ARTICLE IN PRESS

Engineering Science and Technology, an International Journal xxx (2017) xxx–xxx

HOSTED BY

ELSEVIER

Contents lists available at ScienceDirect

Engineering Science and Technology, an International Journal

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

JESTECH

Full Length Article

Optimization of turning process using Amended Differential Evolution Algorithm

Parthiv B. Rana *, D.I. Lalwani

Mechanical Engineering Department, Sardar Vallabhbhai National Institute of Technology (SVNIT), Ichchhanath, Surat, Gujarat 395 007, India

ARTICLE INFO

Article history:
Received 4 April 2017
Revised 19 August 2017
Accepted 23 August 2017
Available online xxxx

1. Introduction

Most of the real-life optimization problems in the area of engineering, science and economics are distinct in nature and can be classified as continuous or discrete, linear or non-linear, constrained or unconstrained, single objective or multi-objective problems. In addition, an optimization problem becomes more complex when number of constraints are more, and objective function and constraint functions are non-linear. The complex problems that are not differentiable and difficult to solve using classical optimization methods are solved by Evolutionary Algorithms (EAs). Most of the EAs have certain characteristics, which are based on naturally inspired behavior, such as biological, molecular, swarm insects and neurobiological systems, for example, Genetic Algorithm (GA), Artificial Bee Colony (ABC), Differential Evolution (DE), Artificial Immune Algorithm (AIA), Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), Biogeography Based Optimization (BBO), Shuffle Frog Leaping Algorithm (SFLA), Fuzzy optimization system, Heat Transfer Search (HTS), etc. EAs are population based iterative algorithms where set of solutions is obtained from an entire search space. The next solutions from the current population largely depend on two important characteristics, i.e., exploration and exploitation of a search space. In exploration, there are more chances of getting a new solution by moving away from current solution using large increment in current solution. In exploitation, there are more chances of getting better solutions by moving a little from a current solution using a small increment in current solution [1]. Therefore, performance of any EA depends on appropriate balance of exploration and exploitation of a search space. In most of the EAs, selection of control parameters plays an important role for appropriate balancing of exploration and exploitation of a search space.

Differential Evolution (DE) is one of the robust algorithms for stochastic real-parameter optimization [15] and it is proposed by Storn and Price in 1997 [2]. As stated in literature, DE has capability to solve variety of optimization problems and it has been widely applied for function optimization of unconstrained problems as well as constrained problems (using suitable constrained handling method) [3–9] and multi-objective problems [10]. There are three operations in DE, namely, mutation, crossover and selection. The performance of DE is sensitive to control parameters, namely, scale factor (F) and crossover rate (CR) where F is associated with mutation operation and CR is associated with crossover operation. Further, the performance of DE is influenced by mutation strategy and population size (PS) [11,12]. Therefore, appropriate selection of control parameters, mutation strategy and population size is important for better performance of DE. In the past, different types of optimization problems are solved by many researchers using DE. Further, attempts have been made to improve the performance by modifying the original DE. Mallipeddi et al. [6] proposed EPSDE algorithm that consists of ensemble of mutation strategy and associated control parameters to obtain competitive offspring. Zou et al. [9] proposed a Novel Modified DE (NMDE) algorithm that selects the control parameters, viz., scale factor and crossover rate, using adaptive strategy to solve constrained optimization problems. Gong and Cai [10] proposed an improved DE algorithm for Multi-objective Optimization Problems (MOPs) that is associated with several features of previously proposed EAs in a unique way. Noman and Iba [13] developed an Accelerating Differential Evolution that consists of local search technique to improve performance of DE. Rahnamayan et al. [14] proposed an Opposition-based DE (ODE) in which population size is initiated using Opposition-Based learning and the convergence rate of DE is enhanced using opposite numbers. Qin et al. [15] proposed Self-adaptive DE (SaDE) where generation of two trial vector

* Corresponding author.
E-mail addresses: ranaparthiv@gmail.com (P.B. Rana), dil@med.svnit.ac.in (D.I. Lalwani).

Peer review under responsibility of Karabuk University.

http://dx.doi.org/10.1016/j.jestch.2017.08.006
2215-0986/

Please cite this article in press as: P.B. Rana, D.I. Lalwani, Optimization of turning process using Amended Differential Evolution Algorithm, Eng. Sci. Tech., Int. J. (2017), http://dx.doi.org/10.1016/j.jestch.2017.08.006

schemes and associated control parameters are gradually self-adapted based on the previous optimum solution. Zhang and Sanderson [16] proposed JADE that is a modified version of DE. In JADE, control parameters are dynamically modified and a new mutation strategy, 'DE/current-to-*p*best/1', is implemented to enhance the performance of DE. Takahama and Sakai [17] proposed ε-constrained Rank-based DE in which control parameters are varied for each member of population on the basis of Rank (*R*). The Rank is related to the quality of a solution (i.e., *R* is 1 for the best solution and *R* is equal to population size (*PS*) for the worst solution). Xiang et al. [18] proposed an Enhanced DE (EDE) algorithm to prevent premature convergence and local trapping of mutation strategy, i.e., 'DE/best/1'; further, they guessed population size using opposition based learning method, and proposed new integrated mutation strategies (DE/current/1 and DE/*p*best/1) to accelerate DE algorithm. Gong et al. [19] proposed two adaptive strategies for DE algorithm, namely, 'Probability Matching' and 'Adaptive Pursuit' where the advantage of recent effect on the optimization process is considered to select the most appropriate strategy for solution. Salehpour et al. [20] tuned the scale factor (*F*) of DE using fuzzy-logic inference system to improve the exploration and exploitation of a search space. Gogoi et al. [43] used the simplex search and differential evolution based inversed methods to evaluate the process parameters of reheat regenerative power cycle where they found that DE based inversed method provides sufficient scope to select the appropriate set of process parameters to meet the particular power requirement.

Many researchers proposed the parameter adaptation techniques for scale factor and crossover rate to improve the performance using various mutation strategies. In the present work, some modifications are proposed in original DE as listed below. (i) Concept of Design of Experiments (DoE) is used to generate initial quality random population. (ii) Modification in the DE/rand/1 mutation strategy is suggested. (iii) The range of control parameters (*F* and *CR* is selected on the basis of Rank (*R*) for each individual of population and (iv) New condition for selection vector is used. Further, the above mentioned modifications are incorporated in DE and modified DE is named as Amended Differential Evolution Algorithm (ADEA). Moreover, 'Σ-constrained handling method' is proposed and applied with ADEA. The performance of ADEA is evaluated on some real-life optimization problems of turning process.

The rest of the paper is arranged as follow: Section 2 presents basics of Differential Evaluation (DE) algorithm and various mutation strategies proposed till now. Section 3 discusses Amended Differential Evolution Algorithm (ADEA). Section 4 discusses optimization of turning process using ADEA. Section 5 presents results and discussion, and Section 6 concludes the work.

2. Differential Evolution (DE) Algorithm

DE is a population based competitive stochastic search algorithm for global optimization over continuous space that is developed by Storn and Price [2]. There are basically four steps, i.e., initialization, mutation, crossover and selection in DE and they are explained in subsequent sub-sections.

2.1. Initialization

In this step, population Size (*PS*), number process parameters or decision variables (*D*) and bounds (lower and upper) of process parameters are inputs. Population Size (*PS*) is selected by a user and population is generated randomly. The population is a matrix and its size is $PS \times D$ where each row of the population matrix is known as a population vector or a target vector, X_i ($i = 1$ to *PS*) and elements or process parameters of the population vector (X_i) are denoted as x_{ij} (for every i^{th} population vector, *j* is varied from 1 to *D*). Each population vector has value of process parameters within a lower (l_b) and upper (u_b) bound and generated randomly using Eq. (2.1)

$$x_{i,j} = l_{bj} + (u_{bj} - l_{bj}).rand \tag{2.1}$$

where *rand* is a MATLAB function that generates uniformly distributed random value between 0 and 1, *G* represents the generation (iteration) number, and l_b and u_b are the lower and upper bounds for the j^{th} process parameter, respectively. The optimization algorithm has three main operations, namely, (i) mutation, (ii) crossover and (iii) selection. In each generation (*G*), current vectors of population become target vectors.

2.2. Mutation

The first operation is mutation operation and it is carried out after the generation of initial random population. The elements of population vector are changed by mathematical operation to generate a new offspring, i.e., mutant vector ($M_i^G = [m_{i,1}^G, m_{i,2}^G, \ldots, m_{i,D}^G]$). Some mutation strategies proposed by various researchers are given in Eqs. (2.2) to (2.8) to obtain mutant vector from population vectors.

1. DE/rand/1 [21]:

$$M_i^G = X_a^G + F \cdot (X_b^G - X_c^G) \tag{2.2}$$

2. DE/rand/2 [15]:

$$M_i^G = X_a^G + F \cdot (X_b^G - X_c^G) + F \cdot (X_d^G - X_e^G) \tag{2.3}$$

3. DE/best/1 [21]:

$$M_i^G = X_{best}^G + F \cdot (X_a^G - X_b^G) \tag{2.4}$$

4. DE/best/2 [21]:

$$M_i^G = X_{best}^G + F \cdot (X_a^G - X_b^G) + F \cdot (X_c^G - X_d^G) \tag{2.5}$$

5. DE/rand-to-best/1 [15] or DE/target-to-best/1 [22]:

$$M_i^G = X_i^G + K \cdot (X_{best}^G - X_i^G) + F \cdot (X_a^G - X_b^G - X_c^G - X_d^G) \tag{2.6}$$

6. DE/rand-to-best/2 [21] or DE/target-to-best/2 [22]:

$$M_i^G = X_i^G + K \cdot (X_{best}^G - X_i^G) + F \cdot (X_a^G - X_b^G) \tag{2.7}$$

7. DE/current-to-rand/1 [23]:

$$M_i^G = X_i^G + K \cdot (X_a^G - X_i^G) + F \cdot (X_b^G - X_c^G) \tag{2.8}$$

where *a*, *b*, *c*, *d* and *e* are the indices of randomly selected population vectors from the population. They are mutually different from each other and also different from the running index (*i*) of population. *F* is mutation scale factor (selected by a user) that scales the differential variations (exploration characteristic) [2]. X_{best}^G is the best population vector that has best fitness value from the population of current generation *G*. *K* is random number generated between 0 and 1.

2.3. Crossover

Crossover operation controls which and how many process parameters (elements) are to be replaced with other. A crossover vector or trial vector (C_i^G) is obtained by replacing process parameters of target vector (X_i^G) with a mutant vector (M_i^G) using Eq. (2.9).