Journal of Computational Science 5 (2014) 298–311

Contents lists available at ScienceDirect

Journal of Computational Science

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

A new genetic algorithm for global optimization of multimodal continuous functions

Manoj Thakur *

School of Basic Sciences, Indian Institute of Technology Mandi, Mandi 175001, India

ARTICLE INFO

Article history:
Received 6 March 2013
Received in revised form 3 May 2013
Accepted 17 May 2013
Available online 6 June 2013

Keywords:
Genetic algorithms
Global optimization
Pareto crossover
Power mutation

ABSTRACT

In this paper a new genetic algorithm is developed to find the near global optimal solution of multimodal nonlinear optimization problems. The algorithm defined makes use of a real encoded crossover and mutation operator. The performance of GA is tested on a set of twenty-seven nonlinear global optimization test problems of variable difficulty level. Results are compared with some well established popular GAs existing in the literature. It is observed that the algorithm defined performs significantly better than the existing ones.

1. Introduction

Many real life problems, when they are represented as a mathematical model, turn out to be optimization problem. Often these optimization problems require the global optimal value of an objective function which depends upon one or more than one decision variables. The problem of finding the global optimal solution of a multimodal function of several variables encountered in many areas such as sciences, engineering, economics and finance etc. (for further details refer Goldberg [1], Michalewicz [2] and Deb [3]). Though, optimization problems in real life scenario involve constraints; we restrict our discussion in this work to unconstrained global optimization problems only. Without loss of generality, an unconstrained nonlinear programming problem (NLPP) may be defined as

$$\text{Min}\, f(x), \quad \text{where} \quad f : R^n \to R,$$

where $x \in S$, and S is an n-dimensional rectangular hypercube in R^n identified by $a_i \le x_i \le b_i$, $i = 1, 2, 3, \ldots, n$. These are often called bounds on the decision variables. A point $x^* \in S$ is called a local minima of f if $f(x^*) \le f(x)$, $\forall x \in N_\varepsilon(x^*) \cap S$, where $N_\varepsilon(x*) = \left\{ x |\, \|x - x*\| < \varepsilon, \varepsilon > 0 \right\}$ is a small neighborhood of the point x^*. If $f(x^*) \le f(x)\, \forall x \in S$ then x^* is said to be the global minima of f.

A variety of populations based probabilistic techniques such as genetic algorithms (GA), simulated annealing (SA), differential evolution (DE), particle swarm optimization (PSO), etc. have been proposed in the literature to solve optimization problems. A thorough description of these algorithms could be found in Deb [3], Kirkpatrick et al. [5], Price et al. [6], Eberhart et al. [7]. Among these algorithms, genetic algorithms have emerged as a powerful class of algorithm to solve optimization problems (Back and Schwefel [8]). Genetic algorithms are robust and efficient search methods based on Darwin's principle of natural selection and genetic inheritance. Since their introduction, GAs has frequently been applied as a search and optimization tool in numerous applications in engineering and sciences (Li and Wang [9]). GAs can be straightforwardly applied to solve multidimensional and multimodal optimization problems seeking global optima. The main attraction of these algorithms is that no extra information such as the continuity, differentiability of objective function and/or constraint is required while solving the problem. These algorithms are well-matched for the problems where the feasible region is disconnected.

Most of the classical optimization techniques are iterative methods in which the initial solution is selected based on the nature of the problem and are revised using deterministic update rules which usually depend upon the problem structure. GA works with a population of candidate solutions (chromosome). Initially, a suitable encoding scheme is chosen which corresponds to the candidate represented in search space and each chromosome is represented by vector of length equivalent to the number of decision variables defining the dimension of the search space. A population of randomly chosen candidate solutions is constructed and objective function values are evaluated corresponding to these solutions. Every solution is assigned a fitness value which is a monotone function of the objective function value. The population of

* Tel.: +91 01905237927.
E-mail address: manojpma@gmail.com

1877-7503/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.jocs.2013.05.005

chromosomes is evolved using three operations: selection, crossover, and mutation.

In earlier studies, the chromosomes were represented by binary strings of 0 and 1. Binary encoding maps the continuous search space in to a discrete one having grids. The distance between two adjacent grids depends upon the length of the binary string used to represent the chromosome. Binary encoding scheme was found to be working well with problem having moderate number of decision variables and problems requiring less precision in the solution but does not perform satisfactorily when the problem is having large dimensional search space or require high accuracy in the solution. The problems with large number of variables may be solved better by increasing population size and high precision may be achieved by increasing the string length. In later studies Goldberg [10] found that the computation time increases exponentially as the string size increases. Some other modifications to accelerate the convergence binary genetic algorithms were reported in Jing and Yang [11].

In early nineties the concept of real coded GA was introduced in which instead of binary strings, vector of real-valued genes were used to represent a chromosome (Davis [12], Wright [13], Janikow and Michalewicz [14], Michalewicz [2], Deb and Agrawal [15], Herrera and Lozano [16], Herrera et al. [17,18], Tsoulos [19]). This encoding scheme appears naturally suited for problems having continuous variables. Like binary coded GA, real coded GA also uses similar genetic operators: selection, crossover, and mutation. The real encoding is advantageous over the earlier used binary encoding in the sense that it gets better customized to optimization of problems in continuous domain. Though real encoded GA over comes many difficulties of binary encoded GA, it still suffers from problems like premature convergence to suboptimal solution and also slow convergence. These situations arise due to the lack of population diversity and inability of GA to exploit locally the information of solutions in the population (Yuan et al. [20]). Considerable amount of work have been done to enhance the performance of GA by improving exploration and exploitation potential. The exploration capabilities of a GA primarily depend upon the crossover operator used because it utilizes the information about individuals in current population and directs the search in other promising region of the search space. This is one of the major reasons researchers being paying more attention on designing effective real encoding based crossover operators.

An important class of crossover operators for real encoding based chromosomes is parent centric crossover operators. Deb et al. [21] reported that parent centric crossover operators offer quite effective and important means of solving real-parameter optimization problems. In the present work a new parent centric crossover operator based on Double Pareto distribution is described. A new generational GAs is proposed which make use of this crossover operator and its performance is compared with seven existing GAs proposed in Deep and Thakur [22,23], Deb [4], Meittinen et al. [24] and Maaranen et al. [25]. A pair wise comparison of the proposed GA with the existing GAs is done and then all eight algorithms are compared simultaneously.

The paper is ordered as follows: a brief review of existing real coded crossover operators is given in Section 2. The proposed double Pareto crossover is defined in Section 3. The operators used in other genetic algorithms considered for the comparison are explained in Section 4. The description of the new genetic algorithms based on double Pareto crossover and other seven genetic algorithms are given in Section 5. In Section 6, the experimental setup and parameter settings of all the algorithms is described. Performance of various algorithms and their comparative study is shown in Section 7. The conclusions drawn based on current study are presented in Section 8. In Appendix A the formulations of all test problems and the performance index used for comparing the algorithms is explained.

2. Real coded crossover operators

A variety of real coded crossover operators have been proposed in the GA literature. In simple crossover (Michalewicz [26], Wright [13]), a gene position is randomly selected from a pair of parents and two new offspring solutions are made by simply exchanging the genes. In flat crossover (Radcliffe [27]) an offspring is produced using uniform distribution between the genes of the two parents. Muhlebein and Schlierkamp-Voosen [28] suggested extended line crossover and extended intermediate crossover. Both crossovers enhance the search capabilities of flat crossover by allowing exploration in a fixed interval beyond the parents. Eshelman and Schaffer [29] generalized the idea of Radcliffe [27] and Muhlebein and Schlierkamp-Voosen [28] by introducing the blend crossover. This crossover has a user defined parameter α which allows exploring the interval not only in between the parent genes but also on the interval extending equally on either side of the parents. It is also noted that extended intermediate crossover is a special case of blend-crossover for $\alpha = 0.25$. Wright [13] proposed heuristic crossover which make use of fitness function value of the parent solutions to generate a child solution from a pair of parents. The solution produced is biased in a linear manner in the direction of the relatively fit parent. Michalewicz [30] presented a class of arithmetical crossover which makes use of the idea of creating children sandwiched between the parents. In single arithmetical crossover a gene is picked randomly and an offspring is produced using uniform distribution between the genes. Another offspring is generated which is mirror image of the earlier generated offspring with respect to the mean of the parents. In simple arithmetic crossover a gene is randomly picked and the method used in single arithmetic crossover is used to all genes after this point. In whole arithmetic crossover the procedure used in single arithmetical crossover is applied to all genes of the parents. Voigt et al. [31,32] presented a fuzzy min–max and fuzzy recombination operators. Herrera and Lozano [16] proposed two types of dynamic heuristic fuzzy connectives based crossovers. Dynamic fuzzy connective based crossovers were designed to keep appropriate succession between exploration and exploitation during the evolution process. Heuristic fuzzy connective based crossovers were intended to explore the region near the best parents in order to maintain the diversity and keeping convergence speed up in an advantageous manner. Tsutsui et al. [33] proposed a simplex crossover which is a multi-parent crossover operator and produces offspring solution uniformly distributed over the simplex formed by parent solutions. Simulated binary crossover (Deb and Agrawal [15]) replicates the functioning of single-point crossover on a string of binary alphabets in continuous domain. Tutkun [34] proposed a crossover operator based on Gaussian distribution. Kaelo and Ali [35] suggested integration of different crossover rules in the genetic algorithm and recommended some modifications in applying the crossover rules and localization of searches in their study. Deep and Thakur [22] presented Laplace crossover operator which generates a pair of offspring solution from a pair of parent solutions using Laplace distribution.

Generally the crossover operator is applied on a pairs of parent solutions to produce two children solutions. However, crossover operators with multiple descendants have also been proposed in the literature. Linear crossover operator creates three offspring solutions using two parents (Wright [13]). Ono and Kobayashi [36] introduced unimodal normally distributed crossover operator (UNDX) in which three parents participate in the crossover process and two or more offspring are produced. Later Ono et al. [37] enhanced the performance of UNDX by incorporating uniform crossover (UX). A generalized multi-parent UNDX operator having more than three parents was reported in Kita et al. [38]. A parent centric crossover (PCX) operator which is a multi-parent crossover