering) degree from Jadavpur University in 2005 and 2012 respectively. He joined in the department of Computer Science and Engineering of Jadavpur University as an Assistant Professor in 2008, where he is now working as an Associate Professor. He received Fulbright-Nehru Fellowship (USIEF) for post-doctoral research in University of Maryland, College Park, USA in 2014-15. His areas of current research interest are Image Processing, Pattern Recognition, Machine Learning, and Bioinformatics. He is a senior member of the IEEE, U.S.A.