Classifier Ensemble Reduction Using a Modified Firefly Algorithm: An Empirical Evaluation

Li Zhang[1], Worawut Srisukkham[1], Siew Chin Neoh[2], Chee Peng Lim[3] and Diptangshu Pandit[1]

[1]Computational Intelligence Research Group
Department of Computing Science and Digital Technologies
Faculty of Engineering and Environment
University of Northumbria
Newcastle, NE1 8ST, UK

[2]Faculty of Engineering, Technology and Built Environment
UCSI University
Malaysia

[3]Institute for Intelligent Systems Research and Innovation
Deakin University
Waurn Ponds, VIC 3216, Australia

Email: **{li.zhang; srisukkham.worawut; diptangshu.pandit}@northumbria.ac.uk; u_jane80@yahoo.co.uk; chee.lim@deakin.edu.au**

Abstract. In this research, we propose a variant of the firefly algorithm (FA) for classifier ensemble reduction. It incorporates both accelerated attractiveness and evading strategies to overcome the premature convergence problem of the original FA model. The attractiveness strategy takes not only the neighbouring but also global best solutions into account, in order to guide the firefly swarm to reach the optimal regions with fast convergence while the evading action employs both neighbouring and global worst solutions to drive the search out of gloomy regions. The proposed algorithm is subsequently used to conduct discriminant base classifier selection for generating optimized ensemble classifiers without compromising classification accuracy. Evaluated with standard, shifted, and composite test functions, as well as the Black-Box Optimization Benchmarking test suite and several high dimensional UCI data sets, the empirical results indicate that, based on statistical tests, the proposed FA model outperforms other state-of-the-art FA variants and classical metaheuristic search methods in solving diverse complex unimodal and multimodal optimization and ensemble reduction problems. Moreover, the resulting ensemble classifiers show superior performance in comparison with those of the original, full-sized ensemble models.

Keywords: Ensemble reduction, classification, and firefly algorithm.

I. INTRODUCTION

Ensemble methods have been widely used for improving classification performance. There are many well-known ensemble construction techniques, such as bagging and boosting (Han et al., 2011). However, base classifier redundancy is still a challenging research problem, which has received much research attention in the area of computational intelligence. On the other hand, because of the superior search capabilities of evolutionary algorithms, they have been widely used for solving diverse optimization problems. Many state-of-the-art swarm intelligence algorithms are available in the literature for feature optimization and dimensionality reduction, e.g. see Jothi and Inbarani (2016). Motivated by the success of metaheuristic optimization, we aim to employ evolutionary algorithms for discriminant base model selection and to construct optimized classifier ensembles without compromising classification accuracy.

In this research, we propose a modified firefly algorithm (FA) for classifier ensemble reduction. It integrates two new search mechanisms, i.e. an accelerated attractiveness behaviour and an evading action, to mitigate premature convergence of the original FA model. The proposed attractiveness operation employs not only the neighbouring but also global promising solutions to guide the search process, while the evading action is advised by both local and global worst solutions to lead the search out of gloomy regions. These two new search mechanisms work cooperatively to overcome stagnation, and reach global optimality with fast convergence. The proposed algorithm is useful for identifying discriminant base classifiers to realize ensemble reduction, and obtaining the best trade-off between classification accuracy and ensemble complexity. Evaluated with standard, shifted, and composite test functions, Black-Box Optimization Benchmarking (BBOB) testbeds (Hansen et al., 2010a; 2010b; 2012) and high dimensional UCI data sets (Bache and Lichman, 2013), the proposed algorithm outperforms other state-of-the-art FA variants, BBOB optimizers, and several classical search methods, significantly, in solving diverse challenging unimodal and multimodal optimization and ensemble reduction problems. The resulting classifier ensembles also achieve competitive performance in comparison with those of the full-sized, unreduced original ensembles.

The paper is organized as follows. Section 2 discusses diverse ensemble techniques and swarm intelligence-based optimization methods. We introduce the proposed algorithm with the new attractiveness and evading search behaviours in Section 3. Section 4 presents the evaluation of the proposed algorithm and other metaheuristic search methods using standard and complex benchmark functions and high dimensional data sets for classifier ensemble reduction. Section 5 draws the conclusions and suggests a number of areas for further research.

2. RELATED WORK

In this section, we introduce state-of-the-art ensemble classification techniques and diverse evolutionary metaheuristic search methods.

2.1 Ensemble Classification Techniques

There are many state-of-the-art ensemble methodologies proposed in recent years. Galar et al. (2012) conducted a comprehensive review of ensemble techniques for solving the class imbalance problem. Their studies indicated the efficiency of integrating random under-sampling techniques with bagging and boosting ensembles for classification of imbalanced data sets. Their work emphasized the positive synergy between sampling techniques and bagging ensemble learning. An ordering-based ensemble pruning technique was also proposed in their recent work (Galar et al., 2016) to better tackle imbalanced classification problems. Diao et al. (2014) proposed harmony search based ensemble reduction. They employed bagging and random subspaces for base classifier pool generation. Since bagging uses randomly selected diverse subsets of training instances to build the base models, and the random subspaces method uses randomly selected subsets of attributes for base classifier generation, the base models produced by these two methods showed great diversity. Evaluated with several UCI benchmark data sets, their model achieved impressive accuracy and outperformed ensemble classifiers using the full-sized or any randomly selected base models. Farid et al. (2013) proposed an adaptive ensemble model for data stream classification with concept drifting. They employed three base decision tree classifiers for ensemble construction. The weakest base classifier in the ensemble was updated automatically by a stand-by, newly generated base decision tree learner with the knowledge of the updated class information, in order to represent the most recent concepts in the incoming data streams. Evaluated with UCI benchmark data sets, their model achieved great efficiency for concept-drifting data stream classification. Zhang et al. (2015) developed adaptive ensemble classifiers for facial expression recognition and facial action intensity estimation. After extracting the initial dynamic motion-based facial features, the minimal-redundancy-maximal-relevance criterion (mRMR) was used to identify discriminative facial features for subsequent intensity estimation of facial action units. A set of ensemble classifiers was integrated with a distance-based clustering algorithm to identify six basic emotions as well as new unseen novel emotion classes. The complementary neural network was used as the base learner in their work, which possessed the capability of providing uncertainty measure for classification of each instance. The base classifiers and clustering algorithm worked collaboratively to inform the arrival of novel emotion classes. Evaluated with the Bosphorus facial expression database and with online real user testing, it achieved impressive performance for expression recognition and novel class detection.

Sun et al. (2016) proposed a model known as Class-Based ensemble for Class Evolution (CBCE) to deal with class evolution and concept drifting in data stream mining. Their proposed method generated a base learner for each class. Similarly, it adjusted to the class evolution by constantly updating the base learners to reflect the latest concepts in

the data stream. Moreover, their work addressed a side-effect of class evolution, i.e. the class imbalance problem, by employing an under-sampling method for the base learners. Their model showed superiority over other existing methods for class evolution adaptation. Guan et al. (2015) proposed an ensemble classification method for covariate-invariant gait recognition. They first employed the random subspaces method to generate the base classifiers. They also employed local enhancing and hybrid decision-level fusion to identify more discriminant features and eliminate inefficient base classifiers, respectively. Importantly, their work claimed that feature subsets randomly generated by the random subspaces method were less discriminant, because its feature selection process was conducted in a random manner without using class label information, therefore causing a low recognition accuracy rate with an expensive computational cost. As such, their work employed two local enhancing supervised learning methods, i.e. two-dimensional linear discriminant analysis (LDA) and IDR/QR, to address the above problems and identify more discriminant features. A majority voting strategy was used to generate the final classification result based on the outputs of the base models. Huang et al. (2016) proposed a deep Convolutional Neural Network (CNN) based ensemble classifier for image retrieval. Both AlexNet and Network in Network (NIN) were deployed to extract the initial image features. Subsequently, weighted average feature vectors based on the outputs of both AlexNet and NIN were generated. Evaluated with CIFAR-10 and CIFAR-100 databases, the proposed aggregate ensemble model outperformed single CNN for image classification and retrieval tasks. Liew et al. (2016) constructed an ensemble reduction method by using the Genetic Algorithm (GA) and a Bayesian Information Criterion fitness evaluation mechanism. Fuzzy extreme learning machines were used as the base learners. Evaluated with the database for Emotion Analysis using Physiological Signals, their model achieved the best trade-off between classification accuracy and ensemble complexity. Sołtys et al. (2015) utilized ensemble models such as bagging and random forests for uplift modeling, whereas Pietruczuk et al. (2017) proposed two theorems for determining the optimal ensemble size for data stream classification based on both classification accuracy and memory requirements.

2.2 FA and Modified FA Models

Swarm intelligence-based algorithms have been extensively studied for diverse optimization problems. As one of the more recently proposed swarm intelligence-based metaheuristic search methods, FA shows natural search capability of dealing with multimodal optimization problems as compared with other algorithms such as GA and Particle Swarm Optimization (PSO) (Yang, 2009). FA employs the following principles to perform the search process. Firstly, each firefly has a light intensity which denotes the solution quality. Secondly, fireflies with lower light intensities are attracted to neighbouring fireflies with higher light intensities regardless of their sex. Finally, the attractiveness decreases as the distance between two fireflies increases (Yang, 2009). A Levy-flight Firefly Algorithm (LFA) has also been proposed by Yang (2010). Instead of using Gaussian distribution, LFA implements Levy flights as random walk to overcome local optima traps.

In LFA, the attractiveness behaviour defined in Equation (1) is used to guide a firefly with lower light intensity to move towards the brighter ones in the neighbourhood.

$$x_i = x_i + \beta_0 e^{-\gamma r_{ij}^2}\left(x_j - x_i\right) + \alpha \,\text{sign}\left[\text{rand} - \frac{1}{2}\right] \oplus \text{Levy} \tag{1}$$

where x_i and x_j denote the positions of fireflies *i* and *j*, respectively, and γ is the fixed light absorption coefficient. Note that β_0 represents the initial attractiveness measure when $r = 0$. A Levy distribution is used as random walk with α as the randomized step parameter. Here, r_{ij} represents the Euclidean distance between fireflies *i* and *j* as defined in Equation (2).

$$r_{ij} = \left\|x_i - x_j\right\| = \sqrt{\sum_{k=1}^{d}(x_{i,k} - x_{j,k})^2} \tag{2}$$

where $x_{i,k}$ represents the *k*-th dimension of position x_i for the *i*-th firefly, and d indicates the dimension of a given problem.

A comprehensive review of FA was presented in Fister et al. (2013), where many modified or hybrid FA models were described. Besides that, an elitist strategy with the generation of m-uniform random vectors was used to improve the random movement of the global best solution in Tilahun and Ong (2012). Gaussian and Levy distributions were used as popular strategies for random walks in modified FA models (Yang, 2009; Fister et al., 2013; Chou and Ngo, 2017). Since the variation of light intensity, and the attraction and randomization parameters play important roles in affecting the performance of FA, chaotic maps are utilized for formulating these parameters

in modified FA models to enhance performance. For instance, the randomization parameter and light absorption coefficient were adjusted by chaotic maps in Coelho et al. (2011), while 12 different chaotic maps were explored for tuning the attraction and light absorption coefficients in Gandomi et al. (2013). The findings in Gandomi et al. (2013) indicated that the modified FA models using Gauss and Sinusoidal maps as the attractiveness and light absorption coefficients, respectively, achieved the best performance for solving unimodal and multimodal benchmark functions. Many meta-heuristic or learning/search mechanisms, such as GA, PSO, DE, Ant-Lion Optimization (ALO), memetic algorithm, cellular learning automata and neural networks, have been hybridized with FA to increase search diversity (Fister et al., 2013). These modified and hybrid FA methods have been used widely to solve continuous, combinatorial, constrained, multi-objective and dynamic optimization problems. Fister Jr. et al. (2015) conducted a comprehensive review of chaos-based FA models. It included theoretical introduction of several popular chaotic maps, including Logistic, Gauss, Kent, Sine and Iterative maps. These maps have been extensively used to enhance and adjust several key parameters of FA, which showed great efficiency in solving global and engineering optimization problems. Yang (2009) explored the effectiveness of FA in dealing with multimodal optimization problems. Evaluated with several popular multimodal benchmarks, FA showed great superiority over PSO and GA.

A number of FA variants have also been proposed in recent years to overcome the local optimum traps of the original FA model. These variants are used for solving dimensionality reduction, optimal parameter selection, combinatorial and dynamic optimization problems. As an example, Baykasoğlu and Ozsoydan (2014) proposed an FA variant, which incorporated partial random restarts and an adaptive move procedure, to deal with dynamic multidimensional knapsack problems. Specifically, besides the original attractiveness factor, a probability-based mechanism was formulated, which was based on the iteration counter of the algorithm and frequency in dynamic environment to control the movement of a firefly. Their proposed FA variant also employed a modified attractiveness parameter for position updating. The model was extended in Baykasoğlu and Ozsoydan (2015) to deal with constrained non-linear design optimization problems. The extended model also employed an adaptive attractiveness coefficient to achieve a higher intensification capability. Both randomization and incumbent local search mechanisms along with corresponding control parameters were used to overcome local optima traps and enhance the global best solution, respectively. A probabilistic combination of three chaotic maps, i.e. Sinusoidal, Logistic and Tent maps, was used to further improve performance. Their model showed great superiority in tackling diverse engineering design problems. Chou and Ngo (2017) developed another modified FA model for solving multidimensional structural design optimization problems. Their model employed the Logistic map and the Gauss/mouse maps for population initialization and fine-tuning of the attractiveness coefficient, respectively. An adaptive inertia weight and Lévy flight were used for the position updating mechanism to increase search diversity. Wang et al. (2017) proposed an FA model with neighbourhood attraction, known as NaFA. In NaFA, each firefly was purely attracted to brighter fireflies in a pre-defined neighbourhood, instead of the whole population, to reduce computational complexity. A hybrid firefly differential evolution (HFDE) model was developed by Dash et al. (2017). HFDE combined improved DE (IDE) with FA to increase global exploration capabilities of IDE. In HFDE, IDE employed an adaptive weighting factor, as opposed to a fixed one in conventional DE, to overcome local optima. The best three individuals identified by IDE were subsequently used to generate offspring using FA movements. These offspring were used to replace the three worst solutions in the swarm if they had better fitness scores. The empirical results indicated that HFDE benefited from both DE and FA search mechanisms, and outperformed other optimal design methods.

Jothi and Inbarani (2016) proposed a hybrid supervised feature selection method, known as Tolerance Rough Set Firefly based Quick Reduct (TRSFFQR), by combining the Tolerance Rough Set (TRS) with FA for brain tumour image classification. The shape, intensity and texture features in real-value were firstly extracted from segmented MRI images. Then, the proposed TRSFFQR model was applied to identify the most significant feature subsets for tumour classification. In comparison with Artificial Bee Colony (ABC), Cuckoo Search (CS), Supervised Tolerance Rough Set-PSO based Relative Reduct (STRSPSO-RR) and Supervised Tolerance Rough Set-PSO based Quick Reduct (STRSPSO-QR), the proposed algorithm showed significant performance improvements. Kazem et al. (2013) proposed a stock market price forecasting model by integrating chaotic maps with FA for Support Vector Regression (SVR) hyper-parameter selection. Their work re-constructed the phase space dynamics using a delay coordinate embedding method. Then, FA integrated with chaos theory, known as the chaotic FA (CFA) model, was used to fine-tune SVR hyper-parameters. The proposed model achieved impressive performance enhancement in comparison with those of chaotic GA-based SVR, FA-based SVR and neural networks (NN), based on several challenging stock market prediction data sets. Su et al. (2017) proposed a hyperspectral image classification system

using FA-based band selection and optimal parameter identification for an extreme learning machine. FA was employed not only to identify an optimal subset of bands and reduce the network complexity, but also to identify the optimal parameter settings such as the regularization coefficient, Gaussian kernel and the hidden number of neurons. Their model outperformed PSO-based and other band selection methods. A hybrid FA model was developed by Zhang et al. (2016a) by integrating FA with Differential Evolution (DE) for solving diverse unimodal and multimodal optimization problems. Their model combined the exploration (i.e. diversification) capability of FA and the exploitation (i.e. intensification) capability of DE to guide the search process. The proposed algorithm divided the overall population into two sub-swarms, and FA and DE were used to lead the search in each sub-swarm, respectively. The sub-swarms were merged subsequently in each iteration with the global best solution identified. Tested with several unimodal and multimodal benchmark functions, their FA variant outperformed FA, DE and PSO. Long et al. (2015) developed rough sets based feature optimization and an interval type-2 fuzzy logic system (IT2FLS) for heart disease detection. In their work, FA and rough sets were used to perform feature optimization in high dimensional disease data. Alweshah and Abdullah (2015) proposed two modified FA models, i.e. SFA and LSFA, where simulated annealing (SA) was used to enhance the global best solutions obtained by FA and LFA, respectively. The two FA variants were used to improve the performance of a probabilistic neural network by adjusting its weights. Evaluated with several real-life benchmark data sets, LSFA outperformed SFA and the original FA model. Krawczyk (2015) proposed an FA-based ensemble reduction technique for one-class classification problems. FA was used as a clustering method to group fireflies into different clusters. The brightest firefly among each cluster was selected as an ensemble member to represent its group, therefore reducing the size of the base models. It also employed the diversity measurement as the distance measurement between the fireflies. Subsequently the weight of each selected representative classifier was generated by averaging the intensities of all fireflies in the corresponding cluster. A weighted majority voting method was used to determine the final classification output. There are three major differences between the work in Krawczyk (2015) and our proposed research, as follows. (1) The model in Krawczyk (2015) focuses on solving one-class problems while our proposed model tackles both binary and multi-class problems; (2) instead of using clustering-based pruning strategies where each firefly represents one base classifier as in Krawczyk (2015), our research is motivated by feature selection methods where each firefly represents all the base models with each element/dimension denoting a classifier; (3) except for a new distance calculation strategy, the model in Krawczyk (2015) employs the original FA attraction operation where the search is purely led by the neighbouring brighter fireflies. When this attraction-based search action stagnates, there is no mechanism in the model proposed by Krawczyk (2015) to drive the search out of local optima. On the contrary, two new search operations, i.e. attraction and evading mechanisms, are introduced in our research. These two search mechanisms not only employ both local and global promising or worst signals to guide the search process, but also work in a collaborative manner to overcome stagnation and mitigate premature convergence of the original FA model.

FA variants have been adopted to solve other real-life optimization problems. A moth-firefly algorithm was developed by Zhang et al. (2016b) for dimensionality reduction in facial expression classification, whereas Srivatsava et al. (2013) employed FA for optimal test path generation in software testing. A discrete FA (DFA) model was proposed by Sayadi et al. (2013) for solving discrete optimization problems. Xu and Liu (2013) proposed a multi-population FA (MFA) model for correlated data routing in underwater wireless sensor networks. Three types of fireflies were introduced, i.e. searching, listening, and updating fireflies, to improve adaptability of building, selecting, and optimizing routing paths. The searching fireflies identified a routing path from the source node to the sink. The listening fireflies collected information of mobile data stored in the searching fireflies to advise the routing path. The updating fireflies updated the light intensity of nodes along the routing path. MFA was also used to remove redundant information when merging the correlated data packets to improve efficiency. It outperformed existing protocols significantly. Ozsoydan and Baykasoglu (2015) proposed another multi-population FA with chaotic maps to solve dynamic optimization problems. An explorer swarm was used for the search of existing peaks, while colony swarms were generated to conduct local exploitation of the detected peaks. An exclusion strategy was proposed to remove non-contributing sub-populations by comparing the distance between a pair of the best solutions from any two sub-swarms with a threshold value. In order to increase swarm diversity, Chebyshev, ICMIC, Tent and Logistic maps were used for generating explorer populations, respectively. Their model showed impressive performance especially when dealing with complex instances for the moving peaks benchmark problem. Zhou et al. (2014) proposed a multi-population discrete FA model incorporated with the k-opt algorithm for solving traveling salesman problem. Variants for other metaheuristic search methods, such as PSO, have also been proposed. As an example, an enhanced leader PSO (ELPSO) was proposed in Jordehi (2015), which employed a series of mutation strategies to improve the global best solution identified by PSO. A micro GA embedded PSO model was developed

in Mistry et al. (2016) for discriminative facial feature selection and expression recognition. It incorporated a micro GA motivated small-population secondary swarm, a modified velocity updating strategy, and a sub-dimension based search mechanism to mitigate premature convergence of the original PSO algorithm.

3. THE PROPOSED FA VARIANT

In this research, we propose a modified FA model for classifier ensemble reduction. It mitigates the premature convergence problem of the original FA model by embedding two new search strategies, i.e. accelerated attractiveness and evading operations. The attractiveness behaviour is guided by both neighbouring and global promising solutions to reach local and global optimality with fast convergence, while the enemy evading action is advised by the neighbouring and global worst solutions to avoid gloomy search regions. We introduce these two new search mechanisms in detail, as follows.

FA possesses an original attractiveness behaviour, as in Equation (1), which is guided purely by the neighbouring promising solutions to move the current firefly with a lower light intensity towards brighter ones in the neighbourhood. In comparison with this original attractiveness movement, the newly proposed attractiveness operation takes not only brighter fireflies in the neighbourhood but also the current global best solution, g_{best}, into account for position updating. This accelerated attractiveness behaviour is defined in Equation (3).

$$\begin{aligned} x_i &= x_i + \beta_0 e^{-\gamma r_{ij}^2}(x_j - x_i) + \alpha' \varepsilon\, (g_{best} - x_i) + \alpha' \text{sign}[\text{rand} - \tfrac{1}{2}] \oplus \text{Levy} \\ &= x_i(1 - \beta_0 e^{-\gamma r_{ij}^2}) + x_j\, \beta_0 e^{-\gamma r_{ij}^2} + \alpha' \varepsilon\, (g_{best} - x_i) + \alpha' \text{sign}[\text{rand} - \tfrac{1}{2}] \oplus \text{Levy} \end{aligned} \quad (3)$$

where α' is an adaptive parameter defined in Equation (4), and ε is a randomly generated vector with each element in the range of [0, 1].

$$\alpha' = \alpha' \times \left(\frac{10^{-4}}{0.9}\right)^{\frac{1}{nGen}} \quad (4)$$

where $nGen$ denotes the maximum number of iterations and α' is initialized to 0.5 according to Yang (2009). This strategy is motivated by a MATLAB version of the original FA model by Yang (2009). In this way, α' is tuned throughout generations. It starts with a larger value to increase diversity of the solution vectors, and decreases to a smaller value in subsequent iterations to enable fine-tuning of the solution vectors. This strategy is not included in the original FA model, but employed in our proposed algorithm to accelerate convergence. Owing to the fact that position updating of each firefly is conducted by using both neighbouring and global optimal solutions simultaneously, this new search mechanism enables the fireflies to reach the optimal regions more efficiently with fast convergence.

In addition to the abovementioned attractiveness behaviour, we propose an evading search mechanism defined in Equation (5). This evading movement is activated when the neighbouring fireflies have lower fitness values than that of the current firefly. It leads the current firefly *i* to move away from both the neighbouring firefly *j* with a lower light intensity and the current global worst solution, g_{worst}, simultaneously.

$$x_i = x_i \left(1 - \beta_0' e^{-\gamma r_{ij}^2}\right) - x_j \beta_0' e^{-\gamma r_{ij}^2} - \alpha' \varepsilon\, (g_{worst} - x_i) - \alpha' \text{sign}[\text{rand} - \tfrac{1}{2}] \oplus \text{Levy} \quad (5)$$

where α' is the adaptive parameter defined in Equation (4) and β_0' represents the evading coefficient, $0 \leq \beta_0' \leq 1$, to tune the effects of the evading action from the neighbouring worst solutions. In this research, we assign β_0'=0.2 based on trial and errors in a series of experiments.

As mentioned earlier, the search process of the original FA model is based on the sole dominating attractiveness search action to move the current firefly *i* towards a brighter neighbouring firefly *j*. There is no search mechanism to deal with the situation when the current firefly *i* is brighter than the neighbouring firefly *j*. In the original FA model, the brightest firefly in the neighbourhood tends to merely take a random walk action. The proposed evading behaviour goes beyond the limitations of the search behaviours of the original FA model, and guides the current promising firefly *i* to move away from both local and global gloomy search regions, in order to diversify the search

and accelerate convergence.

Furthermore, the proposed attractiveness and evading operations work cooperatively to accelerate the search process and reduce the probability of premature convergence. These two newly proposed search strategies enable the swarm to explore wider and more distinctive search regions in comparison with those of the original FA model owing to the diversified position updating mechanisms. Therefore, the proposed FA model has better capability of finding global optimum solutions and escaping from local optima traps. These proposed search mechanisms also enable each firefly to follow multiple local and global promising solutions to enhance the natural multimodal search capabilities of the original FA model. The empirical results indicate that it shows great efficiency and robustness in dealing with diverse unimodal and multimodal optimization problems. In comparison with other FA variants and classical search methods, it shows superior discriminating capabilities over other methods for discriminant base model selection. The proposed algorithm is illustrated in Algorithm 1.

Algorithm 1: Pseudo-Code of the Proposed FA Model

```
1.  Start
2.  Initialize a population of fireflies randomly;
3.  Set the light absorption coefficient and other search parameters;
4.
5.  While (the stopping criterion is not satisfied)// until it finds the optimal solution or the
    maximum number of iterations is reached.
6.  {
7.     Evaluate the population and update the light intensity;
8.     Rank the fireflies and find the current global best and worst solutions, i.e. g_best and
       and g_worst, respectively;
9.     For i = 1 to n do //external loop
10.     {
11.          For j = 1 to n do  //internal loop
12.          {
13.            If (I_j > I_i)
14.            {
15.              //Using the attractiveness operation to guide the search
16.              Move firefly i towards firefly j and the global best solution, g_best, using
                 Equation (3);
17.            }
18.           Else
19.            {
20.              //Using the evading action to guide the search
21.              Move firefly i away from firefly j and the global worst solution, g_worst,
                 using Equation (5);
22.            }
23.           End If
24.          } End For
25.     } End For
26. }End While
27. Output the most optimal solution(s);
28. End
```

As indicated in Algorithm 1, after initializing the swarm, the fitness of each firefly is evaluated using the objective function. Subsequently, the fireflies are ranked based on their fitness values with the global best and worst solutions identified. For each current firefly *i*, when the neighbouring firefly *j* has a better fitness than that of the current firefly *i*, the newly proposed attractiveness movement defined in Equation (3) is used to guide firefly *i* to move towards both the promising neighbouring firefly *j*, and the current global best solution, g_{best}, simultaneously. On the other hand, when the neighbouring firefly *j* is less optimal than the current firefly *i*, the evading behaviour defined in Equation (5) is activated to move the current firefly *i* away not only from the neighbouring worse solution *j*, but also from the current global worst solution, g_{worst}. Both proposed attractiveness and evading behaviours work in a collaborative manner to avoid local optimum traps and move towards the global optima.

The objective function in Equation (6) is used to evaluate the fitness of each solution (i.e. each set of recommended base models) for ensemble generation, which is widely employed for feature selection and dimensionality reduction (Mistry et al., 2016; Zhang et al., 2016b). This fitness evaluation considers two criteria, i.e. classification accuracy and the number of selected base classifiers, as follows.

$$fitness_x = w_a * accuracy_x + w_b * (number\_base\_models_x)^{-1} \quad (6)$$

where w_a and w_b denote the weights for classification accuracy and the number of selected base models, respectively, and $w_a + w_b = 1$. Since classification accuracy is generally more important than the number of selected base models, w_a is higher than w_b. The termination criteria are as follows, i.e. either (1) the maximum number of iterations is reached, or (2) the optimal solution is found. An evaluation is conducted to assess efficiency of the proposed FA model and other state-of-the-art and classical search methods using diverse benchmark test functions and high dimensional data sets. Detailed evaluation and analysis of the results are provided in Section 4.

4. EVALUATION

To evaluate the proposed FA variant, we have implemented a number of state-of-the-art FA and PSO variants and conventional search methods for performance comparison. Firstly, we employ mathematical benchmark functions and BBOB testbeds to evaluate the efficiency and robustness of the proposed FA model for solving diverse unimodal and multimodal optimization problems. Secondly, several UCI benchmark and other medical image data sets are used for evaluation of ensemble reduction and aggregation. These test data sets are selected because their samples are large in size and high in dimension. Since the proposed FA model and other search algorithms are stochastic methods, a benchmark of 30 runs has been conducted to evaluate each test function and each test data set.

4.1 Evaluation Using Standard Benchmark Functions

Firstly, we employ 10 standard benchmark functions for evaluation. These test functions have been widely used for the evaluation of swarm intelligence-based algorithms (Jordehi, 2015). They are: Ackley, Dixon-Price, Griewank, Levy, Rastrigin, Rotated Hyper-Ellipsoid, Rosenbrock, Sphere, Sum of Different Powers, and Zakharov. The detailed descriptions of these test functions are available in (Jordehi, 2015). Moreover, these functions represent diverse unimodal and multimodal optimization problems. The Dixon-Price, Rotated Hyper-Ellipsoid, Rosenbrock, Sphere, Sum of Different Powers and Zakharov functions have single global minima, while others such as Ackley, Griewank, Levy, and Rastrigin contain multiple global minima. The optimization task is to find the global minima (i.e. zero) in the landscapes defined by the test functions. The FA variants implemented for performance comparison include Opposition and Dimensional based modified FA (ODFA) (Verma et al., 2016), CFA (Kazem et al., 2013), LSFA (Alweshah and Abdullah, 2015), SFA (Alweshah and Abdullah, 2015), NaFA (Wang et al., 2017), and HFDE (Dash et al., 2017). The classical search methods included for comparison are FA, SA, Bat Swarm Optimization (BSO), CS, PSO, Dragonfly Algorithm (DA) (Mirjalili, 2016a) and ALO (Mirjalili, 2015a).

The following settings, i.e. dimension=50 and population size=20, have been used for experimental studies. The classical search methods, CFA, NaFA and the proposed FA model, employ the following number of function evaluations, i.e. 20 (population size) × 100 (maximum number of generations) = 2000. Other FA variants, i.e. LSFA, SFA, ODFA and HFDE, utilize comparatively higher numbers of function evaluations because of their internal search mechanisms. The number of function evaluations for LSFA and SFA is the sum of the cost for both LFA/FA and SA, i.e. 20 (population size) × 100 (maximum number of generations) + 20 (maximum trials of SA) × 100 (maximum number of generations) = 4000. ODFA employs a dimensional approach to retrieve the global best solution and has the following number of function evaluations, i.e. 20 (population size) × 100 (maximum number of generations) × 50 (dimension) = 100000. The number of function evaluations for HFDE is the sum of the cost for both DE (i.e. the fitness evaluation for both parent and offspring populations) and FA, i.e. (2 × 20 (population size) + 3 offspring generated using FA) × 100 (maximum number of generations) = 4300. For each test function, we conduct 30 runs using each method. The mean, minimum, maximum and standard deviation values for each test method are tabulated in Table 1. The results in bold indicate the best performances across all methods.

The parameters of each algorithm for this and subsequent experiments are set as follows. According to Yang (2009; 2010), the FA model employs the following setting to balance between computational efficiency and performance, i.e. initial attractiveness=1.0, randomization parameter=0.5, absorption coefficient=1.0, and Levy's index=1.5.

According to the empirical studies in Yang (2008), the SA parameters that achieve the best trade-off between local exploitation and global exploration are as follows: cooling factor=0.95, initial temperature=1.0, and finial stopping temperature=1e-10. These parameters of FA and SA have been used for the experimental settings of LSFA and SFA. NaFA and HFDE also employ the abovementioned setting of FA. In addition, NaFA utilizes a neighbourhood size of 3 (as recommended in Wang et al. (2017)) while HFDE uses a new coefficient of 1.8 for weighting factor calculation (as recommended in Dash et al. (2017)). We have also conducted a number of trials to confirm the efficiency of these parameter settings of NaFA and HFDE for diverse optimization tasks. Other FA variants, i.e. ODFA, CFA and the proposed FA model, also utilize the abovementioned settings of FA for subsequent experimental studies with an extra evading coefficient=0.2 for the proposed FA model.

The following optimal settings recommended by theoretical studies and empirical evaluations (Yang, 2008) have been used for the GA, i.e. crossover probability=0.6, and mutation probability=0.05. In accordance with Yang (2008), PSO employs the following parameter settings: maximum velocity=0.6, inertia weight=0.5, and acceleration constants c_1=c_2=1.5. Note that the same PSO settings have been utilized in ELPSO (Jordehi, 2015). DA employs the following parameter settings: separation factor=0.1, alignment factor=0.1, cohesion factor=0.7, food factor=1, enemy factor=1, and inertia weight=0.9 – m × ((0.9-0.4)/maxi_iterations), where m and maxi_iterations represent the current and maximum iteration numbers, respectively (Mirjalili, 2016a). ALO provided by Mirjalili (2015a) uses adaptive parameters associated with the number of iterations without any additional parameters required. Finally, CS uses the discovery rate of alien eggs/solutions=0.25, while BSO employs 0.5 for both loudness parameter and pulse rate (Yang, 2008).

Table 1 Evaluation results for 10 benchmark functions with dimension=50

		Prop. FA	NaFA	HFDE	ODFA	SFA	LSFA	CFA	FA	SA	BSO	CS	PSO	DA	ALO
Ackley	**mean**	**6.57E-03**	1.23E+01	1.29E+01	1.68E+01	9.31E+00	1.91E+01	1.86E+01	8.04E+00	1.20E+01	1.84E+01	1.64E+01	1.44E+01	1.95E+01	1.90E+01
	min	**5.62E-03**	1.13E+01	1.12E+01	1.47E+01	6.27E+00	1.75E+01	1.81E+01	5.49E+00	3.57E+00	1.75E+01	1.44E+01	1.24E+01	1.53E+01	1.90E+01
	max	**7.29E-03**	1.45E+01	1.46E+01	1.80E+01	1.18E+01	2.00E+01	1.92E+01	1.03E+01	2.02E+01	1.94E+01	1.73E+01	1.59E+01	1.99E+01	1.90E+01
	std	4.33E-04	7.79E-01	9.21E-01	8.91E-01	1.43E+00	8.63E-01	2.24E-01	1.41E+00	7.42E+00	5.72E-01	6.33E-01	9.50E-01	8.52E-01	**1.45E-14**
Dixon-Price	**mean**	**9.97E-01**	1.25E+04	2.49E+03	1.84E+06	3.43E+00	3.67E+04	3.82E+06	4.12E+03	4.19E+06	1.13E+06	7.37E+04	5.69E+03	1.71E+04	6.09E+06
	min	9.95E-01	5.21E+03	4.06E+02	7.67E+05	**6.69E-01**	1.10E+04	2.05E+06	1.22E+03	2.92E+06	3.55E+05	2.28E+04	1.39E+03	7.45E+01	4.33E+06
	max	**9.98E-01**	2.38E+04	1.01E+04	3.34E+06	4.55E+01	8.06E+04	4.88E+06	1.38E+04	5.21E+06	4.06E+06	2.28E+05	1.93E+04	1.93E+05	8.19E+06
	std	**7.41E-04**	5.19E+03	2.05E+03	6.35E+05	8.78E+00	1.84E+04	6.39E+05	2.80E+03	5.81E+05	7.07E+05	3.92E+04	3.97E+03	3.99E+04	1.08E+06
Griewank	**mean**	**2.61E-03**	3.43E+01	2.00E+01	5.76E+02	1.21E+01	1.13E+03	8.25E+02	1.08E+01	9.27E+02	4.21E+02	1.08E+02	2.57E+01	2.93E+01	1.14E+03
	min	**1.71E-03**	1.35E+01	8.52E+00	3.56E+02	3.43E+00	9.00E+02	6.95E+02	3.10E+00	7.62E+02	2.41E+02	6.40E+01	1.20E+01	1.45E+00	9.40E+02
	max	**4.08E-03**	4.88E+01	5.15E+01	7.48E+02	2.64E+01	1.27E+03	9.44E+02	2.17E+01	1.08E+03	6.44E+02	1.41E+02	4.67E+01	8.64E+01	1.32E+03
	std	**5.85E-04**	9.36E+00	9.07E+00	1.11E+02	5.86E+00	9.23E+01	6.04E+01	4.81E+00	6.47E+01	1.18E+02	1.91E+01	7.77E+00	2.58E+01	9.49E+01
Levy	**mean**	**5.07E+00**	2.12E+02	2.17E+02	5.40E+02	1.04E+02	5.56E+02	1.75E+03	1.16E+02	1.89E+03	1.31E+03	7.40E+02	2.45E+02	8.05E+02	2.68E+03
	min	**5.07E+00**	7.93E+01	1.07E+02	2.73E+02	1.53E+01	3.87E+02	1.37E+03	5.56E+01	1.58E+03	6.35E+02	5.92E+02	1.17E+02	2.70E+02	1.89E+03
	max	**5.07E+00**	4.36E+02	5.79E+02	8.82E+02	2.71E+02	9.67E+02	2.05E+03	3.41E+02	2.14E+03	2.43E+03	9.92E+02	4.71E+02	1.75E+03	3.37E+03
	std	**6.15E-04**	7.04E+01	9.56E+01	1.81E+02	5.02E+01	1.27E+02	1.67E+02	5.49E+01	1.55E+02	3.56E+02	1.08E+02	6.95E+01	3.47E+02	3.99E+02
Rastrigin	**mean**	**1.37E-03**	2.46E+02	2.47E+02	4.74E+02	2.78E+02	3.68E+02	6.69E+02	2.75E+02	6.87E+02	4.83E+02	4.25E+02	1.83E+02	3.19E+02	7.99E+02
	min	**9.66E-04**	1.75E+02	1.27E+02	3.05E+02	1.92E+02	2.47E+02	5.98E+02	1.98E+02	5.20E+02	3.24E+02	3.73E+02	1.31E+02	1.50E+02	7.41E+02
	max	**1.67E-03**	3.62E+02	4.53E+02	6.00E+02	4.02E+02	4.70E+02	7.14E+02	3.56E+02	7.46E+02	5.99E+02	4.69E+02	2.38E+02	5.09E+02	8.74E+02
	std	**1.53E-04**	3.96E+01	1.20E+02	6.53E+01	5.31E+01	4.40E+01	2.89E+01	3.72E+01	4.14E+01	5.94E+01	1.95E+01	2.84E+01	8.71E+01	3.26E+01
Rotated Hyper-Ellipsoid	**mean**	**2.55E-02**	3.75E+04	1.80E+04	7.63E+05	7.07E+03	9.73E+05	9.10E+05	1.69E+04	1.04E+06	4.98E+05	1.21E+05	2.77E+04	3.56E+04	1.39E+06
	min	**1.83E-02**	1.34E+04	7.72E+03	4.68E+05	7.58E+02	7.96E+05	7.51E+05	7.12E+03	9.39E+05	2.35E+05	7.82E+04	1.26E+04	7.50E+01	1.17E+06
	max	**3.30E-02**	7.37E+04	3.24E+04	1.07E+06	1.58E+04	1.19E+06	1.11E+06	3.43E+04	1.17E+06	1.02E+06	1.88E+05	5.67E+04	1.10E+05	1.62E+06
	std	**3.80E-03**	1.28E+04	7.68E+03	1.24E+05	4.06E+03	8.57E+04	8.73E+04	6.36E+03	6.41E+04	1.79E+05	2.64E+04	1.19E+04	2.80E+04	1.14E+05
Rosenbrock	**mean**	**4.90E+01**	6.51E+10	1.36E+10	4.54E+05	5.11E+01	1.45E+04	1.16E+06	5.82E+03	2.07E+06	9.47E+05	1.49E+05	1.91E+04	1.81E+05	3.74E+06
	min	4.90E+01	7.48E+09	2.23E+09	8.20E+04	**7.07E-02**	7.28E+02	8.37E+05	1.20E+03	8.89E+02	4.35E+05	7.00E+04	1.72E+03	4.58E+04	2.63E+06
	max	**4.90E+01**	1.91E+11	3.31E+10	1.02E+06	1.48E+02	3.52E+04	1.41E+06	1.53E+04	2.84E+06	1.81E+06	2.82E+05	7.61E+04	5.54E+05	4.41E+06
	std	**4.90E-03**	4.41E+10	9.57E+09	2.04E+05	4.43E+01	9.91E+03	1.33E+05	3.47E+03	6.15E+05	3.79E+05	4.47E+04	2.21E+04	1.31E+05	5.05E+05
Sphere	**mean**	**6.59E-06**	9.04E+00	4.97E+00	2.03E+02	6.72E-04	6.10E-02	2.43E+02	4.36E+00	2.70E+02	9.98E+01	3.19E+01	7.39E+00	6.60E+00	3.45E+02
	min	**5.14E-06**	3.00E+00	1.60E+00	1.28E+02	3.46E-04	1.40E-03	2.08E+02	1.76E+00	2.49E+02	4.73E+01	2.13E+01	3.27E+00	6.34E-01	2.80E+02
	max	**8.13E-06**	1.72E+01	1.18E+01	2.62E+02	1.09E-03	4.87E-01	2.71E+02	8.90E+00	3.04E+02	1.76E+02	4.11E+01	1.61E+01	1.87E+01	4.06E+02
	std	**8.31E-07**	3.54E+00	2.52E+00	3.69E+01	1.75E-04	1.04E-01	1.64E+01	1.60E+00	1.48E+01	3.56E+01	4.91E+00	2.75E+00	5.28E+00	2.78E+01
Sum of Different Powers	**mean**	**9.81E-11**	5.58E-06	1.25E-05	2.03E+02	6.28E-07	1.37E-06	1.03E+00	1.62E-05	4.76E-06	5.59E-03	7.41E-04	1.40E-04	7.36E-04	1.24E+00
	min	**3.54E-14**	1.00E-07	6.68E-08	1.28E+02	1.33E-07	4.22E-07	1.97E-01	5.94E-07	1.73E-07	1.01E-05	2.24E-05	4.79E-06	2.04E-07	6.71E-01
	max	**7.06E-10**	1.55E-05	1.58E-04	2.62E+02	2.19E-06	3.88E-06	1.56E+00	9.41E-05	1.36E-05	3.18E-02	5.09E-03	6.31E-04	7.99E-03	2.06E+00
	std	**1.88E-10**	4.60E-06	2.83E-05	3.69E+01	4.22E-07	7.44E-07	4.14E-01	2.18E-05	3.32E-06	8.22E-03	9.60E-04	1.38E-04	1.73E-03	3.26E-01
Zakharov	**mean**	**2.85E-03**	3.90E+02	4.37E+02	6.14E+02	5.00E+02	9.26E+02	9.42E+02	3.68E+02	1.17E+03	8.81E+02	6.43E+02	3.98E+02	6.24E+02	1.25E+03
	min	**2.37E-03**	2.87E+02	2.97E+02	3.86E+02	2.12E+02	6.79E+02	8.59E+02	2.82E+02	1.02E+03	6.56E+02	5.80E+02	2.50E+02	3.45E+02	1.21E+03
	max	**3.56E-03**	5.62E+02	7.10E+02	8.11E+02	6.41E+02	1.12E+03	1.01E+03	4.76E+02	1.28E+03	1.09E+03	7.32E+02	6.10E+02	8.00E+02	1.25E+03
	std	**2.87E-04**	5.40E+01	1.31E+02	1.05E+02	8.09E+01	1.09E+02	3.94E+01	5.33E+01	6.42E+01	1.12E+02	4.09E+01	6.13E+01	1.13E+02	7.82E+00

As illustrated in Table 1, the proposed FA model achieves the best mean global minima (very small values close to zero) and outperforms other state-of-the-art FA variants and conventional search methods, significantly, for all 10 test functions. Overall, the empirical results indicate that the proposed FA variant shows superiority in solving diverse unimodal and multimodal optimization problems over other search methods. We also employ the Wilcoxon

rank sum test (Derrac et al., 2011) to indicate the significance level of the proposed model. This two sided non-parametric statistical test is used to determine if two solutions have equal medians. A *p*-value is generated by the test to indicate the rejection of the null hypothesis of equal medians, or otherwise, at the default 5% significance level. As such, an algorithm is significantly better than another if the *p*-value is lower than 0.05. The statistical test results for all 10 standard benchmark functions are provided in Table 2. All the *p*-values are lower than 0.05, which indicates that the proposed model outperforms other methods, statistically, for all the test functions.

Table 2 The *p*-values from the Wilcoxon rank sum test for 10 standard test functions with dimension=50

Functions	NaFA	HFDE	ODFA	SFA	LSFA	CFA	FA	SA	BSO	CS	PSO	DA	ALO
Ackley	3.02E-11	3.02E-11	3.02E-11	3.02E-11	1.62E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	1.21E-12
Dixon-Price	3.02E-11	3.02E-11	3.02E-11	3.99E-04	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
Griewank	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	8.48E-09	3.02E-11
Levy	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
Rastrigin	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
Rotated Hyper-Ellipsoid	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
Rosenbrock	3.02E-11	3.02E-11	3.02E-11	2.71E-02	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.01E-11
Sphere	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
Sum of Different Powers	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	4.20E-10	3.02E-11
Zakharov	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	1.21E-12

4.2 Evaluation Using Shifted and Composite Benchmark Functions

To further ascertain efficiency of the proposed FA model, a set of 18 shifted and composite benchmark functions is employed for evaluation. These benchmark functions are popular testbeds (Mirjalili, 2015a; 2015b) and have been commonly used for evaluation of swarm intelligence-based algorithms, which include Moth-Flame (Mirjalili, 2015b), ALO (Mirjalili, 2015a), Sine Cosine (Mirjalili, 2016b), and Whale Optimization (Mirjalili and Lewis, 2016) algorithms. These benchmark functions include 7 unimodal (F1-F7), 5 multimodal (F8-F12), and 6 complex composite functions (F13-F18) as defined in Tables A.1-A.3 in Appendix. Specifically, the unimodal and multimodal test functions are shifted to increase the difficulty level of these benchmark functions. The composite functions are taken from the CEC 2005 test suite (Mirjalili, 2015a; 2015b; 2016b), which represent the combination of diverse rotated, shifted, and biased unimodal and multimodal test functions. These composite functions contain a large number of local optima with irregularity and represent diverse challenging landscapes, and they are popular testbeds for benchmarking the performance of different optimization methods (Mirjalili, 2015a; 2015b; 2016b).

We have utilized the following settings in our experiments, i.e. dimension=30 or 50, and population size=20. The classical search methods, CFA, NaFA and the proposed FA model employ the following number of function evaluations, i.e. 20 (population size) × 500 (maximum number of generations) = 10000. Again, other FA variants, i.e. LSFA, SFA, ODFA and HFDE, utilize comparatively higher numbers of function evaluations. As an example, LSFA and SFA employ the following number of function evaluations, i.e. 20 (population size) × 500 (maximum number of generations) + 20 (maximum trials of SA) × 500 (maximum number of generations) = 20000. The number of function evaluations for ODFA is 20 (population size) × 500 (maximum number of generations) × 30 or 50 (dimension) = 300000 or 500000. The number of function evaluations for HFDE is (2 × 20 (population size) + 3 offspring generated using FA) × 500 (maximum number of generations) = 21500. A total of 30 trials have been conducted for each algorithm for each test function. The detailed results are provided in Tables 3-4. The best results for each function are presented in bold. As indicated in Tables 3-4, our proposed FA model outperforms other FA variants and classical search methods for nearly all test functions for dimensions 30 and 50, except for functions 6, 11 and 12, where our algorithm achieves similar distributions or marginally lower than some of the baseline results. Specifically, SFA, FA and NaFA achieve the best results for functions 6, 11 and 12, respectively, in dimension 30, while SFA performs the best for functions 6 and 12 with ODFA showing the best performance for function 11 in dimension 50. The Wilcoxon rank sum test is conducted to further indicate the significance level of the proposed algorithm. Tables 5-6 show the *p*-values of the statistical rank sum test for dimensions 30 and 50, respectively. The *p*-values lower than 0.05 are marked in red.

Table 3 The evaluation results for 18 shifted and composite functions with dimension=30

		Prop. FA	NaFA	HFDE	LSFA	SFA	ODFA	CFA	FA	SA	BSO	CS	PSO	DA	ALO
F1	mean	**1.10E-03**	2.26E-03	1.97E+00	5.74E+04	1.32E-03	1.01E+01	3.85E+04	1.20E-02	5.07E+04	2.65E+04	9.99E+00	3.95E+01	7.19E+02	7.21E+04
	min	7.73E-04	7.93E-04	1.97E+00	4.51E+04	9.50E-04	1.11E-06	2.84E+04	4.96E-03	3.94E+04	1.45E+04	3.28E+00	7.33E-01	**0.00E+00**	5.95E+04
	max	**1.40E-03**	3.63E-03	1.97E+00	6.95E+04	2.04E-03	1.38E+02	4.38E+04	1.85E-02	6.19E+04	4.26E+04	2.06E+01	2.61E+02	4.75E+03	7.91E+04

	std	**1.47E-04**	6.71E-04	1.58E-15	5.63E+03	2.40E-04	2.65E+01	4.00E+03	3.38E-03	5.57E+03	9.44E+03	5.28E+00	6.48E+01	1.16E+03	5.03E+03
F2	mean	**1.47E-02**	1.24E-01	1.02E+01	3.72E+01	6.00E-02	1.15E+00	7.06E+05	4.71E-01	8.94E+08	1.25E+06	5.59E+00	1.35E+01	1.41E+01	4.67E+13
	min	1.21E-02	2.26E-02	1.50E-02	2.59E+01	3.09E-02	6.65E-05	2.51E+02	1.83E-01	9.44E+05	5.39E+01	1.53E+00	4.59E+00	**0.00E+00**	1.10E+08
	max	**1.65E-02**	6.68E-01	3.78E+01	5.22E+01	5.12E-01	7.52E+00	5.02E+06	7.95E-01	7.22E+09	3.69E+07	1.58E+01	2.80E+01	3.40E+01	1.01E+15
	std	**1.13E-03**	1.51E-01	1.05E+01	6.17E+00	9.06E-02	1.96E+00	1.38E+06	1.56E-01	1.73E+09	6.73E+06	3.64E+00	5.80E+00	1.00E+01	1.83E+14
F3	mean	**2.99E-03**	4.83E+03	5.48E+03	1.02E+05	3.08E+03	2.05E+02	5.41E+04	3.96E+03	7.30E+04	7.73E+04	1.54E+03	3.11E+03	1.21E+04	1.66E+05
	min	**9.48E-04**	2.74E+03	3.39E+02	3.78E+04	1.38E+03	6.79E-03	3.88E+04	1.84E+03	5.63E+04	1.67E+04	7.21E+02	7.41E+02	3.30E+01	7.79E+04
	max	**6.03E-03**	7.78E+03	2.13E+04	1.96E+05	5.63E+03	1.28E+03	6.83E+04	7.11E+03	8.82E+04	1.70E+05	2.70E+03	8.06E+03	3.11E+04	3.01E+05
	std	**1.31E-03**	1.51E+03	5.52E+03	4.20E+04	1.01E+03	3.25E+02	6.84E+03	1.33E+03	7.82E+03	3.61E+04	4.93E+02	2.32E+03	8.33E+03	5.94E+04
F4	mean	**1.44E-02**	1.03E+00	1.42E+01	8.55E+01	6.20E-02	8.53E-01	8.96E+01	1.08E-01	7.79E+01	6.28E+01	1.45E+01	1.84E+01	3.11E+01	8.74E+01
	min	1.21E-02	5.07E-02	2.77E-02	8.12E+01	2.78E-02	**1.41E-03**	7.46E+01	6.82E-02	6.94E+01	3.81E+01	1.01E+01	9.78E+00	7.97E+00	7.19E+01
	max	**1.68E-02**	3.90E+00	4.24E+01	8.98E+01	1.54E-01	6.90E+00	9.65E+01	2.09E-01	8.53E+01	8.64E+01	2.26E+01	2.86E+01	6.00E+01	9.30E+01
	std	**1.21E-03**	1.05E+00	1.17E+01	2.18E+00	3.03E-02	1.69E+00	5.87E+00	2.90E-02	3.26E+00	1.26E+01	3.20E+00	4.43E+00	1.45E+01	5.91E+00
F5	mean	**2.90E+01**	3.18E+02	2.53E+05	8.63E+07	2.04E+02	1.80E+03	9.90E+07	3.11E+02	1.55E+08	3.37E+07	8.56E+02	3.56E+03	1.52E+06	2.70E+08
	min	2.89E+01	2.60E+01	2.89E+01	3.21E+07	2.11E+01	**2.37E-07**	4.89E+07	2.72E+01	9.23E+07	3.75E+06	1.00E+02	8.93E+01	1.14E+04	2.06E+08
	max	**2.90E+01**	2.18E+03	1.71E+06	1.17E+08	1.30E+03	5.39E+04	1.32E+08	2.46E+03	2.01E+08	9.90E+07	2.67E+03	9.05E+04	8.05E+06	3.52E+08
	std	**1.22E-02**	5.75E+02	4.31E+05	1.91E+07	3.35E+02	9.83E+03	1.92E+07	5.79E+02	2.46E+07	2.08E+07	5.99E+02	1.64E+04	1.91E+06	4.10E+07
F6	mean	3.80E-01	2.20E-03	6.96E+02	5.79E+04	**8.69E-04**	1.98E+01	3.90E+04	1.29E-02	5.35E+04	2.34E+04	9.73E+00	4.36E+01	7.17E+02	7.13E+04
	min	1.18E-01	1.19E-03	7.44E-02	4.45E+04	4.55E-04	**1.93E-04**	2.62E+04	3.34E-03	4.55E+04	1.23E+04	4.52E+00	8.33E-01	3.58E+00	5.70E+04
	max	7.87E-01	3.67E-03	4.67E+03	7.78E+04	**1.15E-03**	2.96E+02	4.44E+04	2.36E-02	5.91E+04	4.18E+04	2.15E+01	2.69E+02	2.61E+03	8.51E+04
	std	1.67E-01	5.53E-04	1.16E+03	6.60E+03	**1.74E-04**	6.23E+01	4.49E+03	4.50E-03	3.59E+03	8.09E+03	4.18E+00	6.51E+01	7.03E+02	7.32E+03
F7	mean	**4.05E-02**	9.24E-02	4.77E-02	3.68E+00	4.50E-02	1.21E-01	4.31E+01	8.03E-02	6.82E+01	1.77E+01	1.45E-01	7.91E-01	4.94E-01	1.37E+02
	min	3.97E-03	3.19E-02	3.57E-03	2.70E+00	**1.19E-03**	2.48E-03	2.71E+01	3.01E-02	3.18E+01	3.74E+00	3.56E-02	2.67E-01	1.61E-02	8.73E+01
	max	**9.08E-02**	2.30E-01	1.31E-01	4.97E+00	1.51E-01	5.09E-01	6.58E+01	2.30E-01	9.95E+01	4.60E+01	3.40E-01	1.63E+00	3.87E+00	1.69E+02
	std	**2.74E-02**	4.95E-02	3.44E-02	6.04E-01	3.90E-02	1.11E-01	9.82E+00	4.06E-02	1.41E+01	9.74E+00	6.63E-02	3.48E-01	8.27E-01	2.19E+01
F8	mean	**6.00E-04**	5.11E+01	1.23E+02	2.06E+02	1.28E+02	1.43E+00	3.34E+02	4.12E+01	3.75E+02	1.14E+02	9.85E+01	6.69E+01	1.47E+02	4.51E+02
	min	2.93E-04	2.79E+01	3.08E+01	1.31E+02	7.06E+01	**1.47E-10**	3.01E+02	2.11E+01	3.00E+02	4.58E+01	6.60E+01	4.09E+01	6.58E+01	3.72E+02
	max	**7.70E-04**	9.35E+01	2.60E+02	2.84E+02	1.93E+02	3.37E+01	3.61E+02	6.88E+01	4.14E+02	2.26E+02	1.22E+02	9.37E+01	2.55E+02	5.11E+02
	std	**1.08E-04**	1.67E+01	9.71E+01	3.63E+01	2.96E+01	6.14E+00	1.62E+01	1.24E+01	2.45E+01	4.29E+01	1.31E+01	1.51E+01	5.81E+01	3.15E+01
F9	mean	**8.12E-03**	8.62E+01	4.92E+00	2.00E+01	1.58E+00	4.68E-01	1.96E+01	5.06E-02	2.01E+01	1.72E+01	5.65E+00	9.78E+00	1.95E+01	2.00E+01
	min	6.56E-03	4.58E+01	4.44E-03	2.00E+01	9.89E-03	**4.00E-05**	1.89E+01	2.41E-02	1.90E+01	1.53E+01	3.87E+00	5.07E+00	1.89E+01	2.00E+01
	max	**9.27E-03**	8.76E+01	1.36E+01	2.00E+01	2.96E+00	3.29E+00	2.00E+01	7.67E-02	2.05E+01	1.86E+01	1.10E+01	1.35E+01	2.00E+01	2.00E+01
	std	6.81E-04	7.63E+00	4.06E+00	**7.23E-15**	1.05E+00	6.52E-01	2.80E-01	1.42E-02	2.73E-01	7.29E-01	1.83E+00	1.94E+00	1.83E-01	**7.23E-15**
F10	mean	**1.83E-03**	5.85E-03	6.05E-03	1.92E-03	2.22E-03	5.77E-01	3.47E+02	7.86E-03	4.58E+02	2.33E+02	1.09E+00	2.08E+00	8.91E+00	6.53E+02
	min	8.13E-04	1.95E-03	6.05E-03	1.37E-03	1.44E-04	4.86E-04	2.47E+02	4.89E-03	3.33E+02	1.43E+02	1.01E+00	1.17E+00	**0.00E+00**	5.04E+02
	max	3.44E-03	1.60E-02	6.05E-03	**2.47E-03**	1.52E-02	2.31E+00	4.00E+02	1.24E-02	5.25E+02	4.51E+02	1.24E+00	5.28E+00	7.16E+01	7.49E+02
	std	5.13E-04	4.17E-03	**0.00E+00**	2.98E-04	4.71E-03	5.38E-01	3.64E+01	1.60E-03	4.42E+01	6.88E+01	5.28E-02	1.13E+00	1.36E+01	5.97E+01
F11	mean	5.61E-02	2.24E-02	4.97E+03	2.62E+08	6.93E-03	3.13E-01	1.74E+08	**4.76E-04**	3.01E+08	4.35E+07	4.54E+00	1.44E+01	6.93E+05	6.45E+08
	min	1.68E-02	1.40E-05	1.09E-02	1.02E+08	1.04E-05	**2.62E-08**	3.88E+07	8.37E-05	1.01E+08	2.90E+06	1.45E+00	5.25E+00	8.57E+00	3.27E+08
	max	9.82E-02	5.57E-01	8.25E+04	4.25E+08	1.04E-01	8.17E+00	2.70E+08	**1.67E-03**	4.53E+08	1.50E+08	8.40E+00	2.48E+01	1.20E+07	8.92E+08
	std	2.38E-02	1.03E-01	1.84E+04	7.60E+07	2.63E-02	1.49E+00	5.41E+07	**3.60E-04**	7.87E+07	4.21E+07	1.44E+00	5.50E+00	2.33E+06	1.47E+08
F12	mean	8.74E-01	**6.42E-04**	5.44E+05	6.10E+08	8.24E-03	1.79E+00	3.94E+08	3.25E-03	6.47E+08	1.47E+08	4.79E+01	4.45E+01	3.28E+06	1.17E+09
	min	4.44E-01	1.12E-04	2.27E-02	2.28E+08	9.30E-05	**1.37E-06**	1.64E+08	4.68E-04	2.64E+08	1.41E+07	3.42E+00	3.18E+01	1.79E+02	5.24E+08
	max	1.53E+00	**1.12E-02**	1.36E+07	1.00E+09	4.07E-02	3.54E+01	5.64E+08	1.51E-02	8.78E+08	7.21E+08	9.55E+02	6.58E+01	2.20E+07	1.79E+09
	std	2.81E-01	**2.00E-03**	2.49E+06	1.67E+08	1.11E-02	6.50E+00	9.39E+07	2.59E-03	1.49E+08	1.37E+08	1.72E+02	7.99E+00	5.52E+06	3.03E+08
F13	mean	**7.73E-06**	1.87E-05	7.72E+00	1.22E-02	1.29E-04	7.58E-02	2.47E+02	5.42E-03	3.39E+02	9.37E+01	6.44E-02	6.44E-01	6.71E+00	4.76E+02
	min	4.89E-06	1.31E-05	2.42E-06	1.39E-03	4.27E-05	1.48E-12	1.16E+02	5.48E-04	2.39E+02	4.69E+01	1.41E-02	1.15E-01	**0.00E+00**	3.12E+02
	max	**1.06E-05**	3.32E-05	3.74E+01	9.84E-02	2.97E-04	1.79E+00	3.00E+02	2.47E-02	4.07E+02	1.91E+02	1.90E-01	1.73E+00	2.32E+01	5.62E+02
	std	**1.35E-06**	3.98E-06	1.10E+01	2.37E-02	5.01E-05	3.26E-01	3.41E+01	4.76E-03	3.66E+01	3.88E+01	3.78E-02	4.62E-01	6.72E+00	6.04E+01
F14	mean	**1.59E-03**	1.82E-01	2.24E+01	4.59E+01	2.10E+00	4.07E+00	2.77E+02	4.06E+00	3.71E+02	1.18E+02	1.36E+01	1.41E+01	2.49E+01	4.97E+02
	min	1.16E-03	1.31E-02	1.47E-03	7.07E+00	8.06E-03	1.11E-08	2.17E+02	1.17E+00	2.96E+02	4.96E+01	4.04E+00	1.34E+00	**0.00E+00**	4.16E+02
	max	**2.23E-03**	6.98E-01	5.75E+01	9.07E+01	1.02E+01	2.29E+01	3.22E+02	8.62E+00	4.26E+02	2.53E+02	2.11E+01	2.66E+01	5.19E+01	6.02E+02
	std	**2.98E-04**	1.78E-01	2.16E+01	2.17E+01	2.33E+00	7.21E+00	2.85E+01	1.59E+00	3.23E+01	4.37E+01	3.85E+00	8.32E+00	1.52E+01	4.85E+01
F15	mean	**2.54E-04**	4.24E-01	2.46E+02	1.09E+02	4.19E-01	3.55E-01	1.71E+03	1.84E+00	1.74E+03	1.34E+03	1.00E+01	2.06E+01	1.35E+02	1.76E+03
	min	1.45E-04	1.40E-03	1.33E-04	1.18E-02	1.91E-03	7.61E-10	1.68E+03	6.83E-01	1.72E+03	9.68E+02	1.62E+00	5.56E-01	**0.00E+00**	1.74E+03
	max	**3.56E-04**	1.25E+01	1.18E+03	9.28E+02	1.25E+01	4.76E+00	1.73E+03	4.38E+00	1.75E+03	1.70E+03	3.51E+01	1.60E+02	6.22E+02	1.78E+03
	std	**4.41E-05**	2.27E+00	3.42E+02	2.52E+02	2.27E+00	9.91E-01	1.43E+01	8.63E-01	5.76E+00	2.29E+02	8.21E+00	3.47E+01	1.56E+02	7.45E+00
F16	mean	**8.45E-01**	5.98E+00	2.94E+02	9.58E+02	1.08E+02	2.35E+01	8.53E+02	3.40E+01	9.30E+02	7.58E+02	2.01E+02	4.38E+02	2.97E+02	1.02E+03
	min	7.41E-01	1.20E+00	7.25E-01	8.08E+02	4.13E-01	1.36E-02	8.02E+02	1.11E+01	8.63E+02	6.13E+02	8.68E+01	3.17E+02	**0.00E+00**	9.12E+02
	max	**9.26E-01**	4.16E+01	5.79E+02	1.05E+03	2.66E+02	1.36E+02	8.86E+02	5.66E+01	9.94E+02	9.71E+02	3.41E+02	5.50E+02	5.35E+02	1.12E+03
	std	**4.81E-02**	8.94E+00	2.06E+02	5.28E+01	8.90E+01	3.59E+01	2.39E+01	1.40E+01	2.91E+01	9.00E+01	7.70E+01	5.21E+01	1.60E+02	4.70E+01
F17	mean	**8.46E-01**	1.63E+01	1.38E+02	8.32E+02	9.50E+01	1.54E+01	7.14E+02	5.18E+01	7.86E+02	5.93E+02	8.83E+01	2.78E+02	1.63E+02	8.77E+02
	min	7.49E-01	1.22E+00	8.07E-01	7.49E+02	3.03E+01	1.10E-03	6.42E+02	3.35E+01	7.20E+02	4.67E+02	5.92E+01	2.13E+02	**0.00E+00**	7.57E+02
	max	**9.62E-01**	4.38E+01	4.24E+02	8.92E+02	1.32E+02	8.58E+01	7.44E+02	6.92E+01	8.34E+02	7.35E+02	1.22E+02	3.84E+02	3.76E+02	9.58E+02
	std	**5.24E-02**	1.51E+01	1.45E+02	4.04E+01	2.16E+01	2.45E+01	2.43E+01	8.43E+00	2.46E+01	6.48E+01	1.57E+01	4.63E+01	1.11E+02	4.13E+01
F18	mean	**1.42E+01**	2.08E+02	4.56E+02	9.06E+02	6.22E+02	5.71E+01	1.00E+03	2.71E+02	1.06E+03	8.87E+02	5.32E+02	6.21E+02	4.73E+02	1.15E+03
	min	1.27E+01	1.62E+02	1.72E+02	7.93E+02	3.90E+02	2.69E-01	9.53E+02	2.22E+02	1.01E+03	7.56E+02	4.45E+02	5.03E+02	**0.00E+00**	1.08E+03
	max	**1.57E+01**	2.69E+02	7.09E+02	9.84E+02	8.40E+02	2.25E+02	1.04E+03	3.22E+02	1.11E+03	1.09E+03	6.22E+02	7.32E+02	8.22E+02	1.20E+03
	std	**8.00E-01**	2.29E+01	1.79E+02	4.25E+01	1.19E+02	7.49E+01	2.22E+01	2.47E+01	2.06E+01	7.65E+01	3.73E+01	6.01E+01	2.08E+02	2.88E+01

Table 4 The evaluation results for 18 shifted and composite functions with dimension=50

		Prop. FA	**NaFA**	**HFDE**	**LSFA**	**SFA**	**ODFA**	**CFA**	**FA**	**SA**	**BSO**	**CS**	**PSO**	**DA**	**ALO**
F1	mean	**2.22E-03**	1.19E-02	2.11E+03	1.17E+05	2.49E-03	3.65E+00	8.28E+04	6.38E-02	1.03E+05	4.75E+04	2.87E+02	6.80E+02	3.04E+03	1.31E+05
	min	1.79E-03	7.57E-03	2.33E-03	9.63E+04	1.75E-03	**2.17E-04**	6.70E+04	3.29E-02	8.68E+04	2.83E+04	1.28E+02	1.01E+02	2.99E+00	1.10E+05
	max	**2.62E-03**	1.76E-02	1.30E+04	1.29E+05	3.54E-03	3.54E+01	9.31E+04	1.05E-01	1.15E+05	8.40E+04	6.00E+02	1.95E+03	1.47E+04	1.49E+05
	std	**2.16E-04**	2.48E-03	3.67E+03	8.18E+03	4.66E-04	7.34E+00	6.82E+03	1.74E-02	7.06E+03	1.37E+04	1.02E+02	4.92E+02	3.56E+03	9.43E+03
F2	mean	**2.56E-02**	7.14E+00	1.52E+01	1.07E+02	5.69E+00	8.95E-01	2.20E+13	1.91E+00	7.92E+17	3.82E+09	8.67E+09	4.42E+01	3.63E+01	1.61E+24
	min	2.19E-02	7.54E-01	2.72E-01	9.08E+01	1.11E-01	**7.84E-05**	1.24E+08	8.69E-01	1.97E+14	9.07E+01	1.39E+02	2.23E+01	1.13E+00	5.60E+17

	max	**2.90E-02**	2.48E+01	8.52E+01	1.22E+02	2.79E+01	1.36E+01	2.12E+14	4.62E+00	9.33E+18	7.80E+10	1.00E+10	8.96E+01	8.46E+01	3.29E+25
	std	**1.80E-03**	5.99E+00	2.31E+01	9.87E+00	6.54E+00	2.52E+00	4.64E+13	1.04E+00	1.78E+18	1.53E+10	3.46E+09	1.62E+01	2.27E+01	6.15E+24
F3	mean	**9.31E-03**	2.05E+04	1.66E+04	2.64E+05	1.79E+04	2.64E+03	1.43E+05	1.89E+04	1.87E+05	2.11E+05	8.86E+03	1.36E+04	3.83E+04	4.61E+05
	min	**2.67E-03**	1.30E+04	3.17E+03	1.26E+05	1.17E+04	6.43E-02	7.12E+04	1.47E+04	1.29E+05	5.75E+04	6.09E+03	4.41E+03	9.65E+03	2.00E+05
	max	**2.40E-02**	3.18E+04	4.93E+04	4.47E+05	3.05E+04	3.60E+04	1.89E+05	2.51E+04	2.27E+05	7.37E+05	1.20E+04	3.87E+04	1.23E+05	7.87E+05
	std	**4.92E-03**	4.86E+03	1.24E+04	9.20E+04	4.78E+03	7.14E+03	2.39E+04	2.79E+03	2.63E+04	1.41E+05	1.48E+03	7.47E+03	2.28E+04	1.42E+05
F4	mean	**1.75E-02**	1.40E+01	2.09E+01	9.07E+01	8.46E+00	3.38E-01	9.87E+01	9.61E+00	8.61E+01	6.73E+01	2.31E+01	3.29E+01	4.23E+01	9.31E+01
	min	1.45E-02	6.75E+00	7.36E+00	8.53E+01	1.59E+00	**1.42E-04**	9.87E+01	2.42E+00	8.24E+01	5.04E+01	1.43E+01	2.35E+01	1.49E+01	8.36E+01
	max	**2.05E-02**	2.50E+01	4.33E+01	9.55E+01	1.59E+01	1.72E+00	9.87E+01	1.87E+01	9.14E+01	8.57E+01	3.23E+01	6.52E+01	6.68E+01	9.61E+01
	std	1.26E-03	4.02E+00	1.06E+01	2.55E+00	3.93E+00	5.02E-01	**1.45E-14**	4.23E+00	2.27E+00	8.93E+00	3.94E+00	9.23E+00	1.35E+01	2.61E+00
F5	mean	**4.90E+01**	1.34E+03	5.89E+05	2.88E+08	1.72E+02	1.39E+04	2.69E+08	2.53E+02	3.71E+08	9.43E+07	6.32E+04	2.95E+04	7.69E+05	5.18E+08
	min	4.89E+01	4.56E+01	4.88E+01	2.12E+08	4.18E+01	**4.21E-07**	1.84E+08	4.96E+01	2.28E+08	2.35E+07	1.20E+04	1.49E+03	4.89E+01	3.40E+08
	max	**4.90E+01**	1.26E+04	8.13E+06	4.03E+08	1.04E+03	4.16E+05	3.49E+08	1.49E+03	4.61E+08	4.64E+08	1.63E+05	1.73E+05	3.36E+06	6.41E+08
	std	**1.46E-02**	2.96E+03	1.85E+06	4.30E+07	2.44E+02	7.60E+04	4.10E+07	3.54E+02	4.94E+07	8.72E+07	3.98E+04	3.65E+04	8.40E+05	7.55E+07
F6	mean	2.41E+00	1.31E-02	2.66E+03	1.16E+05	**2.57E-03**	1.72E+02	8.37E+04	7.10E-02	1.03E+05	4.46E+04	3.01E+02	6.32E+02	2.48E+03	1.28E+05
	min	1.14E+00	8.18E-03	1.94E-01	9.37E+04	1.47E-03	**1.11E-03**	6.91E+04	4.21E-02	9.09E+04	2.29E+04	1.24E+02	1.00E+02	1.06E+01	1.09E+05
	max	3.56E+00	1.93E-02	1.30E+04	1.32E+05	**4.00E-03**	3.40E+03	9.46E+04	1.31E-01	1.13E+05	7.39E+04	6.21E+02	1.51E+03	1.20E+04	1.43E+05
	std	5.98E-01	3.45E-03	4.08E+03	1.14E+04	**5.96E-04**	6.74E+02	6.66E+03	1.96E-02	6.07E+03	1.20E+04	1.19E+02	3.93E+02	3.53E+03	8.58E+03
F7	mean	**4.26E-02**	1.49E-01	5.64E-02	5.98E+00	8.84E-02	1.30E-01	2.09E+02	1.34E-01	2.75E+02	3.56E+01	6.70E-01	3.91E+00	1.53E+00	4.41E+02
	min	**9.45E-04**	6.14E-02	2.99E-03	4.07E+00	3.30E-02	1.37E-02	1.40E+02	6.41E-02	2.14E+02	1.11E+01	3.59E-01	1.56E+00	4.28E-02	3.27E+02
	max	2.06E-01	2.54E-01	**1.31E-01**	8.57E+00	1.42E-01	7.06E-01	2.53E+02	2.61E-01	3.30E+02	7.21E+01	1.29E+00	6.46E+00	1.17E+01	5.27E+02
	std	4.58E-02	4.86E-02	4.23E-02	9.28E-01	**2.77E-02**	1.34E-01	2.95E+01	4.91E-02	2.88E+01	1.58E+01	2.15E-01	1.19E+00	2.38E+00	5.33E+01
F8	mean	**1.11E-03**	1.05E+02	2.93E+02	3.72E+02	2.12E+02	3.99E+00	6.37E+02	9.74E+01	7.02E+02	1.65E+02	2.02E+02	1.39E+02	3.20E+02	7.94E+02
	min	8.03E-04	6.58E+01	3.68E+01	3.04E+02	1.55E+02	**5.95E-09**	5.63E+02	6.36E+01	6.41E+02	9.39E+01	1.61E+02	7.51E+01	1.16E+02	6.84E+02
	max	**1.33E-03**	1.50E+02	4.67E+02	4.57E+02	2.95E+02	4.88E+01	6.76E+02	1.77E+02	7.42E+02	2.91E+02	2.82E+02	2.12E+02	4.62E+02	8.47E+02
	std	**1.31E-04**	1.91E+01	1.67E+02	4.23E+01	3.80E+01	1.11E+01	2.76E+01	2.46E+01	2.37E+01	4.73E+01	2.38E+01	2.86E+01	8.70E+01	3.97E+01
F9	mean	**8.63E-03**	1.66E-01	4.69E+00	2.00E+01	7.44E-01	1.27E+00	2.02E+01	1.89E-01	2.05E+01	1.71E+01	9.45E+00	1.32E+01	1.96E+01	2.00E+01
	min	7.62E-03	2.20E-02	1.23E-02	2.00E+01	1.58E-02	**1.81E-06**	1.99E+01	1.18E-01	2.02E+01	1.49E+01	6.76E+00	1.10E+01	1.90E+01	2.00E+01
	max	**9.57E-03**	1.04E+00	1.51E+01	2.00E+01	2.95E+00	1.00E+01	2.04E+01	3.84E-01	2.07E+01	1.86E+01	1.27E+01	1.54E+01	2.00E+01	2.00E+01
	std	4.63E-04	2.97E-01	4.69E+00	**7.23E-15**	9.32E-01	2.19E+00	1.34E-01	6.16E-02	1.29E-01	9.43E-01	1.72E+00	1.18E+00	1.94E-01	**7.23E-15**
F10	mean	**2.19E-03**	2.98E-02	1.76E+01	1.16E+03	8.16E-03	1.05E+00	7.60E+02	2.08E-02	9.21E+02	3.84E+02	3.88E+00	1.44E+01	1.83E+01	1.17E+03
	min	1.35E-03	8.84E-03	3.85E-03	9.81E+02	2.66E-03	8.98E-06	6.55E+02	1.07E-02	7.50E+02	2.03E+02	2.48E+00	3.34E+00	**0.00E+00**	9.47E+02
	max	**3.03E-03**	6.54E-02	1.11E+02	1.30E+03	2.17E-02	8.71E+00	8.49E+02	3.89E-02	1.03E+03	6.32E+02	8.99E+00	9.48E+01	1.08E+02	1.28E+03
	std	**3.24E-04**	1.39E-02	3.01E+01	9.00E+01	4.07E-03	1.57E+00	4.73E+01	6.49E-03	5.99E+01	1.15E+02	1.56E+00	1.69E+01	2.20E+01	7.94E+01
F11	mean	1.20E-01	1.96E+00	3.98E+05	8.31E+08	3.18E-02	**1.98E-02**	6.13E+08	7.99E-02	7.99E+08	1.35E+08	3.29E+01	2.61E+01	5.42E+05	1.28E+09
	min	8.54E-02	4.08E-01	2.95E-02	5.97E+08	3.22E-05	**1.47E-11**	3.47E+08	5.97E-04	5.68E+08	4.67E+06	6.87E+00	1.13E+01	1.34E+01	7.12E+08
	max	**1.67E-01**	4.26E+00	4.73E+06	1.08E+09	2.62E-01	2.72E-01	8.47E+08	6.67E-01	9.80E+08	7.92E+08	5.63E+02	4.78E+01	5.88E+06	1.77E+09
	std	**2.41E-02**	1.27E+00	1.06E+06	1.37E+08	6.43E-02	6.29E-02	1.40E+08	1.64E-01	8.95E+07	1.72E+08	1.00E+02	9.13E+00	1.27E+06	2.28E+08
F12	mean	2.41E+00	6.01E+00	1.50E+06	1.63E+09	**3.69E-03**	3.07E+00	1.27E+09	1.77E-02	1.59E+09	3.31E+08	1.23E+04	2.47E+02	4.35E+06	2.33E+09
	min	1.64E+00	2.24E-03	8.55E-02	1.16E+09	2.81E-04	**2.69E-08**	8.33E+08	6.05E-03	1.02E+09	7.74E+07	8.58E+01	5.58E+01	1.24E+02	1.87E+09
	max	3.09E+00	3.07E+01	1.33E+07	1.91E+09	**2.15E-02**	5.19E+01	1.57E+09	3.37E-02	1.82E+09	1.02E+09	6.75E+04	3.32E+03	2.67E+07	2.71E+09
	std	4.06E-01	7.97E+00	3.89E+06	1.97E+08	**5.78E-03**	1.06E+01	1.74E+08	6.92E-03	1.96E+08	2.34E+08	1.75E+04	5.96E+02	7.24E+06	2.21E+08
F13	mean	**8.76E-06**	2.31E+01	4.94E+00	2.85E+01	7.25E-04	2.17E-01	3.35E+02	4.89E-02	4.14E+02	1.34E+02	1.34E+00	4.53E+00	9.46E+00	5.26E+02
	min	6.74E-06	1.36E+01	3.12E-05	1.46E+01	3.45E-04	2.44E-12	2.81E+02	2.71E-02	3.62E+02	6.05E+01	5.96E-01	4.16E-01	**0.00E+00**	3.99E+02
	max	**1.04E-05**	3.38E+01	4.39E+01	3.95E+01	1.06E-03	4.86E+00	3.71E+02	9.14E-02	4.54E+02	3.39E+02	2.80E+00	1.14E+01	3.78E+01	5.94E+02
	std	**8.77E-07**	5.15E+00	1.08E+01	6.52E+00	1.61E-04	8.93E-01	2.42E+01	1.56E-02	2.67E+01	5.73E+01	5.12E-01	2.57E+00	1.02E+01	4.88E+01
F14	mean	**1.18E-03**	3.99E+01	1.90E+01	4.44E+01	2.33E-01	4.13E+00	3.46E+02	7.31E+00	4.25E+02	1.39E+02	1.69E+01	2.07E+01	2.41E+01	5.37E+02
	min	8.37E-04	2.97E+01	1.63E+01	3.13E+01	1.57E-02	**1.58E-08**	3.00E+02	4.22E+00	3.49E+02	8.35E+01	1.54E+01	1.69E+01	4.33E+00	4.53E+02
	max	**1.44E-03**	5.38E+01	6.46E+01	6.49E+01	9.18E-01	1.71E+01	3.77E+02	8.83E+00	4.67E+02	2.72E+02	1.83E+01	3.18E+01	4.30E+01	6.06E+02
	std	**1.70E-04**	6.92E+00	2.19E+01	6.92E+00	2.54E-01	5.75E+00	2.11E+01	1.06E+00	3.08E+01	4.83E+01	5.62E-01	3.09E+00	8.69E+00	3.38E+01
F15	mean	**2.85E-04**	9.33E+00	2.60E+02	6.90E+02	1.30E-02	6.54E-01	1.60E+03	1.63E+01	1.62E+03	1.41E+03	4.37E+01	1.41E+02	2.70E+02	1.65E+03
	min	1.72E-04	1.07E+00	3.44E-01	2.36E+02	7.40E-03	1.65E-11	1.59E+03	5.30E+00	1.60E+03	9.55E+02	1.54E+01	4.09E+00	**0.00E+00**	1.63E+03
	max	**3.85E-04**	4.13E+01	1.17E+03	1.53E+03	1.87E-02	7.76E+00	1.61E+03	3.41E+01	1.63E+03	1.57E+03	9.59E+01	1.11E+03	9.99E+02	1.66E+03
	std	**5.10E-05**	8.92E+00	3.90E+02	2.83E+02	2.61E-03	1.75E+00	5.77E+00	6.54E+00	7.24E+00	1.59E+02	2.01E+01	2.16E+02	2.88E+02	8.97E+00
F16	mean	**8.90E-01**	1.33E+02	2.69E+02	9.98E+02	2.42E+02	1.38E+01	9.13E+02	7.64E+01	9.85E+02	7.22E+02	3.64E+02	5.17E+02	3.29E+02	1.05E+03
	min	8.24E-01	1.77E+01	4.54E+00	9.13E+02	1.38E+02	8.06E-03	8.58E+02	4.64E+01	9.56E+02	6.11E+02	3.00E+02	4.44E+02	**0.00E+00**	9.66E+02
	max	**9.71E-01**	2.92E+02	5.92E+02	1.05E+03	3.76E+02	6.18E+01	9.49E+02	1.53E+02	1.02E+03	8.55E+02	4.22E+02	6.01E+02	5.85E+02	1.11E+03
	std	**3.76E-02**	7.68E+01	2.10E+02	3.45E+01	7.06E+01	2.00E+01	2.43E+01	2.69E+01	1.61E+01	5.29E+01	3.74E+01	4.07E+01	1.77E+02	3.62E+01
F17	mean	**8.81E-01**	4.35E+01	1.33E+02	8.73E+02	1.06E+02	2.96E+01	7.75E+02	6.12E+01	8.31E+02	5.95E+02	1.64E+02	3.74E+02	2.36E+02	9.13E+02
	min	7.71E-01	2.97E+01	2.56E+01	8.05E+02	6.82E+01	**7.16E-03**	7.45E+02	3.79E+01	7.96E+02	4.74E+02	1.34E+02	3.28E+02	2.35E+01	8.41E+02
	max	**9.79E-01**	6.23E+01	4.55E+02	9.24E+02	1.56E+02	2.13E+02	8.12E+02	8.08E+01	8.65E+02	7.53E+02	1.97E+02	4.45E+02	3.98E+02	9.81E+02
	std	**4.55E-02**	8.33E+00	1.24E+02	3.28E+01	2.06E+01	4.76E+01	1.68E+01	9.04E+00	1.63E+01	6.38E+01	2.01E+01	3.64E+01	9.83E+01	3.55E+01
F18	mean	**1.47E+01**	3.29E+02	5.52E+02	9.25E+02	7.61E+02	9.39E+01	1.06E+03	3.21E+02	1.10E+03	9.05E+02	6.17E+02	7.39E+02	4.60E+02	1.18E+03
	min	1.37E+01	2.35E+02	3.20E+02	8.56E+02	5.55E+02	**1.46E-01**	1.02E+03	2.71E+02	1.03E+03	8.05E+02	5.47E+02	6.62E+02	7.99E+01	1.12E+03
	max	**1.65E+01**	4.16E+02	8.01E+02	9.97E+02	8.87E+02	3.76E+02	1.08E+03	3.71E+02	1.14E+03	1.01E+03	6.89E+02	8.32E+02	8.56E+02	1.22E+03
	std	**6.18E-01**	4.38E+01	1.48E+02	3.20E+01	7.51E+01	1.10E+02	1.61E+01	2.54E+01	2.37E+01	4.87E+01	3.17E+01	4.66E+01	1.85E+02	2.50E+01

Table 5 The *p*-Values of the Wilcoxon rank sum test for 18 test functions with dimension=30

	NaFA	HFDE	LSFA	SFA	ODFA	CFA	FA	SA	BSO	CS	PSO	DA	ALO
F1	1.07E-07	2.44E-09	3.02E-11	7.74E-06	5.57E-10	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	1.06E-07	3.02E-11
F2	3.02E-11	1.46E-10	3.02E-11	3.02E-11	7.73E-02	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	5.57E-10	3.02E-11
F3	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.34E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
F4	1.10E-06	6.71E-05	9.46E-06	9.46E-06	3.55E-04	6.69E-05	6.71E-05	6.71E-05	6.71E-05	6.71E-05	6.71E-05	6.71E-05	6.71E-05
F5	2.71E-02	8.48E-09	3.02E-11	3.79E-01	1.07E-07	3.02E-11	7.20E-05	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
F6	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.99E-04	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
F7	1.77E-03	4.29E-01	3.02E-11	5.49E-01	1.24E-03	3.02E-11	2.42E-02	3.02E-11	3.02E-11	1.47E-07	3.02E-11	1.68E-03	3.02E-11
F8	3.02E-11	3.02E-11	3.02E-11	3.02E-11	7.55E-01	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
F9	3.69E-11	9.51E-06	1.21E-12	3.02E-11	1.07E-07	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	2.31E-11	1.21E-12

F10	2.87E-10	2.19E-08	3.02E-11	7.38E-10	8.89E-10	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	5.57E-10	3.02E-11
F11	1.11E-06	3.02E-11	3.02E-11	3.02E-11	5.57E-10	3.02E-11	3.02E-11	3.02E-11	3.02E-11	5.57E-10	3.02E-11	3.02E-11	3.02E-11
F12	3.02E-11	3.02E-11	3.02E-11	3.02E-11	1.11E-06	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
F13	3.02E-11	1.06E-07	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	1.06E-07	3.02E-11
F14	3.02E-11	5.57E-10	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	6.71E-05	3.02E-11
F15	3.02E-11	1.31E-08	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	8.48E-09	3.02E-11
F16	3.02E-11	6.52E-09	3.02E-11	2.71E-02	3.53E-02	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	1.06E-07	3.02E-11
F17	3.02E-11	2.67E-09	3.02E-11	3.02E-11	6.63E-01	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	9.46E-06	3.02E-11
F18	3.02E-11	3.02E-11	3.02E-11	3.02E-11	5.89E-01	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	5.57E-10	3.02E-11

Table 6 The *p*-Values of the Wilcoxon rank sum test for 18 test functions with dimension=50

	NaFA	**HFDE**	**LSFA**	**SFA**	**ODFA**	**CFA**	**FA**	**SA**	**BSO**	**CS**	**PSO**	**DA**	**ALO**
F1	3.02E-11	9.92E-11	3.02E-11	9.79E-05	1.11E-06	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
F2	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.79E-01	3.02E-11	3.02E-11	3.02E-11	3.02E-11	4.11E-12	3.02E-11	3.02E-11	3.02E-11
F3	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
F4	2.68E-04	4.28E-06	3.73E-07	3.73E-07	2.94E-08	7.74E-07	4.28E-06	4.28E-06	4.28E-06	4.28E-06	4.28E-06	4.28E-06	4.28E-06
F5	9.51E-06	1.03E-06	3.02E-11	3.79E-01	7.48E-02	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	4.62E-10	3.02E-11
F6	3.02E-11	1.07E-07	3.02E-11	3.02E-11	6.77E-05	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	1.07E-07	3.02E-11
F7	8.15E-11	9.94E-01	3.02E-11	1.87E-05	1.32E-04	3.02E-11	2.19E-08	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.16E-10	3.02E-11
F8	3.02E-11	3.02E-11	3.02E-11	3.02E-11	9.75E-01	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
F9	3.02E-11	3.02E-11	1.21E-12	3.02E-11	1.07E-07	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	2.83E-11	1.21E-12
F10	3.02E-11	3.02E-11	3.02E-11	3.69E-11	1.29E-09	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	8.48E-09	3.02E-11
F11	7.73E-02	6.77E-05	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
F12	1.07E-07	3.02E-11	3.02E-11	3.02E-11	1.07E-07	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
F13	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	5.57E-10	3.02E-11
F14	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
F15	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	1.06E-07	3.02E-11
F16	3.02E-11	3.02E-11	3.02E-11	3.02E-11	6.63E-01	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	1.06E-07	3.02E-11
F17	3.02E-11	3.02E-11	3.02E-11	3.02E-11	7.96E-03	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
F18	3.02E-11	3.02E-11	3.02E-11	3.02E-11	1.95E-03	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11

As indicated in Table 5, the *p*-values are lower than 0.05 for nearly all the test functions with dimension 30, which indicate that our proposed algorithm achieves statistically significant improvements over other methods. The exceptions are functions 5 and 7, where the proposed algorithm shows similar results as compared with those of SFA, and functions 2, 8, and 17-18, where our algorithm has similar results as compared with those of ODFA. In addition, HFDE shows similar result distributions as compared with those of the proposed model for function 7. The proposed algorithm outperforms other FA variants and classical search methods significantly in other test cases. A similar observation is obtained for test functions with dimension 50. As illustrated in Table 6, the majority of the *p*-values are lower than 0.05. The proposed FA model shows similar results as compared with those of NaFA, HFDE, SFA and ODFA, for function 11, function 7, function 5, and functions 2, 5, 8, and 16, respectively. The proposed model shows statistically significant performance improvements over other FA variants and classical methods in all other test cases. Among the baseline methods, ODFA and SFA perform the best. These two methods as well as LSFA also employ an extra number of function evaluations for identification or further improvement of the global best solution in each iteration (Verma et al., 2016; Alweshah and Abdullah, 2015). Comparing with our proposed model, HFDE (Dash et al., 2017) also employs a higher number of function evaluations to increase global exploration by integrating DE with FA.

Overall, our algorithm shows impressive capability of dealing with the shifted and composite benchmark functions in dimensions 30 and 50. The empirical results indicate that our results are statistically better than those of other methods in most of the test cases.

4.3 Evaluation of Different Proposed Strategies

To identify how each strategy contributes to the overall performance of the proposed FA model, a number of additional experiments have been conducted using either solely the newly added evading mechanism or attraction mechanism. Specifically, we evaluate the version of FA + the proposed attraction behaviour guided by both local and global optimal solutions, and the version of FA + the proposed evading action advised by both local and global worst solutions, respectively. Both the 10 standard test functions with dimension 50 and the number of function evaluations of 2000, and the 18 shifted and composite benchmarks with dimensions 30 and 50 and the number of function evaluations of 10000, have been used for evaluation. The detailed results are provided in Tables 7-8. The results obtained using each strategy are compared against those obtained by the full version of the proposed FA model with both attraction and evading mechanisms, as well as the original FA model. The best performances are marked in bold cases.

Table 7 Comparison of different proposed strategies using the 10 standard test functions with dimension=50

		Full version	Evading only	Attraction only	FA
Ackley	**Mean**	**6.57E-03**	6.66E-03	8.93E+00	9.13E+00
	Min	5.46E-03	**5.15E-03**	6.27E+00	6.60E+00
	Max	**7.01E-03**	7.20E-03	1.09E+01	1.14E+01
	Std	**3.39E-04**	3.96E-04	1.44E+00	1.08E+00
Dixon	**Mean**	9.97E-01	**9.97E-01**	3.54E+03	3.09E+03
	Min	9.96E-01	**9.95E-01**	6.92E+02	9.37E+02
	Max	**9.99E-01**	9.99E-01	9.10E+03	6.34E+03
	Std	**7.45E-04**	7.92E-04	1.79E+03	1.23E+03
Griewank	**Mean**	**2.55E-03**	2.56E-03	1.47E+01	1.20E+01
	Min	**1.65E-03**	2.02E-03	4.02E+00	3.76E+00
	Max	3.50E-03	**3.29E-03**	3.70E+01	3.00E+01
	Std	4.59E-04	**3.47E-04**	7.87E+00	6.36E+00
Levy	**Mean**	5.07E+00	**5.07E+00**	1.12E+02	1.15E+02
	Min	5.07E+00	**5.07E+00**	6.56E+01	5.26E+01
	Max	**5.07E+00**	5.08E+00	1.90E+02	3.08E+02
	Std	**6.98E-04**	9.04E-04	3.28E+01	5.54E+01
Rastrigin	**Mean**	1.37E-03	**1.30E-03**	2.84E+02	2.76E+02
	Min	**9.07E-04**	1.06E-03	2.35E+02	1.82E+02
	Max	**1.59E-03**	1.64E-03	3.89E+02	4.12E+02
	Std	1.70E-04	**1.52E-04**	3.83E+01	5.00E+01
Rotated Hyper-Ellipsoid	**Mean**	2.75E-02	**2.53E-02**	1.89E+04	1.70E+04
	Min	**1.74E-02**	1.79E-02	8.46E+03	4.99E+03
	Max	3.50E-02	**3.08E-02**	3.16E+04	3.49E+04
	Std	4.56E-03	**3.59E-03**	6.44E+03	8.79E+03
Rosenbrock	**Mean**	4.90E+01	**4.90E+01**	8.85E+03	8.23E+03
	Min	**4.90E+01**	4.90E+01	1.72E+03	1.68E+03
	Max	4.90E+01	**4.90E+01**	3.09E+04	2.02E+04
	Std	4.65E-03	**3.09E-03**	6.07E+03	5.10E+03
Sphere	**Mean**	**6.76E-06**	6.78E-06	5.71E+00	4.48E+00
	Min	4.91E-06	**4.40E-06**	1.81E+00	1.67E+00
	Max	9.06E-06	**8.47E-06**	1.43E+01	1.39E+01
	Std	1.03E-06	**9.20E-07**	2.90E+00	2.47E+00
Sum of Different Powers	**Mean**	**6.42E-11**	1.29E-10	1.81E-05	1.94E-05
	Min	**9.38E-14**	8.29E-13	7.49E-07	1.14E-06
	Max	**3.70E-10**	6.66E-10	7.27E-05	5.38E-05
	Std	**1.06E-10**	1.88E-10	1.60E-05	1.53E-05
Zakharov	**Mean**	**2.77E-03**	2.97E-03	3.73E+02	3.63E+02
	Min	**1.91E-03**	1.96E-03	2.97E+02	2.77E+02
	Max	**3.47E-03**	3.54E-03	5.18E+02	4.60E+02
	Std	4.14E-04	**3.29E-04**	5.38E+01	5.06E+01

Table 8 Comparison of different proposed strategies using the shifted and composite test functions with dimensions 30 and 50

		30D				50D			
		Full version	**Evading only**	**Attraction only**	**FA**	**Full version**	**Evading only**	**Attraction only**	**FA**
F1	mean	**1.10E-03**	1.14E-03	1.36E-02	1.20E-02	2.22E-03	**2.15E-03**	7.33E-02	6.38E-02
	min	7.73E-04	**6.93E-04**	5.39E-03	4.96E-03	1.79E-03	**1.67E-03**	3.64E-02	3.29E-02
	max	**1.40E-03**	1.65E-03	3.10E-02	1.85E-02	2.62E-03	**2.54E-03**	1.13E-01	1.05E-01
	std	**1.47E-04**	2.07E-04	5.76E-03	3.38E-03	**2.16E-04**	2.28E-04	2.00E-02	1.74E-02
F2	mean	1.47E-02	**1.45E-02**	4.19E-01	4.71E-01	**2.56E-02**	2.58E-02	1.88E+00	1.91E+00
	min	1.21E-02	**1.17E-02**	1.51E-01	1.83E-01	**2.19E-02**	2.31E-02	8.57E-01	8.69E-01
	max	**1.65E-02**	1.73E-02	5.98E-01	7.95E-01	**2.90E-02**	2.90E-02	4.61E+00	4.62E+00
	std	**1.13E-03**	1.35E-03	1.02E-01	1.56E-01	1.80E-03	**1.30E-03**	9.68E-01	1.04E+00
F3	mean	**2.99E-03**	3.40E-03	3.81E+03	3.96E+03	**9.31E-03**	9.86E-03	1.74E+04	1.89E+04
	min	**9.48E-04**	1.63E-03	1.73E+03	1.84E+03	**2.67E-03**	4.91E-03	1.12E+04	1.47E+04
	max	**6.03E-03**	6.04E-03	7.18E+03	7.11E+03	2.40E-02	**2.37E-02**	2.39E+04	2.51E+04
	std	1.31E-03	**1.19E-03**	1.45E+03	1.33E+03	4.92E-03	**4.55E-03**	3.41E+03	2.79E+03
F4	mean	1.44E-02	**1.29E-02**	1.21E-01	1.08E-01	**1.75E-02**	1.78E-02	1.04E+01	9.61E+00
	min	1.21E-02	**1.08E-02**	7.42E-02	6.82E-02	**1.45E-02**	1.57E-02	2.14E+00	2.42E+00
	max	1.68E-02	**1.43E-02**	3.57E-01	2.09E-01	2.05E-02	**2.03E-02**	2.93E+01	1.87E+01
	std	1.21E-03	**9.50E-04**	5.00E-02	2.90E-02	**1.26E-03**	1.43E-03	6.89E+00	4.23E+00
F5	mean	2.90E+01	**2.90E+01**	1.70E+02	3.11E+02	4.90E+01	**4.90E+01**	4.29E+02	2.53E+02
	min	2.89E+01	2.89E+01	**2.65E+01**	2.72E+01	**4.89E+01**	4.89E+01	5.03E+01	4.96E+01
	max	2.90E+01	**2.90E+01**	6.57E+02	2.46E+03	4.90E+01	**4.90E+01**	2.25E+03	1.49E+03
	std	1.22E-02	**9.56E-03**	1.70E+02	5.79E+02	1.46E-02	**1.44E-02**	6.56E+02	3.54E+02
F6	mean	3.80E-01	1.55E+00	1.41E-02	**1.29E-02**	2.41E+00	1.24E+01	**6.64E-02**	7.10E-02
	min	1.18E-01	5.37E-01	5.50E-03	**3.34E-03**	1.14E+00	1.23E+01	**2.87E-02**	4.21E-02
	max	7.87E-01	2.36E+00	2.57E-02	**2.36E-02**	3.56E+00	1.24E+01	**1.23E-01**	1.31E-01
	std	1.67E-01	4.51E-01	4.79E-03	**4.50E-03**	5.98E-01	2.24E-02	1.98E-02	**1.96E-02**

F7	mean	**4.05E-02**	5.33E-02	9.04E-02	8.03E-02	**4.26E-02**	4.90E-02	1.08E-01	1.34E-01
	min	3.97E-03	**9.09E-04**	1.99E-02	3.01E-02	9.45E-04	**5.99E-05**	6.17E-02	6.41E-02
	max	**9.08E-02**	1.83E-01	2.84E-01	2.30E-01	2.06E-01	2.56E-01	**1.86E-01**	2.61E-01
	std	**2.74E-02**	4.40E-02	6.81E-02	4.06E-02	4.58E-02	5.90E-02	**3.79E-02**	4.91E-02
F8	mean	**6.00E-04**	6.04E-04	4.21E+01	4.12E+01	**1.11E-03**	1.14E-03	1.09E+02	9.74E+01
	min	**2.93E-04**	4.27E-04	1.62E+01	2.11E+01	**8.03E-04**	8.55E-04	6.41E+01	6.36E+01
	max	7.70E-04	**7.69E-04**	9.99E+01	6.88E+01	**1.33E-03**	1.33E-03	1.67E+02	1.77E+02
	std	1.08E-04	**7.82E-05**	1.62E+01	1.24E+01	1.31E-04	**1.29E-04**	2.17E+01	2.46E+01
F9	mean	**8.12E-03**	8.26E-03	6.47E-02	5.06E-02	**8.63E-03**	8.64E-03	2.24E-01	1.89E-01
	min	6.56E-03	**6.52E-03**	3.25E-02	2.41E-02	7.62E-03	**7.49E-03**	1.03E-01	1.18E-01
	max	9.27E-03	**9.02E-03**	3.20E-01	7.67E-02	9.57E-03	**9.43E-03**	5.42E-01	3.84E-01
	std	6.81E-04	**5.89E-04**	5.07E-02	1.42E-02	4.63E-04	**4.38E-04**	1.09E-01	6.16E-02
F10	mean	**1.83E-03**	1.94E-03	7.84E-03	7.86E-03	**2.19E-03**	2.30E-03	2.21E-02	2.08E-02
	min	**8.13E-04**	1.11E-03	2.56E-03	4.89E-03	**1.35E-03**	1.38E-03	9.81E-03	1.07E-02
	max	3.44E-03	**2.67E-03**	3.21E-02	1.24E-02	**3.03E-03**	3.04E-03	3.93E-02	3.89E-02
	std	5.13E-04	**3.62E-04**	5.47E-03	1.60E-03	**3.24E-04**	4.18E-04	5.98E-03	6.49E-03
F11	mean	5.61E-02	8.43E-01	**4.44E-04**	4.76E-04	1.20E-01	1.20E-01	**7.49E-02**	7.99E-02
	min	1.68E-02	7.54E-01	1.34E-04	**8.37E-05**	8.54E-02	6.89E-02	7.94E-04	**5.97E-04**
	max	9.82E-02	9.34E-01	**1.37E-03**	1.67E-03	**1.67E-01**	1.95E-01	1.80E+00	6.67E-01
	std	2.38E-02	5.91E-02	**2.95E-04**	3.60E-04	**2.41E-02**	3.66E-02	3.29E-01	1.64E-01
F12	mean	8.74E-01	9.42E-01	3.38E-03	**3.25E-03**	2.41E+00	2.62E+00	**1.36E-02**	1.77E-02
	min	4.44E-01	3.27E-01	9.96E-04	**4.68E-04**	1.64E+00	1.79E+00	**3.43E-03**	6.05E-03
	max	1.53E+00	1.57E+00	**1.41E-02**	1.51E-02	3.09E+00	3.72E+00	**2.72E-02**	3.37E-02
	std	2.81E-01	2.07E-01	2.94E-03	**2.59E-03**	4.06E-01	5.83E-01	**5.21E-03**	6.92E-03
F13	mean	7.73E-06	**7.64E-06**	6.08E-03	5.42E-03	8.76E-06	**8.42E-06**	4.81E-02	4.89E-02
	min	**4.89E-06**	5.48E-06	1.87E-04	5.48E-04	6.74E-06	**5.43E-06**	2.06E-02	2.71E-02
	max	1.06E-05	**9.25E-06**	1.59E-02	2.47E-02	1.04E-05	**1.02E-05**	9.15E-02	9.14E-02
	std	1.35E-06	**1.08E-06**	4.54E-03	4.76E-03	**8.77E-07**	1.04E-06	1.51E-02	1.56E-02
F14	mean	1.59E-03	**1.53E-03**	3.84E+00	4.06E+00	**1.18E-03**	1.21E-03	7.44E+00	7.31E+00
	min	1.16E-03	**8.67E-04**	1.41E+00	1.17E+00	8.37E-04	**6.94E-04**	5.03E+00	4.22E+00
	max	2.23E-03	**2.22E-03**	6.85E+00	8.62E+00	**1.44E-03**	1.53E-03	1.02E+01	8.83E+00
	std	**2.98E-04**	3.28E-04	1.46E+00	1.59E+00	**1.70E-04**	2.04E-04	1.53E+00	1.06E+00
F15	mean	2.54E-04	**2.52E-04**	2.26E+00	1.84E+00	2.85E-04	**2.78E-04**	1.45E+01	1.63E+01
	min	1.45E-04	**1.19E-04**	3.01E-01	6.83E-01	**1.72E-04**	1.75E-04	6.43E+00	5.30E+00
	max	**3.56E-04**	3.74E-04	1.30E+01	4.38E+00	3.85E-04	**3.70E-04**	3.59E+01	3.41E+01
	std	**4.41E-05**	6.66E-05	3.05E+00	8.63E-01	**5.10E-05**	5.23E-05	7.23E+00	6.54E+00
F16	mean	**8.45E-01**	8.49E-01	3.47E+01	3.40E+01	**8.90E-01**	8.91E-01	7.50E+01	7.64E+01
	min	7.41E-01	**7.18E-01**	1.51E+01	1.11E+01	8.24E-01	**8.22E-01**	4.71E+01	4.64E+01
	max	**9.26E-01**	9.72E-01	7.95E+01	5.66E+01	9.71E-01	**9.48E-01**	1.56E+02	1.53E+02
	std	**4.81E-02**	6.07E-02	1.51E+01	1.40E+01	3.76E-02	**3.71E-02**	2.51E+01	2.69E+01
F17	mean	**8.46E-01**	8.54E-01	5.31E+01	5.18E+01	**8.81E-01**	8.92E-01	6.28E+01	6.12E+01
	min	7.49E-01	**7.41E-01**	2.20E+01	3.35E+01	**7.71E-01**	8.13E-01	5.05E+01	3.79E+01
	max	**9.62E-01**	9.83E-01	8.01E+01	6.92E+01	9.79E-01	**9.50E-01**	7.14E+01	8.08E+01
	std	**5.24E-02**	5.83E-02	1.36E+01	8.43E+00	4.55E-02	**4.32E-02**	5.45E+00	9.04E+00
F18	mean	1.42E+01	**1.40E+01**	2.66E+02	2.71E+02	**1.47E+01**	1.49E+01	3.22E+02	3.21E+02
	min	1.27E+01	**1.16E+01**	2.08E+02	2.22E+02	1.37E+01	**1.34E+01**	2.78E+02	2.71E+02
	max	1.57E+01	**1.54E+01**	3.34E+02	3.22E+02	1.65E+01	**1.59E+01**	3.85E+02	3.71E+02
	std	**8.00E-01**	8.88E-01	2.55E+01	2.47E+01	**6.18E-01**	6.97E-01	2.52E+01	2.54E+01

As indicated in Tables 7-8, the evading strategy contributes to the superior performance of the proposed FA model. As can be observed, the results from the version of FA + the evading strategy are very close as compared with those from the full version of the proposed FA model with both strategies. The attraction mechanism and the original FA model show great achievements in solving shifted functions 6, 11 and 12 for both dimensions 30 and 50, and outperform the evading strategy for these benchmark functions. Although, in most of the cases, purely using the proposed attraction mechanism shows similar performances as compared with those of the original FA model, when combined with the evading strategy, it boosts the performance of the full version of the proposed FA model slightly in comparison with those obtained using the evading mechanism only in the evaluation of some standard and complex benchmark functions.

In short, the empirical results indicate the efficiency of the proposed evading and attraction mechanisms in comparison with the original FA model. In addition to taking the neighbouring worst solutions into account, the evading strategy employs the global worst solution identified from each iteration as another primary signal to advise the search process, and enable the swarm to move away from unpromising search regions effectively, therefore accelerating convergence. Furthermore, both the attraction and evading mechanisms work in a cooperative manner in the proposed FA model to mitigate premature convergence suffered by the original FA model. As an example, when there is no brighter firefly in the neighbourhood, the original FA model becomes stagnant. In the proposed FA model, when the attraction action stagnates, the evading movement employs the local and global worst signals to divert the search towards optimal regions, in order to avoid stagnation. On the other hand, when the evading action

shows limited improvements, the attraction behaviour is able to lead the search by following local and global optimal signals to drive the search out of local optima. Therefore, the full version of the proposed FA model with both strategies and the version of FA + the evading mechanism achieve more superior performances than those using solely the attraction action and the original FA model.

4.4 Evaluation Using the Black-Box Optimization Benchmarking Test Suite

Another evaluation using the well-known BBOB benchmark test suite has been conducted. We mainly focus on the noisy BBOB cases which represent diverse challenging optimization problems. The MATLAB code in the COCO (COmparing Continuous Optimizers) platform is used, which provides benchmark function testbeds and experimental templates that depict sound comparisons of global optimizers (Hansen et al., 2012). The COCO platform has been used for BBOB competitions associated with the GECCO and CEC special sessions. In our experiments, the maximum number of function evaluations per function is set to 80000D (slightly lower than 10^5D) for the proposed algorithm, where D is the dimension. Therefore, the iteration number and the population size fulfil the following condition, i.e. maximum iterations × population size ≈ 80000D. No parameter tuning is conducted for the proposed algorithm. We evaluate the noisy BBOB benchmark problems with dimension settings of 2, 3, 5, 10, and 20, because the full results from these settings are available for performance comparison across 2009-2017 BBOB competitions.

In the first comparison study, a total of 11 state-of-the-art algorithms have been selected for comparison. These methods are selected because they demonstrate impressive performance for solving noisy BBOB problems, and are top performers for BBOB 2009, 2010 and 2012 competitions. Some of them employ a slightly higher number of function evaluations, i.e. 10^5D, than that of the proposed algorithm. In other cases, higher or lower numbers of function evaluations per function, i.e. 10^6D or 10^4D, are applied. The selected methods include the best 2009 optimizer (provided automatically by the COCO platform) (Hansen et al., 2010a), Separable Natural Evolution Strategies (SNES) (Schaul, 2012a), Exponential NES (xNES) (Schaul, 2012b), xNES with Adaptation Sampling (xNESas) (Schaul, 2012c), Differential Ant-Stigmergy Algorithm (DASA) (Korošec and Šilc, 2009), Simultaneous Perturbation Stochastic Approximation (SPSA) (Finck and Beyer, 2010), PSO hybridized with Estimation of Distribution Algorithm (EDA) (EDA-PSO) (El-Abd and Kamel, 2009a), PSO with adaptive bounds (PSO_Bounds) (El-Abd and Kamel, 2009b), PSO incorporated with DE (DE-PSO) (García-Nieto et al., 2009), BayEDA$_{cG}$ (Gallagher, 2009) and the Pure-Random-Search algorithm (RANDOMSEARCH) (Auger and Ros, 2009).

Among the selected methods, the best 2009 optimizer outperforms all other methods for BBOB 2009 competition. To date, it still is the best optimizer in tackling all problem dimensions for the BBOB test functions (Hansen et al., 2010a). SNES, xNES and xNESas are variants of the Natural Evolution Strategies (NES). xNES employs a search distribution parameterized by a mean vector and a full covariance matrix. On the other hand, xNESas applies Adaptation Sampling for fine-tuning its learning rates, in order to accelerate convergence (Schaul 2012b; 2012c). According to Schaul (2012b; 2012c), xNES and xNESas are the most appropriate NES variants for tackling noisy, non-separable BBOB problems with small dimensions. In addition, SNES scales linearly with the problem dimension with fast convergence, and is regarded as the most appropriate NES variant for noisy, large separable BBOB problems (Schaul 2012a). PSO variants, i.e. PSO_Bounds (El-Abd and Kamel, 2009b), EDA-PSO (El-Abd and Kamel, 2009a), DE-PSO (García-Nieto et al., 2009), and other methods such as DASA (Korošec and Šilc, 2009) and BayEDA$_{cG}$ (Gallagher, 2009) embed diverse search strategies to overcome local optima traps and show competitive performance in tackling BBOB problems. Overall, owing to their impressive performances for solving BBOB test functions with moderate and severe noise levels, the abovementioned methods are selected for comparison in this research.

In addition to the 11 methods, we include additional 10 optimizers in the second study, in order to conduct a wider comparison between the proposed algorithm and other methods in the BBOB community. The majority of these 10 additional methods use a different number of function evaluations, e.g. 10^6D, 10^7D or 10^4D, as compared with that of the proposed FA model. Although in some cases, they are not directly comparable, the results from these methods serve as general performance indicators for solving BBOB problems. These 10 methods are: Adapted Maximum-Likelihood Gaussian Model Iterated Density-Estimation Evolutionary Algorithm (AMaLGaM) (Bosman et al., 2009), Age-Layered Population Structure (ALPS) (Hornby, 2009), Variable Neighbourhood Search (VNS) (García-Martínez and Lozano, 2009), an independent-restart implementation of covariance matrix adaptation (CMA) combined with the simple (1+1) selection scheme (CMA-ES) (Auger and Hansen, 2009), stable noisy optimization by branch and fit (SNOBFIT) (Huyer and Neumaier, 2009a), a modified multistart clustering global optimization

method (GLOBAL) (Pal et al., 2012), NEW Unconstrained Optimization Algorithm (NEWUOA) (Ros, 2010), NEWUOA with a maximum number of interpolation points (full-NEWUOA) (Ros, 2010), multilevel coordinate search (MCS) (Huyer and Neumaier, 2009b), and the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method (Ros, 2009). Among the 10 methods, ALPS, AMaLGaM and VNS show impressive performances with comparatively higher numbers of function evaluations. ALPS (Hornby, 2009) splits the population into multiple layers using a novel measure of age. Newly generated random individuals are inserted into the bottom layer regularly to overcome premature convergence. AMaLGaM (Bosman et al., 2009) embeds both the strategy for scaling up the covariance matrix and the mechanism for anticipating the mean shift, in an attempt to overcome stagnation and accelerate convergence. It outperforms its incremental-learning variant in solving noisy BBOB problems. VNS (García-Martínez and Lozano, 2009) incorporates three evolutionary algorithms designed for generating a good starting point, exploiting local information, and enhancing local search diversity, respectively. It shows great superiority over other state-of-the-art methods in solving noisy BBOB problems.

The noisy BBOB test suite has a total of 30 functions, which include 6 unimodal functions with moderate noise (f101-f106), 15 functions with severe noise (f107-f121), and 9 highly multimodal functions with severe noise (f122-f130) (Hansen et al., 2010b). Each function has 15 instances. The functions with moderate noise include Sphere and Rosenbrock with moderate Gaussian, uniform, and seldom Cauchy noise, while the functions with severe noise include Sphere, Rosenbrock, Step ellipsoid, Ellipsoid, and Different Powers with Gaussian, uniform and seldom Cauchy noise. The highly multimodal functions with severe noise include Schaffer's F7, Composite Griewank-Rosenbrock and Gallagher's Gaussian Peaks 101-me with Gaussian, uniform and seldom Cauchy noise. Figure 1 (automatically generated by the COCO platform) illustrates the empirical cumulative distribution of runtime of each algorithm on all functions f101–f130 and one subgroup (i.e. the severe noisy multimodal subgroup) in dimension 2, for comparison using 11 and 21 selected methods, respectively. Each target precision value, Δf_t, of $\{10^{1.8}, 10^{1.6}, 10^{1.4}, \dots, 10^{-8}\}$, is employed for evaluation, which is denoted as $\Delta f_t \in]10^{2}, 10^{-8}]$.

Overall, Figure 1 represents the proportion of problems solved within a given budget in dimension 2. The *x* axis represents a given budget, i.e. a given number of function evaluations divided by the dimension, while the *y* axis denotes the proportion of the problems where the target precision value is achieved within the given budget (Hansen et al., 2010a). Crosses are used to mark the maximum number of function evaluations from a specific algorithm. In Figure 1, the results of the best 2009 optimizer (provided by the COCO platform automatically) are used as the baseline reference, while the results from other methods are provided by the respective authors as extracted from the BBOB website. In Figure 1, the top two diagrams indicate the performance comparison for all 30 test functions and the subgroup of 9 highly multimodal functions with severe noise using 11 selected methods, respectively, while the bottom two diagrams represent the same comparisons using 21 selected methods.

The results extracted from the BBOB website indicate that a diverse number of function evaluations have been applied to different methods. In the first study, we employ several related methods which have a similar number of function evaluations per function as that of the proposed algorithm. EDA-PSO, PSO_Bounds, and SPSA employ a slightly higher maximum number of function evaluations, i.e. 10^5D, than that of the proposed algorithm. To indicate the efficiency of the proposed FA model, the results from several methods with larger numbers of function evaluations have also been selected, e.g. DASA and RANDOMSEARCH with the maximum of 10^6D number of function evaluations per function. xNES and xNESas use the maximum of 10^5D (or slightly lower) number of function evaluations for dimensions 2 and 3, but higher numbers of function evaluations between 10^5D and 10^6D for dimensions 5, 10 and 20. Other selected methods such as DE-PSO, SNES and BayEDA$_{cG}$ employ the maximum of 10^4D number of function evaluations.

For the additional 10 methods, VNS uses 10^7D as the maximum number of function evaluations while AMaLGaM and ALPS employ 10^6D number of function evaluations per function. NEWUOA, full-NEWUOA, MCS and CMA-ES set the maximum number of function evaluations to 10^4D with the rest of the methods applying smaller numbers of function evaluations.

As indicated in Figure 1 (c), in a general context of comparison based on all 30 test functions, AMaLGaM, VNS and ALPS show the best performance, but with comparatively higher numbers of function evaluations, i.e. 10^6D, 10^7D, and (slightly over) 10^6D, respectively. Other methods such as xNES, xNESas, PSO_Bounds, EDA-PSO and the proposed algorithm show comparable performance with 10^5D or $80000D$ number of function evaluations. As indicated in Figure 1 (a), xNES, xNESas, EDA-PSO and the proposed algorithm solve a larger proportion of

problems than that of PSO_Bounds within 80000D number of function evaluations. With a higher number of function evaluations, PSO_Bounds achieves more competitive performance than those from xNES, xNESas, and EDA-PSO. The proposed algorithm also outperforms SPSA, DASA and RANDOMSEARCH, although these methods use a higher number of function evaluations of 10^5D or 10^6D.

Overall, as indicated in Figure 1 (a) and (c), the proposed algorithm solves a similar proportion of problems as compared with those of other state-of-the-art methods such as xNES, xNESas, and EDA-PSO, and outperforms at least the following methods, i.e. DASA, RANDOMSEARCH, and SPSA, in dimension 2, with a similar or lower number of function evaluations as those of xNES, xNESas, EDA-PSO, and SPSA, and a significantly lower number of function evaluations than those of DASA and RANDOMSEARCH. Note that no parameter tuning is applied to the proposed algorithm.

Figure 1 (b) and (d) show the performance from 11 and 21 selected algorithms pertaining to highly multimodal functions with severe noise in dimension 2, respectively. As indicated in Figure 1 (d), the proposed algorithm shows great efficiency in solving such noisy multimodal functions. Within the budget of 80000D number of function evaluations (indicated by the pink cross), it shows comparable performance as compared with those of xNES, DE-PSO and EDA-PSO, and solves a larger proportion of problems than at least those of PSO_Bounds, xNESas, DASA, RANDOMSEARCH and SPSA.

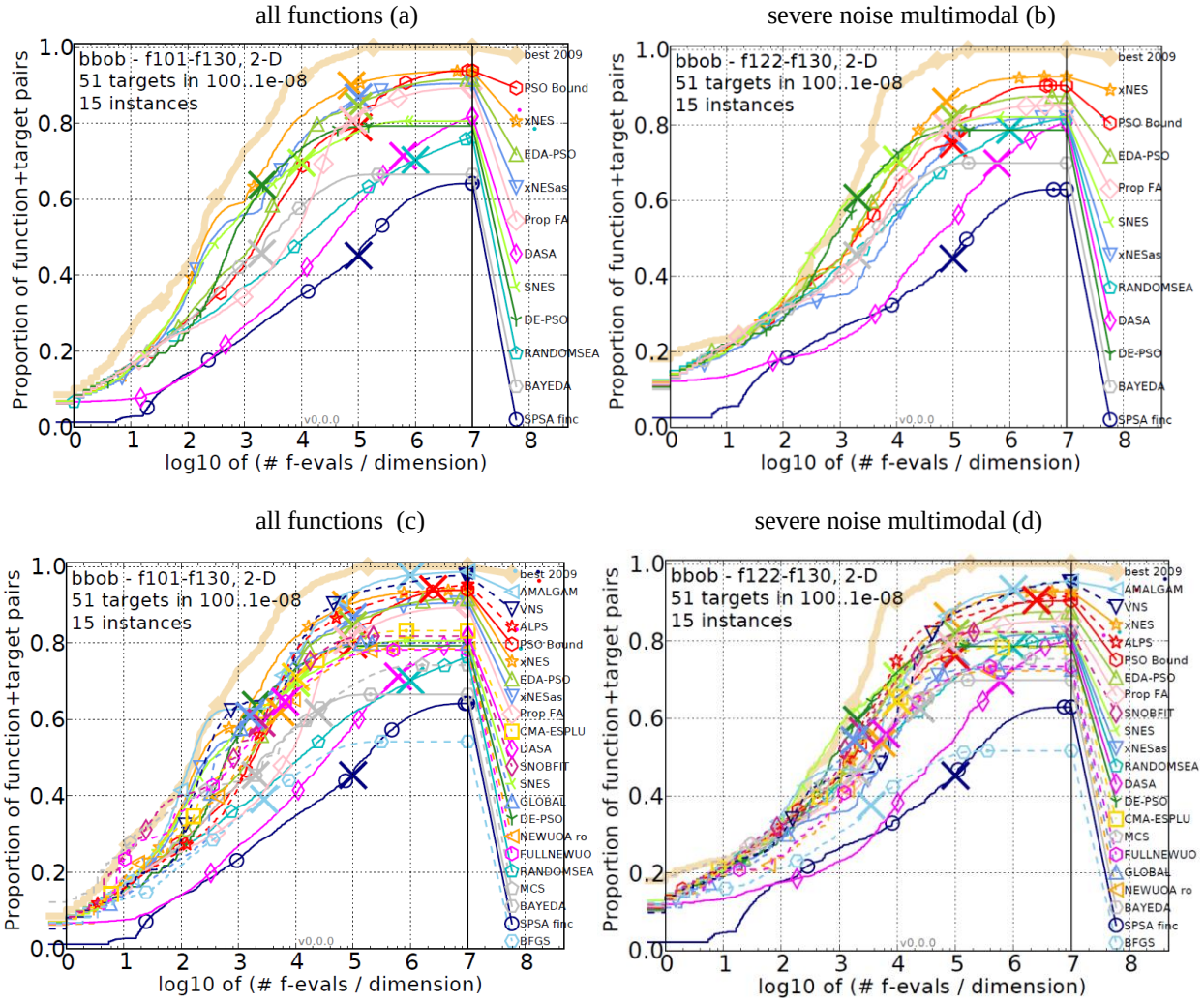


Figure 1 Bootstrapped empirical cumulative distribution of the number of objective function evaluations divided by dimension (FEvals/DIM) for 51 targets with target precision in $10^{[-8..2]}$ for all functions and one subgroup in 2-D.

The "best 2009" line corresponds to the best aRT observed during BBOB 2009 for each selected target (the top two diagrams present the results from 11 selected methods and the bottom diagrams present the results from 21 selected methods)

The empirical results also indicate the proposed algorithm shows improvements in comparison with other state-of-the-art methods for higher dimensions. Figure 2 shows the empirical cumulative distribution of runtime of each algorithm on all functions and one subgroup (i.e. the severe noisy multimodal subgroup) in dimension 3, in comparison with 11 and 21 related methods, respectively. As indicated in Figure 2 (a) and (c), for all 30 functions, the proposed algorithm solves a comparatively larger proportion of problems than at least those of the state-of-the-art methods such as PSO_Bounds, EDA-PSO, DASA, RANDOMSEARCH and SPSA, with a comparatively lower budget. As shown in Figure 2 (b) and (d), the proposed FA model shows great efficiency in solving the 9 multimodal functions with severe noise, and outperforms at least xNESas, EDA-PSO, RANDOMSEARCH, DASA, and SPSA, by employing similar or lower numbers of function evaluations.

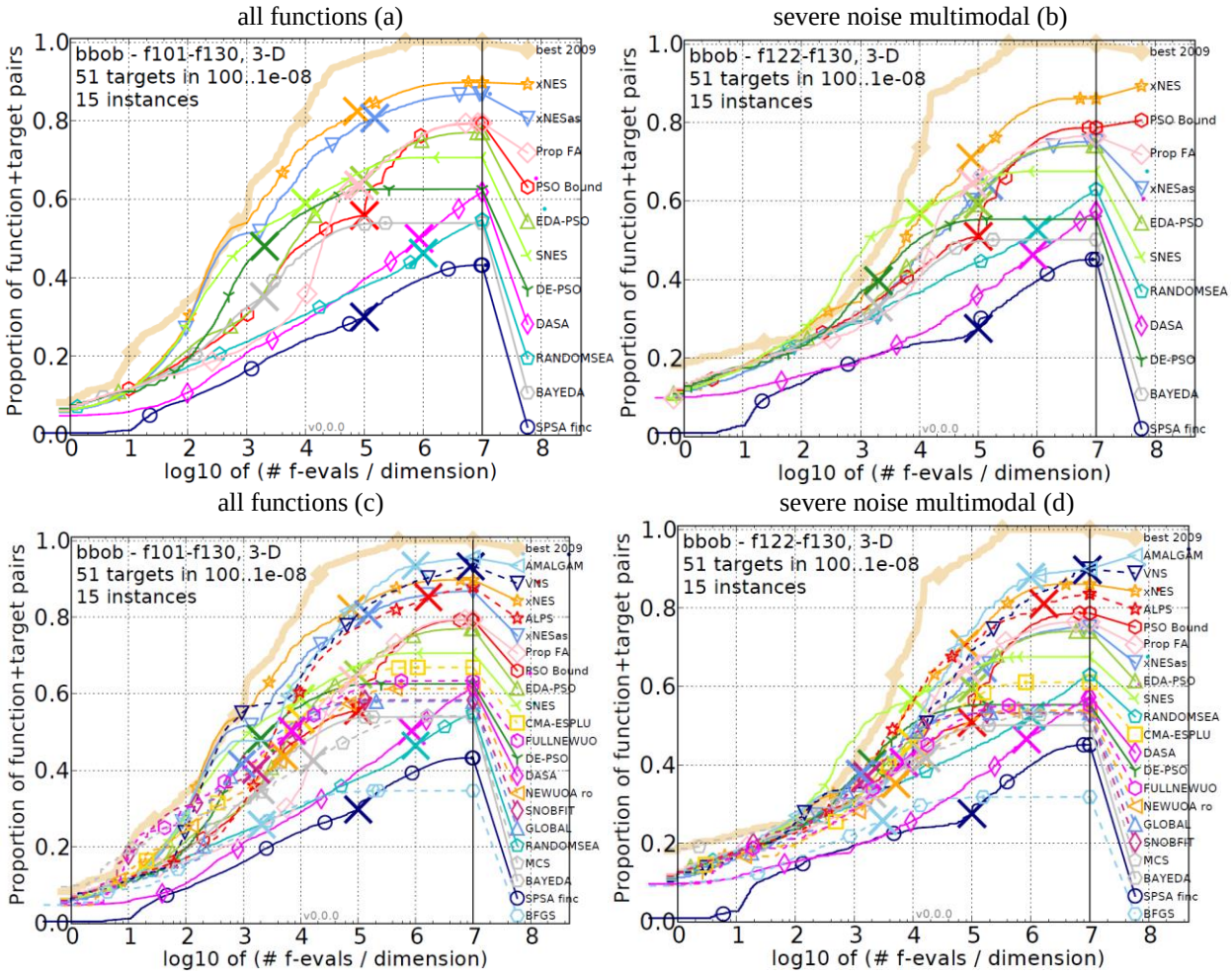


Figure 2 Bootstrapped empirical cumulative distribution of the number of objective function evaluations divided by dimension (FEvals/DIM) for 51 targets with target precision in $10^{[-8..2]}$ for all functions and one subgroup in 3-D. The "best 2009" line corresponds to the best aRT observed during BBOB 2009 for each selected target (the top two diagrams present the result from 11 selected methods and the bottom diagrams present the results from 21selected methods)

In dimensions 5, 10 and 20, the problems become even more challenging and all algorithms demonstrate drastic performance degradation, with AMaLGaM, xNESas and xNES performing the best with comparatively higher numbers of function evaluations (10^6D and over 10^5D). The NES variant, SNES, also shows impressive scalability

with 10^4D number of function evaluations. In addition, xNESas and xNES have increased the number of function evaluations for these three dimensions (i.e. over 10^5D). The top two diagrams in Figure 3 show the detailed empirical cumulative distributions of runtime of each algorithm on all functions and the subgroup of unimodal functions with severe noise, in dimensions 5, with respect to the 21 selected methods, while the bottom two diagrams show similar distributions on all functions in dimensions 10 and 20. Besides the performances obtained using 80000D number of function evaluations, we provide the performances of the proposed algorithm with a slightly higher number of function evaluations, i.e. 10^5D, for these three dimensions. As shown in Figure 3 (a), (c) and (d), the proposed algorithm achieves similar performances under both numbers of function evaluations for dimensions 5, 10 and 20. It is able to outperform at least PSO_Bounds, DASA, RANDOMSEARCH and SPSA for these three dimensions, consistently, with a budget of 80000D function evaluations. Besides that, our algorithm shows impressive capability of dealing with the 15 functions with severe noise, as illustrated in Figure 3(b), and outperforms at least PSO_Bounds, EDA-PSO, DASA, RANDOMSEARCH, and SPSA in dimension 5, with 80000D number of function evaluations, i.e., similar or lower numbers of function evaluations than those of other methods. A similar observation is also obtained for dimension 20 in solving functions with severe noise with a comparatively lower budget, i.e. 80000D. On top of this, with 10^5D number of functions evaluations, the proposed algorithm outperforms at least EDA-PSO, PSO_Bounds, DASA, RANDOMSEARCH, and SPSA consistently for the three dimensions.

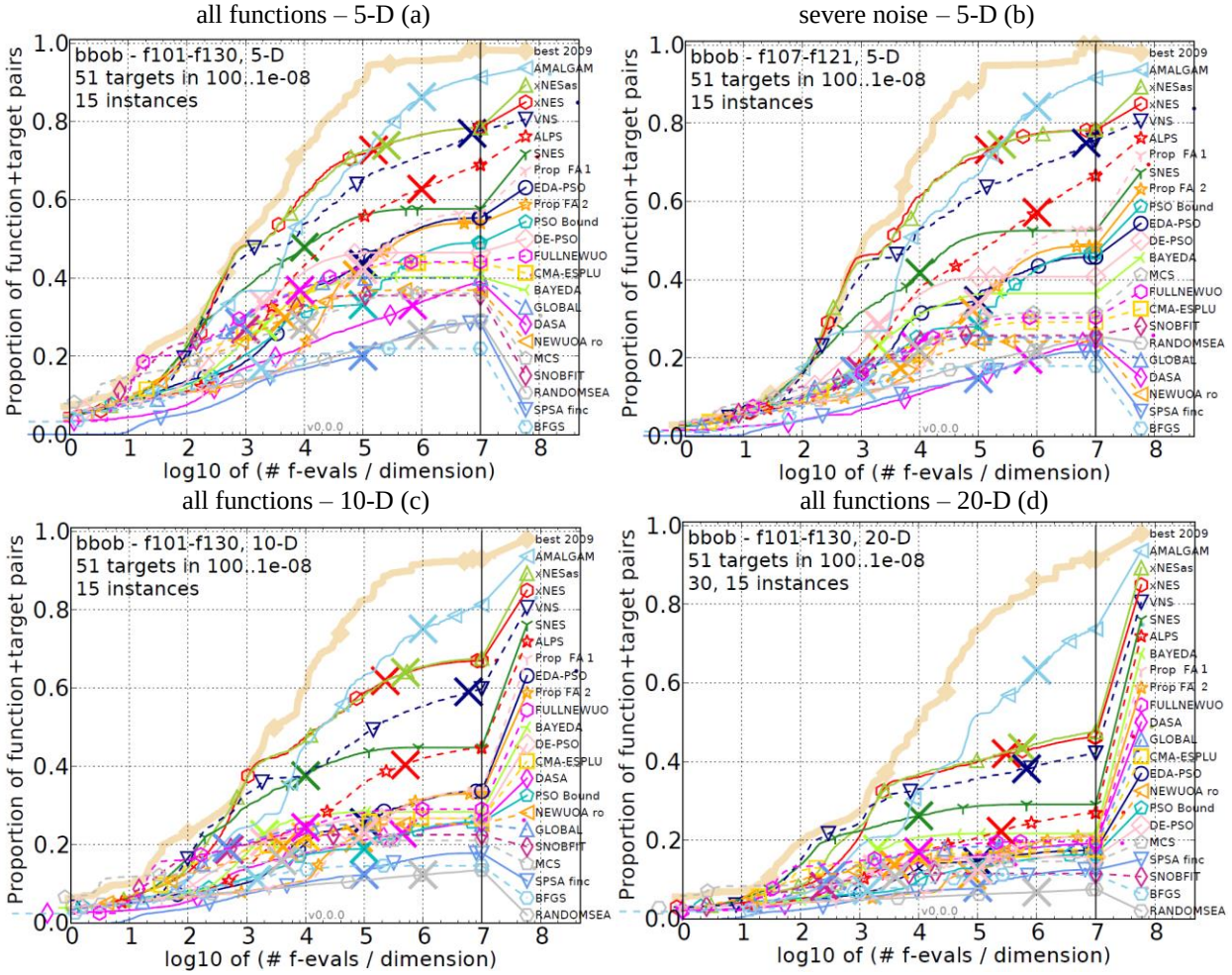


Figure 3 Bootstrapped empirical cumulative distribution of the number of objective function evaluations divided by dimension (FEvals/DIM) for 51 targets with target precision in $10^{[-8..2]}$ for all functions in 5-D, 10-D and 20-D, and one subgroup in 5-D, in comparison with the 21 selected methods. The "best 2009" line corresponds to the best aRT observed during BBOB 2009 for each selected target. (Prop. FA 1 and Prop. FA 2 denote the proposed algorithm with 10^5D and 80000D numbers of function evaluations, respectively.)

Overall, the empirical results indicate that the proposed algorithm is among the top performers, and shows impressive performance for tackling noisy BBOB benchmark problems. In comparison with other state-of-the-art methods (such as EDA-PSO, PSO_Bounds, DASA, etc), it shows comparable or better results for evaluation of especially the 15 unimodal and 9 highly multimodal functions with severe noise in nearly all five test dimensions with similar or lower numbers of function evaluations.

4.5 Ensemble Reduction Evaluation

For ensemble reduction, we use three high dimensional data sets from the UCI machine learning repository and one medical image data set for evaluation. Since LSFA tends to outperform SFA for many real-life optimization problems (Alweshah and Abdullah, 2015), we employ LSFA together with NaFA, HFDE and CFA as the FA variants for ensemble reduction evaluation. Other advanced and conventional search methods, i.e. ELPSO (Jordehi, 2015), PSO, GA and FA, have also been selected owing to their efficiency and robustness in dealing with diverse dimensionality reduction and feature optimization problems (Jordehi, 2015; Jothi and Inbarani, 2016, Mistry et al., 2016; Yang, 2008; Zhang et al., 2016b). Table 9 shows the detailed information of the employed data sets. Specifically, the first three data sets in Table 9 are taken from the UCI machine learning repository, while the last skin lesion image data set is obtained from the Dermofit Image Library (Ballerini et al., 2013). The SVM with Radial Basis Function (RBF) kernel is used as the sole base classifier in this study. Moreover, we compare ensemble models generated by all search methods, with the original full-sized, unreduced ensembles as well as single base classifiers in each experiment.

4.5.1 Base Classifier Pool Generation

First of all, we introduce base classifier pool generation for ensemble reduction evaluation. Since different feature subsets of the training data provide different representations of the same problem and, typically, may lead to different classification outcomes (Christoudias et al., 2008; Diao et al., 2014), we use different feature subsets to generate diverse base models for each test data set in this research. Although many developments (Diao et al., 2014; Guan et al., 2015) employ the random subspaces method for feature subset selection in base model generation, its feature selection process is conducted randomly without an objective evaluation of the selected feature subsets using the class label information. Therefore, such randomly selected feature subsets tend to be less discriminant, resulting in low recognition accuracy rates with high computational costs (Guan et al., 2015; Ho, 1998). Instead of using randomly selected feature subsets, in this research, a wrapper PSO-based feature selection method integrated with the SVM classifier is used to identify diverse optimal discriminative feature subsets for each data set in base classifier pool generation.

Table 9 Data sets used for ensemble reduction evaluation

Data sets	Features	Instances	Classes
sonar (UCI M.L. repository)	60	208	2
ozone (UCI M.L. repository)	72	324	2
libras (UCI M.L. repository)	90	360	15
skin lesion image data set (The Dermofit Image Library)	98	1300	2

This PSO-based feature selection process is initialized with a population of particles. Each particle has N dimensions, where N is the number of features for each test data set. The feature selection process is guided by the abovementioned fitness function defined in Equation (6). Specifically, each particle represents one selected feature subset, and is evaluated by the fitness function during the search process. The global best solution identified by PSO, which represents the most optimal feature subset, is used as the output. Multiple trials of the PSO-based feature selection process are conducted in order to generate multiple feature subsets for base classifier pool generation. Different experimental settings of PSO-based feature selection are also used to generate base models with diversity. Overall, we conduct 30 and 50 trials of the feature selection process to obtain 30 and 50 optimal feature subsets, respectively. Since these feature subsets are obtained by using the fitness evaluation of the metaheuristic search process, they lead to reasonable variations in classification accuracy. Therefore, in this research, we construct 30 and 50 base models, respectively, using the above two sets of generated optimal feature subsets for ensemble reduction evaluation.

In this research, the RBF-based SVM is used as the base classifier. To achieve the optimal classification accuracy rate of each base classifier, we use a grid search to identify optimal parameter settings during the training stage.

Since the combination of the following parameters, i.e. the soft-margin constant, *C*, the kernel parameter, *gamma*, and the *epsilon*, has great impact on the performance of each base classifier, the ranges of $[2^{-5} – 2^{15}]$, $[2^{-10} – 2^{5}]$, and $[2^{-8} – 2^{-1}]$ are used to search for the optimal parameter settings, respectively. A 10-fold cross validation is used to guide the best parameter search, and avoid overfitting. The optimal parameter setting that leads to the best classification accuracy score for a training data set is subsequently used for evaluation of the corresponding test data set. The weighted majority voting method is used to generate the final classification result of each identified ensemble based on the outputs of each base model.

4.5.2 Evaluation Results for Ensemble Reduction

To evaluate the ensemble reduction problem, a series of experiments with diverse numbers of iterations have been conducted. Since the number of base models is small, a large number of iterations tend to reduce the ensemble size drastically (e.g. evaluated with 30 base models, FA selects an average number of 1.8 base classifiers over 30 runs, when 100 iterations are applied). To strike the best trade-off between classification accuracy and ensemble size, we employ 30 as the maximum number of iterations. Therefore, the following experimental settings are applied to each method, i.e. population size=30, and dimension=30 or 50 (representing 30 or 50 base models). The classical search methods, CFA, NaFA and the proposed FA model, employ the following number of function evaluations, i.e. 30 (population size) × 30 (maximum number of generations) = 900. Again, LSFA, HFDE and ELPSO utilize comparatively higher numbers of function evaluations. For instance, LSFA uses the following number of function evaluations, i.e. 30 (population size) × 30 (maximum number of generations) + 20 (maximum trials of SA) × 30 (maximum number of generations) = 1500. The number of function evaluations for HFDE is (2 × 30 (population size) + 3 offspring generated using FA) × 30 (maximum number of generations) = 1890. The number of function evaluations for ELPSO is (30 (population size) + 5 (trials for the improvement of the global best solution)) × 30 (maximum number of generations) = 1050. The most optimal solution identified by each search method is the recommended base model selection for ensemble generation.

Table 10 Evaluation results for the sonar data set using 30 and 50 base models, respectively

sonar	Methods	Selected average number of base models	SVM-based Ensemble (10-fold) 30 runs	SVM-based Ensemble (hold-out) 30 runs
30 base models	**FA**	5.33	0.8572	0.8471
	PSO	7.57	0.8578	0.8431
	GA	20.20	0.8635	0.8554
	ELPSO	7.53	0.8588	0.8451
	LSFA	5.47	0.8593	0.8451
	CFA	4.80	0.8481	0.8407
	NaFA	5.37	0.8417	0.8328
	HFDE	5	0.8381	0.8382
	Prop. FA	**6.87**	**0.8914**	**0.8824**
	Full size	30	0.8833	0.8824
	Base model	1	0.7765	0.7775
50 base models	**FA**	11.13	0.8466	0.8387
	PSO	15.17	0.8499	0.8363
	GA	32.83	0.8566	0.8471
	ELPSO	14.73	0.8497	0.8353
	LSFA	10.87	0.8518	0.8382
	CFA	9.87	0.8460	0.8319
	NaFA	11.2	0.8435	0.8328
	HFDE	9	0.8690	0.8676
	Prop. FA	**13.93**	**0.8926**	**0.8828**
	Full size	50	0.8524	0.8529
	Base model	1	0.7721	0.7726

As shown in Table 9, the following three benchmark data sets from the UCI machine learning repository (Bache and Lichman, 2013) are employed for ensemble reduction evaluation, i.e. sonar, ozone, and libras. The sonar data set has 208 instances with two classes. Each instance is represented by a 60-dimension feature vector. We employ 140 and 68 samples for training and test, respectively. Both training and test data sets have balanced class samples. Based on a training set of 140 samples, PSO-based feature selection is conducted to generate two sets of 30 and 50 optimal feature subsets to build 30 and 50 base models, respectively. These base classifiers are used for ensemble reduction evaluation. Table 10 shows the detailed comparison results between the proposed algorithm and other search methods for ensemble optimization using a test set of 68 instances. To have a fair comparison, a total of 30 trials are

conducted for each method for ensemble generation, and the average accuracy rates of 30 ensembles generated by each method are used for comparison. The accuracy rates of the original full-sized ensemble and the mean accuracy rates of all base models are also provided for comparison. Note that the mean accuracy rates of all base models are calculated by averaging the results of all 30 or 50 base classifiers. They are used to indicate the average performance of a single SVM classifier.

Note the above comparison is conducted based on the average classification accuracy obtained from 30 runs for each test method. In other words, for each trial, one ensemble classifier is constructed based on the base model selection result from each algorithm. This base model selection process is performed 30 times to generate the corresponding 30 ensembles for each method. The average accuracy rates of these 30 ensembles are used for comparison. As indicated in Table 10, for evaluation of 30 base models, using 10-fold cross validation, the ensembles constructed by the proposed FA model achieve the best average accuracy score of 89.14% over 30 runs, and outperform those generated by FA, PSO, GA, ELPSO, LSFA, CFA, NaFA and HFDE by 3.42%, 3.36%, 2.79%, 3.26%, 3.21%, 4.33%, 4.97% and 5.33% respectively. When evaluated using hold-out validation, the ensemble classifiers built by the proposed algorithm have the best average accuracy score of 88.24% over 30 runs, which outperform FA, PSO, GA, ELPSO, LSFA, CFA, NaFA and HFDE by 3.53%, 3.93%, 2.7%, 3.73%, 3.73%, 4.17%, 4.96% and 4.42% respectively. Our generated ensembles have the same or even slightly higher mean accuracy rates than those obtained using the full-sized, unreduced ensemble classifier integrating all 30 base models for hold-out and 10-fold cross validation tests, respectively.

For evaluation using 50 base models, the ensemble classifiers generated by our algorithm have the highest mean accuracy score of 89.26% over 30 runs for 10-fold cross validation. It outperforms FA, PSO, GA, ELPSO, LSFA, CFA, NaFA and HFDE by 4.6%, 4.27%, 3.6%, 4.29%, 4.08%, 4.66%, 4.91% and 2.36% respectively. The models produced by the proposed FA model also obtain the best average result score of 88.28% using the hold-out validation test, which outperform those generated by FA, PSO, GA, ELPSO, LSFA, CFA, NaFA and HFDE by 4.41%, 4.65%, 3.57%, 4.75%, 4.46%, 5.09%, 5% and 1.52% respectively, over 30 runs. The average performances of our recommended ensembles also outperform those of the full-sized ensemble embedding all 50 base models by 4.02% and 2.99% for 10-fold and hold-out validation tests, respectively. The empirical results also indicate that the ensembles constructed by the proposed FA model achieve the best trade-off between classification accuracy and ensemble size in comparison with those produced by other search methods. The proposed algorithm reduces the ensemble size, while maintaining or improving classification accuracy simultaneously.

Table 11 Evaluation results for the ozone data set using 30 and 50 base models, respectively

ozone	Methods	Selected average number of base models	SVM-based Ensemble (10-fold) 30 runs	SVM-based Ensemble (hold-out) 30 runs
30 base models	**FA**	5.23	0.8466	0.8452
	PSO	6.90	0.8594	0.8540
	GA	24.70	0.8620	0.8532
	ELPSO	6.93	0.8656	0.8595
	LSFA	5.43	0.8598	0.8575
	CFA	4.03	0.8492	0.8452
	NaFA	5.17	0.8524	0.8496
	HFDE	4	0.8444	0.8452
	Prop. FA	**6.7**	**0.8758**	**0.8742**
	Full size	30	0.8583	0.8452
	Base model	1	0.8301	0.8302
50 base models	**FA**	11.33	0.8569	0.8500
	PSO	14.57	0.8580	0.8504
	GA	32.93	0.8594	0.8520
	ELPSO	14.97	0.8570	0.8484
	LSFA	11.60	0.8569	0.8544
	CFA	10.37	0.8515	0.8389
	NaFA	11.37	0.8535	0.8512
	HFDE	10	0.8583	0.8452
	Prop. FA	**14.17**	**0.8791**	**0.8750**
	Full size	50	0.8597	0.8571
	Base model	1	0.8278	0.8273

Another data set, i.e. ozone, from the UCI machine learning repository (Bache and Lichman, 2013) is also used for evaluation. The ozone data set has 324 instances with two classes. Each instance is represented by a 72-dimension

vector. We use 280 and 84 instances for training and test, respectively. A balanced number of instances for both classes are used in the training and test sets. PSO-based feature selection is used to identify 30 and 50 optimal feature subsets using the training set to construct 30 and 50 base classifiers, respectively. The test set is used to evaluate the efficiency of the identified ensemble models recommended by each algorithm. A total of 30 runs are conducted for each method for ensemble reduction, and the average accuracy rates of the constructed ensembles over 30 runs are used for comparison. The detailed evaluation results for the ozone data set are provided in Table 11.

As illustrated in Table 11, when 30 base models are used for ensemble member selection, the ensemble classifiers produced by the proposed FA model achieve the highest average accuracy score of 87.58% using 10-fold cross validation over 30 runs, which outperforms those of the models generated by FA, PSO, GA, ELPSO, LSFA, CFA, NaFA and HFDE by 2.92%, 1.64%, 1.38%, 1.02%, 1.6%, 2.66%, 2.34% and 3.14% respectively. Using hold-out validation, the proposed algorithm obtains the best average classification result of 87.42% over 30 runs for ensemble construction, and outperforms FA, PSO, GA, ELPSO, LSFA, CFA, NaFA and HFDE by 2.9%, 2.02%, 2.1%, 1.47%, 1.67%, 2.9%, 2.46% and 2.9% respectively. The ensemble classifiers built by the proposed algorithm also outperform the unreduced, full-sized ensemble model by 1.75% and 2.9% on average for the 10-fold and hold-out validation tests, respectively.

When evaluated using 50 base models, the ensemble classifiers constructed by our algorithm achieve the best classification accuracy scores of 87.91% and 87.50% for the 10-fold and hold-out validation tests, respectively, over 30 trials. Using 10-fold cross validation, our ensembles outperform those from FA, PSO, GA, ELPSO, LSFA, and CFA, NaFA and HFDE by 2.22%, 2.11%, 1.97%, 2.21%, 2.22%, 2.76%, 2.56% and 2.08% respectively, whereas using the hold-out validation, our recommended models outperform those built by FA, PSO, GA, ELPSO, LSFA, CFA, NaFA and HFDE by 2.5%, 2.46%, 2.3%, 2.66%, 2.06%, 3.61%, 2.38% and 2.98% respectively, over 30 runs. The constructed ensemble models also outperform the original unreduced ensemble classifier using all 50 base SVM classifiers by 1.94% and 1.79%, on average, for the 10-fold and hold-out validation tests, respectively.

The libras data set from UCI machine learning repository (Bache and Lichman, 2013) is also employed for evaluation. It has 360 samples from 15 classes, with 24 instances in each class. Each instance has 90 dimensions. We employ the first 5 classes for evaluation. We use two distinctive sets of 84 samples from seven classes (including the 5 reported classes and another two classes as noise data) for training and test, respectively. Both training and test sets contain balanced class samples. Moreover, for each class, PSO-based feature optimization is conducted to identify two sets of optimal feature subsets, i.e. 30 and 50, for producing 30 and 50 base models, respectively. A total of 30 runs are conducted for each method in each experimental setting. Table 12 shows the detailed evaluation results.

As illustrated in Table 12, for evaluation of each class, the proposed FA-based ensemble reduction classifiers outperform those generated by other search methods, as well as the majority of the corresponding full-sized ensemble classifiers and single base classifiers, significantly. The classification results indicate the efficiency of the proposed FA model for striking the best trade-off between classification accuracy and ensemble size among all methods.

To further ascertain the efficiency of our algorithm, another medical image data set, i.e. the Edinburgh Research and Innovation (Dermofit) skin lesion data set (Ballerini et al., 2013), is used for evaluation. It includes 1300 dermoscopic skin lesion images with 850 benign and 450 melanoma cases. The feature extraction process of our previous research (Tan et al., 2016) is used to extract shape, colour, size, and lesion-edge features. Overall, each image is represented by a 98-dimension feature vector. In this research, we employ 660 and 98 images for training and test, respectively. PSO-based feature selection is also used to generate 30 and 50 base models, respectively. A total of 30 runs are conducted for evaluation of each method in ensemble generation. The detailed average accuracy rates over 30 runs for each method for the two experimental settings are illustrated in Table 13.

Table 12 Evaluation results for the libras data set using 30 and 50 base models, respectively

libras		Methods	Selected average number of base models	SVM-based Ensemble (10-fold) 30 runs	SVM-based Ensemble (hold out) 30 runs
Class 1	30 base models	**FA**	5.03	0.9131	0.9107
		PSO	7.77	0.9146	0.9131
		GA	20.47	0.9175	0.9167

		ELPSO	7.63	0.9135	0.9127
		LSFA	5.00	0.9099	0.9083
		CFA	4.53	0.9108	0.9095
		NaFA	5.37	0.9038	0.9008
		HFDE	4	0.9056	0.9048
		Prop. FA	**7.37**	**0.9370**	**0.9258**
		Full size	30	0.9042	0.9047
		Base model	1	0.8583	0.8571
	50 base models	**FA**	10.03	0.9096	0.9075
		PSO	15.37	0.9126	0.9107
		GA	32.80	0.9155	0.9119
		ELPSO	15.13	0.9122	0.9095
		LSFA	11.97	0.9124	0.9095
		CFA	10.77	0.9108	0.9099
		NaFA	11.07	0.9026	0.9016
		HFDE	11	0.8944	0.8929
		Prop. FA	**14.27**	**0.9350**	**0.9235**
		Full size	50	0.9069	0.9047
		Base model	1	0.8583	0.8571
Class 2	30 base models	**FA**	4.43	0.8594	0.8583
		PSO	7.37	0.8591	0.8571
		GA	20.40	0.8583	0.8571
		ELPSO	7.50	0.8587	0.8575
		LSFA	5.87	0.8599	0.8583
		CFA	4.27	0.8587	0.8575
		NaFA	5.37	0.8615	0.8603
		HFDE	4	0.8583	0.8571
		Prop. FA	**7.13**	**0.8809**	0.8790
		Full size	30	0.8805	**0.8809**
		Base model	1	0.8583	0.8571
	50 base models	**FA**	10.03	0.8591	0.8571
		PSO	15.37	0.8583	0.8571
		GA	32.80	0.8583	0.8591
		ELPSO	15.13	0.8591	0.8575
		LSFA	11.77	0.8683	0.8609
		CFA	10.57	0.8607	0.8595
		NaFA	11.07	0.8624	0.8611
		HFDE	11	0.8583	0.8571
		Prop. FA	**14.60**	**0.8911**	**0.8897**
		Full size	50	0.8904	0.8809
		Base model	1	0.8583	0.8571
Class 3	30 base models	**FA**	5.20	0.9430	0.9409
		PSO	7.37	0.9393	0.9385
		GA	20.10	0.9406	0.9393
		ELPSO	7.23	0.9384	0.9377
		LSFA	4.87	0.9383	0.9365
		CFA	5.07	0.9434	0.9421
		NaFA	5.37	0.9431	0.9417
		HFDE	4	0.9403	0.9405
		Prop. FA	**7.03**	**0.9656**	**0.9643**
		Full size	30	0.9417	0.9405
		Base model	1	0.8733	0.8722
	50 base models	**FA**	10.03	0.9396	0.9381
		PSO	15.37	0.9402	0.9389
		GA	32.80	0.9405	0.9397
		ELPSO	15.13	0.9405	0.9401
		LSFA	11.77	0.9387	0.9377
		CFA	10.57	0.9390	0.9385
		NaFA	11.07	0.9390	0.9373
		HFDE	11	0.9417	0.9405
		Prop. FA	**14.60**	**0.9710**	**0.9679**
		Full size	30	0.9417	0.9405
		Base model	1	0.8738	0.8729
Class 4	30 base models	**FA**	5.20	0.8952	0.8944
		PSO	7.37	0.8938	0.8929
		GA	20.10	0.8935	0.8929
		ELPSO	7.23	0.8949	0.8937
		LSFA	5.87	0.8965	0.8952
		CFA	5.33	0.9000	0.8976

		NaFA	5.37	0.8938	0.8921
		HFDE	4	0.8931	0.8929
		Prop. FA	**6.63**	**0.9250**	**0.9218**
		Full size	30	0.8958	0.8929
		Base model	1	0.8603	0.8591
	50 base models	**FA**	10.03	0.8961	0.8952
		PSO	15.37	0.8959	0.8952
		GA	32.80	0.8934	0.8929
		ELPSO	15.13	0.8952	0.8937
		LSFA	11.77	0.9006	0.8960
		CFA	10.57	0.8963	0.8944
		NaFA	11.07	0.8942	0.8921
		HFDE	11	0.8958	0.8929
		Prop. FA	**14.20**	**0.9305**	**0.9302**
		Full size	50	0.8958	0.8929
		Base model	1	0.8632	0.8621
Class 5	30 base models	**FA**	5.20	0.9594	0.9583
		PSO	7.37	0.9614	0.9607
		GA	20.10	0.9619	0.9611
		ELPSO	7.23	0.9605	0.9595
		LSFA	5.87	0.9627	0.9615
		CFA	5.07	0.9602	0.9599
		NaFA	5.03	0.9628	0.9619
		HFDE	4	0.9653	0.9643
		Prop. FA	**7.37**	**0.9985**	**0.9976**
		Full size	30	0.9639	0.9643
		Base model	1	0.9069	0.9064
	50 base models	**FA**	10.03	0.9602	0.9587
		PSO	15.37	0.9598	0.9575
		GA	32.80	0.9612	0.9595
		ELPSO	15.13	0.9602	0.9591
		LSFA	11.77	0.9596	0.9596
		CFA	10.57	0.9582	0.9571
		NaFA	11.07	0.9612	0.9583
		HFDE	11	0.9667	0.9643
		Prop. FA	**14.27**	**1.0000**	**0.9992**
		Full size	50	0.9639	0.9643
		Base model	1	0.8990	0.8983

Table 13 Evaluation results for the skin lesion image data set using 30 and 50 base models, respectively

skin lesion	**Methods**	**Selected average number of base models**	**SVM-based Ensemble (10-fold) 30 runs**	**SVM-based Ensemble (hold-out) 30 runs**
30 base models	**FA**	5.07	0.8743	0.8711
	PSO	7.5	0.8785	0.8738
	GA	20.27	0.8854	0.8789
	ELPSO	7.47	0.8741	0.8694
	LSFA	5.73	0.8828	0.8789
	CFA	5.20	0.8825	0.8782
	NaFA	5.47	0.8806	0.8776
	HFDE	6	0.8778	0.8776
	Prop. FA	**7.37**	**0.9199**	**0.9167**
	Full size	30	0.8878	0.8878
	Base model	1	0.8656	0.8657
50 base models	**FA**	11.43	0.8977	0.8932
	PSO	14.80	0.9026	0.8973
	GA	32.43	0.9031	0.8986
	ELPSO	14.93	0.9093	0.9061
	LSFA	11.23	0.9032	0.8956
	CFA	10.27	0.9037	0.8986
	NaFA	11.83	0.9057	0.9007
	HFDE	10	0.8978	0.8980
	Prop. FA	**14.57**	**0.9337**	**0.9313**
	Full size	50	0.9078	0.9082
	Base model	1	0.8749	0.8747

Table 14 The *p*-values of the Wilcoxon rank sum test for all test data sets

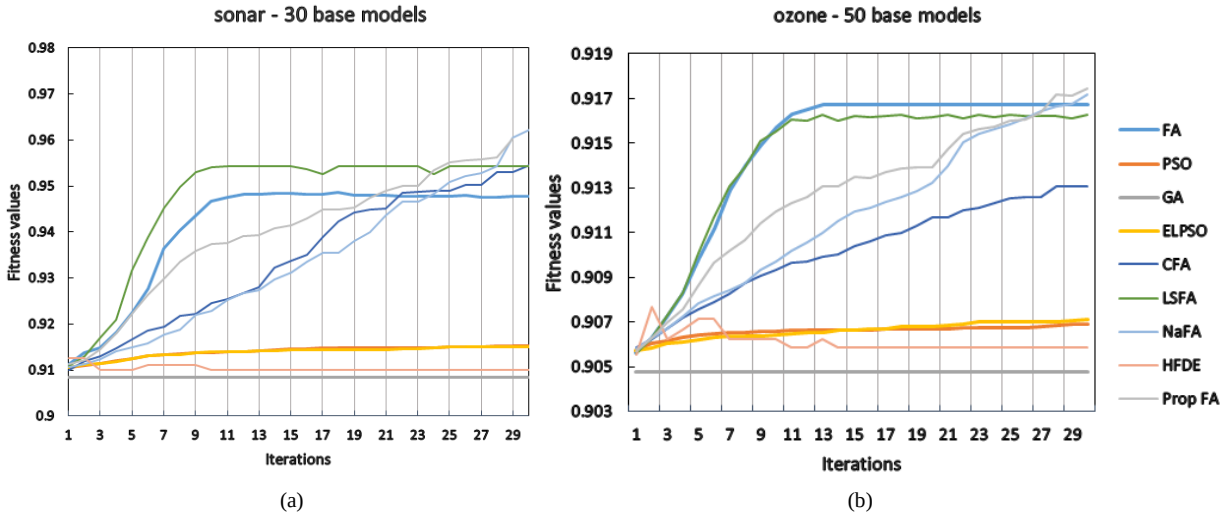
Data sets	Settings	FA	PSO	GA	ELPSO	LSFA	CFA	NaFA	HFDE
sonar	30 base models, 10-fold	1.09E-06	3.03E-06	3.38E-09	5.04E-06	1.42E-08	4.43E-07	1.67E-10	1.16E-12
	30 base models, holdout	3.24E-07	6.27E-08	1.55E-08	6.33E-08	1.10E-07	3.10E-06	2.15E-10	9.50E-13
	50 base models, 10-fold	2.65E-10	1.34E-08	6.12E-11	3.30E-10	4.80E-09	9.04E-10	2.30E-10	3.31E-11
	50 base models, holdout	1.07E-08	7.39E-09	7.06E-08	3.17E-09	4.78E-09	1.55E-08	3.62E-09	3.53E-05
ozone	30 base models, 10-fold	4.58E-10	2.33E-07	1.70E-06	5.53E-03	4.36E-08	6.17E-08	1.54E-08	1.21E-12
	30 base models, holdout	7.98E-09	1.77E-06	1.42E-08	3.33E-04	5.20E-07	1.88E-09	7.07E-08	1.12E-12
	50 base models, 10-fold	1.52E-07	7.36E-07	4.28E-06	2.17E-05	9.53E-08	1.08E-05	1.88E-08	7.38E-10
	50 base models, holdout	5.53E-10	2.19E-08	2.17E-06	2.05E-07	2.69E-08	2.67E-07	4.75E-08	9.91E-13
libras (Class 1)	30 base models, 10-fold	2.43E-07	8.93E-07	5.89E-06	1.08E-07	1.09E-07	1.52E-08	1.83E-09	4.48E-12
	30 base models, holdout	1.33E-03	3.85E-03	5.48E-02	1.67E-03	1.61E-04	1.15E-04	2.20E-09	1.13E-10
	50 base models, 10-fold	1.56E-09	2.76E-08	2.19E-07	1.46E-08	1.33E-08	1.63E-09	2.60E-11	1.20E-12
	50 base models, holdout	6.62E-06	3.82E-04	1.39E-03	8.51E-05	8.51E-05	1.44E-04	5.06E-09	1.06E-12
libras (Class 2)	30 base models, 10-fold	3.43E-11	2.63E-11	4.27E-12	8.98E-12	1.32E-10	8.98E-12	2.88E-09	4.27E-12
	30 base models, holdout	1.21E-11	7.60E-13	7.60E-13	2.03E-12	1.21E-11	2.03E-12	4.50E-10	7.60E-13
	50 base models, 10-fold	1.88E-12	9.49E-13	9.49E-13	1.87E-12	9.49E-13	1.36E-12	9.96E-12	9.49E-13
	50 base models, holdout	5.26E-13	5.26E-13	5.26E-13	7.63E-13	5.26E-13	7.63E-13	9.26E-12	9.49E-13
libras (Class 3)	30 base models, 10-fold	3.35E-06	5.35E-09	1.87E-08	9.66E-09	1.11E-08	9.15E-06	1.39E-05	5.66E-11
	30 base models, holdout	1.90E-07	1.09E-10	3.03E-11	3.23E-10	3.28E-10	3.81E-07	1.12E-06	1.50E-11
	50 base models, 10-fold	4.56E-09	4.76E-09	3.16E-09	2.46E-09	1.29E-09	2.74E-09	1.41E-09	7.59E-09
	50 base models, holdout	3.36E-10	2.99E-10	2.41E-10	2.07E-10	3.39E-10	3.21E-10	9.17E-10	1.73E-10
libras (Class 4)	30 base models, 10-fold	4.87E-11	2.08E-11	2.11E-11	4.89E-11	9.18E-11	1.29E-09	2.38E-11	1.17E-12
	30 base models, holdout	9.84E-12	7.22E-13	7.22E-13	2.78E-12	3.24E-11	7.84E-10	5.05E-12	7.22E-13
	50 base models, 10-fold	2.06E-11	2.03E-11	1.91E-11	2.00E-11	2.32E-11	2.44E-11	2.12E-11	1.10E-12
	50 base models, holdout	1.28E-12	1.28E-12	4.59E-13	6.69E-13	1.69E-12	9.42E-13	9.42E-13	4.59E-13
libras (Class 5)	30 base models, 10-fold	2.84E-12	2.84E-12	2.33E-12	2.79E-12	2.82E-12	2.87E-12	2.96E-12	6.14E-14
	30 base models, holdout	1.83E-12	1.33E-12	7.11E-13	1.17E-12	2.38E-12	2.70E-12	9.05E-12	6.12E-14
	50 base models, 10-fold	9.61E-13	1.11E-12	9.16E-13	1.08E-12	1.03E-12	1.10E-12	1.05E-12	1.69E-14
	50 base models, holdout	7.92E-13	6.50E-13	6.09E-13	7.92E-13	6.09E-13	6.09E-13	6.84E-13	2.71E-14
skin lesion	30 base models, 10-fold	9.79E-10	5.14E-10	3.45E-10	1.06E-10	2.41E-07	2.88E-08	5.06E-09	1.19E-12
	30 base models, holdout	1.20E-09	5.02E-10	2.35E-10	1.02E-10	7.39E-08	3.48E-08	9.55E-09	1.08E-12
	50 base models, 10-fold	2.19E-08	8.22E-08	1.29E-10	3.42E-06	5.00E-10	3.13E-09	9.61E-05	1.17E-12
	50 base models, holdout	1.93E-09	6.28E-09	3.29E-11	2.30E-07	6.51E-11	1.36E-09	9.05E-06	1.03E-12

As illustrated in Table 13, using 30 base classifiers, our ensemble classifiers outperform those produced by FA, PSO, GA, ELPSO, LSFA, CFA, NaFA and HFDE by 4.56%, 4.14%, 3.45%, 4.58%, 3.71%, 3.74%, 3.93% and 4.21% on average over 30 runs for 10-fold cross validation and by 4.56%, 4.29%, 3.78%, 4.73%, 3.78%, 3.85%, 3.91% and 3.91% for hold-out validation. The results are better than those from the full-sized ensemble classifier integrating 30 base models by 3.21% and 2.89% for cross and hold-out validation tests, respectively. Using 50 base models, our ensemble classifiers outperform those produced by FA, PSO, GA, ELPSO, LSFA, CFA, NaFA and HFDE by 3.6%, 3.11%, 3.06%, 2.44%, 3.05%, 3%, 2.8% and 3.59% on average over 30 runs for 10-fold cross validation, and by 3.81%, 3.4%, 3.27%, 2.52%, 3.57%, 3.27%, 3.06% and 3.33% for hold-out validation. The results are better than those from the full-sized ensemble classifier combining 50 base models by 2.59% and 2.31% for cross and hold-out validation tests, respectively.

Statistical tests have been conducted to evaluate the significance level of the proposed algorithm for ensemble reduction based on the UCI and skin lesion data sets. The results of the Wilcoxon rank sum test are shown in Table 14. As indicated in Table 14, the *p*-values are smaller than 0.05 for nearly all test cases of all data sets, which indicate that our algorithm shows significant superiority over other methods, statistically.

Since the base classifiers built by using different feature subsets represent different scenarios of the same problem, maintaining too many base models may cause redundancy and overlapping in scenarios, whereas removing too many base classifiers may lose crucial distinctive scenarios of the original problem and result in performance reduction. The internal mechanisms of each search method determine the best trade-off between classification accuracy and the number of selected base models. The empirical results indicate that the proposed algorithm achieves the best trade-off between these two criteria over other methods. NaFA, HFDE, LSFA, CFA and FA show discriminative capabilities of ensemble member selection, and tend to generate the smallest ensemble sizes, however, at the expense of reduced classification accuracy owing to losing complementary information of the base models to some extent. On the contrary, GA shows competitive classification performance across data sets, however, at the expense of larger ensemble sizes with high complexity. The generated models are more likely to possess a high level of redundant information. ELPSO and PSO generate similar ensemble sizes as compared with those produced by the proposed algorithm. However, the proposed FA model shows more competitive capability of

discriminative base model selection, and produces better classification accuracy rates than those of PSO and ELPSO.



(a) (b)

Figure 4 Average convergence curves of 30 experimental runs for (a) sonar and (b) ozone data sets with 30 and 50 base models respectively

We have also compared the convergence rates of different methods using the training samples of each data set. Figure 4 (a) and (b) show the convergence curves of the sonar and ozone data sets with 30 and 50 base models, respectively. The average convergence performance over 30 runs for each method is used to generate the plots for each data set shown in Figure 4.

As shown in Figure 4, for both sonar and ozone data sets with 30 and 50 base models respectively, LSFA and FA show fast convergence performance in early iterations. NaFA, CFA, and the proposed FA model depict radical improvements in subsequent iterations. HFDE, PSO, ELPSO, and GA show no significant improvements for the convergence rates during the lift-time of the training process. A similar observation is also obtained for the convergence performance of other data sets with 30 and 50 base models for ensemble reduction.

4.5.3 Discussion on Different Search Methods

A theoretical comparison of the internal search mechanisms between the proposed algorithm and other FA and PSO variants is presented, as follows. CFA (Kazem et al., 2013) employs chaotic accelerated population initialization and attractiveness behaviour to guide the search process. ODFA (Verma et al., 2016) utilizes a dimensional approach to retrieve g_{best}. Instead of using brighter fireflies from the whole population, NaFA (Wang et al., 2017) uses those in a pre-defined smaller neighbourhood to guide the attractiveness action. Nevertheless, the mechanism of finding the global best solutions in the above three FA models fully relies on the single attractiveness operation. When this action fails to achieve an optimal solution, or is trapped in local optima, there is no alternative search strategy to lead the search out of stagnation. Similarly, although LSFA (Alweshah and Abdullah, 2015) incorporates SA to improve the global best solution identified by LFA, the main operator of SA performs randomized movements. Such randomized operations possess limited exploitation capabilities, with a considerably slow convergence rate. ELPSO (Jordehi, 2015) employs Gaussian, Cauchy, opposition and DE based mutation operators to improve the global best solution identified by PSO in each iteration. However, the search of global optimality is guided purely by a single swarm leader without consideration of multiple promising solutions in the neighbourhood, and it is more likely to be trapped in local optima. HFDE (Dash et al., 2017) incorporates FA to further improve the global search capabilities of IDE. In HFDE, although the FA movement is used to improve three top solutions identified by IDE, the search process largely relies on the mutation and crossover operations of IDE. In comparison with FA and other FA variants, the global search capability of HFDE is comparatively limited, especially when the search space is large. Moreover, the mutation operator of HFDE mainly employs the global best solution for offspring generation. Therefore, the search process is more likely to be trapped in local optima in comparison with those guided by multiple neighbouring optimal solutions.

The proposed FA model employs two newly introduced attractiveness and evading search behaviours to mitigate premature convergence and explore wider and more distinctive search regions. Therefore, it has better capability of finding global optima and escaping from local optimum traps. In particular, the attractiveness search operation is guided by both local and global best solutions to reduce the probability of falling into local optima. When the attractiveness behaviour stagnates, the evading operation enables the fireflies to move to other optimal regions and to drive the search out of stagnation, and vice versa. The attractiveness and evading actions work in a collaborative manner to lead the search towards global optimality. Overall, both proposed search mechanisms account for the superiority of the proposed algorithm in discriminant base model selection.

Moreover, for evaluation of the standard functions, shifted and composite test suite, and the ensemble reduction problems using UCI data sets, the proposed algorithm has the same number of function evaluations as those of the original FA, SA, PSO, BSO, CS, DA, ALO, CFA, and NaFA models, i.e. (population size)×(maximum number of generations). As an example, in each iteration of SA, there is only one function evaluation for the current solution, therefore we have increased the iteration number to (population size)×(maximum number of generations), in order to have a fair comparison. Other FA and PSO variants, such as LSFA, SFA, ODFA, HFDE and ELPSO, have used higher numbers of function evaluations.

LSFA (Alweshah and Abdullah, 2015) employed SA to further improve the global best solution obtained by LFA in each iteration, therefore the number of function evaluations for LSFA is the sum of the evaluations conducted for both LFA and SA, i.e. (population size)×(maximum number of generations) + (maximum trials of SA)×(maximum number of generations). A similar scenario applies to SFA (Alweshah and Abdullah, 2015), where the global best solution of FA is further enhanced by SA. The number of function evaluations for SFA is the sum of the cost for both FA and SA, i.e. (population size)×(maximum number of generations) + (maximum trials of SA)×(maximum number of generations). ODFA (Verma et al., 2016) employs a dimensional approach to retrieve g_{best} by identifying the best solution in each dimension among all the population members in each generation. Therefore, the number of function evaluations for ODFA is dimension × population size × maximum number of generations. ELPSO (Jordehi, 2015) employs a five-stage strategy to improve the global best solution in each iteration, therefore it has a slightly higher number of function evaluations, i.e. (population size + 5)×(maximum number of generations). Since HFDE (Dash et al., 2017) integrates DE with FA, the number of function evaluations of HFDE is the sum of the cost for both DE (i.e. the fitness evaluation for both parent and offspring populations) and FA, i.e. (2×population size + 3 offspring generated using FA)×(maximum number of generations).

Overall, the proposed algorithm has the same number of function evaluations and similar computational efficiency as those of the original FA model and other classical search methods. Other FA and PSO variants, such as HFDE, ODFA, LSFA, SFA, and ELPSO, have comparatively larger numbers of function evaluations, therefore higher computational costs as compared with those of the proposed algorithm.

5. CONCLUSIONS

In this research, we have proposed a modified FA model which incorporates two newly proposed attractiveness and evading search mechanisms to address the limitations of the original search behaviours of FA. Both search strategies work cooperatively to mitigate the premature convergence of the original FA model. The attractiveness movement not only uses the neighbouring but also global best solutions to guide the search process, while the evading operation leads the swarm to avoid unpromising search regions. Based on a series of standard, shifted, and composite test functions, BBOB testbeds and several high dimensional data sets, the proposed algorithm shows great capability of not only solving diverse challenging complex unimodal and multimodal benchmark problems, but also identifying discriminant base models for ensemble reduction. Moreover, evaluated with the sonar, ozone, libras and skin lesion data sets, the resulting classifier ensembles generated by the proposed algorithm outperform those produced by other search methods, as well as the corresponding full-sized ensemble classifiers and single base classifiers, significantly.

In further work, we aim to further improve the proposed search mechanisms by incorporating a micro GA-based secondary swarm, chaos-based parameter tuning, and mutation-based population diversification for performance enhancement (Neoh et al., 2015; Mistry et al., 2016; Srisukkham et al., 2017). Other adaptive parameter tuning strategies and hybrid evading actions will also be studied to further improve the robustness of the proposed algorithm. In addition, other base classifier pool generation techniques will also be used to further evaluate the

proposed algorithm. We also aim to extend the proposed algorithm to deal with data stream classification with concept drifting and hyper-parameter selection for deep neural networks (Kinghorn et al., 2017a; 2017b). Besides that, multi-objective fitness evaluation will be employed for ensemble reduction.

APPENDIX

Table A.1 Unimodal benchmark functions

Function	Range	Shift position	f_{min}
$f_1(x) = \sum_{i=1}^{n} x_i^2$	[-100,100]	[-30, -30,…, -30]	0
$f_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	[-10,10]	[-3, -3,…, -3]	0
$f_3(x) = \sum_{i=1}^{n} \left(\sum_{j-1}^{i} x_j \right)^2$	[-100,100]	[-30, -30,…, -30]	0
$f_4(x) = \max_i \{ \lvert x_i \rvert, 1 \le i \le n \}$	[-100,100]	[-30, -30,…, -30]	0
$f_5(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	[-30,30]	[-15, -15,…, -15]	0
$f_6(x) = \sum_{i=1}^{n} ([x_i + 0.5])^2$	[-100,100]	[-750,…, -750]	0
$f_7 = \sum_{i=1}^{n} i x_i^4 + \text{random}[0,1]$	[-1.28,1.28]	[-0.25,…, -0.25]	0

Table A.2 Multimodal benchmark functions

Function	Range	Shift position	f_{min}
$f_8(x) = \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i) + 10]$	[-5.12,5.12]	[-2, -2,…, -2]	0
$f_9(x) = -20\exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^{n} x_i^2} \right) - \exp\left(\frac{1}{n} \sum_{i=1}^{n} \cos(2\pi x_i) \right) + 20 + e$	[-32,32]		0
$f_{10}(x) = \frac{1}{4000} \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos(\frac{x_i}{\sqrt{i}}) + 1$	[-600,600]	[-400,…, -400]	0
$f_{11}(x) = \frac{\pi}{n} \left\{ 10\sin(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2 [1 + 10\sin^2(\pi y_{i+1})] + (y_n - 1)^2 \right\} + \sum_{i=1}^{n} u(x_i, 10, 100, 4)$ $y_i = 1 + \frac{x_i + 1}{4}$ $u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a < x_i < a \\ k(-x_i - a)^m & x_i < -a \end{cases}$	[-50,50]	[-30, -30,…, -30]	0
$f_{12}(x) = 0.1 \left\{ \sin^2(3\pi x_1) + \sum_{i=1}^{n} (x_i - 1)^2 [1 + sin^2(3\pi x_i + 1)] + (x_n - 1)^2 [1 + \sin^2(2\pi x_n)] \right\} + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$	[-50,50]	[-100,…, -100]	0

Table A.3 Composite benchmark functions

Function	Range	f_{min}
$f_{13}(CF1)$: $f_1, f_2, f_3, \dots, f_{10} =$ Sphere Function	[-5,5]	0

$[\delta_1, \delta_2, \delta_3, \dots, \delta_{10}] = [1,1,1, \dots, 1]$
$[\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_{10}] = [5/100, 5/100, 5/100, \dots, 5/100]$

$f_{14}(CF2)$: [-5,5] 0
$f_1, f_2, f_3, \dots, f_{10} =$ Griewank's Function
$[\delta_1, \delta_2, \delta_3, \dots, \delta_{10}] = [1,1,1, \dots, 1]$
$[\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_{10}] = [5/100, 5/100, 5/100, \dots, 5/100]$

$f_{15}(CF3)$: [-5,5] 0
$f_1, f_2, f_3, \dots, f_{10} =$ Griewank's Function
$[\delta_1, \delta_2, \delta_3, \dots, \delta_{10}] = [1,1,1, \dots, 1]$
$[\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_{10}] = [1,1,1, \dots, 1]$

$f_{16}(CF4)$: [-5,5] 0
$f_1, f_2 =$ Ackley's Function
$f_3, f_4 =$ Rastrigin's Function
$f_5, f_6 =$ Weierstrass Function
$f_7, f_8 =$ Griewank's Function
$f_9, f_{10} =$ Sphere Function
$[\delta_1, \delta_2, \delta_3, \dots, \delta_{10}] = [1,1,1, \dots, 1]$
$[\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_{10}] = [5/32, 5/32, 1, 1, 5/0.5, 5/0.5, 5/100, 5/100, 5/100, 5/100]$

$f_{17}(CF5)$: [-5,5] 0
$f_1, f_2 =$ Rastrigin's Function
$f_3, f_4 =$ Weierstrass Function
$f_5, f_6 =$ Griewank's Function
$f_7, f_8 =$ Ackley's Function
$f_9, f_{10} =$ Sphere Function
$[\delta_1, \delta_2, \delta_3, \dots, \delta_{10}] = [1,1,1, \dots, 1]$
$[\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_{10}] = [1/5, 1/5, 5/0.5, 5/0.5, 5/100, 5/100, 5/32, 5/32, 5/100, 5/100]$

$f_{18}(CF6)$: [-5,5] 0
$f_1, f_2 =$ Rastrigin's Function
$f_3, f_4 =$ Weierstrass Function
$f_5, f_6 =$ Griewank's Function
$f_7, f_8 =$ Ackley's Function
$f_9, f_{10} =$ Sphere Function
$[\delta_1, \delta_2, \delta_3, \dots, \delta_{10}] = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1]$
$[\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_{10}] = [0.1 * 1/5, 0.2 * 1/5, 0.3 * 5/0.5, 0.4 * 5/0.5, 0.5 * 5/100, 0.6 * 5/100, 0.7 * 5/32, 0.8 * 5/32, 0.9 * 5/100, 1 * 5/100]$

REFERENCES

Alweshah, M., & Abdullah, S. (2015). Hybridizing Firefly Algorithms with a Probabilistic Neural Network for Solving Classification Problems. *Applied Soft Computing*, 35 (2015) 513–524.

Auger, A., & Ros, R. (2009). Benchmarking the Pure Random Search on the BBOB-2009 Noisy Testbed. In *GECCO'09 Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference: Late Breaking Papers*, 2479–2484.

Auger, A., & Hansen, N. (2009). Benchmarking the (1+1)-CMA-ES on the BBOB-2009 Noisy Testbed. In *GECCO'09 Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conferen*ce: *Late Breaking Papers*, 2467–2472.

[dataset] Bache, K., & Lichman, M. (2013). UCI Machine Learning Repository. School of Information and Computer Science, University of California, Irvine, CA, USA. http://archive.ics.uci.edu/ml. Accessed 3rd May 2017.

[dataset] Ballerini, L., Fisher, R.B., Aldridge, B., & Rees, J. (2013). A Color and Texture based Hierarchical K-NN Approach to the Classification of Non-Melanoma Skin Lesions. In *Color Medical Image Analysis*, 63–86. Springer Netherlands.

Baykasoğlu, A., & Ozsoydan, F.B. (2014). An Improved Firefly Algorithm for Solving Dynamic Multidimensional Knapsack Problems. *Expert Systems with Applications*, 41 (8) 3712–3725.

Baykasoğlu, A., & Ozsoydan, F.B. (2015). Adaptive Firefly Algorithm with Chaos for Mechanical Design Optimization Problems. *Applied Soft Computing*, 36 (2015) 152–164.

Bosman, P.A.N., Grahl, J., & Thierens, D. (2009). AMaLGaM IDEAs in Noisy Black-Box Optimization Benchmarking. In *GECCO'09 Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conferen*ce: *Late Breaking Papers*, 2351–2358.

Chou, J.S., & Ngo, N.T. (2017). Modified Firefly Algorithm for Multidimensional Optimization in Structural Design Problems. *Structural and Multidisciplinary Optimization*, 55 (6) 2013–2028.

Christoudias, C.M., Urtasun, R., & Darrell, T. (2008). Multi-View Learning in the Presence of View Disagreement. In *Proceedings of the Twenty-Fourth Conference on Uncertainty in Artificial Intelligence*, 88–96.

Coelho, L.D.S., Bernert, D.L.D.A., & Mariani, V.C. (2011). A Chaotic Firefly Algorithm Applied to Reliability-Redundancy Optimization. In *Proceedings of IEEE Congress on Evolutionary Computation (CEC)*, IEEE, 2011, 89–98.

Dash, J., Dam, B., & Swain, R. (2017). Design of Multipurpose Digital FIR Double-band Filter Using Hybrid Firefly Differential Evolution Algorithm. *Applied Soft Computing*, 59 (2017) 529–545.

Derrac, J., García, S., Molina, D., & Herrera, F. (2011). A Practical Tutorial on the Use of Nonparametric Statistical Tests as a Methodology for Comparing Evolutionary and Swarm Intelligence Algorithms. *Swarm and Evolutionary Computation*, 1 (1) 3–18.

Diao, R., Chao, F., Peng, T., Snooke, N., & Shen, Q. (2014). Feature Selection Inspired Classifier Ensemble Reduction. *IEEE Transactions on Cybernetics*, 44 (8) 1259–268.

El-Abd, M., & Kamel, M.S. (2009a). Black-Box Optimization Benchmarking for Noiseless Function Testbed Using an EDA and PSO Hybrid. In *GECCO'09 Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conferen*ce: *Late Breaking Papers*, 2263–2268.

El-Abd, M., & Kamel, M.S. (2009b). Black-Box Optimization Benchmarking for Noiseless Function Testbed Using PSO_Bounds. In *GECCO'09 Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference*, 2275–2280.

Farid, D., Zhang, L., Hossain, A.M., Rahman, C.M., Strachan, R., Sexton, G., & Dahal, K. (2013). An Adaptive Ensemble Classifier for Mining Concept-Drifting Data Streams. *Expert Systems with Applications*, 40 (15) 5895–5906.

Finck, S., & Beyer, H.G. (2010). Benchmarking SPSA on BBOB-2010 Noisy Function Testbed. In *GECCO'10 Proceedings of the 12th Annual Conference Companion on Genetic and Evolutionary Computation*, 1665–1672.

Fister, I., Fister Jr, I., Yang, X.S., & Brest, J. (2013). A Comprehensive Review of Firefly Algorithms. *Swarm and Evolutionary Computation*, 13 (2013) 34–46.

Fister Jr, I., Perc, M., Kamal, S.M., & Fister, I. (2015). A Review of Chaos-based Firefly Algorithms: Perspectives and Research Challenges. *Applied Mathematics and Computation*, 252 (2015) 155–165.

Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., & Herrera, F. (2012). A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches. *IEEE Transactions on Systems, Man and Cybernetics – Part C: Applications and Reviews*, 42 (4) 463–484.

Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., & Herrera, F. (2016). Ordering-based Pruning for Improving the Performance of Ensembles of Classifiers in the Framework of Imbalanced Datasets. *Information Sciences*, 354 (2016) 178–196.

Gallagher, M. (2009). Black-Box Optimization Benchmarking: Results for the $\text{BayEDA}_{\text{cG}}$ Algorithm on the Noisy Function Testbed. In *GECCO'09 Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference: Late Breaking Papers*, 2383–2388.

Gandomi, A.H., Yang, X.S., Talatahari, S., & Alavi, A.H. (2013). Firefly Algorithm with Chaos. *Communications in Nonlinear Science and Numerical Simulation*, 18 (1) 89–98.

García-Nieto, J., Alba, E., & Apolloni, J. (2009). Particle Swarm Hybridized with Differential Evolution: Black-Box Optimization Benchmarking for Noisy Functions. In *GECCO'09 Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conferen*ce: *Late Breaking Papers*, 2343–2350.

García-Martínez, C., & Lozano, M. (2009). A Continuous Variable Neighbourhood Search Based on Specialised EAs Application to the Noisy BBO-Benchmark 2009 Testbed. In *GECCO'09 Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conferen*ce: *Late Breaking Papers*, 2367–2374.

Guan, Y., Li, C.T., & Roli, F. (2015). On Reducing the Effect of Covariate Factors in Gait Recognition: A Classifier Ensemble Method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37 (7) 1521–1528.

Han, J., Kamber, M., & Pei, J. (2011). *Data Mining: Concepts and Techniques*. 3rd edn. Morgan Kaufmann.

Hansen, N., Auger, A., Ros, R, Finck, S., & Pošík, P. (2010a). Comparing Results of 31 Algorithms from the Black-Box Optimization Benchmarking BBOB-2009. In *GECCO'10 Proceedings of the 12th Annual Conference Companion on Genetic and Evolutionary Computation*, 1689–1696. USA.

Hansen, N., Finck, S., Ros, R., & Auger, A. (2010b). Real-Parameter Black-Box Optimization Benchmarking 2009: Noisy Functions Definitions. *Technical Report RR-6869*, INRIA, 2009. Updated February 2010.

Hansen, N., Auger, A., Finck, S., & Ros, R. (2012). Real-Parameter Black-Box Optimization Benchmarking 2012: Experimental Setup. *Technical Report*, INRIA, 2012.

Ho, T. (1998). The Random Subspace Method for Constructing Decision Forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20 (8) 832–844.

Hornby, G.S. (2009). The Age-Layered Population Structure (ALPS) Evolutionary Algorithm. In *GECCO'09 Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference*.

Huang, H., Chiu, C.F., Kuo, C.H., Wu, Y.C., Chu, N.N.Y., & Chang, P.C. (2016). Mixture of Deep CNN-based Ensemble Model for Image Retrieval. In *Proceedings of IEEE 5th Global Conference on Consumer Electronics*, 1–2.

Huyer, W., & Neumaier, A. (2009a). Benchmarking of SNOBFIT on the Noisy Function Testbed. In *GECCO'09 Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conferenc*e.

Huyer, W., & Neumaier, A. (2009b). Benchmarking of MCS on the Noisy Function Testbed. In *GECCO'09 Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conferenc*e.

Jordehi, A.R. (2015). Enhanced leader PSO (ELPSO): A New PSO Variant for Solving Global Optimisation Problems. *Applied Soft Computing*, 26 (2015) 401–417.

Jothi, G., & Inbarani, H.H. (2016). Hybrid Tolerance Rough Set–Firefly based Supervised Feature Selection for MRI Brain Tumor Image Classification. *Applied Soft Computing*, 46 (2016) 639–651.

Kazem, A., Sharifi, E., Hussain, F.K., Saberic, M., & Hussain, O.K. (2013). Support Vector Regression with Chaos-based Firefly Algorithm for Stock Market Price Forecasting. *Applied Soft Computing*, 13 (2) 947–958.

Kinghorn, P., Zhang, L., & Shao, L. (2017a). A Region-based Image Caption Generator with Refined Descriptions. *Neurocomputing*. (In Press).

Kinghorn, P., Zhang, L., & Shao, L. (2017b). A Hierarchical and Regional Deep Learning Architecture for Image Description Generation. *Pattern Recognition Letters*. (In Press).

Korošec, P., & Šilc, J. (2009). A Stigmergy-Based Algorithm for Black-Box Optimization: Noisy Function Testbed. In *GECCO'09 Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conferenc*e: *Late Breaking Papers*, 2375–2382.

Krawczyk, B. (2015). One-Class Classifier Ensemble Pruning and Weighting with Firefly Algorithm. *Neurocomputing*, 150 (2015) 490–500.

Liew, W.S., Loo, C.K., & Obo, T. (2016). Genetic Optimized Fuzzy Extreme Learning Machine Ensembles for Affect Classification. In *Proceedings of Joint 8th International Conference on Soft Computing and Intelligent Systems (SCIS) and 17th International Symposium on Advanced Intelligent Systems (ISIS)*, 305–310.

Long, N.C., Meesad, P., & Unger, H. (2015). A Highly Accurate Firefly based Algorithm for Heart Disease Prediction. *Expert Systems with Applications*, 42 (21) 8221–8231.

Mirjalili, S. (2015a). The Ant Lion Optimizer. *Advances in Engineering Software*, 83, 80–98.

Mirjalili, S. (2015b). Moth-Flame Optimization Algorithm: A Novel Nature-Inspired Heuristic Paradigm. *Knowledge-based Systems*, 89 (2015) 228–249.

Mirjalili, S. (2016a). Dragonfly Algorithm: A New Meta-Heuristic Optimization Technique for Solving Single-Objective, Discrete, and Multi-Objective Problems. *Neural Computing and Applications*, 27 (4) 1053–1073.

Mirjalili, S. (2016b). SCA: A Sine Cosine Algorithm for Solving Optimization Problems. *Knowledge-based Systems*, 96 (2016) 120–133.

Mirjalili, S., & Lewis, A. (2016). The Whale Optimization Algorithm. *Advances in Engineering Software*, 95 (2016) 51–67.

Mistry, K., Zhang, L., Neoh, S.C., Lim, C.P., & Fielding, B. (2016). A Micro-GA Embedded PSO Feature Selection Approach to Intelligent Facial Emotion Recognition. *IEEE Transactions on Cybernetics*, 1–14.

Neoh, S.C., Zhang, L., Mistry, K., Hossain, M.A., Lim, C.P., Aslam, N, & Kinghorn, P. (2015). Intelligent Facial Emotion Recognition Using a Layered Encoding Cascade Optimization Model. *Applied Soft Computing*, 34 (2015) 72–93.

Ozsoydan, F.B., & Baykasoğlu, A. (2015). A Multi-Population Firefly Algorithm for Dynamic Optimization Problems. In *Proceedings of IEEE International Conference on Evolving and Adaptive Intelligent Systems (EAIS)*, Douai, France.

Pal, L., Csendes, T., Markot, M.C., & Neumaier, A. (2012). Black-Box Optimization Benchmarking of the GLOBAL Method. *Evolutionary Computation*, 20 (4) 609–639.

Pietruczuk, L., Rutkowski, L., Jaworski, M., & Duda, P. (2017). How to Adjust an Ensemble Size in Stream Data Mining?. *Information Sciences*, 381 (2017) 46–54.

Ros, R. (2009). Benchmarking the BFGS Algorithm on the BBOB-2009 Noisy Testbed. In *GECCO'09 Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference.*

Ros, R. (2010). Comparison of NEWUOA with Different Numbers of Interpolation Points on the BBOB Noisy Testbed. In *GECCO'10 Proceedings of the 12th Annual Conference Companion on Genetic and Evolutionary Computation*, 1495–1502.

Sayadi, M.K., Hafezalkotob, A., & Naini, S.G.J. (2013). Firefly-Inspired Algorithm for Discrete Optimization Problems: An Application to Manufacturing Cell Formation. *Journal of Manufacturing Systems*, 32 (1) 78–84.

Schaul, T. (2012a). Benchmarking Separable Natural Evolution Strategies on the Noiseless and Noisy Black-Box Optimization Testbeds. In *GECCO'12 Proceedings of the 14th Annual Conference Companion on Genetic and Evolutionary Computation*, 205–212. USA.

Schaul, T. (2012b). Benchmarking Exponential Natural Evolution Strategies on the Noiseless and Noisy Black-Box Optimization Testbeds. In *GECCO'12 Proceedings of the 14th Annual Conference Companion on Genetic and Evolutionary Computation*, 213–220. USA.

Schaul, T. (2012c). Benchmarking Natural Evolution Strategies with Adaptation Sampling on the Noiseless and Noisy Black-Box Optimization Testbeds. In *GECCO'12 Proceedings of the 14th Annual Conference Companion on Genetic and Evolutionary Computation*, 229–236. USA.

Sołtys, M., Jaroszewicz, S., & Rzepakowski, P. (2015). Ensemble Methods for Uplift Modelling. *Data Mining and Knowledge Discovery*, 29 (6) 1531–1559.

Srisukkham, W., Zhang, L., Neoh, S.C., Todryk, S., & Lim, C.P. (2017). Intelligent Leukaemia Diagnosis with Bare-Bones PSO based Feature Optimization. *Applied Soft Computing*, 56 (2017) 405–419.

Srivatsava, P.R., Mallikarjun, B., & Yang, X.S. (2013). Optimal Test Sequence Generation Using Firefly Algorithm. *Swarm and Evolutionary Computation*, 8, 44–53.

Su, H., Cai, Y., & Du, Q. (2017). Firefly-Algorithm-Inspired Framework with Band Selection and Extreme Learning Machine for Hyperspectral Image Classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10 (1) 309–320.

Sun, Y., Tang, K., Minku, L.L., Wang, S., & Yao, X. (2016). Online Ensemble Learning of Data Streams with Gradually Evolved Classes. *IEEE Transactions on Knowledge and Data Engineering*, 28 (6) 1532–1545.

Tan, T.Y., Zhang, L., & Jiang, M. (2016). An Intelligent Decision Support System for Skin Cancer Detection from Dermoscopic Images. In *Proceedings of the 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*.

Tilahun, S.L., & Ong, H.C. (2012). Modified Firefly Algorithm. *Journal of Applied Mathematics*, Volume 2012 (2012), Article ID 467631, 1–12.

Verma, O.P., Aggarwal, D., & Patodi, T. (2016). Opposition and Dimensional based Modified Firefly Algorithm. *Expert Systems with Applications*, 44 (2016) 168–176.

Wang, H., Wang, W., Zhou, X., Sun, H., Zhao, J., Yu, X., & Cui, Z. (2017). Firefly Algorithm with Neighbourhood Attraction. *Information Sciences*, 382–383 (2017) 374–387.

Xu, M., & Liu, G. (2013). A Multipopulation Firefly Algorithm for Correlated Data Routing in Underwater Wireless Sensor Networks. *International Journal of Distributed Sensor Networks*, Volume 2013, Article ID 865154, 1–14.

Yang, X.S. (2008). *Nature Inspired Metaheuristic Algorithm*. Luniver press.

Yang, X.S. (2009). Firefly Algorithms for Multimodal Optimization. In O. Watanabe, & T. Zeugmann (Eds.), Stochastic Algorithms: Foundations and Applications. SAGA 2009. *Lecture Notes in Computer Science*, Volume 5792. Springer, Berlin, Heidelberg, 169–178.

Yang, X.S. (2010). Firefly Algorithm, Levy Flights and Global Optimization. *Research and Development in Intelligent Systems*, 26 (2010) 209–218.

Zhang, L., Liu, L., Yang, X.S., & Dai, Y. (2016a). A Novel Hybrid Firefly Algorithm for Global Optimization. *PLoS ONE*, 11(9) e0163230.

Zhang, L., Mistry, K., Neoh, S.C., & Lim, C.P. (2016b). Intelligent Facial Emotion Recognition Using Moth-firefly Optimization. *Knowledge-based Systems*, 111 (2016) 248–267.

Zhang, Y., Zhang, L., & Hossain, M.A. (2015). Adaptive 3D Facial Action Intensity Estimation and Emotion Recognition. *Expert Systems with Applications*, 42 (3) 1446–1464.

Zhou, L., Ding, L., & Qiang, X. (2014). A Multi-Population Discrete Firefly Algorithm to Solve TSP. In L. Pan, G. Păun, M.J. Pérez-Jiménez, & T. Song (Eds.), Bio-Inspired Computing – Theories and Applications. *Communications in Computer and Information Science*, Volume 472. Springer, Berlin, Heidelberg, 648–653.