Applied Soft Computing 33 (2015) 304–327

Contents lists available at ScienceDirect

Applied Soft Computing

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

Multi-population differential evolution with balanced ensemble of mutation strategies for large-scale global optimization

Mostafa Z. Ali [a,*], Noor H. Awad [b], Ponnuthurai N. Suganthan [c]

[a] Computer Information Systems Department, Jordan University of Science & Technology, Irbid, Jordan
[b] Computer Engineering Department, Jordan University of Science and Technology, Irbid, Jordan
[c] School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore

ARTICLE INFO

Article history:
Received 2 May 2014
Received in revised form
15 December 2014
Accepted 6 April 2015
Available online 30 April 2015

Keywords:
Differential evolution
Large-scale optimization
DE mutation strategies
Scalability

ABSTRACT

Differential evolution (DE) is a simple, yet very effective, population-based search technique. However, it is challenging to maintain a balance between exploration and exploitation behaviors of the DE algorithm. In this paper, we boost the population diversity while preserving simplicity by introducing a multi-population DE to solve large-scale global optimization problems. In the proposed algorithm, called *m*DE-*b*ES, the population is divided into independent subgroups, each with different mutation and update strategies. A novel mutation strategy that uses information from either the best individual or a randomly selected one is used to produce quality solutions to balance exploration and exploitation. Selection of individuals for some of the tested mutation strategies utilizes fitness-based ranks of these individuals. Function evaluations are divided into epochs. At the end of each epoch, individuals between the subgroups are exchanged to facilitate information exchange at a slow pace. The performance of the algorithm is evaluated on a set of 19 large-scale continuous optimization problems. A comparative study is carried out with other state-of-the-art optimization techniques. The results show that *m*DE-*b*ES has a competitive performance and scalability behavior compared to the contestant algorithms.

© 2015 Elsevier B.V. All rights reserved.

1. Introduction

In the last few decades, various optimization algorithms have been developed for solving many optimization problems. These algorithms include evolutionary algorithms (EAs) [1], simulated annealing (SA) [2], differential evolution (DE) [3], ant colony optimization (ACO) [4], particle swarm optimization (PSO) [5] and other traditional methods like quasi-Newton [6], Tabu search [7] and Nelder–Mead's simplex method [8]. Although these algorithms have shown good performance when solving problems involving small to moderate dimensions, many of them suffer from the curse of dimensionality problem whereby the algorithm's performance degrades with increasing dimensionality. The reason is that the complexity of an optimization problem increases exponentially with its dimensionality, especially for continuous spaces. As a result, the interest in developing efficient large-scale evolutionary algorithms has been increasing.

Differential evolution (DE) was proposed by Storn and Price [3], and is a powerful evolutionary algorithm that has been used for solving different types of optimization problems [3,9–14]. The population space is represented as chromosomes, and each decision variable is encoded by a real value. In DE, offspring is generated by perturbing the solutions with a scaled difference of selected population individuals followed by a crossover strategy. Subsequently, DE employs parent-offspring comparison in which the parent is replaced if the offspring is not worse based on fitness values. The DE has several mutation and crossover schemes, and control parameters such as population size (*NP*), mutation scale factor (*F*) and crossover rate (*CR*). The performance of classical DE is highly dependent on the choice of the strategies and their associated control parameter values. An inappropriate setting of these for a particular problem may lead to premature convergence and degrades the DE algorithm's performance. Therefore, in order to apply DE successfully to solve optimization problems, a trial and error search for the strategies and the associated control values is usually required. On the other hand, some works have been done to adaptively select and tune them. For example, self-adaptive differential evolution (SaDE) [15,16] and self-adaptive differential evolution with neighborhood search (SaNSDE) [17] use multiple strategies adaptively based on the success during previous generations with adaptive values of *F* and *CR* using different schemes. jDE algorithm [18]

* Corresponding author. Tel.: +962 79 0036901.
E-mail address: mzali@just.edu.jo (M.Z. Ali).

http://dx.doi.org/10.1016/j.asoc.2015.04.019
1568-4946/© 2015 Elsevier B.V. All rights reserved.

uses *F* and *CR* values that are assigned to all individuals in the population and these values are updated based on some heuristic rules. JADE [19] introduced a new mutation scheme that utilizes the best solution with an optional external archive and updating schemes for adapting control parameters. Self-adaptive DE techniques outperform the classical DE algorithms without adaptive control [20–22].

In this paper, we propose a multi-population DE technique, called *m*DE-*b*ES which seeks to enhance population diversity, and balance the search between exploration and exploitation. The algorithm first divides the population randomly into sub-populations. The algorithm then generates trial vectors for each sub-population using a different set of strategies. A new mutation strategy is used to support the process of generating quality solutions that use information either from the best or a randomly selected individual probabilistically. This mutation strategy uses a linear combination of vectors to produce a base vector. All these strategies choose individuals based on their fitness-based ranks when producing mutant vectors. The algorithm allows each subgroup to evolve separately for a certain number of generations. Then some individuals are randomly selected based on a certain neighborhood topology to migrate between subgroups to share experiences. Dividing the population into multiple subgroups allows us to observe the decision space from various perspectives. Thus, reduce stagnations risks through imposing high exploitation pressure by each subpopulation. However, using such exploitative power proves to be very promising at the beginning and then results in a premature convergence at later stages because it reduces the diversity. This is due to the incapability of enhancing over the best individuals resulted from excessive exploitation. Information sharing that is presented in the form of migrating individuals via the constructed topologies constitutes a promising direction that preserves diversity during later stages of the search. This idea is powered by employing different mutations strategies and control parameters for these groups that support the production of new search moves that promote the detection of promising regions. Hence, mitigate the risk of lower diversity and enhance the overall algorithmic performance. The performance of the algorithm is evaluated on a set of challenging large-scale global optimization (LSGO) benchmark functions with dimensions ranging from 50 to 1000. The results show that the proposed technique is capable of solving continuous large-scale problems effectively and producing competitive results when compared with the state-of-the-art algorithms that are designed specifically to solve this type of problems.

The rest of the paper is structured as follows. Section 2 presents a brief review of related works on large-scale optimization. In Section 3, the classical DE algorithm is briefly introduced. The proposed algorithm (*m*DE-*b*EM) and its implementation are presented in Section 4. Section 5 presents experimental set-up. Experimental results and analysis are presented in Section 6. Finally, the work is concluded in Section 7.

2. Related Work

The DE algorithm has become a powerful optimizer for solving global continuous optimization problems [23]. Researchers originally used the algorithm with fixed control parameter values for *F*, *CR* and *NP* and this may lead to poor performance when applied to solve high-dimensional problems. In order to enhance the DE algorithm's performance, many improved versions have been proposed in the literature. They can be categorized into three main groups. The first group develops self-adaptive techniques capable of choosing appropriate mutation strategies and/or suitable population topologies and/or the best values for the control parameters [24–27]. The second group is to merge the DE technique with other types of evolutionary algorithms or local search [28–30]. The third group is a hybridization of the first two groups [31,32]. In this section, we introduce some recent DE-based techniques from these categories as well as other state-of-the-art algorithms for optimizing large-scale continuous problems.

Seven DE-based algorithms [24–26,31,33–35] among thirteen were published in the special issue on scalability of evolutionary and other metaheuristic algorithms for large-scale continuous optimization problems. This is an indication of the usefulness of the DE algorithm for solving LSGO problems. In [24], the SOUPDE technique was proposed as a structured multi-population technique that used two simple mechanisms, shuffling and updating. The shuffling is rearranging the individuals randomly over the sub-populations and updating is assigning different values of the scale factor for the sub-populations. For each sub-population, the DE/rand/1 is performed with exponential crossover. The use of multi-populations will drastically reduce the fitness value in highly multivariate fitness landscapes. However, the performance of the algorithm is not stable at all categories of problems from the selected benchmark, especially the hybrid-composition functions, which means diversity of the populations should be enhanced further during the search. The authors in [33] proposed role differentiation mechanism and malleable mating to approach LSGO problems. They rely on the concept of role-formulation for vectors as either receiving or leading or correcting or placing vectors. The authors used two different ways for selecting vectors for crossover and mutation operations. The first mechanism is responsible for defining the attributes based on which the vectors are selected and the second mechanism makes it easy to place vectors to adapt their mating trends. The purpose is to ensure similarity relationships with the correcting and leading vectors. They used DE/rand/1 as a mutation strategy with exponential crossover with fixed values for *F, CR* and *NP*. Although this algorithm enhanced the way DE works with linear algorithmic complexity, its performance deteriorated in the last two categories of problems and had a lower performance compared to its peer algorithms in the competition, as reported in [36]. Adaptability of some of the main operators like the control operator would have enhanced the performance further.

Generalized opposition-based learning (GODE) has been used to improve the performance of the DE to solve large-scale optimization problems in [25]. The idea was to generate a solution with its opposite simultaneously to provide another chance of finding a candidate solution closer to the global optimum. They used DE/rand/1 with exponential crossover with fixed values for *F*, *CR* and *NP* and a pre-specified probability to use opposition-based strategies. This implementation was not able to solve functions F_7 and F_{15}. Moreover, the algorithm was not effective for solving most of the problems in the last two categories of problems (shifted and hybrid composition functions). On the other hand, the GaDE technique in [26] proposed a generalized parameter adaptation scheme using a probability distribution. They used two mutation strategies: DE/rand/1 and DE/current-to-best/1 with adaptive Cauchy and Gaussian distributions for selecting values of *F* and *CR* respectively. Adaptation of the parameters improved the performance over the canonical DE. Performance of the algorithm can be enhanced by combining with other strong optimizers like G-CMA-ES and applying algorithm portfolio techniques between them to enhance the obtained results and reduce the average mean error of the run.

In [34], the authors introduced jDElscop technique in which they used three mutation strategies and a population size reduction mechanism. They used DE/rand/1 with binomial and exponential crossovers and DE/best/1 with binomial crossover in which one strategy is activated in each generation. Moreover, they used self-adaptive control parameters for each strategy. The algorithm was able to obtain very competitive results compared to many other

Download English Version:

https://daneshyari.com/en/article/6905255

Download Persian Version:

https://daneshyari.com/article/6905255

Daneshyari.com