Applied Soft Computing Journal 96 (2020) 106609

Contents lists available at ScienceDirect

Applied Soft Computing Journal

journal homepage: www.elsevier.com/locate/asoc

A collaborative LSHADE algorithm with comprehensive learning mechanism

Fuqing Zhao [a,*], Lexi Zhao [a], Ling Wang [b], Houbin Song [a]

[a] School of Computer and Communication Technology, Lanzhou University of Technology, Lanzhou 730050, China
[b] Department of Automation, Tsinghua University, Beijing, 100084, China

ARTICLE INFO

Article history:
Received 18 February 2020
Received in revised form 9 July 2020
Accepted 30 July 2020
Available online 2 September 2020

Keywords:
Differential evolution
Collaborative scheme
Comprehensive learning mechanism
Competitive reward mechanism
Dimensional reset strategy

ABSTRACT

In this study, a novel L-SHADE variant with collaborative scheme and comprehensive learning mechanism, named LSHADE-CLM, was proposed to improve the exploration and exploitation capabilities of the L-SHADE algorithm. In LSHADE-CLM, a novel cooperative mutation mechanism including "$DE/current-to-pbetter/r$" and "$DE/current-to-pbest-w/1$" is proposed in the mutation operation. In the "$DE/current-to-pbetter/r$" strategy with comprehensive learning mechanism, the population covariance matrix is utilized to generate candidate solutions and guide the search direction. Meanwhile, a competitive reward mechanism is implemented to control the mutation factor F to generate a trial vector for the cooperative mechanism. Moreover, the dimensional reset strategy is applied to enhance the diversity of the population at the dimensional level when stagnation is identified at certain dimension. The proposed LSHADE-CLM is tested on the CEC2017 benchmark functions and compared with the other four state-of-the-art variants of L-SHADE. The experimental results demonstrated that the efficiency and effectiveness of the LSHADE-CLM algorithm for the non-separable optimization problem.

1. Introduction

During the past decades, the optimization problem and its application became the research focus in various fields such as science, engineering and statistics [1]. As a result, the optimization algorithm played an important role in various non-separable optimization problems [2]. The non-separable optimization problem is defined by the objective function of $f(x) = f(x_1, x_2, \ldots, x_D)$ and attempts to find vector x_i, which optimizes (minimizes or maximizes) $f(x_i)$. In $f(x)$, D is the dimensionality of the problem, and the search space is defined by the lower and upper bounds, x_{low} and x_{up}.

Differential evolution (DE) is an evolutionary and population-based optimization algorithm, which was developed by Price [3] and applied in fields of engineering, statistics and finance. A wide variety of practical problems have objective functions, especially for non-differentiable, non-continuous, nonlinear functions with unknown, amounts of multidimensional or stochastic noise, which are unsolvable by conventional analytical method [4]. However, DE has advantages to solve the above-mentioned problems. Meanwhile, similar to other related evolution algorithms, DE easily falls into the local optimum, which means that DE shows premature convergence into local optima. Furthermore, the performance of the DE is sensitive to the control parameter settings and mutation strategy [5]. A suitable control parameter and mutation strategy for the specific problem are usually time-consuming [6].

A series of improvements in DE were proposed to overcome the drawbacks in the past two decades [7]. In view of the diversities of benchmark and practical problems, the collaborative strategies algorithm with different parameter settings showed a desirable performance than the algorithm with single strategy. A new algorithm called jDE was designed by Brest, et al. [8], which introduced a self-adapting control parameters strategy in DE to adjust the control parameters. A composite DE (CoDE) algorithm was proposed by Wang, et al. [9]. In CoDE, multiple mutation strategies were used to solve different problems. Then, a success-history-based parameter adaptation for DE (SHADE), with a success-historical memory for mutation factor F and crossover factor Cr, was proposed by Tanabe and Fukunaga [10]. A new algorithm called LSHADE was proposed by Tanabe and Fukunaga [11] to improve the performance of SHADE. In LSHADE, a population size reduction strategy was introduced to SHADE. Then, a new hybrid algorithm called LSHADE-CMAES was combined by Mohamed and Hadi [12], which combined LSHADE with a semi-parameter adaptation hybrid with covariance matrix adaptation evolution strategy (CMA-ES). However, the diversity of populations is increased by employing other algorithms or

* Corresponding author.
E-mail addresses: Fzhao2000@hotmail.com (F. Zhao), 360818938@qq.com (L. Zhao), wangling@tsinghua.edu.cn (L. Wang), 523712418@qq.com (H. Song).

https://doi.org/10.1016/j.asoc.2020.106609
1568-4946/

mutation operators as a mutation operator in DE, such as algorithmic linking [13]. Meanwhile, the different variants of DE exhibited different competences in solving various optimization problems [14]. The idea of mixing collaborative mutation strategies and the multiple parameter adaption mechanism have attracted a wide variety of attention from researchers and certain related concepts have been proposed, such as hybrid algorithms [15], an ensemble algorithm [16], a multiple-operator-algorithm [17] and algorithm portfolios [18]. In addition, it is a challenging yet important task to combine real-world for optimization problems with DE [19].

However, there are still various factors have not been taken into adequate account in the existing methods. For example, although the integration of multi-mutation strategies has become a research hotspot, it is difficult to select the appropriate strategy for individual mutations from multiple mutation strategies. Meanwhile, it has been demonstrated that DE tends to fall into local optima and lose the diversity of the population in the late evolution process [20]. The factors can be summarized as follows: First, although a different self-adapt parameter strategy and a different mutation strategy were applied in the algorithms mentioned above, the single mutation strategy and single self-adapt parameter strategy were used in most algorithms. Second, if a similar sample occurred in population, the selective pressure was missed in the mutation strategy. Third, the algorithm was originally tested in normal conditions. However, from a multidimensional level perspective, the possibility of losing the diversity of the population and dropping to stagnation early occurred in certain dimensions [21]. All issues discussed above have a great impact on the performance of the optimization issue as the dimension increased. Based on the above analysis, a novel approach was systematically developed in this paper.

In this study, an ensemble LSHADE algorithm with learning mechanisms and a dimensional reset strategy, called LSHADE-CLM, is proposed to improve the performance of LSHADE. In LSHADE-CLM, the dimensional reset strategy was designed to prevent the LSHADE-CLM from premature convergence on the local optima, when the LSHADE-CLM was in stagnation and to guarantee the diversity of the population at various dimensional levels. Meanwhile, a novel mutation scheme named "$DE/current-to-pbetter/r$" is proposed, which take advantage of both DE and CMA-ES. In "$DE/current-to-pbetter/r$" scheme, a rank-based selective pressure method was applied in mutation strategies to ensure that the selected pressure in the best and worst individuals was not identical, and the Gaussian distribution in covariance matrix adaptation evolution strategy is used to guide the search direction. Furthermore, a comprehensive learning strategy was inserted into the two mutation strategies to generate the mutation parameter F with the iteration of the LSHADE-CLM increased. Therefore, all the factors that affect the performance of DE were considered in the LSHADE-CLM. The main contributions of this study were summarized as follows.

- A novel mutation strategy with comprehensive learning mechanism named "$DE/current-to-pbetter/r$" scheme is proposed.
- A cooperative mutation mechanism, which includes "$DE/current-to-pbetter/r$" and "$DE/current-to-pbest-w/1$" are introduced into the LSHADE-CLM and a competitive reward mechanism is used to control the mutation factor F.
- A dimensional reset strategy is designed to improve the diversity of the population at the dimensional level and to help the LSHADE-CLM escape local optima or stagnation in the selection operation.
- Proposed LSHADE-CLM has been tested for scheduling problem and non-separable optimization problem.

The paper was structured as follows. The basic DE and its variants are introduced in Section 2. Section 3 introduces the details of the proposed LSHADE-CLM algorithm, gives pseudo-code and the convergence property of the LSHADE-CLM algorithm is analyzed. The experiment and comparisons are presented in Section 4. The blocking flowshop scheduling problem are solved by using the proposed LSHADE-CLM in Section 5. Finally, Section 6 concludes the paper, and some future work is suggested.

2. Differential evolution and its variants

To simplify formulation exposition, the notations applied in this paper are shown as follows.

g	The number of current generations
$x_{i,g}$	The target vector in a generation g
$u_{i,j}$	The trial vector in a generation g
$v_{i,g}$	The mutant vector in a generation g
D	The dimension of the search space
N_p	The size of the population
$x_{ri,g}$	ri an exclusive and randomly selected number from $[1, N_p]$, for different i
$x_{b,g}$	The best individual in the population in a generation g
$x_{r,g}$	a randomly selected number from the top $N*p$ in a generation g
$nfes$	The number of evolutions
F	The mutation parameter
Cr	The crossover rate, $Cr \in [0, 1]$
$freq$	The frequency of the sinusoidal function
H	The length of historical memory
S_{Cr}	The historical memory of Cr
S_F	The historical memory of F
S_{freq}	The historical memory of $freq$

2.1. Differential evolution

DE is a population-based optimization algorithm introduced by Price [3]. It is one of the fairly effective and popular algorithms for solving the complex numerical optimization problems and the high-dimensional problems. Meanwhile, the development of DE and its variants have been applied in various real-world problems, such as the flow shop scheduling problems [22], no-wait flow shop scheduling problems [23]. At each generation, the three operators, named the mutation operation, the crossover operation and the selection operation, were executed on the target vector [24]. First, the mutant vector was generated by the mutation operation. Second, a trial vector was produced by a mutant vector and its associated target vector in the crossover operation. Third, a desirable vector from the trial vector or from the target vector was selected to access the next generation in the selection operation.

DE and its variants have become one of the popular optimization algorithms over the last twenty years [25]. Various versions of DE were proposed to solve the non-separable optimization problem [26]. However, a wide variety of discrepant conclusions have been drawn in earlier studies of DE. The main reason was that the optimal parameters were different as required by DE for solving different problems [27].

The mutation strategy plays an important role in DE [28]. Various mutation strategies for generating the mutant vector are described as follows.

"$DE/current-to-pbest/1$"

$$v_{i,g} = x_{i,g} + F(x_{pb,g} - x_{i,g}) + F(x_{r1,g} - x_{r2,g}) \tag{1}$$

Table 1
State-of-the-art variants of DE.

Algorithm	Simple description
jSO	jSO is an improved variant of the iL-SHADE algorithm, with a new weighted factor F_w in mutation strategy.
MPEDE	In MPEDE, a multi-population-based framework is proposed to realize the ensemble of multiple DE variants (which in MPEDE was JADE, EPSDE and CoDE).
LSHADE-cnEpsin	In LSHADE-cnEpsin, there are two major modifications, ensemble of sinusoidal approaches based on performance adaptation and covariance matrix learning for the crossover operator.
LSHADE-SPACMA	LSHADE-SPACMA introduced a hybridization framework between a modified version of CMA-ES and LSHADE-SPA. Meanwhile, it uses a new semi-parameter adaptation approach based on randomization.
ELSHADE-SPACMA	This algorithm enhanced the performance of LSHADE-SPACMA by integrating another directed mutation strategy within the hybridization framework.
LSHADE-RSP	This algorithm is a modification of the LSHADE algorithm, and has a rank-based selective pressure strategy.

$$\text{“}DE/current-to-pbest-w/1\text{"}\\ v_{i,g}=x_{i,g}+F_w(x_{pb,g}-x_{i,g})+F(x_{r1,g}-x_{r2,g}) \tag{2}$$

$$\text{“}DE/current-to-pbest/r\text{"}\\ v_{i,g}=x_{i,g}+F_w(x_{pb,g}-x_{i,g})+F(x_{pr1,g}-x_{pr2,g}) \tag{3}$$

After the mutation operation, a trial vector $u_{i,g}$ is generated by the crossover operation between $x_{i,g}$ and $v_{i,g}$. The $u_{i,g}$ is defined as follows.

$$u_{i,g}^j=\begin{cases} v_{i,g}^j & if \quad rand_j(0,1)\le Cr \quad or \quad j=j_{rand} \\ x_{i,g}^j & \qquad otherwise \end{cases} \tag{4}$$

The desirable vector between $x_{i,g}$ (the target vector) and $u_{i,g}$ (the trial vector) was guaranteed by the selection operation to access the next generation after the crossover operation. The selection operation is described as follows.

$$x_{i,g+1}^j=\begin{cases} x_{i,g}^j & iff(x_{i,g}^j)\le f(u_{i,g}^j) \\ u_{i,g}^j & \quad otherwise \end{cases} \tag{5}$$

2.2. L-SHADE algorithm

L-SHADE stands for Linear Population Size Reduction (LPSR) Success History-based Adaptive (SHA) Differential Evolution and is an enhanced version of the SHADE algorithm, based on one of the adaptive DE modifications JADE [29].

The $DE/current-to-pbest/1$ mutation scheme is used in the original LSHADE algorithm. In Eq. (7), the random index $r2$ is uniformly selected from the joint set of the population and the external archive. The only difference is that the jth variable of the trial vector t is the same as the jth variable of the mutant vector v if a randomly generated number in the range (0, 1) is smaller than the crossover rate Cr or if j is equal to $jrand$, where $jrand$ is a randomly chosen index ($jrand = 1, \ldots, D$).

The LPSR scheme is used in the LSHADE algorithm, which greatly improved the performance of the algorithm. The LPSR scheme decreases the number of individuals in the population by deleting least fitting ones at every generation. The new population size in next generation is calculated by the following formula:

$$N_{g+1}=\frac{round(N_{\min}-N_{\max})}{nfes_{\max}}nfes+N_{\max} \tag{6}$$

where $N_{min}=4$, $N_{\max}$ is the initial population size, and $nfes_{\max}$ is the maximal number of function evaluations.

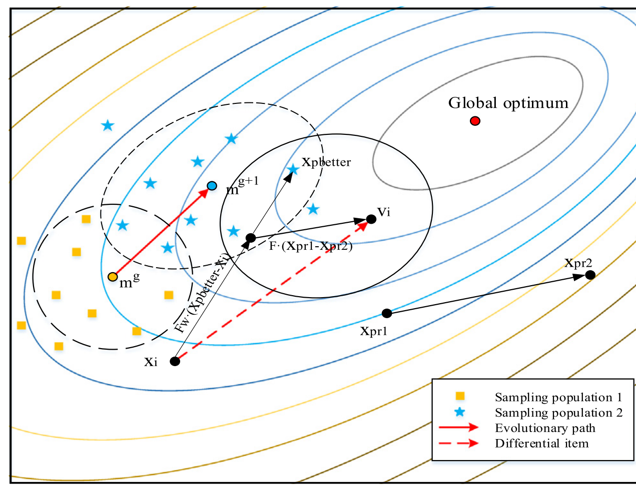


Fig. 1. Illustration of $DE/current-to-pbetter/r$ scheme.

2.3. State-of-the-art variants of DE

DE has attracted attention with its excellent performance at solving non-separable optimization problems. However, the performance of DE is immensely reliant on the parameter settings. The sensitivity of DE to the mutation parameter F and crossover parameter Cr led to a wide variety of attention and concern from researchers [30].

In the standard DE, parameters F and Cr are set to a range of certain fixed values. Thus, a new parameter adaptation method, named SHADE, was proposed by Tanabe and Fukunaga [10]. In SHADE, the initial F was generated by the Cauchy distribution, and the initial Cr was generated by the normal distribution. F and Cr, which were generated a desirable value on the current generation, were selected to participate in the mutation strategy and crossover operation in the next generation.

Meanwhile, a suitable population size and a satisfactory mutation strategy play an important role on setting the parameters. The population size was set from $5D$ to $10D$ in the standard DE algorithm and its variants [31]. The L-SHADE, proposed by Tanabe and Fukunaga [11], introduced a linear population size reduction method to adjust the population size in SHADE. L-SHADE with a semi-parameter adaptation hybrid with CMA-ES (LSHADE-CMAES) was proposed by Mohamed and Hadi [12] to improve the optimization performance of LSHADE. In LSHADE-CMAES, an alternative adaptation approach for the selection of control

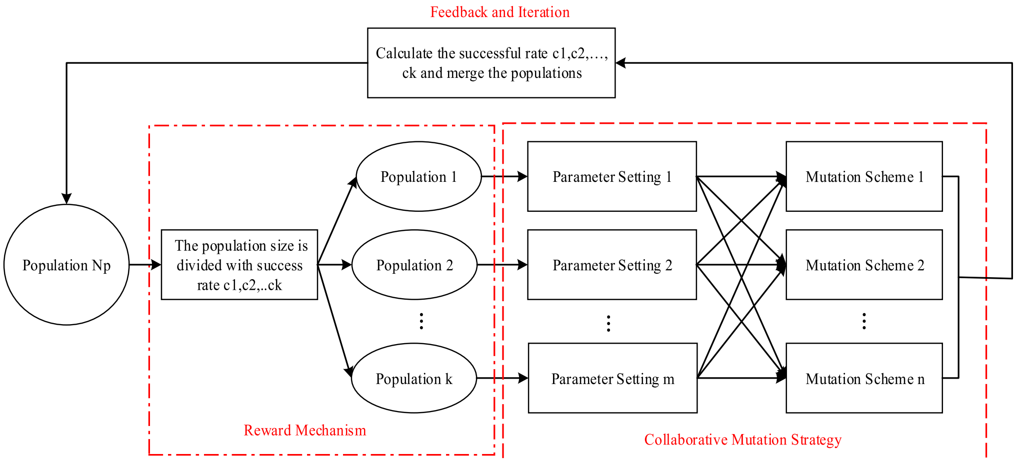


Fig. 2. Illustration of collaborative mutation strategy.

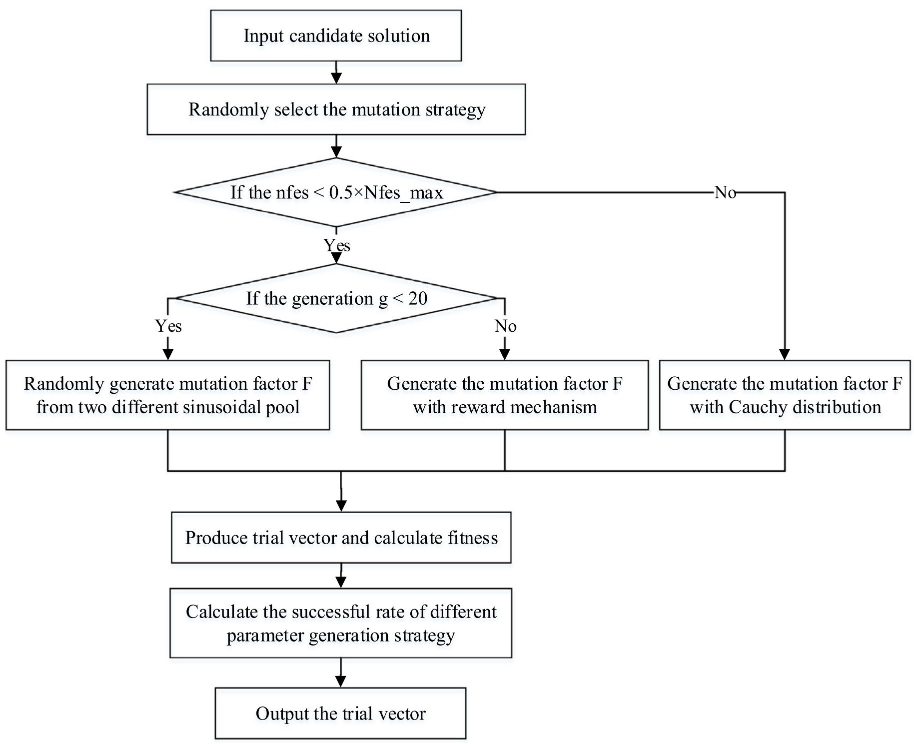


Fig. 3. Procedure of the mutation factor F.

parameters was proposed. Then, the jSO algorithm was proposed by Brest, et al. [32], which introduced a new mutation strategy in L-SHADE. With the further development of L-SHADE, Stanovov and Akhmedova [33] proposed LSHADE-RSP and introduced the rank-based selective pressure strategy in the mutation strategy to ensure selective pressure in the population. Meanwhile, jSO and LSHADE-RSP were the winning algorithms for the bound-constrained continuous optimization problems in CEC2017 and CEC2018, respectively.

Furthermore, there is no single algorithm suitable for all potential optimization problems, which was proven by the no-free-lunch (NFL) theorems. However, the ensemble methods, which promised to take advantage of previous knowledge of DE and its variants to create a new variant, were effective in designing high-quality DE algorithms. Therefore, it was necessary to research the different categorization of DE and its variants. Various state-of-the-art variants of DE are described in Table 1.

3. The proposed LSHADE-CLM algorithm

The LSHADE algorithm and its variants are a potential framework to obtain a desirable performance for DE-based optimization techniques. However, the premature convergence of population and the stagnation of population were not addressed in DE and its variants. In this study, a new method called the LSHADE-CLM algorithm is proposed to the non-separable problem. Different with the canonical LSHADE, a novel collaborative mutation strategy is used in LSHADE-CLM and a new mutation scheme named "$DE/current - to - pbest/r$" is proposed. Meanwhile, a reward mechanism is introduced into the parameter adjustment. Moreover, the diversity and distribution of the population is reconsidered from the dimensional level.

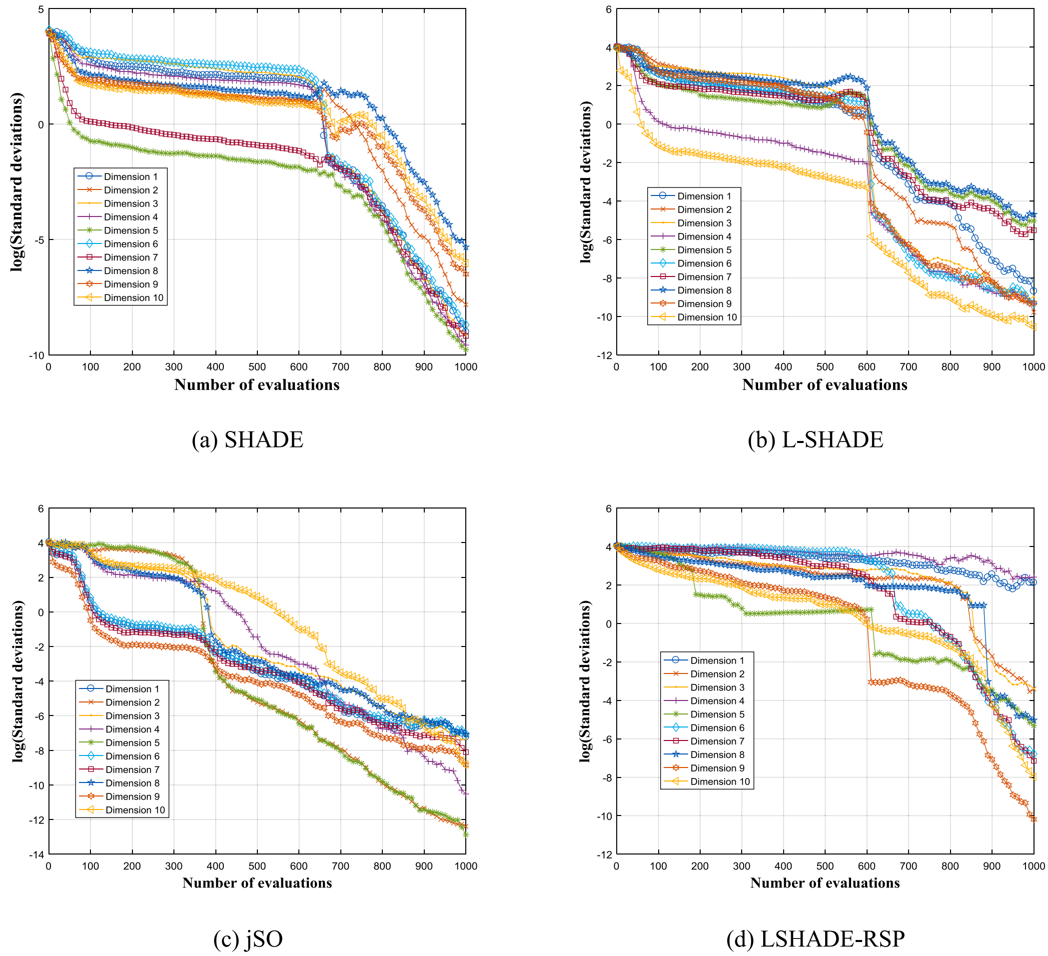


Fig. 4. Log of standard deviations on each dimension $D = 10$.

3.1. The collaborative mutation strategy

A new mutation strategy called "$DE/current - to - pbest/r$" was proposed by Stanovov, et al. [34] to avoid the selective pressure lost in suitable individuals. The collaborative mutation strategy with parameter learning mechanism was a useful way to introduce the select pressure into mutation strategy to improve the ability of DE in solving non-separable optimization problems. However, the NFL theorem was proved by Wolpert and Macready [35], and it also proven that when the search algorithms were applied to all potential optimization problems, which have the same performance on average. In NFL theory, there was no general optimization algorithm better than other optimization algorithms in different fields. Meanwhile, compared with algorithms with no learning mechanism or a single strategy algorithm, the algorithms with learning mechanisms or multiple strategies had a desirable ability to solve different optimization problems by using different strategies or learning mechanisms [36]. For instance, Qin, et al. [37] proposed the SaDE algorithm, which introduced a self-adaptive strategy when setting parameters to adaptation parameters. Awad, et al. [38] proposed a new algorithm called LSHADE-EpSin, which was an ensemble sinusoidal parameter adaptation incorporated with LSHADE. In LSHADE-EpSin and SaDE, learning mechanisms and parameter adaption strategies were introduced to enhance the performance of DE.

3.1.1. A novel mutation strategy with comprehensive learning mechanisms

The covariance matrix adaptation evolution strategy is a stochastic algorithm that learns a covariance matrix from the successful experience in the search process. The CMA-ES is a suitable algorithm for solving non-separable because of its rotational invariance. However, the standard CMA-ES are designed for solving unimodal functions and is easily trapped into local optimal solution.

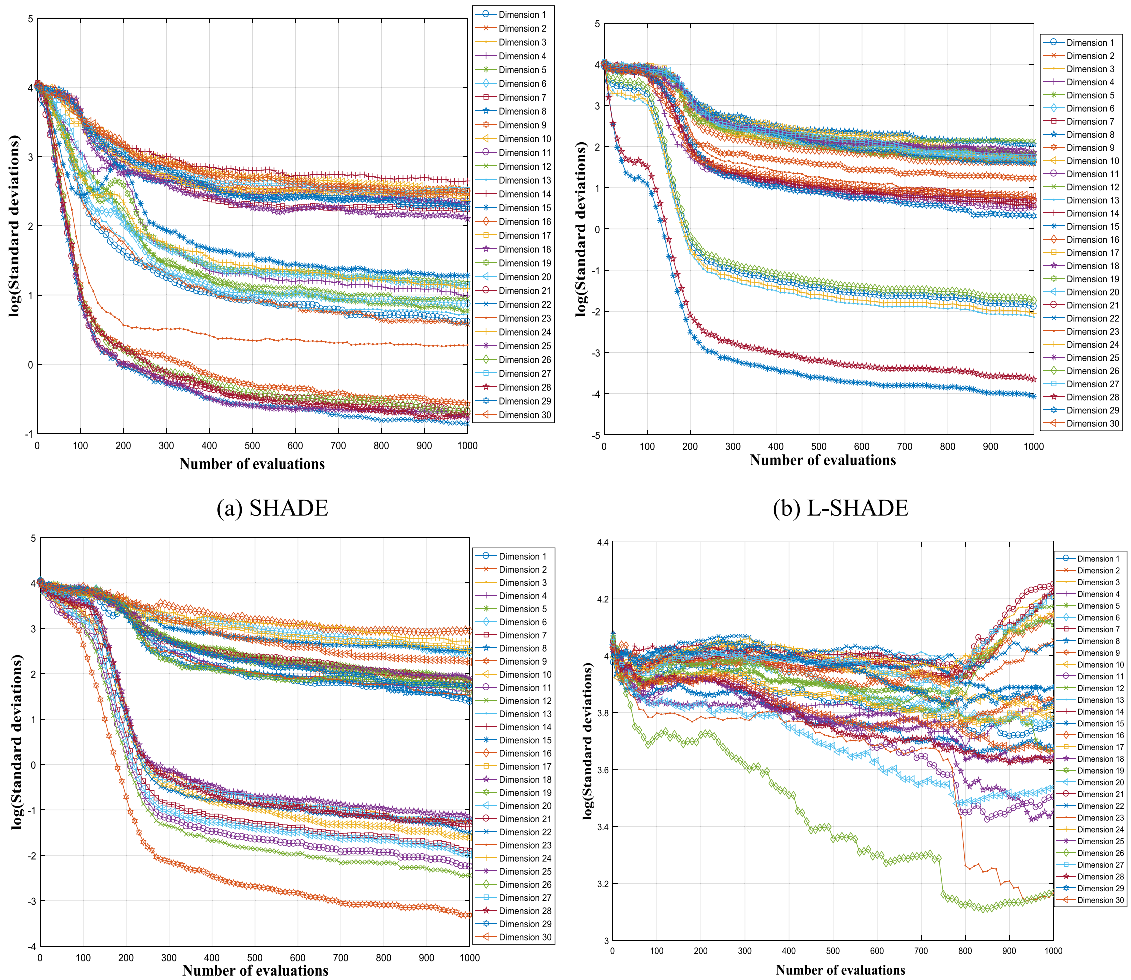
However, the disadvantages mentioned above are solved by a hybrid of DE and CMA-ES. In LSHADE-CLM, a novel mutation strategy named "$DE/current - to - pbetter/r$" is implemented, which take advantage of both DE and CMA-ES. DE is utilized to maintain diversity while CMA-ES performs local search. In "$DE/current - to - pbetter/r$" mutation strategy, a new random variable $x_{pbetter,g}$ is proposed to replace the $x_{pbest,g}$ in Eq. (3). The details of the "$DE/current - to - pbetter/r$" is described as follows.

$$v_{i,g} = x_{i,g} + F_w(x_{pbetter,g} - x_{i,g}) + F(x_{pr1,g} - x_{pr2,g}) \tag{7}$$

$$x_{pbetter,g} \sim N(m^g, C^g) \tag{8}$$

where $x_{i,g}$ is the jth coordinate of the ith individual $x_{i,g}$. $r1$, $r2$ are randomly selected from $[1, Np]$, and i, $r1$, $r2$ are exclusive to each other. "$\sim$" denotes the same distribution of $x_{pbetter,g}$ and the Gaussian distribution $N(m^g, C^g)$. In other words, the guide vector $x_{pbetter,g}$ is generated for each individual $x_{i,g}$ by sampling the Gaussian distribution $N(m^g, C^g)$. The Fig. 1 shows the illustration of "$DE/current - to - pbetter/r$" scheme.

In Eq. (8), the m^g is the mean value and C^g is the covariance matrix. After evaluating the fitness of all the individuals, the top



(a) SHADE (b) L-SHADE

(c) jSO (d) LSHADE-RSP

Fig. 5. Log of standard deviations on each dimension $D = 30$.

$100p\%$ individuals are used to update the mean values for the next generation m^{g+1} and the covariance matrix of the population distribution C^{g+1}. Denote $\mu = [p \times N_p]$, the updating is listed as follows.

$$m^{g+1} = \sum_{i=1}^{u} \omega_i x_{i:Np}^{g+1} \tag{9}$$

$$C^{g+1} = (1 - C_\mu)C^g + C_\mu \sum_{i=1}^{u} \omega_i (x_{i:Np}^{g+1} - m^g)(x_{i:Np}^{g+1} - m^g)^T \tag{10}$$

where c_μ is a controlling factor in Eq. (10), μ is the number of selected individuals, $x_{i:N_p}^{g+1}$ is the ith best individual of the current population and $_i$ is positive weights. The Individuals with desirable fitness values are used to update the Gaussian distribution. In this study, the settings of the parameters p, ω_i and c_μ are the identical as the standard CMA-ES, various details are found in standard CMA-ES [39].

The Gaussian distribution updating in "$DE/current - to - pbetter/r$" is described in Fig. 1. The probability density contours of Gaussian distribution at the generation g and $g + 1$ are described as the dashed line and solid line, respectively. m^g and m^{g+1} are the corresponding mean values. The old distribution (dashed line) is updated through a new one (solid line), and the distribution is able to gradually adapt to the problem landscape. In the other words, the individuals with a desirable fitness have larger probabilities to be generated from sampling this distribution. Different with CMA-ES and DE, the above update method is enhanced in two aspects:

- In classic DE and its variants, the mutation factor F is set to be equal to F_w. However, the F and F_w are separately controlled in "$DE/current - to - pbetter/r$" strategy for the following reason: at the late run of algorithm, set F_w to be a large value to speed up convergence. Meanwhile, set F to be a small value to decrease the population diversity. At the beginning of the run, set F_w to be a small value and F a large value.
- Different with standard CMA-ES, in "$DE/current - to - pbetter/r$" strategy, the search step is controlled by mutation factors F and F_w, and F is generated by learning from the previous experience of producing promising solutions (as discussed later in Section 3.1.2).

3.1.2. Collaborative mutation strategy and parameters control

In the majority of DE variants, only one mutation strategy and only one parameter setting are employed at each generation to generate target vector. Therefore, the search ability of the algorithms is limited. In LSHADE-CLM, a collaborative mutation strategy is proposed, including two different mutation strategy "$DE/current - to - pbetter/r$" and "$DE/current - to - pbest - w/1$". The primary idea of collaborative mutation strategy is to

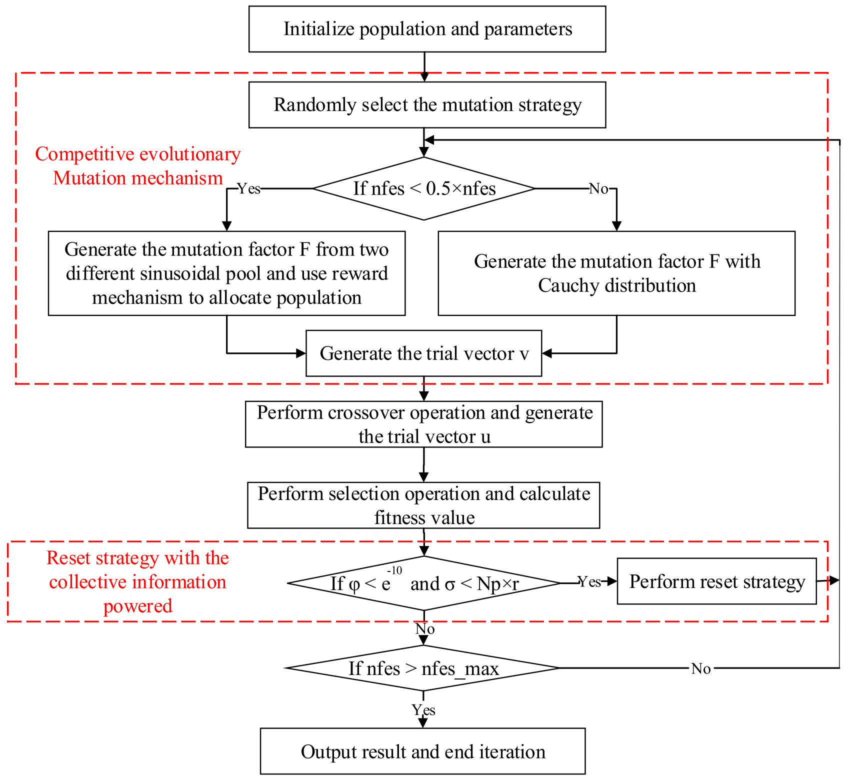


Fig. 6. Flow chart of proposed LSHADE-CLM algorithm.

Table 2
Parameter settings.

Algorithms	Authors	Parameter settings
jSO	Brest, et al. [32]	$N^{\mathrm{init}} = 25 \log D * \sqrt{D}, H = 5, \mu F = 0.5, \mu Cr = 0.8, N^{\mathrm{min}} = 4$
LSHADE-RSP	Stanovov and Akhmedova [33]	$N^{init} = 75 \cdot D^{\frac{2}{3}}, H = 5, \mu F = 0.3, \mu Cr = 0.8, \quad N^{\mathrm{min}} = 4, k = 3$
LSHADE-cnEpsin	Awad, et al. [40]	$N^{init} = 18 \cdot D, H = 5, \mu F = 0.5, \mu Cr = 0.5, N^{\mathrm{min}} = 4,\ \mathrm{freq} = 0.5$
ELSHADE-SPACMA	Hadi, et al. [41]	$N^{\mathrm{init}} = 18 \cdot D, H = 5, \mu F = 0.5, \mu Cr = 0.5, N^{\mathrm{min}} = 4$
EBLSHADE	Mohamed, et al. [42]	$N^{\mathrm{init}} = 18 \cdot D, H = 5, \Delta_m \epsilon [0.2, 0.8], \mu F = 0.5, \mu Cr = 0.5, N^{\mathrm{min}} = 4$

Table 3
Definitions of basic functions.

Function name	Definitions	f_{min}
Bent Cigar function	$f_1(x) = x_1^2 + 10^6 \sum_{(i=2)}^{D} x_i^2$	100
Discus function	$f_2(x) = 10^6 x_1^2 + \sum_{(i=2)}^{D} x_i^2$	200
Weierstrass function	$f_3(x) = \sum_{i=1}^{D} \sum_{k=0}^{k\max} [a^k \cos(2\pi b^k (x_i + 0.5))] - D \sum_{k=0}^{k\max} [a^k \cos(2\pi b^k \times 0.5)]$	300
Modified Schwefel's function	$f_4(x) = 418.9829 \times D - \sum_{i=1}^{D} g\,(z_i)$	400
Katsuura function	$f_5(x) = \frac{10}{D^2} \prod_{i=1}^{D} \left(1 + i \sum_{j=1}^{32} \frac{\left\lvert 2^j x_i - \mathrm{round}\left(2^j x_i\right)\right\rvert}{2^j}\right)^{\frac{10}{D^{1.2}}} - \frac{10}{D^2}$	1200
HappyCat function	$f_6(x) = \left\lvert \sum_{i=1}^{D} x_i^2 - D \right\rvert^{1/4} + \left(0.5 \sum_{i=1}^{D} x_i^2 + \sum_{i=1}^{D} x_i\right) / D + 0.5$	1200
HGBat function	$f_7(x) = \left\lvert \left(\sum_{i=1}^{D} x_i^2\right) - \left(\sum_{i=1}^{D} x_i\right) \right\rvert^{1/2} + \left(0.5 \sum_{i=1}^{D} x_i^2 + \sum_{i=1}^{D} x_i\right) / D + 0.5$	1500
Expanded Griewank's plus Rosenbrocks function	$f_8(x) = f_{11}(f_{10}(x_1, x_2)) + f_{11}(f_{10}(x_2, x_3)) + \ldots + f_{11}(f_{10}(x_{(D-1)}, x_D)) + f_{11}(f_{10}(x_D, x_1))$	800
Expanded Scaffer's F6 function	$f_9(x) = g(x_1, x_2) + g(x_2, x_3) + \ldots + g(x_{(D-1)}, x_D) + g(x_D, x_1)$	1100
Rosenbrock's function	$f_{10}(x) = \sum_{i=1}^{D-1} \left(100\left(x_i^2 - x_{i+1}\right)^2 + (x_i - 1)^2\right)$	800
Greiwank's function	$f_{11}(x) = \sum_{i=1}^{D} \frac{x_i^2}{4000} - \prod_{i=1}^{D} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	1100
Rastrigin's function	$f_{12}(x) = \sum_{i=1}^{D} \left(x_i^2 - 10 \cos(2\pi x_i) + 10\right)$	1000
High Conditioned Elliptic	$f_{13}(x) = \sum_{i=1}^{D} \left(10^6\right)^{\frac{i-1}{D-1}} x_i^2$	1500
Ackley's function	$f_{14}(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^{D} x_i^2}\right) - \exp\left(\frac{1}{D} \sum_{i=1}^{D} \cos(2\pi x_i)\right) + 20 + e$	1200

Table 4
Unimodal functions.

Functions	D	Search range	$f_{min} = F_i^* = F_i(x^*)$
$F_1(x) = f_1(M(x - o_1)) + F_1^*$	10, 30, 50, 100	$[-100, 100]^D$	100
$F_1(x) = f_2(M(x - o_1)) + F_2^*$	10, 30, 50, 100	$[-100, 100]^D$	200

Table 5
Simple multimodal functions.

Functions	D	Search range	$f_{min} = F_i^* = F_i(x^*)$
$F_3(x) = f_3(M((0.5(x - o_3))/100)) + F_3^*$	10, 30, 50, 100	$[-100, 100]^D$	300
$F_4(x) = f_4(M((1000(x - o_4))/100)) + F_4^*$	10, 30, 50, 100	$[-100, 100]^D$	400
$F_5(x) = f_5(M((5(x - o_5))/100)) + F_5^*$	10, 30, 50, 100	$[-100, 100]^D$	500

Table 6
Hybrid functions.

Functions	D	Search range	$f_{min} = F_i^* = F_i(x^*)$
$F_6(x) = f_6(M((5(x - o_6))/100)) + F_6^*(N = 3)$	10, 30, 50, 100	$[-100, 100]^D$	600
$F_7(x) = f_7(M((5(x - o_7))/100)) + F_7^*(N = 4)$	10, 30, 50, 100	$[-100, 100]^D$	700
$F_8(x) = f_8(M((5(x - o_8))/100) + 1) + F_8^*(N = 5)$	10, 30, 50, 100	$[-100, 100]^D$	800

Table 7
Composition functions.

Functions	D	Search range	$f_{min} = F_i^* = F_i(x^*)$
$F_9(x) = f_9(M(x - o_9) + 1) + F_9^*(N = 3)$	10, 30, 50, 100	$[-100, 100]^D$	900
Composite function 2 (N = 3)	10, 30, 50, 100	$[-100, 100]^D$	1000
Composite function 3 (N = 5)	10, 30, 50, 100	$[-100, 100]^D$	1100
Composite function 4 (N = 5)	10, 30, 50, 100	$[-100, 100]^D$	1200
Composite function 5 (N = 5)	10, 30, 50, 100	$[-100, 100]^D$	1300
Composite function 6 (N = 7)	10, 30, 50, 100	$[-100, 100]^D$	1400

randomly combine several mutation strategies with a number of control parameter settings in each generation, using the reward mechanism and historical information in population evolution to generate desirable test vectors. The collaborative mutation strategy is illustrated in Fig. 2.

In this study, two mutation strategies and two control parameter settings are choosing to constitute the collaborative mutation strategy, which include the mutation strategy candidate pool and the parameter candidate pool. The mutation strategy candidate pool is described as follows.

$$\begin{cases} using \quad Eq.\,(2) \quad if \quad rand(0,1) \le \frac{popsize_g}{popsize_g + popsize_{\max}} \\ using \quad Eq.\,(7) \quad otherwise \end{cases} \tag{11}$$

where $popsize_i$ is the current population size in gth generation and $popsize_max$ is the maximum number of population size. The details of F_w in Eqs. (2) and (7) are set as follows.

$$F_w = \begin{cases} 0.7F & if \quad nfes \le 0.2nfes_{\max} \\ 0.8F & if \quad nfes > 0.2nfes_{\max} \quad and \quad nfes \le 0.4nfes \\ 1.2F & otherwise \end{cases} \tag{12}$$

At each generation, each mutation strategies in the strategy candidate pool is used to produce a new trial vector. The control parameter settings of the trial vector are selected from the candidate pool. The parameter candidate pool consists of two parts. The first configuration is non-adaptive sinusoidal decreasing adjustment, which described in Eq. (13). In Eq. (13), g is the number of current iterations, $G_{\max}$ is the number of max iterations and $freq = 0.5$.

$$F_{i,g} = \frac{1}{2}(\sin(2\pi \times freq_i \times g + \pi) \times \frac{G_{\max} - g}{G_{\max}} + 1) \tag{13}$$

In the second configuration, an adaptive sinusoidal increasing adjustment is adjusted by Eq. (14), where $freq_i$ is created by the

Table 8
ANOVA results over calibrating the parameters of LSHADE-CLM.

Source	Sum of squares	Degrees of freedom	Mean Square	F-ratio	p-value
$\mu freq$	252.3	2	126.17	0.25	0.7076
μCr	1690.7	3	563.57	1.14	0.1087
μF	2427.1	3	809.03	1.63	0.0395
p	2809.4	2	1404.7	2.83	0.0239
$\mu freq \cdot \mu Cr$	6359	6	1059.83	2.14	0.0361
$\mu freq \cdot \mu F$	5555.5	6	925.92	1.87	0.0446
$\mu freq \cdot p$	973.2	4	243.3	0.49	0.7429
$\mu Cr \cdot \mu F$	3717.1	9	413.01	0.83	0.5882
$\mu Cr \cdot p$	1473.9	6	245.65	0.49	0.8107
$\mu F \cdot p$	3731.7	6	621.94	1.25	0.1264
Residual	47643.6	96	496.29		
Total	76633.6143				

Cauchy distribution with Eq. (14). Meanwhile, $\mu freq_{ri}$ is randomly selected from S_{freq}, which memorized the success history of $freq_i$ from the previous generations.

$$F_{i,g} = \frac{1}{2}(\sin(2\pi \times freq \times g) \times \frac{g}{G_{\max}} + 1) \tag{14}$$

$$freq_i = randc(\mu freq_{ri}, 0.1) \tag{15}$$

After twenty generation, if $s_{1,j}$ is greater than $s_{2,j}$, Eq. (13) is used to generate F; otherwise, Eq. (14) is used. The success rate s_1, s_2 are set to 0.5 at the beginning and calculated as follows.

Meanwhile, a reward mechanism is introduced to control the choice of parameters. When $nfes \le 0.5nfes_{\max}$, F is chosen from a sinusoidal pool that consists of two configurations in the first twenty generations.

Then, the mutation factor F was generated from the sinusoidal pool based on the reward mechanism and the population size for each configuration is allocated by the success rate s_i. The success

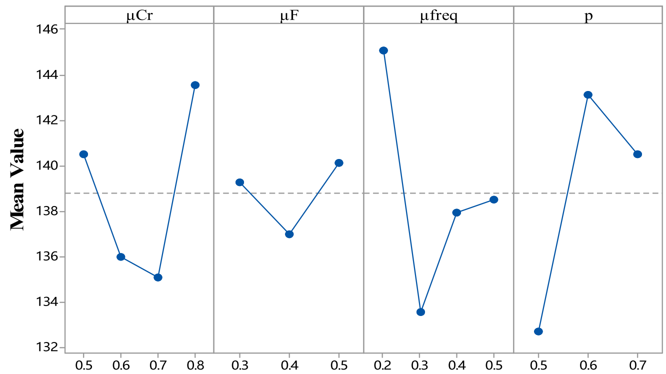


Fig. 7. Main effects plot of the parameters.

Algorithm 1: A self-adapted mutation strategy parameter

---

input : $\mu freq_{ri}$
output: $F_{i,g}$, S_i
if $nfes < 0.5nfes_{\max}$ **then**
  if $g \leq 20$ **then**
    if $rand(0, 1) < s_{1,j}$ **then**
      generate the F with sinusoidal decreasing adjustment mechanism;
    else
      generate the F with sinusoidal increasing adjustment mechanism;
  else
    if $s_{1,j} > s_{2,j}$ **then**
      generate the F with sinusoidal decreasing adjustment mechanism;
    else
      generate the F with sinusoidal increasing adjustment mechanism;
  calculate the successful rate s_1, s_2;
else
  generate the F with Cauchy distribution;

---

rate s_1, s_2 are set to 0.5 at the beginning and calculated as follows.

$$s_i = \frac{\sum_{i=1}^{g} c_{i,g}}{\sum_{i=1}^{g} k_{i,g} + \sum_{i=1}^{g} p_{i,g}} \tag{16}$$

where $c_{i,g}$ was set to zero at the beginning and $i = 1, 2$. Then, if Eq. (13) generated a desirable factor, $c_{1,g} = c_{1,g} + 1$, otherwise, $c_{2,g} = c_{2,g} + 1$. $k_{i,g}$ and $p_{i,g}$ were the number of times to use the first or second configuration in every generation, respectively.

When $nfes > 0.5nfes_{\max}$, the mutation factor F was defined by the Cauchy distribution in the standard L-SHADE algorithm as follows.

$$F = randc(\mu F_i, 0.1) \tag{17}$$

In Algorithm 1, the pseudo-code shows the adjustment of F and the procedure is illustrated in Fig. 3.

3.2. Reset strategy with the collective information powered

When the algorithm falls into a local optimum or in the later phase of iterations, the individuals in the population are considerably close to the fitness values. Meanwhile, the difference vectors generated by the mutation strategies tended to zero and led to insufficient population diversity.

Four different variants of DE (SHADE, L-SHADE, jSO and LSHADE-RSP) were tested in the f_{15} function of CEC2017 in the 10 and 30 dimensions to verify the difference in the diversity of the population in dimensional level. After 1000 evaluations, the logs of standard deviations of the tested algorithm in each dimension are shown in Figs. 4 and 5. The standard deviations in certain dimensions tended to zero after 1000 evaluations rather than all dimensions, which means the diversity of population was not similar in dimensional level and it occurred in the early generation. In the high-dimensional problems, the speed of convergence between the dimensions is different. Meanwhile, the standard deviations in certain dimensions were fixed to zero even the evaluations were increased.

The reset strategy was a useful way to improve the diversity of population at the dimensional level. Meanwhile, regarding the fact that the evolutionary progress in each dimension was asynchronous with the others, the reset strategy was not applied to each dimension.

In the collective information powered reset strategy, the first question is determining how to detect a search process stagnates. Several characteristics manifested when the population was stagnated or the diversity of the population at the dimensional level was insufficient. The standard deviations were closed to zero for certain dimensions, when the diversity of the population at the dimensional level was insufficient. Meanwhile, the algorithms were at a local optimum and had an immutable population distribution.

In the collective information powered reset strategy, $\varphi_{g,j}$ is the variable coefficient of the error values for all the population members, and the value of $\varphi_{g,j}$was calculated as follows.

$$\varphi_{g,j} = \frac{f_{std(g,j)}}{f_{mean(g,j)}} \tag{18}$$

where the $f_{std(g,j)}$ and $f_{mean(g,j)}$ are the standard deviation and the average in jth dimension at the gth generation, respectively. The

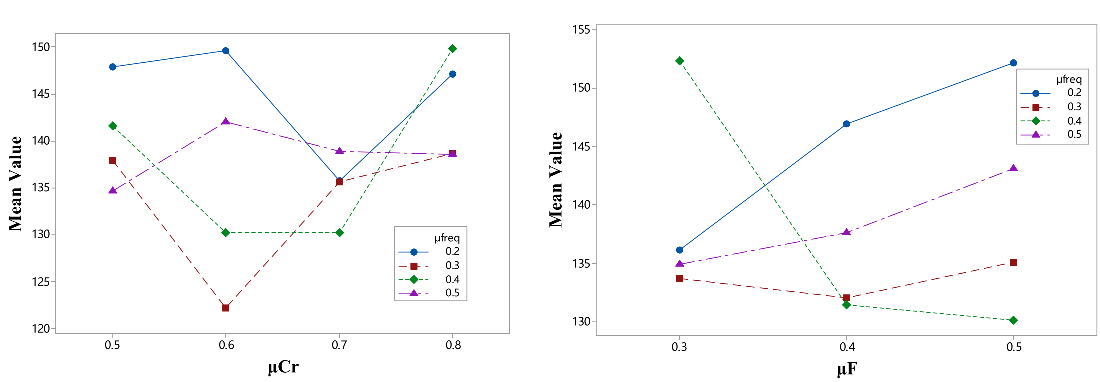


Fig. 8. Interaction plot of the parameters.

reset strategy is executed for each target vector if $\varphi(g, j)$ is less than e^{-10}.

In addition, in the reset strategy, the optimal solution was preserved for the next generation. $\sigma_{i,j}$ was a counter that recorded the number of restarts on jth dimension. The reset strategy was applied to increase the diversity of population when $\sigma_{g,i} \leq Np \times r$ and $\varepsilon_{g,j} = 1$, where Np was the population size and $r = 0.3$. The new individuals $x^j_{i,g+1}$ from the restart strategy were generated as follows.

$$x^j_{i,g+1} = x^j_{low} + (x^j_{low} - x^j_{low}) \times gauss(u_{i,g}, o_{i,g}) \tag{19}$$

where x^j_{low}, x^j_{up}, $u_{j,g}$ and $o_{j,g}$ were calculated as follows.

$$x^j_{low} = (ave_{g,j} + x^j_{\min})/2 \tag{20}$$

$$x^j_{up} = (ave_{g,j} + x^j_{\max})/2 \tag{21}$$

$$u_{j,g} = (ave_{g,j} - x^j_{low})/(x^j_{\max} - x^j_{low}) \tag{22}$$

$$o_{j,g} = e^{k* \frac{nfes_i}{D}} \times \max\left(u_{j,g,1} - u_{j,g}\right) \tag{23}$$

where x^j_{max} and x^j_{min} are the upper and lower bounds of x^j_i and $k = 0.0005$, respectively. The pseudo-code for the reset strategy in dimensional level was given in Algorithm 2.

Algorithm 2: Reset strategy at the dimensional level

input : r, N and x^j_i
output: x^j_i
for $j = 1$ *to* D **do**
  compute $\varphi_{g,j}$;
  if $\varphi_{g,j} \leq e^{-10}$ *and* $\sigma_{i,j} < Np * r$ **then**
    for $i = 1$ *to* Np **do**
      if $x^j_{i,g}$ *is not the best individual* **then**
        Compute $x^j_{i,g+1}$;
        $\sigma_{i,j} = \sigma_{i,j} + 1$;

3.3. LSHADE-CLM algorithm

The framework of LSHADE was applied in the proposed LSHADE-CLM algorithm. However, the LSHADE-CLM algorithm is different in its parameter, mutation strategies and selection operations with LSHADE.

In the selection operation of LSHADE-CLM, the reset strategy was executed in each dimension if the algorithm stagnated or if the diversity of population at the dimensional level was insufficient. The pseudocode of LSHADE-CLM is shown in Algorithm 3 and the procedure is illustrated in Fig. 6. More details about LPSR method is found in [11].

3.4. Convergence analysis of LSHADE-CLM

The Markov chain analysis of differential evolution was given by Hu, et al. [43]. Meanwhile, the classical DE is not converging to the global optimum in probability, which has been proven in existing literature. The main reason resulting in no-convergence in the probability of the classical DE was that, DE fell into and could not escape from a local optimum solution set. In this study, two proposed strategies (i.e., the reset strategy at the dimensional level and learning mechanisms for the parameter in the mutation strategy) were employed in the LSHADE-CLM algorithm. The capacity of the LSHADE-CLM was enhanced by both strategies, and to avoid trapping into a local optimum and premature solution set. Meanwhile, the proposed two strategies also made the LSHADE-CLM converge in probability to the global optimum. The convergence property of LSHADE-CLM was analyzed with the Markov model and the details were as follows.

Algorithm 3: The main procedure of LSHADE-CLM

Initialize population at the first generation g_0,$p_0 = (x_{0,j}, \ldots, x_{N,j})$;
Set archive A as an empty set, $S_C r = \varnothing$, $S_F = \varnothing$, $S_{freq} = \varnothing$ and $T = 0$;
Initialize memory of the first control settings $S_c r$,S_F and second control settings S_{freq};
while $nfes < nfes_{\max}$ **do**
  for $i = 1$ *to* Np **do**
    if $r = H$ **then**
      $F_{i,g} = randc(0.9, 0.1)$ and $Cr_{i,g} = randn(0.9, 0.1)$;
    else
      The details about parameters F selected is shown in Algorithm 1;
      if $\mu Cr_{i,g} < 0$ **then**
        $Cr_{i,g} = 0$
      else
        $Cr_{i,g} = randn(\mu Cr_{i,g}, 0.1)$;
    if $nfes < 0.25 \times nfes_{\max}$ **then**
      $Cr_{i,g} = \max(\mu Cr_g, 0.7)$;
    else if $nfes < 0.5 \times nfes_{\max}$ **then**
      $Cr_{i,g} = \max(\mu Cr_g, 0.6)$;
    if $nfes < 0.6 nfes_{\max}$ *and* $F_{i,g} > 0.7$ **then**
      $F_{i,g} > 0.7$;
    Generate mutant vector v and trial vector t;
    if $Nfes > 0.7 \times nfes_{\max}$ **then**
      The details about reset strategy is shown in Algorithm 2;
    if $f(t) < f(x_i)$ **then**
      $x_i \to A$;
      $Cr_{i,g} \to S_{Cr}$;
      $F_{i,g} \to S_F$;
      $\mu freq \to S_{freq}$;
    Update μCr, μF, p and $\mu freq$ and **LPSR** method is applied;
  Update generation number $g = g + 1$ and $nfes$;

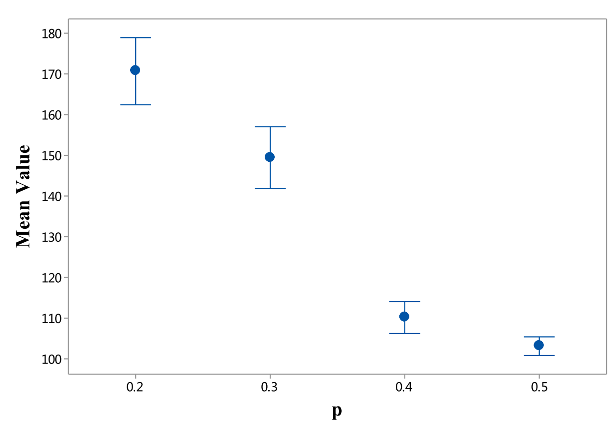


Fig. 9. Means plot and 95% confidence level intervals of p.

Definition 1 (*Convergence in Probability [44]*)**.** Let $\{x(t), t = 0, 1, 2, \ldots\}$ be a population sequence generated by a population-based stochastic algorithm. The stochastic sequence $\{x(t)\}$ weakly converges in probability to the global optimum, if and only if Eq. (24) is true.

$$\lim_{t\to\infty} \{x(t) \cap B^* \neq \emptyset\} = 1 \tag{24}$$

where B^* is the set of the global optima of an optimization problem, and the algorithm holds with a convergence in probability. Otherwise, the sequence $\{x(t)\}$ and the algorithm do not converge in probability.

Definition 2. If a random sequence $\{X_n; n \geq 0\}$ is takes values from a discrete set from a random variable, all the discrete values are in the memory of $H_L = \{j\}$, where H_L is called the state space.

Table 9
Average results of LSHADE-CLM-ND, LSHADE-CLM-NP and LSHADE-CLM ($D = 10$)

Function types	LSHADE-CLM-ND	LSHADE-CLM-NCM	LSHADE-CLM
Unimodal functions ($f_1 - f_3$)	**0.00E+00**$_{0.00E+00}$	0.00E+00$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$
Simple multimodal functions ($f_4 - f_{10}$)	5.17E+00$_{1.23E+00}$	3.60E+00$_{1.11E+00}$	**3.19E+00**$_{5.67E-01}$
Hybrid functions ($f_{11} - f_{20}$)	**5.67E+00**$_{6.09E-01}$	6.15E+00$_{2.02E+00}$	6.02E+00$_{9.20E-01}$
Composition functions ($f_{21} - f_{30}$)	1.72E+03$_{3.46E+01}$	1.07E+03$_{2.87E+01}$	**2.84E+02**$_{1.85E+01}$

Table 10
Average results of LSHADE-CLM-ND, LSHADE-CLM-NP and LSHADE-CLM ($D = 100$)

Function types	LSHADE-CLM-ND	LSHADE-CLM-NCM	LSHADE-CLM
Unimodal functions ($f_1 - f_3$)	1.02E+02$_{8.71E+00}$	1.79E+02$_{1.80E+01}$	**8.02E+01**$_{4.43E+02}$
Simple multimodal functions ($f_4 - f_{10}$)	2.07E+03$_{9.98E+01}$	1.60E+03$_{1.01E+02}$	**1.59E+03**$_{9.64E+01}$
Hybrid functions ($f_{11} - f_{20}$)	1.76E+03$_{1.61E+02}$	1.53E+03$_{3.96E+02}$	**9.45E+02**$_{2.03E+02}$
Composition functions ($f_{21} - f_{30}$)	2.62E+03$_{9.43E+01}$	3.09E+03$_{1.24E+02}$	**2.04E+03**$_{1.26E+02}$

For any $n \geq 1$ and $i_k \in H_L$ ($k \leq n+1$), if Eq. (25) is true, $\{X_n; n \geq 0\}$ is called the Markov chain.

$$P\{x_{n+1} = i_{n+1}|X_n = i_n, \ldots, X_0 = i_0\} = P\{x_{n+1} = i_{n+1}|X_n = i_n\} \tag{25}$$

H_L in a random sequence $\{X_n; n \geq 0\}$ is infinite or finite as needed, and H_L was finite in LSHADE-CLM algorithm.

Definition 3. For time m, the probability of the Markov chain $\{X_n; n \geq 0\}$ from state i to state j is $P_{ij}(m, n)$; if $P_{ij}(m, n) \equiv P_{ij}^n$, the corresponding transition probability is independent of m and the Markov chain $\{X_n; n \geq 0\}$ is homogeneous at the time; where m and n are positive integers, and the calculation of $P_{ij}(m, n)$ is shown as follows.

$$P_{i,j}(m, n) = P\{X_{m+n} = j|X_m = i\} \tag{26}$$

Definition 4. R^0, M^0, S^0, C^0 denote the reset operator, mutation operator, selection operator and crossover operator of the LSHADE-CLM respectively. Let X, Y, Z denote the states of ϕ. Then, $P\{M^0(X) = Y\}$ is the transition probability from X to Y by using a migration operator.

Definition 5. $f(X^*)$ denotes the minimum function value of a population X.

Property 1. *For LSHADE-CLM without a selection operator, all states of its population space ϕ communicate. For all states $X, Y \in \phi$, the one-step transition probabilities from X to Y are greater than 0. Namely,*

$$P\{R^0C^0M^0(X) = Y\} > 0 \tag{27}$$

Proof. For LSHADE-CLM without the diversity selection operator, only three operators, the mutation M^0, the crossover C^0 and the reset operator R^0 were used. Suppose that ϕ denotes the population space of LSHADE-CLM without the selection operator. $X, Y, Z \in \phi$, from Eq. (11) and Definition 4,

$$P\{R^0C^0M^0(X) = Y\} = P\{C^TM^0(X) = Y\}(1 - Um) + Um \sum_{Z\in\phi} \{P\{C^0M^0(X) = Z\}P\{R_1^0(Z) = Y\}\} \tag{28}$$

Um is the probability that Eq. (28) is not true.

$$rand(0, 1) \leq \frac{popsize_g}{popsize_g + popsize_{\max}} \tag{29}$$

For all Z, $P\{R^0(X) = Y\} > 0$, then

$$Um \sum_{Z\in\phi} \{P\{C^0M^0(X) = Z\}P\{R_1^0(Z) = Y\}\} > 0 \tag{30}$$

Table 11
Algorithm complexity.

Algorithm	D	$T0$	$T1$	$T2$	$(T2 - T1)/T0$
LSHADE-CLM	$D = 10$	0.237	**0.342**	**3.79**	**14.5485**
	$D = 30$		**1.317**	**5.649**	**18.2784**
	$D = 50$		**2.743**	**8.103**	**22.616**
LSHADE-RSP	$D = 10$	0.382	0.361	3.2064	7.44869
	$D = 30$		1.007	7.681	17.4712
	$D = 50$		2.212	12.6452	27.312

$$P\{R^0C^0M^0(X) = Y\} > 0 \tag{31}$$

For LSHADE-CLM without the diversity selection operator, all states are communicated. Afterwards, let $f(X^*)$ denote the minimum function value of a population X. A statistic property of the diversity of the selection operator is proven.

Property 2. *Given states (populations) X, Y, Z of the state space ϕ, and $Z \subset X \cup Y$, the diversity selection operator belongs to one of the following two classes:*

1. *If $f(Z^*) \neq \min\{f(X^*), f(Y^*)\}$, Z is not generated by X, Y with using the diversity selection. $P\{S^0(X, Y) = Z\} = 0$*
2. *If $f(Z^*) = \min\{f(X^*), f(Y^*)\}$, Z is generated by X,Y with using the diversity selection. $P\{S^0(X, Y) = Z\} > 0$*

Theorem 1. *The population sequence $\{X(t), t = 0, 1, 2, \ldots\}$ in the LSHADE-CLM algorithm is a finite homogeneous Markov chain.*

Proof. According to Definitions 2 and 3, if the length of the individual in LSHADE-CLM is N and the population size is M, and because the individual values are all set in the real number space, the state space of the population is infinite in theory. However, the problems discussed in practical operations are of finite precision if the dimension is set to j, and the state space of the population is υ^{NM}; thus, the population sequence is finite, and for the current time t, the mutation, crossover and selection operation are all independent of t and $X(t + 1)$ is only related to $X(t)$. Therefore, $\{X_t; t \geq 0\}$ is a finite homogeneous Markov chain.

Theorem 2. *Suppose that $\{X(t), t = 0, 1, 2, \ldots\}$ is the population sequence generated by the LSHADE-CLM, $\{X(t), t = 0, 1, 2, \ldots\}$ converges in probability to the global optimum.*

Proof. Define $K^0 = R^0C^0M^0S^0$, $\forall X, Y, Z \in \phi$, then the transition probability

$$\begin{aligned} P\{x(t + 1) = Z|x(t) = X\} &= P\{S^0(X) = Z\} \\ &= P\{K = S^0(R^0C^0M^0)(X) = Z\} \\ &= \sum_{Y\in\phi} P\{(R^0C^0M^0)(X) = Y\}P\{K^0(X, Y) = Z\} \end{aligned} \tag{32}$$

Table 12
The experimental results of all algorithms on 10 dimensions.

Fun	jSO *Mean$_{Std}$*	LSHADE-RSP *Mean$_{Std}$*	LSHADE-cnEpsin *Mean$_{Std}$*	ELSHADE-SPACMA *Mean$_{Std}$*	EBLSHADE *Mean$_{Std}$*	LSHADE-CLM *Mean$_{Std}$*
F1	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$
F2	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$
F3	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$
F4	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+0**0$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$
F5	1.76E+00$_{7.06E-01}$	1.48E+00$_{7.97E-01}$	1.69E+00$_{7.53E-01}$	3.87E+00$_{2.02E+00}$	2.68E+00$_{1.03E+00}$	**1.07E+00**$_{8.17E-01}$
F6	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$
F7	1.18E+01$_{6.07E-01}$	1.17E+01$_{5.07E-01}$	1.20E+01$_{4.80E-01}$	1.33E+01$_{1.75E+00}$	1.22E+01$_{9.83E-01}$	**1.14E+01**$_{5.99E-01}$
F8	1.95E+00$_{7.44E-01}$	1.50E+00$_{9.53E-01}$	1.80E+00$_{7.71E-01}$	4.10E+00$_{2.51E+00}$	2.34E+00$_{1.93E+00}$	**1.27E+00**$_{8.53E-01}$
F9	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$
F10	3.59E+01$_{5.55E+01}$	2.50E+01$_{4.79E+01}$	4.30E+01$_{2.57E+01}$	2.27E+01$_{1.84E+01}$	1.89E+01$_{2.05E+00}$	**8.60E+00**$_{1.70E+00}$
F11	0.00E+00$_{0.00E+00}$	0.00E+00$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	0.00E+00$_{0.00E+00}$	0.00E+00$_{0.00E+00}$	0.00E+00$_{0.00E+00}$
F12	2.66E+00$_{1.68E+01}$	**3.63E−01**$_{1.69E-01}$	1.01E+02$_{7.30E+0}$1	2.85E+01$_{5.14E+01}$	1.02E+01$_{1.01E-01}$	5.33E+01$_{5.91E+00}$
F13	**2.96E+00**$_{2.35E+00}$	3.55E+00$_{2.25E+00}$	3.66E+00$_{2.66E+00}$	3.57E+00$_{2.21E+00}$	3.76E+00$_{1.07E+00}$	4.29E+00$_{2.01E+00}$
F14	5.85E−01$_{2.36E-01}$	3.90E−02$_{1.93E-01}$	7.80E−02$_{2.70E-01}$	7.80E−02$_{2.70E-01}$	4.56E−01$_{2.00E-02}$	**0.00E+00**$_{0.00E+00}$
F15	2.21E−01$_{2.00E-01}$	1.87E−01$_{2.16E-01}$	3.24E−01$_{2.16E-01}$	2.51E−01$_{2.17E-01}$	**1.60E−01**$_{2.35E-01}$	3.92E−01$_{2.06E-01}$
F16	5.69E−01$_{2.64E-01}$	5.42E−01$_{2.81E-01}$	**5.37E−01**$_{2.93E-01}$	5.62E−01$_{2.55E-01}$	**4.07E−01**$_{5.23E-03}$	5.63E−01$_{2.93E-01}$
F17	5.02E−01$_{3.48E-01}$	1.06E+00$_{2.78E+00}$	3.07E−01$_{3.81E-01}$	1.39E−01$_{1.44E-01}$	**1.09E−01**$_{1.64E-02}$	7.32E−01$_{4.55E-01}$
F18	3.08E−01$_{1.95E-01}$	**2.23E−01**$_{2.12E-01}$	3.86E+00$_{7.63E+00}$	7.10E−01$_{2.80E+00}$	2.65E−01$_{1.83E-01}$	4.13E−01$_{1.50E-01}$
F19	1.07E−01$_{1.25E-02}$	8.50E−03$_{9.60E-03}$	4.47E−02$_{2.09E-01}$	1.55E−02$_{1.14E-02}$	**5.97E−03**$_{1.72E-05}$	1.93E−02$_{1.56E-02}$
F20	3.43E−01$_{1.29E-01}$	4.35E−01$_{1.52E-01}$	2.57E−01$_{2.31E-01}$	**1.41E−01**$_{1.57E-01}$	**0.00E+00**$_{0.00E+00}$	4.72E−01$_{1.57E-01}$
F21	1.32E+02$_{4.84E+01}$	1.22E+02$_{4.23+01}$	1.46E+02$_{5.17E+01}$	**1.02E+02**$_{1.48E+01}$	1.51E+02$_{1.53E+00}$	1.30E+02$_{4.73E+01}$
F22	**1.00E+02**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{6.80E-02}$	**0.00E+00**$_{1.21E-01}$	**0.00E+00**$_{9.31E-07}$	1.00E+02$_{0.00E+00}$
F23	3.01E+02$_{1.59E+00}$	**2.89E+02**$_{5.84E+01}$	3.02E+02$_{1.64E+00}$	3.04E+02$_{2.30E+00}$	3.03E+02$_{1.03E+0}$1	3.01E+02$_{1.44E+00}$
F24	2.97E+02$_{7.93E+01}$	2.93E+02$_{8.34E+01}$	3.16E+02$_{5.45E+01}$	2.91E+02$_{9.54E+01}$	2.99E+02$_{1.71E+01}$	**2.80E+02**$_{9.54E+01}$
F25	**4.06E+02**$_{1.75E+01}$	4.10E+02$_{2.03E+01}$	4.26E+02$_{2.24E+01}$	4.13E+02$_{2.18E+01}$	4.13E+02$_{2.56E+0}$1	4.10E+02$_{2.05e+01}$
F26	**3.00E+02**$_{0.00E+00}$	**3.00E+02**$_{0.00E+00}$	**3.00E+02**$_{0.00E+00}$	**3.00E+02**$_{0.00E+00}$	**3.00E+02**$_{0.00E+00}$	**3.00E+02**$_{0.00E+00}$
F27	**3.89E+02**$_{2.26E-01}$	**3.89E+02**$_{2.23E-01}$	3.90E+02$_{1.96E+00}$	**3.89E+02**$_{1.67E-01}$	**3.89E+02**$_{6.51E-04}$	**3.89E+02**$_{7.09E-01}$
F28	3.39E+02$_{9.65E+01}$	3.18E+02$_{2.89E+0}$0	3.85E+02$_{1.72E+00}$	3.25E+02$_{1.04E+02}$	3.53E+02$_{4.71E+01}$	3.00E+02$_{0.00E+01}$
F29	2.34E+02$_{2.96E+00}$	2.34E+02$_{2.89E+00}$	**2.28E+02**$_{1.72E+00}$	2.30E+02$_{2.26E+00}$	2.34E+02$_{3.51E+00}$	2.34E+02$_{3.07E+00}$
F30	**3.95E+02**$_{4.50E-02}$	**3.95E+02**$_{2.29E-02}$	1.76E+04$_{8.61E+04}$	4.02E+02$_{1.77E+01}$	4.01E+02$_{1.03E+01}$	4.00E+02$_{1.67E+01}$

Table 13
Rankings obtained through the Wilcoxon signed-rank test on 10 dimensions.

Dimension	LSHADE-CLM vs.	R+	R-	R≈	p-value	$\alpha = 0.05$	$\alpha = 0.1$
	jSO	10	8	12	0.616	No	No
	LSHADE-RSP	8	10	12	0.983	No	No
10	LSHADE-cnEpsin	15	6	9	0.009	**Yes**	**Yes**
	ELSHADE-SPACMA	11	9	10	0.37	No	No
	EBLSHADE	11	8	11	0.084	No	**Yes**

Let B^0 be a population set consisting of some populations in which at least one individual is the optimum. Namely, $B^0 \subset \phi$:

$$B^0 = \{X = (x_1, x_2, \ldots, x_m) \in \phi | x_i \in B^*, \exists i \in \{1, 2, \ldots, m\}\} \tag{33}$$

and the transition probability of two classes are discussed.

1. Suppose that $X \in B^0, Z \in B^0$, and $f(Z^*) = \min\{f(X^*), f(Y^*)\}$, which means $P\{K^0(X,Y) = Z\} > 0$. According to Property 1, for all $X, Y \in \phi$,

   $$P\{(R^0C^0M^0)(X) = Y\} > 0 \tag{34}$$

   and according to Eq. (32)

   $$P\{x(t+1) = Z | x(t) = X\} = 0 \tag{35}$$

2. Suppose that $X \in B^0$, $Z \notin B^0$, $f(Z^*) > \min\{f(X^*), f(Y^*)\}$, which means that $f(Z^*) \neq \min\{f(X^*), f(Y^*)\}$. According to Property 2,

   $$P\{K^0(X,Y) = Z\} = 0 \tag{36}$$

   and according to Eq. (32)

   $$P\{x(t+1) = Z | x(t) = X\} > 0 \tag{37}$$

   $$P\{x(t+1) = X | x(t) = Z\} > 0 \tag{38}$$

   from Eqs. (35) and (36), all states of B^0 communicate.

According to Eqs. (35), (37) and (38), B^0 is a positive recurrent, irreducible, a periodic and closed set; Meanwhile, we can obtain that B^0 is a positive recurrent, irreducible, aperiodic and closed set, and ϕ^m/B^0 is a non-recurrent state set. Where ϕ^m/B^0 denotes the cut set of the ϕ^m to the B^0.

In addition, according to the properties of the aperiodic, homogeneous Markov chain [45], we can get that the sequence $\{x(t), t = 0, 1, 2, \ldots\}$ exists a limiting distribution $\pi(Y)$, such that

$$\lim_{t\to\infty} P\{x(t) = Y\} = \begin{cases} \pi(Y) & Y \in B^0 \\ 0 & otherwise \end{cases} \tag{39}$$

According to Eq. (39),

$$\lim_{t\to\infty} P\{x(t) = B^0\} = 1 \tag{40}$$

and

$$\lim_{t\to\infty} P\{x(t) \cap B^0 \neq \phi\} = 1 \tag{41}$$

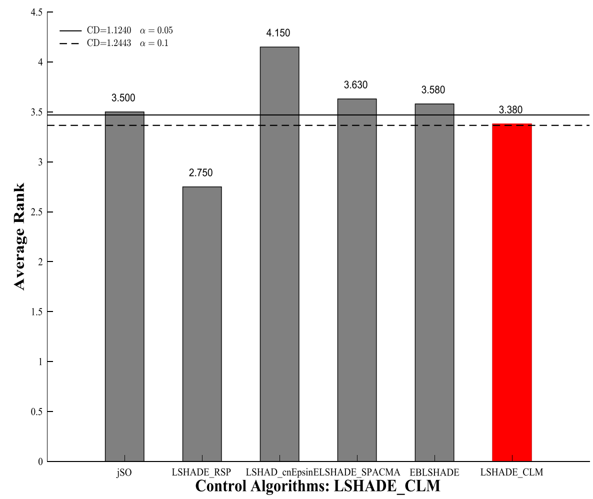
According to Definition 1, LSHADE-CLM converges in probability to the global optimum.

4. The experiments and comparisons

In this study, the CEC2017 competition for non-separable optimization problem [46] was applied to test the performance of LSHADE-CLM and state-of-the-art variant algorithms of the L-SHADE algorithm. The CEC2017 benchmark function consists of four different categorizations: $f_1 - f_3$ are unimodal functions, $f_4 - f_{10}$ are multimodal functions, $f_{11} - f_{20}$ are hybrid functions, and $f_{21} - f_{30}$ are composition functions. All compared algorithms were run 51 times on the benchmark function in dimensions $D = 10, 30, 50, 100$, respectively. According to the requirements of CEC2017 benchmark function, the maximum fitness of function evaluations was set to $D \times 10,000$, and various details were found in the literature [47]. All compared algorithms were programmed

Algorithm	Mean Rank
LSHADE-RSP	2.750
LSHADE-CLM	3.380
jSO	3.500
EBLSHADE	3.580
LSHADE-SPACMA	3.630
ELSHADE- cnEpsin	4.150
Crit.Diff, α=0.05	1.1240
Crit.Diff, α=0.10	1.2443

(a)



(b)

Fig. 10. Rankings for $D = 10$.

Table 14
The experimental results of all algorithms on 30 dimensions.

Fun	jSO *Mean_{Std}*	LSHADE-RSP *Mean_{Std}*	LSHADE-cnEpsin *Mean_{Std}*	ELSHADE-SPACMA *Mean_{Std}*	EBLSHADE *Mean_{Std}*	LSHADE-CLM *Mean_{Std}*
F1	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$
F2	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$
F3	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$
F4	5.87E+01$_{7.78E-01}$	5.86E+01$_{0.00E+00}$	**4.23E+01**$_{3.07e+00}$	5.86E+01$_{0.00E+00}$	5.87E+01$_{0.00E+00}$	5.82E+01$_{0.00E+00}$
F5	8.56E+00$_{2.10E+00}$	7.43E+00$_{1.89E+00}$	1.23E+01$_{2.34E+00}$	1.86E+01$_{8.04E+00}$	**6.51E+00**$_{1.21E+00}$	7.32E+00$_{1.86E+00}$
F6	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{00.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	2.14E−08$_{9.00E-08}$	**0.00E+00**$_{0.00E+00}$
F7	3.89E+01$_{1.46E+00}$	3.86E+01$_{1.91E+00}$	4.33E+01$_{2.17E+00}$	3.89E+01$_{3.43E+00}$	**3.68E+01**$_{4.10E+00}$	3.89E+01$_{2.05E+00}$
F8	9.09E+00$_{1.84E+00}$	7.55E+00$_{2.06E+00}$	1.29E+01$_{2.86E+00}$	1.61E+01$_{7.46E+00}$	**6.81E+0**$_{01.70E+00}$	7.63E+00$_{2.16E+00}$
F9	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$
F10	1.53E+03$_{2.77E+02}$	1.64E+03$_{2.93E+02}$	**1.39E+03**$_{2.10E+02}$	1.70E+03$_{4.06E+02}$	**1.37E+03**$_{1.53E+02}$	1.55E+03$_{2.64E+02}$
F11	3.04E+00$_{2.65E+00}$	4.47E+00$_{1.15E+01}$	1.35E+01$_{1.94E+01}$	7.80E+00$_{1.42E+01}$	2.07E+01$_{3.21E+00}$	6.95E+00$_{1.59E+00}$
F12	1.70E+02$_{1.02E+02}$	**9.40E+01**$_{6.67E+01}$	3.72E+02$_{2.01E+02}$	2.47E+02$_{4.06E+02}$	7.94E+02$_{2.10E+02}$	1.53E+02$_{1.04E+02}$
F13	1.48E+01$_{4.83E+00}$	1.76E+01$_{3.13E+00}$	1.73E+01$_{1.02E+01}$	1.57E+01$_{5.03E+00}$	**1.41E+01**$_{9.03E+00}$	1.65E+01$_{5.34E+00}$
F14	2.18E+01$_{1.25E+00}$	2.19E+01$_{1.31E+00}$	2.16E+01$_{2.26E+00}$	2.37E+01$_{5.25E+00}$	**1.93E+01**$_{7.29E+00}$	2.00E+01$_{4.04E+00}$
F15	1.09E+00$_{6.91E-01}$	**8.90E−01**$_{6.76E-01}$	3.24E+00$_{1.98E+00}$	1.86E+00$_{1.29E+00}$	2.41E+00$_{1.01E+00}$	1.97E+00$_{1.32E+00}$
F16	7.89E+01$_{8.48E+01}$	**1.88E+01**$_{5.96E+00}$	2.29E+01$_{3.07E+01}$	6.68E+01$_{8.35E+01}$	3.89E+01$_{8.30E+00}$	3.18E+01$_{4.48E+00}$
F17	3.29E+01$_{8.08E+00}$	3.57E+01$_{5.87E+00}$	**2.86E+01**$_{5.56E+00}$	2.97E+01$_{6.76E+00}$	3.27E+01$_{5.41E+00}$	3.20E+01$_{6.52E+00}$
F18	2.04E+01$_{2.87+00}$	**2.00E+01**$_{3.86E+00}$	2.11E+01$_{7.52E-01}$	2.09E+01$_{3.03E+00}$	2.14E+01$_{1.05E+00}$	2.11E+01$_{4.98E-01}$
F19	4.50E+00$_{1.73E+00}$	**3.30E+00**$_{8.89E-01}$	5.83E+00$_{1.92E+00}$	4.61E+00$_{1.35E+00}$	4.83E+00$_{9.33E-01}$	3.72E+00$_{1.21E+00}$
F20	2.94E+01$_{5.85E+00}$	3.31E+01$_{6.02E+00}$	3.03E+01$_{7.35E+00}$	**2.73E+01**$_{4.56E+00}$	3.05E+01$_{1.42E+00}$	3.13E+01$_{6.06E+00}$
F21	2.09E+02$_{1.96E+00}$	**2.07E+02**$_{1.88E+00}$	2.12E+02$_{2.56E+00}$	2.22E+02$_{6.64E+00}$	2.07E+02$_{5.53E+00}$	2.05E+02$_{2.03E+00}$
F22	**1.00E+02**$_{0.00E+00}$	**1.00E+02**$_{0.00E+00}$	**1.00E+02**$_{6.80E-02}$	**1.00E+02**$_{1.21E-01}$	**1.00E+02**$_{0.00E+00}$	**1.00E+02**$_{0.00E+00}$
F23	3.51E+02$_{3.30E+00}$	3.51E+02$_{2.98E+00}$	3.56E+02$_{3.73E+00}$	3.69E+02$_{1.05E+01}$	**3.49E+02**$_{4.30E+01}$	3.47E+02$_{2.87E+00}$
F24	4.26E+02$_{2.47E+00}$	4.27E+02$_{1.81E+00}$	4.28E+02$_{2.95E+00}$	4.41E+02$_{7.84E+00}$	4.25E+02$_{2.15E+01}$	**4.23E+02**$_{1.96E+00}$
F25	3.87E+02$_{7.68E-03}$	3.87E+02$_{6.70E-03}$	3.87E+02$_{8.90E-03}$	3.87E+02$_{9.60E-03}$	3.87E+02$_{9.22E-02}$	**3.86E+02**$_{7.17E-03}$
F26	9.20E+0$_{24.30E+01}$	9.29E+02$_{4.17E+01}$	9.49E+02$_{4.60E+01}$	1.08E+03$_{8.68E+01}$	**9.01E+02**$_{9.41E+01}$	9.11E+02$_{2.99E+01}$
F27	**4.97E+02**$_{7.00E+00}$	**4.97E+02**$_{7.96E+00}$	5.04E+02$_{6.70E+00}$	4.99E+02$_{6.15E+00}$	5.01E+02$_{8.30E+00}$	5.07E+02$_{5.54E+00}$
F28	3.09E+02$_{3.03E+01}$	3.04E+02$_{2.21E+01}$	3.15E+02$_{3.86E+01}$	**3.02E+02**$_{1.60E+01}$	3.22E+02$_{1.03E+01}$	3.03E+02$_{2.54E+01}$
F29	4.34E+02$_{1.36E+01}$	4.41E+02$_{1.16E+01}$	4.35E+02$_{7.36E+00}$	4.33E+02$_{1.56E+01}$	**4.32E+02**$_{2.01E+01}$	4.60E+02$_{1.56E+01}$
F30	**1.97E+03**$_{1.90E+01}$	**1.97E+03**$_{7.95E+00}$	1.98E+03$_{4.17E+01}$	1.98E+03$_{3.34E+01}$	1.99E+03$_{3.21E+01}$	1.98E+03$_{3.17E+01}$

with C++ and implemented on PC with 3.4 GHz Intel(R) Core i7-7700 CPU, 16 GB of RAM and a 64-bit OS to ensure that the comparison was fair. The competitive algorithms are listed as follows, and the parameters of the comparison algorithms are shown in Table 2.

4.1. Parameter analysis of LSHADE-CLM

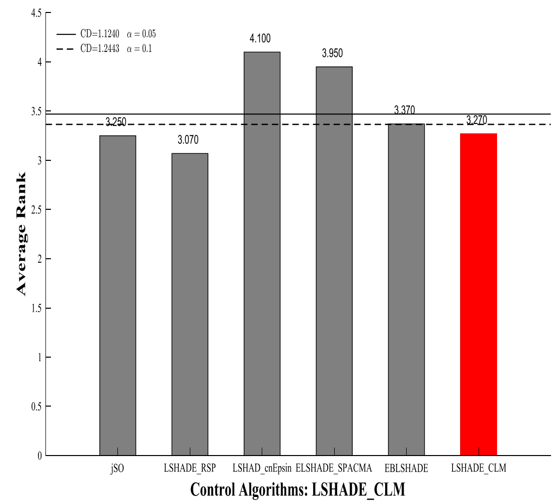
The selection of parameters plays an important role on the effectiveness and efficiency of DE and its variants. The parameters setting of an algorithm are determined in two ways: based on previous literature and experiments [48]. In the first method, the parameters were determined by summarizing the relevant previous literature. The values of parameters in the previous

Table 15
Rankings obtained through the Wilcoxon signed-rank test on 30 dimensions.

Dimension	LSHADE-CLM vs.	R+	R-	R≈	p-value	$\alpha = 0.05$	$\alpha = 0.1$
30	jSO	14	10	6	0.548	No	No
	LSHADE-RSP	13	12	5	0.83	No	No
	LSHADE-cnEpsin	15	8	7	0.23	No	No
	ELSHADE-SPACMA	14	9	7	0.05	**Yes**	**Yes**
	EBLSHADE	11	11	8	0.833	No	No

Algorithm	Mean Rank
LSHADE-RSP	3.070
jSO	3.250
LSHADE-CLM	3.270
EBLSHADE	3.370
ELSHADE-SPACMA	3.950
ELSHADE- cnEpsin	4.100
Crit.Diff, α=0.05	1.1240
Crit.Diff, α=0.10	1.2443

(a)

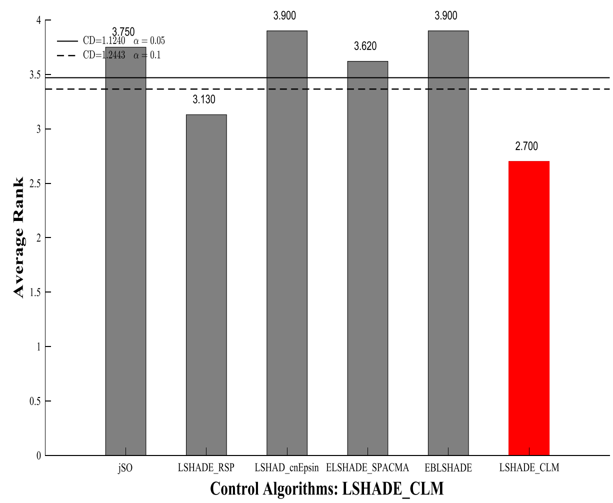


(b)

Fig. 11. Rankings for $D = 30$.

Algorithm	Mean Rank
LSHADE-CLM	2.700
LSHADE-RSP	3.130
ELSHADE-SPACMA	3.620
jSO	3.750
LSHADE-cnEpsin	3.900
EBLSHADE	3.900
Crit.Diff, α=0.05	1.1240
Crit.Diff, α=0.10	1.2443

(a)

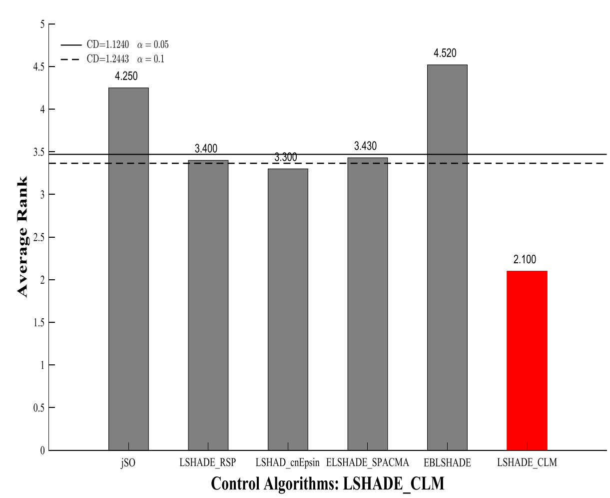


(b)

Fig. 12. Rankings for $D = 50$.

Algorithm	Mean Rank
LSHADE-CLM	2.100
LSHADE-cnEpsin	3.300
LSHADE-RSP	3.400
ELSHADE-SPACMA	3.430
jSO	4.250
EBLSHADE	4.520
Crit.Diff, α=0.05	1.1240
Crit.Diff, α=0.10	1.2443

(a)



(b)

Fig. 13. Rankings for $D = 100$.

Table 16
The experimental results of all algorithms on 50 dimensions.

Fun	jSO $Mean_{Std}$	LSHADE-RSP $Mean_{Std}$	LSHADE-cnEpsin $Mean_{Std}$	ELSHADE-SPACMA $Mean_{Std}$	EBLSHADE $Mean_{Std}$	LSHADE-CLM $Mean_{Std}$
F1	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$
F2	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$
F3	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$
F4	5.62E+01$_{4.88E+01}$	6.20E+01$_{5.07E+01}$	5.14E+01$_{4.43E+01}$	4.36E+01$_{3.62E+0}$1	7.91E+01$_{4.11E+01}$	**3.37E+01**$_{2.86E+01}$
F5	1.64E+01$_{3.46E+00}$	1.35E+01$_{3.30E+00}$	2.52E+01$_{6.44E+00}$	1.39E+01$_{5.55E+00}$	**1.27E+01**$_{3.71E+00}$	1.35E+01$_{3.50E+00}$
F6	1.09E−06$_{2.63E-06}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	2.11E−05$_{0.00E+00}$	9.01E−08$_{0.00E+00}$
F7	6.65E+01$_{3.47E+00}$	6.72E+01$_{3.13E+00}$	7.66E+01$_{6.06E+00}$	**6.15E+01**$_{3.86E+00}$	6.28E+01$_{3.27E+00}$	7.11E+01$_{3.89E+00}$
F8	1.70E+01$_{3.14E+00}$	1.37E+01$_{3.52E+00}$	2.63E+01$_{6.59E+00}$	1.79E+01$_{7.47E+00}$	**1.24E+01**$_{2.15E+00}$	1.59E+01$_{3.55E+00}$
F9	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$
F10	3.14E+03$_{3.67E+02}$	3.60E+03$_{3.17E+02}$	3.20E+03$_{3.40E+00}$2	3.69E+03$_{6.07E+02}$	**3.03E+03**$_{4.11E+02}$	3.50E+03$_{3.23E+02}$
F11	2.79E+01$_{3.33E+00}$	2.40E+01$_{3.59E+00}$	**2.14E+01**$_{2.09E+00}$	2.62E+01$_{3.76E+00}$	4.40E+01$_{5.30E+00}$	2.19E+01$_{3.57E+00}$
F12	1.68E+03$_{5.23E+02}$	1.37E+0$_{34.09E+02}$	1.48E+03$_{3.65E+02}$	1.36E+03$_{3.42E+02}$	2.15E+03$_{1.21E+02}$	**8.06E+02**$_{2.62E+02}$
F13	3.06E+01$_{2.12E+01}$	**2.45E+01**$_{1.68E+01}$	6.94E+01$_{3.45E+01}$	3.68E+01$_{1.72E+01}$	4.60E+01$_{0.00E+00}$	3.13E+01$_{2.67E+01}$
F14	2.50E+01$_{1.87E+00}$	2.28E+$_{011.56E+00}$	2.65E+01$_{2.49E+00}$	3.07E+01$_{3.95E+00}$	2.69E+01$_{4.02E+00}$	2.35E+01$_{1.98E+00}$
F15	2.39E+01$_{2.49E+00}$	2.06E+01$_{1.73E+00}$	2.56E+01$_{4.06E+00}$	**2.28E+01**$_{2.20E+00}$	3.44E+01$_{1.63E+00}$	**2.00E+01**$_{1.79E+00}$
F16	4.51E+02$_{1.38E+02}$	3.94E+02$_{1.66E+02}$	**2.75E+02**$_{9.97E+01}$	4.15E+02$_{1.77E+02}$	3.34E+02$_{7.11E+01}$	3.21E+$_{021.44E+02}$
F17	2.83E+02$_{8.61E+01}$	2.54E+02$_{8.96E+01}$	**2.07E+02**$_{7.31E+01}$	2.30E+02$_{9.68E+01}$	2.35E+02$_{2.10E+01}$	2.28E+0$_{27.62E+01}$
F18	2.43E+01$_{2.02E+00}$	2.25E+01$_{1.18E+00}$	2.43E+01$_{2.12E+00}$	2.51E+$_{012.56E+00}$	3.28E+$_{013.00E+00}$	**2.20E+01**$_{1.47E+00}$
F19	1.41E+01$_{2.26E+00}$	**1.01E+01**$_{1.96E+00}$	1.74E+01$_{2.47E+00}$	1.44E+01$_{2.31E+00}$	1.88E+01$_{3.06E+00}$	1.21E+01$_{2.76E+00}$
F20	1.40E+02$_{7.74E+00}$	1.59E+02$_{7.74E+01}$	1.14E+02$_{3.55E+01}$	**1.08E+02**$_{7.31E+01}$	1.50E+02$_{3.44E+01}$	1.36E+02$_{6.58E+01}$
F21	2.19E+02$_{3.77E+00}$	2.15E+02$_{3.57E+00}$	2.27E+$_{027.06E+00}$	2.42E+02$_{9.52E+00}$	2.13E+02$_{5.82E+00}$	**2.11E+02**$_{3.26E+00}$
F22	1.49E+03$_{1.75E+03}$	2.13E+03$_{1.95E+03}$	1.60E+03$_{1.67E+03}$	**7.16E+02**$_{1.44E+03}$	2.36E+03$_{1.03E+03}$	1.46E+03$_{1.75E+03}$
F23	4.30E+02$_{6.24E+00}$	4.32E+02$_{6.69E+00}$	4.39E+02$_{6.90E+00}$	4.62E+02$_{1.39E+01}$	4.27E+02$_{4.03E+00}$	**4.25E+02**$_{5.59E+00}$
F24	5.07E+02$_{4.13E+00}$	5.09E+02$_{3.74E+00}$	5.13E+02$_{5.59E+00}$	5.34E+02$_{9.14E+00}$	5.05E+02$_{9.20E+00}$	**5.04E+02**$_{3.48E+00}$
F25	4.81E+02$_{2.80E+00}$	4.80E+02$_{9.40E-03}$	4.80E+02$_{1.08E+00}$	4.81E+02$_{2.80E+00}$	4.89E+02$_{1.02E+00}$	**4.77E+02**$_{1.52E-02}$
F26	1.13E+03$_{5.62E+01}$	1.14E+03$_{5.25E+01}$	1.20E+03$_{1.19E+02}$	1.34E+03$_{1.38E+02}$	1.12E+03$_{1.31E+00}$	**1.07E+03**$_{5.06E+01}$
F27	5.11E+02$_{1.11E+01}$	5.11E+02$_{1.16E+01}$	5.25E+02$_{9.21E+00}$	**5.10E+02**$_{9.52E+00}$	5.24E+02$_{5.04E+00}$	5.23E+02$_{1.05E+01}$
F28	4.60E+02$_{6.84E+00}$	4.59E+02$_{0.00E+00}$	4.59E+02$_{1.19E+01}$	4.60E+02$_{6.84E+00}$	4.73E+02$_{4.30E+00}$	**4.54E+02**$_{0.00E+00}$
F29	3.63E+02$_{1.32E+00}$	3.67E+02$_{1.46E+01}$	3.53E+02$_{9.78E+00}$	3.58E+02$_{1.78E+01}$	**3.53E+02**$_{1.10E+01}$	3.60E+02$_{1.26E+01}$
F30	6.01E+05$_{2.99E+04}$	6.01E+05$_{2.64E+04}$	6.58E+05$_{7.24E+04}$	**5.97E+05**$_{2.38E+04}$	6.57E+05$_{5.66E+04}$	6.33E+05$_{3.45E+04}$

literature have a strong instructive significance, especially for different problems, which is further help to narrow down the scope of the selection of parameters. However, the first method does not provide accurate values. In the second method, the optimal calibration is obtained by testing the potential value of each parameter. Furthermore, in the second method, the sensitivity of the parameters was analyzed by using several statistical methods. Meanwhile, the second method consumes a great computational time due to testing all the combinations of parameters, and the accuracy is affected by the potential values of the parameters. Therefore, the advantages of these two methods are recombined in this study.

To investigate the performance of the LSHADE-CLM algorithm with different parameter combinations, the proposed algorithm was tested on the part of benchmarks of the CEC2015 real parameter single objective competition [49]. The Definitions of basic functions were shown in Table 3. The CEC2015 benchmark consists of four different categorizations: Unimodal, Multimodal, Hybrid and Composition. The details are described in Tables 4–7.

Where $F_i^* = F_i(x^*)$, $o_i = [o_{i1}, o_{i2}, o_{iD}]^T$ is the shifted global optimum, which is randomly distributed in $[-80, 80]^D$. M denote rotation matrix, different rotation matrix is assigned to each function and each basic function. Each function has a shift data for CEC'15. All test functions are shifted to o and scalable. In CEC'15 and CEC'17, the same search ranges and dimension range are defined for all test functions as $[-100, 100]^D$ and $D = 10, 30, 50, 100$, respectively. The JAVA, C and Matlab codes for CEC'15 test suite can be downloaded from the website give below: https://github.com/P-N-Suganthan/CEC2015.

The performance of LSHADE-CLM is depends on the parameter settings in the mutation operation, the crossover operation and the selection operation. The LSHADE-CLM algorithm has four critical parameters: Np (population size), μF (the memory mutation factor selected from S_F); μCr (the memory crossover rate selected from S_{Cr}), p (determines how many solutions can be utilized in updating the population distribution) , and $\mu freq$ (the frequency of the sinusoidal function from S_{freq}). Generally, Np was set to the maximum species count. First, the selection range of each parameter in LSHADE-CLM was determined according to the previous literature [32,33] and some preliminary experiments. Several values for a factor were determined by trial and error. Then, to find the best levels for the critical factors, the design of experiment (DOE) approach [50] was employed. The obtained experimental results were analyzed by the Analysis of Variance (ANOVA) technique to determine the main effects and interaction between parameters and to establish the best combination of parameters.

The levels for each factor were set as follows: $\mu F \in \{0.3, 0.4, 0.5\}$, $\mu Cr \in \{0.5, 0.6, 0.7, 0.8\}$, $\mu freq \in \{0.2, 0.3, 0.4, 0.5\}$ and $p \in \{0.4, 0.5, 0.6\}$. These parameters above yield a total of $3 \times 4 \times 4 \times 3 = 144$ different configurations for the LSHADE-CLM algorithm. Then, the DOE approach was employed to study the influence of parameters on the performance of the LSHADE-CLM algorithm. Each configuration was executed 15 times for each instance. As a result, a total of $15 \times 144 \times 15 = 32{,}400$ results were generated in the experiment.

The experimental results were analyzed by ANOVA. In the experiment, three main hypotheses, i.e., normality, homogeneity of variance and independent of the residuals, were checked and accepted. Note that, the F-ratio, which was a clear indicator of significance when the p values, were less than the confidence level. The larger the F-ratio was, the more effect the factor has on the response variable. Moreover, the interactions between more than two factors were not considered owing to the F-ratio have not important effect on the parameter selection. The ANOVA results were reported in Table 8.

In Table 8, the p value of the parameters μF and $\mu freq$ were less than the $\alpha = 0.05$ confidence level, which means that these parameters were important and influenced the performance of the LSHADE-CLM algorithm. Meanwhile, $\mu freq$ has the largest F-ratio among all the candidates, which indicates that $\mu freq$ was the most significant parameter that affects the performance of

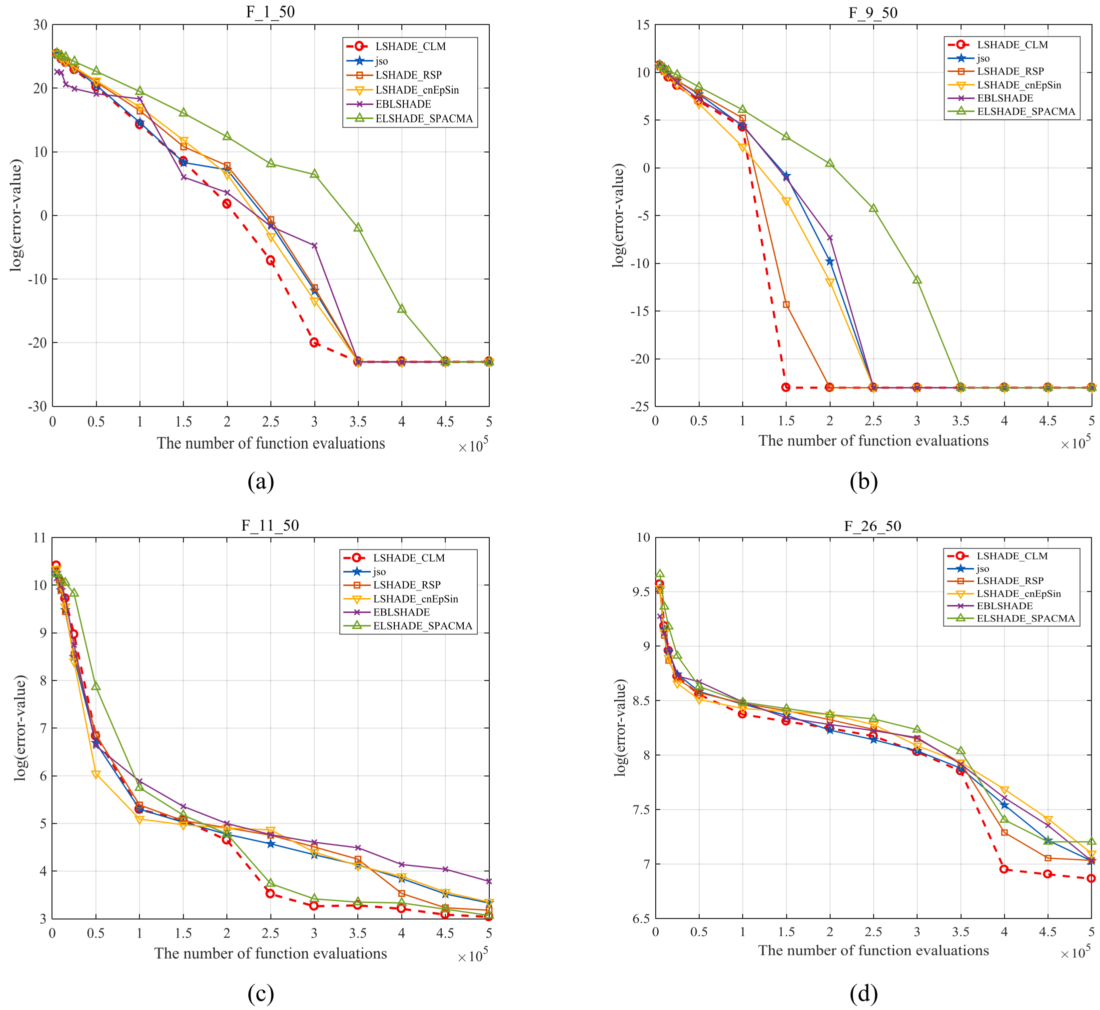


Fig. 14. Rankings for $D = 100$.

the proposed LSHADE-CLM algorithm. The main effects plot of all parameters was given in Fig. 7. It was clearly observed that the choices are $\mu F = 0.4$, $\mu Cr = 0.7$, $p = 0.5$ and $\mu freq = 0.3$.

However, the 2-level interactions of all parameters were shown in Table 8. Meanwhile, the pvalue of the $\mu F * \mu freq$ and $\mu Cr * \mu freq$ in Table 8 were less than 0.05. Thus, it was necessary to consider the interactions between parameters μF and $\mu freq$. The interaction plot of μF and $\mu freq$ was showed in Fig. 8. In Fig. 8, when $\mu freq = 0.3$ and $\mu Cr = 0.6$, the mean value of LSHADE-CLM was at a minimum. Meanwhile, when $\mu freq = 0.4$ and $\mu F = 0.4$, the mean value of LSHADE-CLM was at a minimum. According to Figs. 7 and 8, the best value of $\mu freq$ was 0.3.

However, although the optimal values and interaction plot for each parameter are shown in Figs. 7 and 8, but it is still not confirmed whether the lower p obtain better performance. Therefore, an additional experiment was performed to find the optimal value of p. In the additional experiment, set $p \leq 0.5$, since 0.5 is the current optimal value for p in the previous experiment. The potential values of p are set $\{0.2, 0.3, 0.4, 0.5\}$. The values of others parameters were $\mu F = 0.4$, $\mu Cr = 0.7$ and $\mu freq = 0.3$. These parameters yield a total of 4 different configurations for the LSHADE-CLM algorithm. The new 4 instances were generated in the same method in Section 4.1. As a result, a total of $4 \times 15 \times 15 = 900$ results are generated in this experiment. All of the results mentioned above are also analyzed by the ANOVA technique. The analysis results are reported in the ANOVA plots with 95% confidence intervals in Fig. 9.

In Fig. 9, the experiment result show that the best values of parameter p is $p = 0.5$, which is in consistent with Fig. 7. Based on the analysis above, the optimum combination of parameters in the proposed LSHADE-CLM are suggested as follows: $\mu F = 0.4$, $\mu Cr = 0.7$, $\mu freq = 0.3$ and $p = 0.5$.

4.2. Effectiveness of algorithm components

The LSHADE-CLM without dimensional reset strategy (LSHADE-CLM-ND) and the LSHADE-CLM without collaborative mutation strategies (LSHADE-CLM-NCM) were compared with the proposed LSHADE-CLM algorithm to test the validity of the algorithm components. The average results of LSHADE-CLM-ND algorithm, LSHADE-CLM-NCM algorithm and LSHADE-CLM algorithm in different function types of CEC2017 benchmark were shown below, and the best result for each function was shown in boldface.

The effectiveness of the proposed components was demonstrated by experimental results, as shown in Tables 9 and 10. The proposed LSHADE-CLM algorithm is superior to the LSHADE-CLM-ND and the LSHADE-CLM-NCM algorithms in both high and low dimension problems. As mentioned earlier, the multiple strategy algorithms perform better performance than single strategy algorithms because of the diversity of the benchmark functions, the LSHADE-CLM-ND algorithm outperforms the LSHADE-CLM-NCM algorithm. As above, the performance of LSHADE-CLM was effectively improved by introducing the two proposed strategies. The details of the results in Tables 9 and 10 can be found in supplementary material.

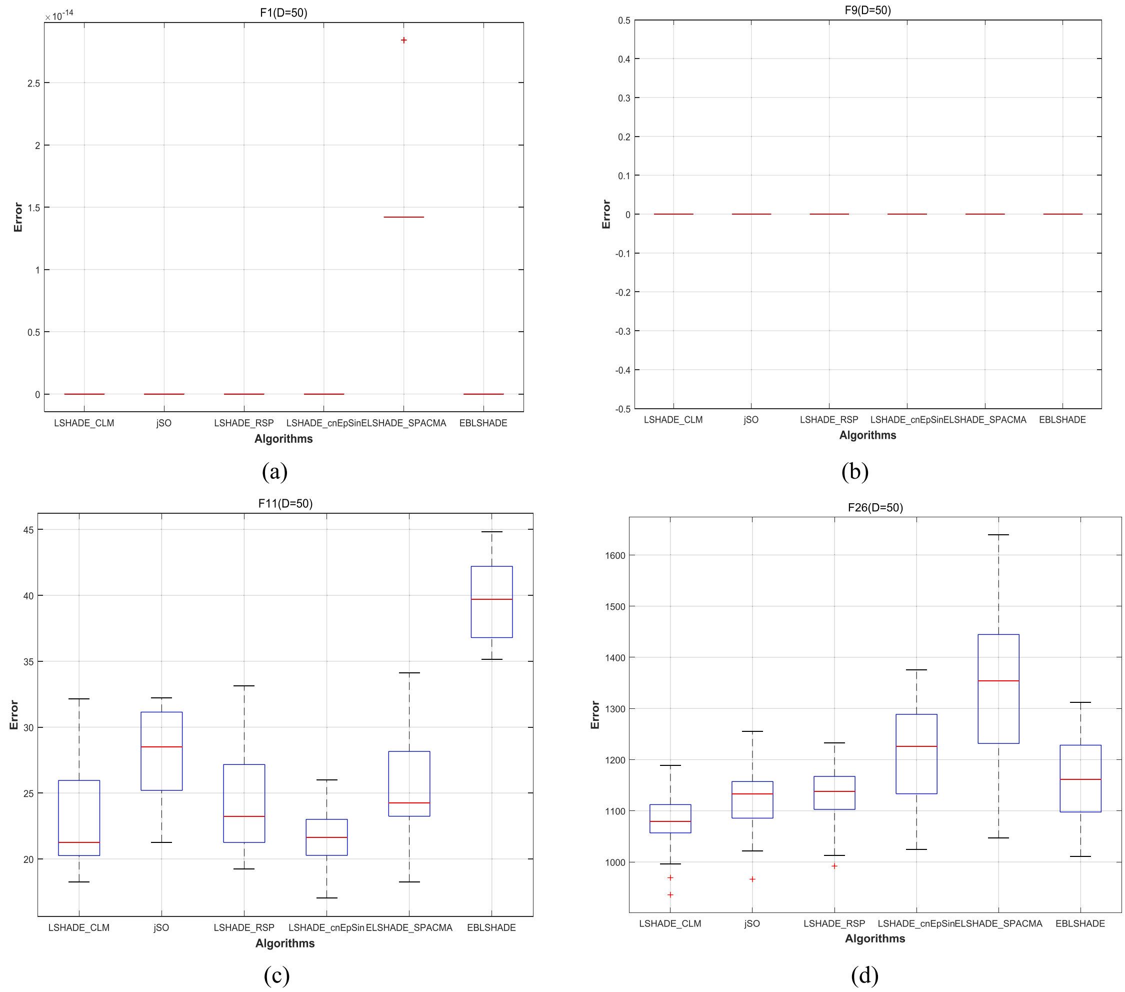


Fig. 15. Convergence curves of some typical benchmark functions ($D = 50$).

4.3. Algorithm complexity

In general, the complexity of the algorithm is the number of floating-point operations and iterations/ recursions required. For meta-heuristic algorithms, the nfes is also a desirable indicator of complexity. In Algorithm 3, the complexity of the LSHADE-CLM algorithm is affected by the following four parts: mutation operation, crossover operation, selection operation and reset operation. The first part is mutation operation, each candidate solution cost Dth to generates trial vector, the computational complexity of the mutation operation is $O(D)$. The second part is crossover operation, the complexity of which is decided by the crossover rate Cr and dimension D of the individuals in the population. Thus, the computational complexity of the crossover operation is $O(D \times Cr)$. The crossover rate $Cr \in [0, 1]$, the complexity of the second part in the worst case is $O(D)$. The third part is selection operation, and the complexity is $O(1)$. The fourth part is reset operation. The reset operation is constructed based on two main factors, the dimension D and the reset rate $\sigma(g, i)$, so the complexity of the fourth part is $O(D \times \sigma_{g,i})$. The range of reset rate $\sigma_{g,i} \in [0, 1]$, in the worst case, the complexity of the fourth part is O(D). From the LSHADE-CLM procedure, the complexity of LSHADE-CLM is at least $O((Cr + 1 + 1 + \sigma_{g,i}) \times D \times nfes)$. According to the results of parameter calibration in Section 4.1, the mutation operation has maximum complexity among all parts of the proposed algorithm. According to the above analysis, the maximum computational complexity of LSHADE-CLM can be calculated as follows:

$$\begin{aligned} &O(D, Cr, 1, \sigma_{g,i}) = O(D) + O(D \times Cr) + O(1) + O(D \times \sigma_{g,i}) \\ &= O\left((Cr + 1 + 1 + \sigma_{g,i})\right) \times D \times nfes \\ &\approx O(D) + O(D \times 1) + O(1) + O(D \times 1) \\ &\approx O(4 \times D \times nfes) \end{aligned}$$

All experiments were implemented and executed using Visual Studio 2017 running on a PC with 3.4 GHz Intel(R) Core i7-7700 CPU, 16GB RAM and a 64-bit OS to ensure that the comparison was fair. The computational complexity was calculated as described in [47].

The complexity of the algorithm on 10, 30 and 50 dimensions is shown in Table 11. $T1$ is the time to execute 200,000 evaluations of benchmark function f_{18} by itself with D dimensions, and $T2$ is the time to execute LSHADE-CLM algorithm with 200,000 evaluations of f_{18} in D dimensions.

4.4. Analysis and discussion

The LSHADE-CLM algorithm was run 51 times for each test problem with several function evaluations equal to 10,000D. The error was set to 0, if the difference between the best solution and the optimal solution is less than 10^{-8}. The statistical results of LSHADE-CLM on $D = 10, 30, 50, 100$ are presented in Table 12, 14, 16 and 18, respectively. Each table gives the mean and the standard deviation of the error value between the best fitness values found in each run and true optimal value over the 51 runs.

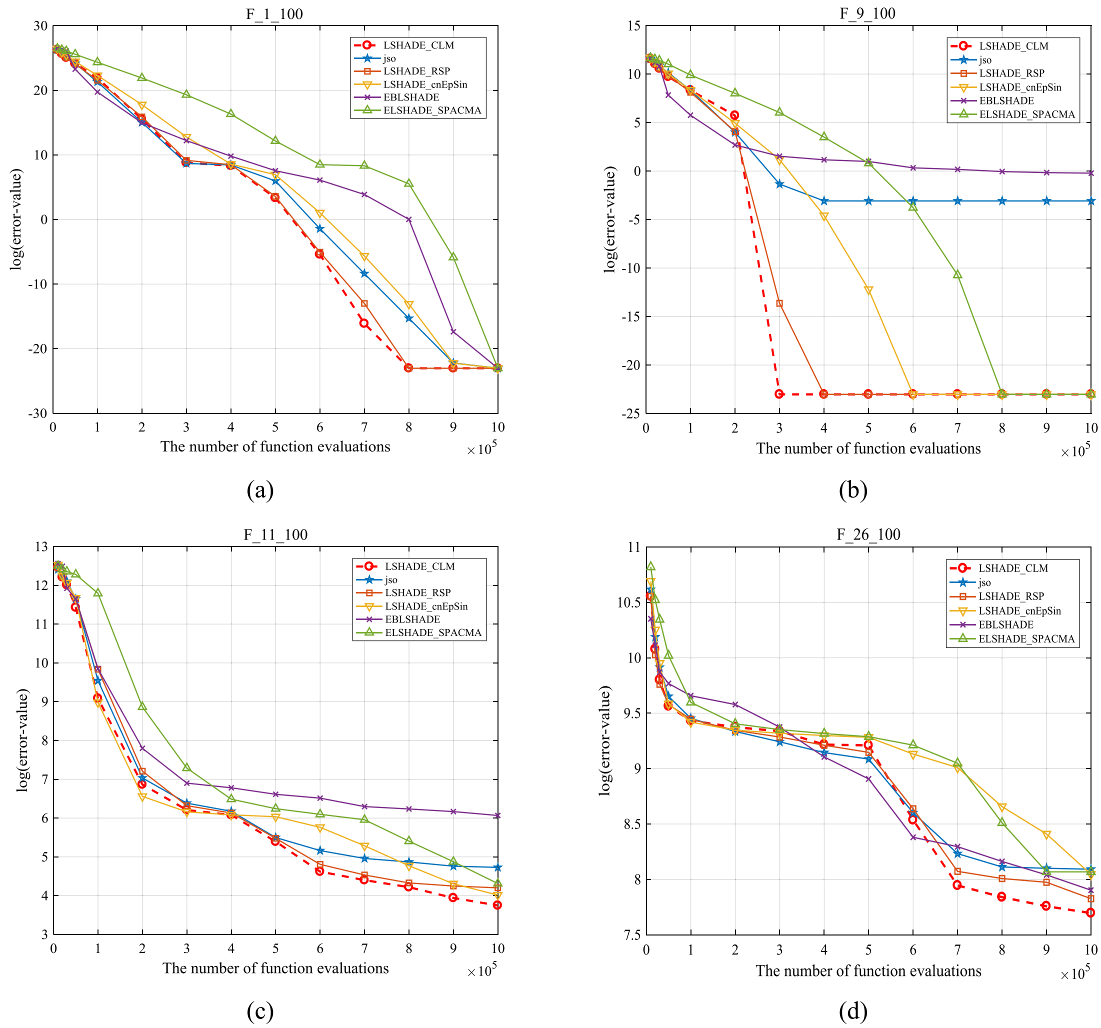


Fig. 16. Convergence curves of some typical benchmark functions ($D = 100$).

The convergence curves of f_1, f_9, f_{11} and f_{26} with $D = 50{,}100$ are shown in Figs. 14 and 16 for LSHADE-CLM and the comparison algorithms. As shown in Figs. 14 and 16, the LSHADE-CLM algorithm has a faster convergence rate than the others algorithms. Meanwhile, at the end of the program execution, the value found by the LSHADE-CLM algorithm was superior to another compared algorithm. Compared with the other algorithms, a collaborative mutation strategy was introduced in the LSHADE-CLM algorithm to protect the diversity of population was not lost. The selection pressure of individuals was balanced by the "$DE/current - to - pbetter/r$" strategies. The algorithm was guaranteed by the proposed mechanisms to gain the global search capability by sacrificing part of the convergence speed. Meanwhile, the diversity of the population was insured by the reset strategy at the dimensional level and helped the LSHADE-CLM algorithm escape a local optimal solution and avoid convergence stagnation. Therefore, the convergence speed of LSHADE-CLM was accelerated late in the program execution. Figs. 14 and 16 also draw a similar conclusion. The box-plots of f_1, f_9, f_{11} and f_{26} with $D = 50{,}100$ are shown in Figs. 15 and 17, respectively.

The results of the Friedman test are shown in Figs. 10–13. The Bonferroni–Dunn's test was introduced to calculate the critical difference for comparing the differences of the algorithms with $\alpha = 0.05$ and $\alpha = 0.1$. The ranking of the algorithms was obtained through the Friedman test and Bonferroni–Dunn's test. To detect the significant difference between LSHADE-CLM and the other compared algorithms, the Wilcoxon signed-rank test was introduced, and the algorithms in pairs were compared, and the results are shown in Tables 13, 15, 17 and 19, respectively. In the Wilcoxon signed-rank test, R+ indicates that LSHADE-CLM was superior to the compared algorithm; on the contrary, R- means the comparison algorithm was better than LSHADE-CLM, and R means the comparison algorithm has a similar value to LSHADE-CLM.

As shown in Tables 17 and 19, the LSHADE-CLM had a desirable performance in high dimensional problems. The reason for this result was that, the number of similar individuals in the population of each dimension increased as the dimension increased. LSHADE-CLM avoided the deterioration of this situation and was improved by using the reset scheme and collaborative mutation strategy. From the previous experimental series and analysis, the results of the data and figures show that the LSHADE-CLM was the desirable algorithm among the compared algorithms on the CEC2017 benchmark.

In various variants of DE, the parameter setting was a key point to improve the performance of DE. In jSO, various parameters in a range were set at different times to produce a desirable value. However, the individuals in the population became similar if the algorithm fell into a local optimal solution. The mutation strategy and crossover operation were unprofitable to generate a new individual, and the population was stagnation. For example, when the CoDE algorithm fell into a local optimal solution,

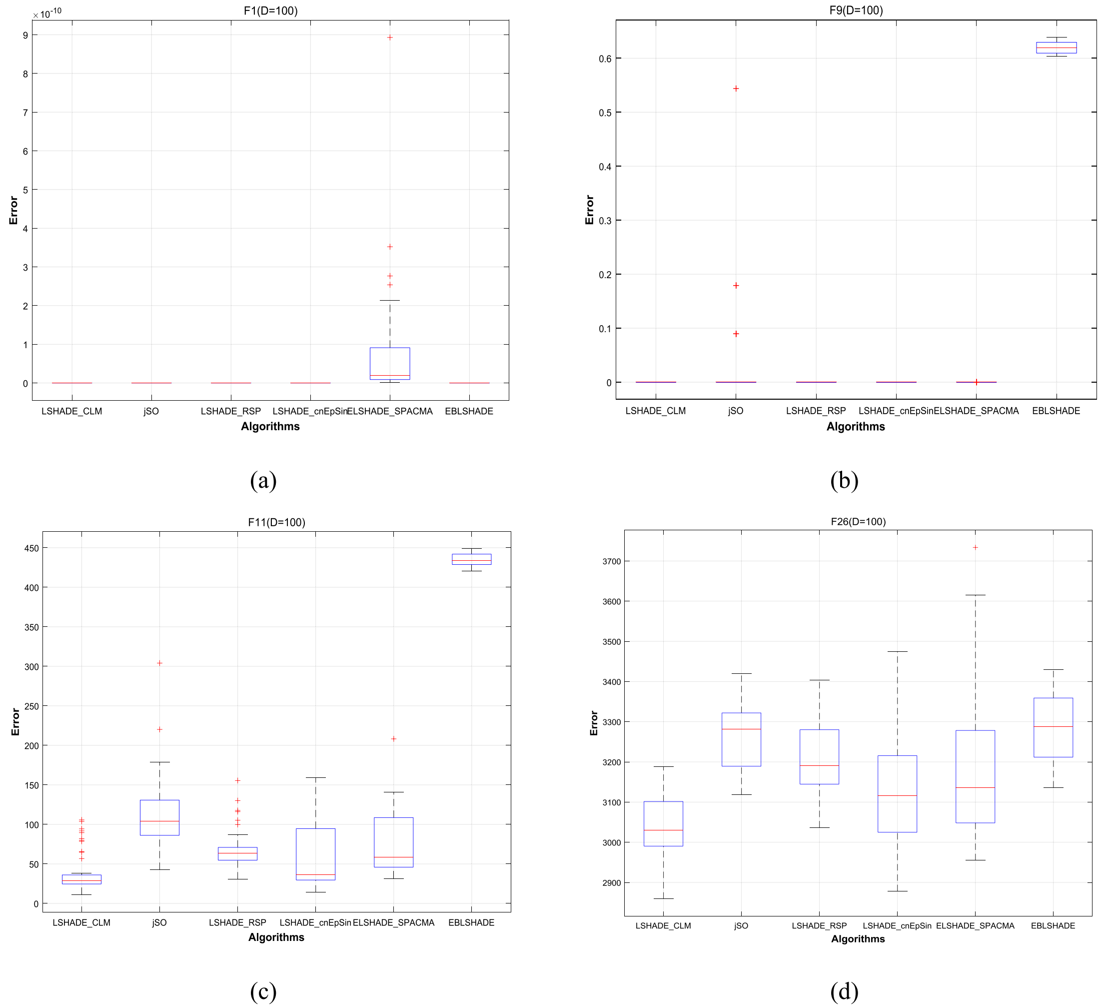


Fig. 17. Box plot for Error of $D = 100$.

although the various mutation strategies were applied in the CoDE algorithm, none of the mutation strategies helped the CoDE algorithm escape a local optimal.

Meanwhile, the diversity of populations at the dimensional level was also important, especially in high dimensional problems. However, different dimensions have different population diversity characteristics, which mean that the overall population diversity was extensive and difficult to handle in certain dimensions. In LSHADE-CLM algorithm, the "$DE/current-to-pbetter/r$" strategy was used to ensure selective pressure in the mutation strategy. A collaborative mutation scheme was introduced in mutation operation. Meanwhile, the parameter adaptive strategies that have a reward mechanism were introduced to the mutation strategy to ensure F self-adaptive according to the experience of the previous generation in the LSHADE-CLM algorithm. Furthermore, the reset strategy at the dimensional level was applied to expedite the LSHADE-CLM algorithm to escape the local optimal solution and guaranteed the diversity of the population at the dimensional level (see Tables 14, 16 and 18).

5. The application of LSHADE-CLM to blocking flowshop scheduling problem

The blocking flowshop scheduling problem (BFSP) is an important scheduling problem and has been widely used in the real-world, e.g., the iron and steel industry, chemical industry, assembly lines and robotic cell, etc. In the BFSP there are no buffers between machines, the job remains in the current machine until the next machine is available for processing [51]. Moreover, existing experiments and literature have shown that BFSP with more than two machines is a typical NP-hard problem.

Table 17
Rankings obtained through the Wilcoxon signed-rank test on 50 dimensions.

Dimension	LSHADE-CLM vs.	R+	R-	R≈	p-value	$\alpha = 0.05$	$\alpha = 0.1$
50	jSO	21	5	4	0.019	**Yes**	**Yes**
	LSHADE-RSP	17	8	5	0.026	**Yes**	**Yes**
	LSHADE-cnEpsin	20	7	3	0.034	**Yes**	**Yes**
	ELSHADE-SPACMA	19	7	4	0.066	No	**Yes**
	EBLSHADE	18	8	4	0.006	**Yes**	**Yes**

5.1. Mathematical model of BFSP

In this study, the objective is to minimize the makespan criterion. Let $\pi = [\pi(1), \pi(2), \ldots, \pi(n)]$ be a permutation sequence. $p_{\pi(i),j}$ is the processing time of job $\pi(i)$ on machine j. $d_{\pi(i),j}$ denotes the departure time of job $\pi(i)$ on machine j.

Table 18
The experimental results of all algorithms on 100 dimensions.

Fun	jSO $Mean_{Std}$	LSHADE-RSP $Mean_{Std}$	LSHADE-cnEpsin $Mean_{Std}$	ELSHADE-SPACMA $Mean_{Std}$	EBLSHADE $Mean_{Std}$	LSHADE-CLM $Mean_{Std}$
F1	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	0.00E+00$_{0.00E+00}$
F2	**8.94E+00**$_{2.42E+01}$	1.27E+$_{048.90E+04}$	9.57E+10$_{6.18E+11}$	1.47E+10$_{3.72E+08}$	8.97E+10$_{7.03E+07}$	2.41E+02$_{1.33E+02}$
F3	2.39E−06$_{2.73E−06}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	1.60E−07$_{4.00E−07}$	1.68E−06$_{2.00E+06}$	5.03E−08$_{1.17E−07}$
F4	1.90E+02$_{2.89E+01}$	1.97E+02$_{1.27E+01}$	1.98E+02$_{8.30E+00}$	2.01E+02$_{8.67E+00}$	**1.54E+02**$_{3.02E+01}$	1.93E+02$_{1.11E+01}$
F5	4.39E+01$_{5.61E+00}$	3.63E+01$_{8.84E+00}$	5.59E+01$_{9.91E+00}$	**1.78E+01**$_{3.85E+00}$	4.06E+01$_{2.31E+00}$	3.29E+0$_{19.77E+00}$
F6	2.02E−04$_{6.20E−05}$	**0.00E+00**$_{0.00E+00}$	6.02E−05$_{2.18E−05}$	**0.00E+00**$_{0.00E+00}$	1.12E−02$_{3.33E−01}$	1.14E−05$_{5.02E−06}$
F7	1.45E+02$_{6.70E+00}$	1.45E+02$_{7.26E+00}$	1.62E+02$_{7.91E+00}$	**1.11E+02**$_{1.48E+00}$	1.40E+02$_{2.31E+00}$	1.37E+02$_{6.86E+00}$
F8	4.22E+01$_{5.52E+00}$	3.63E+01$_{8.01E+00}$	5.35E+01$_{5.39E+00}$	**1.82E+01**$_{3.02E+00}$	4.12E+01$_{2.23E+00}$	3.56E+01$_{7.99E+00}$
F9	4.59E−03$_{1.15E−01}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	**0.00E+00**$_{0.00E+00}$	8.60E−01$_{4.11E−07}$	**0.00E+00**$_{0.00E+00}$
F10	**9.70E+03**$_{6.82E+02}$	1.09E+04$_{7.10E+02}$	1.03E+04$_{5.21E+02}$	1.08E+04$_{9.53E+02}$	1.05E+04$_{2.03E+02}$	1.07E+04$_{6.39E+02}$
F11	1.13E+02$_{4.32E+01}$	6.68E+01$_{2.33E+01}$	4.92E+01$_{3.02E+01}$	7.34E+01$_{4.30E+01}$	4.31E+02$_{2.17+01}$	**3.96E+01**$_{2.87E+01}$
F12	1.84E+04$_{8.35E+03}$	1.13E+04$_{5.12E+03}$	**4.62E+03**$_{6.48E+02}$	7.79E+03$_{2.92E+03}$	2.01E+04$_{3.01E+02}$	4.76E+03$_{1.06E+03}$
F13	1.45E+02$_{3.80E+01}$	1.03E+02$_{3.17E+01}$	1.25E+02$_{3.65E+01}$	1.49E+02$_{3.83E+01}$	2.42E+02$_{7.65E+01}$	**6.02E+01**$_{2.08E+01}$
F14	6.43E+01$_{1.09E+01}$	4.15E+01$_{5.46E+00}$	4.97E+01$_{8.17E+00}$	4.75E+01$_{5.69E+00}$	2.29E+02$_{2.83E+00}$	**3.58E+01**$_{4.07E+00}$
F15	1.62E+02$_{3.81E+01}$	9.21E+01$_{3.16E+01}$	8.99E+01$_{2.83E+01}$	1.08E+02$_{4.34E+01}$	2.53E+02$_{2.66E+01}$	**6.28E+01**$_{3.13E+01}$
F16	1.86E+03$_{3.49E+02}$	1.65E+03$_{3.62E+02}$	**1.22E+03**$_{2.36E+02}$	1.76E+03$_{4.88E+02}$	1.50E+03$_{3.58E+02}$	1.53E+03$_{3.24E+02}$
F17	1.28E+03$_{2.38E+02}$	1.20E+03$_{2.42E+02}$	**9.32E+02**$_{1.74E+02}$	1.27E+03$_{3.45E+02}$	1.10E+03$_{5.05E+02}$	1.09E+03$_{2.67E+02}$
F18	1.67E+02$_{3.65E+01}$	1.23E+02$_{3.31E+01}$	7.79E+01$_{1.99E+01}$	1.05E+02$_{2.56E+01}$	2.33E+02$_{6.20E+01}$	**6.02E+01**$_{1.24E+01}$
F19	1.05E+02$_{2.01E+0}$1	5.49E+01$_{5.56E+00}$	5.55E+01$_{6.05E+00}$	6.05E+01$_{7.55E+00}$	1.75E+02$_{2.17E+00}$	**3.82E+01**$_{5.18E+00}$
F20	1.38E+03$_{2.43E+02}$	1.54E+03$_{2.65E+02}$	1.08E+03$_{2.16E+02}$	**9.28E+02**$_{2.54E+02}$	1.52E+03$_{2.66E+02}$	1.76E+03$_{2.73E+02}$
F21	2.64E+02$_{6.43E+00}$	2.55E+02$_{8.74E+00}$	2.77E+02$_{6.94E+00}$	2.96E+02$_{1.65E+01}$	2.57E+02$_{9.31E+01}$	**2.48E+02**$_{9.12E+00}$
F22	1.02E+04$_{2.18E+03}$	1.16E+04$_{7.80E+02}$	1.04E+04$_{5.30E+02}$	**9.70E+03**$_{1.20E+03}$	1.12E+04$_{1.11E+02}$	1.06E+04$_{7.89E+02}$
F23	5.71E+02$_{1.07E+01}$	5.70E+02$_{1.01E+01}$	5.98E+02$_{7.69E+00}$	6.03E+02$_{2.19E+01}$	5.71E+02$_{1.63E+01}$	**5.59E+02**$_{9.66E+00}$
F24	9.02E+02$_{7.89E+00}$	9.06E+02$_{9.28E+00}$	9.17E+02$_{1.34E+01}$	9.32E+02$_{1.90E+01}$	9.03E+02$_{2.43E+01}$	**8.73E+02**$_{7.80E+00}$
F25	7.36E+02$_{3.53E+01}$	7.21E+02$_{4.04E+01}$	6.84E+02$_{4.34E+01}$	7.00E+02$_{3.99E+01}$	7.47E+02$_{3.17E+01}$	**6.58E+02**$_{3.46E+01}$
F26	3.27E+03$_{8.02E+01}$	3.20E+03$_{9.37E+01}$	3.11E+03$_{1.22E+02}$	3.24E+03$_{2.19E+02}$	3.23E+03$_{2.79E+02}$	**3.04E+03**$_{8.00E+01}$
F27	5.85E+02$_{2.17E+01}$	5.82E+02$_{1.58E+01}$	5.89E+02$_{1.31E+01}$	5.62E+02$_{1.75E+01}$	6.14E+02$_{1.53E+01}$	**5.53E+02**$_{1.77E+01}$
F28	5.27E+02$_{2.73E+01}$	5.23E+02$_{2.11E+01}$	5.15E+02$_{2.20E+01}$	5.21E+02$_{2.38E+01}$	5.31E+02$_{4.36E+01}$	**5.14E+02**$_{1.86E+01}$
F29	1.26E+03$_{1.91E+02}$	1.28E+03$_{1.98E+02}$	1.12E+03$_{1.49E+02}$	1.21E+03$_{1.98E+02}$	**1.08E+03**$_{1.03E+01}$	1.14E+03$_{2.09E+02}$
F30	2.33E+03$_{1.19E+02}$	2.31E+03$_{1.38E+02}$	2.36E+03$_{1.44E+02}$	2.25E+03$_{1.11E+02}$	2.37E+03$_{1.88E+02}$	**2.19E+03**$_{8.18E+01}$

Table 19
Rankings obtained through the Wilcoxon signed-rank test on 100 dimensions.

Dimension	LSHADE-CLM vs.	R+	R-	R≈	p-value	$\alpha = 0.05$	$\alpha = 0.1$
	jSO	24	5	1	0.019	**Yes**	**Yes**
	LSHADE-RSP	25	3	2	0	**Yes**	**Yes**
100	LSHADE-cnEpsin	20	8	2	0.285	No	No
	ELSHADE-SPACMA	22	6	2	0.004	**Yes**	**Yes**
	EBLSHADE	23	3	4	0	**Yes**	**Yes**

To conveniently describe the mathematical model of BFSP, the mathematical program is formalized as follows.

$$d_{\pi(1),0} = 0 \tag{42}$$

$$d_{\pi(1),j} = d_{\pi(1),j-1} + p_{\pi(1),j} \qquad j = 1, \ldots, m \tag{43}$$

$$d_{\pi(1),0} = d_{\pi(i-1),1} \qquad i = 2, \ldots, n \tag{44}$$

$$d_{\pi(1),j} = \max\{d_{\pi(1),j-1} + p_{\pi(1),j}, d_{\pi(1),j+1}\} \quad i = 2, \ldots, n \qquad j = 1, \ldots, m-1 \tag{45}$$

$$d_{\pi(1),m} = d_{\pi(1),m-1} + p_{\pi(1),m} \qquad i = 1, \ldots, n \tag{46}$$

where $d_{\pi(1),0}$ denotes the starting time of processing. The makespan of the schedule π is $C_{\max}(\pi) = \max_{i=1,2,n} d_{\pi(i),m} = d_{\pi(n),m}$.

Considering the following machine $m = 3$ and job $n = 3$, the processing times of J_1, J_2 and J_3 on each machine are $J_1 = (3, 2, 1)$, $J_2 = (1, 3, 2)$ and $J_3 = (2, 1, 3)$, respectively. The makespan of schedule sets $\pi_1 = \{J_1, J_2, J_3\}$ in BFSP is 13. The Gantt chart of π_1 is shown in Fig. 18.

5.2. Experimental settings and tested methods

Because of the continuous nature of the original DE algorithm, the traditional DE algorithm cannot be directly used to solve the combinational optimization problem with discrete characteristics. Therefore, in discrete LSHADE-CLM, the LOV rule is used to represented individuals as discrete job permutations to help the algorithm to work directly in the discrete domain. More details about LOV rule can be found in [54]. Moreover, the results were measured by the index of average relative percent deviation (ARPD) which is calculated as follows.

$$ARPD = \frac{1}{R}\sum_{i=1}^{R}\frac{C_i - C_{\min}}{C_{\min}} \times 100 \tag{47}$$

where R is the number of runs. C_i is the solution generated by a specific algorithm in the ith experiment for a given instance. $C_{\min}$ is the minimum makespan found by all algorithm. Obviously, the smaller the value of ARPD is, the better the performance of the algorithm is.

5.3. Optimization results of BFSP

In this section, the performances of the proposed algorithms are tested on the first 90 of the Taillard's instances [55] for the blocking flowshop scheduling problem. The stopping criterion in this study is set as the maximum elapsed CPU time $M = n \times m \times T$ milliseconds, where T is set to 30. All compared algorithms are independently run 10 times on the function, and each run is started from scratch. The LSHADE-CLM algorithm is compared with two different algorithms, two tabu search (TS) with multi-move approaches algorithm (here denoted as TS+M) and Ron's algorithms (here denoted as RON). The ARPD for each method are listed in Table 21. The parameter combination of the comparison algorithms is shown in Table 20 and more details can be found in [52] and [53].

Each cell within the table is the average relative percent deviation over 100 results. The best solutions found by LSHADE-CLM were shown in Table 22.

In this section, the LSHADE-CLM is used to solve the blocking flowshop scheduling problem. According to Tables 21 and 22, the experiment results show that the proposed LSHADE-CLM algorithm shows excellent performance for solving the BFSP. The

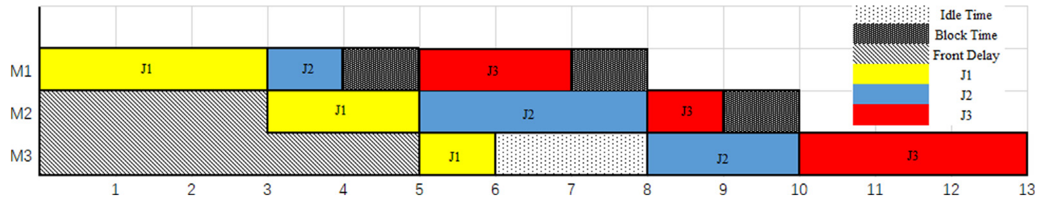


Fig. 18. The Gantt chart of π_1.

Table 20
Parameter settings.

Algorithms	Authors	Parameter settings
TS+M	Grabowski and Pempera [52]	$h = 2 \times LTS, \quad H(l) = 6 \times LTS, \quad H(0) = 0$ $W(l) = \sum_{s=1}^{l} H(s-1) + (l-1) \times h$
RON	Ronconi [53]	$C_{1m} = LC1_1 = \sum_{q=1}^{m} p_{1q}, LC1_i = \max\{LC1_i(k)\}$ $LC2_1 = \sum_{q=1}^{m} p_q^1, \quad LC2_i = \max\{LC2_i(k)\}$

Table 21
The ARPD results of the test algorithms.

$n \times m$	LSHADE-CLM		TS+M		RON	
	ARPD	SD	ARPD	SD	ARPD	SD
20 × 5	**0.236**	0.0504	0.7999	0.1333	0.4625	0.1093
20 × 10	**0.1672**	0.0437	0.6501	0.1571	2.4844	0.276
20 × 20	**0.1281**	0.057	0.3298	0.0986	3.3682	0.5981
50 × 5	**1.7342**	0.4312	4.2682	0.6317	4.8615	0.7433
50 × 10	**0.833**	0.1368	2.925	0.1963	6.6913	1.0092
50 × 20	**0.7642**	0.2376	2.319	0.1428	6.8968	0.9873

effectiveness of LSHADE-CLM is that it combines the advantages of collaborative mutation operation and comprehensive learning mechanism. More specifically, on one hand, the search ability of the algorithm is effectively improved by the collaborative mutation operation. On the other hand, comprehensive learning mechanism is used to help the algorithm find the global optimal solution. Moreover, the reset strategy is introduced in selection operation to ensure the diversity of current population. Therefore, LSHADE-CLM is a competitive algorithm for solving BFSP.

6. Conclusions and future research

In this study, a novel variant algorithm LSHADE-CLM based on the LSHADE algorithm is proposed to improve the performance of the continue optimization problems. In LSHADE-CLM, the comprehensive learning mechanism is introduced into the "$DE/current - to - pbetter/r$" mutation scheme and the population-based covariance matrix is applied to guide the search. The cooperative mechanism integration of different mutation strategies has a significant effect and plays an important role in LSHADE-CLM, which achieve the effect of "$1 + 1 > 2$" instead of simple combination. Meanwhile, a large number of experimental results show that the competitive reward mechanism is an effective method to control parameters and enhance the performance of the algorithm. Moreover, the reset strategy is adopted to avoid stagnation or premature convergence. Finally, the experimental results compared with the state-of-the-arts show that LSHADE-CLM is significantly better than the other compared algorithms. Therefore, LSHADE-CLM is an effective algorithm to solve the continue optimization problems.

For future work, it is necessary to further implement a sensitivity analysis for the parameters of algorithm and to research the selection of mutation strategies. Several directions are suggested. First, more experiments and analysis are needed to determine the optimal combination of parameters and to refine the parameters to multiple decimal places. Second, the proposed LSHADE-CLM algorithm is applied to solve practical problems, such as the job shop scheduling problem. Third, the parameter adaption strategy, which is applied at the dimensional level, is a promising method to enhance the performance of DE and its variants. Fourth, the multiple-operator-based algorithms perform better than the single-operator-based algorithms. Meanwhile, the new mutation strategies and the selection operation are constructed, such as proportional selection or tournament selection, to improve the performance of DE. Finally, it is necessary to find a trade-off between the convergence rate and the reset strategy.

CRediT authorship contribution statement

Fuqing Zhao: Funding acquisition, Investigation, Supervision, review & editing. **Lexi Zhao:** Investigation, Software, Original draft. **Ling Wang:** Methodology, Resources. **Houbin Song:** Visualization, Experiments of the algorithms.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was financially supported by the National Natural Science Foundation of China under grant numbers 61663023 and 61873328. It was also supported by the Key Research Programs of Science and Technology Commission Foundation of Gansu Province, China (2017GS10817), Lanzhou Science Bureau project, China (2018-rc-98), Public Welfare Project of Zhejiang Natural Science Foundation, China (LGJ19E050001), Wenzhou Public Welfare Science and Technology project, China (G20170016), respectively.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.asoc.2020.106609.

Table 22
The best solutions found by TS+M, RON and LSHADE-CLM.

20 × 5	LSHADE-CLM	TS+M	RON	20 × 10	LSHADE-CLM	TS+M	RON
Ta001	**1374**	1387	1384	Ta011	**1698**	**1698**	1736
Ta002	**1408**	1424	1411	Ta012	**1833**	1836	1897
Ta003	**1280**	1293	1294	Ta013	**1659**	1674	1677
Ta004	1479	1451	**1448**	Ta014	**1535**	1555	1622
Ta005	**1341**	1348	1366	Ta015	**1617**	1631	1658
Ta006	1366	1366	**1363**	Ta016	**1590**	1603	1640
Ta007	**1381**	1387	**1381**	Ta017	**1622**	1629	1634
Ta008	**1379**	1388	1384	Ta018	**1731**	1754	1741
Ta009	**1373**	1392	1378	Ta019	**1747**	1759	1777
Ta010	**1283**	1302	**1283**	Ta020	**1782**	**1782**	1847
20 × 20	LSHADE-CLM	TS+M	RON	505	LSHADE-CLM	TS+M	RON
Ta021	**2436**	2449	2530	Ta031	**3033**	3163	3151
Ta022	**2234**	2242	2297	Ta032	**3223**	3348	3395
Ta023	2485	2483	**2560**	Ta033	**3047**	3173	3184
Ta024	**2348**	**2348**	2399	Ta034	**3156**	3277	3303
Ta025	**2437**	2450	2538	Ta035	**3190**	3338	3272
Ta026	**2387**	2398	2467	Ta036	**3201**	3330	3400
Ta027	**2392**	2397	2502	Ta037	**3052**	3168	3228
Ta028	**2328**	2345	2411	Ta038	**3081**	3228	3260
Ta029	**2363**	**2363**	2421	Ta039	**2925**	3068	3104
Ta030	**2323**	2334	2407	Ta040	**3146**	3285	3264
50 × 10	LSHADE-CLM	TS+M	RON	5020	LSHADE-CLM	TS+M	RON
Ta041	**3667**	3776	3913	Ta051	**4517**	4627	4886
Ta042	**3517**	3641	3798	Ta052	**4303**	4411	4668
Ta043	**3508**	3588	3723	Ta053	**4279**	4388	4666
Ta044	**3670**	3786	3885	Ta054	**4377**	4479	4650
Ta045	**3642**	3745	3934	Ta055	**4285**	4359	4475
Ta046	**3624**	3747	3831	Ta056	**4320**	4372	4521
Ta047	**3713**	3778	3957	Ta057	**4329**	4402	4576
Ta048	**3591**	3708	3774	Ta058	**4341**	4444	4688
Ta049	**3549**	3668	3784	Ta059	**4322**	4423	4532
Ta050	**3630**	3729	3928	Ta060	**4430**	4609	4846

References

[1] F.Q. Zhao, S. Qin, Y. Zhang, W.M. Ma, C. Zhang, H.B. Song, A two-stage differential biogeography-based optimization algorithm and its performance analysis, Expert Syst. Appl. 115 (2019) 329–345.

[2] F.Q. Zhao, S. Qin, G.Q. Yang, W.M. Ma, C. Zhang, H.B. Song, A factorial based particle swarm optimization with a population adaptation mechanism for the no-wait flow shop scheduling problem with the makespan objective, Expert Syst. Appl. 126 (2019) 41–53.

[3] K.V. Price, Differential evolution: a fast and simple numerical optimizer, in: Proceedings of North American Fuzzy Information Processing, 1996, pp. 524–527.

[4] M. Liu, W. Huang, Differential evolution enhanced with multiple dimensional scaling, in: 2019 IEEE 9th International Conference on Electronics Information and Emergency Communication, ICEIEC, 2019, pp. 1–4.

[5] N.H. Awad, M.Z. Ali, P.N. Suganthan, Ensemble of parameters in a sinusoidal differential evolution with niching-based population reduction, Swarm Evol. Comput. 39 (2018) 141–156.

[6] R. Mallipeddi, P.N. Suganthan, Q.K. Pan, M.F. Tasgetiren, Differential evolution algorithm with ensemble of parameters and mutation strategies, Appl. Soft Comput. 11 (2011) 1679–1696.

[7] Z. Meng, J. Pan, HARD-DE: Hierarchical archive based mutation strategy with depth information of evolution for the enhancement of differential evolution on numerical optimization, IEEE Access 7 (2019) 12832–12854.

[8] J. Brest, V. Zumer, M.S. Maucec, Self-adaptive differential evolution algorithm in constrained real-parameter optimization, in: 2006 IEEE International Conference on Evolutionary Computation, IEEE, 2006, pp. 215–222.

[9] Y. Wang, Z.X. Cai, Q.F. Zhang, Differential evolution with composite trial vector generation strategies and control parameters, IEEE Trans. Evol. Comput. 15 (2011) 55–66.

[10] R. Tanabe, A. Fukunaga, Success-history based parameter adaptation for differential evolution, in: Evolutionary Computation, 2013, pp. 71–78.

[11] R. Tanabe, A. Fukunaga, Improving the search performance of SHADE using linear population size reduction, in: Evolutionary Computation, 2014, pp. 1658–1665.

[12] A.W. Mohamed, A.A. Hadi, LSHADE with semi-parameter adaptation hybrid with CMA-ES for solving CEC 2017 benchmark problems, in: 2017 IEEE Congress on Evolutionary Computation, CEC, 2017, pp. 145–152.

[13] Z.Y. Meng, J.S. Pan, L.P. Kong, Parameters with adaptive learning mechanism (PALM) for the enhancement of differential evolution, Knowl.-Based Syst. 141 (2018) 92–112.

[14] C. Wang, Y.C. Liu, Q.J. Zhang, H.H. Guo, X.L. Liang, Y. Chen, M.Y. Xu, Y. Wei, Association rule mining based parameter adaptive strategy for differential evolution algorithms, Expert Syst. Appl. 123 (2019) 54–69.

[15] F.Q. Zhao, F.L. Xue, Y. Zhang, W.M. Ma, C. Zhang, H.B. Song, A hybrid algorithm based on self-adaptive gravitational search algorithm and differential evolution, Expert Syst. Appl. 113 (2018) 515–530.

[16] L.Y. Tong, M.G. Dong, C. Jing, An improved multi-population ensemble differential evolution, Neurocomputing 290 (2018) 130–147.

[17] Q.Q. Fan, X.F. Yan, Self-adaptive differential evolution algorithm with discrete mutation control parameters, Expert Syst. Appl. 42 (2015) 1551–1572.

[18] P. Fei, T. Ke, G.L. Chen, Y. Xin, Population-based algorithm portfolios for numerical optimization, IEEE Trans. Evol. Comput. 14 (2010) 782–800.

[19] A. Zamuda, J.D.H. Sosa, Success history applied to expert system for underwater glider path planning using differential evolution, Expert Syst. Appl. 119 (2019) 155–170.

[20] V. Stanovov, S. Akhmedova, E. Semenkin, Selective pressure strategy in differential evolution: Exploitation improvement in solving global optimization problems, Swarm Evol. Comput. 50 (2019) 100463.

[21] S.X. Zhang, W.S. Chan, K.S. Tang, S.Y. Zheng, Restart based collective information powered differential evolution for solving the 100-digit challenge on single objective numerical optimization, in: 2019 IEEE Congress on Evolutionary Computation, CEC, 2019, pp. 14–18.

[22] F.Q. Zhao, Y. Liu, Y. Zhang, W.M. Ma, C. Zhang, A hybrid harmony search algorithm with efficient job sequence scheme and variable neighborhood search for the permutation flow shop scheduling problems, Eng. Appl. Artif. Intell. 65 (2017) 178–199.

[23] F.Q. Zhao, H. Liu, Y. Zhang, W.M. Ma, C. Zhang, A discrete Water Wave Optimization algorithm for no-wait flow shop scheduling problem, Expert Syst. Appl. 91 (2018) 347–363.

[24] W. Gong, Z. Cai, Differential evolution with ranking-based mutation operators, IEEE Trans. Cybern. 43 (2013) 2066–2081.

[25] G.H. Wu, X. Shen, H.F. Li, H.K. Chen, A.P. Lin, P.N. Suganthan, Ensemble of differential evolution variants, Inform. Sci. 423 (2018) 172–186.

[26] Z.Y. Meng, J.S. Pan, K.K. Tseng, PaDE: An enhanced Differential Evolution algorithm with novel control parameter adaptation schemes for numerical optimization, Knowl.-Based Syst. 168 (2019) 80–99.

[27] G.H. Wu, R. Mallipeddi, P.N. Suganthan, Ensemble strategies for population-based optimization algorithmsCA survey, Swarm Evol. Comput. 44 (2019) 695–711.

[28] S. Das, P.N. Suganthan, Differential evolution: A survey of the state-of-the-art, IEEE Trans. Evol. Comput. 15 (2011) 4–31.

[29] J. Zhang, A.C. Sanderson, JADE: Adaptive differential evolution with optional external archive, IEEE Trans. Evol. Comput. 13 (2009) 945–958.
[30] S. Das, S.S. Mullick, P.N. Suganthan, Recent advances in differential evolution C an updated survey, Swarm Evol. Comput. 27 (2016) 1–30.
[31] A.P. Piotrowski, Review of differential evolution population size, Swarm Evol. Comput. 32 (2017) 1–24.
[32] J. Brest, M.S. Maučec, B. Bosković, Single objective real-parameter optimization: Algorithm jSO, in: 2017 IEEE Congress on Evolutionary Computation, CEC, IEEE, 2017, pp. 1311–1318.
[33] V. Stanovov, S. Akhmedova, LSHADE algorithm with rank-based selective pressure strategy for solving CEC 2017 benchmark problems, in: 2018 IEEE Congress on Evolutionary Computation, CEC, 2018, pp. 1–8.
[34] V. Stanovov, S. Akhmedova, E. Semenkin, Selective Pressure Strategy in differential evolution: Exploitation improvement in solving global optimization problems, Swarm Evol. Comput. 50 (2019) 100463.
[35] D.H. Wolpert, W.G. Macready, No free lunch theorems for optimization, IEEE Trans. Evol. Comput. 1 (1997) 67–82.
[36] L. Gui, X.W. Xia, F. Yu, H.R. Wu, R.F. Wu, B. Wei, Y.L. Zhang, X. Li, G.L. He, A multi-role based differential evolution, Swarm Evol. Comput. 50 (2019) 100508.
[37] A.K. Qin, V.L. Huang, P.N. Suganthan, Differential evolution algorithm with strategy adaptation for global numerical optimization, IEEE Trans. Evol. Comput. 13 (2009) 398–417.
[38] N.H. Awad, M.Z. Ali, P.N. Suganthan, R.G. Reynolds, An ensemble sinusoidal parameter adaptation incorporated with L-SHADE for solving CEC2014 benchmark problems, in: 2016 IEEE Congress on Evolutionary Computation, CEC, 2016, pp. 2958–2965.
[39] N. Hansen, The CMA evolution strategy: A comparing review, 2007.
[40] N.H. Awad, M.Z. Ali, P.N. Suganthan, Ensemble sinusoidal differential covariance matrix adaptation with Euclidean neighborhood for solving CEC2017 benchmark problems, in: 2017 IEEE Congress on Evolutionary Computation, CEC, 2017, pp. 372–379.
[41] A. Hadi, A. Wagdy, K. Jambi, Single-Objective Real-Parameter Optimization: Enhanced LSHADE-SPACMA Algorithm, Tech. Rep, King Abdulaziz Univ, Jeddah, Saudi Arabia, 2018.
[42] A.W. Mohamed, A.A. Hadi, K.M. Jambi, Novel mutation strategy for enhancing SHADE and LSHADE algorithms for global numerical optimization, Swarm Evol. Comput. 50 (2019) 100455.
[43] Z.B. Hu, S.W. Xiong, Q.H. Su, Z.X. Fang, Finite Markov chain analysis of classical differential evolution algorithm, J. Comput. Appl. Math. 268 (2014) 121–134.
[44] R.M. Burton, Pointwise properties of convergence in probability, Statist. Probab. Lett. 3 (1985) 315–316.
[45] S.M. Ross, J.J. Kelly, R.J. Sullivan, W.J. Perry, D. Mercer, R.M. Davis, T.D. Washburn, E.V. Sager, J.B. Boyce, V.L. Bristow, Stochastic Processes, Wiley New York, 1996.
[46] J.J. Liang, B.Y. Qu, P.N. Suganthan, Problem Definitions and Evaluation Criteria for the CEC 2013 Special Session on Real-Parameter Optimization, Vol. 201212, Technical Report, Computational Intelligence Laboratory, Zhengzhou University, Nanyang Technological University, Zhengzhou, China, Singapore, 2013, pp. 3–18.
[47] N.H. Awad, M.Z. Ali, J.J. Liang, B.Y. Qu, P.N. Suganthan, Problem Definitions and Evaluation Criteria for the CEC 2017 Special Session and Competition on Single Objective Bound Constrained Real-Parameter Numerical Optimization, Technical Report, Nanyang Technological University Singapore, 2016.
[48] Z.S. Shao, D.C. Pi, W.S. Shao, P.S. Yuan, An efficient discrete invasive weed optimization for blocking flow-shop scheduling problem, Eng. Appl. Artif. Intell. 78 (2019) 124–141.
[49] Q. Chen, B. Liu, Q. Zhang, J.J. Liang, B.Y. Qu, P.N. Suganthan, Problem Definitions and Evaluation Criteria for the CEC 2015 Competition on Learning-Based Real-Parameter Single Objective Optimization, Technical Report201411A, Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore, 2014.
[50] D.C. Montgomery, Design and Analysis of Experiments, John Wiley & Sons, 2017.
[51] N.G. Hall, C. Sriskandarajah, A survey of machine scheduling problems with blocking and no-wait in process, Oper. Res. 44 (1996) 510–525.
[52] J. Grabowski, J. Pempera, The permutation flow shop problem with blocking, A tabu search approach, Omega 35 (2007) 302–311.
[53] D.P. Ronconi, A branch-and-bound algorithm to minimize the makespan in a flowshop with blocking, Ann. Oper. Res. 138 (2005) 53–65.
[54] J.C. Bean, Genetic algorithms and random keys for sequencing and optimization, ORSA J. Comput. 6 (1993) 154–160.
[55] Taillard, Benchmarks for basic scheduling problems, Eur. J. Oper. Res. 64 (1993) 278–285.