Benchmarking Iterative Optimization Heuristics with IOHprofiler⋆

Carola Doerr*

Sorbonne University, Paris, France

Abstract

Iterative optimization heuristics (IOHs) such as local search algorithms, evolutionary and genetic algorithms, or surrogate-based optimization algorithms form an important class of algorithms, designed to find high-quality solutions for challenging optimization problems in an query-based fashion. IOHs alternate between generating solutions, evaluating the quality of these candidates, and using this information to adjust the strategy by which the next solution candidates are generated. Many different IOHs exist, posing a meta-optimization challenge of selecting the most appropriate algorithm for a given problem at hand. One of the key ambitions of our research domain is hence to understand which algorithms work well for which types of problems. A central approach for generating insight into the performance and the behavior of IOHs is benchmarking, i.e., the empirical comparison of algorithms on well-chosen sets of instances, collected to help us understand strengths and weaknesses of different algorithms. In this presentation, I will give a brief introduction to IOHprofiler, our software environment built to support users in benchmarking activities. IOHprofiler is designed as a modular tool, covering all central parts of the entire benchmarking pipeline. For example, the IOHexperimenter module provides easy access to common problem sets (e.g. BBOB functions, CEC competitions, GECCO submodular and star discrepancy competitions, etc.) and can be easily combined with other collections of optimization problems. The resulting logs as well as logs from other platforms (such as COCO and Nevergrad) are fully interoperable with IOHanalyzer, the module designed for interactive performance analysis. A GUI, hosted at https://iohanalyzer.liacs.nl/ makes these analysis tools easy to access. Data from many repositories (e.g. COCO, Nevergrad) are pre-processed, such that the effort required to compare performance to existing algorithms is greatly reduced. IOHprofiler is available at https://iohprofiler.github.io/ Besides applications in research contexts, we will also discuss how to use IOHprofiler for teaching as well as for industrial collaborations.