Applied Soft Computing 48 (2016) 329–346

Contents lists available at ScienceDirect

Applied Soft Computing

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

Utilizing cumulative population distribution information in differential evolution

Yong Wang[a,b,*], Zhi-Zhong Liu[a], Jianbin Li[c], Han-Xiong Li[d,e], Gary G. Yen[f]

[a] School of Information Science and Engineering, Central South University, Changsha 410083, PR China
[b] School of Computer Science and Informatics, De Montfort University, Leicester LE1 9BH, UK
[c] Institute of Information Security and Big Data, Central South University, Changsha 410083, China
[d] Department of Systems Engineering and Engineering Management, City University of Hong Kong, Hong Kong
[e] State Key Laboratory of High Performance Complex Manufacturing, Central South University, Changsha 410083, PR China
[f] School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK 74078, USA

ARTICLE INFO

Article history:
Received 1 October 2015
Received in revised form 30 May 2016
Accepted 5 July 2016
Available online 15 July 2016

Keywords:
Cumulative population distribution information
Differential evolution
Eigen coordinate system
Evolutionary algorithms

ABSTRACT

Differential evolution (DE) is one of the most popular paradigms of evolutionary algorithms. In general, DE does not exploit distribution information provided by the population and, as a result, its search performance is limited. In this paper, cumulative population distribution information of DE has been utilized to establish an Eigen coordinate system by making use of covariance matrix adaptation. The crossover operator of DE implemented in the Eigen coordinate system has the capability to identify the features of the fitness landscape. Furthermore, we propose a *c*umulative *p*opulation distribution *i*nformation based *DE* framework called CPI-DE. In CPI-DE, for each target vector, two trial vectors are generated based on both the original coordinate system and the Eigen coordinate system. Then, the target vector is compared with these two trial vectors and the best one will survive into the next generation. CPI-DE has been applied to two classic versions of DE and three state-of-the-art variants of DE for solving two sets of benchmark test functions, namely, 28 test functions with 30 and 50 dimensions at the 2013 IEEE Congress on Evolutionary Computation, and 30 test functions with 30 and 50 dimensions at the 2014 IEEE Congress on Evolutionary Computation. The experimental results suggest that CPI-DE is an effective framework to enhance the performance of DE.

1. Introduction

Differential Evolution (DE), proposed by Storn and Price [1,2] in 1995, is a very popular evolutionary algorithm (EA) paradigm. During the past two decades, DE has attracted a lot of attention and has been successfully applied to solve a variety of numerical and real-world optimization problems [3–5].

The remarkable advantages of DE are its simple structure and ease of implementation. In DE, each individual in the population is called a target vector. DE contains three basic operators: mutation, crossover and selection. During the evolution, DE generates a trial vector for each target vector through the mutant and crossover operators. Afterward, the trial vector competes with its target vector for survival according to their fitness. DE also involves three control parameters: the population size, the scaling factor, and the crossover control parameter. The performance of DE is dependent mainly on these three operators and three control parameters. In order to further improve the performance of DE, a lot of DE variants have been designed, such as JADE [6], jDE [7], SaDE [8], EPSDE [9], CoDE [10], and so on.

DE is a population-based optimization algorithm; however, population distribution information has not yet been widely utilized in the DE community, which makes DE inefficient especially when solving some optimization problems with complex characteristics. Very recently, two attempts have been made along this line [11,12]. However, the methods proposed in Refs. [11,12] only utilize the distribution information from a single population of one generation, and the cumulative distribution information of the population over the course of evolution has been ignored. Moreover, these methods introduce some extra parameters. Therefore, new insights into the usage of the population distribution information in DE are quite necessary.

* Corresponding author at: School of Information Science and Engineering, Central South University, Changsha 410083, PR China.
E-mail addresses: ywang@csu.edu.cn (Y. Wang), zzliucsu@126.com (Z.-Z. Liu), lijianbin@csu.edu.cn (J. Li), mehxli@cityu.edu.hk (H.-X. Li), gyen@okstate.edu (G.G. Yen).

http://dx.doi.org/10.1016/j.asoc.2016.07.012
1568-4946/

In 2001, Hansen and Ostermeier [13] proposed the well-known covariance matrix adaptation evolution strategy, called CMA-ES. CMA-ES generates offspring by sampling a multivariate normal distribution, which includes three main elements: mean vector of the search distribution, covariance matrix, and step-size. Indeed, covariance matrix reflects the population distribution information to a certain degree [12]. In CMA-ES, the covariance matrix is self-adaptively updated according to the information from the previous and current generations.

In this paper, we make use of the cumulative distribution information of the population to establish an Eigen coordinate system in DE, by considering CMA as an effective tool. Furthermore, we suggest a *c*umulative *p*opulation distribution *i*nformation based *DE* framework called CPI-DE. In CPI-DE, for each target vector, the crossover operator of DE is implemented in both the original coordinate system and the Eigen coordinate system and, as a result, two trial vectors are generated. Subsequently, the target vector is compared with these two trial vectors and the best one will enter the next population. CPI-DE is applied to two classic DE versions as well as three state-of-the-art DE variants. Extensive experiments across two benchmark test sets from the 2013 IEEE Congress on Evolutionary Computation (IEEE CEC2013) [14] and the 2014 IEEE Congress on Evolutionary Computation (IEEE CEC2014) [15] have been implemented to verify the effectiveness of CPI-DE.

The main contributions of this paper can be summarized as follows:

- Due to the fact that single population fails to contain enough information to estimate the covariance matrix reliably, this paper updates the covariance matrix in DE by an adaptation procedure, which makes use of the cumulative distribution information of the population.
- CPI-DE provides a simple yet efficient synergy of two kinds of crossover: the crossover in the Eigen coordinate system and the crossover in the original coordinate system. The former aims at identifying the properties of the fitness landscape and improving the efficiency and effectiveness of DE by producing the offspring toward the promising directions. In addition, the purpose of the latter is to maintain the superiority of the original DE. Moreover, no extra parameters are required in CPI-DE.
- Our experimental studies have shown that CPI-DE is capable of enhancing the performance of several classic DE versions and advanced DE variants.

The rest of this paper is organized as follows. Section 2 describes the basic procedure of DE. Section 3 briefly reviews the recent developments of DE in the last five years. The proposed CPI-DE is presented in Section 4. The experimental results and the performance comparison are given in Section 5. Finally, Section 6 concludes this paper.

2. Differential evolution (DE)

Similar to other EA paradigms, DE starts with a population of *NP* individuals, i.e., $\boldsymbol{P}^{(g)} = \{\vec{x}_i^{(g)} = (x_{i,1}^{(g)}, \ldots, x_{i,D}^{(g)}),\ i = 1, \ldots, NP\}$, where g is the generation number, D is the dimension of the decision space, and NP is the population size. In $\boldsymbol{P}^{(g)}$, each individual is also called a target vector. At $g=0$, the jth decision variable of the ith target vector is initialized as follows:

$$x_{i,j}^{(0)} = L_j + rand(0,1) * (U_j - L_j), i = 1, \ldots, NP, j = 1, \ldots, D \quad (1)$$

where $rand(0,1)$ represents a uniformly distributed random number between 0 and 1, and L_j and U_j are the lower and upper bounds of the jth decision variable, respectively.

After the initialization, DE repeatedly implements three basic operators, i.e., mutation, crossover, and selection, to search for the optimal solution of an optimization problem. Note that in DE, a combination of the mutation operator and the crossover operator is called a trial vector generation strategy.

2.1. Mutation operator

At each generation, a mutant vector is generated for each target vector by the mutation operator. The following are four commonly used mutation operators in the DE community:

- DE/rand/1

$$\vec{v}_i^{(g)} = \vec{x}_{r1}^{(g)} + F * (\vec{x}_{r2}^{(g)} - \vec{x}_{r3}^{(g)}) \quad (2)$$

- DE/rand/2

$$\vec{v}_i^{(g)} = \vec{x}_{r1}^{(g)} + F * (\vec{x}_{r2}^{(g)} - \vec{x}_{r3}^{(g)}) + F * (\vec{x}_{r4}^{(g)} - \vec{x}_{r5}^{(g)}) \quad (3)$$

- DE/current-to-best/1

$$\vec{v}_i^{(g)} = \vec{x}_i^{(g)} + F * (\vec{x}_{best}^{(g)} - \vec{x}_i^{(g)}) + F * (\vec{x}_{r1}^{(g)} - \vec{x}_{r2}^{(g)}) \quad (4)$$

- DE/current-to-rand/1

$$\vec{v}_i^{(g)} = \vec{x}_i^{(g)} + F * (\vec{x}_{r1}^{(g)} - \vec{x}_i^{(g)}) + F * (\vec{x}_{r2}^{(g)} - \vec{x}_{r3}^{(g)}) \quad (5)$$

In the above equations, the indices r_1, r_2, r_3, r_4, and r_5 are distinct integers randomly selected from $[1, \ldots, NP]$ and are also different from i, $\vec{x}_{best}^{(g)}$ is the best target vector in the current population, F is the scaling factor, and $\vec{v}_i^{(g)}$ is the mutant vector.

2.2. Crossover operator

After mutation, the crossover operation is applied to each pair of $\vec{x}_i^{(g)}$ and $\vec{v}_i^{(g)}$ to generate a trial vector $\vec{u}_i^{(g)} = (u_{i,1}^{(g)}, \ldots, u_{i,D}^{(g)})$. The binomial crossover can be expressed as follows:

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

where j_{rand} is a random integer between 1 and D, $rand(0,1)$ is a uniformly distributed random number between 0 and 1, and CR is the crossover control parameter. The condition $j = j_{rand}$ makes the trial vector different from the corresponding target vector by at least one dimension.

2.3. Selection operator

The selection operator of DE adopts a one-to-one competition between the target vector and its trial vector. For a minimization problem, 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:

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

where $f(\bullet)$ is the objective function.

It is evident that NP, F, and CR are three main control parameters of DE. The setting of NP is related to the dimension of the decision space. In general, the higher the dimension of the decision space, the larger the value of NP. In addition, F is always chosen from the