From benchmarking optimization heuristics to dynamic algorithm configuration

Vermetten, D.L.

Chapter 1

Introduction

Optimization problems can be found in a wide variety of contexts, from complex industrial problems where we aim to increase the crashworthiness of a car body to daily life where we want to choose the shortest route home. While we are always looking for better solutions, it is usually impossible to try every possible option. Instead of an exhaustive evaluation, we can introduce a heuristic search procedure to efficiently explore the solution space. At the cost of potentially missing the 'perfect' solution, these types of methods prove to be rather effective in practice.

Because of the prevalence and variety of optimization problems, heuristic optimization algorithms are continuously being developed to solve subsets of problems more effectively than before. This development has led to a thriving research area, and the resulting algorithms have significantly benefited a wide range of real-world applications.

The variety of available optimization heuristics is a clear indication of the strengths of this class of algorithms. However, with so many methods to choose from, it can be hard to gauge which algorithms excel in a given setting. In order to support the continued development and understanding of heuristic optimization algorithms, standardized practices for testing and comparing their results are a necessity. *Benchmarking tools* support this aim by providing access to sets of problems with known properties and fixed pipelines for logging the optimization process. The resulting data can then be compared fairly to data from existing algorithms in a variety of ways, including aggregated visualizations and statistical testing procedures.

In this thesis, we will focus on the IOHprofiler benchmarking environment, and show how this tool supports robust research in a variety of manners. In particular, we highlight the ways in which we can use benchmarking in combination with the modularization of algorithms to gain insight into the benefits of individual algorithmic components, as well as shine a light on the interplay between pairs of components. This shows the inherent complementarity between different algorithmic principles, where some combinations of components outperform others on one type of problem, while the reverse is true for problems with different overarching properties. This notion of complementarity between algorithms lies at the core of *algorithm selection*, where we aim to first identify some functional properties of the problems to be optimized, to then determine which algorithm would be most appropriate to use for the optimization.

By combining the ideas of algorithm selection with the detailed insights gained from tracking the full performance trajectory during benchmarking, we finally get into the domain of *dynamic algorithm selection*. Here, we don't merely exploit differences between algorithms on different types of problems, but instead, we aim to utilize the information about the local state of the optimization algorithm to decide whether to continue optimizing the same way or to switch to a completely different algorithm. Depending on the way in which this problem is framed, we can also consider the combination with *algorithm configuration*, where we tune the parameters of the optimization algorithm, leading to an even larger space of potential choices and corresponding benefits.

1.1 Research Questions

The overarching question of this thesis is: *How can benchmark data be used to gain insights into, and subsequently exploit, different levels of algorithm complementarity?* Since this question encompasses a wide range of topics, we further specify a set of underlying research questions which shine a light on a selected set of components ranging from benchmarking software to dynamic algorithm selection.

RQ1 *How can robust benchmarking pipelines be made accessible and resulting data be made usable by the wider community?* This is the core focus of Chapter 3 (based on [255, 53, 177, 43, 136, 150, 244, 236]), where we introduce *IOHprofiler*, a benchmarking environment with a modular design which aims to lower the barrier of entry to robust experimental design. By examining the wide range of problems made available in IOHprofiler, we show the way in which benchmark design influences the types of conclusions which can be drawn from studies which use these types of problems. In addition to the common performance-oriented benchmarks, we also discuss *behavior-based benchmarking*, illustrating the ways in which this difference of perspective can change the ways in which we look at the differences between iterative optimization heuristics.

RQ2 *How can a modular design aid in the exploration of interactions between different algorithmic ideas?* Most new algorithmic ideas are not proposed in isolation, but instead build upon existing algorithm families. While these modifications are usually compared to a baseline implementation of the original algorithm, their potential interactions with other modifications are usually hard to judge. In Chapter 4 (based on [51, 240, 247, 246]), we explore how the usage of modular design principles can be combined with algorithm configuration techniques to explore the strengths of different modifications within an algorithm family.

RQ3 *To what extent can we exploit performance complementarity between different algorithms by switching between them?* While performance complementarity is classically exploited by algorithm selection or algorithm configuration techniques, we expand these ideas to account for the potential benefit of switching between algorithms *during the search process*. By switching algorithms in this way, we can potentially speed up convergence by combining algorithms' strengths in different phases of the optimization. In Chapter 5 (based on [245, 243, 110, 135, 248]), we explore this potential from a data-driven perspective, as well as by performing switching both within a fixed algorithm family and between completely different optimizers.

RQ4 *How can we fairly judge the performance of meta-learning methods in the context of black-box optimization?* While the previously discussed research questions focus on methods for exploiting algorithm complementarity, it is important to note how the efficacy of these meta-learning techniques depends on the used benchmark suites. In particular, the generalizability of results from algorithm selection and configuration remains an open problem. In Chapter 6 (based on [250, 251, 249]), we discuss MA-BBOB, a new benchmark problem generator, which we introduce to create larger sets of training and testing functions for these kinds of meta-learning methods.

1.2 Other Contributions by the Author

While this thesis covers a variety of aspects of data-driven benchmarking and its role in dynamic algorithm selection, a set of other tangentially related research directions have been omitted for reasons of space. A full list of publications can be found at the end of this thesis.

1.3 Thesis Outline

In Chapter 2, we provide information on a variety of background topics related to the work in the thesis. In particular, we discuss continuous optimization and ways in which we can characterize these types of problems. Chapter 3 discusses benchmarking optimization algorithms, with a focus on the benchmarking platform IOHprofiler. This section also discusses more behaviour-based benchmarking in the form of structural bias detection. Chapter 4 focuses on the problems of algorithm configuration and selection, highlighting the challenges in configuring the inherently stochastic optimization algorithms discussed throughout this thesis. In particular, we show how taking a modular approach to algorithm design can lead to significant improvement in performance when algorithm configuration methods are used. Chapter 5 then introduces dynamic algorithm selection and configuration, highlighting the potential benefits to be gained by switching between optimization algorithms. This section shows several use cases, including the promising per-run trajectory-based selection method. Finally, Chapter 6 discusses the problem of generalizability in the context of continuous optimisation and proposes a new test suite on which this aspect can be further investigated. The core results from the thesis are then summarized in Chapter 7, where we finally discuss future research directions based on the insights obtained.