



Some metaheuristics should be simplified



Adam P. Piotrowski*, Jaroslaw J. Napiorkowski

Institute of Geophysics, Polish Academy of Sciences, Ks. Janusza 64, 01-452 Warsaw, Poland

ARTICLE INFO

Article history:

Received 15 March 2017

Revised 1 August 2017

Accepted 15 October 2017

Available online 16 October 2017

Keywords:

Differential evolution

Evolutionary algorithms

Swarm intelligence

Structural bias

CMA-ES

Complexity

ABSTRACT

Users have a large and constantly increasing number of optimization metaheuristics at their disposal. In pursuit of ever better approaches, popular and successful methods are often improved a number of times in a row, or various methods are hybridized. One should, however, pay attention to whether such deliberately improved or hybridized methods do not become unnecessarily complicated, or if some of their elements do not introduce a structural bias. In the first case the algorithm should be simplified by eliminating the unneeded elements. This will make it more clear to the users, and, as shown in this study, may even improve the results. In the second case, operators that are responsible for over-frequent sampling of some part of the search space have to be removed.

In this study we address the problem of over-complexity of metaheuristics, focusing on two joint winners of the CEC2016 competition on single-objective numerical optimization, L-SHADE-EpSin and UMOEA-II algorithms. It is shown that each of them includes a procedure which introduces a structural bias by attracting population towards the origin $\mathbf{0}$. As discussed in the text, in the case of seven out of 30 benchmark problems considered in the CEC2016 competition the objective function values in the origin $\mathbf{0}$ are better than those found by the vast majority of algorithms, hence structural bias affects the results obtained by L-SHADE-EpSin and UMOEA-II. In this work we simplify both algorithms by removing operators that are the main cause of structural bias. Further, we show that in the L-SHADE-EpSin algorithm the Ensemble Sinusoidal adaptation mechanism of control parameter F , that is used during the first half of the search, is not needed. Slightly better results on both artificial benchmarks used in the CEC2016 competition and on a wide set of real-world problems may be obtained if, during that period of the search, the value of F is simply set to 0.5. Such a simplified algorithm is competitive against a large number of metaheuristics and may be much easier for the users to understand.

© 2017 Elsevier Inc. All rights reserved.

1. Introduction

When developing new population-based metaheuristics one may either start from scratch, attempt to improve an already known method, or hybridize properties of various approaches [4,22]. As more and more metaheuristics are proposed, the first approach becomes less likely to be successful, and may easily lead to reduplication of already existing algorithms under a novel name [44]. Indeed, in a number of recent papers researchers expressed concerns whether we need yet another apparently “novel” types of metaheuristics [13,14,44,54], especially since publication of the proof of No Free Lunch Theorems [55]. Hybridization of existing methods or improvement of the selected one by adding new elements are often

* Corresponding author.

E-mail address: adamp@igf.edu.pl (A.P. Piotrowski).

simpler, have a higher chance of success and better position investigators' work in the context of current research. However, opinions on such approaches are highly diversified; they have been referred to as “adding stuff” in [22], but also praised as “excellent evolution” in [16]. Putting aside such slightly subjective comments, we must admit that adding something new to the already existing algorithms is almost always performed without an in-depth analysis of whether all older operators or procedures are still needed if coupled with the newer ones. Note that discussing the main idea, or philosophy behind the proposed novelty, like for example in [23], is almost never sufficient. As a result, after a few such consecutive additions the novel methods frequently become unnecessarily over-complicated. Both justification and supposed cooperation between their operators may become unclear even to the authors. Some elements of such constructed algorithms may affect the clarity of the method discouraging its potential users, or lead to mistakes during implementation [3]. What is even worse, the more complicated algorithm, the higher the risk that some of its elements introduce structural bias [25] that leads to artificially sampling some part of the search space more frequently than the others.

As an example imagine that, in a specific procedure, the locations of the current individuals along the coordinates in the search space are multiplied by 0.5. In such a trivial example the population of structurally biased algorithm will be attracted towards the origin $\mathbf{0} = [0^1, 0^2, \dots, 0^D]$ (where D is the problem dimensionality) without any reason justified by the fitness landscape. This may artificially help finding good solutions if they are close to the origin or negatively affect the search by wasting the function calls if the solutions in and around the origin are poor. Using an empirical approach proposed in [25], in [35] it was shown that a weak structural bias is present in a large number of metaheuristics (although not in all of them), but its causes are often complicated and hard to find. However, some algorithms are very strongly biased. If the sources of their bias can be determined, they should be removed.

In the present study we address both the problem of structural bias and unnecessary complication of metaheuristics. As an example we use two algorithms that have been declared joint winners of the CEC2016 competition on single-objective numerical optimization, namely L-SHADE-EpSin [1] and UMOEA-II [12]. To facilitate the discussion, our key results are organized into three main sections (2-4).

First, in Section 2 we briefly discuss L-SHADE-EpSin and UMOEA-II algorithms and trace the history of their developments.

In Section 3 we address the problem of structural bias. We show that specific operators of both L-SHADE-EpSin and UMOEA-II are structurally biased and under specific circumstances lead the population “artificially” towards the origin $\mathbf{0}$. The presence of structurally biased operators in both algorithms affect the results of the CEC2016 competition, as in seven out of 30 benchmark minimization problems used in CEC2016 the values of objective function in the neighbourhood of the origin are better than the values that are found by almost all unbiased algorithms. We removed such structurally biased parts of L-SHADE-EpSin and UMOEA-II, and report the results obtained.

Then, in Section 4 we show that the main novelty of the L-SHADE-EpSin algorithm, a complicated mechanism of Ensemble Sinusoidal adaptation of control parameter F that is applied during the first half of the search, is not needed, and that slightly better results, both on CEC2014 benchmarks [27] (that were used during the CEC2016 competition) and on 22 real-world problems from various fields of science [6] may be obtained by setting F to 0.5 for the first half of the search. Hence, the only successful novelty of L-SHADE-EpSin compared with L-SHADE proposed in 2014 by Tanabe and Fukunaga [47] is that in L-SHADE-EpSin the parameter F is managed differently during the first and the second half of the search.

Finally, comparison between such simplified algorithms and various kinds of metaheuristics is given in Section 5. Conclusions are summarized in Section 6.

2. Step-by-step development of L-SHADE-EpSin and UMOEA-II

In Section 2.1 we discuss the complicated history of the development of L-SHADE-EpSin algorithm, and define its specific features. In Section 2.2 we briefly discuss the UMOEA-II algorithm and point to its operators that introduce structural bias.

2.1. L-SHADE-EpSin

The L-SHADE-EpSin algorithm [1] is a good example of a method that has been developed by step-by-step improvements of the previous variants, starting from the Differential Evolution algorithm invented by Storn and Price in 1990's [45]. To discuss the details of L-SHADE-EpSin and understand its features, we should start from its predecessors.

2.1.1. Differential evolution

Without the loss of generality we consider numerical minimization problems by searching for the global optimum solution \mathbf{x}^* such that

$$f(\mathbf{x}^*) = \min_{\mathbf{x} \in \Omega} f(\mathbf{x}) \quad (1)$$

where $f(\mathbf{x})$ is the real-valued function, \mathbf{x} is a D -dimensional vector, $\mathbf{x} = \{x^1, \dots, x^D\}$ with the domain $\Omega \subseteq \mathbf{R}^D$. L-SHADE-EpSin is based on L-SHADE [47], that itself is a much modified version of population-based Differential Evolution (DE) algorithm [45], the starting point for our discussion. In DE, in generation $g=0$, NP individuals (solution vectors) $\mathbf{x}_{i,g} = \{x_{i,g}^1, \dots, x_{i,g}^D\}$, $i=1, \dots, NP$ are randomly initialized from the uniform distribution

$$x_{i,0}^j = L^j + \text{rand}_i^j(0, 1) \cdot (U^j - L^j); \quad j = 1, \dots, D; \quad i = 1, \dots, NP \quad (2)$$

Download English Version:

<https://daneshyari.com/en/article/6856996>

Download Persian Version:

<https://daneshyari.com/article/6856996>

[Daneshyari.com](https://daneshyari.com)