Applied Soft Computing 18 (2014) 232–247

Contents lists available at ScienceDirect

Applied Soft Computing

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

Differential evolution based on covariance matrix learning and bimodal distribution parameter setting

Yong Wang[a,b], Han-Xiong Li[b,c,*], Tingwen Huang[d], Long Li[a]

[a] School of Information Science and Engineering, Central South University, Changsha 410083, PR China
[b] Department of Systems Engineering and Engineering Management, City University of Hong Kong, Hong Kong, PR China
[c] State Key Laboratory of High Performance Complex Manufacturing, Central South University, Changsha 410083, PR China
[d] Texas A&M University at Qatar, Doha 5825, Qatar

ARTICLE INFO

Article history:
Received 19 April 2013
Received in revised form
21 December 2013
Accepted 28 January 2014
Available online 3 February 2014

Keywords:
Differential evolution
Global numerical and engineering optimization
Covariance matrix learning
Bimodal distribution parameter setting

ABSTRACT

Differential evolution (DE) is an efficient and robust evolutionary algorithm, which has been widely applied to solve global optimization problems. As we know, crossover operator plays a very important role on the performance of DE. However, the commonly used crossover operators of DE are dependent mainly on the coordinate system and are not rotation-invariant processes. In this paper, covariance matrix learning is presented to establish an appropriate coordinate system for the crossover operator. By doing this, the dependence of DE on the coordinate system has been relieved to a certain extent, and the capability of DE to solve problems with high variable correlation has been enhanced. Moreover, bimodal distribution parameter setting is proposed for the control parameters of the mutation and crossover operators in this paper, with the aim of balancing the exploration and exploitation abilities of DE. By incorporating the covariance matrix learning and the bimodal distribution parameter setting into DE, this paper presents a novel DE variant, called CoBiDE. CoBiDE has been tested on 25 benchmark test functions, as well as a variety of real-world optimization problems taken from diverse fields including radar system, power systems, hydrothermal scheduling, spacecraft trajectory optimization, etc. The experimental results demonstrate the effectiveness of CoBiDE for global numerical and engineering optimization. Compared with other DE variants and other state-of-the-art evolutionary algorithms, CoBiDE shows overall better performance.

1. Introduction

Differential evolution (DE), proposed by Storn and Price [1,2] in 1995, has become a hotspot in the community of evolutionary computation. Similar to other evolutionary algorithms (EAs), DE is a population-based optimization algorithm. In DE, each individual in the population is called a target vector. DE produces a mutant vector by making use of the mutation operator, which perturbs a target vector using the difference vector of other individuals in the population. Afterward, the crossover operator is applied to the target vector and the mutant vector to generate a trial vector. Finally, the trial vector competes with its target vector for survival according to their objective function values. Due to some advantages, e.g., simple structure, ease of implementation, and fast convergence speed, DE has been widely applied to some fields of science and engineering, such as cluster analysis [3], robot control [4], controller design [5], and graph theory [6].

It is noteworthy that in DE, the crossover operator depends mainly on the coordinate system and the distribution information of the population is usually unreasonably ignored. Moreover, the crossover operator of DE can be considered as a discrete recombination [7], and thus, the interactions among variables have not been systematically studied. As a result, DE often loses its effectiveness and advantages when solving problems with high variable correlation.

In addition, DE is sensitive to its two main control parameters: the scaling factor F and the crossover control parameter CR. These two control parameters have a significant impact on the performance of DE. Moreover, different control parameter settings show different characteristics [8]. For example, a larger F is effective for global search; however, a smaller F can accelerate the convergence. On the other hand, a larger CR results in higher diversity of the population, since the trial vector will inherit more information from the mutant vector. However, a smaller CR focuses on local exploitation since the target vector will contribute more information to the trial vector. Indeed, it is still an open issue to choose suitable settings of

* Corresponding author.
E-mail addresses: ywang@csu.edu.cn (Y. Wang), mehxli@cityu.edu.hk (H.-X. Li), tingwen.huang@qatar.tamu.edu (T. Huang).

1568-4946/$ – see front matter © 2014 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.asoc.2014.01.038

F and CR to balance the exploration and exploitation of DE during the evolution.

Based on the above considerations, in this paper, we present a novel DE, referred as CoBiDE, including two main components: covariance matrix learning and bimodal distribution parameter setting. In CoBiDE, the covariance matrix learning establishes a coordinate system according to the current population distribution, and then the crossover operator is applied according to the coordinate system thus built to generate the trial vector. Furthermore, in CoBiDE, both F and CR are produced according to a bimodal distribution composed of two Cauchy distributions, the aim of which is to balance the global exploration and the local exploitation during the evolution. CoBiDE has been tested on 25 benchmark test functions developed for the 2005 IEEE Congress on Evolutionary Computation (IEEE CEC2005) [9], as well as a variety of real-world application problems [10]. The experimental results suggest that the performance of CoBiDE is better than that of four other DE variants and three other state-of-the-art EAs.

The remainder of this paper is organized as follows. Section 2 briefly introduces DE and its operators. Section 3 reviews the related work and four main research directions of DE. Then, CoBiDE is presented in Section 4. The experimental results are given in Section 5. Section 6 concludes this paper.

2. Differential evolution (DE)

DE is a population-based heuristic search algorithm. Similar to other EAs, DE contains three basic operators: mutation, crossover, and selection. Firstly, DE produces an initial population by randomly sampling several points (each point is called a target vector) from the search space:

$$P_0 = \{\vec{x}_{i,0} = (x_{i,1,0}, x_{i,2,0}, \ldots, x_{i,D,0}), \quad i = 1, 2, \ldots, NP\} \tag{1}$$

where NP denotes the population size and D denotes the number of variables.

At each generation G, a mutant vector $\vec{v}_{i,G} = (v_{i,1,G}, v_{i,2,G}, \ldots, v_{i,D,G})$ $(i \in 1, 2, \ldots, NP)$ is produced by the mutation operator for each target vector $\vec{x}_{i,G}$. Afterward, the crossover operator is implemented on the mutant vector and the target vector to generate a trial vector $\vec{u}_{i,G} = (u_{i,1,G}, u_{i,2,G}, \ldots, u_{i,D,G})$ $(i \in 1, 2, \ldots, NP)$. The crossover operator and the mutation operator together are called trial vector generation strategy. The selection operator of DE is based on a one-to-one competition between the target vector and the trial vector.

Next, the mutation, crossover, and selection operators are introduced.

2.1. Mutation operator

The commonly used mutation operator can be formulated as follows:

$$\vec{v}_{i,G} = \vec{x}_{r1,G} + F \cdot (\vec{x}_{r2,G} - \vec{x}_{r3,G}) \tag{2}$$

where $r1$, $r2$, and $r3$ are mutually different integers randomly chosen from $[1, NP]$ and also different from i, and F is the scaling factor.

2.2. Crossover operator

The crossover operator combines the mutant vector $\vec{v}_{i,G}$ with the target vector $\vec{x}_{i,G}$ to generate a trial vector $\vec{u}_{i,G}$:

$$u_{i,j,G} = \begin{cases} v_{i,j,G}, & \text{if} \quad rand_j(0,1) \le CR \quad \text{or} \quad j = j_{rand} \\ x_{i,j,G}, & \text{otherwise} \end{cases} \tag{3}$$

where j_{rand} is a random integer between 1 and D, resulting in the trial vector being different from the target vector by at least one dimension, $rand_j(0, 1)$ is a uniformly distributed random number between 0 and 1, and CR is the crossover control parameter.

Based on Eq. (3), it is clear that the trial vector is a vertex of the hyper-rectangle defined by the mutant and target vectors [11]. Moreover, since the information of the trial vector is provided by the mutant vector or the target vector, the crossover operator is dependent on the coordinate system.

2.3. Selection operator

The selection operator of DE adopts a one-to-one competition between the target vector $\vec{x}_{i,G}$ and the trial vector $\vec{u}_{i,G}$. If the objective function value of the trial vector is less than or equal to that of the target vector, then the trial vector will survive into the next generation, otherwise, the target vector will enter the next generation:

$$\vec{x}_{i,G+1} = \begin{cases} \vec{u}_{i,G}, & \text{if} \quad f(\vec{u}_{i,G}) \le f(\vec{x}_{i,G}) \\ \vec{x}_{i,G}, & \text{otherwise} \end{cases} \tag{4}$$

3. The related work

During the past fifteen years, DE has attracted much attention by the researchers [12]. The current studies of DE mainly focus on the following four aspects: (1) improving the trial vector generation strategy, (2) adapting the control parameter setting, (3) hybridizing with other techniques, and (4) integrating multiple trial vector generation strategies with multiple control parameter settings.

3.1. Improving the trial vector generation strategy

Fan and Lampinen [13] proposed a trigonometric mutation operator and embedded it into DE to design a new method called TDE. In TDE, a probability parameter M_t is utilized to balance the trigonometric mutation operator and the original mutation operator of DE. The trigonometric mutation can be considered as a local search operator, which is able to enhance the convergence velocity of DE. The performance of TDE has been evaluated on two test functions and two practical problems.

Zhang and Sanderson [14] presented an improved current-to-best/1 operator, called current-to-$p$best/1, which can be formulated as follows:

$$\vec{v}_{i,G} = \vec{x}_{i,G} + F_i \cdot (\vec{x}^{\,p}_{best,G} - \vec{x}_{i,G}) + F_i \cdot (\vec{x}_{r1,G} - \vec{x}'_{r2,G}), \quad i \in \{1, 2, \ldots, NP\} \tag{5}$$

where $\vec{x}^{\,p}_{best,G}$ is randomly chosen from the best $100p\%$ individuals in the current population, and p is chosen from $(0, 1]$. Moreover, the previously generated offspring, which cannot survive into the next population, have been stored into a predefined archive. The individual $\vec{x}'_{r2,G}$ in Eq. (5) is randomly chosen from the union of the archive and the current population. As analyzed in [14], the advantages of the current-to-$p$best/1 operator are twofold: (1) the information of multiple best individuals can balance the greediness of the mutation and the diversity of the population, and (2) the difference between the recently explored inferior individuals and the current population may represent promising directions toward the global optimum.

Das et al. [15] proposed a neighborhood-based mutation operator, which contains two parts: global neighborhood-based mutation and local neighborhood-based mutation. In the method proposed by Das et al. [15], two trial vectors are produced by the global and local neighborhood-based mutation. Moreover, these two trial vectors are combined to form the actual trial vector by using a weight factor. Clearly, the main aim of the