Coupling the design of benchmark with algorithm in landscape-aware solver design

Johann Dréo, Carola Doerr, Yann Semet

To cite this version:
Johann Dréo, Carola Doerr, Yann Semet. Coupling the design of benchmark with algorithm in landscape-aware solver design. Genetic and Evolutionary Computation Conference, Companion Material, Jul 2019, Prague, Czech Republic. pp.1419-1420, 10.1145/3319619.3326821 . hal-02179604

HAL Id: hal-02179604
https://hal.sorbonne-universite.fr/hal-02179604
Submitted on 14 Jan 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Coupling the Design of Benchmark with Algorithm in Landscape-Aware Solver Design

Johann Dreo  
Thales Research and Technology  
john.dreo@thalesgroup.com

Carola Doerr  
Sorbonne Université, CNRS  
Carola.Doerr@lip6.fr

Yann Semet  
Thales Research and Technology  
yann.semet@thalesgroup.com

ABSTRACT
Following a general trend in artificial intelligence, Evolutionary Computation has, in recent years, witnessed substantial performance gains from landscape-aware selection and parameter tuning of algorithmic modules. Such approaches, however, are critically relying on suitable benchmarks, or training sets, that provide the appropriate blend of performance and generality. With this position paper we argue that, on a landscape analysis basis, the benchmark design problem will form a substantial part of the next-generation of automated, on-demand algorithm design principles.

CCS CONCEPTS
• Computing methodologies → Heuristic function construction; • Randomized search; • Software and its engineering → Search-based software engineering;

KEYWORDS
landscape-aware heuristics, algorithm engineering, algorithm design, black-box optimization, benchmarking, benchmark design

1 INTRODUCTION
The automation of the design of programs has been a classical concept in the evolutionary algorithm community from the start [4, 7] and was quickly applied to the design of algorithms themselves [5].

It was also stated very early that the performance of evolutionary algorithms is very parameter-sensitive [6] and that the tuning problem is in itself a (meta-)optimization problem [8]. Since then, problems related to the automated design of evolutionary algorithms have been commonly studied [1, 8].

In the broad sense, the automated design of an optimization solver may encompass any choice that has an impact on its behavior, namely the selection of the algorithm(s) [11, 14], the choice of its module(s) [15], or its parameter setting(s) [1]. Significant progress has been achieved on that side of the problem with automated parameter tuning methods [10] or hyper-heuristics [3].

More recently, the use of fitness landscapes features [12] as a mean of observation of the problems, from the algorithm perspective, prove to be particularly useful for design problems [2, 11].

2 LANDSCAPE-AWARE SOLVER: ALGORITHM + BENCHMARK
In order to design a good, specific, solver, two problems are crucial: algorithm design and benchmark design. The problem of designing the best algorithm is arguably the most studied in the literature, but we argue that it should be coupled with the problem of designing the benchmark (cf. Figure 1) in order to provide adequate feature-wise representation of the intended use.

Modern search heuristics performances are always assessed first on a benchmark, most of the time a “synthetic” one, designed for fast execution and a sufficiently large “footprint” [9, 13], or representativity of potential instance types. As such, their design is very often influenced by the benchmark’s design. This is even more true for heuristics that embed a landscape-aware process, where the given benchmark fully determines the general performance of the algorithm. Following [11, 13], we thus argue that having benchmarks with large “footprint” is a crucial challenge.

Going further, we argue that landscapes may be used for automated benchmark design. In that setting, the problem becomes to design benchmarks that “cover” the landscape space of a given optimization problem well enough, with as few instances as possible. That is to find a fixed set of problem instances from which reasonable performance prediction models can be built.

In the ideal setting, the benchmark generator is a configurable mean to produce problem instances with respect to specified landscape features based performance criteria. Hidden in that setting is the problem of selecting the subset of features that are useful for the benchmark instances themselves.

Figure 1 shows the proposed coupling between the algorithm design and benchmark design optimization problems. The process is heavily influenced by the benchmark’s design. This is even more true for heuristics that embed a landscape-aware process, where the given benchmark fully determines the general performance of the algorithm. Following [11, 13], we thus argue that having benchmarks with large “footprint” is a crucial challenge.

Going further, we argue that landscapes may be used for automated benchmark design. In that setting, the problem becomes to design benchmarks that “cover” the landscape space of a given optimization problem well enough, with as few instances as possible. That is to find a fixed set of problem instances from which reasonable performance prediction models can be built.

In the ideal setting, the benchmark generator is a configurable mean to produce problem instances with respect to specified landscape features based performance criteria. Hidden in that setting is the problem of selecting the subset of features that are useful for the benchmark instances themselves.

Figure 1 shows the proposed coupling between the algorithm design and benchmark design optimization problems. The process is heavily influenced by the benchmark’s design. This is even more true for heuristics that embed a landscape-aware process, where the given benchmark fully determines the general performance of the algorithm. Following [11, 13], we thus argue that having benchmarks with large “footprint” is a crucial challenge.

3 CONCLUSION: COUPLED DESIGN NEEDED
We think that very significant gain can be expected from automating solver design for a specific problem and set of performance criteria by automatically generating the benchmark used by landscape-aware processes. The essential goal is the appropriate generalization level: maximum performance on unforeseen use without counterproductive generalization. The long-term goal is an optimized solver generator in which an end-user would input her problem formalization, her desired benchmark footprint and performance criteria. Managing the interplay between features, benchmarks and algorithms is of course one of the biggest challenges in our suggested approach.
Figure 1: Diagram of the landscape-aware solver design scheme. The coupling between the algorithm design (top-left loop) and the benchmark design (top-right loop) occurs through the optimized set of problem instances and selected features (center). The output is a landscape-aware solver (bottom) that is able to adapt its behaviour to foreseen problem features.

REFERENCES