Applied Soft Computing 15 (2014) 149–168

Contents lists available at ScienceDirect

Applied Soft Computing

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

Repairing the crossover rate in adaptive differential evolution☆

Wenyin Gong*, Zhihua Cai, Yang Wang

School of Computer Science, China University of Geosciences, Wuhan 430074, PR China

ARTICLE INFO

Article history:
Received 22 August 2012
Received in revised form 5 November 2012
Accepted 6 November 2013
Available online 22 November 2013

Keywords:
Differential evolution
Parameter adaptation
Crossover rate repair
Binary string
Numerical optimization

ABSTRACT

Differential evolution (DE) is a simple yet powerful evolutionary algorithm (EA) for global numerical optimization. However, its performance is significantly influenced by its parameters. Parameter adaptation has been proven to be an efficient way for the enhancement of the performance of the DE algorithm. Based on the analysis of the behavior of the crossover in DE, we find that the trial vector is directly related to its binary string, but not directly related to the crossover rate. Based on this inspiration, in this paper, we propose a crossover rate repair technique for the adaptive DE algorithms that are based on successful parameters. The crossover rate in DE is repaired by its corresponding binary string, *i.e.* by using the average number of components taken from the mutant. The average value of the binary string is used to replace the original crossover rate. To verify the effectiveness of the proposed technique, it is combined with an adaptive DE variant, JADE, which is a highly competitive DE variant. Experiments have been conducted on 25 functions presented in CEC-2005 competition. The results indicate that our proposed crossover rate technique is able to enhance the performance of JADE. In addition, compared with other DE variants and state-of-the-art EAs, the improved JADE method obtains better, or at least comparable, results in terms of the quality of final solutions and the convergence rate.

1. Introduction

Differential evolution (DE), proposed by Storn and Price in 1995 [1,2], is a simple, efficient, and versatile population-based evolutionary algorithm (EA) for the global numerical optimization. The advantages are its simple structure, ease of use, speed, and robustness. Due to these advantages, DE has been successfully applied in diverse fields, including data mining, pattern recognition, digital filter design, etc. [3,4]. In addition, recent studies demonstrate the highly competitive performance provided by DE in constrained optimization problems, multi-objective optimization problems, and other complex problems. More details on the state-of-the-art research within DE can be found in two surveys [5,6] and the references therein.

There are three algorithmic parameters in the original DE algorithm, which are (i) the population size *NP*; (ii) the crossover rate *CR*; and (iii) the scaling factor *F*. Originally, these parameters are user-specified and kept fixed during the run. However, recent studies indicate that the performance of DE is very sensitive to the parameter setting and the choice of the best parameters is always problem-dependent [7–9]. In order to obtain acceptable results, we need different parameter settings for different problems at hand. Even for the same problem, different parameters are required at different stages of evolution. Thus, some researchers investigated the parameter adaptation techniques to adaptively choose the parameters for the DE algorithm, including jDE [9], SaDE [10], JADE [11], and so on. These adaptive DE variants obtained very promising results in the DE literature.

In this paper, we first analyze the behavior of the crossover operator. Then, we propose a crossover rate repair technique for the adaptive DE algorithm. The crossover rate in DE is repaired by its corresponding binary string, *i.e.* by using the average number of components taken from the mutant. As it will be explained in the following sections, we can see that the crossover rate repair technique is very simple. In order to evaluate the efficiency of our proposed technique, it is combined with an adaptive DE variant, JADE [11], which is a highly competitive DE variant. Experiments have been conducted on 25 benchmark functions presented in CEC-2005 competition [12] on real-parameter numerical optimization. In addition, the proposed crossover rate repair technique is also incorporated into SaDE [10] and EPSDE [13]. Experimental results indicate that this technique is able to enhance the performance of JADE, SaDE, and EPSDE in the test functions at $D = 30$ and $D = 50$. Moreover, compared with other DE variants and state-of-the-art

☆ This work was partly supported by the National Natural Science Foundation of China under Grant Nos. 61203307, 61075063, and 61375066, the National Natural Science Foundation of Hubei Province under Grant No. 2013CFA004, the Fundamental Research Funds for the Central Universities at China University of Geosciences (Wuhan) under Grant Nos. CUG130413 and CUG090109, and the Research Fund for the Doctoral Program of Higher Education under Grant No. 20110145120009.

* Corresponding author. Tel.: +86 27 67883716; fax: +86 27 67883716.
E-mail addresses: wenyingong@yahoo.com, wygong@cug.edu.cn (W. Gong), zhcai@cug.edu.cn (Z. Cai).

1568-4946/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.asoc.2013.11.005

EAs, the improved JADE method obtains better, or at least comparable, results in terms of the quality of final solutions and the convergence rate.

The rest of this paper is organized as follows. Section 2 briefly introduces the original DE algorithm and the related work. In Section 3 we present our proposed crossover rate repair technique in detail. In Section 4, we comprehensively evaluate the performance of our approach through different experiments. In Section 5, we conclude the work of this paper.

2. Related work

In this section, we first briefly introduce the original DE algorithm. Then, the studies on the influence of crossover in DE are briefly introduced. Finally, the recently proposed adaptive DE variants in the literature are surveyed.

2.1. Differential evolution

DE algorithm is initially proposed to solve numerical optimization problems. Without loss of generality, in this work, we consider the following numerical optimization problem:

$$\text{Minimize} \quad f(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^D, \tag{1}$$

where $\mathbf{x} = [x_1, x_2, \ldots, x_D]^T$, and D is the dimension, *i.e.*, the number of decision variables. Generally, for each variable x_j, it satisfies a boundary constraint, such that:

$$L_j \le x_j \le U_j, j = 1, 2, \ldots, D. \tag{2}$$

where L_j and U_j are respectively the lower bound and upper bound of x_j.

2.1.1. Initialization

The DE population consists of NP vectors. Initially, the population is generated at random. For example, for the ith vector $\mathbf{x}_i$ it is initialized as follows:

$$x_{i,j} = L_j + \text{rndreal}(0,1) \cdot (U_j - L_j) \tag{3}$$

where $i = 1, \ldots, NP$, $j = 1, \ldots, D$, and rndreal(0, 1) is a uniformly distributed random real number in (0, 1).

2.1.2. Mutation

After initialization, the mutation operation is applied to generate the mutant vector $\mathbf{v}_i$ for each target vector $\mathbf{x}_i$ in the current population. There are many mutation strategies available in the literature [3,14,11], the classical one is "DE/rand/1":

$$\mathbf{v}_i = \mathbf{x}_{r_1} + F \cdot (\mathbf{x}_{r_2} - \mathbf{x}_{r_3}) \tag{4}$$

where F is the mutation scaling factor, r_1, r_2, $r_3 \in \{1, \ldots, NP\}$ are mutually different integers randomly generated, and $r_1 \ne r_2 \ne r_3 \ne i$.

2.1.3. Crossover

In order to diversify the current population, following mutation, DE employs the crossover operator to produce the trial vector $\mathbf{u}_i$ between $\mathbf{x}_i$ and $\mathbf{v}_i$. The most commonly used operator is the *binomial* or *uniform* crossover performed on each component as follows:

$$u_{i,j} = \begin{cases} v_{i,j}, & \text{if}(\text{rndreal}(0,1) < CR \text{or} j = j_{rand}) \\ x_{i,j}, & \text{otherwise} \end{cases} \tag{5}$$

where CR is the crossover rate and j_{rand} is a randomly generated integer within [1, D]. It is worth noting that there are other crossover operators in DE, such as the *exponential* crossover [3]. However, in this paper, we only focus on the binomial crossover mentioned above due to its promising performance obtained.

2.1.4. Selection

Finally, to keep the population size constant in the following generations, the selection operation is employed to determine whether the trial or the target vector survives to the next generation. In DE, the *one-to-one tournament selection* is used as follows:

$$\mathbf{x}_i = \begin{cases} \mathbf{u}_i, & \text{if} f(\mathbf{u}_i) \le f(\mathbf{x}_i) \\ \mathbf{x}_i, & \text{otherwise} \end{cases} \tag{6}$$

where $f(\mathbf{x})$ is the objective function to be optimized. For the sake of clarity, the pseudo-code of DE with "DE/rand/1/bin" is given in Algorithm 1, where rndint(1, D) returns a uniformly distributed random integer number between 1 and D.

Algorithm 1. The DE algorithm with "DE/rand/1/bin"

```
Generate the initial population
Evaluate the fitness for each individual
while the halting criterion is not satisfied do
  for i = 1 to NP do
    Select uniform randomly r1 ≠ r2 ≠ r3 ≠ i
    j_rand = rndint(1, D)
    for j = 1 to D do
      if rndreal_j(0, 1) < CR or j is equal to j_rand then
        u_i,j = x_r1,j + F · (x_r2,j − x_r3,j)
      else
        u_i,j = x_i,j
      end if
    end for
  end for
  for i = 1 to NP do
    Evaluate the offspring u_i
    if f(u_i) is better than or equal to f(x_i) then
      Replace x_i with u_i
    end if
  end for
end while
```

2.2. Influence of crossover in DE

The crossover operator, which is designated to enhance the potential diversity of the population, plays an important role in DE. In the DE family of algorithms there are mainly two kinds of crossover methods: *binomial* and *exponential* [3]. Between the two crossover methods, there are two essential differences: (i) the probability distribution of crossover length; and (ii) the inheritance continuity [15]. In the binomial crossover, the relation between the probability distribution and its crossover rate CR is linear; while in the exponential crossover the relation is nonlinear [16,17]. Through exponential crossover the trial vector gets a fraction of the mutant consecutively (in cyclic sense) while the inheritance by binomial crossover is non-consecutive [15].

In the DE literature, there are some studies that have examined the influence of crossover. In [16,17], Zaharie analyzed the influence of the crossover operator and the crossover rate CR on the behavior of DE. The relation between mutation probability p_m and crossover rate CR is also theoretically analyzed for several variants of crossover in [16,17]. Lin et al. presented theoretical analysis and comparative study of different crossover methods in DE to better understand the role of crossover [15]. They also designed two new crossover methods, namely consecutive binomial crossover and non-consecutive exponential crossover. In [15], the authors concluded that the choice of the proper crossover method and its associated parameters is dependent on the features of the problems.

The crossover rate CR is used to control which and how many components to be mutated in each element of the current population [17]. Low values of CR result in a small number of parameters