Journal of Computational and Applied Mathematics 268 (2014) 121–134

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

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

Finite Markov chain analysis of classical differential evolution algorithm

ZhongBo Hu [a,b,*], ShengWu Xiong [a], QingHua Su [a,b], ZhiXiang Fang [c]

[a] School of Computer Science and Technology, Wuhan University of Technology, Wuhan 430070, PR China

[b] School of Mathematics and Statistics, Hubei Engineering University, Xiaogan 432000, Hubei, PR China

[c] State Key Laboratory for Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430072, PR China

ARTICLE INFO

Article history:
Received 7 April 2012
Received in revised form 7 December 2012

Keywords:
Convergence in probability
Continuous optimization
Differential evolution algorithm
Markov chain

ABSTRACT

Theoretical analyses of algorithms are important to understand their search behaviors and develop more efficient algorithms. Compared with the plethora of works concerning the empirical study of the differential evolution (DE), little theoretical research has been done to investigate the convergence properties of DE so far. This paper focuses on theoretical researches on the convergence of DE and presents a convergent DE algorithm. First of all, it is proved that the classical DE cannot converge to the global optimal set with probability 1 by using the property that it cannot escape from a local optimal set. Inspired by the characteristics of the elitist genetic algorithm, this paper proposed a modified DE to overcome the disadvantage. The proposed algorithm employs two operators that assist it in escaping from a local optimal set and enhance the diversity of the population. And it is then verified that the proposed algorithm is capable of converging to global optima with probability 1. The theoretical research of this paper is undertaken in a finite discrete set, and the analysis tool used is the Markov chain. The numerical experiments are conducted on a deceptive function and a set of benchmark functions. The experimental results support the theoretical analyses on the convergence performances of the classical and modified DE algorithm.

1. Introduction

Theoretical analyses of the convergence properties of differential evolution (DE) benefit understanding their search behaviors and developing more efficient algorithms. However, few studies focused on theoretical analyses of guiding to modify the DE. Inspired by Markov chain performance in the theoretical analyses of genetic algorithm (GA), this paper gives the convergence analyses of the classical DE, as well as present a modified DE, which converges to the global optimum with probability 1.

1.1. Background

DE, proposed by Storn and Price in 1995 [1], is a population-based stochastic parallel evolutionary algorithm. The first book on DE was published in 2005 [2]. Das and Suganthan [3] surveyed DE in detail in 2011, which includes basic concepts,

* Corresponding author at: School of Computer Science and Technology, Wuhan University of Technology, Wuhan 430070, PR China. Tel.: +86 13971978456.

E-mail addresses: huzbdd@126.com, huzbdd@yahoo.com.cn (Z. Hu).

http://dx.doi.org/10.1016/j.cam.2014.02.034

major variants, application and theoretical studies. Das et al. [4] identified the four main directions of DE research, which include the basic DE research, the application-general research, the application-specific research and the computational environment specific research.

DE [1–7] emerged as a powerful stochastic real-parameter optimization algorithm. And DE takes few control parameters, which makes it easy to implement. Perhaps these advantages triggered the popularity of DE among researchers within a short time. In recent years, DE has also got many real-world applications [8–10] and a lot of improved DE algorithms have been proposed. Ref. [11] subdivided these modified versions of DE into two classes:

- DE integrating an extra component. This class includes those algorithms [12–16] which use DE as an evolutionary framework which is assisted by additional algorithmic components.
- DE with modified structures. This class includes those algorithms [17–23] which make a substantial modification within the DE structure, in the search logic or the selection etc.

No matter DE integrating an extra component or DE with modified structures, the motivations of these improved DE algorithms are either based on a certain complementarity of the biological mechanism or based on the exploiting and exploring capability of the classical DE. Few motivations improving DE algorithms are based on theoretical analyses on the convergence properties of the classical DE. Perhaps the main reason is that little theoretical research has been presented on investigating the convergence properties of DE so far.

In 2005, Xu et al. [24] performed the mathematical modeling and convergence analysis of continuous multi-objective differential evolution (MODE) under certain simplifying assumptions. The authors assumed that the DE-population is initialized by sampling from a Gaussian distribution with given mean and standard deviation. They then proved that the initial population is Gaussian distributed and contains the Pareto optimal set, and the subsequent populations generated by the MODE without the selection operator are also Gaussian distributed and the population mean converges to the center of the Pareto optimal set. This work was extended in [25].

In 2006, Ter Braak C.J.F. [26] proposed the Differential Evolution Markov Chain algorithm (DE-MC). Its reproduction operator was defined by adding a uniformly random number to the mutation operator of the classical DE. DE-MC abandons crossover operator and uses Metropolis selection operator instead of the greedy one used in the classical DE. This paper also proved DE-MC yields a unique joint stationary distribution. However, it has not been theoretically proven whether DE-MC holds certain asymptotic convergence. In fact, it does not seem difficult to prove that DE-MC holds with convergence in distribution. However, only if its selection operator meets certain conditions, DE-MC can converge to the global optimum with probability 1. For example, the selection pressure [27,28] of Metropolis selection operator satisfies certain conditions and the probability of selecting bad individuals approaches 0 when iteration times tends to infinity. DE-MC's selection operator does not seem to meet the condition.

Whether the classical DE holds certain asymptotic convergence or not, these papers do not analyzed. The area on theoretical analyses, especially for guiding to modify the DE, remains largely open. Some improved DE algorithms, which are based on theoretical analyses on the convergence properties, will need to be further research.

1.2. Motivation and contribution

Like GA, DE uses the similar computational steps (mutation, crossover, and selection operators) at each generation to move its population toward the global optimum. DE and GA are both typical evolutionary algorithm. Through analyzing the reproduction operators of two algorithms, we can get that there are several differences between GA and DE as follows.

(1) DE is usually used to dealing with the continuous optimization problem. The search space of the continuous optimization problem is a continuous closed region (or a union of countable continuous closed regions). Therefore, the solution space of the classical DE with real code is generally continuous, which is opposite to the canonical GA with binary code. In contrast with a discrete solution space, the numbers of individuals in a continuous space is infinite.
(2) Using the mutation operator of the canonical GA, the GA's mutation probability from an individual to another which is an arbitrary point in a search space is greater than 0. However, the DE's mutation probability from an individual to another may equal to 0. In fact, the DE's mutation probability from an individual to another equals to 0 if the distance of the two individuals is enough great.
(3) Compared with the canonical GA which eliminates the best solution with a little-probability, the selection operator of the classical DE always maintains the best solution(s) in the population.

Inspired by Markov chain performance in the theoretical analyses of GA [29–31], this paper will employ the Markov chain as main tool. According to 1st difference, we must discretize the search space. Considering the limitations of the calculation accuracy in computer, this paper maps the continuous search space to a finite discrete set. A Markov chain model is then developed to investigate the classical DE. The theoretical analyses show that the classical DE cannot escape from a local optimal set in case of trapping in. This makes the classical DE not converge in probability to the global optimum of a continue optimization problem.

According to the 2nd difference, the classical DE's populations are not always accessible to each other in population space. And DE, like the elitist GA, always remains the best solution(s) in the current population according to the 3rd difference. Taking these two points above into account, this paper presents two operators, called *uniform mutation* and *diversity selection* operator. The uniform mutation operator generates uniformly distributed solutions with a little probability, while the diversity selection operator accepts poor solutions with a little probability. A modified DE with uniform mutation and diversity selection, called msDE, is then proposed. These two operators make the population space of the msDE have only one positive recurrent, irreducible, aperiodic and closed set, which assists the msDE in escaping from a local optima and premature solutions set. By developing a Markov chain model of the msDE, this paper proves that the msDE can converge in probability to the global optimum. Finally, an empirical comparison between the classical DE and the msDE is presented. The experimental results on the test functions further verify the theoretical conclusions, as well as indicate that the msDE has better robustness.

The rest of the paper is organized as follows: Section 2 introduces the classical DE. Section 3 analyze convergence property of the classical DE. Section 4 proposes two reproduction operators and the proposed msDE. In Section 5, the convergence in probability of the msDE are proved. Experimental design and comparisons are presented in Section 6. Finally, the conclusions, concluding remarks and future works are given in Section 7.

2. Classical differential evolution

DE is used for dealing with the continuous optimization problem. We suppose in this paper that the objective function to be minimized is $f(\vec{x})$, $\vec{x} = (x_1, \ldots, x_n) \in \Re^n$, and the feasible solution space is $\Psi = \prod_{j=1}^{j=n}[L_j, U_j]$. The classical DE [1–6, 18,32] works through a simple cycle of operators including mutation, crossover and selection operator after initialization. The classical DE procedures are described in detail as follows.

2.1. Initialization

The first step of DE is the initialization of a population of m, n-dimensional potential solutions (*individuals*) over the optimization search space. We shall symbolize each individual by $\vec{x}_i^g = (x_{i,1}^g, x_{i,2}^g, \ldots, x_{i,n}^g)$, for $i = 1, \ldots, m$, where $g = 0, 1, \ldots, g_{max}$ is the current generation and g_{max} is the maximum number of generations. At the first generation ($g = 0$) the population should be sufficiently scaled to cover as much as possible of the optimization search space. Initialization is implemented by using a random number distribution to generate the potential individuals in the optimization search space. We can initialize the jth dimension of the ith individual according to

$$x_{i,j}^0 = L_j + rand(0, 1) \cdot (U_j - L_j).$$

Unless otherwise mentioned, $rand(0, 1)$ is a uniformly distributed random number confined in the $[0, 1]$ range.

2.2. Mutation operators

After initialization, DE creates a donor vector $\vec{v}_i^g$ corresponding to each individual $\vec{x}_i^g$ in the gth generation through the mutation operator. Several most frequently referred mutation strategies [32] are presented as follows:

DE/rand/1:

$$\vec{v}_i^g = \vec{x}_{r_1}^g + F(\vec{x}_{r_2}^g - \vec{x}_{r_3}^g); \tag{1}$$

DE/best/1:

$$\vec{v}_i^g = \vec{x}_{best}^g + F(\vec{x}_{r_1}^g - \vec{x}_{r_2}^g); \tag{2}$$

DE/current-to-best/1:

$$\vec{v}_i^g = \vec{x}_i^g + F(\vec{x}_{best}^g - \vec{x}_i^g) + F(\vec{x}_{r_1}^g - \vec{x}_{r_2}^g); \tag{3}$$

DE/best/2:

$$\vec{v}_i^g = \vec{x}_{best}^g + F(\vec{x}_{r_1}^g - \vec{x}_{r_2}^g) + F(\vec{x}_{r_3}^g - \vec{x}_{r_4}^g); \tag{4}$$

DE/rand/2:

$$\vec{v}_i^g = \vec{x}_{r_1}^g + F(\vec{x}_{r_2}^g - \vec{x}_{r_3}^g) + F(\vec{x}_{r_4}^g - \vec{x}_{r_5}^g); \tag{5}$$

where $\vec{x}_{best}^g$ denotes the best individual of the current generation, the indices $r_1, r_2, r_3, r_4, r_5 \in S_r = \{1, 2, \ldots, m\}\backslash\{i\}$ are uniformly random integers mutually different and distinct from the running index i, and $F \in (0, 1]$ is a real parameter, called *mutation factor* or *scaling factor*.

If the element values of the donor vector, $\vec{v}_i$, exceed the pre-specified upper bound or lower bound, we can change the element values by the *periodic mode* rule as follow:

$$v_{i,j} = \begin{cases} U_j - (L_j - v_{i,j})\%|U_j - L_j| & \text{if } v_{i,j} < L_j \\ L_j + (v_{i,j} - U_j)\%|U_j - L_j| & \text{if } v_{i,j} > U_j. \end{cases}$$

```
X = initial._population (m), F, Cr = initial._parameters
while ! termination_condition do
  for g = 0 to m
    v_i^g = x_{r1}^g + F(x_{r2}^g − x_{r3}^g)          //mutation operator
    u_i^g = binomial_crossover(x_i^g, v_i^g)           //crossover operator
    if f(u_i^g) ≤ f(x_i^g) then                        //selection operator
      x_i^g = u_i^g
    end if
    g = g + 1
  end for
end while
```

Fig. 1. Pseudocode of classical DE (DE/rand/1).

2.3. Crossover operator

Following *mutation*, the *crossover*operator is applied to further increase the diversity of the population. In crossover, the target vectors, $\vec{x}_i^g$, are combined with elements from the donor vector, $\vec{v}_i^g$, to produce the trial vector, $\vec{u}_i^g$, using the *binomial crossover*,

$$u_{i,j}^g = \begin{cases} v_{i,j}^g & \text{if } rand(0,1) \le Cr \text{ or } j = j_{rand} \\ x_{i,j}^g & \text{otherwise} \end{cases} \tag{6}$$

where $Cr \in (0, 1)$ is the probability of crossover, j_{rand} is a random integer in $[1, n]$. Let

$$\vec{u}_i^g = binomial\_crossover(\vec{x}_i^g, \vec{v}_i^g)$$

denote the *crossover* operator.

2.4. Selection operator

Finally, the *selection* operator is employed to maintain the most promising trial individuals in the next generation. The classical DE adopts a simple selection scheme. It compares the objective values of the target $\vec{x}_i^g$ and trial $\vec{u}_i^g$ individuals. If the trial individual reduces the value of the objective function then it is accepted for the next generation; otherwise the target individual is retained in the population. The *selection* operator is defined as

$$\vec{x}_i^{g+1} = \begin{cases} \vec{u}_i^g, & \text{if } f(\vec{u}_i^g) < f(\vec{x}_i^g) \\ \vec{x}_i^g, & \text{otherwise.} \end{cases} \tag{7}$$

The pseudocode of the classical DE algorithm (DE/rand/1) is illustrated in Fig. 1.

3. Convergence analysis of classical differential evolution

Unlike other evolutionary algorithms, DE modifies individuals by using differences of randomly sampled pairs of individual vectors from the population. If the population traps in a local optimum set, then differences of any pairs are equal to 0. So DE cannot escape from local optima. This is one of reasons resulting in the fact that the classical DE does not converge to the global optimum with probability 1. In turn, this section will theoretically prove that the classical DE cannot converge in probability to the global optimum.

There are different definitions of *convergence in probability*. The following one is used in this paper.

Definition 1 (*Convergence in Probability [27,28]*)**.** Let $\{x(t), t = 0, 1, 2, \ldots\}$ be a population sequence generated by a population-based stochastic algorithm, the stochastic sequence $\{x(t)\}$ weakly converges in probability to the global optimum, if and only if:

$$\lim_{t\to\infty} p\{x(t) \cap B^* \ne \phi\} = 1,$$

where B^* is the set of the global optima of an optimization problem. That is, the algorithm holds with *convergence in probability*. Otherwise, the sequence $\{x(t)\}$ or the algorithm is called *no-convergence in probability*.

A population sequence $\{x(t)\}$ holds with convergence in probability, that is to say, the probability that the best individual sequence of the population sequence $\{x(t)\}$ converges to the global optimum approaches 1 as the iteration time t increases indefinitely. *No-convergence in probability* implies that the limitation (if exists) of the convergent probability is less than 1, which does not imply that the algorithm must not converge to the global optimum. In fact, the algorithm with no-convergence in probability may hold with a high convergence ratio. However, an algorithm with convergence in probability is generally more robust than one with no-convergence in probability.

We can map the continuous search space Ψ to a finite discrete set Φ due to the limitations of the numerical calculation accuracy in computer. So we consider a stochastic process $\{x(t), t = 0, 1, 2, \ldots\}$ that takes on a finite number of possible values. And define the state space of the stochastic process be the population space of the classical DE. If the population scale is m, then the state space of the stochastic process is

$$\Phi^m = \underbrace{\Phi \times \Phi \times \cdots \times \Phi}_{m}.$$

Let $x(t)$ denote the tth generation population of the classical DE, then evolutionary process of the classical DE can be described as a stochastic process $\{x(t), t = 0, 1, 2, \ldots\}$ that takes on a finite number of possible values. Unless otherwise mentioned, Let X, Y, Z, U, V denote the states of Φ^m, $p\{x(t+1) = V \mid x(t) = X\}$ denote the probability that the process will, when in state X at time t, next make a transition into state V.

Let M^o, C^o, S^o denote the mutation operator, the crossover operator and the selection operator of the classical DE, respectively. Then $p\{M^o(X) = Y\}$ is the probability that X changes into Y at t generation by the mutation operator of the classical DE. Similarly, the probabilities $p\{C^o(X, Y) = Z\}, p\{S^o(X, Z) = V\}$ are not difficult to understand. Hence,

$$p\{x(t+1) = V \mid x(t) = X\} = \sum_{Y,Z\in\Phi^m} \{p\{M^o(X) = Y\} \cdot p\{C^o(X, Y) = Z\} \cdot p\{S^o(X, Z) = V\}\}. \tag{8}$$

Let Ω denote a local optima or premature solutions set. That is, Ω is a subset of the state space Φ^m, every vector (individual) of the population of Ω is equal and is not the global optimum. The mathematical description of Ω will be given at the following Lemma 1.

Lemma 1. *Let $\{x(t), t = 0, 1, 2, \ldots\}$ be the population sequence generated by the classical DE. The classical DE, if an arbitrary population $x(t)$ traps in a local optimum or premature set, cannot escape. That is, suppose that there exists a time t_k such that the population $x(t_k) \in \Omega$, then*

$$\lim_{t\to\infty} p\{x(t) = x(t_k) \mid x(t_k) \in \Omega\} = 1,$$

where

$$\Omega = \{X = (\underbrace{\vec{b}, \vec{b}, \ldots, \vec{b}}_{m}) : \vec{b} \in \Phi \text{ and } \vec{b} \notin B^*\}.$$

Proof. All the vectors of the population $x(t_k)$ is equal when the state $x(t_k) \in \Omega$ at time t_k, so the difference of two arbitrary vectors is 0. By the formula (1)–(5), we can get that

$$p\{M^o(x(t_k)) = x(t_k)\} = 1, \quad \forall\, x(t_k) \in \Omega.$$

Obviously, by the formula (6), we can get that

$$p\{C^o(x(t_k), x(t_k)) = x(t_k)\} = 1, \quad \forall\, x(t_k) \in \Omega.$$

So, by the formula (8), for an arbitrary positive integer t, we can get that

$$\begin{aligned} & p\{x(t+1) = X \mid x(t) = X \in \Omega\} \\ & \quad = \sum_{Y,Z\in\Phi^m} \{p\{M^o(X) = Y \mid X \in \Omega\} \cdot p\{C^o(X, Y) = Z \mid X \in \Omega\} \cdot p\{S^o(X, Z) = X \mid X \in \Omega\}\} \\ & \quad = p\{M^o(X) = X \mid X \in \Omega\} \cdot p\{C^o(X, X) = X \mid X \in \Omega\} \cdot p\{S^o(X, X) = X \mid X \in \Omega\} \\ & \quad = 1. \end{aligned}$$

Hence, for a given positive integer t_k, if the population $x(t_k) \in \Omega$, then

$$\lim_{t\to\infty} p\{x(t) = x(t_k) \mid x(t_k) \in \Omega\} = 1.$$

Theorem 1. *The population sequence generated by the class DE, $\{x(t), t = 1, 2, \ldots\}$, cannot converge in probability to the global optima set B^*. That is,*

$$\lim_{t\to\infty} p\{x(t) \cap B^* \neq \phi\} < 1.$$

Proof. Let $x(0)$ denote the initial population of the classical DE. The individuals of the $x(0)$ are uniformly distributed random vectors lying the discrete space Φ. The probability generating every individual of $x(0)$ is equal. So, according to classical model of probability, the probability that the initial population of the classical DE traps into a local optima or premature solutions set is obviously greater than 0. That is $p\{x(0) \in \Omega\} > 0$.

From the Lemma 1, when the $x(0) \in \Omega$, we have

$$\lim_{t\to\infty} p\{x(t) = x(0) \mid x(0) \in \Omega\} = 1,$$

implying that

$$\lim_{t\to\infty} p\{x(t) \cap B^* = \phi \mid x(0) \in \Omega\} = 1.$$

Hence, we can get

$$\begin{aligned}\lim_{t\to\infty} p\{x(t) \cap B^* \neq \phi\} &= 1 - \lim_{t\to\infty} p\{x(t) \cap B^* = \phi\} \\ &\leq 1 - \lim_{t\to\infty} p\{x(t) \cap B^* = \phi,\ x(0) \in \Omega\} \\ &\leq 1 - \lim_{t\to\infty} p\{x(t) \cap B^* = \phi \mid x(0) \in \Omega\} \cdot p\{x(0) \in \Omega\} \\ &= 1 - p\{x(0) \in \Omega\} < 1,\end{aligned}$$

implying that

$$\lim_{t\to\infty} p\{x(t) \cap B^* \neq \phi\} < 1.$$

According to Definition 1, the sequence $x(t)$ generated by the classical DE holds *no-convergence in probability*.

4. A modified differential evolution algorithm with convergence in probability

The Section 3 proved that the classical DE cannot converge in probability to the global optimum. From the process of the proof, the key reason resulting in *no-convergence in probability* of the classical DE is that it may trap into and cannot escape from a local optimum solution set. In order to overcome the disadvantages, a modified DE algorithm, called msDE, is proposed. The msDE algorithm employs two proposed operators, called *uniform mutation* and *diversity selection* operator. The uniform mutation generates uniformly distributed solutions, while the diversity selection accepts poor solutions with a little probability. Both operators improve the capacity of the msDE to avoid trapping into a local optimum or premature solution set, as well as assist the msDE in escaping from it. This makes the msDE converge in probability to the global optimum. The convergence proof of the msDE will be given in the following Section 5. The msDE works through a cycle of four operators including *mutation, crossover, uniform mutation* and *diversity selection* operator after an initialization. The mutation and the crossover are the same as the classical DE. The uniform mutation, diversity selection operators and the msDE's pseudocode are described in detail as follows.

4.1. Uniform mutation operator

After the classical DE's crossover operator, the msDE creates a *variation* vector $\vec{w}_i^g$ corresponding to each trial vector $\vec{u}_i^g$ by the *uniform mutation* operator. The uniform mutation runs independently on each trial vector by perturbing each element of the trial vector. Each trial vector is replaced with probability Um by a feasible solution randomly generated, where Um is a control parameter. The description of the *uniform mutation* is given as follow.

For each trial vector $\vec{u}_i^g$, generate a uniformly random number λ between 0 and 1. Then, for all $j = 1, 2, \ldots, n$

$$w_{i,j}^g = \begin{cases} L_j + rand(0, 1) \cdot (U_j - L_j) & \lambda \leq Um \quad \text{(a)} \\ u_{i,j}^g \quad \text{otherwise} & \quad\quad\quad\quad \text{(b)} \end{cases} \tag{9}$$

Let $\vec{w}_i^g = uniform\_mutation(\vec{u}_i^g)$ denote the *uniform mutation* operator.

From the formula of the uniform mutation, the variation vector $\vec{w}_i^g$ equals to the trail vector $\vec{u}_i^g$ with probability $1-Um$, or a uniformly distributed random vector in the feasible region with Um. Obviously, the variation vector $\vec{w}_i^g$ equals to $\vec{u}_i^g$ when $Um = 0$. When Um is a smaller number in $(0, 1)$, the variation vector $\vec{w}_i^g$ is regenerated with a smaller probability Um. This makes the population of the msDE can escape from a local optimum and premature solution set. Meanwhile, the variation vector $\vec{w}_i^g$ equals to $\vec{u}_i^g$ with a larger probability. This makes the msDE maintain the search capability of the classical DE. And in order to keep the advantage of the classical DE, Um is generally set to a smaller number in $(0, 1)$.

4.2. Diversity selection operator

To further enhance the potential diversity of the population, a selection operator, called *diversity selection*, is employed after the uniform mutation operator by the msDE. The *diversity selection* determines whether the target or the variation vector survives to the next generation. Unlike the selection operator of the classical DE, the diversity selection operator

```
X = initial._population (m), F, CR = initial._parameters
while ! termination_condition do
  for g = 0 to m
    v_i^g = x_{r1}^g + F(x_{r2}^g − x_{r3}^g)        //mutation operator
    u_i^g = binomial_crossover(x_i^g, v_i^g)    //crossover operator
    w_i^g = uniform_mutation(u_i^g)             //uniform mutation
    x_i^g = diversity_selection(x_i^g, w_i^g)   //diversity selection
    g = g + 1
  end for
end while
```

Fig. 2. Pseudocode of msDE (msDE/rand/1).

retains a best individual to the next generation as well as accepts some poor solutions with a little-probability *Ds*. *Ds* is a control parameter and generally set to a smaller number in (0, 1). The diversity selection enhance the population diversity of the msDE. Thus the capacity of the msDE to avoid trapping into a local optimum or premature solutions set is improved. The diversity selection is described as follow.

Firstly, randomly choose one of the best individuals from all the target vectors $\vec{x}_i^g$ and all the variation vectors $\vec{w}_i^g$, and retain the best individual to the next generation. Suppose that the best individual chosen is $\vec{b}_k^g$, then $\vec{x}_k^{g+1} = \vec{b}_k^g$. Subsequently, at each variation vector $\vec{w}_i^g,\ i \in \{1, 2, \ldots, m\}\backslash\{k\}$, a operator generating its next individual is run as the following rule.

$$\vec{x}_i^{g+1} = \begin{cases} \vec{w}_i^g & f(\vec{x}_i^g) > f(\vec{w}_i^g) \wedge rand(0,1) > Ds \\ & \text{or } f(\vec{x}_i^g) \le f(\vec{w}_i^g) \wedge rand(0,1) \le Ds \\ \vec{x}_i^g & f(\vec{x}_i^g) < f(\vec{w}_i^g) \wedge rand(0,1) > Ds \\ & \text{or } f(\vec{x}_i^g) \ge f(\vec{w}_i^g) \wedge rand(0,1) \le Ds, \end{cases} \tag{10}$$

where the control parameter *Ds* is the probability that the *diversity selection* accepts poor solutions. Generally, *Ds* is set to a smaller value in (0, 1). Let $\vec{x}_i^{g+1} = diversity\_selection(\vec{x}_i^g, \vec{w}_i^g)$ denote the *diversity selection* operator.

The pseudocode of the msDE algorithm (msDE/rand/1) is illustrated in Fig. 2.

5. Convergence analysis of a modified differential evolution algorithm

There are the two classical convergence framework about the evolutionary algorithms. The one is represented by the elitist genetic algorithm [30,33] whose probability of generating any individuals, for each generation population, is always larger than 0. This means the probability getting a optimum in each generation during evolution process is always larger than 0 for arbitrary function as well as initialization. In addition, this type of algorithms also maintains the best solution to the next generation population, which can force the algorithm to converge in probability to the global optima.

The other can often be distinguished with their two characteristics. The main one is that bad individuals can be accepted with a small probability, and the other characteristic is that new individuals may only be generated over a *size*-fixed open set around the parent individual, such as DE-MC algorithm [26] which uses Metropolis selection operate and Rand walk strategy. This type of algorithms can achieve to certain asymptotic convergence when the *probability* and *size* satisfy some conditions. However, the theoretical analyses of Refs. [28,34] show that more parameters are involved in the convergence conditions. This means steady convergence of those algorithms is sensitive to more parameters, which is not conducive to their application.

In msDE algorithm proposed, the uniform mutation operator can generate new individuals over the whole space. Meanwhile the diversity selection operator maintains the best solution to the next population and accepts bad individuals with a small probability. The simultaneous use of these two operations can reduce the sensitivity of algorithm to the parameters.[1] The greedy acceptance of the best solution is important to the algorithm's convergence in probability.

Analogously to Section 3, a finite homogeneous Markov chain model with a positive recurrent set is developed to prove that the msDE can converge in probability to the global optimum. Define the state space of the stochastic process $\{x(t), t = 0, 1, 2, \ldots\}$ be the population space of msDE. Let $\mathcal{M}_1^o$, $\mathcal{M}^o$, and $\mathcal{S}^o$ denote the operators of Eq. (9), the uniform mutation and the diversity selection, respectively. The meaning of other symbols is the same as before.

State Y is said to be *accessible* from state X if for some integer $l \ge 0$, $p\{x(t+l) = Y \mid x(t) = X\} > 0$. Two states X and Y accessible to each other are said to *communicate* [35]. In turn, we will prove the conclusion that all states of the state space generated by the msDE without the diversity selection operate communicate.

[1] Theoretically, msDE can also converge to the global optimum in probability if only the new individual generated by *uniform mutation* is over a size-fixed open set around certain center. However, this brings a *size* parameter to msDE, and the algorithm is more sensitive to the other parameters, such as *Um, Ds*.

Property 1. *For the msDE without the diversity selection operator, all states (populations) of its population space Φ^m communicate. And for all states $X,\ Y \in \Phi^m$, the one-step transition probabilities from X to Y is greater than 0. That is,*

$$p\{\mathcal{M}^o \cdot C^o \cdot M^o(X) = Y\} > 0.$$

Proof. For the msDE without the diversity selection operator, only three operators, the mutation M^o, the crossover C^o and the uniform mutation operator $\mathcal{M}^o$, are used. Suppose Φ^m denote the population space of the msDE without the diversity selection operator, $\forall X, Y, Z \in \Phi^m$, according to the Eqs. (9)(a) and (b), we can get that

$$p\{\mathcal{M}^o \cdot C^o \cdot M^o(X) = Y\} = p\{C^t \cdot M^o(X) = Y\} \cdot (1 - Um) + Um \cdot \sum_{Z\in\Phi^m} \{p\{C^o \cdot M^o(X) = Z\} \cdot p\{\mathcal{M}_1^o(Z) = Y\}\},$$

and from the Eq. (9), we can get that

$$p\{\mathcal{M}_1^o(Z) = Y\} > 0, \quad \text{for all } Z.$$

So we have

$$Um \cdot \sum_{Z\in\Phi^m} \{p\{C^o \cdot M^o(X) = Z\} \cdot p\{\mathcal{M}_1^o(Z) = Y\}\} > 0,$$

hence,

$$p\{\mathcal{M}^o \cdot C^o \cdot M^o(X) = Y\} > 0.$$

That is to say, the one-step transition probability from arbitrary state X to arbitrary state Y is greater than 0 by using the uniform mutation. So, for the msDE without the diversity selection operator, all states communicate.

Let $f(X^*)$ denote the minimum function value of a population X. In turn, a statistic property of the diversity selection operator will be proved.

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

(i) *If $f(Z^*) \neq \min\{f(X^*),\ f(Y^*)\}$, then X, Y cannot generate Z by using the diversity selection. That is*

$$p\{\mathcal{S}^o(X, Y) = Z\} = 0.$$

(ii) *If $f(Z^*) = \min\{f(X^*),\ f(Y^*)\}$, then X, Y can generate Z by using the diversity selection. That is*

$$p\{\mathcal{S}^o(X, Y) = Z\} > 0.$$

Proof. This is practically obvious according to the operator's definition.

Theorem 2. *Suppose that $\{x(t), t = 0, 1, 2, \ldots\}$ is the population sequence generated by the msDE, then*

(i) *$\{x(t), t = 0, 1, 2, \ldots\}$ is a finite homogeneous Markov chain on the state space Φ^m;*
(ii) *$\{x(t), t = 0, 1, 2, \ldots\}$ converges in probability to the global optimum.*

Proof. (i) Like the above Section 3, the continuous search space Ψ is mapped to a finite discrete set Φ. So the state space of $\{x(t), t = 0, 1, 2, \ldots\}$ is finite. The reproduction operators of the msDE are independent of iteration times t and dependent only on the present state. So we obtain that the stochastic sequence $\{x(t), t = 0, 1, 2, \ldots\}$ is a finite homogeneous Markov chain.

(ii) Define $\mathcal{G}^o = \mathcal{S}^o \cdot \mathcal{M}^o \cdot C^o \cdot M^o$, $\forall X, Y, Z \in \Phi^m$, then the transition probability

$$\begin{aligned} p\{x(t+1) = Z \mid x(t) = X\} &= p\{\mathcal{G}^o(X) = Z\} \\ &= p\{\mathcal{S}^o \cdot (\mathcal{M}^o \cdot C^o \cdot M^o)(X) = Z\} \\ &= \sum_{Y\in\Phi^m} p\{(\mathcal{M}^o \cdot C^o \cdot M^o)(X) = Y\} \cdot p\{\mathcal{S}^o(X, Y) = Z\}. \end{aligned} \tag{11}$$

Define B_0 be a population set consist of some populations in which at least one individual is optimum. That is $B_0 \subset \Phi^m$:

$$B_0 = \{X = (\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_m) \in \Phi^m \mid x_i \in B^*, \exists i \in \{1, 2, \ldots, m\}\}.$$

Subsequently, we divided into two classes to discuss the transition probability.

(1) Suppose $X \in B_0, Z \notin B_0$

In this case,

$$f(Z^*) > \min\{f(X^*),\ f(Y^*)\},$$

implying that

$$f(Z^*) \neq \min\{f(X^*),\ f(Y^*)\}.$$

According to the Property 2, we can get

$$p\{\mathcal{S}^o(X, Y) = Z\} = 0,$$

From the formula (11), we can get

$$p\{x(t+1) = Z \mid x(t) = X\} = 0. \quad (12)$$

(2) Suppose $X \in B_0, Z \in B_0$

In this case,

$$f(Z^*) = \min\{f(X^*),\ f(Y^*)\},$$

according to the Property 2, we can get

$$p\{\mathcal{S}^o(X, Y) = Z\} > 0,$$

and from the Property 1, we have

$$p\{\mathcal{M}^o \cdot C^o \cdot M^o(X) = Y\} > 0 \quad \text{for all } X, Y \in \Phi^m.$$

So, from the formula (11)

$$p\{x(t+1) = Z \mid x(t) = X\} > 0. \quad (13)$$

Similarly, we can get

$$p\{x(t+1) = X \mid x(t) = Z\} > 0. \quad (14)$$

That is, all stats of B_0 communicate.

From the formulas (12)–(14), we can obtain that B_0 is a positive recurrent, irreducible, aperiodic and closed set, and $\Phi^m \backslash B_0$ is a non-recurrent state set. Where $\Phi^m \backslash B_0$ denotes the cutset of the Φ^m to the B_0.

In addition, according to the properties of the aperiodic, homogeneous Markov chain [35,28], we can get that the sequence $\{x(t), t = 0, 1, 2, \ldots\}$ exists a limiting distribution $\pi(Y)$, such that

$$\lim_{t\to\infty} p\{x(t) = Y\} = \begin{cases} \pi(Y) & Y \in B_0 \\ 0 & \text{otherwise.} \end{cases}$$

So

$$\lim_{t\to\infty} p\{x(t) \in B_0\} = 1.$$

We have

$$\lim_{t\to\infty} p\{x(t) \cap B^* \neq \phi\} = 1.$$

From the Definition 1, msDE converges in probability to the global optimum.

6. Experimental verification

The competitiveness of an overall convergent algorithm should be in two aspects, that is, capability of escaping from the local optimum set, and the performance of solving higher dimensional complex problems. So two experiments are designed to study the performance of the proposed msDE. One is conducted on DE's deceptive function [36]. As shown in Fig. 3, the function has many local optima and is strong deceptive to DE algorithm. This experiment presents several convergence figures visualizing the process that msDE escapes from the local optimum set. In order to measure the msDE's performance of solving higher dimensional complex problems, the other is conducted on a test suite from the literature [37], which includes 6 well-known benchmark functions with differential characterizations. And we employ the most common method for comparing algorithm's performance in IEEE World Congress on Evolutionary Computation (IEEE-CEC).

All the above algorithms are coded in Visual C++ and the experiments were executed on a ACER 4750G laptop with a 2.30 GHz Intel(R) Core(TM)i5 2410 M CPU and 2 GB RAM.

6.1. On deceptive function

The deceptive function are defined as following:

$$f(x) = \begin{cases} -3\text{sinc}(2x+10) & \text{if } -10 \le x < 0 \\ -\sqrt{x}\sin(x\pi) & \text{if } 0 \le x \le 10 \end{cases}$$

where the function $\text{sinc}(t)$ is given by:

$$\text{sinc}(t) = \begin{cases} 1 & \text{if } t = 0 \\ \sin(\pi t)/(\pi t) & \text{if } t \neq 0. \end{cases}$$

Table 1
Number of FEs to achieve the Ter_Err for the DE deceptive function using DE/best/1 and msDE/best/1.

RandSeed	20 634	827	16 665	17 648	1 290	5 645	4 735	25 706	22 480	17 754
msDE FEs	99 050	2 497 230	186 380	43 430	310	4 302 800	3 800	37 810	270	178 610
DE FEs	–	–	–	–	200	–	220	–	210	–
RandSeed	23 547	5 897	21 242	23 048	19 888	27 637	10 848	21 662	20 946	8 475
msDE FEs	958 100	5 380	34 810	1 204 340	1 339 880	2 330 350	8 487 780	621 590	1 373 070	240
DE FEs	–	–	–	–	–	–	–	–	260	–
RandSeed	28 295	25 759	30 384	29 988	26 309	27 159	14 833	21 927	22 182	13 683
msDE FEs	175 380	310	280	230	230	180	174 010	180	46 410	270
DE FEs	–	–	180	230	240	160	–	200	–	–
RandSeed	6 429	15 773	12 943	31 125	2 912	22 550	1 953	9 256	18 571	15 147
msDE FEs	2 287 910	260	1 902 460	380	876 700	819 320	230	27 200	330	390
DE FEs	210	–	–	–	–	230	240	–	–	280
RandSeed	6 880	4 145	348	**20 276**	917	18 151	27 903	13 883	17 649	17 093
msDE FEs	260	250	150 370	**1 410**	240	3 019 380	70 240	317 930	137 810	709 330
DE FEs	240	–	–	–	–	–	–	–	–	–

'FEs' denotes function evaluations.
'–' indicates that the algorithm cannot find the optimum within Max_FEs.

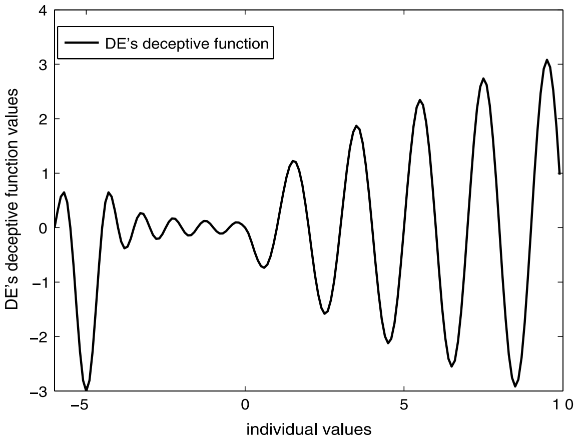


Fig. 3. DE's deceptive function.

The landscape of DE deceptive function is shown in Fig. 3. It can lead the classical DE to trap in the local optimum. The global optimum of the function is $x = -5.0$ with the function value $f(x) = -3$. There is a deceptive local minimum $x = 8.5060$ with function value $f(x) = -2.9160$ in this test function.

Experiments are conducted to compare the two versions of the classic DE with the msDE algorithm (DE/best/1 vs. msDE/best/1, DE/rand/1 vs. msDE/rand/1). From the Ref. [38], the same initialization benefits the comparison. So experiments generate 50 uniformly distributed random integers as the random seeds for initializing population. The 50 integers are listed in Tables 1 and 2. Using the 50 seeds, all experiments were run 50 times.

We set the parameters: Population size $Np = 10$, Mutation factor $F = 0.5$ [18,39], Crossover probability $Cr = 0.9$ [18,39], and the maximum number of function evaluations (Max_FEs) is set to 5,000,000, Uniform mutation probability is equal to Diversity selection probability $Um = Ds = 0.01$.

Table 1 presents the FEs of 50 runs of the classical DE/best/1 and the msDE/best/1 on the deceptive function when algorithm achieves the Ter_Err, while Table 2 reports the FEs of 50 runs of the classical DE/rand/1 and the msDE/rand/1 on the deceptive function. Table 3 analyzes the data of Tables 1 and 2. From the statistics of Table 3, the robustness of the msDE holding uniform mutation and diversity selection operators is better than the classical DE's. Figs. 4 and 5 present the convergence characteristics in terms of unreached global optimum for the classical DE. All of these curves show that the msDE can escape from the local optimum on the test functions while the classical DE traps in.

Of course, we have to note that convergence in probability is a property when the iteration times approaches infinity. The previous theorem and experimental results cannot imply that the msDE can solve all function optimization problems within a finite FEs.

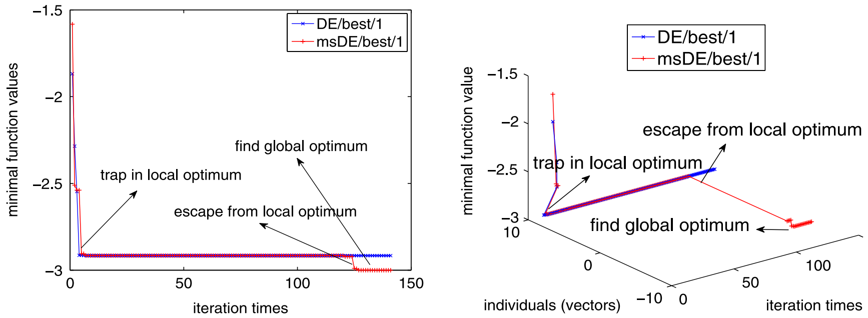
6.2. On a set of benchmark functions

A comparative study of DE and msDE is performed by using a test function set including 6 benchmark functions [37]. These are the Sphere function (f1), the Schwefel's problem 1.2 (f2), the generalized Rosenbrock's function (f3), the Ackley's function

Table 2
Number of FEs to achieve the Ter_Err for the DE deceptive function using DE/rand/1 and msDE/rand/1.

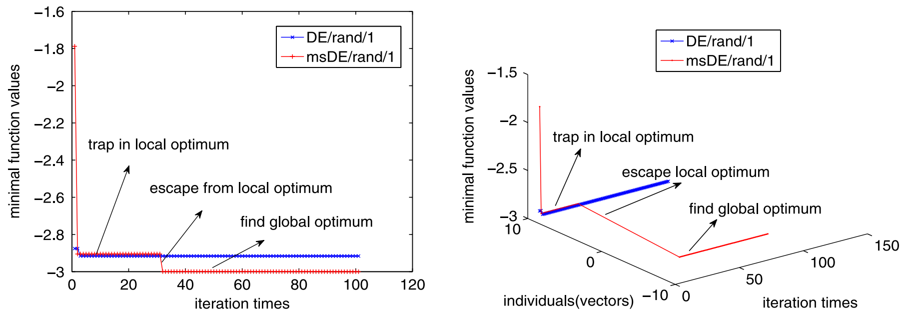
RandSeed	20 634	827	16 665	17 648	1 290	5 645	4 735	25 706	22 480	17 754
msDE FEs	3 390	700	710	5 250	810	820	12 970	33 360	380	500
DE FEs	–	–	320	460	350	340	280	450	260	–
RandSeed	23 547	5 897	21 242	23 048	19 888	27 637	10 848	21 662	20 946	8 475
msDE FEs	700	460	1 360	450	430	830	978 960	400	380	370
DE FEs	–	400	–	410	–	380	380	–	420	330
RandSeed	28 295	25 759	30 384	29 988	26 309	27 159	14 833	21 927	22 182	13 683
msDE FEs	310	1 540	940	430	2 390	470	29 860	380	520	590
DE FEs	–	360	450	340	400	300	530	480	–	–
RandSeed	6 429	15 773	12 943	31 125	2 912	22 550	1 953	9 256	18 571	15 147
msDE FEs	5 190	390	33 330	238 290	8 980	610	920	2 860	1 910	410
DE FEs	–	330	–	–	–	380	–	450	450	–
RandSeed	**6 880**	4 145	348	20 276	917	18 151	27 903	13 883	17 649	17 093
msDE FEs	**1 010**	470	3 643 110	4 150	1 060	46 070	430	360	27 040	610
DE FEs	–	490	–	420	370	400	–	310	310	–

'FEs' denotes function evaluations.
'–' indicates that the algorithm cannot find the optimum within Max_FEs.



(a) 2-dimension. (b) 3-dimension.

Fig. 4. Convergence figure of deceptive function (DE/best/1 vs. msDE/best/1 seed = 20276).



(a) 2-dimension. (b) 3-dimension.

Fig. 5. Convergence figure of deceptive function (DE/rand/1 vs. msDE/rand/1 seed = 6880).

Table 3
Statistical analysis of Tables 1 and 2.

Function	Deceptive function			
Algorithm */*/1	DE/best	msDE/best	DE/rand	msDE/rand
Running times	50	50	50	50
Convergence times	14	50	29	50
Convergence ratio	28%	100%	58%	100%

(f4), the generalized Rastrigin's function (f5) and the Salomon's function (f6). Those hold differential characterizations. The Sphere and the Schwefel's functions are continuous, unimodal and separable. The generalized Rosenbrock's optimum, which

Table 4
The function values achieved via msDE/rand/1 for test problems.

FEs		F1/1e−4	F2/1e−4	F3/1e−3	F4/1e−3	F5/2e−4	F6/1e−3
5e+4	Best	3.722569e−5	1.052869e+3	2.363611e+1	1.454127e−2	1.825215e+2	1.998767e−1
	Median	1.317602e−4	2.024659e+3	2.547796e+1	2.720807e−2	2.057561e+2	2.009906e−1
	Worst	3.354363e−4	2.951507e+3	2.625652e+1	4.375287e−2	2.252633e+2	2.998733e−1
	Mean	1.457969e−4	1.921639e+3	2.531318e+1	2.711668e−2	2.054805e+2	2.138486e−1
	St.d	7.101361e−5	5.704156e+2	5.775845e−1	7.652828e−3	1.005519e+1	2.811292e−2
1e+5	Best	2.909400e−11	2.108018e+1	1.775913e+1	1.273556e−5	1.460694e+2	1.823816e−1
	Median	1.414555e−10	5.033402e+1	1.945910e+1	2.279657e−5	1.914524e+2	1.998733e−1
	Worst	6.147642e−10	1.188379e+2	2.095030e+1	4.414247e−5	2.049187e+2	1.998831e−1
	Mean	1.757349e−10	5.530790e+1	1.960726e+1	2.556075e−5	1.884414e+2	1.991369e−1
	St.d	1.309825e−10	2.706403e+1	8.134248e−1	9.227977e−6	1.357429e+1	3.425007e−3
1.5e+5	Best	0.000000e+0	4.049995e−1	1.102680e+1	1.017461e−8	1.286904e+2	1.005215e−1
	Median	1.000000e−16	1.421101e+0	1.428152e+1	2.687541e−8	1.780161e+2	1.998733e−1
	Worst	7.000000e−16	8.546607e+0	1.640271e+1	6.500403e−8	1.954177e+2	1.998733e−1
	Mean	2.000000e−16	1.911737e+0	1.404771e+1	2.959854e−8	1.747391e+2	1.886040e−1
	St.d	2.000000e−16	1.624867e+0	1.087064e+0	1.263306e−8	1.701901e+1	2.857414e−2

‘FEs’ denotes function evaluations.

Table 5
The function values achieved via DE/rand/1 for test problems.

FEs		F1	F2	F3	F4	F5	F6
5e+4	Best	3.981338e−5	1.120324e+3	2.466296e+1	1.682092e−2	1.778458e+2	1.998739e−1
	Median	1.460035e−4	1.904931e+3	2.558842e+1	2.378208e−2	2.024265e+2	2.000300e−1
	Worst	4.687370e−4	2.874036e+3	2.675050e+1	3.950343e−2	2.239820e+2	2.998734e−1
	Mean	1.606499e−4	1.938913e+3	2.560035e+1	2.366505e−2	2.025327e+2	2.110259e−1
	St.d	9.388241e−5	5.005268e+2	5.312136e−1	4.722956e−3	1.280519e+1	2.698877e−2
1e+5	Best	3.961620e−11	2.656309e+1	1.812013e+1	1.556676e−5	1.559871e+2	1.399475e−1
	Median	1.225084e−10	5.554179e+1	2.020160e+1	2.401075e−5	1.858864e+2	1.998733e−1
	Worst	6.044953e−10	1.376549e+2	2.223397e+1	4.564242e−5	2.043446e+2	1.998739e−1
	Mean	1.768590e−10	5.532780e+1	2.010210e+1	2.558901e−5	1.854608e+2	1.974763e−1
	St.d	1.486436e−10	2.549677e+1	9.339488e−1	6.595276e−6	1.238314e+1	1.174301e−2
1.5e+5	Best	0.000000e+0	7.171423e−1	1.253526e+1	1.137130e−8	1.559871e+2	1.030063e−1
	Median	1.000000e−16	1.500765e+0	1.505183e+1	2.241535e−8	1.759913e+2	1.998733e−1
	Worst	7.000000e−16	4.038457e+0	1.689306e+1	4.104023e−8	1.925887e+2	1.998734e−1
	Mean	2.000000e−16	1.650912e+0	1.476824e+1	2.432976e−8	1.773232e+2	1.902609e−1
	St.d	2.000000e−16	8.560127e−1	1.013068e+0	6.618557e−9	1.048620e+1	2.656269e−2

‘FEs’ denotes function evaluations.

is multimodal and non-separable, lies inside a parabolic shaped flat valley. The Ackley’s function is a multimodal non-separable problem and has many local optima and a narrow global optimum. The generalized Rastrigin function is a complex multimodal separable problem with many local optima. The Salomon’s function is a highly multimodal and non-separable function.

According to the criterion of IEEE-CEC, 25 independent runs were performed for each test function, and the stop criterion was to run up to 15,000 fitness evaluations. The detailed results (best, median, worst, mean, St.d), for 50,000, 1000,000 and 150,000 FEs, are presented in Tables 4 and 5.

We set the parameters: Dimension $D = 30$, Population size $Np = 100$, Mutation factor $F = 0.5$, Crossover probability $Cr = 0.9$, Uniform mutation probability is equal to Diversity selection probability $Um = Ds$, whose values are shown the 1st row of in Table 4.

Tables 4 and 5 show the results for the conducted experiments on msDE and DE, respectively. By comparing these two table, it can be seen that in the simplest Sphere function msDE’s convergence rate is similar to the classical DE, and is faster in the other problems. That is to say, msDE which employed uniform mutation and diversity selection operators is effective for all problems of the benchmark set.

7. Conclusion and future work

DE algorithm is a stochastic search algorithm. The theoretical research field of DE’s global convergence still remains large space. Generally speaking, the global convergence in probability of a stochastic search algorithm is that it can converge in probability to the global optima set for any initialization and any optimization problem. This paper proved that the classical DE does not hold convergence in probability by using the property that it cannot escape from a local optimal set. The theoretical analysis also indicated that we can improve DE algorithm by the following two facts: (1) enhancing the capability of escaping from a local optimal set, (2) reducing the probability trapping into a local optimal set. In this way, this paper proposed a msDE algorithm and proved it's global convergence in probability by developing a Markov chain with only one positive recurrent. A numerical experiment on deceptive function visualized the process that msDE escaped from the local optimum set. The other comparative experiments of msDE and the classical algorithm are conducted on a benchmark function set. The results indicated the superiority of msDE.

Several possible directions for future work can be summarized below:

- Develop rapid DE algorithms with convergence in probability by using strategies enhancing the DE's capability of escaping from a local optimal set. For example, let those strategies which can make populations communicate integrate into some outstanding variants of modified DE algorithms, such as the representative modified DE of Ref. [11] as well as the 2-opt DE algorithm [18]. Of course, whether the algorithms can converge in probability to the global optimum need to be proved. And their convergence rate also needs to be further investigated.
- Analyze the asymptotic convergence and the convergence in distribution of the classical DE. In fact, that the classical DE holds with no-convergence in probability does not mean that the algorithm does not holds with convergence in distribution.
- Develop modified DE algorithms which are almost certainly convergent.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant Nos. 61170202, 61370092, and 61202287, and the Fundamental Research Funds for the Central Universities under Grant No. 2012-YB-19.

References

[1] R. Storn, K.V. Price, Differential Evolution: A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces. Technical Report, International Computer Science Institute, 1995.
[2] Price, K.V. Lampinen, Differential Evolution: A Practical Approach to Global Optimization, Springer press, Berlin, 2005.
[3] S. Das, P.N. Suganthan, Differential evolution: a survey of the state-of- the-art, IEEE Trans. Evol. Comput. 1 (2011) 4–31.
[4] S. Das, P.N. Suganthan, C.A. Coello Coello, Guest editorial: special issue on differential evolution, IEEE Trans. Evol. Comput. 1 (2011) 1–3.
[5] R. Storn, K.V. Price, Differential evolution: a simple and efficient heuristic for global optimization over continuous spaces, J. Global Optim. 4 (1997) 341–359.
[6] R. Storn, K.V. Price, Minimizing the real functions of the ICEC'96 contest by differential evolution, in: IEEE Int. Conf. on Evol. Comput., Nagoya, 1996, pp. 842–844.
[7] K.V. Price, Differential evolution versus the functions of the 2nd ICEO, in: IEEE Int. Conf. on Evol. Comput., Indianapolis, 1997, pp. 153–157.
[8] X.S. Wang, M.L. Hao, Y.H. Cheng, On the use of differential evolution for forward kinematics of parallel manipulators, J. Appl. Math. Comput. 2 (2008) 760–769.
[9] I.D. Falco, A.D. Cioppa, D. Maisto, E. Tarantino, Differential evolution as a viable tool for satellite image registration, J. Appl. Soft Comput. 4 (2008) 1453–1462.
[10] I.L.L. Cruz, L.G. Van Willigenburg, G. Van Straten, E. Tarantino, Efficient differential evolution algorithms for multimodal optimal control problems, J. Appl. Soft Comput. 2 (2003) 97–122.
[11] F. Neri, V. Tirronen, Recent advances in differential evolution: a survey and experimental analysis, J. Artif. Intell. Rev. 33 (2010) 61–106.
[12] N. Noman, H. Iba, Accelerating differential evolution using an adaptive local search, IEEE Trans. Evol. Comput. 1 (2008) 107–125.
[13] S. Das, A. Abraham, U.K. Chakraborty, A. Konar, Differential evolution with a neighborhood-based mutation operator, IEEE Trans. Evol. Comput. 3 (2009) 526–553.
[14] G.E. Michael, K.T. Dimitris, G.P. Nicos, P.P. Vassilis, N.V. Michael, Enhancing differential evolution utilizing proximity-based mutation operators, IEEE Trans. Evol. Comput. 1 (2011) 99–119.
[15] E. Dorroonsoro, P. Bouvry, Improving classical and decentralized differential evolution with new mutation operator and population topologies, IEEE Trans. Evol. Comput. 1 (2011) 67–98.
[16] Y. Wang, Z.X. Cai, Q.F. Zhang, Differential evolution with composite trial vector generation strategies and control parameters, IEEE Trans. Evol. Comput. 15 (1) (2011) 55–66.
[17] H.Y. Fan, J. Lampinen, A trigonometric mutation operation to differential evolution, J. Global Optim. 1 (2003) 105–129.
[18] C.E. Chiang, W.P. Lee, G.P. Heh, A 2-opt based differential evolution for global optimization, J. Appl. Soft Comput. 10 (2010) 1200–1207.
[19] E. Mininno, F. Neri, F. Cupercesco, Compact differential evolution, IEEE Trans. Comput. 1 (2011) 32–54.
[20] J. Brest, M.S. Maučec, Population size reduction for the differential evolution algorithm, J. Appl. Intell. 3 (2008) 228–247.
[21] F. Neri, V. Tirronen, Scale factor local search in differential evolution, J. Memet. Comput. J. 2 (2009) 153–171.
[22] A.K. Qin, V.L. Huang, P.N. Suganthan, Differential evolution algorithm with strategy adaptation for global numerical optimization, IEEE Trans. Evol. Comput. 2 (2009) 398–417.
[23] Z.B. Hu, Q.H. Su, S.W. Xiong, F.G. Hu, Self-adaptive hybrid differential evolution with simulated annealing algorithm for numerical optimization, in: IEEE World Congress Computational Intelligence, IEEE Press, Hongkong, 2008, pp. 1189–1194.
[24] F. Xue, A.C. Sanderson, R.J. Graves, Modeling and convergence analysis of a continuous multiobjective differential evolution algorithm, in: IEEE Cong. Evol. Comput., IEEE Press, Edinburgh, 2005, pp. 228–235.
[25] F. Xue, A.C. Sanderson, R.J. Graves, Multiobjective differential evolution: algorithm, convergence analysis, and applications, in: Proc. IEEE Congr. Evol. Comput., vol. 1, 2005, pp. 743–745.
[26] C.J.F. ter Braak, A Markov chain Monte Carlo version of the genetic algorithm differential evolution: easy Bayesian computing for real parameter spaces, Statist. Comput. 16 (2006) 239–249.
[27] M.B. Robert, Pointwise properties of convergence in probability, Statist. Probab. Lett. 6 (1985) 315–316.
[28] Z.B. Xu, Computational Intelligence: Simulating Evolution Computation, Higher Education Press, Beijing, 2004 (in Chinese).
[29] D.E. Goldberg, P. Segrest, Finite Markov chain analysis of genetic algorithm, in: Proc. Int. Conf. Genetic Algorithms, Hillsdale, 1987.
[30] J. Suzuki, A Markov chain analysis on simple genetic algorithms, IEEE Trans. Syst. Man Cybern. 4 (1995) 655–659.
[31] L. Ming, Y.P. Wang, Y.M. Cheng, On convergence rate of a class of genetic algorithms, in: Proc. World Automation Cong., Budapest, 2006.
[32] M. Weber, F. Neri, V. Tirronen, A study on scale factor in distributed differential evolution, J. Inf. Sci. (2011) 2488–2511.
[33] G. Rudolph, Convergence analysis of canonical genetic algorithms, IEEE Trans. Neural Netw. 1 (1994) 96–101.
[34] Z.B. Xu, Z.K. Nie, W.C. Zhang, Almost sure convergence of genetic algorithms: a martingale approach, J. Comput. 8 (2002) 785–794 (in Chinese).
[35] M.R. Sheldon, Stochastic Processes, Wiley, New York, 1996.

[36] J.Y. Yan, Q. Ling, D.M. Sun, A differential evolution with simulated annealing updating method, in: Proc. Int. Conf. Machine Learning and Cybernetics. DaLian, 2006, pp. 2103–2106.
[37] S.K. Goudos, et al., A comparative study of common and self-adaptive differential evolution strategies on numerical benchmark problems, Procedia Comput. Sci. 3 (2011) 83–88.
[38] P.N. Suganthan, et al., Problem Definition and Evaluation Criteria for the CEC2005 Special Session on Real-Parameter Optimization. Technical Report, Institute of Electrical and Electronics Engineers, 2005.
[39] J. Brest, S. GrFESer, B. Boskovic, M. Mernik, V. Zumer, Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems, IEEE Trans. Evol. Comput. 6 (2006) 646–657.