Information Sciences 373 (2016) 124–148

Contents lists available at ScienceDirect

Information Sciences

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

May the same numerical optimizer be used when searching either for the best or for the worst solution to a real-world problem?

Adam P. Piotrowski[a,*], Maciej J. Napiorkowski[b]

[a] Institute of Geophysics, Polish Academy of Sciences, Ks. Janusza 64, 01-452 Warsaw, Poland
[b] Faculty of Building Services, Hydro and Environmental Engineering, Warsaw University of Technology, Nowowiejska 20, 00-653 Warsaw, Poland

A R T I C L E I N F O

Article history:
Received 23 October 2015
Revised 12 August 2016
Accepted 15 August 2016
Available online 17 August 2016

Keywords:
Metaheuristic
Continuous real-world problem
No Free Lunch
Genetic algorithm
Differential Evolution
Particle Swarm Optimization

A B S T R A C T

Over the last two decades numerous metaheuristics have been proposed and it seems today that nobody is able to understand, evaluate, or compare them all. In principle, optimization methods, including the recently popular Evolutionary Computation or Swarm Intelligence-based ones, should be developed in order to solve real-world problems. Yet the vast majority of metaheuristics are tested in the source papers on artificial benchmarks only, so their usefulness for various practical applications remains unverified. As a result, choosing the proper method for a particular real-world problem is a difficult task. This paper shows that such a choice is even more complicated if one wishes, with good reason, to use metaheuristics twice, once to find the best and then to find the worst solutions for the specific numerical real-world problem. It often occurs that for either case different optimizers are to be recommended. The above finding is based on testing 30 metaheuristics on numerical real-world problems from CEC2011. First we solve 22 minimization problems as defined for CEC2011. Then we reverse the objective function for each problem and search for its maximizing solution. We also observe that algorithms that are highly ranked on average may not perform best for any given specific problem. Rather, the highest average ranking may be achieved by methods that are never among the poorest ones. In other words, occasional winners may get less attention than rare losers.

1. Introduction

For about two decades researchers have been observing an ever increasing number of newly proposed metaheuristics. Many of them have already been quite widely accepted and have found numerous applications, but some of the most recent ones [2] have encountered serious scepticism and have been criticized for offering little novelty but the name, for being inadequately introduced or improperly tested against the older methods [36].

Many practitioners simply need to identify a relevant optimizer for the problem they are currently interested in. The abundance of more or less widely accepted metaheuristics that frequently share many similarities hidden behind sophisticated vocabulary [36] presents a serious problem here. The problem is exacerbated by the fact that most such algorithms

* Corresponding author. Fax: +48 22 6915 915.
E-mail addresses: adampp@igf.edu.pl (A.P. Piotrowski), maciejnap@gmail.com (M.J. Napiorkowski).

http://dx.doi.org/10.1016/j.ins.2016.08.057

are never compared against one another. As a matter of fact, comparisons among a larger number of methods on many real-world problems are rare in the literature. This is the case even though, due to various deficiencies of artificial benchmarks [27], such comparisons will be very important to practitioners. Although a set of 22 continuous real-world problems (of various dimensionality and difficulty) from various fields of science was collected a few years ago for testing metaheuristics [10], it is often at best only partly used [20,45]. The vast majority of recently introduced optimizers are still tested on either artificial benchmarks, or just on a single practical problem. This offers little help for practitioners in choosing the proper method for their specific applications.

According to No Free Lunch theorems for optimization [47], the performance of all heuristic optimizers averaged over all problems is equal. Inasmuch as most of "all problems", as defined in [47], cannot be of interest to anyone [19], it is hard to give a precise definition of "real-world" or "practical" problems. Hence the impact of No Free Lunch cannot be neglected. Notwithstanding theoretical discussions regarding No Free Lunch theorems for continuous [1] real-world problems, very often simplistic empirical examples may cause scepticism among practitioners as to the way the competing metaheuristics are assessed. The latter directly affects the conclusions regarding performance of particular algorithms.

With respect to empirical studies, Vecek et al. [41] showed that various rankings of algorithms may be obtained when different statistical tests are used. Moreover, it was reported in [11] that statistical tests that take into account the convergence speed may highly modify the conclusions which one would draw based on tests that consider only the final results – even for real-world problems. Most metaheuristics have some control parameters that need to be specified by the user; Dymond et al. [12] showed that these parameters should frequently be set quite differently depending on how many function calls may be used in a particular experiment. As a result, when in some tests an unusual number of function calls is assumed, many algorithms may be easily "outperformed" – for example by the "novel" approach. In a very different study [21] it was noticed that even if some metaheuristics seem to perform relatively well, they may be structurally biased. The authors of [48], when introducing a novel approach to constructing multialgorithms, also showed (seemingly as a by-product of their main effort) that changing the set of artificial benchmarks may deeply modify the ranking of algorithms, even if both sets (i.e. before and after the change) are quite large, composed of problems of various difficulties and developed partly by the same authors.

In another empirical study [31] it was shown that when algorithms are tested with respect to maximization instead of minimization of popular artificial benchmark functions (CEC2005 set [38]), quite different rankings of algorithms may be obtained. It was suggested that this may be an effect of the human impact on the construction of such benchmarks. However, the question as to whether searching for the maximum or for the minimum of the same continuous real-world-based functions would lead to similar or much different rankings of algorithms remains open. This question may be alternatively formulated in the following and seemingly more interesting way: Should one use the same metaheuristic to find the best and to find the worst solution to a practical problem? The present paper is a continuation of the research conducted in [31]. It attempts to provide an answer to this question, but also addresses a slightly wider one: To what extent can a practitioner (who needs to choose metaheuristics for a given specific application) rely on rankings based on the performance of various optimizers, which are applied to a number of different numerical real-world problems? More precisely, we ask whether the superiority of some metaheuristics over others determined based on widely used average ranks (averaged over all problems algorithms are tested on) would be confirmed by the number of specific problems for which a particular algorithm turns out to be the best method. Indeed, one may hope that the algorithms that receive best average ranks on a set of problems would also be those that most frequently achieve the best performance across a number of specific problems – otherwise what would be the practical meaning of average-based comparisons?

However, as discussed above, the problem of testing and comparing metaheuristics is very broadly defined and is addressed in various studies from different points of view. One should note that the scope of this paper is rather limited and focuses only on certain aspects of the questions raised. Hence no firm conclusion like "this is the right (or wrong) way of comparing metaheuristics" should be expected.

2. Methods

This paper assumes that the reader is interested in the performance of numerical optimization methods for real-world applications. Consequently, the CEC2011 set of 22 continuous real-world problems [10] is used for testing the ability of metaheuristics to find the best and the worst solutions. This set includes problems of various dimensionality (from 1- to 216-dimensional ones) and difficulty. Our assessment of the level of difficulty is based on intuition, as there is no agreement on how it could be measured or compared [24]. The real-world problems considered come from the very divergent fields of science and engineering – basic information on them is given in Table 1; for more detailed discussion see [10].

The choice of competing metaheuristics is more difficult, and inevitably subjective. The list of 30 algorithms that have been chosen for this study is given in Table 2. Although one may note that among the chosen optimizers attention is mainly focused on Differential Evolution (DE) methods [37], we also consider a number of Particle Swarm Optimization (PSO) variants [35], a genetic algorithm (GA) [13], a multialgorithm [42], a few methods based on simplexes [8,26], and non-population based heuristics [6,33]. Although the choice is subjective, three among the algorithms tested should be included in the comparison because of the results they achieved in the previous tests: AMALGAM [42] and jDElscop [3] performed best for minimization (AMALGAM) or maximization (jDElscop) of CEC2005 [38] artificial problems in [31], whereas GA-MPC [13] was crowned the winner of the CEC2011 competition [10] on real-world problems.