Computer Systems
Science and Engineering

Tech Science Press

DOI: 10.32604/csse.2023.038025
Article

Multi-Strategy Boosted Spider Monkey Optimization Algorithm for Feature Selection

Jianguo Zheng and Shuilin Chen*

Glorious Sun School of Business and Management, Donghua University, Shanghai, 200051, China
*Corresponding Author: Shuilin Chen. Email: 1219088@mail.dhu.edu.cn
Received: 24 November 2022; Accepted: 02 February 2023

Abstract: To solve the problem of slow convergence and easy to get into the local optimum of the spider monkey optimization algorithm, this paper presents a new algorithm based on multi-strategy (ISMO). First, the initial population is generated by a refracted opposition-based learning strategy to enhance diversity and ergodicity. Second, this paper introduces a non-linear adaptive dynamic weight factor to improve convergence efficiency. Then, using the crisscross strategy, using the horizontal crossover to enhance the global search and vertical crossover to keep the diversity of the population to avoid being trapped in the local optimum. At last, we adopt a Gauss-Cauchy mutation strategy to improve the stability of the algorithm by mutation of the optimal individuals. Therefore, the application of ISMO is validated by ten benchmark functions and feature selection. It is proved that the proposed method can resolve the problem of feature selection.

Keywords: Spider monkey optimization; refracted opposition-based learning; crisscross strategy; Gauss-Cauchy mutation strategy; feature selection

1 Introduction

Due to the increasing complexity of the object of the optimization problem, traditional algorithms are difficult to meet the requirements. Therefore, the design of new algorithms is a good way to solve them. Recently, there have been a lot of scholars who prefer metaheuristics due to their simplicity and high efficiency in solving problems. Many metaheuristics are of great importance in several areas [1–3]. According to Fister et al. [4], metaheuristics can be classified into four types: swarm intelligence, evolutionary, physics-based, and other. Swarm intelligence, such as the whale optimization algorithm (WOA) [5], grey wolf optimizer (GWO) [6], slime mould algorithm (SMA) [7], sparrow search algorithm (SSA) [8], chimp optimization algorithm (ChOA) [9], and spider monkey optimization (SMO) [10]. Evolutionary, such as genetic algorithms (GA) [11] and differential evolution (DE) [12]. Physics-based, such as simulated annealing (SA) [13] and central force optimization (CFO) [14]. Other, such as gaining-sharing knowledge based algorithm (GSK) [15] and moth-flame optimization (MFO) [16]. SMO is a new kind of swarm intelligence algorithm, which is based on the social structure of fission-fusion. Compared with other methods, SMO is characterized by a small number of parameters, robust and globally convergent. But, just like other methods, there are some problems, such as slow convergence and local optimization in SMO.

Feature selection, as a kind of data pre-processing technique in machine learning, can get rid of unnecessary noise and redundancy features from the data set and extract essential components at the same time so that it can decrease the dimension of data and accelerate the performance of machine learning algorithms. Feature selection is one of the most critical problems in classification tasks. The search space can generate 2^n results for data with n properties. There are two main approaches to the choice of characteristic: the exhaustive approach and the metaheuristic approach. The exhaustive approach usually has to list all the solutions in the search space to obtain a better set of characteristics, but it is not very easy to choose them because of the large memory requirements. The metaheuristic approach for feature selection is efficient, but they can only get a subset of the best local features. Agrawal et al. [17] proposed an optimal method to apply feature selection. Hussain et al. [18] presented a novel approach that can achieve a maximum of 87% reduction in the performance of low and high-dimensional tasks. Hu et al. [19] applied three strategies to solve feature selection.

A lot of researchers have used a variety of approaches to enhance the SMO's performance. First, in optimization of the control parameters, Sharma et al. [20] split spider monkeys into three types of monkeys by their age. They implemented an age-stratification strategy for the local leadership phase and suggested an age-stratification spider monkey optimization (ASMO), which has a more practical biological significance. Kalpana et al. [21] optimized control parameters in combination with exponentially weighted moving averages. Secondly, in optimizing the local and global leader position update, Menon et al. [22] applied prediction theory to the local and global lead stages of population position update. Gupta et al. [23] introduced a quadratic approximation operator to improve the local search ability. Mumtaz et al. [24] used a genetic operator perturbation mutation to enhance the performance of the algorithm from local space. Despite the improvement in their capacity to escape from local exploitation, unbalanced exploitation and exploration remain to be further improved.

This paper presents a new algorithm based on multi-strategy (ISMO) by introducing four different strategies: refracted opposition-based learning strategy, non-linear adaptive dynamic weight factor strategy, and crisscross and Gauss-Cauchy mutation strategy. The application of ISMO is validated by ten benchmark functions and feature selection. The results indicate that ISMO is superior to other competitors.

The rest of this study is organized as follows. Part "Spider Monkey Optimization (SMO)" describes the mathematical model of the SMO. Part "Improved Spider Monkey Optimization (ISMO)" contains the proposed ISMO. Part "Experiment Results and Discussion" presents the ISMO's performance evaluation and statistical analysis. In part "Feature Selection Optimization Comparison Experiment," the applicability of ISMO in feature selection is evaluated. Finally, part "Conclusions and Future Research" summarizes the conclusions and future work.

2 Spider Monkey Optimization (SMO)

The SMO comprises seven phases, which are addressed in the following subsection.

(1) Initialization phase: the SMO generates a uniformly distributed initial population of N spider monkeys, where each monkey SM_i ($i = 1, 2, \ldots, N$) is a D-dimensional vector. D is the number of variables. Each SM_i is initialized as follows:

$$SM_{ij} = SM_{minj} + U(0,1) \times \left(SM_{maxj} - SM_{minj}\right) \tag{1}$$

where SM_{minj} and SM_{maxj} are the upper and lower of SM_i in the j dimension, U (0, 1) is the range [0, 1].

(2) Local leader phase (LLP): each spider monkey adjusts its current location based on the experience of local group members. The location update formula is calculated as follows:

$$SM_{newij} = SM_{ij} + U(0,1) \times \left(LL_{kj} - SM_{ij}\right) + U(-1,1) \times \left(SM_{rj} - SM_{ij}\right) \tag{2}$$

where LL_{kj} represents the j dimension of the k group of the local leader. SM_{rj} is the j dimension of the r spider monkey chosen randomly within the k group, so $r \neq i$, U (−1, 1) is a random number between −1 and 1.

(3) Global leader phase (GLP): all the spider monkeys update their locations using the experience of global leaders. The location update equation is calculated as follows:

$$SM_{newij} = SM_{ij} + U(0,1) \times \left(GL_j - SM_{ij}\right) + U(-1,1) \times \left(SM_{rj} - SM_{ij}\right) \tag{3}$$

where GL_j represents the j dimension of the global leadership location and $j \in \{1, 2, \cdots, D\}$ is the randomly chosen index.

The locations of spider monkeys are updated based on a probability p_i calculated using their fitness. The p_i is calculated as follows:

$$p_i = x \times \frac{F_t}{F_{max}} + y \tag{4}$$

where F_t is the fitness of the spider monkey, $x + y = 1$, usually $x = 0.9$ and $y = 0.1$.

(4) Global leader learning phase (GLL): the location of the global leader is updated by using the greedy selection in the population. In addition, check that the location of the global leader is being updated, and if not, add 1 to the Global Limit Count.

(5) Local leader learning phase (LLL): the location of the local leader is updated by using the greedy selection in the population. In addition, check that the location of the local leader is being updated, and if not, add 1 to the Local Limit Count.

(6) Local leader decision phase (LLD): if the location of any local leader is not updated up to a predetermined threshold, known as the Local Leader Limit, then all the members of the group will update their locations either through random initialization or using a combination of information from the global and local leader by Eq. (5) based on the perturbation rate (pr).

$$SM_{newij} = SM_{ij} + U(0,1) \times \left(GL_j - SM_{ij}\right) + U(0,1) \times \left(SM_{rj} - LL_{kj}\right) \tag{5}$$

(7) Global leader decision phase (GLD): the location of the global leader is monitored, and if it is not updated up to a predetermined number of iterations called Global Leader Limit, then global leaders will divide the population into smaller groups.

The complete pseudo-code of the SMO is given in reference [10].

3 Improved Spider Monkey Optimization (ISMO)

Based on the above analysis, the improvement of SMO is made in four aspects. First, the initial population is generated by a refracted opposition-based learning strategy to enhance diversity and ergodicity. Second, this paper introduces a non-linear adaptive dynamic weight factor to improve convergence efficiency. Then, using the crisscross strategy, using the horizontal crossover to enhance the global search and vertical crossover to keep the diversity of the population to avoid being trapped in the local optimum. At last, we adopt a Gauss-Cauchy mutation strategy to improve the stability of the algorithm by mutation of the optimal individuals. Therefore, the collaboration of the four search strategies can enhance diversification, exploration, and exploitation.

3.1 Refracted Opposition-Based Learning Strategy

Opposition-based learning is a widely used approach for the estimation of population initialization [25]. The idea is to extend the search by calculating the opposite solution to the current solution and finding a better solution to a given problem [26]. Reference [27–29] is a combination of metaheuristic and opposition-based learning, and it has been demonstrated that opposition-based learning can increase the precision of the algorithm. However, there are some disadvantages to opposition-based learning. Therefore, the opposition-based learning strategy is combined with the principle of refraction [30] to decrease the possibility of premature convergence in the later period. The details are illustrated in Fig. 1.

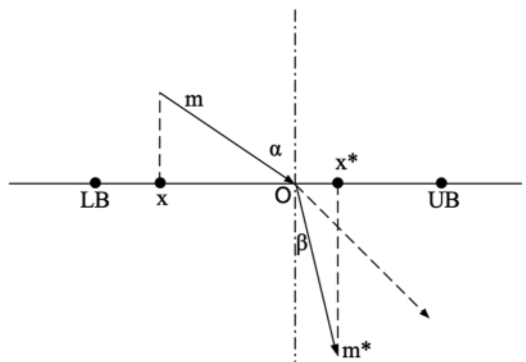

Figure 1: Refracted opposition-based learning

Where the x-axis search interval is known as $[LB, UB]$, the origin O is the midpoint on $[LB, UB]$, α and β are represented by the incidence angle and the refraction angle, respectively. m and $m*$ are the lengths of the incident and the refracted rays, respectively. The above formula can be obtained as follows.

$$\mathrm{n} = \frac{\sin\alpha}{\cos\beta} = \frac{\dfrac{LB+UB}{2} - x}{x^* - \dfrac{LB+UB}{2}} \times \frac{m^*}{m} \tag{6}$$

Let $\sigma = \dfrac{m}{m^*}$ and $n=1$. Substituted into Eq. (6) and expanded to the high-dimensional space of the spider monkey optimization yields the refracted direction solution $x^*_{i,j}$, as follows.

$$x^*_{i,j} = \frac{LB_j + UB_j}{2} + \frac{LB_j + UB_j}{2\sigma} - \frac{x_{i,j}}{\sigma} \tag{7}$$

where $x_{i,j}$ denotes the position of the spider monkey in the population in j dimensions ($i = 1, 2, \cdots, N; j = 1, 2, \cdots, D$), $x^*_{i,j}$ denotes the refracted opposition solution of $x_{i,j}$, LB_j, and UB_j are the lower and upper bounds of the dynamic boundary, respectively.

3.2 Nonlinear Adaptive Dynamic Weight Factor Strategy

Shi et al. [31] are the first to apply the weight factor ω to particle swarm optimization (PSO). Eq. (2) indicates that the current location is updated according to the local leader's experience, the spider monkey's experience within the group, and its own experience. It is easy to lack local search ability in the late iteration. From Eq. (5), it is found that the original location information is ultimately inherited by the new location, and the local search capability of the algorithm is also decreased. Therefore, the introduction of adaptive dynamic weight factor ω is considered in Eqs. (2) and (5).

$$SM_{newij} = \omega(t) \times SM_{ij} + U(0,1) \times \left(LL_{kj} - SM_{ij}\right) + U(-1,1) \times \left(SM_{rj} - SM_{ij}\right) \tag{8}$$

$$SM_{newij} = \omega(t) \times SM_{ij} + U(0,1) \times \left(GL_j - SM_{ij}\right) + U(0,1) \times \left(SM_{rj} - LL_{kj}\right) \tag{9}$$

$$\omega(t) = \omega_i + \left(\omega_f - \omega_i\right) P_s(t) \exp\left(-(\alpha t/T_{max})^2\right) \tag{10}$$

In Eq. (10), ω_i is the initial weight factor, ω_f is the final weight factor in the late iteration, α is the non-linear parameter, t is the current number of iterations, T_{max} is the maximum number of iterations, $P_s(t)$ is the population success rate, and the calculation steps are as follows.

Taking the minimization problem as an example, the success value $S(i,t)$ is defined as follows:

$$S(i,t) = \begin{cases} 1 \; fit(pb_i(t)) < fit(pb_i(t-1)) \\ 0 \; fit(pb_i(t)) = fit(pb_i(t-1)) \end{cases} \tag{11}$$

In Eq. (11), $pb_i(t)$ denotes the optimal historical position, $fit(\cdot)$ is the fitness function, and the success rate $P_s(t)$ of the whole population based on the success value $S(i, t)$ is formulated as follows.

$$P_s(t) = \frac{1}{N}\sum_{i=1}^{N} S(i,t) \tag{12}$$

In Eq. (12), N is the population size, $P_s(t) \in [0, 1]$.

3.3 Crisscross Strategy

The crisscross strategy [32] is introduced to increase the precision of convergence, but it does not influence the convergence rate.

3.3.1 Horizontal Crossover Operation

The horizontal crossover strategy is to perform crossover operations in the same dimension of different populations. The formula is as follows:

$$x_{ij}^t = r_1 \times x_{ij}^t + (1 - r_1) \times x_{kj}^t + c_1 \times \left(x_{ij}^t - x_{kj}^t\right) \tag{13}$$

$$x_{kj}^t = r_2 \times x_{kj}^t + (1 - r_2) \times x_{ij}^t + c_2 \times \left(x_{kj}^t - x_{ij}^t\right) \tag{14}$$

where x_{ij}^t and x_{kj}^t denote the j dimensional generated by the two spider monkeys after later crossover. r_1 and r_2 are random numbers of [0, 1], and c_1 and c_2 are random numbers of [−1, 1].

3.3.2 Vertical Crossover Operation

The vertical crossover operation is performed in all dimensions of the newborn individual, and the crossover operation occurs with less probability than the horizontal crossover. The formula is as follows:

$$x_{ij}^t = r \times x_{ij_1}^t + (1 - r) \times x_{ij_2}^t \tag{15}$$

In Eq. (15), x_{ij}^t is a child individual generated by the j_1 and j_2 dimensions of individual x_{ij}^t by longitudinal crossover, $r \in [0, 1]$.

3.4 Gauss-Cauchy Mutation Strategy

In the late iteration of the SMO, the rapid assimilation of the spider monkeys is prone to local optimal stagnation. The Gauss-Cauchy mutation strategy [33] is used to mutate the global leader, compare the locations before and after the mutation, and select the better location to substitute into the next iteration, as follows:

$$\overline{X_{best}^t} = X_{best}^t \times \left[1 + \lambda_1 Cauchy\left(0, \sigma^2\right) + \lambda_2 Gauss\left(0, \sigma^2\right)\right] \tag{16}$$

$$Cauchy\left(0, \sigma^2\right) = \frac{1}{\pi\left(1 + x^2\right)} \tag{17}$$

$$Gauss\left(0, \sigma^2\right) = \frac{1}{\sqrt{2\pi}} e^{\left(-\frac{x^2}{2}\right)} \tag{18}$$

In Eq. (17), $\overline{X_{best}^t}$ denotes the location of the optimal individual after mutation, σ^2 is the standard deviation of the Gauss-Cauchy mutation, $\lambda_1 = 1 - t^2/T_{max}^2$ and $\lambda_2 = t^2/T_{max}^2$ are dynamic parameters that adjust adaptively with the number of iterations.

The steps of ISMO are illustrated in Algorithm 1.

Algorithm 1: ISMO Algorithm

1: Input: refracted opposition-based learning strategy to initializing population, local leader limit, global leader limit, pr
2: Calculate fitness
3: Select global leader and local leaders by using the greedy selection
4: While (Termination criteria are not satisfied) do
    4.1 Location update for all the spider monkeys based on LLP with Eq. (8).
    4.2 Choosing an optimal location between a new and an existing one based on fitness and using a greedy selection process.
    4.3 Calculate the probability p_i for all the group members using Eq. (4).
    4.4 Use the Crisscross strategy.
    4.5 Apply Gauss-Cauchy mutation strategy.
    4.6 Location update for all the group members selected by p_i based on GLP using Eq. (9).
    4.7 Update local and global leadership locations by applying a greedy selection process to all members of the group.
    4.8 If any of the local leaders still need to update their location for a predefined number of iterations, then use the LLD to redirect all members of the group.
    4.9 The group is divided if the global leader needs to update the location for the predefined number of iterations. If the number of groups present is less than the maximum number (MG), all the subgroups combine to form one single group.
5: Return the best result found so far.

The main steps of the proposed ISMO are illustrated in Fig. 2.

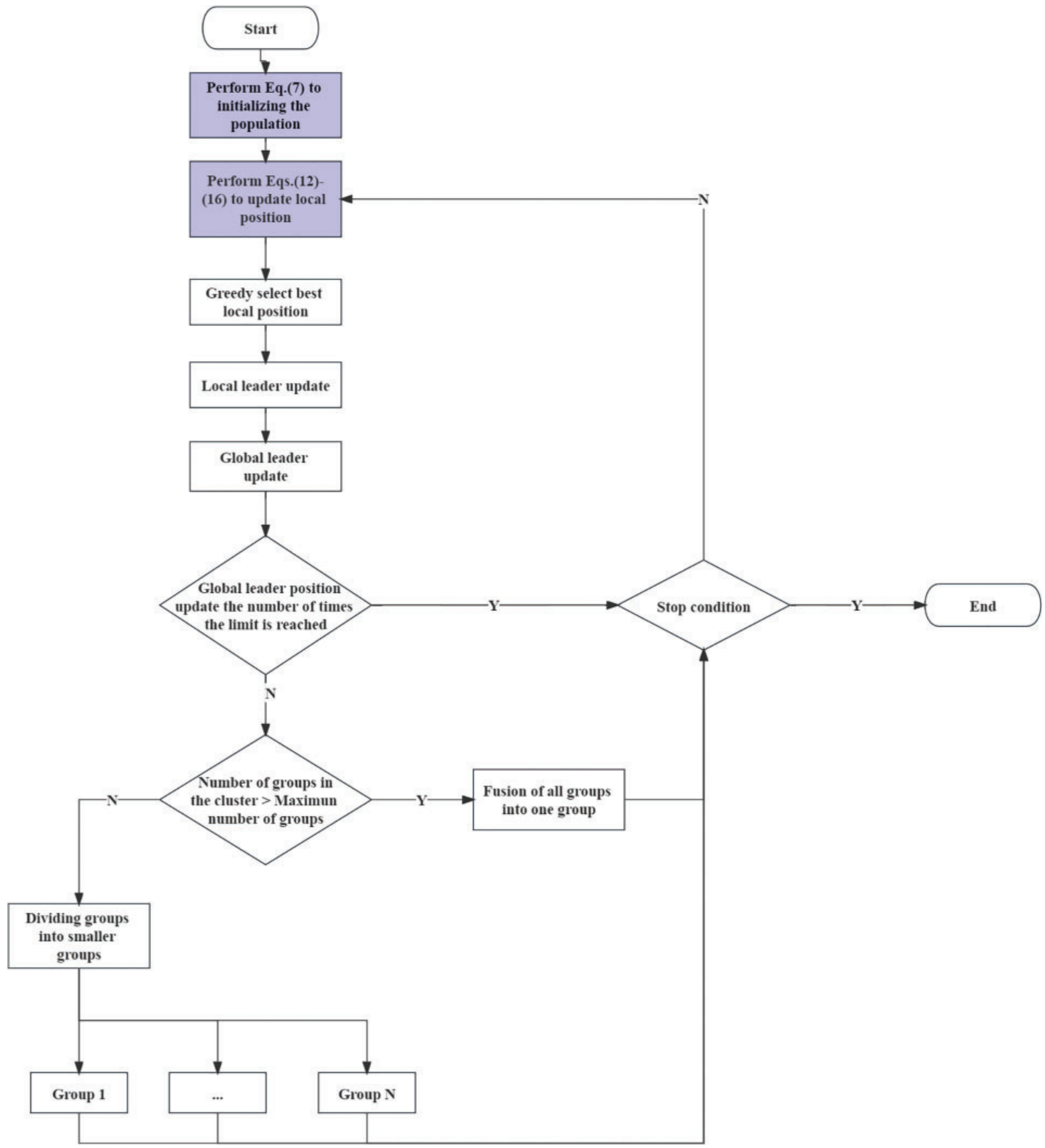

Figure 2: The main structure of the proposed ISMO

4 Experiment Results and Discussion

To prove the validity and robustness of ISMO, we choose ten classical unimodal and multimodal functions, in which F1–F6 aims to check the convergence speed and precision, and F7–F10 aims to measure the ability to overcome local optimum, as illustrated in Table 1.

Table 1: Details of ten classical benchmark functions

Class	Function	Dimension	Range	f_{min}
Unimodal functions	$F_1(x) = \sum_{i=1}^{D} x_i^2$	30	[−100, 100]	0
	$F_2(x) = \sum_{i=1}^{D} \lvert x_i \rvert + \prod_{i=1}^{D} \lvert x_i \rvert$	30	[−10, 10]	0
	$F_3(x) = \sum_{i=1}^{D} \left(\sum_{j=1}^{i} x_j \right)^2$	30	[−100, 100]	0
	$F_4(x) = max_i \{\lvert x_i \rvert, 1 \le i \le D\}$	30	[−100, 100]	0
	$F_5(x) = \sum_{i=1}^{D} i x_i^2$	30	[−5.12, 5.12]	0
	$F_6(x) = \sum_{i=1}^{D} (x_i + 0.5)^2$	30	[−100, 100]	0
Multimodal functions	$F_7(x) = \sum_{i=1}^{n} \left[x_i^2 - 10\cos(2\pi x_i) + 10 \right]$	30	[−5.12, 5.12]	0
	$F_8(x) = \frac{\pi}{D} \left\{ 10\sin(\pi y_1) + \sum_{i=1}^{D-1} (y_i - 1)^2 \left[1 + 10\sin^2(\pi y_{i+1}) \right] + (y_D - 1)^2 \right\} + \sum_{i=1}^{D} u(x_i, 10, 100, 4)$, $y_i = 1 + \frac{x_i - 1}{4}$, $u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a < x_i < a \\ k(-x_i - a)^m & x_i < -a \end{cases}$	30	[−50, 50]	0
	$F_9(x) = -20\exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2} \right) - \exp\left(\frac{1}{D}\sum_{i=1}^{D} \cos(2\pi x_i) \right) + 20 + e$	30	[−32, 32]	0
	$F_{10}(x) = \frac{1}{4000}\sum_{i=1}^{D} x_i^2 - \prod_{i=1}^{D} \cos\left(\frac{x_i}{\sqrt{i}} \right) + 1$	30	[−600, 600]	0

4.1 Comparison Against Other Methods

This study selected the performance of the SMO, ISMO, whale optimization algorithm (WOA), grey wolf optimizer (GWO), sine cosine algorithm (SCA), slime mould algorithm (SMA), sparrow search algorithm (SSA), chimp optimization algorithm (ChOA), and gaining-sharing knowledge based algorithm (GSK) for comparison, the parameters of algorithms are described in Table 2.

As shown in Table 3, ISMO is superior to the other eight algorithms for ten benchmark functions. The average (Ave) and the standard deviation (SD) of ISMO are superior to other methods, which indicates that ISMO has better stability and robustness and ISMO can search not only the exploration space but also guarantee global search capability. Detailed results are presented in Table 3, and Fig. 3 illustrates the convergence values of all compared methods on the benchmark functions.

Table 2: The parameters of the methods

Algorithm	Specifications
SMO	$N = 50$, $MG = \mathrm{N}/10$, $LLL = \mathrm{D} \times \mathrm{N}$, $GLL = \mathrm{N}$, $pr = 0.1$
ISMO	$N = 50$, $MG = \mathrm{N}/10$, $LLL = \mathrm{D} \times \mathrm{N}$, $GLL = \mathrm{N}$, $pr = 0.1$
WOA	α from 2 linearly decreasing to 0, $b = 1$
GWO	α from 2 linearly decreasing to 0, $\gamma_1, \gamma_2 \in [0, 1]$
SCA	$\alpha\ (t)$ from 2 linearly decreasing to 0
SMA	$z = 0.03$
SSA	$pNum = 0.2$ NP, $sNum = 0.2$ NP, $R_2 = 0.8$
ChOA	$m = \mathrm{chaos}(3, 1, 1)$
GSK	$P = 0.1$, $k_f = 0.5$, $k_r = 0.9$, $K = 10$

Table 3: Experimental test results of different methods

Function	Evaluation	SMO	WOA	GWO	SCA	SMA	SSA	ChOA	GSK	ISMO
F1	Ave	5.87E−17	2.98E−58	1.08E−27	4.35E−18	8.62E−268	9.17E−09	1.00E−48	1.01E−03	**0**
	SD	1.02E−17	1.52E−57	1.34E−27	1.78E−17	0	1.80E−08	3.29E−48	2.98E−03	**0**
F2	Ave	2.52E−16	4.94E−26	8.74E−17	1.90E−11	8.89E−157	1.76E−05	1.38E−28	2.19E−03	**0**
	SD	1.19E−17	2.66E−25	2.96E−17	8.13E−11	1.78E−156	1.45E−05	2.24E−28	2.78E−03	**0**
F3	Ave	9.72E−03	4.82E−10	2.69E−06	2.65E−10	1.40E−266	4.48E−07	1.40E−17	7.14E+01	**0**
	SD	0	1.37E−09	3.31E−06	1.23E−09	0	8.82E−07	1.55E−17	1.66E+02	**0**
F4	Ave	2.60E−18	1.07E−14	1.65E−202	1.69E−52	2.16E−271	1.27E−11	7.87E−240	6.70E−72	**0**
	SD	2.61E−18	5.02E−14	0	7.17E−52	0	2.53E−11	0	1.68E−71	**0**
F5	Ave	5.77E−17	1.64E−71	2.70E−29	1.00E−18	3.59E−259	8.10E−07	4.28E−50	8.09E−06	**0**
	SD	8.50E−18	8.68E−71	3.88E−29	5.06E−18	0	1.57E−06	9.09E−50	1.61E−05	**0**
F6	Ave	0	0	0	0	0	0	0	1.27E+01	**0**
	SD	0	0	0	0	0	0	0	1.61E+01	**0**
F7	Ave	4.84E−18	2.97E−84	3.78E+00	6.24E−21	0	8.40E−05	2.91E−44	1.21E+02	**0**
	SD	8.70E−19	1.60E−83	3.83E+00	2.52E−20	0	3.24E−04	1.53E−43	5.68E+01	**0**
F8	Ave	3.59E−01	8.31E−02	4.36E−02	9.14E−01	3.84E−02	2.88E−04	6.37E−02	1.23E−01	**1.92E−07**
	SD	3.58E−02	1.48E−01	6.59E−03	4.95E−02	2.20E−02	1.58E−04	1.39E−02	2.19E−01	**1.44E−07**
F9	Ave	2.57E−16	8.70E−24	8.07E−15	1.81E−10	3.00E−98	5.13E−04	1.07E−25	1.39E+00	**3.00E−98**
	SD	2.62E−17	4.35E−23	6.15E−15	6.33E−10	6.50E−114	5.11E−04	1.53E−25	9.24E−01	**6.50E−114**
F10	Ave	6.38E−19	0	1.49E−02	1.80E−15	0	5.16E−09	0	6.32E−02	**0**
	SD	1.48E−20	0	1.53E−02	9.64E−15	0	5.22E−09	0	5.33E−02	**0**

4.2 Impact of Introduced Strategies

In this section, we compare the performance of ISMO with four different search strategies, for example, spider monkey optimization with refracted opposition-based learning strategy (ISMO-1), spider monkey optimization with non-linear adaptive dynamic weight factor (ISMO-2), spider monkey optimization with crisscross strategy (ISMO-3), and spider monkey optimization with Gauss-Cauchy mutation strategy (ISMO-4). From Table 4, the performance of ISMO is superior to other comparison algorithms.

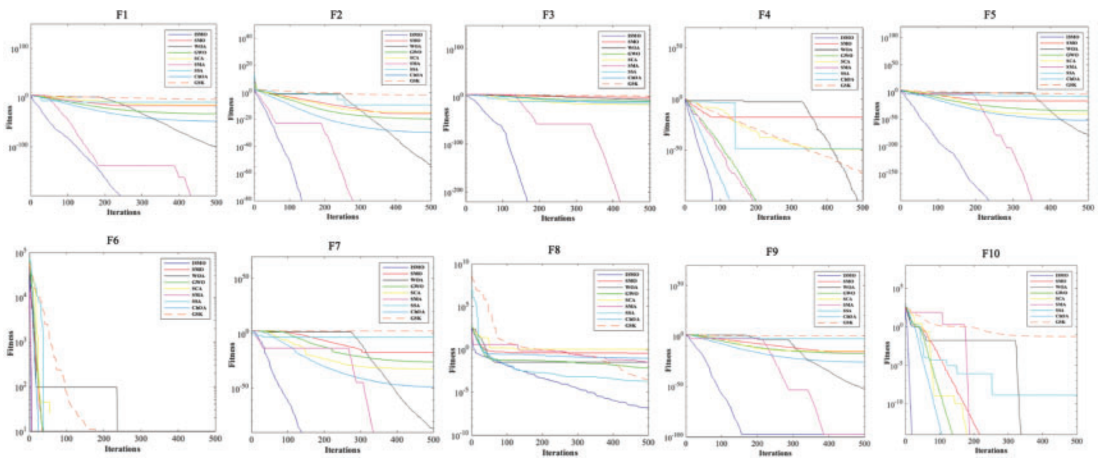

Figure 3: Convergence curves

Table 4: Experimental test results of different strategy

Function	Evaluation	ISMO-1	ISMO-2	ISMO-3	ISMO-4	ISMO
F1	Ave	3.90E−19	1.46E−34	2.45E−61	0	**0**
	SD	3.88E−19	1.30E−34	1.16E−61	0	**0**
F2	Ave	7.44E−13	1.42E−17	7.22E−34	0	**0**
	SD	2.61E−13	5.68E−18	4.34E−35	0	**0**
F3	Ave	4.81E−02	1.77E+03	1.58E+00	0	**0**
	SD	1.18E−02	8.79E+02	6.83E−01	0	**0**
F4	Ave	4.29E−21	1.43E−18	2.64E−31	0	**0**
	SD	3.53E−21	8.47E−19	2.62E−31	0	**0**
F5	Ave	4.03E−21	4.96E−22	1.38E−61	0	**0**
	SD	3.99E−21	4.86E−22	1.01E−61	0	**0**
F6	Ave	0	0	0	0	**0**
	SD	0	0	0	0	**0**
F7	Ave	6.61E−21	4.63E−09	9.54E−30	0	**0**
	SD	8.95E−21	6.55E−09	1.35E−29	0	**0**
F8	Ave	6.24E−06	3.32E−01	1.40E−11	9.04E−03	**1.92E−07**
	SD	5.09E−06	2.51E−02	7.75E−12	3.42E−03	**1.44E−07**
F9	Ave	5.52E−12	2.23E−17	5.88E−31	3.00E−98	**3.00E−98**
	SD	5.25E−12	1.67E−17	1.85E−31	6.49E−114	**6.50E−114**
F10	Ave	7.73E−22	9.05E−23	2.47E−61	0	**0**
	SD	4.73E−22	7.39E−23	2.41E−61	0	**0**

4.3 Nonparametric Statistical Test Analysis

The Wilcoxon signed-rank test and Friedman test [34] are applied to compare the performance of ISMO with other methods, and the significant level is assumed to be 5%. The results of the nonparametric statistical analysis of the ISMO and different methods are presented in Tables 5 and 6, respectively.

Table 5: Experimental results of Wilcoxon signed-rank test

Function	ISMO *vs.* SMO		ISMO *vs.* WOA		ISMO *vs.* GWO		ISMO *vs.* SCA		ISMO *vs.* SMA		ISMO *vs.* SSA		ISMO *vs.* ChOA		ISMO *vs.* GSK	
	p	R	*p*	R	*p*	R	*p*	R	*p*	R	*p*	R	*p*	R	*p*	R
F_1	***	+	***	+	***	+	***	+	***	+	***	+	***	+	***	+
F_2	***	+	***	+	***	+	***	+	***	+	***	+	***	+	***	+
F_3	***	+	***	+	***	+	***	+	***	+	***	+	***	+	***	+
F_4	***	+	***	+	***	+	***	+	***	+	***	+	***	+	***	+
F_5	***	+	***	+	***	+	***	+	***	+	***	+	***	+	***	+
F_6	1.70e−08	+	***	+	1.71e−08	+	4.19e−11	+	***	+	4.44e−11	+	3.48e−08	+	***	+
F_7	***	+	***	+	***	+	***	+	***	+	***	+	***	+	***	+
F_8	***	+	***	+	***	+	***	+	***	+	***	+	***	+	***	+
F_9	***	+	***	+	***	+	***	+	***	+	***	+	***	+	***	+
F_{10}	***	+	***	+	***	+	***	+	***	+	***	+	***	+	***	+
+/=/−	10/0/0		10/0/0		10/0/0		10/0/0		10/0/0		10/0/0		10/0/0		10/0/0	

Notes: ***denotes $p = 2.2e{-}16$.

Table 6: Experimental results of the Friedman test

Algorithm	Rank mean	Rank
SMO	6.35	7
ISMO	1.60	1
WOA	4.30	4
GWO	5.65	5
SCA	6.05	6
SMA	2.30	2
SSA	6.75	8
ChOA	3.60	3
GSK	8.40	9
Chi-square	56.91	
p	1.87E−09	

In Table 5, "+," "−," and "=" indicate the number of functions where ISMO is superior, inferior, and equivalent to other methods. The *p*-values of the ten functions are lower than 5%, meaning that ISMO differs significantly from other methods. In Table 6, the average rank of ISMO is 1.60, which is lower than other methods. As a result, the ISMO has been proven to have excellent performance.

5 Feature Selection Optimization Comparison Experiment

5.1 Feature Selection Problem Description

This study applies ISMO to solve the feature selection. Each individual in the overall ISMO represents a combination of features, also called a feature subset. Each dimension is determined by the number of original features in the dataset. Each vector is made up of 0 and 1, where 0 indicates the unselected feature property, and 1 indicates the selection of the corresponding feature property. In the ISMO population initialization, the individual dimension is randomly [0, 1], so the individual vectors in the population consist of 0 and 1. In this study, the values of each dimension above 0.65 are set to 1, and the rest are set to 0 to obtain the individual vectors of 0 and 1.

To balance the minimization of selection features and the maximization of classification accuracy, the fitness function is given by:

$$F = \alpha \times \gamma_R(D) + \beta \times \frac{|RF|}{|Size|} \tag{19}$$

where $\gamma_R(D)$ denotes the classification error rate of a given classifier (the K-nearest neighbor (KNN) classification algorithm is used to evaluate the merit of feature subsets), |RF| denotes the number of features contained in the current individual, |Size| denotes the original number of features in the dataset, α and β denote the importance of the corresponding classification quality and feature subset length, respectively, α and $\beta \in [0, 1]$, and $\alpha + \beta = 1$. In this study, $\alpha = 0.99$.

To evaluate the performance of ISMO, the mean fitness (Mean), average classification accuracy (AC_{avg}), and average feature selection (FS_{avg}) are chosen as evaluation indicators, and the formula for each indicator is presented in Eqs. (20)–(22).

$$\text{Mean} = \frac{1}{SR}\sum_{i=1}^{SR} F_t \tag{20}$$

where F_t denotes the run of the optimal solution.

$$AC_{avg} = \frac{1}{SR}\sum_{i=1}^{SR} \frac{TP + TN}{P + N} \tag{21}$$

where SR denotes the number of runs, TP, TN, P, and N represent the number of predicted true samples, the number of predicted true negative samples, the number of positive samples, and the number of negative samples, respectively.

$$FS_{avg} = \frac{1}{SR}\sum_{i=1}^{SR} RF \tag{22}$$

where RF denotes the number of optimal feature subsets with element one obtained from each algorithm run.

5.2 Dataset Description

To prove the validity of the ISMO feature selection, we choose seven datasets from the University of California, which are all common test cases for machine learning. A description of the dataset is given in Table 7, including dataset number, name, number of characteristics, number of samples, and number of categories. Ten-fold cross-validation is used for training and testing samples. Every dataset is randomly divided into two groups with different ratios, 90% of the cases are training, and 10% are used for testing. The maximum number of iterations is $T_{max} = 100$, and the result is chosen as the mean of 10 independent experiments. Details of the test dataset are given in Table 7.

Table 7: Test dataset

Number	Name	Characteristics	Samples	Categories
D1	Avila	10	20867	12
D2	Fertility	10	100	2
D3	Wine	13	178	3
D4	Lymphography	18	148	4
D5	Dermatology	33	366	6
D6	Ionosphere	34	351	2
D7	Sonar	60	208	2

5.3 Experimental Results

To compare the overall performance of ISMO, it is compared with other methods, such as grey wolf optimizer (GWO), whale optimization algorithm (WOA), sine cosine algorithm (SCA), and K-nearest neighbor (KNN). Tables 8–10 compare WOA, GWO, SCA, and KNN performance on seven datasets in terms of mean fitness, average classification accuracy, and average feature selections, respectively. The population size is 50, and the maximum number of iterations is 100. Each experiment is run ten times.

Table 8: Comparison of mean fitness with other methods

Dataset	ISMO	WOA	GWO	SCA	KNN
Avila	**98.3711**	82.3762	83.1227	82.5407	72.4228
Fertility	**98.0594**	92.4700	97.4200	87.4000	79.2000
Wine	**97.0192**	94.3334	94.7210	95.0562	71.5000
Lymphography	**90.6400**	83.4237	84.2486	82.8378	72.6000
Dermatology	**99.2603**	97.8879	98.3962	98.4916	85.2500
Ionosphere	**99.3150**	90.4610	91.0776	90.9091	88.0000
Sonar	**92.2088**	80.6644	82.6801	81.4423	80.1429
Mean	**96.4106**	88.8024	90.2380	88.3825	78.4451

Table 9: Comparison of average classification accuracy with other methods

Dataset	ISMO	WOA	GWO	SCA	KNN
Avila	**98.6577**	82.5618	83.3461	82.5407	73.1544
Fertility	**98.5000**	93.0000	98.0000	87.4000	80.0000
Wine	**97.2222**	94.7191	95.0562	95.0562	72.2222
Lymphography	**91.0000**	83.7838	84.5946	82.8378	73.3333
Dermatology	**99.7222**	98.6034	98.9385	98.4916	86.1111
Ionosphere	**99.7183**	90.7955	91.5341	90.9091	88.8889
Sonar	**92.3810**	81.3462	83.3654	81.4423	80.9524
Mean	**96.7431**	89.2585	90.6907	88.3825	79.2375

Table 10: Comparison of the average number of features selected with other methods

Dataset	Features	Number of samples	ISMO	WOA	GWO	SCA	KNN
Avila	9	286	**3.2000**	3.6000	3.9000	3.7000	10.0000
Fertility	10	20867	**4.5556**	6.4444	6.0000	7.0000	10.0000
Wine	10	100	**3.3000**	5.7000	5.0000	6.1000	13.0000
Lymphography	13	178	**8.1000**	9.4000	9.0000	12.1000	18.0000
Dermatology	17	101	**15.1412**	24.0706	18.2471	23.7794	33.0000
Ionosphere	18	148	**13.7999**	14.5000	18.4000	22.0000	34.0000
Sonar	33	366	**14.9000**	52.1000	51.1000	56.7000	60.0000
Mean			**8.9995**	16.5450	15.9496	18.7685	25.4286

Table 8 shows that the mean fitness of ISMO is superior to WOA, GWO, SCA, and KNN. Notably, ISMO performs significantly better than the KNN, particularly the larger datasets. It is demonstrated that ISMO has excellent performance. Details are given in Table 8 and Fig. 4.

Figure 4: Mean fitness compared with other methods

Table 9 demonstrates that ISMO performs better than WOA, GWO, and SCA on average classification accuracy. The classification accuracy of the classifier trained by the selected feature subset is superior to that of the whole feature set. It is proven that ISMO is effective in reducing the effects of irrelevant and redundant features and improving classification performance. Details are given in Table 9 and Fig. 5.

From Table 10, ISMO achieves the minimum feature count based on the average features. Based on the general average, ISMO finds the least efficient subset of features and performs better than all the other competing algorithms. Therefore, ISMO can be used to select features efficiently, reduce the data dimensionality and simplify the learning model. Details are illustrated in Table 10 and Fig. 6.

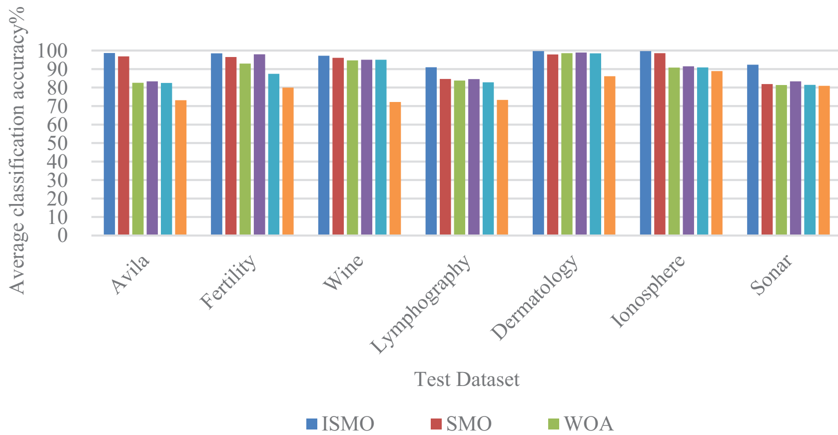

Figure 5: Average classification accuracy compared with other methods

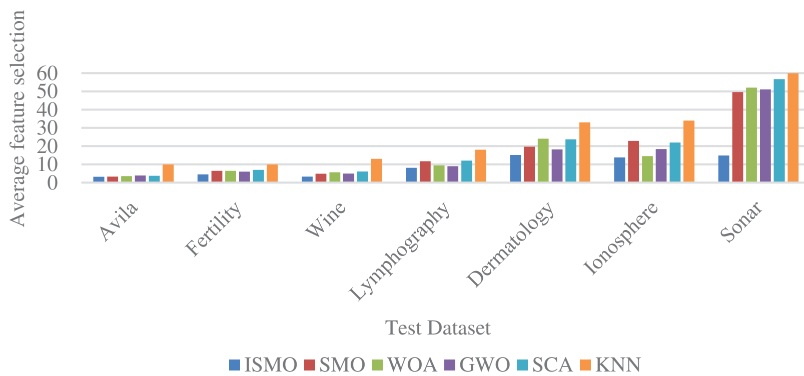

Figure 6: Comparison of average feature selection with other methods

6 Conclusions and Future Research

Aiming at the weakness of SMO, this study presents a new algorithm based on multi-strategy (ISMO). It can draw the following conclusions from the simulation test.

Firstly, inspired by the ideas of refracted opposition-based learning strategy, crisscross strategy, and Gauss-Cauchy strategy, a new position update is proposed to increase the population diversity and enhance global exploration. Moreover, according to the crisscross and Gauss-Cauchy strategies, the target value is updated with mutation to prevent it from getting into the local optimum. Furthermore, to verify the validity of ISMO, we simulated ten benchmark functions and compared them with different methods. Compared with other methods, ISMO has better performance than other methods. Finally, the Wilcoxon signed-rank test and the Friedman test prove that there is a more remarkable difference in ISMO.

Then, the ISMO is validated by the feature selection, and it is proved that ISMO is effective in choosing the best feature subset, reducing the feature dimensionality, and increasing the precision of classification.

Thus, ISMO is likely an excellent solution to numerical optimization problems. In the future, it will be considered how to improve the performance of this algorithm, even in the case of more complicated optimization problems.

Acknowledgement: The authors thank the anonymous reviewers for their valuable comments and suggestions.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare they have no conflicts of interest to report regarding the present study.

References

[1] J. Gholami, F. Mardukhi and H. M. Zawbaa, "An improved crow search algorithm for solving numerical optimization functions," *Soft Computing*, vol. 25, pp. 9441–9454, 2021.

[2] Y. Rao, D. He and L. Qu, "A probabilistic simplified sine cosine crow search algorithm for global optimization problems," *Engineering with Computers*, vol. 38, pp. 1–19, 2022.

[3] J. L. J. Pereira, G. A. Oliver, M. B. Francisco, S. S. Cunha Jr and G. F. Gomes, "Multi-objective Lichtenberg algorithm: A hybrid physics-based meta-heuristic for solving engineering problems," *Expert Systems with Applications*, vol. 187, pp. 115939, 2022.

[4] I. Fister Jr, X. S. Yang, I. Fister, J. Brest and D. Fister, "A brief review of nature-inspired algorithms for optimization," *arXiv preprint arXiv: 1307.4186*, 2013.

[5] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in Engineering Software*, vol. 95, pp. 51–67, 2016.

[6] S. Mirjalili, S. M. Mirjalili and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.

[7] S. Li, H. Chen, M. Wang, A. A. Heidari and S. Mirjalili, "Slime mould algorithm: A new method for stochastic optimization," *Future Generation Computer Systems*, vol. 111, pp. 300–323, 2020.

[8] J. Xue and B. Shen, "A novel swarm intelligence optimization approach: Sparrow search algorithm," *Systems Science & Control Engineering*, vol. 8, no. 1, pp. 22–34, 2020.

[9] M. Khishe and M. R. Mosavi, "Chimp optimization algorithm," *Expert Systems with Applications*, vol. 149, pp. 113338, 2020.

[10] J. C. Bansal, H. Sharma, S. S. Jadon and M. Clerc, "Spider monkey optimization algorithm for numerical optimization," *Memetic Computing*, vol. 6, no. 1, pp. 31–47, 2014.

[11] J. H. Holland, "Genetic algorithms," *Scientific American*, vol. 267, no. 1, pp. 66–73, 1992.

[12] R. Storn and K. Price, "Differential evolution a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.

[13] S. Kirkpatrick, C. D. Gelatt and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.

[14] R. A. Formato, "Central force optimization: A new metaheuristic with applications in applied electromagnetics," *Prog Electromagnet Res*, vol. 77, pp. 425–491, 2007.

[15] A. W. Mohamed, A. A. Hadi and A. K. Mohamed, "Gaining-sharing knowledge based algorithm for solving optimization problems: A novel nature-inspired algorithm," *International Journal of Machine Learning and Cybernetics*, vol. 11, no. 7, pp. 1501–1529, 2020.

[16] Y. Xu, H. Chen, A. A. Heidari, J. Luo, Q. Zhang *et al.,* "An efficient chaotic mutative moth-flame-inspired optimizer for global optimization tasks," *Expert Systems with Applications*, vol. 129, pp. 135–155, 2019.

[17] P. Agrawal, T. Ganesh and A. W. Mohamed, "Chaotic gaining sharing knowledge-based optimization algorithm: An improved metaheuristic algorithm for feature selection," *Soft Computing*, vol. 25, no. 14, pp. 9505–9528, 2021.

[18] K. Hussain, N. Neggaz, W. Zhu and E. H. Houssein, "An efficient hybrid sine-cosine Harris hawks optimization for low and high-dimensional feature selection," *Expert Systems with Applications*, vol. 176, pp. 114778, 2021.

[19] G. Hu, B. Du, X. Wang and G. Wei, "An enhanced black widow optimization algorithm for feature selection," *Knowledge-Based Systems*, vol. 235, pp. 107638, 2022.

[20] A. Sharma, A. Sharma, B. K. Panigrahi, D. Kiran and R. Kumar, "Ageist spider monkey optimization algorithm," *Swarm and Evolutionary Computation*, vol. 28, pp. 58–77, 2016.

[21] P. Kalpana, S. Nagendra Prabhu, V. Polepally and J. R. DB, "Exponentially-spider monkey optimization based allocation of resource in cloud," *International Journal of Intelligent Systems*, vol. 37, no. 3, pp. 2521–2542, 2022.

[22] R. Menon, A. Kulkarni and D. Singh, "Hybrid multi-objective optimization algorithm using Taylor series model and spider monkey optimization," *International Journal for Numerical Methods in Engineering*, vol. 122, no. 10, pp. 2478–2497, 2021.

[23] K. Gupta, K. Deep and J. C. Bansal, "Improving the local search ability of spider monkey optimization algorithm using quadratic approximation for unconstrained optimization," *Computational Intelligence*, vol. 33, no. 2, pp. 210–240, 2017.

[24] J. Mumtaz, Z. Guan, L. Yue, L. Zhang and C. He, "Hybrid spider monkey optimization algorithm for multi-level planning and scheduling problems of assembly lines," *International Journal of Production Research*, vol. 58, no. 20, pp. 6252–6267, 2020.

[25] H. R. Tizhoosh, "Opposition-based learning: A new scheme for machine intelligence," in *Int. Conf. on Computational Intelligence for Modelling, Control and Automation and Int. Conf. on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'06)*, Vienna, Austria, IEEE, pp. 695–701, 2005.

[26] T. K. Sharma and M. Pant, "Opposition-based learning ingrained shuffled frog-leaping algorithm," *Journal of Computational Science*, vol. 21, pp. 307–315, 2017.

[27] X. Yu, W. Xu and C. L. Li, "Opposition-based learning grey wolf optimizer for global optimization," *Knowledge-Based Systems*, vol. 226, pp. 107139, 2021.

[28] A. A. Ewees, M. Abd Elaziz and D. Oliva, "A new multi-objective optimization algorithm combined with opposition-based learning," *Expert Systems with Applications*, vol. 165, pp. 113844, 2021.

[29] Z. Wang, H. Ding, Z. Yang, B. Li, Z. Guan *et al.,* "Rank-driven salp swarm algorithm with orthogonal opposition-based learning for global optimization," *Applied Intelligence*, vol. 52, no. 7, pp. 7922–7964, 2021.

[30] F. Zhao, L. Zhang, Y. Zhang, W. Ma, C. Zhang *et al.,* "An improved water wave optimisation algorithm enhanced by CMA-ES and opposition-based learning," *Connection Science*, vol. 32, no. 2, pp. 132–161, 2020.

[31] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *1998 IEEE Int. Conf. on Evolutionary Computation Proc.*, Anchorage, AK, USA, IEEE, pp. 69–73, 1998.

[32] A. B. Meng, Y. C. Chen, H. Yin and S. Z. Chen, "Crisscross optimization algorithm and its application," *Knowledge-Based Systems*, vol. 67, pp. 218–229, 2014.

[33] W. C. Wang, L. Xu, K. Chau and D. M. Xu, "Yin-Yang firefly algorithm based on dimensionally Cauchy mutation," *Expert Systems with Applications*, vol. 150, pp. 113216, 2020.

[34] J. Derrac, S. García, D. Molina and 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*, vol. 1, no. 1, pp. 3–18, 2011.