Soft Comput manuscript No.
(will be inserted by the editor)

Hybrid Evolutionary Programming using Adaptive Lévy Mutation and Modified Nelder-Mead Method

Jinwei Pang · Jun He · Hongbin Dong

Received: date / Accepted: date

Abstract Evolutionary programming has been widely applied to solve global optimization problems. Its performance is related to both mutation operators and fitness landscapes. In order to make evolutionary programming more efficient, its mutation operator should adapt to fitness landscapes. The paper presents novel hybrid evolutionary programming with adaptive Lévy mutation, in which the shape parameter of Lévy probability distribution adapts to the roughness of local fitness landscapes. Furthermore, a modified Nelder-Mead method is added to evolutionary programming for enhancing its exploitation ability. The proposed algorithm is tested on 39 selected benchmark functions and also benchmark functions in CEC2005 and CEC2017. The experimental results demonstrate that the overall performance of the proposed algorithm is better than other algorithms in terms of the solution accuracy.

Keywords Global Optimization · Evolutionary Programming · Fitness Landscape · Lévy distribution · Nelder-Mead method · Algorithm Design

The work is supported by the National Science Foundation of China (No. 61472095) and the EPSRC under Grant No. EP/I009809/1.

Jinwei Pang
Department of Computer Science and Technology, Harbin Engineering University, Harbin, China

Jun He
School of Science and Technology, Nottingham Trent University, Clifton Campus, Nottingham NG11 8NS, UK

Hongbin Dong
Department of Computer Science and Technology, Harbin Engineering University, Harbin, China
E-mail: donghongbin@hrbeu.edu.cn

1 Introduction

Evolutionary programming (EP), alongside genetic algorithms and evolution strategy, stems from natural biological evolution. EP was first proposed by Fogel in 1960s who simulated evolution using finite-state machines and studied the generation of artificial intelligence [17]. Today it has been applied widely to solve diverse problems. One of its most important applications is global optimization [39, 22, 9, 24].

EP simulates evolution at species' level with mutation and selection but without no crossover. Mutation is the main search operator in EP. Several mutation operators have been proposed such as Gaussian, Cauchy and Lévy mutation. The performance of EP depends on both mutation operators and fitness landscapes. For example, Gaussian mutation is good on unimodal fitness landscapes; Cauchy mutation is better on multimodal fitness landscapes but less effective on unimodal fitness landscapes. Therefore, in order to take advantages of different mutation operators, it is necessary to design adaptive EP which integrates several mutation operators into one algorithm. Different approaches have been suggested for implementation the integration, such as linear combination [9], mixed strategy [13, 27], reinforcement learning [40] and ensemble [26].

Lévy distribution is a generalization of Gaussian and Cauchy distributions because the latter is special cases of the former. When the shape parameter of Lévy distribution $\alpha = 1$, Lévy distribution is reduced to Cauchy distribution. When $\alpha = 2$, it becomes Gaussian distribution. Therefore a reasonable hypothesis is that the shape parameter α of Lévy distribution should adapt to fitness landscapes. If a fitness landscape looks like a unimodal landscape, then the shape parameter α of Lévy distribution should be set to a large value

for a short-tailed distribution; if a fitness landscape is a multimodal landscape with many local optima, then the shape parameter α should be assigned to a small value for a long-tailed distribution. The research questions are how to characterize fitness landscapes and how to adapt Lévy mutation to fitness landscapes.

Nevertheless, the current design of adaptive EP seldom utilizes any explicit characteristic of fitness landscapes. An early work is the mixed strategy EP described in [37] which links the choice of mutation operators to local fitness landscapes. A mixed strategy in [37] is a probability distribution of choosing Cauchy or Gaussian mutation operator. The work in [37] is based on supervised learning. First, a training data set is constructed which consists of several typical fitness landscapes, and then the best mixed strategy for a specific landscape is learned on the training data set. Next, the same or similar strategy is applied on the same or similar fitness landscape on the testing data set (consisting of new fitness landscapes).

This paper aims to design novel EP with a Lévy mutation operator adapting to fitness landscapes. It is established on a new idea, briefly described as follows: a population is viewed as an observation of a local fitness landscape in which the population resides; the roughness of a local fitness landscape is measured by the number of optima (both local and global) in the population; and the shape parameter α in Lévy mutation depends on the roughness of a local fitness landscape.

Because evolutionary algorithms are good at exploration but not at exploitation, an essential design principle is widely used in memetic algorithms [31], that is, local search can be added into evolutionary algorithms for improving their exploitation ability. Thus, a modified Nelder-Mead is also added to the proposed EP for enhancing its exploitation ability.

The rest of the paper is organized as follows. Section 2 reviews EP. Section 3 presents our hybrid EP hybrid EP using the Lévy mutation operator adapting to local fitness landscapes and modified Nelder-Mead exploitation operator. Section 4 reports the result of experiments. Section 5 concludes the paper.

2 Literature Review of Evolutionary Programming

EP was originally proposed to optimize finite state machines [17]. Currently it is widely applied into solving optimization problems. This paper focuses on global minimization problems with bounded constraints. It can be formulated as a pair (S, f), where $S \subseteq \mathbb{R}^n$ is a bounded set, $f(x) : S \to \mathbb{R}$ is a real-valued function and n is the dimension. The task is to find a point $x_{\min} \in S$ such that $f(x_{\min})$ is a global minimum,

$$f(x_{\min}) \leq f(x), \quad \forall x \in S.$$

EP simulates the evolution at species level. Therefore, no crossover operator is employed. Depending on the mutation operator used to produce variation in the population, different versions of EP were subsequently proposed. In EP, the representation of an individual is a pair of real vectors (x, σ),

$$x = (x(1), x(2), \cdots, x(n)),$$
$$\sigma = (\sigma(1), \sigma(2), \cdots, \sigma(n)),$$

where x is the location of an individual in $\mathbb{R}^n$ and σ is the step size. A population consists of μ individuals.

Conventional EP which uses a single mutation operator can be described as follows [39,22,9,24]:

1. **Initialization:** Set the generation counter $t = 1$. Generate an initial population of μ individuals. Each individual is taken as a pair of real-valued vectors $(x_i, \sigma_i), i \in \{1, \ldots, \mu\}$. Evaluate the fitness score of each individual in the population.
2. **Mutation:** For each parent (x_i, σ_i), creates a single offspring (x'_i, σ'_i) as follows:

   for $j = 1, \ldots, \mu$ **do**

   $\sigma'_i(j) = \sigma_i(j) \exp(\tau_a N(0,1) + \tau_b N_j(0,1))$;

   $x'_i(j) = x_i(j) + \sigma'_i(j) N_j(0,1)$ ▷ classical EP using Gaussian mutation (CEP);

   [or $x'_i(j) = x_i(j) + \sigma'_i(j)\delta_j$ ▷ fast EP using Cauchy mutation (FEP);]

   [or $x'_i(j) = x_i(j) + \sigma'_i(j) L_j(\alpha)$ ▷ EP using Lévy mutation (LEP);]

   end for

   where $x_i(j), x'_i(j)$, $\sigma_i(j)$, $\sigma'_i(j)$ are the jth component of vectors $x_i, x'_i, \sigma_i, \sigma'_i$ respectively. $N(0,1)$ is a normally distributed one-dimensional random number with zero mean and standard deviation one. $N_j(0,1)$, δ_j and $L_j(\alpha)$ stand for Gaussian, Cauchy and Lévy random variables generated for each value of j respectively. τ_a, τ_b and α are set to $(\sqrt{2\mu})^{-1}$, $(\sqrt{2\sqrt{\mu}})^{-1}$ and 0.8 respectively.
3. **Fitness evaluation:** Calculate the fitness of each offspring (x'_i, σ'_i).
4. **q-Tournament Selection:** The union of parents (x_i, σ_i) and offspring $(x'_i, \sigma'_i), i \in \{1, \ldots, \mu\}$ is formed. For every individual in the union, q opponents are chosen uniformly at random from the combined population. For each comparison, if the individual's objective value is smaller (i.e. minimization problem) than the opponent's, it receives a "win". From the union of parents and offspring, individuals with the most wins are selected to be the parents in the next generation.

5. **Iteration:** Repeat steps 2-4 until a stopping criterion is satisfied.

As shown in the above algorithm, several mutation operators have been proposed in EP, such as Gaussian, Cauchy [39] and Lévy [22] mutation operators. Gaussian mutation is the classical mutation operator whose performance is good on unimodal fitness landscapes. Cauchy mutation can outperform Gaussian mutation on multimodal fitness landscapes but is less effective on unimodal fitness landscapes. Lévy mutation is claimed to be more flexible than both Gaussian and Cauchy mutation. Lévy mutation is a generalization of Gaussian and Cauchy mutation because Gaussian and Cauchy probability distributions are special cases of the Lévy probability distribution. However, for any fixed shape parameter α, EP using Lévy mutation still cannot solve all test functions efficiently. As each mutation operator has its advantages and disadvantages, the overall performance of EP can be improved by applying different mutation operators simultaneously or by adaptively choosing mutation operators.

In order to integrating several mutation operators into one algorithm, numerous methods have been proposed in EP. A simple implementation proposed by [9] is a linear combination of Gaussian and Cauchy distributions. This combination can be viewed as a new mutation operator, whose probability distribution is a convolution of Gaussian and Cauchy probability distributions. [24] considered the Lévy distribution with various shape parameters. Mixed mutation strategies which integrate several mutation operators were proposed in [20, 13]. In a mixed strategy EP, several mutation operators, such as Gaussion, Cauchy, Lévy, and single-point mutation operators, were employed. Each individual chooses one of the four mutation operators according to a certain probability distribution in each generation to produce an offspring. An operator which produces a higher fitness offspring will receive a better reward and then be chosen with a higher probability. The probability distribution is dynamically adjusted based on the performance of the mutation strategies. [40] adopted reinforcement learning theory to learn optimal policies by maximizing the accumulated rewards. The selection of mutation operators is mapped into a reinforcement learning problem. [26] investigated an ensemble approach where each mutation operator has its associated population and different parameter values. [21] gave another learning method. They used genetic programming to learn good probability distributions of mutation operators. It aims at an automatic design of EP using genetic programming. [6] took fitness tracking and treat locally trapped individuals separately, so the incorporation of mutation operators is dependent of individuals. [32] mixed Gaussian, Cauchy and Lévy mutation operators by using Shapley value to assign weights to these three operators.

Different from existing EP design approaches, the novelty in our work is to link the shape parameter α to the roughness of a fitness landscape. Different individuals in a population may take different Lévy probabilities distributions (including both Gaussian and Cauchy distributions).

3 Hybrid Evolutionary Programming

This section presents our new hybrid EP which is established upon three components: a roughness measurement of local fitness landscapes, a Lévy mutation operator adapting to local fitness landscapes and a modified Nelder-Mead exploitation operator.

3.1 Roughness of Local Fitness Landscapes

The fitness landscape is one of the most commonly used metaphors to describe the behaviour of EAs in optimisation. However giving an exact definition of the concept sometimes is not easy while several different explanations exist [35]. There exist three approaches to illustrate the features of a fitness landscape: mathematical characterization, statistic measures and practical studies [35].

An exact description of a continuous fitness landscape in $\mathbb{R}^n$ is usually intractable since it needs an infinite sample points. Instead, what can be obtained is a population consisting of finite sample points. Each population is regarded as an observation of a local fitness landscape in which the population resides. If the population contains sufficient different points with good population diversity, then it is good approximation of the local fitness landscape.

An important characteristic of fitness landscapes is their roughness. If a fitness landscape is unimodal, then its roughness is smaller; if it is multimodal, then its roughness is larger. Roughness can be measured by the number of optima in a fitness landscapes. In practice, a population is used to represent an observation of a fitness landscape.

The concept of an optimum is based on the neighborhood. A point x is likely to be an optimum (either local or global) if it is far away for any better point y with $f(y) < f(x)$ or there is no other better point y such that $f(y) < f(x)$. The optimum likeness of x is calculated as follows: given an $x \in P$, let

$$d_{\min}(x) = \min\{d(x, y) \mid f(y) \leq f(x)\}$$

be the minimum distance to any points better than x where $d(x, y)$ is Euclidean distance. If the distance $d_{\min}(x)$ is greater than a threshold, then the optimum likeness of x is 1; otherwise the likeness is between $[0, 1)$ which depends on the distance $d_{\min}(x)$. This idea is visualized in Figure 1. In the figure, x_1 is a global optimum and x_2 is a local optimum; x_3 and x_5 are not optimal points; x_4 is a local optimum because it is far away from other better points.

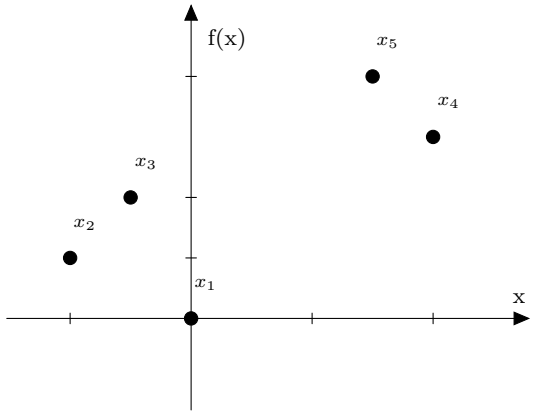

Fig. 1 A local fitness landscape represented by a population

Algorithm 1 describes the procedure of calculating the roughness of a population. The algorithm is explained step by step as follows.

In Line 1, individuals are sorted in the order of the function value from low to high. Lines 2-6 are used to determine a threshold whether an individual is "far away" from another one. In Lines 2-5, the maximum distance $d_{\max}$ between x_1 and other individuals in the population P is calculated. In Line 6, the distance threshold is set to $0.2d_{\max}$. The threshold means that if the distance $d(x, y)$ between two points x, y is larger than $0.2d_{\max}$, then x is called called "far away" from y.

Lines 7-14 are the procedure of calculating the optimum likeness. Line 7 states that x_1 is always an optimum within P. Lines 8-14 are used to calculate the optimum likeness of individuals $x_2, \cdots, x_\mu$. Lines 8-12 are to find the minimum distance $d_{\min}(x_i)$ between x_i and other individuals better than x_i. Line 13 is to calculate the optimum likeness of x_i using the following formula.

$$l(x_i) = \begin{cases} 1, & \text{if } d_{\min}(x_i) \geq \text{threshold}, \\ \left(\frac{d_{\min}(x_i)}{\text{threshold}}\right)^2, & \text{otherwise}. \end{cases} \tag{3}$$

According to the above formula, if the minimum distance $d_{\min}(x_i) \geq$ threshold, then the likeness of x_i being an optimum is 1; otherwise the likeness is between $[0, 1)$.

Algorithm 1 Roughness calculation

Input: A population P consisting of μ individuals.

1: Sort individuals in P in the order: $f(x_1) < f(x_2) < \cdots < f(x_\mu)$.
2: **for** $i = 2, \cdots, \mu$ **do**
3: Calculate $d(x_1, x_i)$. ▷ $d(x, y)$ Euclidean distance
4: **end for**
5: Calculate the maximum distance between the best individual x_1 and other individuals in P:

$$d_{\max} = \max\{d(x_1, x_\mu), i = 1, \cdots, \mu\}.$$

6: Set the distance threshold: threshold $= 0.2d_{\max}$.
7: Set $l(x_1) = 1$, ▷ the optimum likeness of the best individual x_1 is 1.
8: **for** $i = 2, \cdots, \mu$ **do**
9: **for** $j = 1, \cdots, i-1$ **do**
10: Calculate the distance $d(x_i, x_j)$ between x_i and a point x_j with $f(x_j) \leq f(x_i)$.
11: **end for**
12: Find the minimum distance

$$d_{\min}(x_i) = \min\{d(x_i, x_j), j = 1, \cdots, i-1\}.$$

13: Calculate the optimum likeness of x_i:

$$l(x_i) = \begin{cases} 1, & \text{if } d_{\min}(x_i) \geq \text{threshold}, \\ \left(\frac{d_{\min}(x_i)}{\text{threshold}}\right)^2, & \text{otherwise}. \end{cases} \tag{1}$$

14: **end for**
15: Calculate the roughness of P by the sum

$$r(P) = l(x_1) + \cdots + l(x_\mu). \tag{2}$$

Output: the degree of roughness $r(P)$.

The larger the distance $d_{\min}(x_i)$ is, the more likely x_i is an optimum.

Finally in Line 15, the roughness of the whole population is obtained as the sum of the optimum likeness values of all individuals.

$$r(P) = l(x_1) + \cdots + l(x_\mu). \tag{4}$$

The larger $r(P)$ is, the rougher the population P is.

3.2 Lévy Mutation Operator Adapting to Fitness Landscape

The Lévy probability distribution was discovered by Lévy in the 1930s. The probability density of a symmetrical Lévy process has the following form [24]:

$$L_{\alpha,\gamma}(z) = \frac{1}{\pi} \int_0^\infty e^{-\gamma q^\alpha} \cos(qz) dq, \tag{5}$$

where α and γ are two parameters characterizing the distribution. The shape parameter α satisfies $0 < \alpha \leq 2$ which controls the shape of the probability distribution. The smaller the parameter α is, the longer the tail is.

In particular, for $\alpha = 1$, the distribution is equivalent to the Cauchy probability distribution. When $\alpha \to 2$, the distribution approaches the Gaussian distribution. γ is the scaling factor satisfying $\gamma > 0$. Without loss of generality γ is set to 1.

In order to maximize the efficiency of Lévy mutation operator, a natural idea is to let its probability distribution adapt to local fitness landscapes. On a rough fitness landscape, the shape parameter α is set to a smaller value so that the probability distribution has a longer tail. On a smooth fitness landscape, a larger value of α is chosen so that the probability distribution has a shorter tail. According to the roughness of a local fitness landscape, different values of α are set for different individuals.

Although it is possible to calculate Lévy probability distribution for any shape parameter $\alpha \in (0, 2.0]$, this will make the algorithm implementation very complex. Instead, it is sufficient to consider several typical values of α. For the sake of calculation, 14 values of α are chosen for calculating Lévy probability distribution, which are

$$0.8, 1, 1.1, 1.2, \cdots, 1.8, 1.9, 1.95, 1.99, 2.0. \tag{6}$$

Algorithm 2 illustrates the procedure of adapting the shape parameter α to local fitness landscapes. It is explained step by step as follows.

In Line 1, individuals in the population P are sorted in the order of the fitness value from low to high. In Line 2, the degree of roughness of P is obtained by using Algorithm 1. In Lines 3, an array A keeps the preset values of α given by (6).

Lines 4-8 are used to assign the α value to each individual. In Line 4, the α value for the best individual x_1 is set to 2, equivalent to the Gaussian probability distribution. The α value for the worst individual x_μ is set to 0.8 with the longest tail, longer than Cauchy probability distribution. Lines 5-8 assign α values to individuals $x_2, \cdots, x_{\mu-1}$ according to the degree of roughness. In Line 6, a shape factor $\alpha'(i)$ is calculated for the ith individual based on the roughness $r(P)$.

$$\alpha'(i) = 2 - \frac{r(P) + i}{2 \times \mu - 1}. \tag{10}$$

It is easy to verify that $\alpha'(i) \in [1, 2)$ because $r(P) \in (1, \mu]$ and $i \in [1, \mu - 1]$. However, $\alpha'(i)$ may not be a preset value in the array A. Thus in Line 7, α' is rounded to the nearest preset α value from the upper side.

Algorithm 3 illustrates the Lévy mutation operator adapting to local fitness landscapes. The algorithm requires shape parameters which are generated by Algorithm 2.

Algorithm 2 Shape parameters adaptation

Input: A population P with μ individuals.

1: Sort individuals in P such that $f(x_1) < f(x_2) < \cdots < f(x_\mu)$.

2: Calculate the degree of roughness $r(P)$ (Algorithm 1).

3: An array $A[0], \cdots, A[13]$ keeps 14 preset values of α, which are

$$\begin{aligned} &A[0] = 0.8, \\ &A[k] = 1.0 + 0.1(k-1), \qquad k = 1, \cdots, 10, \\ &A[11] = 1.95, \\ &A[12] = 1.99, \\ &A[13] = 2. \end{aligned}$$

4: Set the α value for the 1st and μth individuals:

$$\alpha(1) = A[13] = 2, \quad \alpha(\mu) = A[0] = 0.8. \tag{7}$$

▷ Set the α value for the best individual with the shortest tail and the worst individual with the longest tail.

5: **for** $i = 2, \cdots, \mu - 1$ **do**

6: Set the α value for x_i

$$\alpha'(i) = 2 - \frac{r(P) + i}{2 \times \mu - 1}. \tag{8}$$

7: Round α' to the nearest preset α value from upper:

$$\alpha(i) = \begin{cases} A[1], \text{ if } \alpha'(i) \le A[1], \\ A[j], \text{ if } A[j-1] < \alpha'(i) \le A[j], \\ \qquad j \in [2, \cdots, 13]. \end{cases} \tag{9}$$

8: **end for**

Output: Shape parameters $\alpha(1), \cdots, \alpha(\mu)$.

Algorithm 3 Lévy Mutation Operator Adapting to Fitness Landscape

Input: population P consisting of μ individuals and shape parameters $(\alpha(1), \cdots, \alpha(\mu))$.

1: **for** $i = 1, \cdots, \mu$ **do**
2: **for** $j = 1, \ldots, n$ **do**
3: $\sigma_i'(j) = \sigma_i(j) \exp(\tau_a N(0,1) + \tau_b N_j(0,1))$
4: $x_i'(j) = x_i(j) + \sigma_i'(j) L_j(\alpha(i))$
5: **end for**
6: **end for**

Output: population $R \leftarrow \{x_1', \cdots, x_\mu'\}$.

In Line 3, the step size $\sigma_i(j)$ is adjusted by the following formula: for $j = 1, \ldots, n$

$$\sigma_i'(j) = (\sigma_b + \sigma_i(j)) \exp\{\tau_a N(0,1) + \tau_b N_j(0,1)\}. \tag{11}$$

To avoid the step size σ falling too low to zero, a lower bound σ_b usually is put on σ [16]. $\sigma_b > 0$ is the minimum value of the step size σ.

In Line 4, a new x_i' is generated using the following formula: for $j = 1, \ldots, n$

$$x_i'(j) = x_i(j) + \sigma_i'(j) L_j(\alpha(i)), \tag{12}$$

where $L_j(\alpha(i))$ is a Lévy random variable which is generated anew for each component j. Its shape parameter $\alpha(i)$ is provided by Algorithm 2.

3.3 Modified Nelder-Mead Exploitation Operator

The Nelder-Mead method [30] is a local search algorithm used to find the minimum of an nonlinear objective function in a multidimensional space without derivatives. It makes down-hill search using a simplex which adapts to fitness landscapes. The Nelder-Mead method is good at exploitation whereas EP is good at exploration. Therfore the combination of these two methods naturally has attracted researchers' interests [14, 10, 25, 15, 28].

Based on the Nelder-Mead method [30], we design an exploitation operator, called a modified Nelder-Nead exploitation operator. The main change is the number of testing points which is set to 3 in the new operaror, rather than $(n+1)$ testing points used in the original method [30].

Algorithm 4 provides the pseudo-code of the modified Nelder-Mead exploitation operator. It is explained step by step blow.

Line 2 is used to select ν best individuals (denoted by the population P') from the population P for exploitation, where ν is a multiple of three. In experiments, the ν value is set to a small value 6.

Lines 3-29 apply the modified Nelder-Mead method to points (individuals) in the population P'. For each iteration, three points are selected and processed.

In Line 4, three points are selected from P' which are the best individual x_1 in P', the nearest x_2 and sub-nearest individual x_3 of the individual x_1 in P'. Then these points are removed from P' in Line 5. In Line 6, these points are sorted in the order of their function value from low to high such that $f(x_1) \leq f(x_2) \leq f(x_3)$.

Lines 7-28 are the main procedure of the modified Nelder-Mead method, including centroid, reflection, expansion and contraction. The modified Nelder-Mead method is visualized in Figure 2. Because the number of testing points is only 3, the complexity of expansion, reflection and contraction operations in the modified Nelder-Mead method is reduced to $O(1)$.

In Line 7, two types of centroid points are generated which are $x_o = \frac{1}{3}\sum_{k=1}^{2} x_k$ with probability 0.03, or $x_o = \frac{1}{3}\sum_{k=1}^{3} x_k$ with probability 0.97. This is a little different from the original Nelder-Mead method.

Lines 8-28 are normal reflection, expansion and contraction operations, which are the same as those in the Nelder-Mead method but only acting on three points, rather than $(n+1)$ points (a simplex in the n-dimension space).

Algorithm 4 A modified Nelder-Mead exploitation operator

Input: A population P and a sub-population size ν where ν is a multiple of three.

1: Population $R \leftarrow \emptyset$.
2: Population $P' \leftarrow$ find the best ν individuals in P.
3: **while** P' is not empty **do**
4: Population $Q \leftarrow$ find the best individual x_1 in P', the nearest x_2 and sub-nearest individual x_3 of the individual x_1 in P'.
5: $P' \leftarrow P' \setminus Q$,
6: Sort the three points in the order: $f(x_1) \leq f(x_2) \leq f(x_3)$.
7: [Centroid] Calculate the centroid x_o as follows: $x_o = \frac{1}{3}\sum_{k=1}^{2} x_k$ with probability 0.03; or $x_o = \frac{1}{3}\sum_{k=1}^{3} x_k$ with probability 0.97.
8: [Reflection] Compute the reflected point $x_r = x_o + \alpha(x_o - x_3)$. ▷ α is a reflection coefficient. Its standard value is $\alpha = 1$.
9: **if** $f(x_1) \leq f(x_r) < f(x_3)$, **then**
10: $x'_3 \leftarrow$ the reflected point x_r.
11: **else if** $f(x_r) < f(x_1)$ **then**
12: [Expansion] Compute the expanded point $x_e = x_o + \gamma(x_r - x_o)$. ▷ γ is an expansion coefficient. Its standard value is $\gamma = 2$.
13: **if** $f(x_e) < f(x_r)$ **then**
14: $x'_3 \leftarrow$ the expanded point x_e
15: **else**
16: $x'_3 \leftarrow$ the reflected point x_r.
17: **end if**
18: **else**
19: [Contraction] Compute the contracted point $x_c = x_o + \rho(x_3 - x_o)$. ▷ ρ is a contraction coefficient. Its standard values is $\rho = 1/2$.
20: **if** $f(x_c) < f(x_3)$ **then**
21: $x'_3 \leftarrow$ with the contracted point x_c.
22: **else**
23: **for** $i = 2, 3$ **do**
24: [Shrink] $x'_i = x_1 + \sigma(x_i - x_1)$. ▷ σ is a shrink coefficient. Its standard value is $\sigma = 1/2$.
25: **end for**
26: **end if**
27: **end if**
28: $R \leftarrow R \cup \{x'_1, x'_2, x'_3\}$. ▷ Let $x'_i = x'_i$ if x'_i is not assigned to a value.
29: **end while**

Output: The population R.

The space complexity of the Nelder-Mead variant (within one generation) is a constant $O(1)$ because the number of testing points is a constant multiple of 3. Furthermore, because the Nelder-Mead variant changes a constant multiple of 3 points in one generation, the runtime complexity of it is $O(G_{max} * D)$, where G_{max} is maximum number of generations, and D is the dimension of the problem.

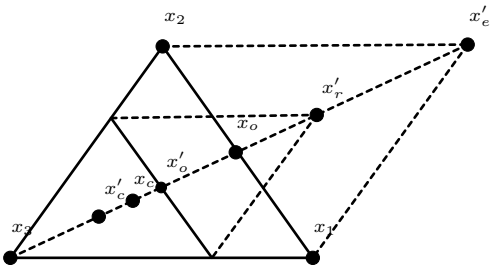

Fig. 2 x_1, x_2, x_3 are three test points. x'_r, x'_e, x'_c are reflected, expanded and contracted points points respectively, which correspond to the centroid x'_o.

3.4 Hybrid Evolutionary Programming

Now we design new hybrid EP using the Lévy mutation operator adapting to local fitness landscapes and modified Nelder-Mead exploitation operator (HEP in short). The pseudo-code of HEP is given in Algorithm 5. The flow chart of HEP is given in Figure 3. The procedure of HEP is explained step by step as follow.

Algorithm 5 Hybrid EP

Input: fitness function f, tournament size q, subpopulation size ν.
1: Population $P \leftarrow$ generateμ individuals at random.
2: Evaluate the fitness value of individuals in P.
3: **for** $t = 0, \cdots, t_{\max}$ **do** ▷ $t_{\max}$ is the maximum number of iterations
4: Sort individuals in P in the order: $f(x_1) < f(x_2) < \cdots < f(x_\mu)$.
5: **if** $t \equiv 0 (\mathrm{mod} \kappa)$ **then** ▷ κ is an interval of adjusting shape parameters.
6: Calculate shape parameters $\alpha(1)$, $\cdots$, $\alpha(\mu)$ for population P (using Algorithm 2).
7: **end if**
8: Population $Q \leftarrow$ mutate P by Lévy mutation operator (using Algorithm 3).
9: Evaluate the fitness value of individuals in Q.
10: Population $R \leftarrow$ select μ individuals from $P \cup Q$ by q-tournament selection.
11: $P \leftarrow$ exploit R by the modified Nelder-Mead exploitation operator (using Algorithm 4).
12: **end for**
Output: the best solution $(x_{\min}, f_{\min})$ found during the iteration.

Lines 1 and 2 are initialization. An initial population P is generated at random consisting of μ individuals. Then their fitness value is evaluated.

Lines 4-7 are roughness calculation. Given a population P as an observation of a local fitness landscape, individuals are sorted according to the fitness value from low to high. Then the roughness of P is calculated using Algorithm 1 per κ generations. Since it is not necessary to calculate the roughness at each generation, the parameter κ introduced to control a generation interval. Experiment results show that a good value of κ is 30. The detail of calculating roughness has been given in Algorithm 1.

Line 8 is the new Lévy mutation operator adapting to local fitness landscape. Its detail has been given by Algorithm 3.

Lines 9-10 are q-tournament selection. First the fitness values of offspring are evaluated. The fitness values of the parent have been evaluated from previously. Then q-tournament selection is used to select μ children from parent and offspring populations as the next generation parent. The procedure is the same as that in the traditional EP (see Section 2).

Line 11 is the modified Nelder-Mead exploitation operator which is used to conduct further exploitation for improving solution accuracy. Its detail has been given in Algorithm 4.

Different from other EP algorithms using the Lévy mutation [24], HEP has two new features that make it adapt to local fitness landscapes.

1. **Roughness calculation:** It analyzes the roughness of local fitness landscapes. The analysis is implemented per κ generations. Experiment results show that a good value of κ is 30 so it doesn't increase too much computation.
2. **Lévy mutation with adaptive shape parameters:** The shape parameter α in Lévy probability distribution is linked to the roughness of a local fitness landscape.

4 Experiments and Results

4.1 Benchmark Functions

In experiments, thirty nine single-objective benchmark functions are used for validating the effectiveness of proposed algorithm. These functions cover different types of multi-modal fitness landscapes. Table 1 shows the name and references of these functions. Their parameters, dimensions and variable bounds are the same as those of the references. Their detail can be found in the relevant references listed in Table 1.

4.2 Parametric Analyses

In HEP, there are two parameters needed to tune. One is the generation interval for calculating toughness, κ; the other is the value of ν which is used to select ν top-ranked individuals in the ν-local search method.

In order to determine the best value of κ, we test various values of κ at 5, 10, 15, 20, 25, 30, 35, 40, 45, 50.

Table 1 Descriptions of Functions

Function	Name	Dimension	Bounds	Global minimum	Reference
F1	Generalized Schwefels Problem 2.26	30	$[-500, 500]^n$	12569.5	[39]
F2	Generalized Rastrigins Function	30	$[-5.12, 5.12]^n$	0	[39]
F3	Ackleys Function	30	$[-32, 32]^n$	0	[39]
F4	Generalized Griewank Function	30	$[-600, 600]^n$	0	[39]
F5	Generalized Penalized Functions	30	$[-50, 50]^n$	0	[39]
F6	Generalized Penalized Functions2	30	$[-50, 50]^n$	0	[39]
F7	Shekels Foxholes Function	2	$[-65.53, 65.53]^n$	1	[39]
F8	Kowaliks Function	4	$[-5, 5]^n$	0.0003075	[39]
F9	Six-Hump Camel-Back Function	2	$[-5, 5]^n$	-1.0316285	[39]
F10	Branin Function	2	$[-5, 10] \times [0, 15]$	0.398	[39]
F11	Goldstein-Price Function	2	$[-2, 2]^n$	3	[39]
F12	Hartmans Family1	3	$[0, 1]^n$	-3.86	[39]
F13	Hartmans Family2	6	$[0, 1]^n$	-3.32	[39]
F14	Shekels Family1	4	$[0, 10]^n$	-10	[39]
F15	Shekels Family2	4	$[0, 10]^n$	-10	[39]
F16	Shekels Family3	4	$[0, 10]^n$	-10	[39]
F17	Sum of different power	30	$[-1, 1]^n$	0	[34]
F18	Beale function	10	$[-5, 10]^n$	0	[34]
F19	Alpine function	10	$[-10, 10]^n$	0	[34]
F20	Inverted cosine wave function (Masters)	10	$[-5, 5]^n$	-n + 1	[34]
F21	Hyper-Ellipsoid	10	$[-100, 100]^n$	0	[33]
F22	Neumaier #3	30	$[-900, 900]^n$	-4930	[33]
F23	Salomon	10	$[-10, 10]^n$	0	[33]
F24	LennardJones	15	$[-2, 2]^n$	-	[33]
F25	Odd Square	20	$[-5\pi, 5\pi]^n$	-1.14383	[33]
F26	Katsuura	10	$[-1000, 1000]^n$	1	[33]
F27	Bohachevsky 1 Problem (BF1)	2	$[-50, 50]^n$	0	[1]
F28	Camel Back 3 Three Hump Problem (CB3)	2	$[-5, 5]^n$	0	[1]
F29	Cosine Mixture Problem (CM)	10	$[-100, 100]^n$	-	[1]
F30	Easom Problem (EP)	2	$[-10, 10]^n$	-1	[1]
F31	Epistatic Michalewicz Problem (EM)	10	$[0, \pi]^n$	-9.660152	[1]
F32	Exponential Problem (EXP)	30	$[-1, 1]^n$	-1	[1]
F33	Meyer and Roth Problem	3	$[-10, 10]^n$	0.00004	[1]
F34	Modified Rosenbrock Problem (MRP)	2	$[-5, 5]^n$	0	[1]
F35	Multi-Gaussian Problem (MGP)	2	$[-2, 2]^n$	1.29695	[1]
F36	Paviani Problem (PP)	10	$[2, 10]^n$	-45.778	[1]
F37	Schaffer 2 Problem (SF2)	2	$[-10, 10]^n$	0	[1]
F38	Shubert Problem (SBT)	2	$[-10, 10]^n$	-186.7309	[1]
F39	Sinusoidal Problem (SIN)	10	$[0, 180]^n$	-3.5	[1]

For all functions, we set $\nu = 6$ in the modified Nelder-Mead exploitation operator. Each test is repeated 50 times. The best and average fitness at different interval values are shown in Table 2 3. From the results, we observe that for most functions, the best result is obtained at the interval $\kappa = 30$. Therefore we set the interval of calculating roughness $\kappa = 30$.

In order to determine the best value of ν, we test various values of ν at 3, 6, 9, 12, 15. Each test is repeated 50 times. The best and average fitness at different values of ν are shown in Table 4. From these results, we see that for most functions, the best result is obtained at $\nu = 6$. Therefore, we set $\kappa = 6$ in experiments.

4.3 Comparison Experiment 1

The first experiment compares the performance of HEP with other EP algorithms, which are

1. Classical EP (CEP),
2. Fast EP (FEP) [39],
3. EP with Adaptive Lévy Mutation (ALEP) [24]
4. Mixed EP (MEP) [5],
5. Reinforcement Learning EP (RLEP) [40],
6. Adaptive FEP (AFEP) [27]
7. HEP without the modified Nelder-Mead exploitation operator (HEP without NM).

The set of 25 benchmark functions in the CEC2005 special session on real-parameter optimization [38] are utilized in the experiment. The detail of these functions can be referred to [38] and is omitted here. This benchmark suit includes unimodal functions (F1-F5), basic

Table 2 Fitness using Different Intervals of Calculating Roughness Part 1

Parameter		F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13
5	best	-1.11E+04	0.00E+00	0.00E+00	0.00E+00	7.00E-10	1.07E-08	9.98E-01	3.57E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.32E+00
	mean	-1.05E+04	1.79E+00	0.00E+00	0.00E+00	2.07E-02	2.20E-03	9.98E-01	5.81E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.32E+00
	std	5.64E+02	3.58E+00	0.00E+00	0.00E+00	4.15E-02	4.39E-03	0.00E+00	1.28E-04	5.82E-13	5.70E-14	2.17E-12	6.35E-10	1.06E-09
10	best	-1.08E+04	0.00E+00	0.00E+00	0.00E+00	6.27E-10	7.03E-09	9.98E-01	3.56E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.32E+00
	mean	-1.04E+04	2.19E+00	0.00E+00	0.00E+00	9.90E-10	2.20E-03	1.40E+00	5.85E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.25E+00
	std	2.57E+02	4.38E+00	0.00E+00	0.00E+00	2.91E-10	4.39E-03	4.87E-01	1.33E-04	1.12E-13	6.77E-14	3.17E-11	3.01E-10	5.82E-02
15	best	-1.10E+04	0.00E+00	0.00E+00	0.00E+00	6.83E-10	9.05E-09	9.98E-01	3.08E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.32E+00
	mean	-1.03E+04	1.23E+01	0.00E+00	0.00E+00	1.03E-09	4.39E-03	1.20E+00	5.20E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.27E+00
	std	5.63E+02	1.42E+01	0.00E+00	0.00E+00	1.92E-10	5.38E-03	3.98E-01	1.68E-04	9.06E-13	7.43E-14	2.11E-11	5.21E-09	5.82E-02
20	best	-1.10E+04	0.00E+00	0.00E+00	0.00E+00	7.54E-10	1.07E-08	9.98E-01	5.30E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.32E+00
	mean	-1.01E+04	3.39E+00	0.00E+00	0.00E+00	2.08E-01	2.20E-03	2.19E+00	6.22E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.32E+00
	std	5.07E+02	6.77E+00	0.00E+00	0.00E+00	4.16E-01	4.39E-03	1.16E+00	9.52E-05	8.38E-13	2.36E-13	1.46E-11	1.67E-09	6.89E-10
25	best	-1.06E+04	0.00E+00	0.00E+00	0.00E+00	5.72E-10	8.63E-09	9.98E-01	3.10E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.32E+00
	mean	-1.03E+04	2.59E+00	0.00E+00	0.00E+00	2.07E-02	8.36E-03	1.99E+00	5.37E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.30E+00
	std	3.04E+02	5.17E+00	0.00E+00	0.00E+00	4.15E-02	1.20E-02	8.87E-01	1.75E-04	1.15E-12	8.09E-14	2.26E-11	5.28E-11	4.76E-02
30	best	-1.11E+04	0.00E+00	0.00E+00	0.00E+00	5.34E-10	9.24E-09	9.98E-01	3.09E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.32E+00
	mean	-1.10E+04	2.58E+00	0.00E+00	0.00E+00	1.08E-09	1.10E-08	9.99E-01	5.09E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.32E+00
	std	4.09E+02	5.92E+00	0.00E+00	0.00E+00	3.62E-10	1.68E-09	3.98E-01	1.21E-04	2.96E-14	1.80E-13	4.72E-13	1.11E-09	4.75E-02
35	best	-1.08E+04	0.00E+00	0.00E+00	0.00E+00	7.93E-10	5.40E-09	9.98E-01	3.83E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.32E+00
	mean	-1.04E+04	0.00E+00	0.00E+00	0.00E+00	2.07E-02	1.37E-08	1.98E+00	5.23E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.30E+00
	std	3.29E+02	0.00E+00	0.00E+00	0.00E+00	4.15E-02	4.89E-09	1.97E+00	1.14E-04	1.38E-13	5.84E-14	3.94E-11	2.97E-10	4.76E-02
40	best	-1.07E+04	0.00E+00	0.00E+00	0.00E+00	9.58E-10	6.53E-09	9.98E-01	3.38E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.32E+00
	mean	-1.03E+04	1.15E+01	0.00E+00	0.00E+00	1.87E-09	2.06E-08	1.40E+00	5.88E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.30E+00
	std	2.97E+02	1.48E+01	0.00E+00	0.00E+00	9.95E-10	1.22E-08	4.87E-01	1.38E-04	1.18E-12	2.00E-14	4.63E-12	8.18E-10	4.76E-02
45	best	-1.07E+04	0.00E+00	0.00E+00	0.00E+00	8.77E-10	5.75E-09	9.98E-01	3.36E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.32E+00
	mean	-1.01E+04	2.79E+00	0.00E+00	0.00E+00	1.53E-09	4.39E-03	9.98E-01	6.94E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.25E+00
	std	4.37E+02	5.57E+00	0.00E+00	0.00E+00	4.32E-10	5.38E-03	0.00E+00	2.98E-04	1.00E-12	1.76E-13	5.87E-12	1.75E-10	5.82E-02
50	best	-1.06E+04	0.00E+00	0.00E+00	0.00E+00	5.01E-10	1.21E-08	9.98E-01	5.43E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.32E+00
	mean	-1.01E+04	9.55E+00	0.00E+00	0.00E+00	9.51E-10	2.20E-03	9.98E-01	6.36E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.30E+00
	std	4.23E+02	7.97E+00	0.00E+00	0.00E+00	3.57E-10	4.39E-03	0.00E+00	7.42E-05	1.31E-13	4.78E-14	3.16E-11	5.56E-10	4.76E-02
parameter		F14	F15	F16	F17	F18	F19	F20	F21	F22	F23	F24	F25	F26
5	best	-1.02E+01	-1.04E+01	-1.05E+01	0.00E+00	9.41E-05	0.00E+00	-7.17E+00	0.00E+00	-1.79E+03	9.99E-02	-9.10E+00	-8.00E-02	1.00E+00
	mean	-1.02E+01	-9.07E+00	-9.20E+00	0.00E+00	9.41E-05	1.98E-05	-6.75E+00	0.00E+00	-1.59E+03	1.20E-01	-9.10E+00	-4.51E-02	1.00E+00
	std	1.36E-07	2.67E+00	2.68E+00	0.00E+00	2.14E-14	1.51E-05	2.92E-01	0.00E+00	1.18E+02	4.00E-02	1.76E-07	2.20E-02	0.00E+00
10	best	-1.02E+01	-1.04E+01	-1.05E+01	0.00E+00	9.41E-05	0.00E+00	-7.17E+00	0.00E+00	-1.69E+03	0.00E+00	-9.10E+00	-2.86E-01	1.00E+00
	mean	-1.02E+01	-8.71E+00	-8.91E+00	0.00E+00	9.41E-05	8.29E-06	-6.82E+00	0.00E+00	-1.62E+03	7.99E-02	-9.10E+00	-1.43E-01	1.00E+00
	std	1.13E-06	2.18E+00	3.24E+00	0.00E+00	4.67E-14	6.17E-06	2.20E-01	0.00E+00	1.02E+02	3.99E-02	3.61E-08	9.20E-02	0.00E+00
15	best	-1.02E+01	-1.04E+01	-1.05E+01	0.00E+00	9.41E-05	0.00E+00	-7.17E+00	0.00E+00	-1.69E+03	9.99E-02	-9.10E+00	-1.10E-01	1.00E+00
	mean	-7.17E+00	-1.04E+01	-1.05E+01	0.00E+00	9.41E-05	1.01E-05	-6.71E+00	0.00E+00	-1.58E+03	1.20E-01	-9.10E+00	-3.36E-02	1.00E+00
	std	3.66E+00	3.82E-08	5.17E-07	0.00E+00	8.02E-14	7.35E-06	2.38E-01	0.00E+00	8.51E+01	4.00E-02	5.02E-08	3.93E-02	0.00E+00
20	best	-1.02E+01	-1.04E+01	-1.05E+01	0.00E+00	9.41E-05	0.00E+00	-7.69E+00	0.00E+00	-1.74E+03	9.99E-02	-9.10E+00	-1.10E-01	1.00E+00
	mean	-9.36E+00	-8.88E+00	-8.11E+00	0.00E+00	9.41E-05	1.08E-05	-6.54E+00	0.00E+00	-1.65E+03	9.99E-02	-9.10E+00	-3.51E-02	1.00E+00
	std	1.59E+00	3.05E+00	2.99E+00	0.00E+00	5.61E-14	1.53E-05	8.08E-01	0.00E+00	5.61E+01	1.86E-17	3.23E-08	3.86E-02	0.00E+00
25	best	-1.02E+01	-1.04E+01	-1.05E+01	0.00E+00	9.41E-05	0.00E+00	-7.43E+00	0.00E+00	-1.60E+03	9.99E-02	-9.10E+00	-8.00E-02	1.00E+00
	mean	-1.02E+01	-1.04E+01	-1.05E+01	0.00E+00	9.41E-05	6.06E-06	-6.82E+00	0.00E+00	-1.51E+03	9.99E-02	-9.10E+00	-4.35E-02	1.00E+00
	std	3.09E-07	1.43E-07	2.03E-07	0.00E+00	1.61E-14	5.19E-06	3.37E-01	0.00E+00	6.38E+01	1.64E-17	1.11E-07	3.25E-02	0.00E+00
30	best	-1.02E+01	-1.04E+01	-1.05E+01	0.00E+00	9.41E-05	0.00E+00	-8.48E+00	0.00E+00	-1.74E+03	0.00E+00	-9.10E+00	-5.82E-02	1.00E+00
	mean	-1.02E+01	-1.04E+01	-1.05E+01	0.00E+00	9.41E-05	2.83E-36	-8.38E+00	0.00E+00	-1.59E+03	7.99E-02	-9.10E+00	-3.82E-02	1.00E+00
	std	2.02E+00	2.91E-08	2.20E-07	0.00E+00	8.67E-15	3.48E-06	6.83E-01	0.00E+00	1.10E+02	3.99E-02	3.01E-08	1.93E-02	0.00E+00
35	best	-1.02E+01	-1.04E+01	-1.05E+01	0.00E+00	9.41E-05	0.00E+00	-7.17E+00	0.00E+00	-1.63E+03	9.99E-02	-9.10E+00	-1.10E-01	1.00E+00
	mean	-8.13E+00	-1.04E+01	-1.05E+01	0.00E+00	9.41E-05	9.76E-06	-6.80E+00	0.00E+00	-1.58E+03	9.99E-02	-9.10E+00	-2.72E-02	1.00E+00
	std	2.48E+00	6.21E-08	4.45E-08	0.00E+00	2.80E-14	9.67E-06	2.10E-01	0.00E+00	3.89E+01	1.74E-05	3.46E-03	4.14E-02	0.00E+00
40	best	-1.02E+01	-1.04E+01	-1.05E+01	0.00E+00	9.41E-05	0.00E+00	-8.21E+00	0.00E+00	-1.74E+03	9.99E-02	-9.10E+00	-1.51E-01	1.00E+00
	mean	-7.19E+00	-9.07E+00	-1.05E+01	0.00E+00	9.41E-05	6.18E-06	-7.38E+00	0.00E+00	-1.49E+03	9.99E-02	-9.10E+00	-5.69E-02	1.00E+00
	std	3.63E+00	2.67E+00	4.42E-07	0.00E+00	2.09E-14	1.24E-05	6.06E-01	0.00E+00	1.84E+02	1.76E-17	7.48E-08	4.92E-02	0.00E+00
45	best	-1.02E+01	-1.04E+01	-1.05E+01	0.00E+00	9.41E-05	0.00E+00	-7.69E+00	0.00E+00	-1.72E+03	5.31E-17	-9.10E+00	-1.51E-01	1.00E+00
	mean	-1.02E+01	-1.04E+01	-1.05E+01	0.00E+00	9.41E-05	2.40E-06	-6.76E+00	0.00E+00	-1.64E+03	8.04E-02	-9.10E+00	-6.13E-02	1.00E+00
	std	9.90E-07	1.12E-07	2.44E-06	0.00E+00	8.42E-14	2.94E-06	4.94E-01	0.00E+00	6.52E+01	4.02E-02	6.41E-08	5.88E-02	0.00E+00
50	best	-1.02E+01	-1.04E+01	-1.05E+01	0.00E+00	9.41E-05	0.00E+00	-7.43E+00	0.00E+00	-1.76E+03	9.99E-02	-9.10E+00	-1.10E-01	1.00E+00
	mean	-8.12E+00	-9.07E+00	-9.32E+00	0.00E+00	9.41E-05	7.19E-06	-6.90E+00	0.00E+00	-1.62E+03	1.20E-01	-9.10E+00	-7.18E-02	1.00E+00
	std	2.49E+00	2.67E+00	2.43E+00	0.00E+00	2.67E-14	1.44E-05	3.31E-01	0.00E+00	7.84E+01	4.00E-02	6.17E-08	2.63E-02	0.00E+00

multimodal functions (F6-F12), expanded multimodal functions (F13-F14) and hybrid composition functions (F15-F25). All the functions are tested in 10 dimensions. Twenty-five independent runs of each algorithm on each problem are taken. Each run is continued up to 100,000 Function Evaluations (FES). The mean and the standard deviation (std) of the best-of-run errors are reported in Table 5. Note that the best-of-the-run error corresponds to absolute difference between the best-of-the-run value and the actual optimum of a particular objective function.

From the data given in Table 5, we can make several observations and conclusions. First, for unimodal functions, HEP shows the best performance. HEP without exploitation is also very competitive. Secondly, for basic multi-modal benchmark functions, HEP shows a better performance than other comparative ones. Thirdly, as for expanded multimodal functions, HEP exhibits better overall performance than most of these comparative EP variants except MEP and AFEP on function F13, and AFEP on function F14. Finally, with regard to the more complex hybrid composition functions, HEP generally performs worse than AFEP and FEP. However, HEP is superior to AFEP on functions F15, F22 and F23. And it outperforms FEP on functions F15, , F16, F17, F20 and F23. Furthermore, it equals to AFEP and FEP on function F24. In summary, HEP has the best over all performance compared with other seven competitors on all the 25 benchmark functions with 10 variables.

Table 3 Fitness using Different Intervals of Calculating Roughness Part 2

parameter		F27	F28	F29	F30	F31	F32	F33	F34	F35	F36	F37	F38	F39
5	best	0.00E+00	0.00E+00	-1.00E+05	-1.00E+00	-8.65E+00	-1.00E+00	1.90E-03	2.62E-14	-1.30E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.50E+00
	mean	0.00E+00	6.24E-18	-1.00E+05	-1.00E+00	-8.29E+00	-1.00E+00	1.90E-03	1.48E-03	-1.28E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.18E+00
	std	0.00E+00	1.25E-17	1.37E-02	2.17E-13	2.61E-01	0.00E+00	1.64E-07	2.97E-03	3.21E-02	1.34E-09	0.00E+00	4.07E-11	3.96E-01
10	best	0.00E+00	9.41E-54	-1.00E+05	-1.00E+00	-8.99E+00	-1.00E+00	1.90E-03	7.16E-14	-1.30E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.50E+00
	mean	8.88E-18	8.51E-22	-1.00E+05	-1.00E+00	-7.97E+00	-1.00E+00	2.09E-03	1.48E-03	-1.28E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.50E+00
	std	1.78E-17	1.15E-21	6.69E-03	3.29E-13	7.50E-01	0.00E+00	3.77E-04	2.97E-03	3.21E-02	1.19E-09	0.00E+00	1.50E-09	1.29E-09
15	best	0.00E+00	0.00E+00	-1.00E+05	-1.00E+00	-8.42E+00	-1.00E+00	1.90E-03	1.82E-12	-1.30E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.50E+00
	mean	0.00E+00	6.91E-18	-1.00E+05	-1.00E+00	-7.51E+00	-1.00E+00	1.90E-03	1.48E-03	-1.30E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.34E+00
	std	0.00E+00	1.38E-17	9.44E-03	5.54E-14	6.94E-01	0.00E+00	1.96E-08	2.97E-03	3.77E-03	9.02E-10	0.00E+00	4.10E-10	3.23E-01
20	best	0.00E+00	0.00E+00	-1.00E+05	-1.00E+00	-9.20E+00	-1.00E+00	1.90E-03	1.50E-14	-1.30E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.50E+00
	mean	5.99E-17	3.21E-21	-1.00E+05	-1.00E+00	-7.95E+00	-1.00E+00	1.90E-03	1.04E-05	-1.30E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.34E+00
	std	9.92E-17	5.46E-21	4.49E-03	5.64E-14	7.40E-01	0.00E+00	3.51E-06	2.09E-05	1.80E-13	9.40E-10	0.00E+00	1.29E-09	3.23E-01
25	best	0.00E+00	2.92E-52	-1.00E+05	-1.00E+00	-8.77E+00	-1.00E+00	1.90E-03	7.13E-15	-1.30E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.50E+00
	mean	0.00E+00	2.00E-23	-1.00E+05	-1.00E+00	-7.66E+00	-1.00E+00	1.90E-03	7.53E-12	-1.30E+00	-4.58E+01	1.63E-74	-1.87E+02	-3.06E+00
	std	0.00E+00	3.81E-23	1.57E-02	9.37E-14	6.67E-01	0.00E+00	6.45E-08	1.31E-11	3.11E-03	8.28E-10	3.27E-74	2.62E-09	5.65E-01
30	best	0.00E+00	0.00E+00	-1.00E+05	-1.00E+00	-9.11E+00	-1.00E+00	1.90E-03	1.34E-16	-1.30E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.50E+00
	mean	0.00E+00	4.48E-31	-1.00E+05	-1.00E+00	-9.01E+00	-1.00E+00	1.90E-03	2.97E-08	-1.30E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.34E+00
	std	1.47E-16	8.95E-21	3.90E-03	1.99E-14	6.78E-01	0.00E+00	4.16E-07	3.63E-03	2.16E-02	1.42E-09	0.00E+00	1.26E-12	3.23E-01
35	best	0.00E+00	0.00E+00	-1.00E+05	-1.00E+00	-8.47E+00	-1.00E+00	1.90E-03	4.93E-16	-1.30E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.50E+00
	mean	1.11E-17	1.99E-19	-1.00E+05	-1.00E+00	-8.15E+00	-1.00E+00	1.90E-03	4.85E-06	-1.28E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.18E+00
	std	2.22E-17	3.97E-19	9.69E-03	8.80E-14	4.20E-01	0.00E+00	2.57E-07	9.69E-06	3.21E-02	9.09E-10	0.00E+00	2.18E-10	3.96E-01
40	best	0.00E+00	0.00E+00	-1.00E+05	-1.00E+00	-9.31E+00	-1.00E+00	1.90E-03	1.12E-14	-1.30E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.50E+00
	mean	0.00E+00	3.08E-21	-1.00E+05	-1.00E+00	-7.93E+00	-1.00E+00	1.90E-03	1.48E-03	-1.30E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.50E+00
	std	0.00E+00	6.14E-21	4.92E-03	6.66E-14	7.51E-01	0.00E+00	2.72E-06	2.97E-03	1.72E-13	7.49E-10	0.00E+00	4.68E-08	1.20E-09
45	best	0.00E+00	1.46E-24	-1.00E+05	-1.00E+00	-8.75E+00	-1.00E+00	1.90E-03	1.61E-12	-1.30E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.50E+00
	mean	0.00E+00	8.84E-20	-1.00E+05	-1.00E+00	-8.16E+00	-1.00E+00	1.90E-03	1.48E-03	-1.30E+00	-4.58E+01	0.00E+00	-1.86E+02	-3.50E+00
	std	0.00E+00	1.75E-19	9.14E-03	4.76E-14	3.92E-01	0.00E+00	1.79E-08	2.97E-03	7.32E-04	9.68E-10	0.00E+00	2.36E+00	1.65E-09
50	best	0.00E+00	0.00E+00	-1.00E+05	-1.00E+00	-8.64E+00	-1.00E+00	1.90E-03	1.05E-12	-1.30E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.50E+00
	mean	9.77E-17	7.80E-20	-1.00E+05	-1.00E+00	-8.15E+00	-1.00E+00	1.90E-03	1.48E-03	-1.30E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.18E+00
	std	1.69E-16	1.56E-19	8.67E-03	1.05E-13	3.54E-01	0.00E+00	5.05E-07	2.97E-03	1.52E-13	6.82E-10	0.00E+00	1.24E-10	3.96E-01

4.4 Comparison Experiment 2

This experiment compares HEP with other state-of-art algorithms plus HEP without the modified Nelder-Mead exploitation operator, which are

1. Classical EP (CEP),
2. Fast EP (FEP) [39],
3. Lévy distributed function EP (LEP) [24],
4. Exponential distributed function EP (EEP) [29],
5. Reinforcement Learning EP (RLEP) [40],
6. Mixed EP (MEP) [5],
7. Shifted Classical EP (SCEP) [3],
8. Momentum Coefficient EP (MCEP) [4],
9. Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [19],
10. Increasing Inertia Weighting PSO (IIW-PSO) [11],
11. Jumping Gene GA (JG-GA) [36],
12. Fuzzy Routing Method for Evolutionary Programming (FREP) [2].

In order to make a fair comparison, all parameters of the above algorithms are set to the same as introduced in Table 6.

The algorithms are tested on 39 benchmark functions. Each test is repeated 50 times. The average results are shown in Table 7. All algorithms are run until a pre-specified number of generations is reached. The number of generations is shown in the last column of Table 7. This table compares the accuracy of the algorithms in obtaining the global minimum. The proposed method performs better (or at least equal to other methods) in more than 75% of the benchmark functions. The distances between the optimal points found by HEP and other algorithms show that the performance of the proposed method is notable.

In recent years, the use of non-parametric statistical tests becomes an important methodology for comparing a group of evolutionary algorithms [18,12]. In this paper, the Friedman test is employed to estimate the differences among the 14 algorithms. Table 7 demonstrates that HEP is significantly different from other algorithms because of $p < 0.01$ except FREP, which is similar to HEP because of $p > 0.05$.

4.5 Comparison Experiment 3

The performance of HEP is evaluated according to the rules of the CEC2017 Special Session and Competition on Single Objective Real-Parameter Numerical Optimization, bound constrained case[8]. The test suite which consists of 30 minimization test functions is divided into four categories viz., unimodal functions (F1-F3), multimodal functions (F4-F10), hybrid functions (F11-F20) and composition functions (F21-F30). 51 independent runs of the algorithm were executed for all 30 functions in 10 dimensions. Each run was terminated after a maximum of 100000 fitness evaluations of the objective function.

The third experiment compares the performance of HEP with other algorithms, which are

1. CMA-ES with increasing population size (IPOP-CMA-ES) [7],
2. Teaching Learning Based Optimization with Focused Learning (TLBO-FL) [23],
3. Classical EP (CEP),

Table 4 Fitness using Different ν Values

parameter		F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13
3	best	-1.08E+04	0.00E+00	0.00E+00	0.00E+00	3.85E-10	1.11E-08	9.98E-01	6.39E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.32E+00
	mean	-1.04E+04	1.07E+01	0.00E+00	0.00E+00	2.07E-02	4.39E-03	2.38E+00	6.94E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.32E+00
	std	2.77E+02	6.11E+00	0.00E+00	0.00E+00	4.15E-02	5.38E-03	1.93E+00	4.77E-05	1.31E-13	2.67E-13	7.12E-12	5.43E-10	1.43E-09
6	best	-1.11E+04	0.00E+00	0.00E+00	0.00E+00	5.34E-10	9.24E-09	9.98E-01	3.09E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.32E+00
	mean	-1.10E+04	2.58E+00	0.00E+00	0.00E+00	1.08E-09	1.10E-08	9.99E-01	5.09E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.32E+00
	std	4.09E+02	5.92E+00	0.00E+00	0.00E+00	3.62E-10	1.68E-09	3.98E-01	1.21E-04	2.96E-14	1.80E-13	4.72E-13	1.11E-09	4.75E-02
9	best	-1.15E+04	1.09E+01	3.59E-04	7.40E-03	1.95E-09	2.59E-08	9.98E-01	3.23E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.32E+00
	mean	-1.10E+04	1.35E+01	4.12E-04	6.56E-02	8.29E-02	4.39E-03	1.79E+00	6.01E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.30E+00
	std	3.81E+02	3.75E+00	4.81E-05	6.30E-02	4.15E-02	5.38E-03	1.15E+00	1.51E-04	1.03E-12	9.33E-13	6.28E-11	1.37E-09	4.76E-02
12	best	-1.14E+04	4.98E+00	2.84E-04	2.70E-02	1.53E-09	1.28E-08	9.98E-01	3.61E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.32E+00
	mean	-1.09E+04	1.36E+01	3.67E-04	2.00E-01	1.45E-01	2.20E-03	2.18E+00	5.29E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.30E+00
	std	3.44E+02	6.25E+00	7.19E-05	2.11E-01	2.42E-01	4.39E-03	1.91E+00	1.15E-04	1.52E-12	5.71E-13	1.07E-10	6.13E-09	4.76E-02
15	best	-1.14E+04	7.96E+00	3.44E-04	1.12E-08	1.33E-09	2.56E-08	9.98E-01	4.89E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.32E+00
	mean	-1.08E+04	1.11E+01	3.97E-04	3.62E-02	2.29E-09	2.20E-03	1.40E+00	6.22E-04	-1.03E+00	3.98E-01	3.00E+00	-3.86E+00	-3.27E+00
	std	4.22E+02	4.05E+00	4.24E-05	2.69E-02	7.40E-10	4.39E-03	4.87E-01	9.19E-05	6.67E-13	1.49E-13	2.69E-10	4.07E-10	5.82E-02
parameter		F14	F15	F16	F17	F18	F19	F20	F21	F22	F23	F24	F25	F26
3	best	-1.02E+01	-1.04E+01	-1.05E+01	0.00E+00	9.41E-05	0.00E+00	-7.43E+00	0.00E+00	-1.81E+03	9.99E-02	-9.10E+00	-3.08E-02	1.00E+00
	mean	-9.14E+00	-8.88E+00	-1.05E+01	0.00E+00	9.41E-05	1.86E-05	-6.80E+00	0.00E+00	-1.62E+03	9.99E-02	-9.10E+00	-1.08E-02	1.00E+00
	std	2.02E+00	3.05E+00	4.17E-08	0.00E+00	7.30E-14	2.14E-05	3.14E-01	0.00E+00	1.10E+02	2.71E-17	3.63E-08	1.32E-02	0.00E+00
6	best	-1.02E+01	-1.04E+01	-1.05E+01	0.00E+00	9.41E-05	0.00E+00	-8.48E+00	0.00E+00	-1.74E+03	0.00E+00	-9.10E+00	-5.82E-02	1.00E+00
	mean	-1.02E+01	-1.04E+01	-1.05E+01	0.00E+00	9.41E-05	2.83E-36	-8.38E+00	0.00E+00	-1.59E+03	7.99E-02	-9.10E+00	-3.82E-02	1.00E+00
	std	2.02E+00	2.91E-08	2.20E-07	0.00E+00	8.67E-15	3.48E-06	6.83E-01	0.00E+00	1.10E+02	3.99E-02	3.01E-08	1.93E-02	0.00E+00
9	best	-1.02E+01	-1.04E+01	-1.05E+01	3.84E-12	9.41E-05	3.86E-05	-6.90E+00	3.71E-08	-1.63E+03	2.00E-01	-9.10E+00	-4.80E-08	1.00E+00
	mean	-7.67E+00	-1.04E+01	-8.27E+00	6.09E-12	9.41E-05	9.75E-05	-6.74E+00	4.85E-08	1.34E+04	3.00E-01	-9.10E+00	-9.65E-09	1.00E+00
	std	3.12E+00	5.35E-06	2.85E+00	1.71E-12	8.92E-14	5.17E-05	1.25E-01	1.11E-08	1.01E+04	1.10E-01	1.73E-07	1.92E-08	0.00E+00
12	best	-1.02E+01	-1.04E+01	-1.05E+01	1.44E-12	9.41E-05	1.97E-05	-7.69E+00	1.98E-08	-3.68E+02	2.00E-01	-9.10E+00	-8.44E-07	1.00E+00
	mean	-7.15E+00	-1.04E+01	-1.05E+01	8.46E-12	9.41E-05	3.35E-05	-6.90E+00	3.14E-08	1.51E+04	3.00E-01	-8.93E+00	-1.69E-07	1.00E+00
	std	3.67E+00	3.65E-06	4.28E-06	7.33E-12	1.66E-13	1.14E-05	4.06E-01	1.18E-08	1.21E+04	6.32E-02	3.51E-01	3.38E-07	0.00E+00
15	best	-1.02E+01	-1.04E+01	-1.05E+01	2.68E-12	9.41E-05	8.32E-06	-6.90E+00	4.02E-08	-4.05E+03	3.00E-01	-9.10E+00	-3.57E-04	1.00E+00
	mean	-7.64E+00	-9.35E+00	-1.05E+01	9.83E-12	9.41E-05	4.53E-05	-6.52E+00	4.60E-08	6.57E+03	3.40E-01	-9.10E+00	-7.14E-05	1.00E+00
	std	3.17E+00	2.11E+00	1.54E-04	4.30E-12	3.20E-14	3.43E-05	3.92E-01	8.33E-09	7.54E+03	8.00E-02	4.97E-08	1.43E-04	0.00E+00
parameter		F27	F28	F29	F30	F31	F32	F33	F34	F35	F36	F37	F38	F39
3	best	0.00E+00	1.76E-25	-1.00E+05	-1.00E+00	-8.03E+00	-1.00E+00	1.90E-03	1.65E-13	-1.30E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.50E+00
	mean	1.96E-13	1.78E-19	-1.00E+05	-1.00E+00	-7.43E+00	-1.00E+00	1.90E-03	2.97E-03	-1.28E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.34E+00
	std	3.93E-13	2.24E-19	3.78E-03	8.22E-14	4.38E-01	0.00E+00	5.32E-08	3.64E-03	3.21E-02	8.96E-10	0.00E+00	8.09E-10	3.23E-01
6	best	0.00E+00	0.00E+00	-1.00E+05	-1.00E+00	-9.11E+00	-1.00E+00	1.90E-03	1.34E-16	-1.30E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.50E+00
	mean	0.00E+00	4.48E-31	-1.00E+05	-1.00E+00	-9.01E+00	-1.00E+00	1.90E-03	2.97E-08	-1.30E+00	-4.58E+01	0.00E+00	-1.87E+02	-3.34E+00
	std	1.47E-16	8.95E-21	3.90E-03	1.99E-14	6.78E-01	0.00E+00	4.16E-07	3.63E-03	2.16E-02	1.42E-09	0.00E+00	1.26E-12	3.23E-01
9	best	1.36E-12	2.40E-14	-1.00E+05	-1.00E+00	-8.83E+00	-1.00E+00	1.90E-03	4.79E-12	-1.30E+00	-4.58E+01	4.66E-04	-1.87E+02	-3.50E+00
	mean	2.78E-11	4.12E-13	-1.00E+05	-1.00E+00	-8.22E+00	-1.00E+00	1.90E-03	4.45E-03	-1.30E+00	-4.58E+01	5.61E-04	-1.87E+02	-3.50E+00
	std	1.99E-11	3.05E-13	8.81E-03	3.10E-13	5.48E-01	7.96E-07	5.08E-08	3.63E-03	2.31E-03	9.52E-10	7.97E-05	2.85E-10	2.00E-09
12	best	5.33E-12	8.70E-14	-1.00E+05	-1.00E+00	-8.38E+00	-1.00E+00	1.90E-03	3.28E-12	-1.30E+00	-4.58E+01	3.68E-04	-1.87E+02	-3.50E+00
	mean	1.54E-11	8.13E-13	-1.00E+05	-1.00E+00	-7.63E+00	-1.00E+00	1.90E-03	2.97E-03	-1.30E+00	-4.58E+01	5.41E-04	-1.87E+02	-3.34E+00
	std	9.19E-12	4.45E-13	3.42E-03	3.22E-13	7.36E-01	1.17E-05	2.99E-07	3.64E-03	3.79E-12	1.57E-09	1.31E-04	9.82E-10	3.23E-01
15	best	3.28E-13	1.25E-13	-1.00E+05	-1.00E+00	-8.27E+00	-1.00E+00	1.90E-03	2.21E-12	-1.30E+00	-4.58E+01	4.63E-04	-1.87E+02	-3.50E+00
	mean	7.32E-12	5.47E-13	-1.00E+05	-1.00E+00	-7.87E+00	-1.00E+00	1.90E-03	1.48E-03	-1.29E+00	-4.58E+01	6.45E-04	-1.87E+02	-3.34E+00
	std	7.41E-12	4.92E-13	6.55E-03	4.58E-13	4.31E-01	4.73E-07	8.34E-08	2.97E-03	2.03E-02	8.41E-10	9.91E-05	1.41E-09	3.23E-01

4. Fast EP (FEP) [39],
5. EP with Adaptive Lévy Mutation (ALEP) [24],
6. HEP without the modified Nelder-Mead exploitation operator (HEP without NM).

The mean and the standard deviation (std) of the best-of-run errors are given in Table 8. From the data we can see, for unimodal functions, HEP shows the best performance. HEP without exploitation also shows a better performance. Furthermore, for multimodal benchmark functions, HEP shows a better overall performance than other comparative ones. However, with regard to hybrid functions, HEP exhibits a better performance than most of other comparative ones except ALEP. Nevertheless, as for composition functions, HEP generally performs better than other competitors.

Generally, HEP shows a good performance on most of the test functions particularly on unimodal functions. However, for other functions, the performance of HEP is poor. The main reason leading to this issue is probably that the modified Nelder-Mead method works well on most of the functions, but the method is hard to find the global optimum via updating three points each time on other functions, especially when the three points belong to different peaks.

5 Conclusions

The performance of EP is related to both mutation operators and fitness landscapes. In order to make EP more efficient, its mutation operator should adapt to fitness landscapes. The paper presents a novel hybrid EP algorithm with adaptive Lévy mutation (HEP). In HEP, the shape parameter α of Lévy probability distribution adapts to the roughness of a local fitness landscape. Since a fitness landscape can be observed only via a population of finite sample points, its roughness is measured by the number of optima (both local and global) in a population.

Furthermore, HEP follows an essential design principle in memetic algorithms: local search can be added into evolutionary algorithms for improving their exploitation ability. Thus a modified Nelder-Mead method is added into HEP.

The proposed algorithm is tested on the benchmark suit in the CEC2005 and CEC2017 real-parameter optimization and a set of 39 benchmark functions. The experimental results demonstrates that the proposed algorithm has a better overall performance in terms of the solution accuracy than other evolutionary algorithms.

Table 5 Computational result of benchmark functions in CEC2005 with 10 variables

Algorithms		F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
CEP	mean	2.74E-03(7)	1.92E-06(3)	3.83E+07(8)	6.83E-01(5)	3.83E-01(6)	4.79E+01(7)	**2.83E-02**(1)	2.05E+01(7)	1.06E+01(8)	**2.19E+00**(1)
	std	5.74E-03	1.03E-06	1.18E+03	3.01E-01	2.83E-02	3.73E+01	3.83E-02	1.83E-01	1.24E+00	1.18E+00
FEP	mean	5.78E-04(6)	5.01E-04(6)	7.27E+06(7)	2.66E+01(7)	3.07E-01(5)	2.8281+01(6)	1.64E-01(3)	2.04E+01(6)	5.26E+00(7)	1.88E+01(5)
	std	2.83E-04	2.76E-04	3.88E+06	1.66E+00	8.64E-02	2.77E+00	4.73E-01	2.62E-01	2.83E+00	2.62E+00
ALEP	mean	8.34E-01(8)	3.78E-04(5)	2.88E+06(6)	1.79E+00(6)	1.63E-01(4)	1.83E+01(4)	7.64E-01(5)	2.04E+01(5)	2.13E+00(3)	1.91E+01(6)
	std	4.83E-02	2.77E-03	1.63E+04	2.88E+00	9.36E-02	3.77E+00	2.66E-01	3.12E-01	3.73E+00	6.77E+00
MEP	mean	8.22E-08(3)	2.12E-02(7)	3.58E+05(3)	4.45E+01(8)	9.27E+02(8)	3.40E+03(8)	1.27E+03(6.5)	2.02E+01(3)	**1.35E-05**(1)	3.55E+01(8)
	std	4.83E-08	8.72E-02	3.67E+05	3.93E+01	1.42E+03	1.08E+03	0	1.04E-01	1.09E-05	1.26E+01
RLEP	mean	1.93E-04(5)	1.89E-01(8)	6.87E+05(4)	3.76E-01(4)	2.6012+01(7)	2.67E+01(5)	4.66E-01(4)	2.05E+01(8)	2.87E+00(4)	1.45E+01(4)
	std	2.88E-05	2.09E-02	2.62E+05	1.65E-01	8.73E-01	8.64E+00	2.73E-01	2.73E-01	2.56E-01	1.05E-01
AFEP	mean	3.36E-06(4)	1.92E-06(4)	8.29E+05(5)	2.83E-06(3)	8.28E-03(3)	7.64E+00(3)	3.88E-02(2)	2.03E+01(4)	9.95E-01(2)	1.07E+01(3)
	std	2.45E-07	3.93E-06	2.93E+05	3.93E-06	1.55E-03	2.72E+00	1.73E-02	1.63E-01	1.72E-01	1.77E-01
HEP	mean	**8.87E-13**(1)	**1.19E-12**(1)	**1.68E+03**(1)	**9.66E-13**(1)	**1.78E-04**(1)	**7.97E-01**(1)	1.27E+03(8)	**2.01E+01**(1)	3.98E+00(5.5)	1.01E+01(2)
	std	2.32E-13	6.50E-13	8.34E+02	4.45E-13	3.83E-05	1.59E+00	1.00E-01	3.39E-02	2.14E+00	1.71E+00
HEP without NM	mean	1.58E-12(2)	2.24E-12(2)	3.92E+03(2)	3.00E-12(2)	2.06E-04(2)	1.59E+00(2)	1.27E+03(6.5)	2.01E+01(2)	3.98E+00(5.5)	2.17E+01(7)
	std	3.66E-13	5.04E-13	2.81E+03	8.76E-13	3.31E-05	1.95E+00	6.56E-06	2.02E-02	1.41E+00	4.68E+00

Algorithms		F11	F12	F13	F14	F15	F16	F17	F18	F19	F20
CEP	mean	9.25E+00(6)	2.85E+01(2)	8.36E-01(6)	3.84E+00(7)	4.79E+02(8)	1.83E+02(7)	1.52E+02(8)	4.12E+02(5)	3.74E+02(4)	9.64E+02(8)
	std	2.64E+00	2.61E+01	2.66E-01	2.56E-01	2.7621+01	2.55E+01	3.07E+01	7.77E+01	2.01E+01	7.63E+01
FEP	mean	7.74E+00(5)	8.97E+01(3)	1.14E+00(7)	3.86E+00(8)	4.22E+02(5)	1.24E+02(3)	1.13E+02(4)	4.06E+02(4)	3.54E+02(3)	8.00E+02(4)
	std	2.84E+00	2.63E+00	2.82E-01	1.61E-01	2.61E+01	3.73E+01	1.06E+01	8.82E+01	3.71E+01	1.73E+00
ALEP	mean	2.17E+01(7)	4.73E+02(6)	6.89E-01(4)	3.77E+00(6)	4.75E+02(7)	2.04E+02(8)	1.47E+02(7)	4.00E+02(3)	3.44E+02(2)	8.01E+02(5)
	std	1.53E+01	1.73E+02	2.85E-20	1.53E-01	1.65E+00	8.54E+01	1.66E+01	2.7271-01	1.65E+10	3.57E+12
MEP	mean	3.55E+01(8)	3.85E+02(5)	**3.19E-01**(1)	3.72E+00(5)	**1.20E+02**(1)	1.71E+02(6)	1.43E+02(5)	8.79E+02(7)	8.83E+02(7)	8.87E+02(6)
	std	1.26E+01	5.86E+02	7.81E-02	3.68E-01	1.96E+02	2.97E+02	2.31E+01	2.17E+02	2.20E+02	1.82E+02
RLEP	mean	4.86E+00(4)	7.25E+02(7)	1.88E+00(8)	3.42E+00(3)	4.08E+02(4)	1.60E+02(5)	**1.12E+01**(1)	3.61E+02(2)	4.62E+02(5)	5.00E+02(2)
	std	2.1728E-01	3.62E+02	3.67E-01	1.44E-01	8.73E+01	2.79E+01	7.50E+00	1.54E+01	1.57E+01	1.09E+02
AFEP	mean	4.78E+00(3)	5.83E+03(8)	4.62E-01(2)	**2.51E+00**(1)	4.54E+02(6)	**1.08E+02**(1)	1.13E+01(2)	**3.53E+02**(1)	**3.10E+02**(1)	**3.72E+02**(1)
	std	2.73E+00	2.90E+03	2.82E-01	5.26E-01	2.70E+01	1.13E+01	8.66E+00	2.54E-01	7.67E+00	1.72E+01
HEP	mean	**3.92E+00**(1)	**1.88E+01**(1)	6.67E-01(3)	3.22E+00(2)	3.04E+02(2)	1.18E+02(2)	1.04E+02(3)	8.86E+02(8)	8.32E+02(6)	7.76E+02(3)
	std	7.01E-01	6.19E+01	1.56E-01	2.86E-01	1.66E+02	7.19E+00	3.44E+00	1.07E+02	6.49E+01	1.37E+02
HEP without NM	mean	4.36E+00(2)	1.47E+02(4)	7.18E-01(5)	3.43E+00(4)	3.74E+02(3)	1.25E+02(4)	1.46E+02(6)	8.29E+02(6)	9.20E+02(8)	9.11E+02(7)
	std	8.34E-01	2.83E+02	2.23E-01	4.79E-01	1.22E+02	2.00E+01	1.43E+01	5.77E+01	1.00E+02	9.27E+01

Algorithms		F21	F22	F23	F24	F25	Avg.rank	Friedman test	Post-hoc(p)
CEP	mean	5.46E+02(6)	1.09E+03(8)	5.77E+02(2)	**2.00E+02**(3)	9.44E+02(6)	5.56	$\chi^2 = 31.69$	5.946E-04
	std	3.63E+01	2.83E+01	1.63E+02	0	2.08E+02			
FEP	mean	**5.00E+02**(1)	**7.41E+02**(1)	5.87E+02(3)	**2.00E+02**(3)	3.47E+02(4)	4.64	p =4.64E-05	0.025
	std	2.78E-04	2.78E-01	2.72E-04	0	3.53E+02			
ALEP	mean	5.08E+02(2)	8.62E+02(6)	6.64E+02(5)	2.02E+02(6)	3.29E+02(3)	5.16		3.524E-03
	std	6.57E+01	2.77E+02	2.63E+02	1.58E-02	5.09E+01			
MEP	mean	9.24E+02(8)	8.52E+02(5)	8.94E+02(8)	2.90E+02(8)	**2.40E+02**(1.5)	5.48		8.659E-04
	std	3.23E+02	8.61E+01	3.07E+02	2.07E+02	1.04E+02			
RLEP	mean	5.36E+02(4)	8.78E+02(7)	6.67E+02(6)	2.02E+02(7)	3.73E+02(5)	4.92		9.092E-03
	std	2.53E+01	1.73E+02	2.71E+02	3.63E-02	2.72E+01			
AFEP	mean	5.34E+02(3)	7.88E+02(4)	5.87E+02(4)	**2.00E+02**(3)	**2.40E+02**(1.5)	2.98		0.707
	std	1.77E+01	4.09E+01	1.66E+01	0	4.26E+01			
HEP	mean	5.40E+02(5)	7.73E+02(3)	**5.59E+02**(1)	**2.00E+02**(3)	1.77E+03(7)	2.9		-
	std	1.10E+02	2.19E+01	3.07E+02	1.44E-10	3.70E+00			
HEP without NM	mean	6.33E+02(7)	7.62E+02(2)	8.62E+02(7)	**2.00E+02**(3)	1.79E+03(8)	4.36		0.059
	std	2.61E+02	8.01E+00	2.63E+02	1.79E-10	1.43E+01			

Table 6 Parameter setting in 13 Algorithms

General	Population size	100	Number of repetition	50
	Tournament size q	10	Initial standard deviation	3
FEP	Parameter t for Cauchy distribution function 1			
LEP	Value of a for Lévy distribution function	1.5		
CMA-ES	All parameters are similar to the source codes in [19]		Number of transposon	1
JG	Length of transposon	2	crossover	Uniform
	Mutation rate	0.1	Acceleration coefficients	2
IIW-PSO	Linearly increasing inertia weight	From 0.5 to 1.5	Maximum velocity	$\pm X_{max}$

In HEP, roughness is taken as the characteristic of local fitness landscapes. It is worth investigating other characteristics of fitness landscapes in the future work.

Acknowledgements We would like to acknowledge the support from the National Science Foundation of China (No. 61472095) and EPSRC under Grant No. EP/I009809/1.

References

1. Ali, M.M., Khompatraporn, C., Zabinsky, Z.B.: A numerical evaluation of several stochastic algorithms on selected continuous global optimization test problems. Journal of global optimization **31**(4), 635–672 (2005)
2. Alipouri, Y., Poshtan, J., Alipour, H.: Global minimum routing in evolutionary programming using fuzzy logic. Information Sciences **292**, 162–174 (2015)
3. Alipouri, Y., Poshtan, J., Alipouri, Y.: A modification to classical evolutionary programming by shifting strategy parameters. Applied intelligence **38**(2), 175–192 (2013)
4. Alipouri, Y., Poshtan, J., Alipouri, Y., Alipour, M.R.: Momentum coefficient for promoting accuracy and convergence speed of evolutionary programming. Applied Soft Computing **12**(6), 1765–1786 (2012)
5. Anik, M.T.A., Ahmed, S.: A mixed mutation approach for evolutionary programming based on guided selection strategy. In: Proceedings of 2013 International Conference on Informatics, Electronics & Vision (ICIEV), pp. 1–6. IEEE (2013)
6. Anik, M.T.A., Ahmed, S., Islam, K.R.: Self-adaptive mutation strategy for evolutionary programming based on fitness tracking scheme. In: Proceedings of 2013 IEEE

Table 7 Experimental Comparison of the Average Fitness among 14 Algorithms

Function	CEP	FEP	LEP	EEP	RLEP	MEP	RCEP	MCEP	CMAES	IIW-PSO	GA-JG	FREP	HEP	HEP without NM	Generation
F1	-8455(9)	**-12493**(1)	-12251(2)	-1.1026(6)	-11878.2(3)	-11868(4)	-9410(7)	-6267.56(13)	-2572(14)	-8917(8)	-7040(11)	-8384(10)	-11033.9664(5)	-6331.4583(12)	3000
F2	57.01(9)	6.76(5)	98.59(11)	389.6(14)	106.8(13)	9.45(7)	30.34(8)	5.87(4)	8.06(6)	73.77(10)	103(12)	5.57(3)	**2.579**(1)	3.5819(2)	3000
F3	6.36(13)	1.13(8)	0.26(7)	20.01(14)	1.842(9)	2.05(10)	0.006(6)	2.00E-07(5)	8.00E-15(2)	5(11)	5.14(12)	5.00E-12(4)	**0**(1)	1.421E-14(3)	3000
F4	8.16(14)	1.63(11)	0.44(8)	3.31(13)	0.928(9)	0.36(7)	0.0098(5)	0.03(6)	**0**(2)	1.03(10)	2.32(12)	**0**(2)	**0**(2)	7.40E-03(4)	3000
F5	15.65(11)	0.66(8)	0.002(5)	2.42(9)	0.149(7)	15.9(12)	3.80E-06(4)	0.06(6)	21.74(14)	18.52(13)	3.32(10)	1.54E-24(2)	1.08E-09(3)	**1.67E-32**(1)	3000
F6	62.6(13)	10.32(10)	0.16(7)	86.38(14)	0.02(6)	0.01(5)	5.24(9)	0.002(4)	1.2(8)	31.6(12)	13.3(11)	9.70E-26(2)	1.10E-08(3)	**1.35E-28**(1)	3000
F7	1(7.5)	1(7.5)	1(7.5)	1.246(13)	1(7.5)	1(7.5)	1(7.5)	1(7.5)	2.98(14)	1(7.5)	1(7.5)	1(7.5)	**0.999**(1.5)	**0.999**(1.5)	2000
F8	0.006(10.5)	0.01(12.5)	0.005(8)	0.003(3)	0.0056(9)	0.004(4.5)	0.0043(6.5)	0.0043(6.5)	0.01(12.5)	0.02(14)	0.006(10.5)	0.004(4.5)	**5.09E-04**(1)	7.05E-04(2)	400
F9	**-1.0316**(4)	**-1.0316**(4)	**-1.0316**(4)	**-1.0316**(4)	**-1.0316**(4)	-1.031(10)	-1.031(10)	-1.031(10)	-0.99(14)	-1.031(10)	-1.03(13)	-1.031(10)	**-1.0316**(4)	**-1.0316**(4)	100
F10	**0.397**(5.5)	**0.397**(5.5)	**0.397**(5.5)	**0.397**(5.5)	**0.397**(5.5)	0.398(13.5)	0.398(13.5)	**0.397**(5.5)	**0.397**(5.5)	**0.397**(5.5)	**0.397**(5.5)	**0.397**(5.5)	0.3979(11.5)	0.3979(11.5)	100
F11	**3**(7.5)	**3**(7.5)	**3**(7.5)	**3**(7.5)	**3**(7.5)	**3**(7.5)	**3**(7.5)	**3**(7.5)	**3**(7.5)	**3**(7.5)	**3**(7.5)	**3**(7.5)	**3**(7.5)	**3**(7.5)	100
F12	-3.86(12)	-3.86(12)	-3.86(12)	-3.86(12)	-3.86(12)	-3.862(6)	-3.862(6)	-3.862(6)	-3.862(6)	-3.862(6)	-3.862(6)	-3.862(6)	**-3.8628**(1.5)	**-3.8628**(1.5)	100
F13	-3.298(7.5)	-3.298(7.5)	-3.274(11)	-3.25(13)	-3.32(3.5)	-3.23(14)	-3.3(6)	-3.32(3.5)	-3.31(5)	-3.28(9.5)	-3.28(9.5)	-3.26(12)	**-3.321**(1.5)	**-3.321**(1.5)	500
F14	-10.15(4.5)	-10.15(4.5)	-10.15(4.5)	-7.17(12)	-10.15(4.5)	-7.53(10)	-9.04(7)	-7.16(13)	-5.74(14)	-8.13(9)	-7.31(11)	-8.2(8)	**-10.153**(1.5)	**-10.153**(1.5)	100
F15	-9.5(5)	-8.82(10)	-9.01(7)	-7.86(12)	-8.91(8)	-6.66(13)	-10.17(3.5)	-8.86(9)	-6.58(14)	-8.43(11)	-9.26(6)	-10.17(3.5)	**-10.403**(1.5)	**-10.403**(1.5)	100
F16	-9.15(9)	-8.71(10)	-9.9(5.5)	-8.55(11)	-9.6(7)	-8.49(12)	-10.25(4)	-9.9(5.5)	-5.73(14)	-7.41(13)	-9.59(8)	-10.52(3)	**-10.536**(1.5)	**-10.536**(1.5)	100
F17	2.76E-39(6)	1.09E-18(9)	0.003(14)	3.40E-34(7)	5.21E-05(13)	1.20E-08(11)	2.50E-23(8)	1.00E-50(5)	5.10E-60(3)	1.10E-12(10)	1.54E-08(12)	1.10E-82(2)	0(1)	1.88E-52(4)	3000
F18	7.15E-06(3)	1.15(14)	0.003(11)	0.0011(9)	0.001(8)	1.30E-03(10)	0.005(12)	1.00E-04(6)	**1.00E-126**(1)	3.80E-04(7)	0.06(13)	1.00E-07(2)	9.41E-05(4.5)	9.41E-05(4.5)	1000
F19	4.38E-15(8)	0.016(13)	5.00E-16(6)	4.10E-15(7)	2.54E-16(5)	2.70E-06(11)	1.86E-06(10)	1.24E-17(4)	3.20E-14(9)	0.1(14)	0.007(12)	6.90E-19(3)	**2.83E-36**(1)	5.15E-35(2)	1000
F20	-7.191(12)	-8.333(9)	-8.257(10)	-8.341(3.5)	-8.341(3.5)	-8.34(6.5)	-8.34(6.5)	-8.34(6.5)	-5.16(14)	-6.9(13)	-8.15(11)	-8.34(6.5)	**-8.3762**(1.5)	**-8.3762**(1.5)	1000
F21	6.53E-12(7)	14.67(12)	3.00E-13(6)	9.61E-06(9)	3.79E-06(8)	7.8(11)	1.30E-02(10)	4.00E-25(5)	1.00E-121(2)	33.71(14)	26.84(13)	1.10E-43(4)	**0**(1)	2.34E-49(3)	1000
F22	270736(14)	5129(12)	18935(10)	127880(13)	5951(8)	3536(7)	4929.95(2)	-127(5)	**-4930**(1)	15485(9)	21654(11)	-1053(4)	-1590.3907(3)	47.498(6)	3000
F23	0.195(6)	0.98(13)	0.19(5)	0.599(12)	0.199(7)	1.29(14)	0.29(9)	0.43(10)	0.14(4)	0.48(11)	0.22(8)	0.1(3)	**0.0799**(1)	0.09987(2)	1000
F24	-5.288(5.5)	-5.286(7)	-5.288(5.5)	-5.11(12)	-4.07(14)	-5.28(9.5)	-6.03(4)	-5.28(9.5)	-5.28(9.5)	-5.28(9.5)	-4.73(13)	-8.35(3)	**-9.104**(1.5)	**-9.104**(1.5)	1000
F25	-0.03(9)	-3.20E-13(14)	-0.06(7)	-2.30E-07(13)	-0.085(6)	-3.00E-07(12)	-0.64(2)	-0.4(4)	**-0.74**(1)	-0.001(10)	-2.50E-04(11)	-0.63(3)	-0.0382(8)	-0.393(5)	2000
F26	1(4.5)	7.07(10)	1(4.5)	4.093(9)	1(4.5)	844(12)	1(4.5)	8.5(11)	1(4.5)	5.00E+11(14)	3.40E+10(13)	1(4.5)	1(4.5)	1(4.5)	1000
F27	5.40E-13(7)	0.032(13)	5.00E-12(9)	5.50E-17(5)	2.96E-07(10)	1.00E-12(8)	2.30E-14(6)	0.004(12)	**0**(2.5)	2.00E-05(11)	0.08(14)	**0**(2.5)	**0**(2.5)	**0**(2.5)	100
F28	2.03E-15(7)	2.70E-06(12)	2.00E-13(9)	8.20E-16(6)	8.72E-09(10)	1.30E-14(8)	2.30E-16(5)	3.00E-18(4)	0.1(14)	3.00E-08(11)	5.40E-05(13)	1.10E-26(3)	**4.48E-31**(1)	1.64E-29(2)	100
F29	**-1.00E+06**(1.5)	-99620(9)	-9998(8)	-92418(10)	**-1.00E+06**(1.5)	-83262(12)	-63548(14)	-99999(4.5)	-65299(13)	-99999(4.5)	-84095(11)	-99999(4.5)	-99998.9452(7)	-99999(4.5)	1000
F30	**-1**(7.5)	**-1**(7.5)	**-1**(7.5)	**-1**(7.5)	**-1**(7.5)	**-1**(7.5)	**-1**(7.5)	**-1**(7.5)	**-1**(7.5)	**-1**(7.5)	**-1**(7.5)	**-1**(7.5)	**-1**(7.5)	**-1**(7.5)	100
F31	-7.287(7)	-7.649(5)	-5.921(8)	-7.38(6)	-7.724(3)	-4.14(13)	-4.29(12)	-3.5(14)	-4.4(11)	-5.77(9)	-5.47(10)	-7.67(4)	**-9.01**(1)	-8.90(2)	1000
F32	-0.98(7.5)	-0.974(9)	-0.17(14)	-0.97(10.5)	-0.55(13)	-0.823(12)	**-1**(3.5)	**-1**(3.5)	**-1**(3.5)	-0.98(7.5)	-0.97(10.5)	**-1**(3.5)	**-1**(3.5)	**-1**(3.5)	1000
F33	0.0195(14)	0.0021(11)	0.002(8)	**0.0019**(3)	**0.0019**(3)	0.0041(12)	**0.0019**(3)	0.002(8)	0.006(13)	0.002(8)	0.002(8)	**0.0019**(3)	**0.0019**(3)	**0.0019**(3)	300
F34	**1.23E-32**(3.5)	2.56E-30(7)	**1.23E-32**(3.5)	**1.23E-32**(3.5)	**1.23E-32**(3.5)	1.20E-24(9)	0.001(14)	9.00E-30(8)	**1.23E-32**(3.5)	4.00E-24(10)	1.76E-06(13)	**1.23E-32**(3.5)	2.97E-08(12)	8.66E-11(11)	100
F35	-1.288(7)	-1.279(12)	-1.282(11)	-1.286(9)	-1.295(5)	-1.284(10)	-1.287(8)	-1.296(3)	-1.22(14)	-1.296(3)	-1.261(13)	-1.296(3)	**-1.29625**(1)	-1.289332(6)	200
F36	-45.77(7)	-45.55(12)	-45.77(7)	-45.77(7)	-45.77(7)	-45.77(7)	-42.31(14)	-45.77(7)	-45.77(7)	-45.77(7)	-45.31(13)	-45.77(7)	**-45.7785**(1.5)	**-45.7785**(1.5)	1000
F37	0.181(12)	0.11(11)	1.40E-17(5)	1.40E-22(4)	2.49E-09(6)	0.101(10)	6.60E-04(8)	0.001(9)	3.3(14)	1.10E-05(7)	0.2(13)	2.10E-34(2)	**0**(1)	2.44E-024(3)	500
F38	-186(10)	-186(10)	-186(10)	-186(10)	-186(10)	-185.4(13.5)	-186.7(4)	-186.7(4)	-185.4(13.5)	-186.6(6)	-186.5(7)	-186.7(4)	**-186.7309**(1.5)	**-186.7309**(1.5)	100
F39	**-3.5**(5.5)	**-3.5**(5.5)	**-3.5**(5.5)	**-3.5**(5.5)	**-3.5**(5.5)	**-3.5**(5.5)	-0.3(14)	-2.77(12)	**-3.5**(5.5)	-2.951(11)	-2.42(13)	**-3.5**(5.5)	**-3.5**(5.5)	**-3.5**(5.5)	5000
Avg.rank	8.038	9.256	7.6154	8.833	7.141	9.603	7.474	7.038	8.295	9.513	10.654	4.705	3.154	3.679	
Friedman test	$\chi^2 = 146.76$	p = 9.23E-25													
Post-hoc(p)	2.520E-07	1.180E-10	2.482E-06	2.032E-09	2.567E-05	9.948E-12	5.097E-06	4.121E-05	5.736E-08	1.913E-11	2.443E-15	0.099	-	0.495	

Table 8 Computational result of benchmark functions in CEC2017 with 10 variables

Function		1	2	3	4	5	6	7	8	9	10	11	12
IPOP-CMA-ES	Mean	2.50E+08	8.16E+09	9.05E+03	1.44E-08	6.46E+01	5.08E+01	3.14E+02	5.97E+01	1.30E+03	1.37E+03	7.64E+01	1.13E+07
	Std	5.68E+08	4.38E+10	2.11E+04	5.571E-09	2.38E+01	1.07E+01	7.23E+01	2.00E+01	5.77E+02	3.68E+02	5.84E+01	2.13E+07
	Rank	7	7	7	5	6	7	6	7	7	7	7	7
TLBO-FL	Mean	2.00E+03	2.00E-02	1.10E-04	3.00E+00	**8.80E+00**	8.40E-08	2.80E+01	**1.20E+01**	**8.90E-03**	9.50E+02	**4.10E+00**	6.60E+04
	Std	2.50E+03	1.40E-01	7.80E-04	1.20E+00	5.60E+00	4.40E-07	4.00E+00	4.40E+00	6.40E-02	2.10E+02	1.50E+00	5.50E+04
	Rank	3	6	6	7	7	1	3	1	1	6	1	6
CEP	Mean	2.97E+03	**0.00E+00**	2.73E-10	1.36E-09	3.28E+01	3.52E+00	3.72E+01	2.61E+01	1.84E+02	7.61E+02	1.52E+01	1.04E+04
	Std	2.96E+03	0.00E+00	3.49E-11	1.55E-09	9.16E+00	2.13E+00	1.33E+01	1.36E+01	2.32E+02	2.26E+02	1.19E+01	1.36E+04
	Rank	5	3	2	2	5	6	7	6	6	5	5	3
FEP	Mean	2.12E+03	**0.00E+00**	7.60E-10	1.13E-08	2.05E+01	1.40E-05	2.74E+01	1.87E+01	2.73E+00	5.98E+02	1.34E+01	1.94E+04
	Std	2.17E+03	0.00E+00	2.11E-10	2.64E-09	5.75E+00	7.99E-07	7.22E+00	5.61E+00	3.55E+00	3.30E+02	6.72E+00	9.32E+03
	Rank	4	3	5	6	2	4	2	3	5	4	4	5
ALEP	Mean	3.29E+03	**0.00E+00**	2.86E-10	**1.18E-10**	2.31E+01	9.24E-06	2.92E+01	2.05E+01	1.09E-01	5.80E+02	7.90E+00	9.73E+03
	Std	2.47E+03	0.00E+00	2.28E-11	2.22E-11	4.64E+00	1.16E-06	7.10E+00	4.43E+00	1.76E-01	9.45E+01	9.39E+00	1.03E+04
	Rank	6	3	3	1	3	3	4	4	2	3	2	2
HEP	Mean	**1.23E+03**	**0.00E+00**	**2.32E-10**	4.52E-09	**1.55E+01**	8.93E-06	**2.72E+01**	1.83E+01	1.01E+01	**5.48E+02**	1.29E+01	**4.87E+03**
	Std	1.79E+03	0.00E+00	6.55E-11	8.24E-10	1.19E+01	3.37E-06	1.21E+01	6.68E+00	1.26E+01	6.05E+01	4.73E+00	6.43E+03
	Rank	1	3	1	3	1	2	1	2	3	1	3	1
HEP without NM	Mean	1.78E+03	**0.00E+00**	3.77E-10	5.51E-09	2.59E+01	1.20E+00	3.12E+01	2.07E+01	2.70E+01	5.55E+02	2.26E+01	1.56E+04
	Std	9.10E+02	0.00E+00	6.52E-11	9.92E-10	1.01E+01	1.87E+00	4.36E+00	6.27E+00	2.62E+01	1.45E+02	1.08E+01	1.23E+04
	Rank	2	3	4	4	4	5	5	5	4	2	6	4

Function		13	14	15	16	17	18	19	20	21	22	23	24
IPOP-CMA-ES	Mean	2.35E+04	4.64E+03	4.12E+04	5.03E+02	2.49E+02	1.26E+06	1.33E+05	2.35E+02	2.33E+02	1.21E+02	4.20E+02	4.22E+02
	Std	2.65E+04	5.51E+03	7.38E+04	1.82E+02	1.47E+02	4.12E+06	7.66E+05	1.00E+02	6.02E+01	1.48E+02	5.50E+01	1.14E+02
	Rank	7	7	7	7	7	7	7	7	7	7	7	7
TLBO-FL	Mean	2.40E+03	6.70E+01	1.30E+02	**8.90E+00**	3.80E+01	6.20E+03	6.10E+01	**1.50E+01**	1.40E+02	9.30E+01	**3.10E+02**	**3.10E+02**
	Std	2.20E+03	1.80E+01	4.30E+01	2.20E+01	7.80E+00	5.60E+03	3.20E+01	9.40E+00	5.20E+01	2.30E+01	3.80E+00	6.90E+01
	Rank	6	3	6	1	5	6	3	1	4	2	1	1
CEP	Mean	1.87E+03	4.83E+02	5.89E+01	1.63E+02	6.57E+01	2.15E+03	1.19E+02	7.16E+01	1.26E+02	1.06E+02	3.46E+02	3.62E+02
	Std	8.65E+02	2.93E+02	4.64E+01	1.01E+02	4.90E+01	3.59E+03	1.79E+02	5.04E+01	5.02E+01	7.19E+00	2.26E+01	7.57E+00
	Rank	5	6	5	6	6	3	4	6	3	6	6	5
FEP	Mean	3.87E+01	5.30E+01	3.63E+01	1.01E+02	2.16E+01	3.40E+03	5.82E+02	2.13E+01	1.45E+02	1.03E+02	3.23E+02	3.18E+02
	Std	3.27E+01	4.15E+01	3.74E+01	9.21E+01	1.10E+01	3.36E+03	5.52E+02	2.29E+00	5.54E+01	1.38E+00	6.86E+00	5.81E+01
	Rank	2	2	4	4	2	5	5	2	5	4	4	2
ALEP	Mean	**1.23E+01**	7.48E+01	**6.37E+00**	9.55E+01	2.93E+01	**4.99E+01**	**4.32E+00**	2.26E+01	1.25E+02	**7.31E+01**	3.29E+02	4.00E+02
	Std	4.06E+00	7.14E+01	3.66E+00	4.74E+01	7.72E+00	3.03E+01	3.06E+00	1.12E+01	1.26E+00	3.56E+01	8.27E+00	1.22E+02
	Rank	1	4	1	3	4	1	1	3	2	1	5	6
HEP	Mean	3.84E+02	**3.93E+01**	1.07E+01	1.24E+02	**2.07E+01**	8.01E+02	7.11E+02	3.18E+01	**1.24E+02**	1.02E+02	3.18E+02	3.41E+02
	Std	3.03E+02	1.63E+01	7.13E+00	7.59E+01	1.60E+01	7.39E+02	7.90E+02	8.50E+00	6.12E+01	1.30E+00	8.85E+00	1.10E+01
	Rank	4	1	2	5	1	2	6	4	1	3	2	3
HEP without NM	Mean	1.44E+02	2.09E+02	2.27E+01	8.68E+01	2.26E+01	2.91E+03	5.48E+01	3.73E+01	2.01E+02	1.04E+02	3.19E+02	3.53E+02
	Std	7.40E+01	2.21E+02	1.61E+01	7.20E+01	2.53E+01	2.98E+03	5.36E+01	2.00E+01	5.06E+01	2.51E+00	3.42E+00	1.25E+02
	Rank	3	5	3	2	3	4	2	5	6	5	3	4

Function		25	26	27	28	29	30	R	R2	Z	P	KA^2	p
IPOP-CMA-ES	Mean	4.21E+02	6.14E+02	4.49E+02	4.77E+02	5.45E+02	1.61E+07						
	Std	5.10E+01	5.74E+02	5.00E+01	1.57E+02	1.81E+02	2.27E+07						
	Rank	3	7	7	7	7	7	6.733	45.338	7.709	1.266E-14	80.486	2.836E-15
TLBO-FL	Mean	4.30E+02	3.00E+02	**3.90E+02**	4.50E+02	2.70E+02	2.80E+05						
	Std	2.20E+01	4.60E+01	3.30E+00	1.60E+02	1.40E+01	4.90E+05						
	Rank	6	2	1	6	2	6	3.667	13.444	2.211	0.027		
CEP	Mean	4.39E+02	3.29E+02	4.04E+02	**3.00E+02**	3.34E+02	2.07E+03						
	Std	1.95E+01	1.81E+02	3.15E+00	3.37E-06	5.41E+01	1.56E+03						
	Rank	7	6	6	1	6	5	4.9	24.01	4.422	9.763E-06		
FEP	Mean	4.28E+02	**2.49E+02**	3.99E+02	3.00E+02	2.79E+02	1.15E+03						
	Std	2.48E+01	1.26E+02	3.26E+00	5.72E-06	2.98E+01	4.31E+02						
	Rank	5	1	5	3	4	3	3.633	13.201	2.151	0.031		
ALEP	Mean	4.17E+02	3.00E+02	3.97E+02	3.23E+02	**2.48E+02**	1.12E+03						
	Std	2.27E+01	2.47E-06	2.76E+00	4.63E+01	9.75E+00	1.93E+02						
	Rank	2	3	3	4	1	2	2.767	7.654	0.598	0.474		
HEP	Mean	**4.09E+02**	3.43E+02	3.96E+02	4.19E+02	2.77E+02	**1.05E+03**						
	Std	2.10E+01	8.59E+01	4.13E+00	1.46E+02	1.86E+01	4.30E+02						
	Rank	1	5	2	5	3	1	2.433	5.921	-	0.75		
HEP without NM	Mean	4.27E+02	3.12E+02	3.99E+02	3.00E+02	2.81E+02	1.29E+03						
	Std	2.35E+01	2.37E+01	1.33E+00	7.06E-06	2.29E+01	7.31E+02						
	Rank	4	4	4	2	5	4	3.867	14.95	2.570	0.010		

Congress on Evolutionary Computation (CEC'13), pp. 2221–2228. IEEE (2013)

7. Auger, A., Hansen, N.: A restart cma evolution strategy with increasing population size. In: Evolutionary Computation, 2005. The 2005 IEEE Congress on, vol. 2, pp. 1769–1776. IEEE (2005)
8. Awad, N., Ali, M., Liang, J., Qu, B., Suganthan, P.: Problem definitions and evaluation criteria for the cec 2017 special session and competition on single objective bound constrained real-parameter numerical optimization. In: Technical Report. NTU, Singapore (2016)
9. Chellapilla, K.: Combining mutation operators in evolutionary programming. IEEE transactions on Evolutionary Computation **2**(3), 91–96 (1998)
10. Chelouah, R., Siarry, P.: Genetic and nelder–mead algorithms hybridized for a more accurate global optimization of continuous multiminima functions. European Journal of Operational Research **148**(2), 335–348 (2003)
11. De Oca, M.A.M., Stutzle, T., Birattari, M., Dorigo, M.: Frankenstein's pso: a composite particle swarm optimization algorithm. IEEE Transactions on Evolutionary Computation **13**(5), 1120–1132 (2009)
12. Derrac, J., García, S., Molina, D., Herrera, F.: A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm and Evolutionary Computation **1**(1), 3–18 (2011)
13. Dong, H., He, J., Huang, H., Hou, W.: Evolutionary programming using a mixed mutation strategy. Information Sciences **177**(1), 312–327 (2007)
14. Durand, N., Alliot, J.M.: A combined nelder-mead simplex and genetic algorithm. In: Proceedings of the Genetic and Evolutionary Computation Conference GECCO'99, pp. 1–7 (1999)
15. Fajfar, I., Puhan, J., Bűrmen, Á.: Evolving a nelder–mead algorithm for optimization with genetic programming. Evolutionary computation (2016)
16. Fogel, D.B., Fogel, G.B., Ohkura, K.: Multiple-vector self-adaptation in evolutionary algorithms. BioSystems **61**(2), 155–162 (2001)

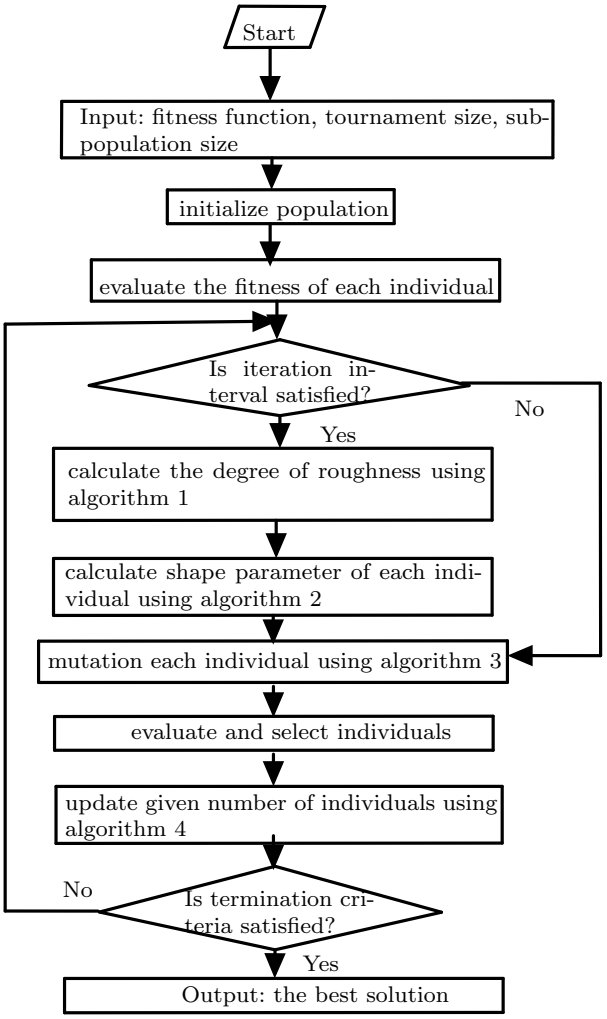

Fig. 3 Flow chart of HEP.

17. Fogel, L., Owens, A., Walsh, M.: Artificial Intelligence Through Simulated Evolution. John Wiley & Sons (1966)
18. García, S., Molina, D., Lozano, M., Herrera, F.: A study on the use of non-parametric tests for analyzing the evolutionary algorithms behaviour: a case study on the cec2005 special session on real parameter optimization. Journal of Heuristics **15**(6), 617–644 (2009)
19. Hansen, N.: The cma evolution strategy: A tutorial. arXiv preprint arXiv:1604.00772 (2016)
20. He, J., Yao, X.: A game-theoretic approach for designing mixed mutation strategies. Advances in Natural Computation pp. 435–435 (2005)
21. Hong, L., Woodward, J., Li, J., Özcan, E.: Automated design of probability distributions as mutation operators for evolutionary programming using genetic programming. In: European Conference on Genetic Programming, pp. 85–96. Springer (2013)
22. Iwamatsu, M.: Generalized evolutionary programming with lévy-type mutation. Computer Physics Communications **147**(1-2), 729–732 (2002)
23. Kommadath, R., Kotecha, P.: Teaching learning based optimization with focused learning and its performance on cec2017 functions. In: Evolutionary Computation (CEC), 2017 IEEE Congress on, pp. 2397–2403. IEEE (2017)
24. Lee, C.Y., Yao, X.: Evolutionary programming using mutations based on the lévy probability distribution. IEEE Transactions on Evolutionary Computation **8**(1), 1–13 (2004)
25. Liao, S.H., Hsieh, J.G., Chang, J.Y., Lin, C.T.: Training neural networks via simplified hybrid algorithm mixing nelder–mead and particle swarm optimization methods. Soft Computing **19**(3), 679–689 (2015)
26. Mallipeddi, R., Mallipeddi, S., Suganthan, P.N.: Ensemble strategies with adaptive evolutionary programming. Information Sciences **180**(9), 1571–1581 (2010)
27. Mallipeddi, R., Suganthan, P.N.: Evaluation of novel adaptive evolutionary programming on four constraint handling techniques. In: Evolutionary Computation, 2008. CEC 2008.(IEEE World Congress on Computational Intelligence). IEEE Congress on, pp. 4045–4052. IEEE (2008)
28. Mesbahi, T., Khenfri, F., Rizoug, N., Bartholomeüs, P., Le Moigne, P.: Combined optimal sizing and control of li-ion battery/supercapacitor embedded power supply using hybrid particle swarm–nelder–mead algorithm. IEEE Transactions on Sustainable Energy **8**(1), 59–73 (2017)
29. Narihisa, H., Kohmoto, K., Taniguchi, T., Ohta, M., Katayama, K.: Evolutionary programming with only using exponential mutation. In: Proceedings of 2006 IEEE Congress on Evolutionary Computation, pp. 552–559. IEEE (2006)
30. Nelder, J.A., Mead, R.: A simplex method for function minimization. The computer journal **7**(4), 308–313 (1965)
31. Neri, F., Cotta, C., Moscato, P.: Handbook of memetic algorithms. Springer (2012)
32. Pang, J., Dong, H., He, J., Feng, Q.: Mixed mutation strategy evolutionary programming based on shapley value. In: Evolutionary Computation (CEC), 2016 IEEE Congress on, pp. 2805–2812. IEEE (2016)
33. Price, K., Storn, R.M., Lampinen, J.A.: Differential evolution: a practical approach to global optimization. Springer Science & Business Media (2006)
34. Rahnamayan, S., Tizhoosh, H.R., Salama, M.M.: Opposition-based differential evolution. IEEE Transactions on Evolutionary computation **12**(1), 64–79 (2008)
35. Reeves, C.R.: Fitness landscapes. In: Search methodologies, pp. 681–705. Springer (2014)
36. Ripon, K.S.N., Kwong, S., Man, K.F.: A real-coding jumping gene genetic algorithm (rjgga) for multiobjective optimization. Information Sciences **177**(2), 632–654 (2007)
37. Shen, L., He, J.: A mixed strategy for evolutionary programming based on local fitness landscape. In: Proceedings of 2010 IEEE Congress on Evolutionary Computation (CEC'10), pp. 1–8. IEEE (2010)
38. Suganthan, P.N., Hansen, N., Liang, J.J., Deb, K., Chen, Y.P., Auger, A., Tiwari, S.: Problem definitions and evaluation criteria for the cec 2005 special session on real-parameter optimization. KanGAL report **2005005**, 2005 (2005)
39. Yao, X., Liu, Y., Lin, G.: Evolutionary programming made faster. IEEE Transactions on Evolutionary computation **3**(2), 82–102 (1999)
40. Zhang, H., Lu, J.: Adaptive evolutionary programming based on reinforcement learning. Information Sciences **178**(4), 971–984 (2008)

Compliance with Ethical Standards: The authors declare that they have no conflict of interest. This article does not contain any studies with human participants or animals performed by any of the authors. Informed consent was obtained from all individual participants included in the study.