

A Scattering and Repulsive Swarm Intelligence Algorithm for Solving Global Optimization Problems

Diptangshu Pandit¹, Li Zhang¹, Samiran Chattopadhyay², Chee Peng Lim³ and Chengyu Liu⁴

¹Computational Intelligence Research Group
Department of Computer and Information Sciences
Faculty of Engineering and Environment
University of Northumbria
Newcastle, NE1 8ST, UK

²Department of Information Technology
Jadavpur University
Kolkata, India

³Institute for Intelligent Systems Research and Innovation
Deakin University
Waurm Ponds, VIC 3216, Australia

⁴School of Instrument Science and Engineering, Southeast University, Nanjing, 210018, P.R. China

Email: {[d.pandit](mailto:d.pandit@northumbria.ac.uk); [li.zhang](mailto:li.zhang@northumbria.ac.uk)}@northumbria.ac.uk; samirancju@gmail.com;
chee.lim@deakin.edu.au; bestlcy@sdu.edu.cn

Abstract.

The firefly algorithm (FA), as a metaheuristic search method, is useful for solving diverse optimization problems. However, it is challenging to use FA in tackling high dimensional optimization problems, and the random movement of FA has a high likelihood to be trapped in local optima. In this research, we propose three improved algorithms, i.e., Repulsive Firefly Algorithm (RFA), Scattering Repulsive Firefly Algorithm (SRFA), and Enhanced SRFA (ESRFA), to mitigate the premature convergence problem of the original FA model. RFA adopts a repulsive force strategy to accelerate fireflies (i.e. solutions) to move away from unpromising search regions, in order to reach global optimality in fewer iterations. SRFA employs a scattering mechanism along with the repulsive force strategy to divert weak neighbouring solutions to new search regions, in order to increase global exploration. Motivated by the survival tactics of hawk-moths, ESRFA incorporates a hovering-driven attractiveness operation, an exploration-driven evading mechanism, and a learning scheme based on the historical best experience in the neighbourhood to further enhance SRFA. Standard and CEC2014 benchmark optimization functions are used for evaluation of the proposed FA-based models. The empirical results indicate that ESRFA, SRFA and RFA significantly outperform the original FA model, a number of state-of-the-art FA variants, and other swarm-based algorithms, which include Simulated Annealing, Cuckoo Search, Particle Swarm, Bat Swarm, Dragonfly, and Ant-Lion Optimization, in diverse challenging benchmark functions.

Keywords: Optimization, Metaheuristic Search Algorithms, and Firefly Algorithm.

1. INTRODUCTION

Nature inspired algorithms have gained popularity and been widely used for solving various global optimization problems. Among them, the firefly algorithm (FA) [1] is a popular metaheuristic search method that has been applied to undertaking diverse optimization problems in engineering, medical, and social sciences [2]. In FA, each firefly represents a solution in the search space. Its light intensity is determined by an objective function. Each firefly moves to the optimal region by following multiple optimal solutions in the neighbourhood. Overall, the FA search strategies enable the fireflies (i.e. solutions) with lower light intensities (i.e. fitness values) to move towards those with higher light intensities in the neighbourhood, in

order to achieve global optimality. Although each firefly serves as a search agent to increase global exploration, the empirical results [3, 4] indicate that high dimensional optimization problems still pose great challenges to FA, and the random movement of FA has a high likelihood to be trapped in local optima.

One of the limitations of the original FA model is its pure attractiveness force that moves each firefly towards the brighter counterparts in the neighbourhood. If there is no brighter firefly in the neighbourhood, the attractiveness action stagnates, and there is no alternative mechanism to drive the search out of the local optima traps. In other words, there is no effective strategy to avoid poor solutions while moving towards the optimal ones. Moreover, by following the neighbouring brighter fireflies, there is an increased likelihood that part of the population could be clustered in the same region, therefore reducing the possibility of finding the global optimum residing elsewhere. Solving the first issue results in better performance and faster convergence, while resolving the second issue tackles the premature convergence problem. Both challenges constitute the key motivation of this research.

To deal with the abovementioned challenges, this research proposes three FA variants, i.e., Repulsive Firefly Algorithm (RFA), Scattering Repulsive Firefly Algorithm (SRFA), and Enhanced SRFA (ESRFA). Besides the conventional attractiveness movement of the FA model, RFA uses a repulsive force strategy to drive the search to reach global optimality in fewer iterations. SRFA, in addition to using the repulsive force action, incorporates a scattering mechanism to explore rarely exploited distinctive search spaces to increase search diversity and overcome premature convergence. ESRFA incorporates hawk-moths' local hovering and sharp dive escaping movements to diversify the attractiveness and repulsive force actions of SRFA, as well as exchanges with historical best experiences in the neighbourhood to accelerate convergence. Specifically, the research novelties are five-fold. (1) The proposed repulsive force works together with the original attractive force to enable the search procedure to converge towards the global optima and, at the same time, avoid poor solutions. (2) The scattering strategy overcomes premature convergence by diverting a number of weak solutions to unexploited regions. (3) Exploitation-driven attractiveness and exploration-based evading mechanisms are used to enhance the search operations in diversifying the search process. (4) Interactions with the historical personal best experiences of other fireflies are conducted to accelerate convergence. (5) The proposed strategies cooperate with each other to overcome premature convergence, especially in solving high dimensional optimization problems. We evaluate the three proposed FA models with ten standard and CEC2014 [5] benchmark optimization functions. RFA, SRFA and ESRFA significantly outperform state-of-the-art FA variants and other evolutionary search methods, which include Particle Swarm Optimization (PSO) [6], Simulated Annealing (SA) [7], FA, Bat Swarm Optimization (BSO) [8, 9], Cuckoo Search Optimization (CSO) [10], Dragonfly Optimization (DFO) [11] and Ant-Lion Optimization (ALO) [12].

The research contributions are summarized, as follows:

- Three FA variants, i.e., RFA, SRFA and ESRFA, are proposed. RFA incorporates a repulsive force strategy to enable fireflies with higher light intensities to jump out of unpromising search regions to achieve fast convergence.
- SRFA employs the repulsive force action and a scattering mechanism to avoid local optima. The latter diverts a proportion of weak neighbouring solutions to an unexploited distinctive search space to increase search diversity. The repulsive behaviour and the scattering mechanism in SRFA work cooperatively to mitigate premature convergence of the original FA model. On one hand, when the repulsive force action stagnates, the scattering mechanism is able to extend the search to rarely explored regions to reduce the probability of premature convergence. On the other hand, when the scattering behaviour is unable to generate fitter solutions, the repulsive force action enables each firefly to conduct long jumps to move towards optimal regions to escape from local optima.
- ESRFA incorporates exploitation and exploration coefficients to diversify the attractiveness and repulsive operations of SRFA, respectively, and interacts with neighbouring historical best experiences to accelerate convergence. It has three key properties. Firstly, the exploitation factor simulates the mid-air hovering of hawk-moths around attraction, which enables a refined random examination of a promising neighbourhood and overcomes the local optima traps, as compared with the original attractiveness operation in FA. Secondly, the exploration-driven escaping coefficient simulates a sharp dive of moths in response to predators, which enables the search to explore a wider search space while evading from the worse solutions. The newly proposed attractiveness operation increases local exploitation while the updated evading action increases global exploration. In other words, both properties enable the search process to balance well between local exploitation and global exploration. Thirdly, a distinctive attractiveness-based operation guided by the historical personal best experiences of neighbouring fireflies is conducted to accelerate convergence.

- A comprehensive evaluation with diverse unimodal, multimodal and challenging CEC2014 optimization functions is conducted. The proposed RFA, SRFA and ESRFA models outperform the original FA model and advanced FA variants, as well as other metaheuristic search methods, significantly. They also show great robustness and superiority in dealing with complex high dimensional optimization problems.

The paper is organised as follows. A literature review on related work is presented in Section 2. The proposed RFA, SRFA and ESRFA models are introduced in detail in Section 3. Comprehensive experiments for evaluation of three proposed models are presented in Section 4. Finally, conclusions and directions for future work are given in Section 5.

2. RELATED WORK

Swarm intelligence (SI) based optimization methods have gained popularity recently [13, 14]. As a recent SI algorithm, FA is an effective metaheuristic search method on par with other existing models, in solving diverse optimization problems, which include PSO, Genetic Algorithm (GA), CSO, Artificial Bee Colony (ABC) and Ant Colony Optimization (ACO). In this section, we discuss the basic concepts of FA, different FA variants, and other recently proposed metaheuristic search methods.

2.1 Firefly Algorithm

Introduced by Yang [1], FA is inspired by the movement of fireflies based on their bioluminescence. It employs three strategies to guide the search process: (1) fireflies are unisex, and are attracted to each other; (2) attraction is proportional to the degree of brightness and inversely proportional to the distance between a pair of fireflies. As a result, less bright fireflies move towards the brighter ones in the neighbourhood. The brightest firefly moves randomly; (3) the brightness of each firefly represents the solution quality. In FA, each firefly represents a solution, which is characterised by its position and light intensity (the fitness value). The light intensity decreases as the distance to its source increases. The variation of light intensity is defined by

$$I(r) = I_0 e^{-\gamma r^2} \quad (1)$$

where r denotes the distance and I_0 represents the original light intensity when the distance $r = 0$. γ represents a fixed light absorption coefficient. The attractiveness of a firefly is defined by

$$\beta(r) = \beta_0 e^{-\gamma r^2} \quad (2)$$

where β_0 indicates the initial attractiveness when the distance $r = 0$. Equation (3) illustrates the position updating formula of each firefly, which moves a firefly, i , with a lower light intensity towards a brighter one, j , in the neighbourhood, i.e.,

$$x_i^{t+1} = x_i^t + \beta_0 e^{-\gamma r_{ij}^2} (x_j^t - x_i^t) + \alpha_t \varepsilon_t \quad (3)$$

where x_i^t and x_j^t denote the positions of fireflies i and j at the t -th iteration, respectively. r_{ij} denotes the distance between two fireflies, i and j . α_t is a randomization parameter that controls the step size of the randomized move with ε_t as a random walk behaviour defined by a Gaussian (or other) distribution. The second term in Equation (3) denotes the attractiveness behaviour while the third term implements random exploitation. The pseudo-code of the FA model [1] is presented in Algorithm 1.

Algorithm 1: Pseudo-Code of the Firefly Algorithm

1. **Start**
2. Initialize a population of fireflies randomly;
3. Evaluate each firefly using the fitness/objective function, $f(x)$;
4. Set light intensity $I = f(x)$ and light absorption coefficient γ ;
5. **While** (stopping criterion is not satisfied)// until it finds the optimal solution or the maximum number of iterations is met.
6. {
7. **For** $i = 1$ to n **do** //for all fireflies
8. {
9. **For** $j = 1$ to n **do** //for all fireflies
10. {

```

11.      If ( $I_j > I_i$ )
12.      {
13.          Move firefly  $i$  towards firefly  $j$  using Equation (3);
14.      } End IF
15.          Vary attractiveness with distance  $r$  using  $\exp[-\gamma r^2]$ ;
16.          Evaluate new solutions and update the light intensity values;
17.      } End For
18.  } End For
19.  Rank the fireflies and find the current global best,  $g_{best}$ ;
20. } End While
21. Output the most optimal solution,  $g_{best}$ ;
22. End

```

FA has several distinctive advantages. FA performs automatic subdivision of the population into subgroups. Unlike the particles in PSO, which purely follow the global best solution, each firefly in FA is a search agent, and follows multiple brighter fireflies (i.e. optimal solutions) in the neighbourhood to increase global exploration. Therefore, FA shows a great capability of solving multimodal optimization problems by finding local and global optima simultaneously.

2.2 FA Variants and Other Evolutionary Search Methods

Recently, numerous modified FA models have been proposed. Fister et al. [2] discussed different FA variants proposed for challenging real-life engineering optimization applications. Known as LFA, Yang [15] replaced the Gaussian distribution with a Lévy flight for randomization to increase exploration of the original FA model. Chaos-enhanced FA with parameter tuning was also proposed by Yang [16] to overcome stagnation. Apostolopoulos and Vlachos [17] employed the original FA model for solving economic emission load dispatch problem by minimizing both fuel cost and emission of the generating units. In addition, the FA model was used in Yang et al. [18] to solve complex non-convex economic dispatch problems with nonlinear characteristics. Horng [19] combined FA with the Linde-Buzo-Gray (LBG) algorithm for vector quantization in digital image compression. Their model outperformed LBG, PSO, and quantum PSO, significantly. Hassanzadeh et al. [20] optimized the hyper-parameters of Structure Equivalent Fuzzy Neural Network using the FA model for speech recognition tasks. An FA variant was proposed by Coelho and Mariani [21] for solving chiller loading problem in heating, ventilation, and cooling systems. In their work, Gaussian distribution was used to fine-tune the randomization parameter as well as the absorption coefficient. Huang et al. [22] developed a hybrid FA model for reactor planning and placement in underground cable transmission systems. In their model, a local random search was conducted to further improve the global best solution identified by the FA model to overcome premature convergence. Evaluated with real power systems, their work showed impressive performance. Kavousi-Fard et al. [23] incorporated a modified FA model to fine-tune the hyper-parameters of a Support Vector Regressor (SVR) for short-term electrical load forecast. Their model employed mutation, crossover, and an adaptive randomization parameter to maintain swarm diversity and balance between exploitation and exploration. Marichelvam et al. [24] undertook multi-objective hybrid flowshop scheduling problems using a discreet FA model to minimize both makespan and mean flow time. Fister et al. [25] proposed several randomized FA models by employing various probability distributions as randomization strategies, including Uniform, Gaussian, Lévy, Kent, and Logistic Chaotic maps, and random sampling in turbulent fractal cloud. Coelho et al. [26] incorporated chaotic maps to fine-tune the light absorption coefficient, γ , and the step parameter, α , for randomized movements. The proposed model was evaluated with reliability-redundancy optimization problems, and outperformed the original FA model significantly. Massan et al. [27] employed the FA model for optimization of large wind farm allocation. The model outperformed GA and a finite difference method, significantly.

Zhang et al. [28] proposed an intelligent facial expression recognition system with moth-firefly feature optimization. The model integrated FA with moth-flame optimization (MFO). It employed the spiral search capability of moths to increase local exploitation of fireflies, and the attractiveness search behaviours of FA to increase global exploration of moths. Evaluated using facial expression recognition tasks with frontal-view and side-view images, the proposed model outperformed PSO, GA, FA, MFO, and other PSO and FA variants, significantly. Twelve chaotic maps were employed by Gandomi et al. [29] to fine-tune the attractiveness coefficient of FA. The Gauss map for parameter tuning yielded the most competitive performance in undertaking the Sphere and Griewank functions. Mitić et al. [30] proposed a chaotic fruit fly optimization algorithm (FOA). Ten different chaotic maps were used for dynamic parameter adjustment. Evaluated using fourteen benchmark functions, the Chebyshev map produced the best performance. FOA also outperformed other advanced chaotic accelerated search methods, including chaotic BSO, chaotic FA, chaotic CSO, chaotic ABC and chaotic PSO. Another FA variant was proposed by Amiri et al. [31] to deal

with multi-objective optimization problems for community detection in complex social networks. Besides employing the Logistic chaotic map for randomization parameter tuning, it used self-adaptive probabilistic mutation mechanisms to enable each firefly in the population to have a chance to mutate, in order to increase population diversity and overcome local optima. Evaluated using several real world and synthetic data sets, the proposed model showed superiority in achieving global optima. Teshome et al. [32] proposed an FA variant for optimization of power electronics applications. Instead of using the location of each brighter neighbouring firefly for position updating, the average position of all brighter fireflies in the neighbourhood was used to guide the search. In comparison with the original FA model, the proposed model achieved fast convergence toward optimal positions directly. Yang [33] proposed a multi-objective FA model to deal with multi-criteria design optimization in industrial engineering. Verma et al. [3] developed a modified FA algorithm, i.e. Opposition and Dimensional based FA (ODFA), for solving high-dimensional optimization problems. It used an opposition-based methodology to initialize a second swarm, which was opposite to the positions of the original population to increase exploration. Their global best solution was obtained by identifying the best solution in each dimension. Instead of using neighbouring brighter fireflies for position updating, their method used the current global best solution to update each firefly. Evaluated with eleven benchmark functions, ODFA outperformed basic search methods such as FA, PSO, and ACO. SFA and LSFA were developed by Alweshah and Abdullah [34]. SA was employed to further improve the global best solutions of FA and LFA, respectively. Both FA variants were used to optimize the weights of a neural network classifier to improve classification accuracy. Evaluated with eleven standard benchmark data sets, the proposed algorithms achieved impressive classification accuracy, outperforming FA and LFA. Kazem et al. [35] proposed a chaotic FA (CFA) for optimization of hyper-parameters of SVR, and demonstrated its application to stock market price forecasting. The proposed CFA model employed the Logistic chaotic map for population initialization. The Logistic map was also used to provide chaotic attractiveness movement by fine-tuning the randomization component. Evaluated with challenging stock market time series data from NASDAQ, the model outperformed GA-based SVR, chaotic GA-based SVR, FA-based SVR, artificial neural networks, and adaptive neuro-fuzzy inference systems, significantly.

Modified FA models have also been used to solve diverse real-life optimization problems. Zhao et al. [36] proposed a multi-objective FA model for solving radio-frequency identification (RFID) network planning. A hybrid firefly differential evolution (HFDE) algorithm was proposed by Dash et al. [37] for designing multipurpose digital linear phase double band filters. To increase global search capabilities, HFDE integrated FA with DE. It outperformed other search methods in terms of design accuracy and computational efficiency. Kaur and Ghosh [38] developed a fuzzy-firefly model for optimizing unbalanced distribution networks with the aim to reduce power losses and improve voltage profiles. Fuzzified objective functions were employed in the model. Evaluated using 25-node and 19-node unbalanced networks, the proposed model showed impressive performance, and outperformed GA, PSO, and ABC. Xiong et al. [39] conducted interval-valued time series forecasting of stock price index using a Multiple-output Support Vector Regression (MSVR) model. FA was used to identify the optimal parameter setting of MSVR. The resulting model, FA-MSVR, yielded better performances in comparison with those of PSO-MSVR and GA-MSVR for one- and multi-step-ahead stock price index forecasting. Kalantzis et al. [40] developed a GPU-based FA model for constrained optimization of intensity modulated radiation therapy treatment planning. The model performed better than a sequential version of the algorithm executed on a CPU for evaluation of both prostate and head and neck cancer cases. Kisi et al. [41] employed SVM integrated with FA for prediction of lake levels. FA was used to identify the optimal SVM parameters. The model outperformed neural networks combined with genetic programming (GP) for one-day ahead lake level prediction. A similar model that combined both SVM and FA was applied to region-based image retrieval by Kanimozhi and Latha [42]. Mistry et al. [43] proposed a modified FA model for feature selection and facial expression recognition. The global best solution of FA was enhanced by Gaussian, Cauchy and Levy distributions. The model outperformed PSO, GA, and other facial expression recognition methods. Su et al. [44] used FA for optimal band selection and parameter setting identification for the extreme learning machine (ELM). The model was applied to hyperspectral image classification, and outperformed other classical search methods in experiments using two hyperspectral image data sets, i.e. HYDICE and HYMAP. Zhang et al. [45] proposed a return-cost-based binary FA model for feature selection, while Rodrigues et al. [46] employed FA for multi-thresholding in image segmentation.

Other state-of-the-art swarm-based metaheuristic search methods have also been proposed for solving diverse optimization problems, e.g. an improved simplified swarm optimization [47] and chaotic BSO [48]. Motivated by the hunting mechanism of antlions, Mirjalili [12] proposed an ALO model that comprised hunting prey actions of ants such as randomization, building trap, entrapment, catching preys, and re-building traps. Tested with nineteen benchmark functions, and constrained optimization problems, ALO

showed superior performance in avoiding local optima and possessed a great capability of solving unimodal and multimodal optimization problems. Mistry et al. [49] proposed micro-GA embedded PSO feature optimization for facial expression recognition. The proposed PSO algorithm included a new velocity updating mechanism with sub-dimension based local facial feature search. It also adopted a non-replaceable memory and a small-population secondary swarm to increase search diversity. Ensemble classifiers were used for recognition of seven facial expressions. Evaluated with challenging facial expression recognition tasks, the proposed algorithm outperformed PSO, GA and other state-of-the-art PSO variants significantly. Srisukkhom et al. [50] employed Bare-Bones PSO based feature selection for blood cancer detection. Wan et al. [51] integrated binary coded ACO (BACO) with the GA for feature optimization. Two models were proposed, i.e., a visibility density model and a pheromone density model. The GA outputs were used as the initial visibility and pheromone information of BACO, respectively. The algorithm outperformed the GA, binary PSO, binary ACO and binary differential evolution (DE) significantly when evaluated with well-known UCI data sets. Neoh et al. [52] proposed both direct similarity and Pareto-based feature selection with a layered cascade model for facial expression recognition. The former took within-class variation into account while the latter employed both between- and within-class variations for discriminative facial feature selection. Haklı and Uğuz [53] incorporated PSO with Levy flight. Their work re-distributed the particles that did not achieve improvement in a number of iterations to new unexploited search regions. Moreover, Kang et al. [54] combined PSO with the artificial immune system for structure damage detection. Non-dominated Sorting Genetic Algorithm-II (NSGA-II) was utilized for ensemble classifier construction by Zavaschi et al. [55]. Optimization was performed by minimization of both classification error and ensemble size, in order to construct the most optimal ensemble classifier. NSGA-II was also employed by Soyel et al. [56] for discriminative facial feature optimization to enhance class separation in facial expression recognition. A modified NSGA-II based feature selection model was developed by Huang et al. [57] for churn prediction in telecommunications. The model was able to identify feature subsets of different sizes to overcome the limitations of original NSGA-II, in which the search process purely focused on finding features of fixed sizes. Evaluation results indicated the efficiency of the proposed model. Other metaheuristics models for optimization tasks are also available in the literature, which include parameter identification [58], image segmentation [59], and feature selection [60], and promising future directions are discussed in [61].

3. METHODOLOGY

In this research, we propose three FA variants, i.e., RFA, SRFA and ESRFA, to mitigate the premature convergence problem of the original FA model. RFA employs the repulsive force strategy while SRFA uses both the repulsive force and the scattering mechanisms to increase local and global search capabilities. Motivated by both RFA and SRFA, another enhanced algorithm, ESRFA, is proposed, which embeds the hovering-based attractiveness behaviour, exploration-based repulsive action, and interaction with the best memories of neighbouring fireflies to accelerate convergence.

3.1 The Proposed RFA and SRFA Models

We introduce the proposed RFA and SRFA models in detail in this section. The repulsive force strategy complements the attractiveness behaviour of the original FA model. The aim is to accelerate brighter fireflies to move away from less promising search regions to achieve fast convergence. The attractiveness and repulsive search behaviours work alternatively to guide the search process in RFA. Specifically, when there are brighter fireflies in the neighbourhood, the less bright ones are attracted to the brighter ones by the attractiveness movement as in the original FA model. On the contrary, when there are no brighter fireflies in the neighbourhood, instead of purely performing randomized movement, the fireflies employ the repulsive force action to move away from those with lower light intensities. This repulsive force movement is defined in Equation (4).

$$x_i^{t+1} = x_i^t - \beta_0 e^{-\gamma r_{ij}^2} (x_j^t - x_i^t) \times r_f - \alpha_t \varepsilon_t \quad (4)$$

where r_f denotes a repulsive force factor with the assumption that firefly i has better fitness than that of firefly j . Equation (4) indicates that brighter firefly i moves away from the less optimal solution j . Alternatively, this can be interpreted as the less bright firefly j pushes the brighter firefly i away from itself.



Figure 1 Flowcharts of the proposed two FA variants, i.e. (a) RFA and (b) SRFA

In comparison with the attractiveness behaviour in Equation (3), the repulsive movement enables the current brighter fireflies to move to the opposite direction of those with less brightness. The repulsive force parameter, r_f , determines the impact of the repulsive force for position updating of the current firefly i . In other words, it determines the fraction of repulsive force to be applied for fine-tuning the repulsive movement.

Figure 1 (a) shows a flowchart of the proposed RFA and SRFA variants. The key steps of RFA are as follows. An initial population of fireflies is generated by randomly distributing the fireflies in the search space. Then, each firefly is evaluated using the objective (fitness) function formulated with respect to the problem domain. Next, the fireflies are ranked based on their fitness values and the number of neighbours of each firefly is calculated based on a neighbour distance threshold (DT). Subsequently, the attractiveness and

repulsive force movements are used alternatively to move fireflies towards optimal search regions. The attractiveness or repulsive movement is applied based on whether the neighbouring firefly j has higher or lower degree of brightness than that of the current firefly i , respectively. If firefly j has better fitness than firefly i , firefly i moves towards firefly j based on the attractive movement as defined in the original FA model. Otherwise, if firefly j has less fitness than firefly i , firefly i is pushed away from j by employing the repulsive force movement defined in Equation (4). A repulsive immune threshold, r_t , is also employed to ensure a certain portion (i.e., 10% of the population in this research) of top ranked fireflies are kept unaffected by the repulsive force movement. The immune threshold ensures that certain top ranking fireflies are not pushed away from their optimal positions, in an attempt to balance between convergence speed and search diversity. Overall, in comparison with the original FA model where the brighter firefly only performs random movement, RFA enables the brighter firefly not only performs random walk, but also carries out the repulsive force action to avoid unpromising search regions and achieve fast convergence.

Another FA variant, SRFA, is also proposed in this research. SRFA incorporates both the repulsive force movement and a scattering mechanism to overcome premature convergence of the original FA model. The scattering mechanism diverts a proportion of weak neighbouring solutions randomly to other distinctive search spaces, therefore increasing search diversity. As shown in Figure 1 (b), extra steps are dedicated to SRFA for performing the scattering movement (marked with dotted squares). The pseudo-code of SRFA, which includes both the repulsive force and the scattering actions, is provided in Algorithm 2.

In SRFA, the neighbouring distance threshold, DT , is used to set the maximum distance between two fireflies to be considered as neighbours. After employing the repulsive force action to move a brighter firefly, i , away from the less optimal solution, j , the neighbouring distance threshold is used to determine if the weak firefly j is a neighbour of firefly i . If it is, the neighbour counter of firefly i is incremented by one. In this way, the weak solutions in the neighbourhood of the current promising solution i , can be identified. In order to balance between search diversity and convergence speed, some neighbours (i.e. 50% in this research) of firefly i are preserved while others are scattered away to new search regions. As such, a threshold for the maximum permissible number of neighbours, nt , is used for firefly i . If the neighbour number exceeds the nt percentage of the population, SRFA scatters extra neighbours randomly to other rarely explored search regions, in order to increase exploration. As a result, the scattering mechanism keeps a maximum nt percentage of the population as the neighbours for a promising firefly i and scatters extra neighbouring fireflies away to increase search diversity. This random scatter behaviour is defined in Equation (5).

$$x_k^d = L_b^d + \mu \times (U_b^d - L_b^d) \quad (5)$$

where μ is a randomly generated vector with each element in the range of $[0, 1]$. L_b^d and U_b^d represent the lower and upper bounds in the d -th dimension, respectively, while x_k^d denotes the position of firefly k in the d -th dimension.

The repulsive force action and the scattering behaviour embedded in SRFA work cooperatively to overcome local optima. If the repulsive force action is unable to find a fitter offspring, the scattering behaviour is able to discover other distinctive regions to widen the search region and avoid local optima. On the other hand, if the random scattering behaviour is not able to identify more optimal search regions, the repulsive force action enables the brighter fireflies to perform long jumps and move towards more promising search regions, in order to overcome premature convergence and achieve global optimality.

Algorithm 2: Pseudo-Code of the Scattering Repulsive Firefly Algorithm (SRFA)

Initialization

Objective function: $f(X)$, $X=(x_1, x_2, x_3, x_4, \dots, x_d)$;
 Generate an initial population of fireflies X_i ($i=1, 2, 3, \dots, n$);
 Formulate light intensity I so that it is associated with $f(X)$;
 Define absorption coefficient γ ;
 Initialize thresholds as follows;

Default Thresholds

- | | | |
|---------------------------------|---|--|
| 1) Repulsive Force Factor | $r_f=0.6$; | //1/2 (default) |
| 2) Repulsive Immune Threshold | $r_t = 0.1$; | //10% of the population (default) |
| 3) Neighbour Distance Threshold | | //20% of root square differences in each dimension, |
| | $DT = dt \times \sqrt{(U_b - L_b)^2} \times D$, where $dt = 0.2$; | where U_b and L_b represent upper and lower bounds |

<pre> 4) Neighbour Number Threshold $nt = 0.5$; 1 Start 2 $t = 0$; 3 While ($t < \text{Max_Generation}$) { 4 $S = \text{Zeros}(n)$; 5 Vary attractiveness with distance r via $\exp(-\gamma r^2)$; 6 Evaluate light intensities of the population; 7 Rank the fireflies; 8 For $i = 1$ to n (all n fireflies) { 9 $b_i = 0$; 10 For $j = 1$ to n (all n fireflies) { 11 If ($S_j = 1$) { 12 Continue; 13 } End If 14 Calculate distance (r) between firefly i and firefly j; 15 If ($I_j > I_i$) { 16 Move firefly i towards j using Equation (3); 17 } Else If ($i > r_t \times n$) { 18 Move firefly i away from firefly j using Equation (4); 19 If ($r < DT$) { 20 $b_i = b_i + 1$; 21 add j to NA; 22 } End If 23 } End If 24 } End For j 25 If ($b_i > (nt \times n)$) { 26 For $k = (nt \times n)$ to b_i { 27 $S_{NA[k]} = 1$; 28 Scatter firefly $NA[k]$ using Equation (5); 29 } End For k 30 } End If 31 Clear NA; 32 } End For i 33 $t = t + 1$; 34 Rank all the fireflies and find the current global best; 35 } End While 36 Output the most optimal solution; 37 End </pre>	<p>for each dimension respectively with D denoting the problem dimensions. //50% of the population (default)</p> <p>//S is the array containing scattering flags for each firefly, which are initialized as '0'. S_i represents the scattering flag for firefly i.</p> <p>//Number of neighbours initialized as 0 for firefly i</p> <p>//Skip if the firefly is marked for scattering //Go back to line 10 and increase j by 1.</p> <p>//Firefly j is brighter than firefly i</p> <p>//Repulsive immune threshold – 10% of the population by default</p> <p>//Less than the neighbour distance threshold, DT //Increase the neighbour counting //NA is the array containing the indices of neighbouring fireflies</p> <p>//Start of the scattering part (more than the neighbour number threshold, $nt \times n$)</p> <p>//Scatter the k^{th} firefly in NA</p> <p>//End of the scattering part</p>
--	---

3.2 The Proposed ESRFA Model

An enhanced SRFA model, i.e. ESRFA, is also proposed in this research. ESRFA integrates both hovering/exploitation-driven attractiveness and exploration-driven escaping mechanisms, as well as interaction with historical best experiences of neighbouring fireflies to overcome premature convergence. We introduce each key proposed mechanism in detail, as follows.

Firstly, we identify several aspects to further improve the abovementioned SRFA model. As an example, SRFA employs the attractiveness action of the original FA model shown in Equation (3) to guide the fireflies with lower light intensities to move towards brighter ones. When the distance between two fireflies are close, $e^{-\gamma r_{ij}^2} \rightarrow 1$, and based on Equation (3), the movement towards the brighter firefly is more likely to take a comparatively larger step. In order to enable fine-tuning the movement towards the neighbouring promising solution, we propose a new attractiveness operation by incorporating an exploitation factor. Such a fine-

tuned movement enables a granular random examination of the neighbourhood around a promising solution to avoid local optima traps. Equations (6)-(8) define the updated attractiveness operation.

$$x_i^{t+1} = x_i^t + (\beta \times h)(x_j^t - x_i^t) + \alpha_t \varepsilon_t \quad (6)$$

$$h = \sin(\sin(rand^2)) \quad (7)$$

$$\beta = (1 - \beta_{min}) \times e^{-\gamma r_{ij}^2} + \beta_{min} \quad (8)$$

where h and β denote the exploitation and attractiveness coefficients, respectively. Equation (7) defines the proposed exploitation factor h , while Equation (8) represents the updated attractiveness coefficient β , with $\beta_{min} = 0.2$, according to Yang [1, 15].

The exploitation factor is incorporated with the attractiveness coefficient to implement fine-tuning. Such an exploitation factor simulates the hovering behaviour of hawk-moths around attraction (e.g. lights) or food (e.g. flower). Owing to the efficiency in employing standard sine and cosine functions (i.e. $\sin(2\pi \times rand)$ or $\cos(2\pi \times rand)$) to explore the search space [62], we consider variations of such functions for the implementation of the exploitation coefficient. Since we aim to implement both refined and erratic examination of optimal neighbourhoods to simulate the hovering actions of moths, Equation (7) that represents an irregular and subtle movement is proposed.

Figure 2 illustrates the exploitation flight/path (the orange line) generated by Equation (7) using $\sin(\sin(rand^2))$ and a reference path (the black line) produced by a standard $\sin(rand)$ function. As shown in Figure 2, Equation (7) yields a refined search step (as indicated by the orange line) than that generated using the standard $\sin(rand)$ function (as indicated by the black line). Therefore, the former produces a granular step as the exploitation factor to enable more detailed examination of a promising neighbourhood in comparison to that generated by a standard sine function. Moreover, the proposed $\sin(\sin(rand^2))$ function in Equation (7) denotes comparatively more irregular/erratic wave-like patterns than those generated by the standard sine function, which represent regular waveforms. Therefore, Equation (7) is selected as an attempt to simulate the rapid swing-hovering behaviours of hawk-moths to implement the local exploitation factor.

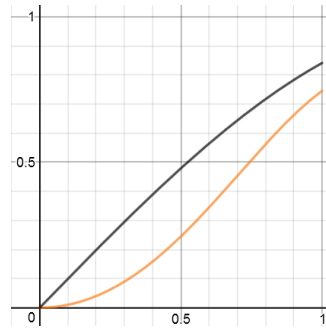


Figure 2 The exploitation path generated by Equation (7) (orange line) and the reference path generated by a standard $\sin(rand)$ function (black line)

As shown in Figure 2, for a randomly selected value between (0, 1), it produces an exploitation coefficient in the range of (0, $\sin(\sin(1))$). Since multiplying this exploitation factor with the attractiveness coefficient tunes down the movement step, it enables a random examination with different degrees of granularity pertaining to the local neighbourhood of promising firefly j to achieve fine-tuning.

Besides the attractiveness operation, we further improve the proposed repulsive force action in SRFA by incorporating an escaping coefficient to increase global exploration. It simulates the anti-predator behaviour of moths which take a sharp dive or fall in response to the sonar clicks of bats to explore a wider search space. Equations (9)-(10) define the newly updated repulsive force action. It employs a comparatively larger repulsive factor to conduct a wider global exploration in comparison with that of the repulsive action defined in Equation (4).

$$x_i^{t+1} = x_i^t - (x_j^t - x_i^t) \times \theta - \alpha_t \varepsilon_t \quad (9)$$

$$\theta = (r_f - \theta_{min}) \times e^{-\gamma r_{ij}^2} + \theta_{min} \quad (10)$$

where θ indicates an escaping coefficient defined in Equation (10). The repulsive force factor r_f is set to 0.6 according to our experimental studies (see Section 4) with $\theta_{min} = 0.2$ based on empirical experiments.

In comparison with the repulsive factor of SRFA, $e^{-\gamma r_{ij}^2} \times r_f$, employed in Equation (4), the escaping coefficient θ defined in Equation (10) generates a comparatively stronger repulsive impact. As shown in Figure 3, the orange line indicates the coefficients generated by Equation (10) while the black line shows the factors generated by $e^{-\gamma r_{ij}^2} \times r_f$ in Equation (4). It enables the repulsive action defined in Equation (9) to perform a larger movement away from less optimal regions. Therefore, the evading operation defined in Equation (9) explores a wider search space in comparison with the SRFA movement denoted by Equation (4). Overall, the updated repulsive operation enhances global exploration to avoid stagnation. In other words, the proposed attractiveness and evading operations increase local exploitation and global exploration, respectively, and cooperate with each other to attain global optima.



Figure 3 The escaping coefficients generated using Equation (10) (orange line) and the formula, $e^{-\gamma r_{ij}^2} \times r_f$, used in Equation (4) (black line), respectively

Besides the proposed attractiveness and repulsive actions, the search process is further improved by taking the historical best personal experience of each neighbouring firefly into account. The detailed operation is defined in Equations (11)-(12).

$$x_i^{t+1} = x_i^t + (pBest_j - x_i^t) \times \beta^p \quad (11)$$

$$\beta^p = \beta_0 e^{-\gamma r_{ipBest_j}^2} \times \left(\frac{fitness_i - fitness_{pBest_j}}{fitness_{max} - fitness_{min}} \right) \quad (12)$$

where $pBest_j$ represents the historical personal best experience of neighbouring firefly j . β^p denotes the neighbouring historical attractiveness coefficient, as defined in Equation (12). The distance parameter r in Equation (12) is calculated using the distance between the current solution i and the historical best personal experience of the neighbouring firefly j . Moreover, as indicated in Equation (12), the fitness score difference between the current firefly and the best experience of neighbouring firefly j is used for determining β^p , whereby $fitness_i$ and $fitness_{pBest_j}$ denote the fitness scores of the current firefly i , and the historical best experience of firefly j , respectively, with $fitness_{max}$ and $fitness_{min}$ representing the maximum and minimum fitness scores of the current iteration. This operation is activated when the current firefly has a lower fitness score than that of the personal best of neighbouring firefly j . In this way, the historical best experiences of all neighbouring individuals are used to accelerate convergence. Moreover, note the movement step is adjusted not only by the distance between two fireflies but also by the difference of their fitness/brightness. The scattering action of SRFA is also embedded in ESRFA to divert a proportion of weak neighbouring solutions of firefly i to an unexploited distinctive search space to increase search diversity.

Algorithm 3 shows the detailed pseudo-code of the proposed ESRFA model. Overall, the three proposed strategies diversify the search process by embedding hawk-moth motivated exploitation and exploration coefficients, as well as learning from neighbouring historical best experiences. We evaluate the proposed ESRFA, SRFA and RFA models using diverse challenging optimization problems, as follows.

Algorithm 3: Pseudo-Code of the Proposed ESRFA Model

- | |
|---|
| <ol style="list-style-type: none"> 1. Start 2. Initialize a population of fireflies randomly; 3. Evaluate each firefly using the fitness/objective function, $f(x)$; |
|---|

```

4. Set light intensity  $I = f(x)$  and light absorption coefficient  $\gamma$ ;
5. Initialize all thresholds dedicated to SRFA;
6. While (stopping criterion is not satisfied)// until it finds the optimal solution or the maximum number of
iterations is met.
7. {
8.   For  $i = 1$  to  $n$  do //for all fireflies
9.   {
10.    For  $j = 1$  to  $n$  do //for all fireflies
11.    {
12.     If ( $j$  is marked for scattering)
13.     Continue; //Skip if the firefly is marked for scattering
14.     End If
15.     If ( $I_j > I_i$ )
16.     Move firefly  $i$  towards firefly  $j$  using Equation (6);
17.     Else If ( $i > r_t \times n$ )
18.     Move firefly  $i$  away from firefly  $j$  using Equation (9);
19.     End If
20.     If personal best experience of  $j$ ,  $pBest_j$ , is better than  $i$ ,
21.     Move firefly  $i$  towards  $pBest_j$  using Equation (11);
22.     End If
23.     Increase the neighbour counting;
24.   } End For  $j$ 
25.   Scatter if more neighbours than the allowed threshold;
26. } End For  $i$ 
27. Rank the fireflies and find the current global best,  $g_{best}$ ;
28. } End While
29. Output the most optimal solution,  $g_{best}$ ;
30. End

```

4. EVALUATION

A comprehensive evaluation has been conducted to evaluate RFA, SRFA and ESRFA models. To indicate efficiency of the proposed models, several FA variants proposed in recent years, i.e. ODFA [3], SFA [34], LSFA [34], and CFA [35] and other metaheuristic search methods, i.e. PSO, SA, FA, BSO, CSO, DFO and ALO, have been implemented for comparison. All algorithms are implemented using MATLAB R2016b. Details of the experimental study are as follows.

4.1 Experimental Setup

We use ten standard benchmark optimization functions to evaluate and compare our proposed models with state-of-the-art FA variants and conventional methods. These benchmark functions have been widely used for evaluation of swarm-based optimization algorithms [3, 11, 12, 30, 48]. We list these standard benchmark functions in Table 1, which include various physical properties and shapes. Artificial landscapes 1, 3, 4 and 5 contain many local minima; artificial landscapes 6, 8, and 9 are bowl-shaped functions; landscapes 2 and 7 are valley-shaped functions, while function 10 as a plate-shaped function. These artificial landscapes also represent different optimization problems including multimodal (1, 3, 4, and 5) and unimodal (2, 6, 7, 8, 9, and 10) problems.

In addition, challenging CEC2014 benchmark functions [5] have been employed for evaluation. This CEC test suite includes 30 test functions with 3 rotated unimodal, 13 shifted and rotated multimodal, 6 hybrid, and 8 composition functions. These test functions represent challenging real-parameter optimization problems. We use these shifted, rotated, hybrid and composition benchmark functions from CEC2014 test suite to further evaluate the proposed models and compare the results with those from other methods.

The following settings of the proposed models and other methods are used for evaluation of standard and CEC2014 test functions. These default settings are based on empirical studies and related publications in the literature [1, 3, 6, 7, 8, 10, 11, 12, 15, 28, 34, 35, 49, and 60], which yield the best trade-off between model performance and computational efficiency for diverse general optimization problems. As an example, we apply the following default settings of FA, i.e. initial attractiveness=1.0, randomization parameter=0.25, and absorption coefficient=1.0. These settings have been applied to the proposed models and other FA variants (i.e., ODFA [3], LSFA [34], SFA [34] and CFA [35]). SA applies the following settings [1], i.e. the initial temperature=1.0, cooling factor=0.95 with the final stopping temperature=1e-10. These settings of FA and SA have been applied to SFA [34] and LSFA [34]. Based on empirical studies [1], the discovery probability

of CSO is set to 0.25, while the PSO parameters are, maximum velocity=0.6, inertia weight=0.78, and acceleration constants=1.2. The weights of the five social behaviours pertaining to dragonflies in DFO are, separation weight=0.1, alignment weight=0.1, cohesion weight=0.7, food factor=1, and enemy factor=1. Moreover, the initial inertia weight in DFO is set to 0.9, which decreases according to the number of iterations. Finally, BSO employs the following settings, i.e. loudness=0.5 and pulse rate=0.5. ALO does not require any operation parameters. In the experiments, we have used the following default settings for our models, i.e. repulsive force factor, $r_f=0.6$, repulsive immune threshold, $r_t=0.1$, and neighbour number threshold, $nt = 0.5$. Table 2 shows the parameter settings for all the employed methods. We apply the maximum iterations of 500, the population size in the range of [10, 100], and the dimension of 2^i , where $i=1$ to 8, for all methods to evaluate their efficiency using the abovementioned ten standard benchmark functions.

Table 1 The standard benchmark functions

No.	Name	Formula	Range
1.	Ackley	$f(\mathbf{x}) = -a \exp \left(-b \sqrt{\frac{1}{d} \sum_{i=1}^d x_i^2} \right) - \exp \left(\frac{1}{d} \sum_{i=1}^d \cos(cx_i) \right) + a + \exp(1)$ $a = 20, b = 0.2 \text{ and } c = 2\pi$	[-15, 30]
2.	Dixon-Price	$f(\mathbf{x}) = (x_1 - 1)^2 + \sum_{i=2}^d i (2x_i - x_{i-1})^2$	[-10, 10]
3.	Griewank	$f(\mathbf{x}) = \sum_{i=1}^d \frac{x_i^2}{4000} - \prod_{i=1}^d \cos \left(\frac{x_i}{\sqrt{i}} \right) + 1$	[-600, 600]
4.	Levy	$f(\mathbf{x}) = \sin^2(\pi w_1) + \sum_{i=1}^{d-1} (w_i - 1)^2 [1 + 10 \sin^2(\pi w_i + 1)] + (w_d - 1)^2 [1 + \sin^2(2\pi w_d)]$, where $w_i = 1 + \frac{x_i - 1}{4}$, for all $i = 1, \dots, d$	[-15, 30]
5.	Rastrigin	$f(\mathbf{x}) = 10d + \sum_{i=1}^d [x_i^2 - 10 \cos(2\pi x_i)]$	[-5.12, 5.12]
6.	Rotated Hyper-Ellipsoid	$f(\mathbf{x}) = \sum_{i=1}^d \sum_{j=1}^i x_j^2$	[-65, 65]
7.	Rosenbrock	$f(\mathbf{x}) = \sum_{i=1}^{d-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	[-5, 10]
8.	Sphere	$f(\mathbf{x}) = \sum_{i=1}^d x_i^2$	[-5.12, 5.12]
9.	Sum of Different Powers	$f(\mathbf{x}) = \sum_{i=1}^d x_i ^{i+1}$	[-1, 1]
10.	Zakharov	$f(\mathbf{x}) = \sum_{i=1}^d x_i^2 + \left(\sum_{i=1}^d 0.5ix_i \right)^2 + \left(\sum_{i=1}^d 0.5ix_i \right)^4$	[-5, 10]

Table 2 Summary of parameter settings for different optimization algorithms

Methods	Parameter settings
SA	Initial temperature = 1.0, Cooling factor = 0.95, Stopping temperature = 1e-10
CSO	Discovery probability = 0.25
PSO	Maximum velocity = 0.6, Inertia weight = 0.78, Acceleration constants = 1.2
DFO	Separation weight = 0.1, Alignment weight = 0.1, Cohesion weight = 0.7, Food factor = 1, Enemy factor = 1, Initial inertia weight = 0.9
BSO	Loudness = 0.5, Pulse rate = 0.5
LSFA [34], SFA [34]	FA – Attractiveness = 1.0, Randomization parameter = 0.25, Absorption coefficient = 1.0 SA – Initial temperature = 1.0, Cooling factor = 0.95, Stopping temperature = 1e-10
FA, ODFA [3], CFA [35]	Attractiveness = 1.0, Randomization parameter = 0.25, Absorption coefficient = 1.0
RFA, SRFA, ESRFA	Attractiveness = 1.0, Randomization parameter = 0.25, Absorption coefficient = 1.0 Repulsive force factor = 0.6, Repulsive immune threshold = 0.1, Neighbour number threshold = 0.5 (for SRFA and ESRFA).

4.2 Evaluation Results

We conduct several experiments to compare the proposed models with advanced FA variants and other metaheuristic search methods. The first experiment focuses on the convergence speed among different algorithms using the standard benchmark functions in Table 1. We then conduct performance comparison among different algorithms by increasing the dimension exponentially while maintaining a fixed size of population and the number of iterations. Experiments have been conducted by increasing the population size with an incremental step of 10 while maintaining a fixed size of dimension and the number of iterations. Finally, we use the standard benchmark functions with fixed high dimensional experimental settings for performance comparison. The CEC2014 benchmark functions have also been employed to further demonstrate the efficiency of the proposed models. The comparison between the proposed models and an exact algorithm is also conducted for solving the Traveling Salesman Problem (TSP).

4.2.1 Performance Comparison of Different Parameter Settings

To measure the impact of different parameter settings on the performances of SRFA, a series of experiments has been conducted using diverse settings of the following key optimization parameters, i.e. repulsive force factor r_f , repulsive immune threshold, r_t , neighbour number threshold, nt , and neighbour distance threshold, DT . The standard benchmark functions shown in Table 1 are evaluated with 30 test runs to generate the results using each parameter setting. The average results of all ten test functions pertaining to r_f and r_t are shown in Figure 4.

As indicated in Figure 4, for each test function, we conduct 30 runs using each parameter setting in the range of $[0, 1]$ with an incremental step of 0.1 for both r_f and r_t . The gradual increase of the repulsive force factor, r_f , indicates that the repulsive force action has more and more impact on the search process. Figure 4 (a) indicates that SRFA is able to achieve optimal performances when r_f is in the range of $[0.1, 0.9]$. Therefore, 0.6 is selected as the default setting for the proposed RFA, SRFA and ESRFA models in our experiments. On the other hand, the increase of the repulsive immune threshold, r_t , indicates that more and more proportions of the population are eliminated from the repulsive behaviour, and the proposed models are more and more likely to behave like the original FA model. Figure 4 (b) indicates that SRFA achieves better performances when r_t is in the range of $[0, 0.7]$. Therefore, we use the setting of $r_t = 0.1$, i.e. 10% of the population, in SRFA. This indicates that 10% of the top solutions are unaffected by the repulsive strategies, and remain at the original optimal positions to enable fast convergence.

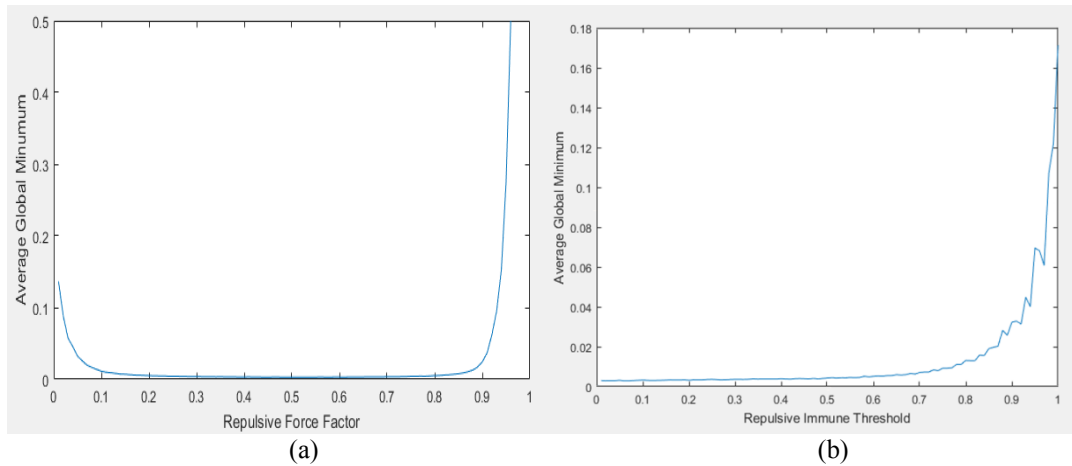


Figure 4 The evaluation results of SRFA along with the increase of repulsive force factor r_f (a) and repulsive immune threshold r_t (b), respectively (with fixed settings for other optimization parameters)

As discussed earlier, the neighbour number threshold nt is used to determine the maximum number of neighbours to keep for a promising firefly, i . If the neighbour number exceeds nt (in percentage), SRFA scatters any extra neighbours randomly to other rarely explored search regions to increase exploration. In SRFA, nt is used as the criterion to activate the scattering behaviour. A smaller nt is more likely to activate the scattering behaviour while a larger value becomes less likely to activate the corresponding action. Figure 5 (a) shows the mean global minima of 30 runs along with nt with an increment step of 10% for evaluation of all standard benchmark functions. SRFA achieves the best performances with nt in the range of $[0.5, 0.9]$. Therefore, we employ a balanced threshold, i.e. $nt = 0.5$, for activation of the scattering behaviour in SRFA, in order to achieve the best trade-off between swarm diversity and convergence speed.

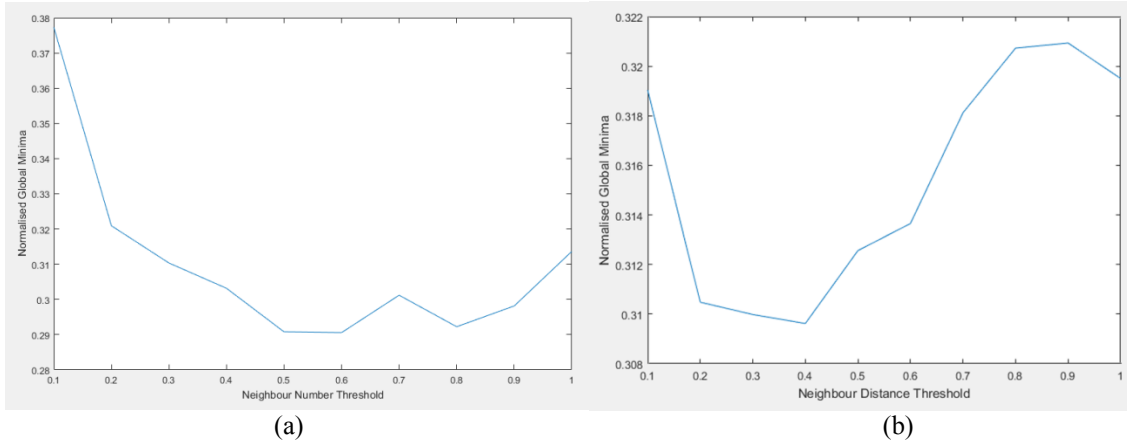


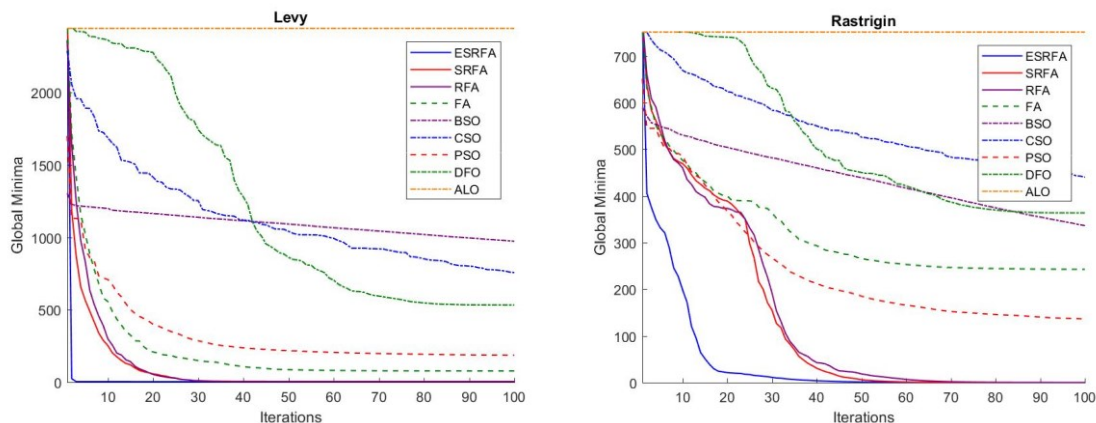
Figure 5 The evaluation results of SRFA along with the increase of (a) neighbour number threshold nt and (b) dt as the key parameter in the neighbour distance threshold, respectively (with fixed settings for other optimization parameters)

The neighbour distance threshold, DT , employs a dt proportion of the root square differences between the lower and upper boundaries in each dimension for identifying the neighbouring fireflies. A larger dt imposes a comparatively larger search space as the neighbourhood, therefore it is more likely to gain a large number of neighbours to activate the scattering behaviour to increase search diversity. On the contrary, a smaller dt contributes a smaller neighbourhood with fewer fireflies as neighbours, therefore the scattering behaviour is less likely to be activated, which may lead to premature convergence. The evaluation results of increasing nt in the range of $[0, 1]$ are shown in Figure 5 (b). The average of the mean global minima for the evaluation of all the test functions with 30 runs for each parameter setting is used to generate the plot. As illustrated in Figure 5 (b), SRFA achieves the best performances with dt in the range of $[0.2, 0.4]$. Therefore, we employ $dt=0.2$, i.e. 20% of root square differences in each dimension, as the criterion for identification of neighbours in SRFA.

Based on the above experimental observations, we set the key parameters of SRFA as follows: $r_f = 0.6$, $r_t = 0.1$, $nt = 0.5$ and $dt = 0.2$, in all subsequent experiments. These optimal settings are also employed in ESRFA.

4.2.2 Convergence Comparison

In this experiment, we compare the convergence performance of RFA SRFA and ESRFA with FA variants and other swarm-based methods using the ten standard benchmark functions shown in Table 1.



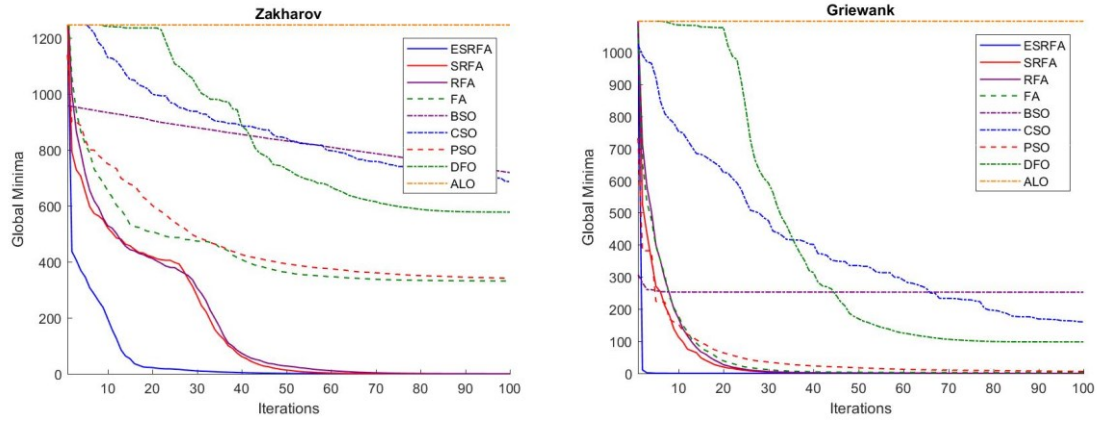


Figure 6 Convergence curves of the proposed models and other relevant methods for the Levy, Rastrigin, Zakharov, and Griewank functions

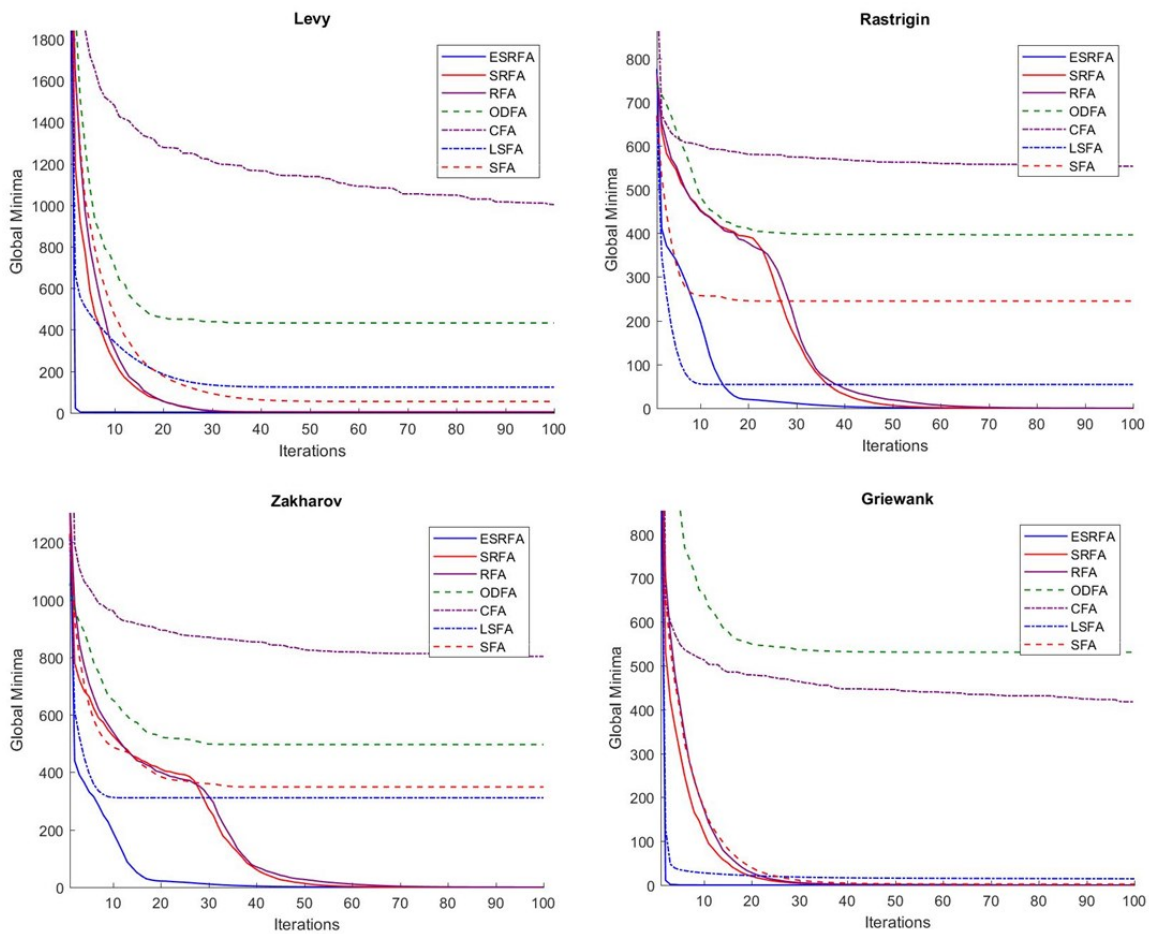


Figure 7 Convergence curves of the proposed models and other advanced FA variants for the Levy, Rastrigin, Zakharov, and Griewank functions

We keep the population size, dimension, and the number of iterations fixed, i.e. the population size=40, dimension=50 and the number of iterations=100, in this experiment. Note that other population and dimension settings have also been employed, which achieve similar convergence performances. The convergence curves over 30 runs for four standard functions (i.e. Levy, Rastrigin, Zakharov and Griewank) are shown in Figure 6. The results of SA are not included, since it takes a larger number of iterations to converge, and extends beyond the range of the plots in Figure 6 for most cases. Overall, ESRFA, SRFA and RFA achieve the best convergence for four functions of artificial landscapes in most cases.

We have compared RFA, SRFA and ESRFA with other state-of-the-art FA variants. Figure 7 shows the convergence curves of 30 runs using the Levy, Rastrigin, Zakharov, and Griewank functions. Our proposed models incorporate diverse attractiveness, repulsive and scattering behaviours to accelerate the search process and achieve fast convergence, in comparison with other FA variants. Since ODFA employs a dimensional-based approach for updating the global best solution in each dimension and LSFA embeds SA to further improve the global best solution in each iteration, both models have a high computational cost and require a large number of iterations to converge. A similar observation can be obtained for other test functions, as shown in Figure 7.

As indicated in Figures 6-7, the three proposed models show significant convergence improvements against other methods owing to the proposed attractiveness, repulsive force and scattering strategies. In RFA, the repulsive force strategy enables brighter fireflies in the neighbourhood to reach optimal regions in fewer iterations in comparison with those in the original FA model. SRFA shows a slight convergence improvement in most cases in comparison with RFA, owing to the scattering mechanism, which diverts a small portion of population congested in a converging location to a random search space to overcome stagnation. ESRFA outperforms SRFA, RFA and state-of-the-art FA variants for all test functions, owing to the proposed exploitation and exploration search mechanisms and the interaction with the neighbouring historical best experiences.

4.2.3 Diverse Dimensions

In the second experiment, we compare different methods by increasing the dimension exponentially, i.e. 2^i where $i=1$ to 8, while fixing the population size to 30 individuals and the number of iterations to 100. Specifically, a total of 30 runs are conducted using the standard benchmark functions in Table 1, with the dimension increasing from 2 to 256.

Figures 8 and 9 show the mean minima of different search methods and FA variants, respectively, in comparison with SRFA and RFA. As shown in Figure 8, the final outcomes of all other search methods tend to drift further away from the global minima (i.e. zero) with increasing dimension of the test function. As the dimension increases, it becomes more challenging for the compared search methods to find the global optima, and their performances deteriorate drastically. However, increasing the dimension has little impact on the final outcomes of RFA and SRFA, as shown in Figure 8. A similar observation is also depicted in Figure 9. All other FA variants (i.e. CFA [35], LSFA [34], ODFA [3] and SFA [34]) tend to be affected significantly by the increase of dimension, whereas RFA and SRFA show great flexibility and outperform other FA variants. The RFA and SRFA results are very close to the global minima, which nearly overlap with the x axis for all experiments. A similar observation applies to ESRFA, in which its results nearly overlap with the x axis for all the test functions. ESRFA also outperforms SRFA and RFA for all the test cases.

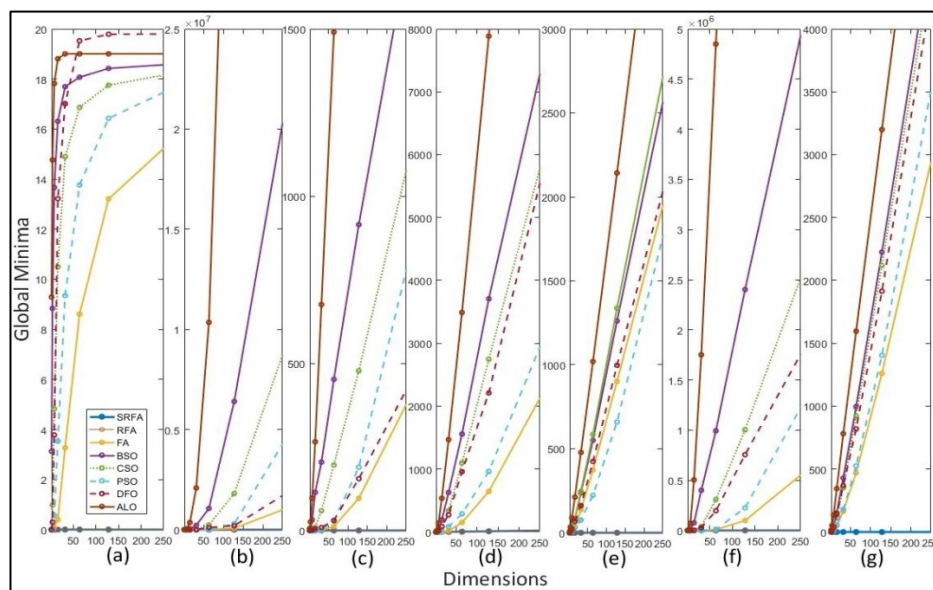


Figure 8 The global minima results for the proposed RFA and SRFA models and other methods subject to an increment of dimension. The x axis indicates the exponential increase of dimension, i.e., 2^i ($i=1$ to 8), while

the y axis shows the global minima of (a) Ackley, (b) Dixon-Price, (c) Griewank, (d) Levy, (e) Rosenbrock, (f) Rastrigin and (g) Zakharov, respectively

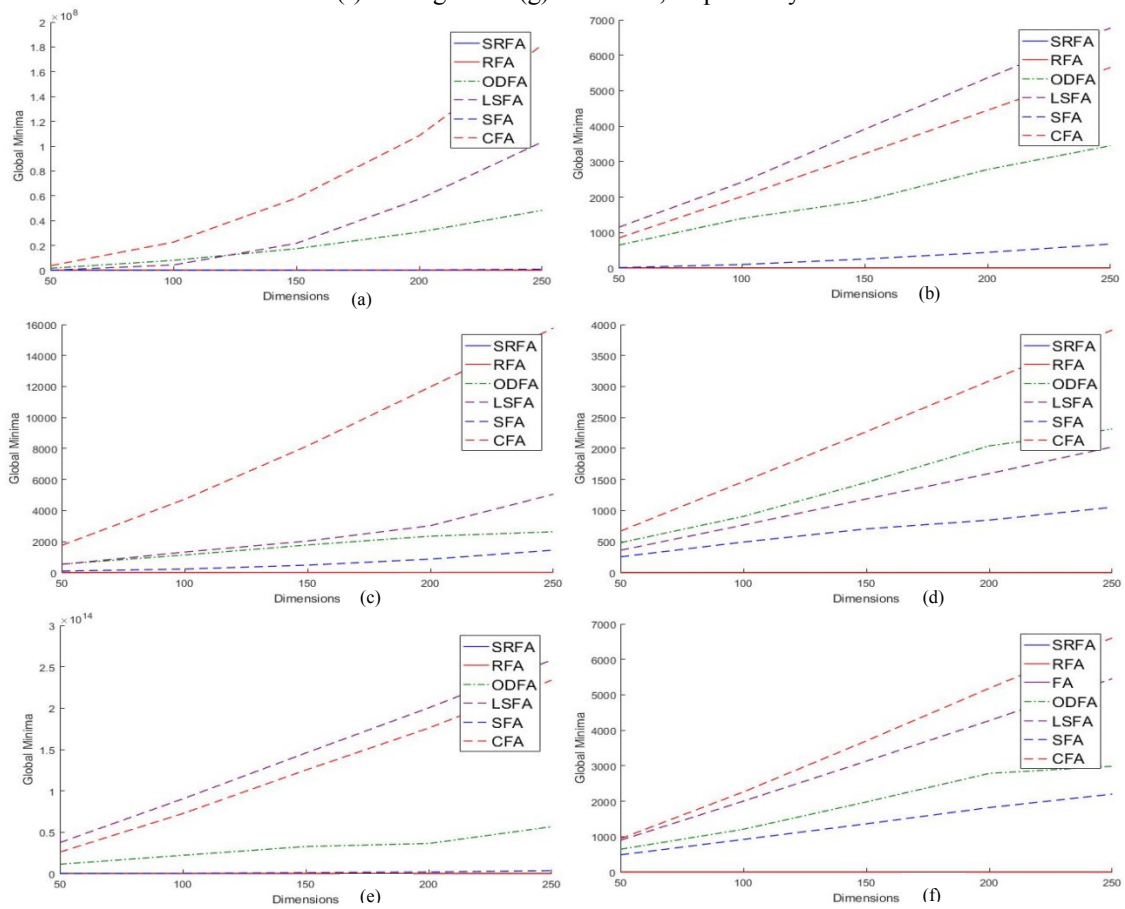


Figure 9 The global minima results for the proposed RFA and SRFA models and other FA variants subject to an increment of dimension. The x axis indicates the exponential increase of dimension, i.e., 2^i ($i=1$ to 8), while the y axis shows the global minima of (a) Dixon-Price, (b) Griewank, (c) Levy, (d) Rastrigin, (e) Rosenbrock, and (f) Zakharov, respectively

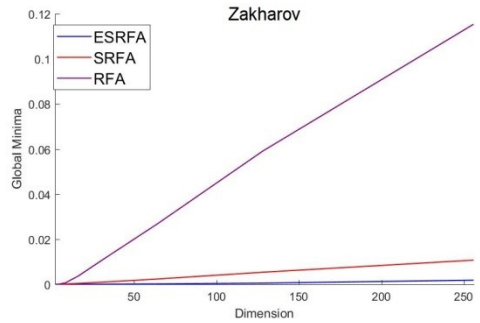
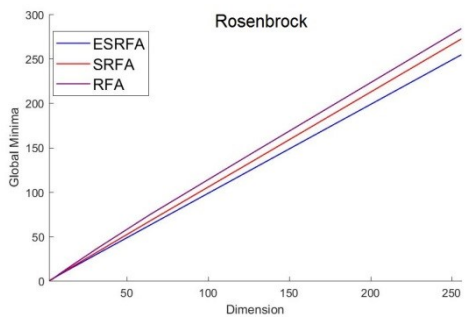
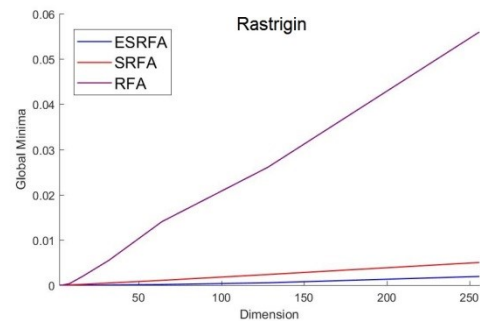
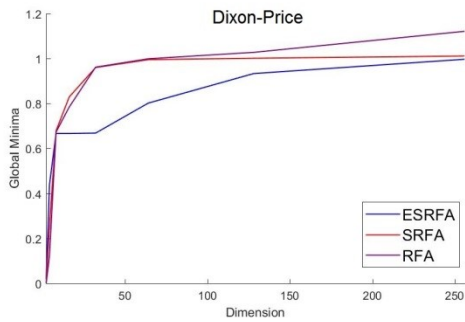


Figure 10 The global minima results for the proposed RFA, SRFA and ESRFA models subject to an increment of dimension. The x axis indicates the exponential increase in dimension, i.e., 2^i ($i=1$ to 8), and the y axis shows the global minima of Dixon-Price, Rastrigin, Rosenbrock, and Zakharov, respectively

Figure 10 illustrates the performance comparison between the proposed ESRFA, SRFA and RFA models subject to the increment of dimension. To better present the results of the proposed models with respect to the test function dimension, the results are plotted by scaling the y axis. Specifically, owing to the cooperation between the attractiveness, repulsive force and the scattering movements, ESRFA and SRFA show a better capability of escaping from local optima in comparison with RFA, therefore yielding the best global minima. Furthermore, because of the embedded exploitation and exploration mechanisms, ESRFA outperforms SRFA for most test cases in all dimensional settings.

4.2.4 Population Size

We have evaluated all methods with diverse population settings subject to a fixed problem dimension of 20 over 30 test runs. Figures 11 and 12 indicate the mean global minima for the standard test functions with the population size ranging from 10 to 100, for SRFA and RFA in comparison with state-of-the-art FA variants and other related methods, respectively.

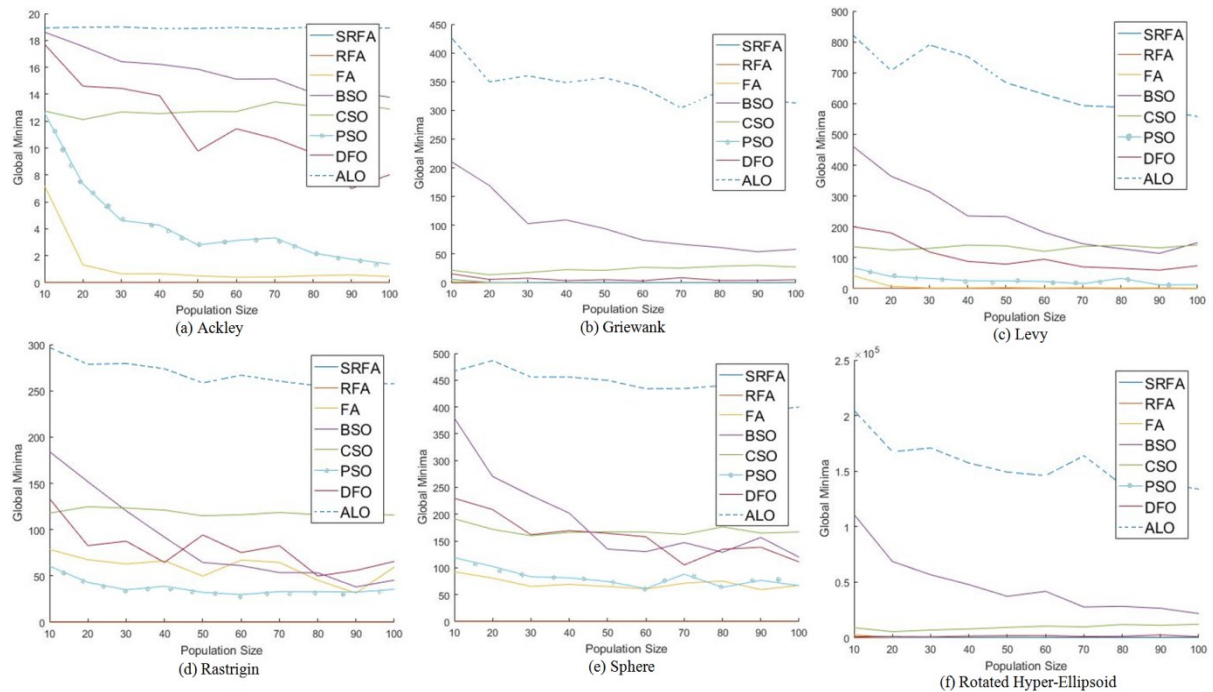


Figure 11 The global minima results for the proposed RFA and SRFA models and other methods subject to increasing population size of (a) Ackley, (b) Griewank, (c) Levy, (d) Rastrigin, (e) Sphere and (f) Rotated Hyper-Ellipsoid, respectively

As shown in Figures 11-12, increasing the population size from 10 to 100 improves the performances of most of the search methods and FA variants, owing to the increase of swarm diversity. On the contrary, the increase of the population size shows very little impact on the performances of RFA and SRFA, whose results remain very close to global minima (i.e. 0), and outperform those of all other methods and FA variants. Moreover, as indicated in the scale-up comparison among ESRFA, SRFA and RFA in Figure 13, the increase of the population size shows very little impact on the ESRFA performances as well, in which the results remain very close to global minima. Obviously, ESRFA outperforms SRFA and RFA. A similar observation can also be obtained for other standard test functions.

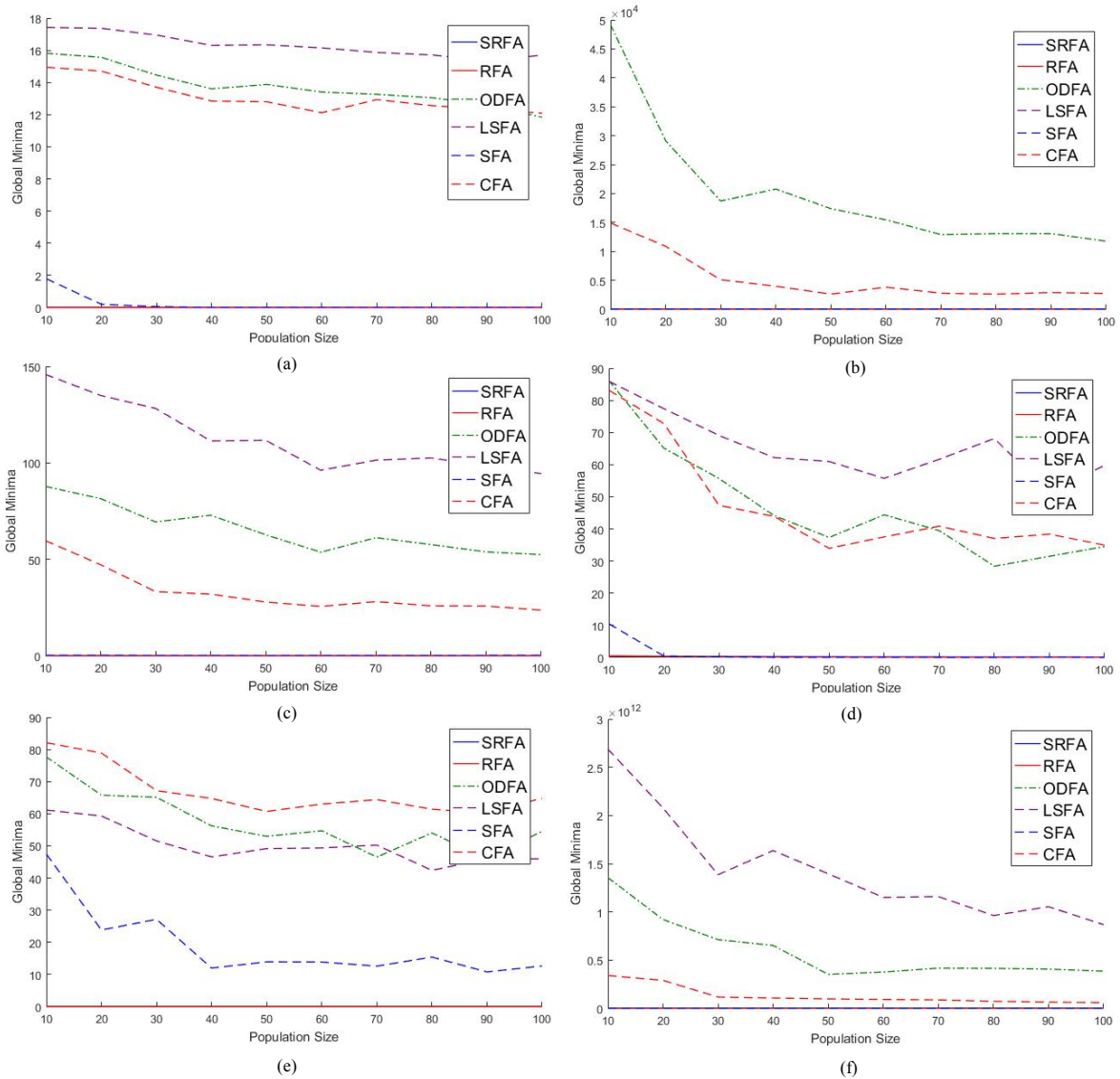


Figure 12 The global minima results for the proposed RFA and SRFA models and other FA variants subject to increasing population size of (a) Ackley, (b) Dixon-Price, (c) Griewank, (d) Levy, (e) Rastrigin, and (f) Rosenbrock, respectively

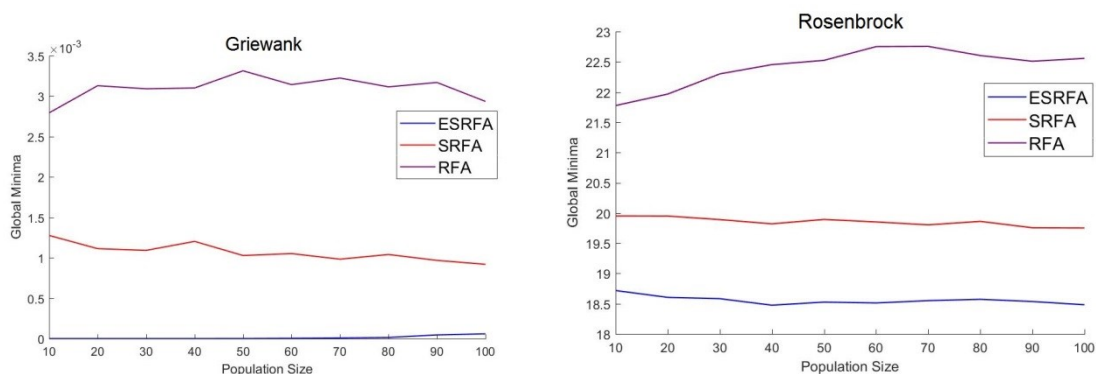


Figure 13 The global minima results for the proposed RFA, SRFA and ESRFA models subject to increasing population size of Griewank and Rosenbrock, respectively

4.2.5 Evaluation Using the Standard Benchmark Functions

To further ascertain efficiency of the proposed models, a number of FA variants (ODFA [3], SFA [34], LSFA [34] and CFA [35]) as well as other search methods have been implemented for comparison using ten

function with dimension 10. The following settings are applied, i.e. population size=20 and number of iterations=500. Table 5 shows the detailed evaluation results.

ESRFA, SRFA and RFA outperform ODFA, CFA, LSFA, SFA, and other search methods for nearly all CEC2014 test functions. ESRFA also achieves better global optima than those of SRFA and RFA in most test cases. Specifically, ESRFA outperforms all advanced FA variants and other methods for F1-F22 and F28-F30, while SRFA achieves the best performances for F1-F22, and F27-F30, with RFA outperforming the baselines for F1-F22, and F29-F30. CSO performs the best for F23, F25, and F26, while SFA achieves the best result for F24.

The statistical Wilcoxon rank sum test has been conducted to indicate the significance level of ESRFA results. The detailed results are provided in Table 6. ESRFA shows statistically significant improvement over SRFA, RFA, advanced FA variants and other search methods for most of the test cases, except for F26, where it shows similar result distributions with those of ODFA, BSO, and ALO as well as for F27, where it has the same median as that of BSO.

Among the existing FA variants, SFA and CFA show impressive performances, while FA, CSO, and PSO achieve the best global minima among all other classical methods. Furthermore, the proposed ESRFA, SRFA and RFA models demonstrate better computational efficiency in comparison with those of ODFA [3], SFA [34] and LSFA [34]. This is owing to the fact that ODFA includes a dimensional-based method for the current global best updating procedure in each iteration. It needs to traverse through the whole population in each dimension, and has a computational cost of $n \times d \times \text{maximum_iteration}$, where n is the population size and d is the dimension. The high computational cost of SFA and LSFA is attributed by the slow convergence rate of SA in improving the current best solution in each generation.

To further evaluate the proposed models, we have increased the dimension of CEC2014 test functions to 30 and 50. Other key parameter settings are, population size=20, number of iterations=500, and number of trials=30. Tables 7-8 (in Appendix) show the detailed evaluation results for dimensions 30 and 50, respectively. For both dimensions, the proposed ESRFA model outperforms the existing FA variants and other methods for nearly all test functions, except for F25-F27 in dimension 30 and F26-28 in dimension 50. For both dimensions, SRFA outperforms all the baseline methods for F1, F3-F25 and F27-F30 with RFA outperforming all other methods for F1, F3-F11, F13-F25, and F29-F30. In addition, CFA and SFA perform the best among the existing FA variants, while FA, CSO, and PSO achieve the best global minima among all other search methods.

The Wilcoxon rank sum test has been conducted to indicate the significance level of ESRFA performance for both dimensions. The detailed test results are shown in Tables 9-10 (in Appendix). For dimension 30, as indicated in Table 9, ESRFA outperforms SRFA, RFA, all existing FA variants and other search methods for nearly all test functions. The exceptions are, (i) F26, where ESRFA shows similar distributions with those of FA, ODFA, SFA, BSO and DFO, (ii) F25 and F27, where ESRFA shows similar distributions with those of LSFA and BSO, respectively, and (iii) F24, where ESRFA and SRFA show similar result distributions. A similar observation is also obtained for the statistical analysis for dimension 50. As illustrated in Table 10, ESRFA shows statistically significant improvement over SRFA, RFA and all other methods for nearly all test cases, except for F25-F26, where it shows similar result distributions with those of ODFA.

4.2.7 Discussion

A discussion on theoretical comparison between the proposed models and other related methods is as follows. ODFA, SFA, LSFA, and CFA employ single search mechanism, i.e. either customized or original attractiveness movement, for position updating. As an instance, CFA [35] employs a chaotic accelerated attraction operation to move fireflies with lower light intensities toward brighter ones. If the chaotic attractiveness movement is not able to find fitter solutions, there is no alternative search strategy embedded in CFA to drive the search out of local optima. Similarly, SFA [34] and LSFA [34] purely rely on the original attraction movements of FA and LFA, respectively, for finding the global best solution. Although SA is used to further improve the best solution identified by FA or LFA, the operation of SA largely relies on the random walk to generate new solutions, and has very weak local exploitation capabilities. Therefore, SA converges very slowly, and the achievement of global optimality is at the cost of a large number of function evaluations [1, 7]. This leads to a comparatively low computational efficiency as compared with the proposed models. ODFA [3] employs a dimensional-based method to update the global best solution in each iteration. This search mechanism requires a high computational cost, as it incurs a large number of function evaluations. Furthermore, it relies on single global best solution in the position updating procedure of each

firefly without considering multiple neighbouring promising solutions. Since the search process is guided by only single best solution, similar to the characteristics of PSO, it is susceptible to the local minima problem. Their search strategies also show limitations in dealing with complex multimodal and composition optimization problems.

ESRFA, SRFA and RFA employ multiple position updating mechanisms, and incorporate diversified attractiveness, repulsive, and scattering strategies as well as neighbourhood historical attraction to overcome the limitations of the attractiveness action of the original FA model. RFA uses the repulsive force strategy and the attractiveness movement to work alternatively to identify optimal search regions effectively. SRFA integrates both the repulsive force action and the scattering movement to increase local exploitation and global exploration to mitigate premature convergence of the original FA model. ESRFA not only employs hawk-moth motivated exploitation and escaping coefficients to diversify the search operations, but also learns from other individuals' historical best experiences to accelerate convergence.

In FA, a random exploration is initiated when there is no better solution in the neighbourhood. If this random walk action is unable to generate a fitter offspring, this may lead to local optimum. To avoid stagnation, the repulsive force action in RFA pushes the current firefly out of gloomy local search regions to reach better positions. Moreover, the scattering mechanism and the repulsive force action work cooperatively to drive the search out of local optima in SRFA. When the repulsive force behaviour fails to produce a better offspring, the scattering mechanism re-distributes a proportion of weak neighbouring solutions to other comparatively less exploited search regions, therefore increasing search diversity and avoiding local optima. On the contrary, when the scattering movement is unable to produce better solutions, the repulsive force behaviour drives the fireflies out of unpromising search regions effectively to avoid premature convergence.

ESRFA further enhances the search actions of SRFA by integrating mechanisms that comprise exploitation-based attraction, exploration-based evasion, and interaction with other individuals' historical personal best experiences. These strategies enable a refined examination of promising neighbouring regions, expelling from less optimal regions efficiently and learning from neighbouring best past experiences. They cooperate with each other to diversify the search process, overcome local optima traps, and accelerate convergence. Overall, the newly proposed mechanisms enable RFA, SRFA and ESRFA to outperform original FA and its variants in the literature, significantly, in undertaking diverse optimization problems.

Based on the ten standard functions as well as the shifted, rotated, hybrid and composite test functions from CEC2014, FA, CSO, PSO and DFO outperform all other search methods consistently. PSO utilizes personal and global best solutions to guide the search process while CSO employs switching factor based local and Levy flight based global search strategies to lead the search towards optimal regions. DFO employs static and dynamic swarming behaviours of dragonflies by modelling their social behaviours in searching for food and avoiding enemies. It possesses sufficient search diversity than other search methods such as SA and ALO for finding global optima. FA enables each search agent to follow multiple optimal neighbouring solutions and avoid local optima. It shows great efficiency in solving diverse optimization problems, especially shifted and rotated multimodal, hybrid and composition test functions. ESRFA, SRFA and RFA not only preserve but also extend the multimodal search capability of the original FA model by using the proposed attractiveness, repulsive force and scattering mechanisms as well as neighbourhood historical attraction. These characteristics account for the superior performance of ESRFA, SRFA and RFA in comparison with all other methods.

Table 5 Evaluation results for the CEC2014 benchmark functions with dimension=10

		ESRFA	SRFA	RFA	ODFA [3]	SFA [34]	LSFA [34]	CFA [35]	FA	SA	BSO	CSO	PSO	DFO	ALO
1	Mean	1.00E+02	1.01E+02	1.02E+02	8.59E+05	1.59E+05	2.09E+07	6.46E+02	3.08E+03	6.35E+07	1.12E+08	7.34E+03	8.28E+04	7.88E+06	4.48E+08
	Min	1.00E+02	1.00E+02	1.01E+02	2.72E+04	1.03E+02	2.45E+06	2.25E+02	2.10E+02	2.39E+07	1.89E+07	1.60E+03	1.77E+03	9.37E+05	57440287
	Max	1.01E+02	1.02E+02	1.05E+02	5.16E+06	3.67E+06	4.84E+07	2.29E+03	1.41E+04	1.35E+08	3.16E+08	1.33E+04	3.77E+05	2.12E+07	1.04E+09
	Std	1.11E-01	4.53E-01	9.44E-01	1.01E+06	6.73E+05	9.27E+06	5.03E+02	3.51E+03	3.06E+07	8.03E+07	3.68E+03	1.02E+05	5.73E+06	3.34E+08
2	Mean	2.00E+02	2.72E+02	3.27E+02	3.21E+03	2.58E+06	3.52E+09	4.33E+02	1.88E+03	7.98E+09	7.37E+09	2.40E+06	2.41E+03	4.29E+07	1.69E+10
	Min	2.00E+02	2.27E+02	2.12E+02	3.92E+02	2.00E+02	1.58E+09	2.00E+02	2.00E+02	4.03E+09	1.28E+09	9.38E+03	2.00E+02	8.78E+03	5.47E+09
	Max	2.00E+02	3.15E+02	4.38E+02	7.32E+03	6.31E+07	4.96E+09	8.47E+02	9.22E+03	1.25E+10	1.61E+10	3.10E+07	1.10E+04	1.57E+08	2.98E+10
	Std	1.08E-02	2.21E+01	5.52E+01	2.27E+03	1.15E+07	7.68E+08	2.46E+02	2.44E+03	1.88E+09	3.54E+09	6.76E+06	2.81E+03	4.64E+07	5.45E+09
3	Mean	3.00E+02	3.00E+02	3.00E+02	1.20E+04	5.58E+03	2.58E+04	5.77E+02	1.88E+03	7.61E+04	1.27E+05	3.01E+02	3.36E+03	2.92E+04	25806574
	Min	3.00E+02	3.00E+02	3.00E+02	2.63E+03	3.00E+02	1.24E+04	3.00E+02	3.01E+02	2.23E+04	1.27E+04	3.00E+02	3.35E+02	4.02E+03	35737.12
	Max	3.00E+02	3.00E+02	3.00E+02	2.15E+04	3.34E+04	3.54E+04	2.45E+03	1.45E+04	3.81E+05	6.51E+05	3.02E+02	1.29E+04	5.84E+04	1.92E+08
	Std	1.22E-03	2.83E-03	1.90E-03	4.42E+03	9.23E+03	6.39E+03	4.90E+02	2.78E+03	7.85E+04	1.63E+05	4.35E-01	3.02E+03	1.44E+04	49515116
4	Mean	4.00E+02	4.00E+02	4.00E+02	4.08E+02	4.00E+02	6.09E+02	4.00E+02	4.03E+02	8.93E+02	1.44E+03	4.02E+02	4.05E+02	4.65E+02	3110.331
	Min	4.00E+02	4.00E+02	4.00E+02	4.07E+02	4.00E+02	5.18E+02	4.00E+02	4.00E+02	5.85E+02	5.76E+02	4.00E+02	4.00E+02	4.05E+02	1110.231
	Max	4.00E+02	4.00E+02	4.00E+02	4.10E+02	4.07E+02	7.13E+02	4.04E+02	4.06E+02	1.27E+03	3.69E+03	4.07E+02	4.67E+02	6.91E+02	5984.321
	Std	5.35E-07	5.04E-06	9.63E-06	8.01E-01	1.19E+00	4.66E+01	7.32E-01	1.49E+00	1.79E+02	6.73E+02	1.75E+00	1.21E+01	7.85E+01	1219.741
5	Mean	5.00E+02	5.00E+02	5.00E+02	5.17E+02	5.02E+02	5.20E+02	5.05E+02	5.20E+02	5.21E+02	5.20E+02	5.20E+02	5.17E+02	5.20E+02	521.1409

29	Max	3.82E+03	3.00E+03	3.56E+03	3.36E+03	4.12E+03	3.25E+03	4.70E+03	3.62E+03	4.34E+03	5.13E+03	3.19E+03	3.88E+03	3.96E+03	5577.285
	Std	1.50E+02	2.04E-03	8.51E+01	5.59E+01	2.91E+02	1.26E+01	3.82E+02	1.08E+02	2.26E+02	4.86E+02	9.09E+00	2.34E+02	2.45E+02	507.0566
	Mean	3.10E+03	3.10E+03	3.10E+03	3.69E+03	3.67E+03	9.13E+03	3.15E+03	3.26E+03	2.83E+05	4.90E+04	3.12E+03	3.50E+03	1.61E+05	8100768
	Min	3.10E+03	3.10E+03	3.10E+03	3.24E+03	3.10E+03	3.62E+03	3.10E+03	3.13E+03	2.19E+04	3.74E+03	3.10E+03	3.21E+03	3.35E+03	394122
30	Max	3.10E+03	3.10E+03	3.10E+03	5.23E+03	1.04E+04	2.39E+04	3.26E+04	3.81E+03	7.79E+05	4.55E+05	3.13E+03	5.49E+03	4.62E+06	22843511
	Std	6.30E-03	9.63E-04	1.96E-03	4.44E+02	1.63E+03	4.83E+03	4.04E+01	1.22E+02	2.20E+05	9.08E+04	5.55E+00	5.31E+02	8.43E+05	6038376
	Mean	3.20E+03	3.20E+03	3.20E+03	3.59E+03	3.29E+03	5.18E+03	3.35E+03	3.68E+03	2.32E+04	7.38E+04	3.32E+03	3.77E+03	4.34E+03	870179.2
	Min	3.20E+03	3.20E+03	3.20E+03	3.32E+03	3.20E+03	4.23E+03	3.21E+03	3.34E+03	5.03E+03	3.59E+03	3.26E+03	3.31E+03	3.38E+03	64763.05
	Max	3.20E+03	3.20E+03	3.20E+03	3.97E+03	4.18E+03	7.16E+03	3.78E+03	4.33E+03	8.83E+04	1.07E+06	3.37E+03	4.33E+03	6.77E+03	4405391
	Std	4.99E-02	2.60E-02	1.91E-02	1.86E+02	1.99E+02	7.60E+02	1.30E+02	2.54E+02	1.72E+04	2.02E+05	2.59E+01	2.80E+02	7.18E+02	861683.7

Table 6 The p -values of the Wilcoxon rank sum test for the CEC2014 benchmark functions with dimension=10

F _n	SRFA	RFA	FA	ODFA	CFA	LSFA	SFA	SA	BSO	CSO	PSO	DFO	ALO
1	7.39E-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
2	3.02E-11	3.02E-11	3.02E-11	6.07E-11	3.02E-11	1.31E-08	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	4.20E-10	3.02E-11
3	7.60E-07	4.69E-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	3.02E-11	3.02E-11
4	3.02E-11	3.02E-11	3.02E-11	1.70E-08	3.02E-11	1.17E-02	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
5	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
6	4.42E-06	1.49E-06	3.16E-10	1.87E-07	3.02E-11	2.37E-10	4.44E-07	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.01E-11
7	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
8	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
9	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
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	3.02E-11	3.02E-11	3.02E-11	3.02E-11
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	3.02E-11
12	3.02E-11	3.02E-11	3.02E-11	3.47E-10	3.02E-11	1.21E-10	3.34E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
13	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
14	6.07E-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
15	1.73E-11	1.73E-11	1.73E-11	9.22E-11	1.73E-11	1.73E-11	1.73E-11	1.73E-11	1.73E-11	1.73E-11	1.73E-11	1.73E-11	1.73E-11
16	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
17	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
18	5.19E-07	3.26E-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	3.02E-11	3.02E-11
19	3.02E-11	3.02E-11	3.02E-11	1.61E-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
20	7.04E-07	1.47E-07	3.02E-11	8.15E-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
21	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
22	5.53E-08	7.12E-09	3.02E-11	7.12E-09	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
23	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	7.96E-03	3.02E-11	3.02E-11	3.02E-11
24	4.44E-07	2.15E-10	3.02E-11	3.02E-11	3.02E-11	2.71E-02	1.01E-08	3.02E-11	3.02E-11	3.02E-11	1.11E-04	6.77E-05	3.02E-11
25	3.02E-11	3.02E-11	3.20E-09	5.49E-11	6.05E-07	3.02E-11	1.87E-07	3.02E-11	3.02E-11	3.02E-11	9.06E-08	6.20E-04	3.01E-11
26	4.69E-08	4.08E-11	5.57E-10	3.33E-01	3.47E-10	1.11E-06	3.02E-11	6.55E-04	8.53E-01	3.02E-11	9.26E-09	2.60E-08	1.71E-01
27	5.57E-10	3.50E-09	5.57E-10	8.10E-10	1.01E-08	7.09E-08	5.57E-10	6.05E-07	1.76E-01	5.57E-10	8.48E-09	1.01E-08	2.27E-06
28	5.57E-10	5.57E-10	7.77E-09	4.62E-10	5.57E-10	1.46E-10	5.57E-10	1.21E-10	1.61E-10	5.57E-10	4.62E-10	3.47E-10	3.02E-11
29	8.20E-07	6.53E-07	3.02E-11	1.46E-10	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
30	2.20E-07	4.31E-08	3.02E-11	4.08E-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

4.2.8 Comparison with an Exact Algorithm Using the Traveling Salesman Problem

To further evaluate the proposed three models, a performance comparison with an exact algorithm, i.e. Integer Linear Programming (ILP), for solving the TSP, is carried out. TSP is an NP-hard problem. It aims to solve the following task – given a list of cities, identify the shortest possible path that visits each city exactly once and returns to the original city. According to [64], among various exact methods, ILP is very efficient in solving TSP. Therefore, it is used in this comparison study. We use the ILP algorithm provided by the MATLAB Optimization Toolbox. Dedicated equality and inequality constraints are employed, with the objective of minimizing the overall cost (fitness) function of a tour in TSP.

The ILP algorithm firstly employs two types of equality constraints to initialize the problem, i.e. (1) there must be n trips, where n is the number of cities and (2) each city is visited exactly once and left exactly once (one for arrival and one for departure). Then, the ILP algorithm with the ‘round-diving’ heuristics is used to identify the initial solution. Subsequently, sub-tour elimination inequality constraints are used to eliminate the sub-tours in the current solution. This procedure iterates by calling the optimizer repeatedly until the current solution has only one sub-tour.

We formulate the TSP for our proposed models as follows. Unlike the ILP algorithm, which employs explicit guidance such as equality and inequality constraints to lead the search process, the proposed models treat TSP as a black-box optimization problem. Therefore, the search process is controlled by the maximum number of iterations. We employ the number of cities as the problem dimension. The following parameter settings are used in the experiments, i.e. trials=30, population size=100, and the maximum number of iterations=3,000 for 10-50 dimensions, and 10,000 for 60-100 and 150 dimensions, respectively. The experiments using such an iteration setting yield the best trade-off between performance and computational efficiency for all dimensions, although for some dimensions (e.g. 10-40), using a lower number (e.g. 200-500) of iterations achieves similar performances.

The cost (i.e. the length) of the tour in an TSP is minimized by a fitness function. For each specific dimension, the three proposed models employ the same number of function evaluations (i.e. population size \times maximum number of iterations). Different numbers of cities (problem dimensions), i.e. 10-100 and 150,

are evaluated, and the mean results over 30 runs are shown in Table 11. For all experiments, we use a Personal Computer with Intel 4th Generation Core i5 processors with 3.5 GHz clock speed and 32GB dual channel DDR3 memory. The average time and the corresponding average iteration numbers for each algorithm to reach the most optimal solutions for the first time over 30 runs are provided in Table 12. Figure 14 illustrates the convergence curves of the proposed models over 30 runs. Note that no parameter tuning and/or customized setting are applied to the proposed models in solving this TSP challenge.

Table 11 The average performance of the proposed models over 30 runs in comparison with the optimal results of ILP using 10-100 and 150 cities

		ILP	ESRFA	SRFA	RFA
10	min	1.672116	1.672116	1.672116	1.672116
	mean	-	1.672116	1.672116	1.672116
20	min	2.4329	2.432885	2.741636	2.544178
	mean	-	2.579692	3.115744	3.219473
30	min	3.580475	3.580475	4.632441	4.444965
	mean	-	3.820052	5.876807	5.642440
40	min	4.11964	4.707957	7.471717	7.194112
	mean	-	6.183399	7.778622	8.655341
50	min	4.356541	6.061257	10.13456	9.554801
	mean	-	6.670493	11.03801	11.39157
60	min	4.746534	7.045064	10.94685	13.27867
	mean	-	8.726815	13.29579	14.28897
70	min	5.036186	8.529599	14.9446	15.69023
	mean	-	10.88965	16.55569	16.61345
80	min	5.280799	9.463684	18.13303	18.64085
	mean	-	13.07872	19.90855	19.79657
90	min	5.447962	15.29622	22.56021	21.63666
	mean	-	15.67242	23.6314	22.90575
100	min	5.802158	13.35048	23.52844	24.08626
	mean	-	15.09897	24.63503	25.18497
150	min	7.144701	26.91510	40.48137	39.51841
	mean	-	32.65665	45.27255	42.02122

Table 12 The computational cost (in seconds) of ILP and the average time and corresponding average iteration numbers of the proposed algorithms when they first achieve the most optimal solutions

	ILP (time)	ESRFA (time & iter)	SRFA (time & iter)	RFA (time & iter)
10	1.60	0.752024 (iter - 229)	0.916855 (iter - 603)	0.499047 (iter - 649)
20	1.95	2.468966 (iter - 840)	1.140217 (iter - 1019)	0.993385 (iter - 1802)
30	2.27	2.209506 (iter - 1204)	1.546964 (iter - 2378)	0.843462 (iter - 2397)
40	2.31	2.834179 (iter - 1511)	1.215104 (iter - 1924)	0.984352 (iter - 2927)
50	2.97	7.987160 (iter - 1637)	4.817862 (iter - 2814)	2.611842 (iter - 2842)
60	3.72	7.910138 (iter - 5964)	3.968862 (iter - 9501)	2.344511 (iter - 9302)
70	3.95	12.44020 (iter - 7705)	4.179294 (iter - 8288)	2.961191 (iter - 9720)
80	6.10	12.66899 (iter - 7139)	4.316745 (iter - 8823)	2.743825 (iter - 8683)
90	6.74	11.76082 (iter - 7012)	4.276014 (iter - 8963)	2.685908 (iter - 8863)
100	7.93	13.69362 (iter - 7752)	4.052976 (iter - 8345)	3.008126 (iter - 9340)
150	30.65	51.61561 (iter - 8677)	10.06821 (iter - 9463)	6.410333 (iter - 9671)

As an exact algorithm, ILP is able to find the exact optimal solution with precision to a given optimization problem, while the proposed models obtain near optimum solutions. As indicated in Table 11, ESRFA is capable of providing competitive approximate optimal solutions as compared with those of ILP in most cases. As an example, for visiting 10-70 cities (10-70 dimensions), the minimum global optima (i.e. the best solutions) obtained by ESRFA are very close to those achieved by ILP. In particular, for dimensions 10-30, ESRFA (even with a very low number of iterations) is able to yield the same optimal solutions as those of ILP.

ESRFA outperforms SRFA and RFA for all dimensions. SRFA performs better than RFA for dimensions 20, 40-70, and 100, while RFA outperforms SRFA for dimensions 30, 80, 90, and 150. As indicated in Figure 14, ESRFA is able to converge faster than SRFA and RFA for all test cases. The computational costs shown in Table 12 for the three proposed algorithms to first reach the most optimal solutions are comparable with those of ILP. Since ESRFA and SRFA contain more strategies than those of RFA, they have a slightly higher computational cost than that of RFA in each iteration. Therefore, RFA has the best overall computational efficiency, although ESRFA and SRFA converge faster.

Unlike ILP or other exact algorithms, the proposed models do not require any explicit guidance, and they treat TSP as a black-box optimization problem. ESRFA achieves similar promising near-optimal solutions for some higher dimensional cases, but with slightly higher computational costs in comparison with those of ILP. Moreover, SRFA and RFA obtain reasonable trade-off between performance and efficiency, and depict better computational efficiency than those of ILP in most cases. Nevertheless, the improvements in

convergence of SRFA and RFA are not as significant as those of ESRFA, with respect to increasing the maximum number of iterations.

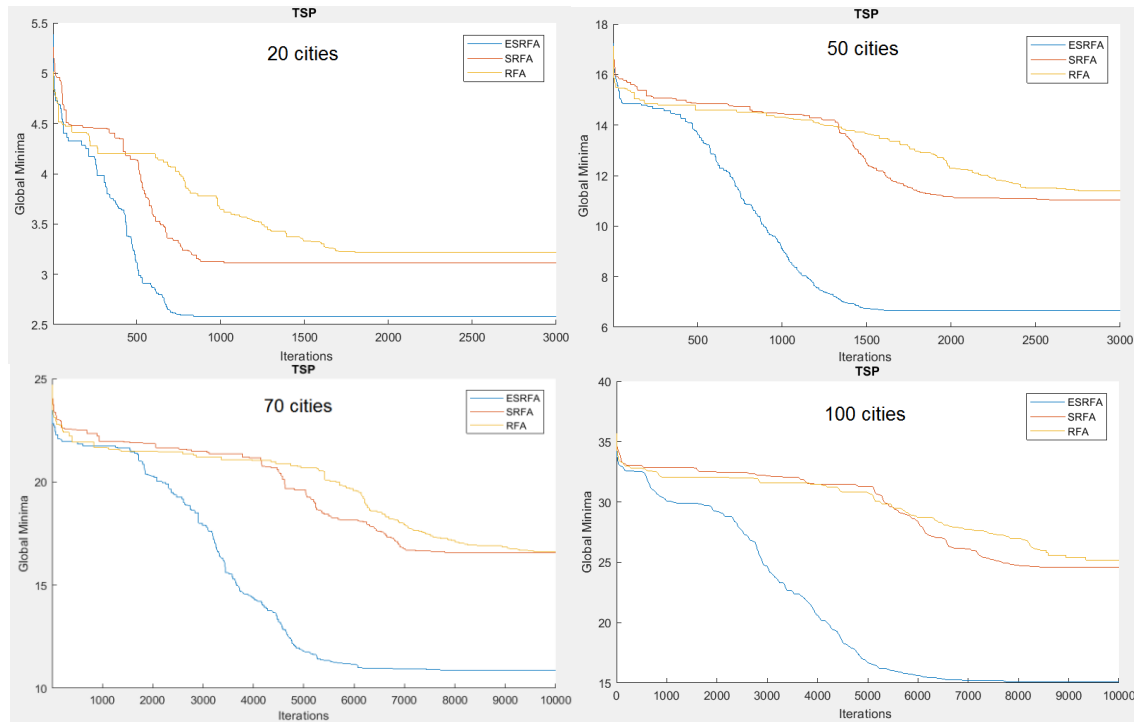


Figure 14 Convergence curves of ESRFA, SRFA, and RFA for solving the TSP with 20, 50, 70 and 100 cities, respectively

In summary, ESRFA provides an efficient approximation of optimal solutions, and yields comparable performance with those of ILP for solving the TSP. It is also more flexible than ILP. It does not require explicit guidance, and treats the TSP as a black-box optimization problem. While ILP is efficient for the class of problems that can be solved in polynomial time, it becomes less flexible and is time-consuming in undertaking large, complex real-world problems (e.g. hardly constrained or multimodal problems), as compared with heuristic methods [64-67]. A hybrid model by embedding the computationally efficient models (such as RFA and SRFA) within ILP to improve the resulting performance could be investigated, e.g. using the proposed models to obtain initial solutions or vice versa, in order to accelerate the search process [64, 65].

5. CONCLUSIONS

We have proposed three improved FA models, i.e., RFA, SRFA and ESRFA, to mitigate premature convergence of the original FA model. RFA attains global optimality with a fast convergence rate by utilizing a repulsive force strategy. SRFA employs both the repulsive force and scattering mechanisms to increase local and global search capabilities. ESRFA uses hawk-moth motivated exploitation and exploration mechanisms and the interaction with neighbouring historical personal best experiences to accelerate convergence. Evaluated with standard and challenging CEC2014 benchmark functions, RFA, SRFA, and ESRFA outperform state-of-the-art FA variants (i.e. ODFa, LSFA, SFA, and CFA), the original FA model, and other search methods (i.e. PSO, SA, BSO, CSO, DFO, ALO), significantly, in terms of accuracy and convergence rate. The proposed alternative search mechanisms account for the significance and superiority of the proposed models in undertaking diverse optimization problems, as ascertained by the comprehensive experimental results along with statistical tests. In particular, ESRFA possesses distinctive attractiveness operations that resemble hawk-moth movements. It also embeds exceptional exploration-driven repulsive actions as well as unique historical neighbouring best experience-based attraction to enhance its performance. These characteristics are absent in original FA and other proposed models. Therefore, ESRFA outperforms SRFA, RFA, advanced FA variants, and the original FA model, significantly in the experimental study conducted in this research.

For future work, chaotic maps (e.g. Gauss and Logistic chaotic maps) and mutation mechanisms could be embedded into the proposed models to enable diverse chaotic mutated scattering and repulsive behaviours, in an attempt to enhance their performances.

In terms of potential applications, the proposed models can be applied to diverse real-life single and multi-objective optimization problems, such as job scheduling [68], design optimization [37], RFID network planning [36], radiation therapy treatment planning [40], optimal parameter identification [39, 41, 42], feature selection [28, 43, 49, 60, 69-71], colour image segmentation [46], image retrieval and classification [42, 44]. As an example, the proposed models can be employed to identify the most significant discriminative features for facial and bodily expression [49, 60], skin cancer [69], heart disease [70], and brain tumour classification [71]. They can also be used in conjunction with clustering algorithms for microscopic image segmentation in blood cancer detection [50, 59]. On the other hand, the proposed models are useful for optimal parameter selection pertaining to diverse classification and regression models, e.g. multi-step-ahead stock price index forecasting [39], lake level prediction [41], region-based image retrieval [72], and hyperspectral image classification [44]. Recently, FA and its variants have gained increasing popularity in deep learning. Therefore, it would be useful to conduct further studies on how the proposed models could be used to generate evolving optimized deep neural networks for machine translation, visual saliency detection, image description generation, and visual question generation [72-75].

ACKNOWLEDGEMENT

This research is supported by European Union (EU) sponsored (Erasmus Mundus) eLINK (Centre of excellence for Learning, Innovation, Networking and Knowledge) project (EU Grant No. 2645).

REFERENCES

- [1] X.S. Yang, Firefly Algorithms for Multimodal Optimization. Stochastic Algorithms: Foundations and Applications. Volume 5792, *Lecture Notes in Computer Science*. 169-178. 2009.
- [2] I. Fister, I. Fister Jr, X.S. Yang, J. Brest, A comprehensive review of firefly algorithms. *Swarm and Evolutionary Computation*, Volume 13, December 2013, Pages 34–46.
- [3] O.P. Verma, D. Aggarwal, T. Patodi, Opposition and Dimensional based modified firefly algorithm, *Expert Systems with Applications*. 44 (2016) 168–176.
- [4] A. Abdullah, S. Deris, M.S. Mohamad, S.Z.M. Hashim, A New Hybrid Firefly Algorithm for Complex and Nonlinear Problem. S. Omatu et al. (Eds.): *Distributed Computing and Artificial Intelligence*, Volume 151, Advances in Intelligent and Soft Computing. 673-680. 2012.
- [5] J. J. Liang, B-Y. Qu, P. N. Suganthan, Problem Definitions and Evaluation Criteria for the CEC 2014 Special Session and Competition on Single Objective Real-Parameter Numerical Optimization", Technical Report 201311, Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou, China and Technical Report, Nanyang Technological University, Singapore, December 2013.
- [6] J. Kennedy, R. Eberhart, Particle swarm optimization. In *Proc. IEEE Int. Conf. Neural Network*, Volume 4, 1942–1948. 1995.
- [7] S. Kirkpatrick, C.D. Gelatt Jr, M.P. Vecchi, Optimization by Simulated Annealing. *Science*. Vol. 220, Issue 4598, 671-680. 1983.
- [8] X.S. Yang, A New Metaheuristic Bat-Inspired Algorithm. Nature Inspired Cooperative Strategies for Optimization (NICSO 2010), Volume 284, *Studies in Computational Intelligence*. 65-74. 2010.
- [9] X.S. Yang, A.H. Gandomi, Bat algorithm: a novel approach for global engineering optimization, *Engineering Computations*, Vol. 29 Issue 5, 464 – 483. 2012.
- [10] X.S. Yang, S. Deb, Cuckoo search via Lévy flights. In *Proceedings of world congress on nature & biologically inspired computing (NaBIC 2009)*. USA: IEEE Publications. 2009: 210-214.
- [11] S. Mirjalili, Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems, *Neural Computing and Applications*. Volume 27, Issue 4, 1053–1073. 2016.
- [12] S. Mirjalili, The Ant Lion Optimizer. *Advances in Engineering Software*, Volume 83, 80–98. 2015.

- [13] L. Zhang, W. Srisukkhom, S.C. Neoh, C.P. Lim, D. Pandit, Classifier ensemble reduction using a modified firefly algorithm: An empirical evaluation. *Expert Systems with Applications*. 93 (2018) 395–422.
- [14] L. Zhang, K. Mistry, C.P. Lim, S.C. Neoh, Feature Selection Using Firefly Optimization for Classification and Regression Models. *Decision Support Systems*. 106 (2018) 64–85.
- [15] X.S. Yang, Firefly algorithm, levy Flights and global optimization, *Research and Development in Intelligent Systems*. 26 (2010) 209–218.
- [16] X. S. Yang, Chaos-enhanced firefly algorithm with automatic parameter tuning. *International Journal of Swarm Intelligence Research (IJSIR)*. 2(4), 1-11, 2011.
- [17] T. Apostolopoulos, A. Vlachos, Application of the Firefly Algorithm for Solving the Economic Emissions Load Dispatch Problem. *International Journal of Combinatorics*. Volume 2011, Article ID 523806, 1-23. 2011.
- [18] X.S. Yang, S.S.S. Hosseini, A.H. Gandomi, Firefly Algorithm for solving non-convex economic dispatch problems with valve loading effect. *Applied Soft Computing*. 12 (2012) 1180–1186.
- [19] M.H. Horng, Vector quantization using the firefly algorithm for image compression. *Expert Systems with Applications*. 39 (1) 1078-1091. 2012.
- [20] T. Hassanzadeh, K. Faez, G. Seyfi, A speech recognition system based on Structure Equivalent Fuzzy Neural Network trained by Firefly algorithm. In *Proceedings of International Conference on Biomedical Engineering (ICoBE)*, 2012.
- [21] L.D.S. Coelho, V.C. Mariani, Improved firefly algorithm approach applied to chiller loading for energy conservation, *Energy and Buildings*, 59, 273-278, 2013.
- [22] S.J. Huang, X.Z. Liu, W.F. Su, S.H. Yang, Application of Hybrid Firefly Algorithm for Sheath Loss Reduction of Underground Transmission Systems. *IEEE Transactions on Power Delivery*, 28 (4), 2013. 2085-2092.
- [23] A. Kavousi-Fard, H. Samet, F. Marzbani, A new hybrid Modified Firefly Algorithm and Support Vector Regression model for accurate Short Term Load Forecasting. *Expert Systems with Applications*. 41 (2014) 6047–6056.
- [24] M.K. Marichelvam, T. Prabakaran, X.S. Yang, A Discrete Firefly Algorithm for the Multi-Objective Hybrid Flowshop Scheduling Problems. *IEEE Transactions on Evolutionary Computation*. 18 (2) 301-305, 2014.
- [25] I. Fister, X.S. Yang, J. Brest, I. Fister Jr, On the Randomized Firefly Algorithm. In X.-S. Yang (ed.), *Cuckoo Search and Firefly Algorithm*, 2014.
- [26] L. dos Santos Coelho, D. L. de Andrade Bernert, V.C. Mariani, A chaotic firefly algorithm applied to reliability – redundancy optimization, In *Proceedings of IEEE Congress on Evolutionary Computation (CEC)*, IEEE, 2011, 517 – 521.
- [27] S. Massan, A.I. Wagan, M.M. Shaikh, R. Abro, Wind turbine micrositeing by using the firefly algorithm. *Applied Soft Computing*. 27 (2015) 450-456.
- [28] L. Zhang, K. Mistry, S.C. Neoh, C.P. Lim, Intelligent facial emotion recognition using moth-firefly optimization. *Knowledge-Based Systems* (2016) 1–20.
- [29] A.H. Gandomi, X.S. Yang, S. Talatahari, A.H. Alavi, Firefly algorithm with chaos. *Communications in Nonlinear Science and Numerical Simulation*. Volume 18, Issue 1, January 2013, 89–98.
- [30] M. Mitić, N. Vuković, M. Petrović, Z. Miljković, Chaotic Fruit Fly Optimization Algorithm. *Knowledge-Based Systems* 89 (2015) 446–458.
- [31] B. Amiri, L. Hossain, J.W. Crawford, R.T. Wigand, Community Detection in Complex Networks: Multi-objective Enhanced Firefly Algorithm. *Knowledge-Based Systems* 46 (2013) 1–11.
- [32] D. Teshome, C.H. Lee, Y.W. Lin, K.L. Lian, A Modified Firefly Algorithm for Photovoltaic Maximum Power Point Tracking Control Under Partial Shading. *IEEE Journal of Emerging and Selected Topics in Power Electronics*, Volume: PP, Issue: 99, 2016. DOI: 10.1109/JESTPE.2016.2581858
- [33] X.S. Yang, Multiobjective firefly algorithm for continuous optimization. *Engineering with Computers* (2013) 29:175–184.
- [34] M. Alweshah, S. Abdullah, Hybridizing firefly algorithms with a probabilistic neural network for solving classification problems, *Applied Soft Computing*. 35 (2015) 513–524.

- [35] A. Kazem, E. Sharifi, F.K. Hussain, M. Saberik, O.K. Hussain, Support vector regression with chaos-based firefly algorithm for stock market price forecasting, *Applied Soft Computing*. 13 (2) (2013) 947–958.
- [36] C. Zhao, C. Wu, J. Chai, X. Wang, X. Yang, J.M. Lee, M.J. Kim, Decomposition-based multi-objective firefly algorithm for RFID network planning with uncertainty. *Applied Soft Computing*. 55 (2017) 549-564.
- [37] J. Dash, B. Dam, R. Swain, Design of multipurpose digital FIR double-band filter using hybrid firefly differential evolution algorithm. *Applied Soft Computing*. 59 (2017) 529-545.
- [38] M. Kaur, S. Ghosh, Network reconfiguration of unbalanced distribution networks using fuzzy-firefly algorithm. *Applied Soft Computing*. 49 (2016) 868-886.
- [39] T. Xiong, Y. Bao, Z. Hu, Multiple-output support vector regression with a firefly algorithm for interval-valued stock price index forecasting. *Knowledge-Based Systems*. 55 (2014) 87–100.
- [40] G. Kalantzis, C. Shang, Y. Lei, T. Leventouri, Investigations of a GPU-based levy-firefly algorithm for constrained optimization of radiation therapy treatment planning. *Swarm and Evolutionary Computation*. 26 (2016) 191-201.
- [41] O. Kisi, J. Shiri, S. Karimi, S. Shamshirband, S. Motamedi, D. Petkovi, R. Hashim, A survey of water level fluctuation predicting in Urmia Lake using support vector machine with firefly algorithm. *Applied Mathematics and Computation*. 270 (2015) 731-743.
- [42] T. Kanimozhi, K. Latha, An integrated approach to region based image retrieval using firefly algorithm and support vector machine. *Neurocomputing*, 151 (3) 1099-1111, 2015.
- [43] K. Mistry, L. Zhang, G. Sexton, Y. Zeng, M. He, Facial expression recognition using firefly-based feature optimization. In *Proceedings of IEEE Congress on Evolutionary Computation (CEC)*, 2017. Spain.
- [44] H. Su, Y. Cai, Q. Du, 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) 2017. 309–320.
- [45] Y. Zhang, X.F. Song, D.W. Gong, A return-cost-based binary firefly algorithm for feature selection. *Information Sciences*, 418–419 (2017) 561-574.
- [46] P.S. Rodrigues, G.A. Wachs-Lopes, H.R. Erdmann, M.P. Ribeiro, G.A. Giraldi, Improving a firefly meta-heuristic for multilevel image segmentation using Tsallis entropy. *Pattern Analysis and Applications*. 2017, 20 (1) 1–20.
- [47] W.C. Yeh, An improved simplified swarm optimization. *Knowledge-Based Systems* 82 (2015) 60–69.
- [48] A.R. Jordehi, Chaotic bat swarm optimisation (CBSO). *Applied Soft Computing*. Volume 26, January 2015, 523–530.
- [49] K. Mistry, L. Zhang, S.C. Neoh, C.P. Lim, B. Fielding, A Micro-GA Embedded PSO Feature Selection Approach to Intelligent Facial Emotion Recognition. *IEEE Transactions on Cybernetics*, (2017). 1-14.
- [50] W. Srisukham, L. Zhang, S.C. Neoh, S. Todryk, C.P. Lim, Intelligent Leukaemia Diagnosis with Bare-Bones PSO based Feature Optimization. *Applied Soft Computing*, 56. pp. 405-419. 2017.
- [51] Y. Wan, M. Wang, Z. Ye, X. Lai, A feature selection method based on modified binary coded ant colony optimization algorithm. *Applied Soft Computing*. Volume 49, December 2016, 248–258.
- [52] S.C. Neoh, L. Zhang, K. Mistry, M.A. Hossain, C.P. Lim, N. Aslam, P. Kinghorn, Intelligent Facial Emotion Recognition Using a Layered Encoding Cascade Optimization Model. *Applied Soft Computing*. Volume 34, September 2015, 72–93. 2015.
- [53] H. Hakli, H. Uğuz, A novel particle swarm optimization algorithm with Levy flight. *Applied Soft Computing*. 23 (2014) 333-345.
- [54] F. Kang, J.J., Li, Q. Xu, Damage detection based on improved particle swarm optimization using vibration data. *Applied Soft Computing*, 12 (8) 2329-2335, 2012.
- [55] T.H.H. Zavaschi, A.S. Britto Jr., L.E.S. Oliveira, A.L. Koerich, Fusion of feature sets and classifiers for facial expression recognition. *Expert Systems with Applications*, 40(2), 646-655. 2013.
- [56] H. Soyel, U. Tekguc, H. Demirel, Application of NSGA-II to feature selection for facial expression recognition. *Computers & Electrical Engineering*, Volume 37, Issue 6, 2011, 1232–1240.

- [57] B. Huang, B. Buckley, T.M. Kechadi, Multi-objective feature selection by using NSGA-II for customer churn prediction in telecommunications. *Expert Systems with Applications*, Volume 37, Issue 5, 2010, 3638–3646.
- [58] D. Herremans, K. Sørensen, Composing first species counterpoint with a variable neighbourhood search algorithm. *Journal of Mathematics and the Arts*, 6(4) (2012) 169-189.
- [59] S.C. Neoh, W. Srisukkhom, L. Zhang, S. Todryk, B. Greystoke, C.P. Lim, A. Hossain, N. Aslam, An Intelligent Decision Support System for Leukaemia Diagnosis using Microscopic Blood Images. *Scientific Reports*, 5 (14938), 2015.
- [60] Y. Zhang, L. Zhang, S.C. Neoh, K. Mistry, A. Hossain, Intelligent affect regression for bodily expressions using hybrid particle swarm optimization and adaptive ensembles, *Expert Systems with Applications*. 42 (22) (2015) 8678–8697.
- [61] K. Sørensen, Metaheuristics—the metaphor exposed. *International Transactions in Operational Research* 22 (2015) 3-18.
- [62] S. Mirjalili, SCA: A Sine Cosine Algorithm for Solving Optimization Problems, *Knowledge-based Systems*, 96 (2016) 120-133.
- [63] J. Derrac, S. García, D. Molina, F. Herrera, 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, 2011.
- [64] A. Fischer, F. Fischer, G. Jäger, J. Keilwagen, P. Molitor, I. Grosse, Exact algorithms and heuristics for the Quadratic Traveling Salesman Problem with an application in bioinformatics. *Discrete Applied Mathematics*. 166 (2014) 97-114.
- [65] N. Veerapen, G. Ochoa, M. Harman, E.K. Burke, An Integer Linear Programming approach to the single and bi-objective Next Release Problem. *Information and Software Technology*. 65 (2015) 1-13.
- [66] P. Xu, L. Wang, An exact algorithm for the bilevel mixed integer linear programming problem under three simplifying assumptions. *Computers & Operations Research*. 41 (2014) 309-318.
- [67] X. Ma, X. Zhu, B. Chen, Exact Algorithms for Maximizing Lifetime of WSNs using Integer Linear Programming. In *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*. 2017, USA.
- [68] C.J. Tan, S.C. Neoh, C.P. Lim, S. Hanoun, W.P. Wong, C.K. Loo, L. Zhang, S. Nahavandi, Application of an evolutionary algorithm-based ensemble model to job-shop scheduling. *Journal of Intelligent Manufacturing*. (2017). 1-12.
- [69] T.Y. Tan, L. Zhang, M. Jiang, An intelligent decision support system for skin cancer detection from dermoscopic images. In *Proceedings of 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*. 2016. 2194-2199.
- [70] D. Pandit, L. Zhang, C. Liu, S. Chattopadhyay, N. Aslam, C.P. Lim, A lightweight QRS detector for single lead ECG signals using a max-min difference algorithm. *Computer Methods and Programs in Biomedicine*, 144 (2017) 61-75.
- [71] J. Ganesan, H. Inbarani, Hybrid Tolerance Rough Set–Firefly based supervised feature selection for MRI brain tumor image classification, *Applied Soft Computing*, 46 (2016) 639-651.
- [72] P. Kinghorn, L. Zhang, L. Shao, A region-based image caption generator with refined descriptions. *Neurocomputing*. 272 (2018) 416-424.
- [73] P. Kinghorn, L. Zhang, L. Shao, A Hierarchical and Regional Deep Learning Architecture for Image Description Generation. *Pattern Recognition Letters*. (In Press).
- [74] M. Shafiee, E. Barshan, A. Wong, Evolution in Groups: A deeper look at synaptic cluster driven evolution of deep neural networks. In *Proceedings of Future Technologies Conference (FTC)*, 2017.
- [75] S. Fong, S. Deb, X. S. Yang, How meta-heuristic algorithms contribute to deep learning in the hype of big data analytics, in *Advances in Intelligent Systems and Computing*, 2017, vol. 518, 3–25.

23	Min	2.20E+03	2.20E+03	2.20E+03	2.23E+03	2.20E+03	3.13E+03	2.23E+03	2.26E+03	3.87E+03	3.30E+03	2.37E+03	2.37E+03	2.62E+03	4981.42	
	Max	2.20E+03	2.20E+03	2.20E+03	2.73E+03	2.59E+03	4.31E+03	2.93E+03	3.08E+03	6.54E+03	3.66E+04	2.99E+03	3.32E+03	3.74E+03	160524.9	
	Std	8.81E-02	1.88E-02	2.61E-02	1.50E+02	8.85E+01	2.69E+02	2.31E+02	2.07E+02	6.92E+02	8.53E+03	1.29E+02	2.48E+02	2.91E+02	41363.75	
	Mean	2.50E+03	2.50E+03	2.50E+03	2.51E+03	2.50E+03	2.98E+03	2.50E+03	2.51E+03	3.31E+03	3.19E+03	2.50E+03	2.52E+03	2.61E+03	4379.714	
24	Min	2.50E+03	2.50E+03	2.50E+03	2.50E+03	2.50E+03	2.71E+03	2.50E+03	2.50E+03	3.08E+03	2.60E+03	2.50E+03	2.50E+03	2.54E+03	3264.272	
	Max	2.50E+03	2.50E+03	2.50E+03	2.54E+03	2.51E+03	3.10E+03	2.50E+03	2.56E+03	3.61E+03	4.27E+03	2.51E+03	2.58E+03	2.75E+03	6369.239	
	Std	8.16E-06	1.29E-03	2.37E-03	1.06E+01	3.85E+00	8.08E+01	3.93E-04	1.22E+01	1.40E+02	3.84E+02	1.35E+00	1.79E+01	5.27E+01	679.7059	
	Mean	2.60E+03	2.60E+03	2.60E+03	2.61E+03	2.60E+03	2.80E+03	2.61E+03	2.62E+03	2.87E+03	2.81E+03	2.62E+03	2.65E+03	2.67E+03	2978.693	
25	Min	2.60E+03	2.60E+03	2.60E+03	2.61E+03	2.60E+03	2.78E+03	2.60E+03	2.61E+03	2.79E+03	2.71E+03	2.62E+03	2.63E+03	2.65E+03	2796.427	
	Max	2.60E+03	2.60E+03	2.60E+03	2.63E+03	2.61E+03	2.83E+03	2.62E+03	2.63E+03	2.91E+03	2.93E+03	2.64E+03	2.68E+03	2.74E+03	3084.484	
	Std	1.55E-02	4.15E-02	3.95E-02	3.21E+00	1.37E+00	1.29E+01	3.48E+00	6.49E+00	2.80E+01	5.40E+01	6.61E+00	1.34E+01	1.88E+01	61.71759	
	Mean	2.70E+03	2.70E+03	2.70E+03	2.71E+03	2.70E+03	2.78E+03	2.70E+03	2.71E+03	2.83E+03	2.83E+03	2.71E+03	2.71E+03	2.73E+03	3065.786	
26	Min	2.70E+03	2.70E+03	2.70E+03	2.70E+03	2.70E+03	2.75E+03	2.70E+03	2.70E+03	2.77E+03	2.75E+03	2.70E+03	2.70E+03	2.71E+03	2871.719	
	Max	2.70E+03	2.70E+03	2.70E+03	2.71E+03	2.70E+03	2.81E+03	2.70E+03	2.72E+03	2.90E+03	2.91E+03	2.72E+03	2.73E+03	2.77E+03	3344.874	
	Std	2.66E-02	2.00E-02	2.37E-02	1.94E+00	3.00E-02	1.34E+01	1.29E+00	4.55E+00	3.07E+01	4.68E+01	1.97E+00	5.38E+00	1.41E+01	113.3193	
	Mean	2.80E+03	2.72E+03	2.79E+03	2.75E+03	2.76E+03	2.71E+03	2.79E+03	2.77E+03	2.74E+03	2.84E+03	2.71E+03	2.79E+03	2.77E+03	3169.766	
27	Min	2.72E+03	2.71E+03	2.71E+03	2.71E+03	2.71E+03	2.71E+03	2.71E+03	2.71E+03	2.71E+03	2.71E+03	2.71E+03	2.71E+03	2.71E+03	2712.328	
	Max	2.80E+03	2.77E+03	2.80E+03	2.80E+03	2.71E+03	2.80E+03	2.80E+03	2.92E+03	3.04E+03	2.80E+03	2.80E+03	2.87E+03	2.87E+03	3390.504	
	Std	1.63E+01	1.45E+01	2.84E+01	4.79E+01	4.30E+01	2.91E-01	2.87E+01	4.57E+01	4.10E+01	1.16E+02	2.42E+01	3.26E+01	5.54E+01	160.4466	
	Mean	4.25E+03	2.90E+03	3.69E+03	3.26E+03	3.07E+03	4.04E+03	3.71E+03	3.28E+03	4.08E+03	4.26E+03	3.22E+03	3.64E+03	3.81E+03	4539.916	
28	Min	3.86E+03	2.90E+03	3.35E+03	3.16E+03	2.90E+03	3.75E+03	3.17E+03	3.15E+03	3.73E+03	3.68E+03	3.18E+03	3.16E+03	3.26E+03	3947.795	
	Max	4.64E+03	2.90E+03	4.15E+03	3.35E+03	4.22E+03	4.15E+03	4.07E+03	3.47E+03	4.30E+03	4.42E+03	3.30E+03	3.99E+03	4.17E+03	5307.483	
	Std	1.74E+02	3.29E-03	2.25E+02	6.30E+01	3.16E+02	9.21E+01	2.90E+02	8.00E+01	1.62E+02	1.32E+02	3.46E+01	3.21E+02	2.58E+02	226.1254	
	Mean	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	1131.3193
29	Min	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	10083.62	
	Max	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	3.00E+03	10083.62	
	Std	7.31E-06	5.75E-03	1.51E+03	4.24E+02	7.91E+02	2.17E+02	8.81E+02	9.54E+02	6.88E+02	1.78E+03	2.46E+02	5.88E+02	1.20E+03	916.6444	
	Mean	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	1311.8
30	Min	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	1311.8	
	Max	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	3.10E+03	1311.8	
	Std	1.38E-02	4.36E-03	7.95E+03	4.33E+03	6.30E+04	5.47E+06	6.03E+01	1.15E+03	4.55E+07	3.50E+05	4.48E+03	1.87E+06	3.87E+07	1.87E+08	
	Mean	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	10001451
31	Min	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	45029.19	
	Max	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	45029.19	
	Std	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	3.20E+03	21924770
	Mean	5.78E-02	1.54E-02	2.29E-02	6.26E+03	2.26E+04	1.72E+05	3.74E+02	1.33E+03	9.83E+05	3.95E+06	7.76E+02	2.28E+04	2.72E+05	4734927	

Table 8 Evaluation results for the CEC2014 benchmark functions with dimension=50

		ESRFA	SRFA	RFA	ODFA [3]	SFA [34]	LSFA [34]	CFA [35]	FA	SA	BSO	CSO	PSO	DFO	ALO
1	Mean	1.02E+02	1.44E+02	2.13E+02	3.49E+07	2.22E+06	2.17E+09	1.14E+05	1.02E+07	3.40E+09	3.27E+09	2.87E+07	2.08E+07	5.07E+08	7.36E+09
	Min	1.00E+02	1.22E+02	1.72E+02	1.10E+07	1.69E+02	1.39E+09	1.01E+04	1.06E+06	2.17E+09	1.10E+09	1.08E+07	4.45E+06	4.16E+07	4.46E+09
	Max	1.06E+02	1.66E+02	2.91E+02	8.08E+07	2.67E+07	2.79E+09	1.56E+06	2.34E+07	4.65E+09	5.83E+09	4.89E+07	5.97E+07	1.19E+09	1.2E+10
	Std	1.62E+00	9.82E+00	3.02E+01	1.46E+07	5.73E+06	3.56E+08	3.00E+05	6.16E+06	5.50E+08	1.29E+09	9.11E+06	1.50E+07	2.19E+08	2.02E+09
2	Mean	2.00E+02	1.74E+03	4.57E+03	1.34E+05	6.61E+07	1.27E+11	6.41E+02	4.54E+03	1.53E+11	7.83E+10	1.00E+10	1.34E+09	1.49E+10	1.98E+11
	Min	2.00E+02	1.27E+03	3.44E+03	7.11E+04	2.00E+02	1.07E+11	2.10E+02	2.02E+02	1.32E+11	3.81E+10	1.00E+10	1.95E+08	1.93E+09	1.5E+11
	Max	2.00E+02	2.13E+03	5.59E+03	2.47E+05	1.03E+09	1.44E+11	2.09E+03	2.63E+04	1.73E+11	1.63E+11	1.00E+10	3.06E+09	3.38E+10	2.42E+11
	Std	4.44E-02	2.35E+02	6.23E+02	4.05E+04	2.02E+08	8.33E+09	4.99E+02	6.97E+03	1.12E+10	2.75E+10	0.00E+00	8.29E+08	7.12E+09	2.36E+10
3	Mean	3.00E+02	3.00E+02	3.00E+02	1.04E+05	1.40E+03	2.20E+05	2.61E+03	4.02E+04	2.93E+05	5.80E+05	4.13E+04	4.76E+04	1.76E+05	33953991
	Min	3.00E+02	3.00E+02	3.00E+02	6.84E+04	3.00E+02	1.45E+05	1.07E+03	1.74E+04	1.91E+05	1.26E+05	2.31E+04	1.79E+04	6.02E+04	271947
	Max	3.00E+02	3.00E+02	3.00E+02	1.53E+05	1.38E+04	2.75E+05	7.58E+03	6.52E+04	6.54E+05	4.91E+06	6.27E+04	1.09E+05	2.79E+05	4.35E+08
	Std	5.23E-04	4.54E-03	3.86E+03	2.02E+04	2.61E+03	3.00E+04	1.43E+03	1.03E+04	1.07E+05	9.16E+05	1.01E+04	2.18E+04	5.65E+04	81996993
4	Mean	4.00E+02	4.00E+02	4.00E+02	4.59E+02	4.01E+02	2.14E+04	4.02E+02	4.52E+02	4.03E+04	1.95E+04	6.44E+02	7.31E+02	2.48E+03	65117.18
	Min	4.00E+02	4.00E+02	4.00E+02	4.42E+02	4.00E+02	1.53E+04	4.00E+02	4.42E+02	2.82E+04	8.02E+03	5.48E+02	5.61E+02	6.80E+02	40469.49
	Max	4.00E+02	4.00E+02	4.00E+02	5.36E+02	4.12E+02	2.65E+04	4.12E+02	5.17E+02	5.09E+04	3.64E+04	7.61E+02	1.04E+03	5.09E+03	90559.9
	Std	6.59E-09	5.20E-05	1.33E-04	2.33E+01	2.80E+00	2.83E+03	3.62E+00	1.77E+01	5.09E+03	7.09E+03	5.30E+01	1.05E+02	1.13E+03	12711.36
5	Mean	5.00E+02	5.00E+02	5.00E+02	5.20E+02	5.03E+02	5.21E+02	5.20E+02	5.20E+02	5.21E+02	5.20E+02	5.21E+02	5.21E+02	5.21E+02	521.5005
	Min	5.00E+02	5.00E+02	5.00E+02	5.20E+02	5.00E+02	5.21E+02	5.18E+02	5.20E+02	5.21E+02	5.20E+02	5.21E+02	5.21E+02	5.21E+02	521.3706
	Max	5.00E+02	5.00E+02	5.00E+02	5.20E+02	5.18E+02	5.21E+02	5.20E+02	5.20E+02	5.21E+02	5.20E+02	5.21E+02	5.21E+02	5.21E+02	521.5846
	Std	9.71E-06	2.19E-03	2.69E-03	5.28E-02	4.09E+00	3.11E-02	5.28E-01	6.38E-03	4.95E-02	2.42E-02	4.67E-02	5.27E-01	1.21E-01	0.051611
6	Mean	6.00E+02	6.00E+02	6.00E+02	6.09E+02	6.04E+02	6.76E+02	6.21E+02	6.13E+02	6.80E+02	6.76E+02	6.56E+02	6.47E+02	6.61E+02	686.8852
	Min	6.00E+02	6.00E+02	6.00E+02	6.04E+02	6.00E+02	6.70E+02	6.14E+02	6.08E+02	6.73E+02	6.64E+02	6.45E+02	6.39E+02	6.32E+02	681.3669
	Max	6.00E+02	6.00E+02	6.01E+02	6.13E+02	6.18E+02	6.80E+02	6.31E+02	6.22E+02	6.83E+02	6.87E+02	6.61E+02	6.53E+02	6.73E+02	691.383
	Std	5.71E-02	1.50E-02	2.11E-02	2.26E+00	4.32E+00	1.97E+00	3.83E+00	3.22E+00	1.85E+00	5.21E+00	4.31E+00	3.55E+00	8.67E+00	2.538596
7	Mean	7.00E+02	7.00E+02	7.00E+02	7.00E+02	7.01E+02	1.94E+								

