Available online at www.sciencedirect.com
ScienceDirect
European Journal of Operational Research 181 (2007) 1137–1147
EUROPEAN JOURNAL OF OPERATIONAL RESEARCH
www.elsevier.com/locate/ejor

Differential evolution with preferential crossover

M.M. Ali *

School of Computational and Applied Mathematics, Witwatersrand University, Wits-2050, Johannesburg, South Africa

Received 27 October 2004; accepted 16 June 2005
Available online 18 April 2006

Abstract

We study the mutation operation of the differential evolution algorithm. In particular, we study the effect of the scaling parameter of the differential vector in mutation. We derive the probability density function of points generated by mutation and thereby identify some drawbacks of the scaling parameter. We also visualize the drawbacks using simulation. We then propose a crossover rule, called the preferential crossover rule, to reduce the drawbacks. The preferential crossover rule uses points from an auxiliary population set. We also introduce a variable scaling parameter in mutation. Motivations for these changes are provided. A numerical study is carried out using 50 test problems, many of which are inspired by practical applications. Numerical results suggest that the proposed modification reduces the number of function evaluations and cpu time considerably.

Keywords: Global optimization; Differential evolution; Mutation; Crossover; Auxiliary population set; Continuous variable; Probability density function

1. Introduction

The global optimization problem in this paper follows the form:

$$\text{minimize } f(x) \quad \text{subject to } x \in \Omega, \tag{1}$$

where x is a continuous variable vector with domain $\Omega \subset \mathbb{R}^n$, and $f(x)$:$\Omega \mapsto \mathbb{R}$ is a continuous real-valued function. The domain Ω is defined by specifying upper (u^j) and lower (l^j) limits of each component j. We denote the global optimal solution by x^*, with its corresponding global optimal function value $f(x^*)$ or f^* for a short hand notation. The paper is concerned with the differential evolution (DE) algorithm [1]. The DE algorithm is a population algorithm [2] and is purely heuristic. All population direct search methods use a population set S. The initial set

$$S = \{x_1, x_2, \ldots, x_N\} \tag{2}$$

consists of N random points in Ω. A contraction process is then used to drive these points to the vicinity of the global minimizer. The contraction process involves replacing bad point(s) in S with better point(s), per generation. In particular, DE attempts to replace all points in S by new points at each generation. It progresses in an epoch or era base. During each epoch, N new function values are evaluated on N trial points. Trial points are generated using mutation and crossover. A brief description of DE is given below.

* Tel.: +27 11 717 6139; fax: +27 11 717 6149.
E-mail address: mali@cam.wits.ac.za

0377-2217/$ - see front matter
doi:10.1016/j.ejor.2005.06.077

2. A brief description of DE

The DE algorithm attempts to replace each point in S with a new better point. Therefore, in each generation, N competitions are held to determine the members of S for the next generation. The ith ($i = 1, 2, \ldots, N$) competition is held to replace x_i in S. Considering x_i as the target point, a trial point y_i is found from two points (parents), the point x_i, i.e., the target point and the mutated point $\hat{x}_i$ determined by the mutation operation. In its mutation phase, DE randomly selects three distinct points $x_{p(1)}$, $x_{p(2)}$ and $x_{p(3)}$, with replacement, from the current set S. None of these points should coincide with the current target point x_i. The weighted difference of any two points is then added to the third point which can be mathematically described as

$$\hat{x}_i = x_{p(1)} + F(x_{p(2)} - x_{p(3)}), \tag{3}$$

where $F > 0$ is a scaling parameter, and $x_{p(1)}$ is known as the 'base vector'. If the point $\hat{x}_i \notin \Omega$ then the mutation operation is repeated. The trial point y_i is found from its parents x_i and $\hat{x}_i$ using the following crossover rule:

$$y_i^j = \begin{cases} \hat{x}_i^j & \text{if } R^j \leqslant C_R \text{ or } j = I_i, \\ x_i^j & \text{if } R^j > C_R \text{ and } j \neq I_i, \end{cases} \tag{4}$$

where I_i is an integer randomly chosen with replacement from the set I, i.e., $I_i \in I = \{1, 2, \ldots, n\}$; the superscript j represents the jth component of respective vectors; $R^j \in (0, 1)$, drawn uniformly for each j. The ultimate aim of the crossover rule (4) is to obtain the trial point y_i with components coming from the components of the target point x_i and mutated point $\hat{x}_i$. This is ensured by introducing C_R and the set I. Notice that for $C_R = 1$ the point y_i is the replica of the mutated point $\hat{x}_i$. The effect of C_R has been studied in [2,3] and it was found that $C_R = 0.5$ is a good choice. The targeting process continues until all members of S are considered. After all N trial points y_i have been generated, acceptance is applied. In the acceptance phase, the function value at the trial point, $f(y_i)$, is compared to $f(x_i)$, the value at the target point. If $f(y_i) < f(x_i)$ then y_i replaces x_i in S, otherwise, S retains the original x_i. Reproduction (mutation and crossover) and acceptance continue until some stopping conditions are met. It can be seen from (3) that mutation is the core point generation mechanism of DE. This operation calculates the coordinates of new points. The crossover operation (4) chooses the coordinates of a trial point from the known coordinates of two points using a distribution controlled by C_R.

An important issue that needs to be addressed is the value of the scaling parameter F in (3). To the best of our knowledge, no optimal choice of the scaling parameter F has been suggested in the literature of DE. In the original DE [1], F was chosen to lie in $(0, 2]$. Other empirical choices of F were values close to 0.8 [4] and to 1 [2]. In a recent study using 50 test problems the value 0.5 was found to be a good choice [3]. It appears that the choice of F depends upon the problem at hand. Indeed, we observed that a value of F which can be a good choice for a problem but a bad choice for a different problem in the set of 50 problems. Besides, our numerical experiments found that mutation often generates trial points outside the feasible region Ω and the number of points that fall outside Ω varies from problem to problem. We also observed that the larger the F, the higher the number of such points is. On the other hand, the smaller the F, the higher the probability of DE getting trapped in a local minimizer. The choice of F is therefore a delicate issue.

In this paper, we derive the probability density function of the mutated points and show how these points can fall outside Ω. We also visualize this phenomenon by simulation. We then propose an alternative approach that can dispense with the fixed choice of F. Firstly, we replace the fixed scaling parameter value with a variable one and secondly we introduce the preferential crossover rule. We also introduce an auxiliary set of points. These points are normally discarded in DE. The preferential crossover rule uses the auxiliary set in generating trial points within the feasible region.

This paper is divided into seven sections. The next section derives the probability density function (pdf) of the point generation using (3). In Section 4, we present the simulation of mutated points to visualize the effect of F. In Section 5, we present the new algorithm. Results are presented in Section 6 and conclusions are made in Section 7.

We denote the pdf of a random variable (RV), say X by f_X and the joint pdf for RVs, say X and Y by f_{XY}.

3. Probability density of trial points

In this section, we derive the pdf of mutated points generated by (3). We use $F = 1$ for this purpose. The motivation for this choice is as follows. If we take $F > 1$ then the differential vector $(x_{p(2)} - x_{p(3)})$ in (3)