



A niching cross-entropy method for multimodal satellite layout optimization design

Jialiang Sun¹ · Xianqi Chen¹ · Jun Zhang² · Wen Yao²

Received: 25 July 2020 / Accepted: 14 February 2021 / Published online: 12 March 2021
© The Author(s) 2021

Abstract

Satellite layout optimization design (SLOD) relies on solving a high-dimensional and multimodal optimization problem, in which there exist multiple global optimal solutions. Existing algorithms for SLOD focus on seeking only one approximate global optimum. However, finding multiple solutions simultaneously could provide more design diversity for the designers. To alleviate this problem, multimodal optimization method is studied for SLOD in this paper, and an improved niching-based cross-entropy method (INCE) is proposed. INCE consists of an improved niching strategy, cross-entropy method-based offspring generation and a cross operator. CEC2013 benchmarks and satellite layout optimization design problem are investigated to verify the validity and feasibility of the proposed INCE. Compared with several state-of-the-art algorithms, the proposed algorithm performs better.

Keywords Multimodal optimization · Cross entropy · Niching method · Satellite layout optimization design

Introduction

Satellite overall design is a typical multidisciplinary design optimization (MDO) problem [1–3]. Satellite layout optimization design (SLOD) is considered as a key step in the satellite MDO [4], which studies how to optimize the place position of satellite objects to ensure meeting the requirement of mission and engineering constraints. Previous work usually takes the moment of inertia as the objective [4]. The smaller moment of inertia means less energy that the whole satellite system consumes. With the increasing of the number of layout components and the shapes of layout components becoming more complex, the solution of SLOD remains

challenging due to its high dimensionality and complicated constraints. What is more, it should be noted that SLOD is a typical multimodal optimization problem, which has multiple optimal solutions simultaneously. How to obtain more and better near optimal layout solutions simultaneously over one single optimization in SLOD remains an open problem.

Recently, to obtain near global optimal scheme for designers, much work about the application of meta-heuristic optimization methods has been devoted to solving SLOD [5] problem. For instance, Cagan et al. [6] applied the simulated annealing algorithm to solve the general three-dimensional component layout. Sun and Teng [7] proposed a two-staged layout method, where the ant colony optimization algorithm is adopted to optimize the detailed layout of satellite module. Huo et al. [8] applied a human-guided genetic algorithm to solve the layout design. Zhang et al. [9] developed a hybrid algorithm integrating particle swarm optimization, genetic algorithm and quasi-principal component analysis. Teng et al. [10] adopted a dual-system cooperative coevolution algorithm based on genetic algorithm to overcome the shortcoming of premature convergence. Shi et al. [11] proposed a modified artificial bee colony algorithm with adding a gauss mutation to improve local search ability. A new energy landscape paving heuristic method, as a kind of Monte-Carlo-based global optimization algorithm, is presented for satellite layout optimization design in [12]. Chen

✉ Wen Yao
wendy0782@126.com

Jialiang Sun
sun1903676706@163.com

Xianqi Chen
chenxianqi12@nudt.edu.cn

Jun Zhang
mcgrady150318@163.com

¹ College of Aerospace Science and Engineering, National University of Defense Technology, Changsha 410073, China

² National Innovation Institute of Defense Technology, Chinese Academy of Military Science, Beijing 100000, China

et al. [4] proposed a practical satellite layout optimization design approach based on the enhanced finite circle method. These researches could obtain only one global optimal design scheme in one single optimization. However, designers usually need multiple global optimal layout schemes or superior local optimal schemes for the design reference. To alleviate this problem, multimodal optimization for SLOD should be developed.

Unlike many multi-objective optimization algorithms that intend to solve multi-objective problem [13–15], the aim of multimodal optimization algorithms is to solve the single-objective problem with multiple global optima. The process of their solutions mainly includes two steps. The first is to divide the whole population into multiple subpopulations. The second is to solve the optimum for each subpopulation using the appropriate algorithm efficiently.

In the first step, in general, there are mainly three types of methods to divide the population into multiple subpopulation. The first is using complicated models including several sub-models to simulate the problem [16]. For instance, Gaussian mixture model (GMM) is constructed to solve the multimodal optimization problem. However, the estimation of parameters in GMM based on expectation maximization algorithm (EM) needs a large amount of computation [16]. What is more, it is still difficult to construct the proper GMM because the number of Gaussian distributions needs to be pre-determined. The second type is based on clustering strategy such as K-Means [17]. However, in this methodology, the number of clustering is difficult to determine adaptively according to the landscape of problem. It may lead to two problems. One is that the individuals in one cluster may be located around multiple peaks. Another one is that the individuals in multiple clusters may be located around one peak, which brings out more calculation cost. Both of them would result in the decreasing of the efficiency of multimodal optimization. The last is based on niching strategy to divide the sampling populations into subpopulations, which is intrinsically a clustering method based on fitness sharing scheme. So far, many meta-heuristic algorithms are combined with niching techniques to solve multimodal optimization problems [18]. However, niching strategies also suffer from various drawbacks such as the sensitivity of parameter including niche radius and clustering size, inferior performance in irregular multimodal surfaces and reliance in fitness landscapes.

In the second step, after dividing the population into multiple groups using the methods above, multimodal optimization would search for the optimum belonging to each group. So, the high efficiency of the adopted algorithm plays a key role in multimodal optimization. Unlike genetic algorithm (GA) [19,20] and particle swarm optimization (PSO) [21,22], CEM proposed by Rubinstein in 1997 [23] is a

kind of probabilistic-based meta-heuristic algorithms, which selects the elite samples to update the probabilistic model. Compared with other algorithms, it possesses faster convergence and less pre-defined parameters. Recently, as a member of the class of meta-heuristic, many improved strategies have been conducted based on traditional CEM for optimization. Generally, current researches mainly focus on the improvement of single optimization methods, which hope to enhance the global search ability of algorithms and decrease the computing cost. For example, Murat and Onur [24] enhanced the efficiency of algorithm in both exploration and exploitation by separating the samples into two parts to calculate the mean value and the standard deviation. P. Lopez and E. Onieva [25] presented a hybrid heuristic of GA and CEM to solve the continuous optimization problem, which show excellent performance for high-dimensionality instances. Dirk P et al. [26] demonstrated the effectiveness of the cross-entropy method for solving difficult continuous multi-extremal optimization problems which still only hopes to find one global optimum. However, there exists no related work to conduct multimodal optimization based on CEM.

Aiming at the multimodal SLOD problem, this paper designs a novel multimodal optimization algorithm called improved niching-based cross-entropy method (INCE). First, aiming at the sensitivity of the parameters in current niching strategies in the first step of the multimodal optimization, an improved radius-adaptive niching strategy based on speciation and crowding clustering is utilized to divide the population. Second, to improve the efficiency of the optimization algorithm in the second step, the improved CEM with elitism strategy and local search is adopted. Third, to ensure the balance of the exploitation ability and the exploration ability of the algorithm, a cross operator between different subpopulations is utilized.

The rest of the paper is structured as follows. In section “Background”, the satellite layout optimization design problem is presented, wherein the optimization formulation and the constraints are described in detail. Then, the basic concepts of relative multimodal optimization are explained including the niching strategy and classical CEM. In section “Improved niching-based cross-entropy method”, the proposed INCE is elaborated with the niching strategy incorporated. In Sect. “CEC2013 benchmark functions”, the evaluation criterion of multimodal optimization is discussed. Then, the feasibility and validity of the proposed method are tested on widely used CEC2013 test function set. In section “14-component satellite layout problem”, the proposed method is adopted to solve a practical satellite layout optimization design problem successfully. The conclusion is discussed in section “Conclusion”.

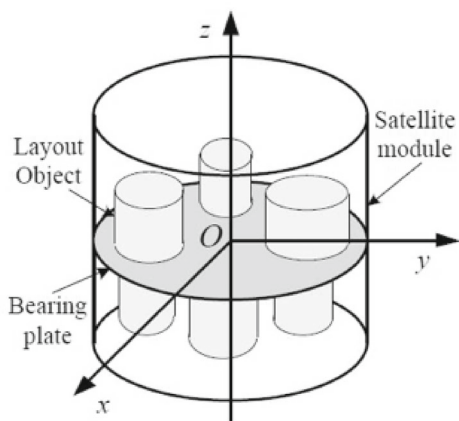


Fig. 1 The illustration of SLOD model

Background

Mathematical model for SLOD

In this section, a three-dimensional satellite layout optimization design model is constructed [5]. The factors including optimization objective, design variables and constraints are discussed.

The whole mathematical model of SLOD could be expressed as follows:

$$\left\{ \begin{array}{l} \text{find } X = \{X_1, X_2, \dots, X_N\} \\ \quad = \{X_i = (x_i, y_i) \mid i = 1, 2, \dots, N\} \\ f(X) = J_{x'}(X) + J_{y'}(X) + J_{z'}(X) \\ \text{s.t.: } \begin{cases} g_1(X) = \sum_{i=0}^{N-1} \sum_{j=i+1}^N \Delta V_{ij} \leq 0 \\ g_2(X) = |x_c - x_e| - \delta x_e \leq 0 \\ g_3(X) = |y_c - y_e| - \delta y_e \leq 0 \\ g_4(X) = |\theta_{x'}(X)| - \delta \theta_{x'} \leq 0 \\ g_5(X) = |\theta_{y'}(X)| - \delta \theta_{y'} \leq 0 \\ g_6(X) = |\theta_{z'}(X)| - \delta \theta_{z'} \leq 0 \end{cases} \end{array} \right. \quad (1)$$

where $(\theta_{x'}, \theta_{y'}, \theta_{z'})$ are the inertia angles, N denotes the number of layout components, $(\delta \theta_{x'}, \delta \theta_{y'}, \delta \theta_{z'})$ are the corresponding allowable angle errors, $J_{x'}(X)$, $J_{y'}(X)$, $J_{z'}(X)$ are the moments of inertia along the x , y , z axis of the whole satellite. (x_c, y_c) represents the real centroid coordinates of system and (x_e, y_e) is the expected centroid. $(\delta x_e, \delta y_e)$ is the maximum permissible centroid error. ΔV_{ij} stands for the overlap volume between the i_{th} object and j_{th} object. More details could be seen in [5]. According to Eq. 1, the SLOD is a single-objective optimization problem with complicated constraints. To handle these six constraints, the penalty function method is adopted to convert the layout optimization problem to an unconstrained optimization problem. The final

objective fitness function could be expressed as

$$F(X) = f(X) + \sum_{i=1}^6 \omega_i \max(0, g_{i+1}(X)), \quad (2)$$

where ω_i stands for the weight factor of the penalty function to control the constraints.

Classical cross-entropy algorithm

The process of CEM as an optimization tool could be divided into two steps:

- 1) Generate random samples according to given probability distribution.
- 2) Update distribution parameters according to elite samples to get optima.

One advantage of Cross-entropy method is that it could utilize any type of probability distribution according to specific problems. Generally, normal distribution is adopted due to the fast convergence when deviation standard is small. In detail, the solution process of CEM for optimization purposes is presented in Algorithm 1.

Algorithm 1: Cross-Entropy method (CEM)

Input:

Input: Population P with N members, elite individuals percent ρ , tolerance error ε , distribution parameters (μ_0, σ_0) ;

Output:

The global optimum y^* ;

- 1 Randomly initialize the population P
 - 2 **while** termination is not meet **do**
 - 3 Generate N independent individuals and calculate their fitness value;
 - 4 Sort the whole population in descending order and select ρN elite individuals;
 - 5 Update distribution parameters according to Eq. 3 and Eq. 4;
 - 6 **end**
 - 7 Return The global optimum y^*
-

$$\mu^{(t)} = \frac{\sum_{k=1}^N I \{S(y_k \geq \gamma)\} y_k}{N_e} \quad (3)$$

$$\sigma^{(t)} = \sqrt{\frac{\sum_{k=1}^N I \{S(Y_k \geq \gamma)\} (x_k - \mu^{(t)})^2}{N_e}} \quad (4)$$

where $I(\cdot)$ is defined as

$$I \{S(y_k \geq \gamma)\} = \begin{cases} 1 & \text{if } S(y_k \geq \gamma) \\ 0 & \text{if } S(y_k < \gamma) \end{cases} \quad (5)$$

Assume maximizing a function S_y over a given set Y , denote the only maximum($S(y^*) = \max_{x \in Y} S(y)$) by

y^* . CEM regards the solution of optimum as a rare probability event and reaches the global optimum by sampling based on pre-defined distribution. First, initial random samples are generated by normal distribution (μ_0, σ_0) , where μ_0 controls the mean value center position of samples and σ_0 controls the boundary of sampling, representing the ability of exploitation and exploration, respectively. Then, we calculate the fitness values $S(y_i)$ and sort them in descending order. γ is the pre-fined threshold to control the selected elite samples.

Niching strategy for multimodal optimization

Generally, to cope with multimodal optimization efficiently, various improved niching methods are proposed and combined with EAs [27]. Currently, speciation and crowding are two main basic niching methods.

$$S_i = \left\{ \text{dist}(x_{\text{seed}}, x_j) = \sqrt{\sum_{d=1}^D (x_{\text{seed}}^d - x_j^d)^2} \leq r \right\}, \quad (6)$$

where S_i is the i_{th} niche, P is the rest population which excludes the individuals in $i - 1$ niches, x_j stands for the individuals in P , x_{seed} is regarded as the best individual in current population P and r is the niche radius. D is the dimensionality of design space. $\text{dist}(a, b)$ represents for the Euclidean distance between a and b .

In crowding, each generated child is compared with the nearest individual from a crowd formed by randomly selecting K parents in the population. Then if the child is better, it will replace the compared parent. This process is formulated as

$$\arg \min_{x_j \in C_i} \text{dist}(x_j, x_{\text{seed}}) = \sqrt{\sum_{d=1}^D (x_{\text{seed}}^d - x_j^d)^2}. \quad (7)$$

Improved niching-based cross-entropy method

Overall, the key step of realizing multimodal optimization faces many challenges. In the first step, the key of variated niching methods is how to divide the populations into multiple subpopulations reasonably. It mainly faces the challenges of determining the parameters of niching methods such as niche radius and maintaining the diversity of the population. Besides, some typology-based methods [28] [29] also bear the large computational burden. In the second step, the choice of the appropriate algorithm has great effects on the performance of multimodal optimization. Its challenges result from

adopting more efficient algorithm that possesses fast convergence and high accuracy simultaneously.

Confronted with these difficulties above, to improve the effect of CEM in multimodal optimization, we develop an INCE with three improved strategies. First, differing from many multimodal optimization methods based on niching method with randomly selected parameter, an improved two-staged niching method is proposed to realize adaptive-radius clustering. Second, to enhance the efficiency of CEM, a new distribution parameters estimation method with local search such as SQP is conducted during the process of evolution. Third, a cross operator is used to generate new individuals between different niches to enhance the exploration ability of the algorithm.

Algorithm 2: Improved niching-based cross-entropy method (INCE)

Input:

Population P with NP members, CEM tolerance error ε , CEM population size n_{cem} , CEM elite sample ratio ρ , elitism archive Ω , iteration number t ;

Output:

Elitism archive Ω including multiple optima;

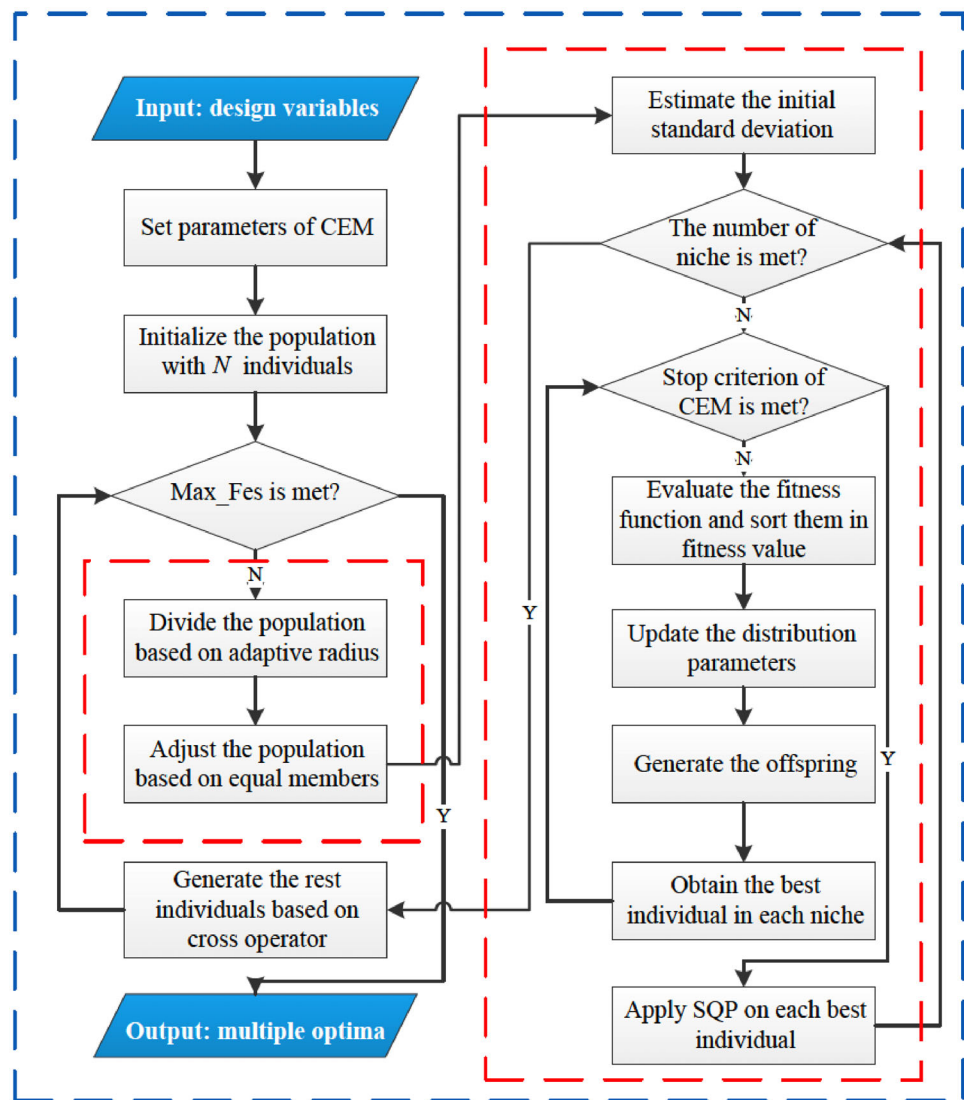
```

1 Randomly initialize the population  $P$ ,  $t = 0$ ;
2 // Clustering the population into multiple niches
3 Using Algorithm 3 to partition the population into  $c$  groups;
4 // Evolution by CEM and find the promising areas
5 if  $t = 0$  then
6   Estimate the distribution parameters  $\sigma_i^0 (i = 1, 2, \dots, c)$  of
   each group according to Eq. 10;
7 end
8  $t = t + 1$ ;
9 for each group do
10  while  $\sigma_i^t < \varepsilon$  do
11    Calculate the fitness value of each individual;
12    Estimate the distribution parameters  $\mu_t$  and  $\sigma_t^t$  according
    to Eq. 9 and Eq. 11, respectively;
13  end
14 end
15 // Local search
16 for each group do
17   Select the best individual  $s_i$  after the CEM evolution;
18   Apply SQP in the initial value of  $s_i$ ;
19 end
20 // Elitism strategy and cross operator
21 Combine all the best individuals  $s_i$  into elitism archive  $\Omega$  and
   calculate the size of best individuals  $n_{best}$ , then keep the  $NP$  best
   ones;
22 Generate  $NP - n_{best}$  individuals according to Eq. 12;
23 Stop if the termination criterion is met. Otherwise go to step 3;
24 Return Elitism archive  $\Omega$  in including multiple optima

```

The flowchart of INCE is shown in Fig. 2. The entire procedure is summarized as Algorithm 2. After discussing the primary idea behind this paper, the concrete description of each component of the algorithm will be elucidated in the following sections.

Fig. 2 The flowchart of INCE



The improved two-staged niching method

Generally, speciation-based niching includes two types of cluster methods, which are performed based on two cluster criteria. One is clustering the population based on size, the other is based on niche radius. In the process of clustering, the population would be sorted in order according to their fitness values. Then, size-based speciation would divide them into multiple niches with equal numbers of individuals. Radius-based speciation would divide them based on their similarities according to Eq. 6.

The first strategy could guarantee the fixed size of individuals in each niche. However, it could cause the loss of diversity because two individuals located in different peaks might be divided into a niche due to similar fitness value. In the second strategy, niche radius is the only one parameter

which influences the performance of the algorithm. If it is too small, the computational cost will increase and the algorithm will face the difficulty of easily getting trapped in the local optimum region. If it is set to a large number, the performance of the well-designed algorithm for multimodal optimization will become worse and the niche strategy will lose its advantage. To solve it, though some dynamic radius strategies are adopted [30], the sensitivity of parameters remains challenging.

In this section, an improved adaptive-radius niching method is proposed. The main idea includes two stages. In the first stage, a more rational adaptive niche radius strategy is proposed according to the landscape of fitness function. In the second stage, to guarantee the diversity of the population, the numbers of each niche would be adjusted by generating new individuals or eliminating poor individuals.

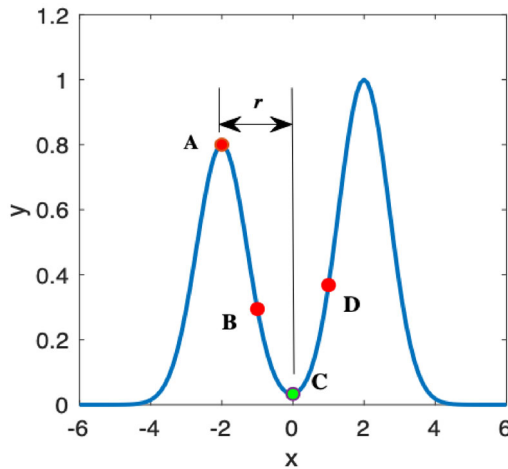


Fig. 3 The illustration the first stage of improved niching method

– First stage: Divide the population based on adaptive dynamic radius

First, the best individual in the current population would be determined as the seed. Second, all other individuals would be sorted in order according to the distance between any individual and the best individual. Third, the fitness values of the neighboring individuals would be compared. The adaptive niching radius could be determined if the poorer individual than the former individuals is founded. The criterion is

$$f(x_i) > f(x_{i-1}). \quad (8)$$

We take a simple function as an example. The process of the first stage in the improved niching strategy could be illustrated in Fig. 3. In it, A, B, C and D are the points that represent the population (for maximization problem). A represents the best individual in current population, while C stands for the poorest one. Denote x_{i-1} , x_i as B and C, respectively, then the distance between A and C would be selected as the niche radius.

– Second stage: Adjust population based on the equal members

After dividing the population into multiple groups, to improve the balance between exploitation and exploration, a kind of population adjusting strategy is adopted. If a group includes farther more individuals than others, then the poor individuals in this group would be eliminated. Reversely, new individuals in small groups should be generated. Detailed process could be seen in Algorithm 3.

Algorithm 3: Improved two-staged niching method

Input:
Input: Population P with NP members;

Output:
the whole population;

- 1 // **First stage: Divide the population based on adaptive dynamic radius**
- 2 Sort the individuals in descending order according to fitness values;
- 3 **while** P is not empty **do**
- 4 Sort the individuals in descending order according to fitness values;
- 5 The best individual is set as a seed and calculate the distances of all the rest individuals and it;
- 6 Sort the corresponding distance in ascending order and calculate the distance r_i of the individual which make the distance shaking firstly with the seed;
- 7 Select the individuals according to Eq. 6 as the species and delete them from P ;
- 8 **end**
- 9 // **Second stage: Population adjusting based on the equal members**
- 10 Calculate the number of species and the population size of each species c ;
- 11 **for** each species **do**
- 12 **if** the population size is more than c **then**
- 13 Sort all the individuals in order and delete the poor individuals;
- 14 **end**
- 15 **if** the population size is less than c **then**
- 16 Select the individual that is farthest with the seed in this species and denote it τ ;
- 17 **while** the population size is more than c **do**
- 18 Randomly generate new individuals within the radius of the species seed;
- 19 If the distance of the new individual with the seed is farther than that of τ and the seed, then delete it from the population;
- 20 **end**
- 21 **end**
- 22 **end**
- 23 Return the whole population

CEM-based offspring generation

After partitioning the population into multiple niches, INCE starts to estimate the probability distribution of each niche. Suppose the population size is NP , and the selected niche radius is r , the total number of niche n_{niche} would be determined according to Eq. 6.

Among the methods of estimating parameters in CEM, Gaussian mixture model (GMM) is adopted to realize multimodal optimization due to the mechanism of including multiple sub-models. However, expectation-maximization algorithm (EM) is widely used to estimate the distribution parameters of GMM for a more precise model, which brings out huge computation cost [16]. In our multimodal optimization, after the whole design space is partitioned multiple subregions, the key step of CEM in each niche is enhancing

the efficiency not constructing complex model to approximate the landscapes. Thus, a new estimation strategy of distribution parameters with elitism strategy and local search is proposed.

The estimation of the other parameters is discussed as follows in detail.

A) The estimation of μ_i

In most improved strategies in CEM, the updating mechanism of μ_i is calculating the mean value vector of selected individual. For complicated objective functions in single optimization, the scheme could have better global search ability because it merges different individual information though sometimes converging slowly. However, the search spaces after dividing the population into many niches are relatively narrow in multimodal optimization. To enhance the convergence speed, the position of the best individual is selected as μ_i , given by

$$\mu_i = x_{i,best} \tag{9}$$

where $x_{i,best}$ is denoted as the best individual in i_{th} niche. By this scheme, the center of sampling could be shifted rapidly to detect more promising area only if one individual far from the μ_i is found better.

B) The estimation of σ_i

In CEM, the standard deviation σ_i plays a vital role in the exploitation and exploration of the algorithm. If σ_i is set as a large number, algorithm could search more design space. Conversely, small σ_i could provide strong exploitation ability.

– The estimation of the initial σ_i^0

After the first clustering by improved niching method, each niche could only exist several individuals, which could not give enough valid standard deviation information. To solve it at first iteration, a sigma coefficient α related to the upper bound and lower bound is adopted to control the distribution of the initial population to sample enough individuals.

$$\sigma_{i,d}^0 = \frac{1}{\alpha} \times (ub(d) - lb(d)) \quad (d = 1, 2 \dots D). \tag{10}$$

– The estimation of the $\sigma_i^0 (t > 0)$

CEM is a kind of evolutionary algorithm based on Monte-Carlo technique, which samples in whole design space. So, the population size of CEM is usually set to large such 2000. However, the feasible solution space is divided into many sub-spaces in INCE, whose corresponding search area is relatively narrow. To save computational resource, generating less individuals such as 50 could still meet the purpose of

optimizing in each niche. To provide enough sample information, all the individuals in the niche would be utilized to calculate the standard deviation, given by

$$\sigma_i^2 = \frac{\sum_{n \in \ell_i} (x - \mu_i)^2}{n_i}, \tag{11}$$

where ℓ_i represents all the individuals in niche, n_i stands for the size of population in i_{th} niche.

C) Local search based on SQP

Some current reports prove that the combination of local search and meta-heuristic algorithms could obviously enhance the performance of meta-heuristic algorithms. Due to the stochastic process in meta-heuristic algorithms, the robustness and accuracy are inferior to the gradient-based method such as SQP. Local search such as SQP also faces the sensitivity of the position of the selected initial individual. Combing the gradient-based method and the meta-heuristic algorithm could solve the dilemma of them, which means the appropriate initial individual could be provided by the process of heuristic process.

The framework of INCE enhances the diversity of the population by sampling in multiple groups. In this way, multiple promising areas covering global optimum could be found. However, on one hand, due to the less populations in each group compared with CEM, INCE does not have strong exploitation ability even though having detecting the global optimum region. On the other hand, pursuing the high accuracy of computation would result in the loss of population diversity. To solve this problem, SQP as a kind of local search is utilized in INCE, which is used to search local optima when the appropriate initial point is given by INCE.

Cross operator

The performance of clustering strategy has great influence on the number of global peaks searched by the multimodal optimization algorithm. What is more, the diversity of the population gets decreased greatly within the evolutionary process. To overcome the loss of the diversity to some extent, a cross operator between the best individuals in each niche is proposed after evolutionary process by CEM. In detail, denote the size of the best individuals in each niche n_{best} , to keep the total population size NP , then the rest $NP - n_{best}$ individuals would be generated according to Eq. 12.

$$x_i^d = x_j^d + \text{rand}() \times (x_k^d - x_j^d) \quad (d = 1, 2, 3 \dots D), \tag{12}$$

where x_j^d, x_k^d represent the best individuals of the j_{th}, k_{th} niche. x_i^d stands for the new generated individual by cross operator. D is the dimension of the design variables.

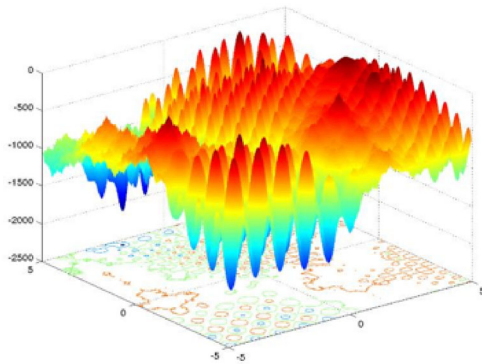


Fig. 4 An illustration of multimodal function

Numerical experiment

CEC2013 benchmark functions

To verify the validation and practicability of INCE proposed in this paper, we compare it with speciation-based DE (SDE) [31], crowding-based DE (CDE) [32] and multimodal estimation of distribution algorithm (LMSEDA) [27] on CEC'2013 multimodal function set containing 20 test functions. There exists mass of local optima around the global optimum in these test functions (Fig. 4). The peak information of the test function set could be seen in [33]. Generally, the evaluation of the multimodal algorithms mainly includes four criteria, namely the Average Number of Function evaluation (ANF), the Peak Rate (PR), the Success Rate (SR) and the Average Distance (ADC) [34].

PR measures the average percentage of all known global optima found over multiple runs.

$$PR = \frac{\sum_{i=1}^{N_n} NPF(i)}{NKP \times NR}, \quad (13)$$

where PR denotes the number of global optima obtained at the end of the i_{th} run, NKP is the number of known global optima, and NR is the number of runs.

SR measures the percentage of successful runs (a successful run is defined as a run where all known global optima are found) out of all runs.

$$SR = \frac{NSR}{NR}, \quad (14)$$

where NSR denotes the number of successful runs.

ANF measures the average computational cost over multiple runs.

$$ANF = \frac{\sum_{i=1}^{NR} NUM(i)}{NR}, \quad (15)$$

where NUM_i denotes the number of function calculation at the end of the i_{th} run.

ADC measures the average computational accuracy over multiple runs.

$$ADC = \frac{\sum_{i=1}^{N_k} |f(x^*) - f(x_i^{opt})|}{NR}, \quad (16)$$

where $f(x^*)$ denotes the actual optima and $f(x_i^{opt})$ denotes the optima found by optimization method at the end of the i_{th} run.

To make fair comparisons, the initial population sizes are all set as the same for four algorithms. The maximum number of fitness evaluations (Max_Fes) for all the algorithms are set the same as the reports [32,34]. In INCE, ρ is set as 0.1. When the procedure is stopped, the number of function evaluation is recorded. In this paper, 1E-5 is chosen as the accuracy level, which means that an optimum is considered to be found if a solution with the distance below 1E-5 to actual optima. The results are averaged over 30 independent runs of each method in our experiments. Other parameters are illustrated in Table 2.

Accuracy and multimodality

The obtained statistical results by different multimodal optimization algorithms are presented in Table 1. The '+' represents the number of the best results among 4 algorithms on 20 test functions. The best ADC, ANF, SR and PR are highlighted in bold. In addition, to have a better view of the performance of INCE, we visualize the final fitness landscape of multiple global optimum selected from the final population when algorithm terminates on nine visual functions, which are shown in Fig. 5. Red points stand for the optima found by INCE (Table 2).

Observing Table 1 and Fig. 5, we can draw the following conclusions:

From the perspective of computational accuracy, we can see that the performance of INCE for all test functions is significantly superior to CDE and SDE. In detail, the average value of ADC over 30 runs could reach over E-14, while that of SDE and CDE could only reach E-4. It mainly results from the local search of SQP in INCE.

From the perspective of the PR value, which is the most important criterion to evaluate the performance of the multimodal optimization algorithms. Overall, we can see that INCE could achieve the best performance over 19 test functions; while, LMSEDA, SDE and CDE could only reach 10, 4 and 4, respectively. F1, F2, F3, F4 and F5 are all low-dimensional functions, and four algorithms all seek all optimum. However, with the increasing of the number of the global optimum, other algorithms such as SDE and CDE are

Table 1 The comparison of INCE, SDE, CDE and LMSEDA on multimodal test functions(Accuracy level 1E-5)

F	INCE				SDE			
	ADC	ANF	SR	PR	ADC	ANF	SR	PR
F1	0	692	1	1	3.87E-02	3.24E+04	0.373	0.657
F2	9.12E-14	733	1	1	7.33E-02	3.65E+04	1	1
F3	2.81E-07	793	1	1	4.98E-02	8.33E+02	1	1
F4	1.42E-14	2874	1	1	3.36E-02	5.00E+04	1	1
F5	1.11E-16	1936	1	1	2.54E-02	1.62E+04	1	1
F6	5.80E-12	1.29E+05	0.9	0.993	5.34E-02	2.00E+05	0	0.056
F7	4.35E-15	2.00E+05	0	0.786	8.57E-04	2.00E+05	0	0.054
F8	7.03E-10	4.00E+05	0	0.84	1.67E-03	4.00E+05	0	0.015
F9	7.32E-12	4.00E+05	0	0.298	9.03E-04	4.00E+05	0	0.011
F10	2.11E-13	17884	1	1	5.82E-03	2.00E+05	0	0.147
F11	2.26E-13	2.00E+05	1	1	6.54E-03	2.00E+05	0	0.314
F12	7.05E-12	7.33E+03	0	0.75	6.65E-04	2.00E+05	0	0.208
F13	4.81E-11	2.00E+05	0	0.667	2.83E-04	2.00E+05	0	0.297
F14	8.36E-13	4.00E+05	0	0.667	1.75E-04	4.00E+05	0	0.216
F15	7.92E-13	4.00E+05	0	0.75	8.87E-04	4.00E+05	0	0.108
F16	1.81E-10	4.00E+05	0	0.667	3.45E-04	4.00E+05	0	0.108
F17	7.76E-11	4.00E+05	0	0.625	5.83E-04	4.00E+05	0	0.076
F18	3.46E-12	4.00E+05	0	0.667	6.27E-04	4.00E+05	0	0.026
F19	7.92E-12	4.00E+05	0	0.5	5.96E-04	4.00E+05	0	0.105
F20	1.81E-12	4.00E+05	0	0.375	8.26E-04	4.00E+05	0	0
+	20	6	7	19	0	0	4	4
F1	9.97E-02	1.88E+02	1	1	–	3.41E+02	1	1
F2	2.43E-02	2.41E+03	1	1	–	1.29E+03	1	1
F3	4.67E-02	1.73E+03	1	1	–	7.21E+02	1	1
F4	3.36E-02	2.51E+04	0.627	0.907	–	8.66E+03	1	1
F5	4.54E-02	3.52E+03	1	1	–	4.79E+03	1	1
F6	6.34E-02	2.00E+05	0	0.461	–	1.44E+05	0.588	0.972
F7	4.77E-4	2.00E+05	0	0.427	–	2.00E+05	0	0.658
F8	3.34E-03	4.00E+05	0	0.011	–	4.00E+05	0	0.556
F9	2.13E-03	4.00E+05	0	0.085	–	4.00E+05	0	0.228
F10	2.50E-03	1.84E+05	0.118	0.843	–	1.35E+04	0.98	0.998
F11	1.44E-03	2.00E+05	0	0.627	–	1.92E+05	0.333	0.879
F12	6.35E-02	2.00E+05	0	0.353	–	1.24E+05	0.902	0.988
F13	1.03E-03	2.00E+05	0	0.611	–	2.00E+05	0	0.667
F14	6.21E-03	4.00E+05	0	0.369	–	4.00E+05	0	0.667
F15	6.87E-03	4.00E+05	0	0.15	–	4.00E+05	0	0.735
F16	7.84E-03	4.00E+05	0	0.082	–	4.00E+05	0	0.667
F17	9.23E-03	4.00E+05	0	0.01	–	4.00E+05	0	0.576
F18	5.23E-04	4.00E+05	0	0.033	–	4.00E+05	0	0.657
F19	4.94E-03	4.00E+05	0	0	–	4.00E+05	0	0.458
F20	5.29E-03	4.00E+05	0	0	–	4.00E+05	0	0.248
+	0	1	4	4	–	6	6	8

* ‘-’ means that the result is not recorded in the corresponding report

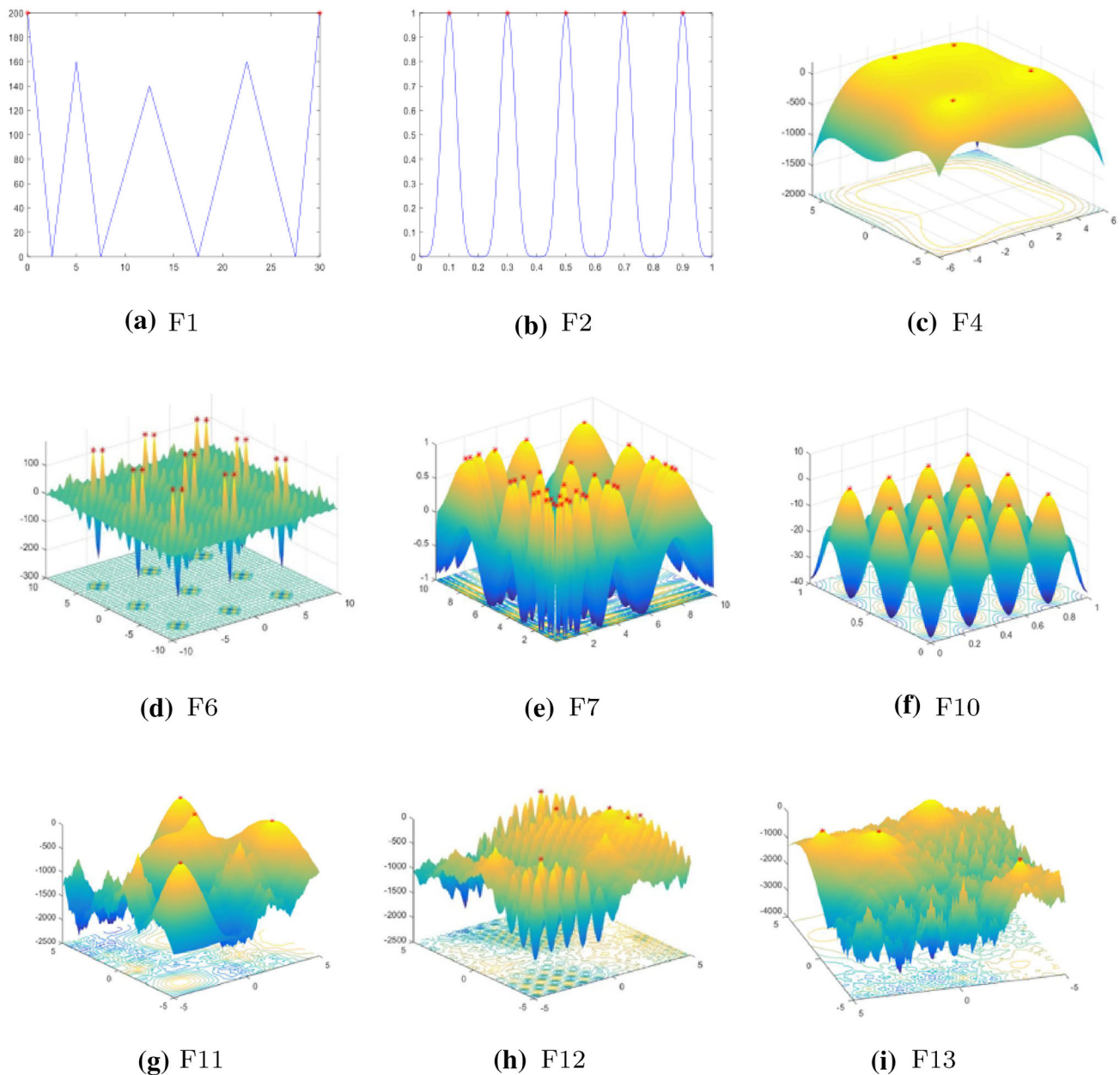


Fig. 5 Results in final fitness landscapes by INCE on nine functions that can be visualized. **a** F1, **b** F2, **c** F4, **d** F6, **e** F7, **f** F10, **g** F11, **h** F12, **i** F13

difficult to seek global optimum, while INCE still possesses the validity of multimodal optimization. LMSEDA defeats INCE only in test function F12, which has lots of local optima around the global optimum.

Effect of parameters of CEM (population size, sigma coefficient and tolerance error)

To observe the sensitivity and robustness of the different parameters of CEM, we conduct numerical experiments on these 20 test functions over 30 runs by changing three parameters including population size, sigma coefficient and

tolerance error. When the parameter is changed, other parameters are set the same as Table 1. Differently, the PR value is selected as the only one criterion. In detail, the population sizes are set as 20, 30, 50 and 100 respectively. The sigma coefficients are set as $1/10, 1/20, 1/30$ and $1/50$. The tolerance errors are set as 0.001, 0.01, 0.05 and 0.1. The results in Table 1 are used to make some comparisons, which is highlighted in italic in Table 3. Other statistical results are shown in Table 3.

From Table 3, we could obtain these conclusions:

- The effect of the population size of CEM

Table 2 Parameter setting

FUNCTION	Max_Fes	<i>NP</i>	<i>n_{cem}</i>	ϵ	α
F1-F5	5.0E+4	80	20	0.1	1/30
F6	2.0E+4	100	20	0.1	1/30
F7	2.0E+4	300	20	0.01	1/30
F8-F9	4.0E+4	300	20	0.1	1/30
F10	2.0E+4	100	20	0.01	1/30
F11-F13	2.0E+4	200	20	0.1	1/30
F14-F15	4.0E+4	200	50	0.01	1/30
F16-F18	4.0E+4	200	100	0.001	1/30
F19-F20	4.0E+4	200	100	0.0001	1/30

Table 3 The PR results of INCE by changing the parameters of CEM

	Population size				Sigma coefficient				Tolerance error			
	20	30	50	100	1/10	1/20	1/30	1/50	0.0001	0.001	0.01	0.1
F1	<i>I</i>	1	1	1	1	1	<i>I</i>	1	1	1	1	<i>I</i>
F2	<i>I</i>	1	1	1	1	1	<i>I</i>	1	1	1	1	<i>I</i>
F3	<i>I</i>	1	1	1	1	1	<i>I</i>	1	1	1	1	<i>I</i>
F4	<i>I</i>	1	1	1	1	1	<i>I</i>	1	1	1	1	<i>I</i>
F5	<i>I</i>	1	1	1	1	1	<i>I</i>	1	1	1	1	<i>I</i>
F6	0.993	0.994	0.994	0.972	0.994	0.998	0.993	0.961	0.982	0.983	0.994	0.993
F7	0.786	0.517	0.467	0.419	0.611	0.621	0.786	0.644	0.781	0.778	0.786	0.646
F8	0.84	0.941	0.93	0.822	0.901	0.891	0.84	0.723	0.794	0.884	0.886	0.84
F9	0.298	0.234	0.195	0.17	0.316	0.298	0.298	0.277	0.273	0.293	0.306	0.298
F10	<i>I</i>	1	0.997	0.997	1	1	<i>I</i>	0.997	1	1	<i>I</i>	0.994
F11	<i>I</i>	1	1	1	1	1	<i>I</i>	1	1	1	1	<i>I</i>
F12	0.75	0.75	0.75	0.625	0.75	0.75	0.75	0.675	0.75	0.75	0.75	0.75
F13	0.667	0.667	0.667	0.667	0.667	0.667	0.667	0.667	0.667	0.667	0.667	0.667
F14	0.667	0.667	0.667	0.667	0.667	0.667	0.667	0.667	0.667	0.667	0.667	0.667
F15	0.375	0.667	0.75	0.75	0.625	0.75	0.75	0.708	0.75	0.75	0.75	0.25
F16	0.167	0.583	0.667	0.667	0.667	0.667	0.667	0.108	0.667	0.667	0.667	0.167
F17	0.125	0.208	0.625	0.625	0.667	0.667	0.625	0.625	0.625	0.625	0.5	0.125
F18	0	0	0.333	0.667	0.667	0.667	0.667	0.667	0.667	0.667	0.5	0.125
F19	0	0	0.375	0.5	0.5	0.5	0.5	0.105	0.5	0.25	0	0
F20	0	0	0.176	0.375	0.375	0.375	0.375	0.25	0.375	0.375	0	0
+	11	12	13	14	18	17	16	8	16	15	16	9

From the left part, it could be seen that the population size of CEM is important for the performance of INCE. As for low-dimensional test functions, it would be better to select a small population size. As for high-dimensional test functions, it would be better to select a big population size. It is because with more individuals, INCE could possess better search ability due to the population-based characteristic.

– The effect of the sigma coefficient of CEM

From the middle part, we could know that the coefficient does not influence the performance of INCE greatly. The algo-

rithms with different coefficients obtain similar PR values. The size of the coefficient is helpful to ensure the balance the exploration ability and exploitation ability of INCE. If it is small, it could possess better exploitation ability. Conversely, it has better exploration ability if the coefficient is large.

– The effect of the tolerance error of CEM

From the right part, we can conclude that the tolerance error of CEM influences the performance of INCE greatly. With the increasement of the dimension of the problem, the

Table 4 The results of INCE with local search and without local search

F	CEM	CEMSQP	F	CEM	CEMSQP
F1	0.0019	0	F11	0.0016	4.631e-13
F2	7.357e-08	4.130e-14	F12	6.392e-06	1.137e-12
F3	1.733e-06	1.715e-07	F13	1.011e-06	3.962e-07
F4	8.118e-06	2.842e-14	F14	0.0012	8.763e-04
F5	1.818e-07	4.441e-16	F15	1.718e-05	1.467e-11
F6	2.916e-04	3.609e-12	F16	2.422e-04	2.105e-12
F7	3.721e-07	3.331e-16	F17	1.211e-04	5.311e-05
F8	0.0081	4.911e-11	F18	2.514e-08	1.427e-08
F9	1.349e-05	2.664e-15	F19	3.098e-07	1.814e-07
F10	1.821e-04	2.487e-13	F20	7.907e-07	4.285e-07

smaller tolerance error should be selected to improve the converging process.

Effect of local search

Similarly, to observe the influence of SQP to the performance of algorithm, we also conduct some experiments on 20 test functions by single CEM with SQP and without it. The tolerance errors of F1–F17 and F17–F20 are set as 0.001 and 0.0001, respectively. The corresponding population sizes of CEM are set as 400 and 800, respectively. The statistical results over 20 runs are shown in Table 4.

From Table 4, we could see that SQP is helpful to enhance the exploitation ability of the multimodal optimization algorithm, particularly in the low-dimensional objective functions. Apart from it, local search in multimodal optimization also accelerates the evolutionary process, which means more computation resources could be saved to detect other areas. Without SQP, the enhancement of the search accuracy of CEM needs the high-leveled tolerance error (ε) to control, which brings out more computation cost compared with SQP. Though the effect of SQP is not apparent in high-dimensional test functions, it also helps to improve the accuracy of the obtained solution.

Effect of cross operator for enhancing diversity

Furthermore, to study the influence of the cross operator to the performance of the algorithm, we also conduct the numerical experiments without the cross operator to make a comparison with INCE. Other parameters are all set as the same as section “CEC2013 benchmark functions”. The statistical results over 30 runs are shown in Table 5.

From Table 5, except F1–F5 and F9, the cross operator is helpful to enhance the exploration ability of the algorithm. Both of them on F1–F5 all obtain all global optima due to

the briefness of these test functions. However, in other test functions such as F12, the cross operator could help to detect other areas to locate more optimum, the PR value of which without cross operator could only reach 0.688. Though the niching strategy is used to cluster the population again, the diversity of that still decreases rapidly. The cross operator helps to enhance the diversity of the population, which is the key of multimodal optimization.

Discussion about the improved two-staged niching method

The method of estimating the radius r is important in multimodal optimization. Compared with previous work, the criteria of determining a peak in INCE is simple but efficient. It is better that the number of groups after clustering is lightly more than the real number of global optima of the optimization problem. In our experiments, the number of groups after clustering by our method is about 1.5–2 times that of practical optima.

The proposed niching method can also be combined with other heuristic algorithms. Apart from the adaptive clustering strategy, the second stage to adjust the number of the population is also adopted. It can help to improve the balance of exploitation and exploration ability of the algorithm. After the first stage, some sharp areas which have a global optimum may do not have enough individuals. Then this method could generate more individuals around it to seek the optimum. Conversely, if the broad areas which have a global optimum have too many individuals, we could delete some poor individuals to reduce the calculation cost. It could be easily combined with other heuristic algorithm to realize the purpose of multimodal optimization. It could still be improved such as merging some clustering according to some metrics. Then, it may be more efficient.

Table 5 The performance of INCE with cross operator and without cross operator

Accuracy level 1E-5

	Cross operator				Non cross operator			
	ADC	ANF	SR	PR	ADC	ANF	SR	PR
F1	0	692	1	1	0	651	1	1
F2	9.12E-14	733	1	1	3.13E-14	1063	1	1
F3	2.81E-07	793	1	1	1.72E-14	1006	1	1
F4	1.42E-14	2874	1	1	1.96E-14	2305	1	1
F5	1.11E-16	1936	1	1	4.44E-17	2126	1	1
F6	5.80E-12	1.29E+05	0.9	0.993	4.69E-12	1.50E+05	0.8	0.978
F7	4.35E-15	2.00E+05	0	0.786	3.17E-15	2.00E+05	0	0.748
F8	7.03E-10	4.00E+05	0	0.84	6.11E-10	4.00E+05	0	0.727
F9	7.32E-12	4.00E+05	0	0.298	1.49E-14	4.00E+05	0	0.366
F10	2.11E-13	17884	1	1	2.56E-13	84567	0.933	0.972
F11	2.26E-13	2.00E+05	1	1	6.11E-13	2.00E+05	0.625	0.906
F12	7.05E-12	7.33E+03	0.688	0.948	5.50E-12	2.00E+05	0	0.688
F13	4.81E-11	2.00E+05	0	0.667	6.59E-12	2.00E+05	0	0.7
F14	8.36E-13	4.00E+05	0	0.667	4.32E-13	4.00E+05	0	0.667
F15	7.92E-13	4.00E+05	0	0.75	1.98E-12	4.00E+05	0	0.625
F16	1.81E-10	4.00E+05	0	0.667	3.11E-12	4.00E+05	0	0.667
F17	7.76E-11	4.00E+05	0	0.625	2.71E-12	4.00E+05	0	0.5
F18	3.46E-12	4.00E+05	0	0.667	8.17E-11	4.00E+05	0	0.375
F19	7.92E-12	4.00E+05	0	0.5	8.36E-12	4.00E+05	0	0.5
F20	1.81E-12	4.00E+05	0	0.375	1.06E-11	4.00E+05	0	0.375
+			8	18			5	11

Table 6 The comparison of the obtained optimal solution by INCE and other algorithms in SLOD

Algorithm	Overlap	Centroid offsets(mm)		Inertia angles offsets(rad)			Moments of inertia	Multimodal
		x-axis	y-axis	x-axis	y-axis	z-axis		
PSO	0	-2.0937	-2.7342	-0.0208	-0.0119	0.0252	2.4159 kg · m ²	No
PSOSQP	0	0.0036	0.5153	0.0000	-0.1480	-0.0239	2.3965 kg · m ²	No
DESQP	0	-0.1748	1.4470	0.0211	-0.0142	0.0136	2.4098 kg · m ²	No
DESQPDE	0	-1.5704	-1.4752	0.0145	-0.0087	0.0019	2.3636 kg · m ²	No
CEM	0	-0.4506	0.0228	-0.2683	-0.2673	-0.1971	2.5844 kg · m ²	No
CEMSQP	0	0.2138	0.5695	-0.0465	-0.0105	-3.50E-03	2.5698 kg · m ²	No
CDE	0	-1.2644	2.4957	-0.0208	0.0221	-0.0115	2.3779 kg · m ²	Yes
	0	-2.5865	1.1439	0.0027	-0.0046	4.472E-04	2.3873 kg · m ²	
	0	-0.0125	0.1026	2.53E-04	-3.89E-05	0.1147	2.3813 kg · m ²	
SDE	0	-1.2292	0.8924	0.0047	0.0123	0.00571	2.4462 kg · m ²	Yes
	0	0.2500	0.3823	0.0087	-0.0109	0.0153	2.4658 kg · m ²	
	0	0.3374	2.0611	-0.0145	-4.943E-04	0.0140	2.4611 kg · m ²	
INCE	0	-2.5183	1.3600	0.0166	-0.0047	0.0102	2.1614 kg · m ²	Yes
	0	-2.7908	0.1596	-0.0071	0.0118	-0.0036	2.1782 kg · m ²	
	0	0.3082	-1.1729	-0.0258	0.0068	0.0017	2.1471 kg · m ²	

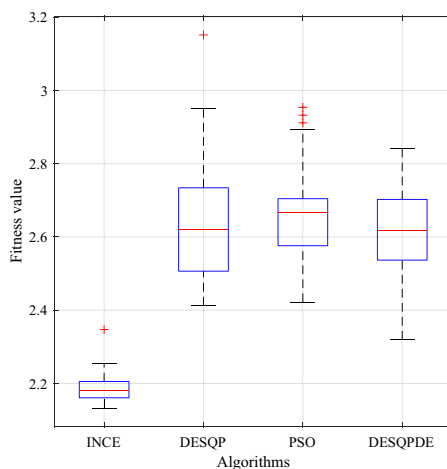
* Multimodal denotes whether the algorithm is multimodal optimization algorithm or not

Table 7 The statistical result of the obtained optimal solution by INCE and other algorithms over 50 runs

Algorithm	Centroid offsets(mm)		Inertia angles offsets(rad)		Moment of inertia ($kg \cdot m^2$)		Success rate (%)
	Mean	Std	Mean	Std	Mean	Std	
PSO	2.8185	1.3437	0.0505	0.0162	2.6559	0.1515	22
PSOSQP	2.2138	1.5695	0.0356	0.0105	2.6234	0.1398	50
DESQP	2.7699	2.3767	0.0684	0.0819	2.6398	0.1522	50
DESQPDE	1.1365	0.1153	0.0034	0.0973	2.6113	0.1193	100
CEM	2.6559	1.5153	1.0304	0.1480	2.6877	0.1232	18
CEMSQP	2.8939	1.3570	0.0514	0.0151	2.6658	0.1395	30
INCE	1.3024	2.1729	1.0258	0.1432	2.1891	0.0395	100

*Success rate: the optimization is regarded as successful if the obtained optimum meets all of the constraints.

*The centroid offsets is the sum of the absolute value of centroid error in x, y axes, i.e., $|x_c - x_e| + |y_c - y_e|$. The inertia angles error is the sum of the absolute value of the inertia angle in x, y, z axes, Mean is the average value and Std is the standard deviation

**Fig. 6** The boxplot of INCE and three other algorithms over 50 runs

14-component satellite layout problem

In this section, the performance of INCE is tested on a 14-components layout optimization problem. Combined with the SLOD model introduced in section “Mathematical model for SLOD”, the case is to optimize the positions of 14 cylinder-components placed in the simplified cylinder satellite module. Enhanced DE is developed to solve it [5]. However, it could not give multiple optimum schemes simultaneously. To demonstrate the validity of INCE proposed in SLOD, we conduct numerical experiments on them to intend to offer more design schemes.

Problem description and parameters setting

In this problem, 14 cylinder components need to be installed on the two surfaces of the bearing plate in the simplified satellite module (see Fig. 1). The detailed data information is illustrated in Table 8. The left 7 components are installed on the up surface of the bearing plate, while the right 7 com-

ponents are installed on the down surface of that. All the components are of the same height as 10 mm and the radius of the satellite module is 50 mm. Combined with Eq. 1, the number of layout component $N = 14$. Thus, the dimension of design variable of the optimization problem is 28. Denote the position of the i_{th} component as $X_i = (x_i, y_i)$, then the constraint $-50 \text{ mm} < x_i < 50 \text{ mm}$, $-50 \text{ mm} < y_i < 50 \text{ mm}$ must be met. Apart from it, the constraints of non-overlap between the different layout components also need to be met.

In addition, the given relevant allowable values of parameters related with constraints are set as $\delta x_e = \delta y_e = 3 \text{ mm}$, $\delta \theta_{x'} = \delta \theta_{y'} = \delta \theta_{z'} = 0.3 \text{ rad}$. The violation in any constraints in the engineering problem is applied via penalty function to the objective of problem to handle inequality and equality constraints, which has been illustrated in Sect. 2.1. The six weight factors of the penalty function are set as: $w_i = 1000 (i = 1, 2 \dots 6)$. The population size, elite sample percent, sigma coefficient and tolerance error are set as 400, 0.1, 1/10 and 0.0001, respectively.

Comparison with other existing global optimization algorithms

To testify the global search ability of INCE in SLOD, we compare it with four other global optimization algorithms. These algorithms mainly include PSO, PSOSQP, DESQP, DESQPDE, CEM and CEMSQP. Among them, DESQP, DESQPDE and PSOSQP have been adopted to solve SLOD in previous work [4,5]. We also use CEM and CEMSQP to make a comparison to illustrate the improvement of CEM by us. Detailed parameters setting for DESQP could be seen in [5]. The population size of pure CEM and CEMSQP is set as 1000. The elite sample percentages are all set as 0.1, which is the same as INCE. All of these algorithms are averaged over 50 runs.

Table 8 The size and mass of 14 layout components. Radius unit(mm), mass unit(kg)

No/Up	Radius	Mass	No/Down	Radius	Mass
1	10	100	8	10	100
2	11	121	9	11	121
3	12	144	10	12	144
4	11.5	132.25	11	11.5	132.25
5	9.5	90.25	12	9.5	90.25
6	8.5	72.25	13	8.5	72.25
7	10.5	110.25	14	10.5	110.25

Statistical report comparing INCE with them in solving this problem which obtained the state-of-the-art optimal design in similar methods is presented in Table 6. The final four rows present the best near optimal layout schemes of four algorithms. We also select three superior near optimal layout schemes obtained by INCE, which is illustrated in the first three rows. The boxplot of the statistical results of these algorithms is also presented in Fig. 6. From Fig. 6, we could clearly see that INCE is better than other algorithms. The moment of inertia of the seached near optimal scheme could reach below $2.2kg \cdot m^2$. The current state-of-art result is above $2.3kg \cdot m^2$, which is reported in [5]. What’s more, almost all of runs of INCE could obtain the similar result that is below $2.3kg \cdot m^2$. We present the iteration history of INCE, PSOSQP and DESQPDE in one single optimization, which is shown in Fig. 7. PSOSQP and DESQPDE are two superior algorithms to solve SLOD, which present the state-of-the-art performance in previous work. In one random experiment in Fig. 7, we could see that INCE possesses faster convergence speed. Futhermore, INCE could find better optimal layout scheme compared with these two algorithms. The final fitness values obtained by INCE, PSOSQP and DESQPDE are $3.012kg \cdot m^2$, $2.451kg \cdot m^2$ and $2.223kg \cdot m^2$ respectively. From the results, we could see that INCE outperforms DESQPDE not only in the number of obtained optima but also the obtained best layout scheme. In addition, the corresponding best layout schemes of INCE and DESQPDE are shown in Table 9, Figs. 10 and 11.

Apart from comparing the best optimal scheme optimized by them, the robustness of these algorithms is a also important. Due to the strong and complex constraints of SLOD, the optimized layout scheme may do not meet the constraint. In our statistical result, we also record the success rate in Table 7. From Table 7, we could see that the solution of INCE is robust. The success rate of it is 100%, which is the same as DESQPDE. The success rate of pure CEM and PSO are both below 50%. DE with local search also could only obtain the success rate of 50%. Futhermore, the final mean fitness value obtained by INCE is below all of other algorithms, which is only $2.1891kg \cdot m^2$. In addition, the std of the results obtained

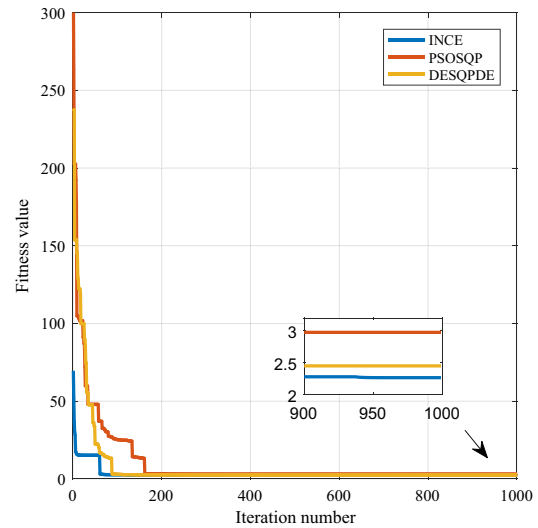


Fig. 7 The iteration history of PSOSQP, DESQPDE and INCE in SLOD

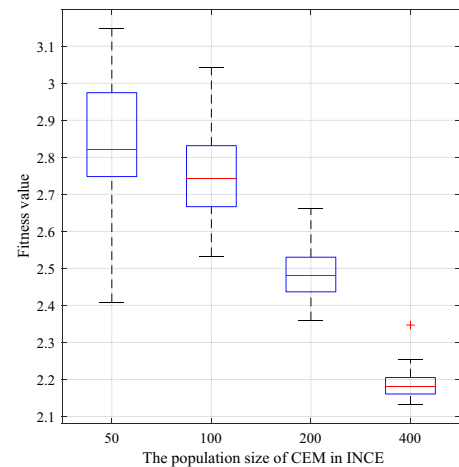


Fig. 8 The performance of INCE with different population sizes of CEM in INCE over 50 runs

by INCE is only 0.0395. However, the std value obtained by other algorithms are all above 0.1, which is also shown in Fig. 6. From Table 7, we also observe that the standard deviations of centroid offset and inertia angles offsets by INCE are not

Table 9 The detailed obtained optimal solution by INCE and DESQPDE

NO	INCE		DESQPDE	
	$x(mm)$	$y(mm)$	$x(mm)$	$y(mm)$
1	-23.112	-1.504	-12.048	-1.496
2	11.42	-15.064	27.248	9.506
3	2.0292	25.3343	2.380	15.111
4	-10.8257	-19.148	9.726	-7.262
5	17.337	10.119	-19.704	16.438
6	-19.558	17.213	12.433	-27.412
7	-1.919	2.554	-30.049	-16.159
8	-22.92	9.679	-14.912	15.383
9	-5.406	25.036	-32.964	-1.849
10	-0.589	1.946	31.164	-1.522
11	2.984	-24.031	9.158	19.867
12	18.096	14.0224	13.478	-23.980
13	18.096	-7.906	7.186	-0.805
14	-18.633	-11.614	-11.993	-6.597

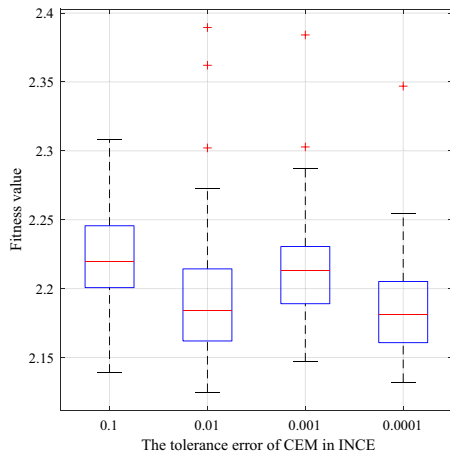


Fig. 9 The performance of INCE with different tolerance errors of CEM in INCE over 50 runs

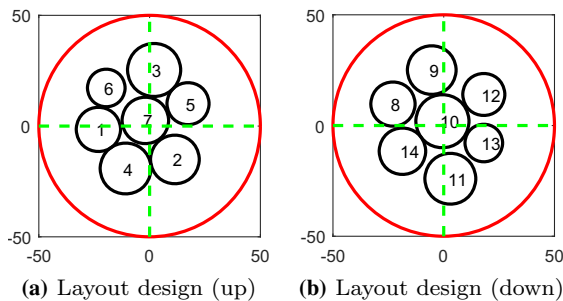


Fig. 10 The obtained layout design scheme by INCE

smaller than other algorithms. It also proves the diversity of obtained optimal schemes by multimodal optimization to some extent (Table 10).

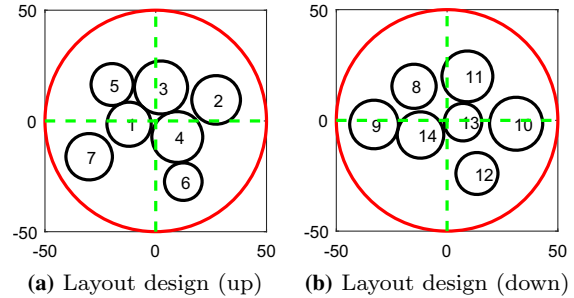


Fig. 11 The obtained layout design scheme by DESQPDE

To study the influence of hyper-parameters on the performance of INCE in SLOD, we also change the population size of CEM and the tolerance error, which is presented in Figs. 8 and 9. From Fig. 8, we could see that with the increasing of the population size of CEM in INCE, the global search ability and robustness are both enhanced. From Fig. 8, we could see that there exists one optimal tolerance error value to make the performance of INCE the best in SLOD. When applied to solve the engineering problem, INCE with the tolerance error 0.01 and 0.0001 both present the superior performance. This conclusion is corresponded with that in multimodal test functions.

Comparison with two other existing multimodal optimization algorithms

To testify the multimodal optimization ability of our proposed INCE, we also compare it with SDE and CDE. In the real-world engineering problem, a level regarded as the optimum is determined, which means the objective value below

Table 10 The results of 14 layout components by INCE with different population sizes in one single optimization. Best, mean and worst unit ($kg \cdot m^2$)

Population size of CEM	50	100	200	400
Best	2.5385	2.4124	2.358	2.1471
Mean	2.9395	2.7761	2.7148	2.3139
Worst	3.4991	2.8919	2.8022	2.4499
Number of obtained optimal solutions	14	21	114	147

* In this single optimization, $2.7kg \cdot m^2$ is regarded as the level of the optimum

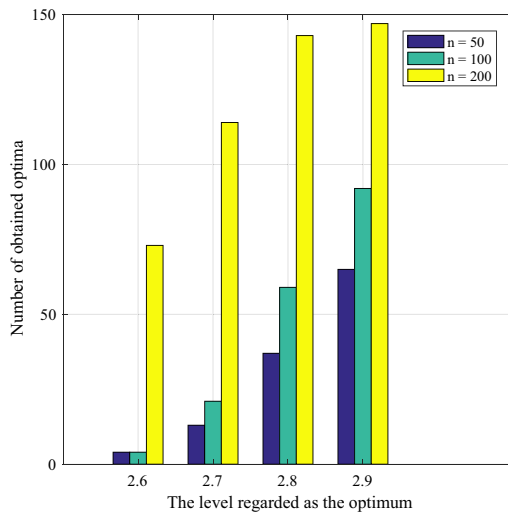


Fig. 12 The number of the obtained layout design schemes by INCE

that would be regarded as the optimum. It is because that the landscape of SLOD is not like the test function, which has multiple absolutely equal optima. It is also why we need to test the performance of our proposed method in multimodal test functions before solving directly SLOD.

First, the relationship between the population size of CEM in INCE and the multimodal performance is investigated, which is presented in Table 10 and Fig. 12. Different from the study in Sect. 4.2.2, we just run the algorithms for once time and evaluate the number of obtained near optimal layout schemes. We change the population size in CEM of INCE as 50, 100, 200 and 400, respectively. to solve this 14-component layout optimization problem. Four level values including $2.6kg \cdot m^2$, $2.7kg \cdot m^2$, $2.8kg \cdot m^2$ and $2.9kg \cdot m^2$ are selected, respectively. From the results, we could observe

that INCE could obtain competitive multimodal performance when the population size of CEM is set as 200 or 400. When $2.7kg \cdot m^2$ is regarded as the level of the optimum, they could obtain 114 and 147 different layout schemes in one single optimization, respectively. Similar to the conclusion in Sect. 4.2.2, with the increasing number of population size, the multimodal optimization ability of INCE also gets enhanced.

We also implement two other multimodal optimization algorithms including CDE and SDE to solve SLOD to make a comparison with INCE. The population sizes of CDE and SDE are both set as 400. The numbers of objective function evaluation for these three algorithms are all set as 100000. The r in SDE is set as 150. The K in CDE is set as 5. The result obtained by them is listed in Table 11. From Table 11, we could see that INCE outperforms CDE and SDE not only in the obtained best layout scheme but also the number of obtained optima. Though SDE and CDE could also obtain multiple near optimal schemes simultaneously, the number of that is relatively less than INCE. It results from that the setting of r or K in them is difficult to determine to some extent. INCE could obtain 147 layout schemes even though the level regarded as the optimum is set as only $2.5kg \cdot m^2$. Thus our proposed method could provide more and better layout schemes for designers in one single optimization.

Discussion about the performance of INCE in solving SLOD

With the increasing of number of layout components, the solving of SLOD becomes more complex due to the increasing of dimensionality. The search space is also larger. Thus, existing global optimization algorithms of solving SLOD are easily trapped in local optimum such as the results presented in Figs. 6 and 7.

Table 11 The number of optimal solutions obtained by three multimodal optimization algorithms

The level regarded as the optimum ($kg \cdot m^2$)	2.2	2.3	2.4	2.5	2.6	2.7	2.8
CDE	0	0	3	14	53	62	73
SDE	0	0	0	23	34	41	57
INCE	3	62	132	147	–	–	–

The proposed INCE in this paper divides the search space by improved clustering strategy. Then, combined with CEM-based offspring generation and local search, the algorithm iteratively search multiple optima in sub-space. Due to the schemes of clustering the population and reproducing new individuals by estimating the distribution parameters, the whole algorithm possess better balance between the exploitation and exploration. The above statistical result also proves that INCE is farther better than existing algorithms to solve the same layout optimization problem including the global optimization ability and multimodal optimization ability.

Conclusion

In this paper, an improved niching-based cross-entropy method is proposed for multimodal satellite layout optimization design. Unlike traditional optimization methods in SLOD that only could obtain one near optimal scheme, the proposed INCE could achieve multiple near optimal schemes simultaneously in single optimization. In this method, aiming at the two key steps in multimodal optimization, three improved strategies are proposed. First, an improved two-staged niching scheme is proposed to conduct adaptive-radius clustering on the population. Besides, to enhance the efficiency of CEM, a new parameters estimation method with elite strategy and local search is adopted, which helps to search the optimum more fastly compared with traditional methods. Finally, a cross operator between different niches is proposed to enhance the diversity of population. To verify the effectiveness of the proposed INCE, we conduct numerical experiments on CEC2013 multimodal functions test set and a practical layout optimization design problem. INCE achieves the best performance in 19 multimodal test functions compared with three multimodal algorithms. When applied to a 14-components layout problem, INCE could obtain the state-of-art optimal design, the moments of inertia of which could reach $2.1471 \text{ kg} \cdot \text{m}^2$. INCE could still obtain nearly 150 different design schemes when the level regarded as the optimum is $2.5 \text{ kg} \cdot \text{m}^2$. Thus, the feasibility and validity of the proposed INCE are testified in test functions and layout design engineering problems.

Acknowledgements This work was supported in part by National Natural Science Foundation of China under Grant No.51675525 and 11725211.

Compliance with ethical standards

Conflict of interest statement The authors declare that they have no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adap-

tation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Yao W, Tang G, Wang N, Chen X (2019) An improved reliability analysis approach based on combined FORM and Beta-spherical importance sampling in critical region. *Struct Multidiscip Optimiz* 60(1):35–58. <https://doi.org/10.1007/s00158-019-02193-y>, <http://link.springer.com/10.1007/s00158-019-02193-y>
2. Yao W, Chen X, Ouyang Q, van Tooren M (2012) A surrogate based multistage-multilevel optimization procedure for multidisciplinary design optimization. *Struct Multidiscip Optimiz* 45(4):559–574. <https://doi.org/10.1007/s00158-011-0714-z>
3. Zheng X, Yao W, Xu Y, Chen X (2019) Improved compression inference algorithm for reliability analysis of complex multistate satellite system based on multilevel Bayesian network. *Reliab Eng Syst Saf* 189:123–142. <https://doi.org/10.1016/j.res.2019.04.011>, <https://linkinghub.elsevier.com/retrieve/pii/S095183201830721X>
4. Chen X, Yao W, Zhao Y, Chen X, Zheng X (2018) A practical satellite layout optimization design approach based on enhanced finite-circle method. *Struct Multidiscip Optimiz* 58(6):2635–2653. <https://doi.org/10.1007/s00158-018-2042-z>
5. Chen X, Yao W, Zhao Y, Chen X, Zhang J, Luo Y (2018) The Hybrid Algorithms Based on Differential Evolution for Satellite Layout Optimization Design. In: 2018 IEEE Congress on Evolutionary Computation (CEC), IEEE, Rio de Janeiro, Brazil, pp 1–8, doi:10.1109/CEC.2018.8477969
6. Cagan J, Degentesh D, Yin S (1998) A simulated annealing-based algorithm using hierarchical models for general three-dimensional component layout. *Comput-Aided Design* 30(10):781–790. [https://doi.org/10.1016/S0010-4485\(98\)00036-0](https://doi.org/10.1016/S0010-4485(98)00036-0)
7. Fakoor M, Mohammad Zadeh P, Momeni Eskandari H (2017) Developing an optimal layout design of a satellite system by considering natural frequency and attitude control constraints. *Aerosp Sci Technol* 71:172–188. <https://doi.org/10.1016/j.ast.2017.09.012>
8. Sun ZG, Teng HF (2003) Optimal layout design of a satellite module. *Eng Optimiz* 35(5):513–529. <https://doi.org/10.1080/03052150310001602335>
9. Zhang B, Teng HF, Shi YJ (2008) Layout optimization of satellite module using soft computing techniques. *Appl Soft Comput* 8(1):507–521. <https://doi.org/10.1016/j.asoc.2007.03.004>, <https://linkinghub.elsevier.com/retrieve/pii/S1568494607000361>
10. Hong-fei Teng Yu, Chen Wei Zeng, Shi Yan-jun, Qing-hua Hu (2010) A Dual-System Variable-Grain Cooperative Coevolutionary Algorithm: Satellite-Module Layout Design. *IEEE Trans Evol Comput* 14(3):438–455. <https://doi.org/10.1109/TEVC.2009.2033585>
11. Shi Y, Li B, Zhang Z (2011) Layout design of satellite module using a modified artificial bee colony algorithm. *Adv Sci Lett* 4(8–10). <https://doi.org/10.1166/asl.2011.1370>
12. JfLiu, Huang J, Li G, WjLiu, Tz Guan, Hao L (2016) A new energy landscape paving heuristic for satellite module layouts. *Front Inf Technol Electron Eng* 17(10):1031–1043. <https://doi.org/10.1631/FITEE.1500302>



13. Wang Z, Ong Y, Sun J, Gupta A, Zhang Q (2019) A generator for multiobjective test problems with difficult-to-approximate pareto front boundaries. *IEEE Trans Evol Comput* 23(4):556–571. <https://doi.org/10.1109/TEVC.2018.2872453>
14. Wang Z, Ong Y, Ishibuchi H (2019) On scalable multi-objective test problems with hardly dominated boundaries. *IEEE Trans Evol Comput* 23(2):217–231. <https://doi.org/10.1109/TEVC.2018.2844286>
15. Wang Z, Zhang Q, Li H, Ishibuchi H, Jiao L (2017) On the use of two reference points in decomposition based multiobjective evolutionary algorithms. *Swarm Evol Comput* 34
16. Boskovic B, Brest J (2017) Clustering and differential evolution for multimodal optimization. In: 2017 IEEE Congress on Evolutionary Computation (CEC), IEEE, Donostia, San Sebastián, Spain, pp 698–705. <https://doi.org/10.1109/CEC.2017.7969378>, <http://ieeexplore.ieee.org/document/7969378/>
17. Sidarto KA, Kania A, Sumarti N (2017) Finding Multiple Solutions of Multimodal Optimization Using Spiral Optimization Algorithm with Clustering. *MENDEL* 23(1):95–102. <https://doi.org/10.13164/mendel.2017.1.095>, <https://mendel-journal.org/index.php/mendel/article/view/58>
18. Basak A, Das S, Tan KC (2013) Multimodal Optimization Using a Biobjective Differential Evolution Algorithm Enhanced With Mean Distance-Based Selection. *IEEE Trans Evol Comput* 17(5):666–685. <https://doi.org/10.1109/TEVC.2012.2231685>
19. Hussain A, Muhammad, YS (2020) Trade-off between exploration and exploitation with genetic algorithm using a novel selection operator. *Complex Intell Syst* p 1–14. <https://doi.org/10.1007/s40747-019-0102-7>
20. Pal R, Yadav S, Karnwal (2020) Eewc: energy-efficient weighted clustering method based on genetic algorithm for hwsns. *Complex Intell Syst* p 391–400. <https://doi.org/10.1007/s40747-020-00137-4>
21. Unal A, Kayakutlu G (2020) Multi-objective particle swarm optimization with random immigrants. *Complex Intell Syst* p 635–650. <https://doi.org/10.1007/s40747-020-00159-y>
22. Liu F, Zhang J, Liu T (2020) A pso-algorithm-based consensus model with the application to large-scale group decision-making. *Complex Intell Syst* p 287–298. <https://doi.org/10.1007/s40747-020-00144-5>
23. Alon G, Kroese DP, Raviv T, Rubinstein RY (2005) Application of the Cross-Entropy Method to the Buffer Allocation Problem in a Simulation-Based Environment. *Ann Oper Res* 134(1):137–151. <https://doi.org/10.1007/s10479-005-5728-8>, <http://link.springer.com/10.1007/s10479-005-5728-8>
24. Altun M, Pekcan O (2017) A modified approach to cross entropy method: Elitist stepped distribution algorithm. *Appl Soft Comput* 58:756–769. <https://doi.org/10.1016/j.asoc.2017.04.032>, <https://linkinghub.elsevier.com/retrieve/pii/S1568494617302144>
25. Lopez-Garcia P, Onieva E, Osaba E, Masegosa A, Perallos A (2016) GACE: A meta-heuristic based in the hybridization of Genetic Algorithms and Cross Entropy methods for continuous optimization. *Expert Syst Appl* 55:508–519. <https://doi.org/10.1016/j.eswa.2016.02.034>, <https://linkinghub.elsevier.com/retrieve/pii/S0957417416300653>
26. Kroese DP, Porotsky S, Rubinstein RY (2006) The Cross-Entropy Method for Continuous Multi-Extremal Optimization. *Methodol Comput Appl Probab* 8(3):383–407. <https://doi.org/10.1007/s11009-006-9753-0>, <http://link.springer.com/10.1007/s11009-006-9753-0>
27. Yang Q, Chen WN, Li Y, Chen CLP, Xu XM, Zhang J (2017) Multimodal Estimation of Distribution Algorithms. *IEEE Trans Cybern* 47(3):636–650. <https://doi.org/10.1109/TCYB.2016.2523000>, <http://ieeexplore.ieee.org/document/7407318/>
28. Stoean C, Preuss M, Stoean R, Dumitrescu D (2010) Multimodal Optimization by Means of a Topological Species Conservation Algorithm. *IEEE Trans Evol Comput* 14(6):842–864. <https://doi.org/10.1109/TEVC.2010.2041668>, <http://ieeexplore.ieee.org/document/5491155/>
29. Li L, Tang K (2015) History-Based Topological Speciation for Multimodal Optimization. *IEEE Trans Evol Comput* 19(1):136–150. <https://doi.org/10.1109/TEVC.2014.2306677>, <http://ieeexplore.ieee.org/document/6742601/>
30. Zhang G, Li D, Zhou X, Xu D (2015) Differential evolution with dynamic niche radius strategy for multimodal optimization. In: The 27th Chinese Control and Decision Conference (2015 CCDC), IEEE, Qingdao, China, pp 3059–3064. <https://doi.org/10.1109/CCDC.2015.7162446>, <http://ieeexplore.ieee.org/document/7162446/>
31. Li X (2005) Efficient differential evolution using speciation for multimodal function optimization. In: Proceedings of the 2005 conference on Genetic and evolutionary computation - GECCO '05, ACM Press, Washington DC, USA, p 873. <https://doi.org/10.1145/1068009.1068156>, <http://portal.acm.org/citation.cfm?doi=1068009.1068156>
32. Thomsen R (2004) Multimodal optimization using crowding-based differential evolution. In: Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No.04TH8753), IEEE, Portland, OR, USA, pp 1382–1389. <https://doi.org/10.1109/CEC.2004.1331058>, <http://ieeexplore.ieee.org/document/1331058/>
33. Li X, Engelbrecht A, Epitropakis, G M (2014) Benchmark Functions for CEC'2013 Special Session and Competition on Niching Methods for Multimodal Function Optimization pp 10–23
34. Li X (2016) Multimodal optimization using niching methods. In: Wiley Encyclopedia of Electrical and Electronics Engineering, John Wiley & Sons, Inc., Hoboken, NJ, USA, pp 1–8. <https://doi.org/10.1002/047134608X.W8338>, <http://doi.wiley.com/10.1002/047134608X.W8338>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.