Multi-objective Performance Measurement: Alternatives to PAR10 and Expected Running Time

Jakob Bossek(✉) and Heike Trautmann

Information Systems and Statistics, University of Münster, Münster, Germany
{bossek,trautmann}@uni-muenster.de

Abstract. A multiobjective perspective onto common performance measures such as the PAR10 score or the expected runtime of single-objective stochastic solvers is presented by directly investigating the tradeoff between the fraction of failed runs and the average runtime. Multi-objective indicators operating in the bi-objective space allow for an overall performance comparison on a set of instances paving the way for instance-based automated algorithm selection techniques.

Keywords: Algorithm selection · Performance measurement

1 Introduction

Benchmarking and comparisons of (single-objective) optimization algorithms strongly rely on adequate performance measurement of the respective solvers. However, assessing solver performance in general is not straightforward at all as usually multiple views and requirements have to be considered simultaneously such as minimizing runtime, minimizing function evaluations, maximizing quality, etc.. For stochastic solvers specifically, minimizing variability across runs or maximizing the number of successful runs might be of interest. Quite often though, only a single indicator or a single-objective combination of indicators is focussed in practice.

Common measures like PAR10 (e.g. [1], mostly in combinatorial optimization) and Expected Running Time (ERT [4], mostly in continuous optimization) for example try to combine several aspects into a single performance indicator while the core concept is the distinction between successful and unsuccessful runs as well as possible penalization of the latter. Usually, a run is denoted as successful if it solves an instance to optimality within a given time limit, e. g., for the Traveling Salesperson Problem (TSP).

In our approach we propose to address the tradeoff between minimizing the fraction of unsuccessful runs and the minimization of the average running time directly by treating it as a multi-objective optimization problem for which specific multi-objective techniques and indicators can be used to measure overall algorithm performance across an instance/benchmark set. This provides the

R. Battiti et al. (Eds.): LION 12 2018, LNCS 11353, pp. 215–219, 2019.
https://doi.org/10.1007/978-3-030-05348-2_19

basis for automated instance-based algorithm selection techniques based on the suggested performance measure generated by multi-objective techniques related to the idea of automated multi-objective configuration presented in [2]. Moreover, the underlying concept generalizes to other kinds of performance indicator combinations (also of higher degree) as well.

2 Performance Measurement

Common Approaches. In combinatorial optimization the so-called penalised average runtime (**PAR10**, e.g. [1]) score is a widely used hybrid performance measure. It is defined as the average of the runtimes with unsolved instances penalized with $10 \cdot T$ where T is the cutoff time. It is thus a combined measure of number of successful runs and average running time.

The Expected Running Time (**ERT**, [4]) basically measures the average number of function evaluations (across multiple runs of a solver on an instance) that are needed to solve it. Usually applied in single-objective continuous black-box optimization success here means that the resulting solution differs at most by a predefined precision value from the global optimum. The ERT is a weighted sum of the expected average running time of the succesful runs and the cutoff time T, while it is weighted by the fraction of failure and success probability.

Multi-Objective Perspective. Both widely used performance measures PAR10 and ERT implicitly address the two goals of maximizing probability of success and minimizing the expected running time of a solver. However, the actual tradeoff between those objectives is concealed by solely focussing on the aggregated performance measure. Moreover, often, high penalty values bias the performance analysis while the extent of the used penalty is more or less arbitrary chosen. We therefore propose an alternative bi-criteria performance measure PF^{MO}. Let T_A^s be the random variable that describes the running time of successful runs of algorithm A and $p_f = 1 - p_s$ the probability of failure, then

$$\text{PF}^{MO} := (r_s = E(T_A^s), p_f).$$

Obviously, we aim to minimize both criteria, i. e., a "good" algorithm will both minimize the number of unsuccessful runs and simultaneously minimize the expected running time of successful runs. Note, that $\text{PF}^{MO} \in [0, T] \times [0, 1]$ and $(0, 0)$ is the desirable ideal or utopia point. The measure may be depicted in a 2D scatterplot either for each instance and algorithm combination or aggregated over all k instances of the respective instance set (see Fig. 2 for an example).

Multi-Objective Assessment. Within the resulting two-dimensional space, we may adopt the concept of *Pareto-dominance* [3] in order to compare solver performances on and across instances. Assuming point labels reflecting algorithms and both components of PF^{MO} as axis labels in Fig. 1 we see that while algorithms A and B are non-dominated, i. e.,

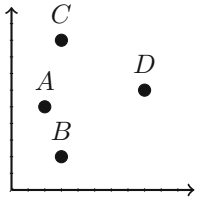

Fig. 1. Concept illustration.

A shows better average runtime, but worse failure rate than algorithm B and vice versa, e. g., A and B dominate C and D. Thus, we prefer algorithms that are located on low non-domination levels, ideally have a non-domination rank of one. By this means algorithms of the same rank become incomparable.

Moreover, the dominated Hypervolume (HV, [3,6]) of a point (i. e., its HV contribution) can be used to reflect a performance ranking of algorithms. It measures the size of the dominated space bounded by an anti-optimal reference point (which is implicitly given by $(T, 1)$). It is compliant with the Pareto dominance relation in that a lower nondomination rank leads to a higher HV value. HV contributions per algorithm can be aggregated over an instance set inducing a total order of algorithms.

3 Exemplary Illustration

This section is based on a benchmark study including feature-based automated algorithm selection of state-of-the-art inexact TSP solvers such as LKH, EAX as well as their restart variants and MAOS performed in [5]. Performances were measured in terms of PAR10 (using more robust median instead of mean for aggregation) on different representative kinds of instance sets (rue, tsplib, vlsi, netgen, morphed) and EAX+restart turned out to be the single best solver across all instance sets followed by LKH+restart. Figure 2 adopts the presented multi-objective view by showing the tradeoffs between both failure rate and average running time of successful runs.

In the aggregated version, i. e. averaged across the respective instance set per dimension, we see that EAX+restart clearly dominates the remaining solvers. However, differences to LKH and LKH+restart clearly are due to differences in runtime while both EAX and MAOS show substantially higher failure rates on average.

Table 1 shows summary statistics of the individual non-domination ranks and HV values of all algorithm performance results, i.e. of all points in the upper left subfigure of Fig. 2 allowing for an overall solver ranking across all instances by using averages across the instance set. Interestingly, the HV based ranking (see also bottom part of Fig. 2) is very much in line with the PAR10 based results in [5] while the non-domination ranks favor LKH+restart. However, differences in many cases are not statistically significant across the whole instance set but on several subsets such as e.g. netgen.

4 Conclusions

The bi-objective performance measure PF^{MO} as an alternative to PAR10 and ERT is presented by directly investigating the tradeoff between failure rate and average running time of successful runs. Thereby, the concept of Pareto dominance and multiobjective performance assessment in terms of dominated Hypervolume and non-dominated sorting offers very promising perspectives and new

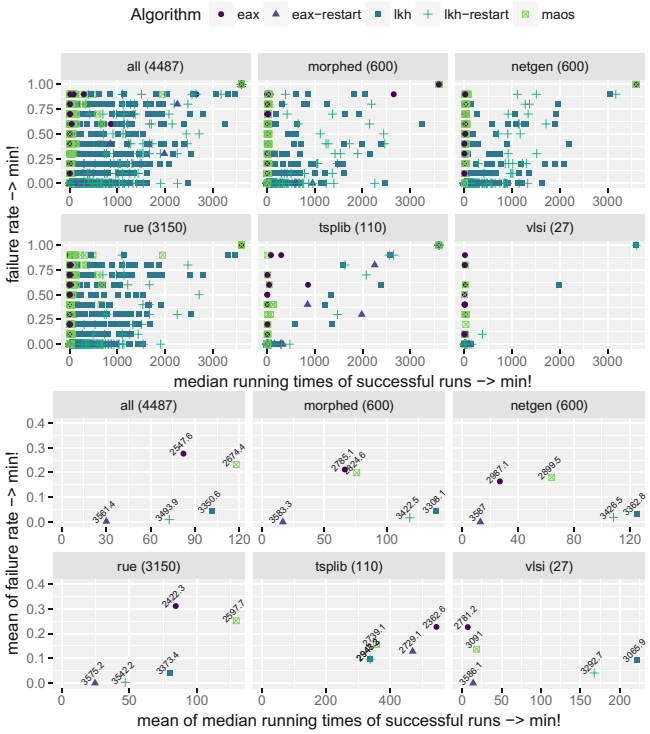

Fig. 2. Scatterplots of raw (all instances, top) and aggregated (r_S, p_f) vectors per instance set. Aggregation is performed componentwise, i. e., mean of failure rates and mean of running times of successful runs. The numbers indicate the unnormalized average dominated Hypervolume.

Table 1. Summary statistics of non-domination ranks and HV aggregated across all instances.

Algorithm	Avg. rank	SDev. rank	Avg. HV	SDev. HV
eax	2.25	1.05	2599.71	1007.71
eax-restart	2.28	1.03	**3569.18**	**178.45**
lkh	2.18	1.08	3382.47	603.23
lkh-restart	**2.03**	**0.97**	3510.41	360.99
maos	3.50	1.16	2749.20	1004.75

insights into algorithm behaviour. However, the core concept generalizes to other kinds of indicator combinations as well.

First conceptual studies of PF^{MO} on a TSP benchmark of inexact solvers hint at interesting aspects of solver behaviour which will be further analysed in

future studies together with thoroughly comparing properties of PAR10, ERT and PF^{MO}.

Acknowledgements. The authors acknowledge support from the European Research Center for Information Systems (ERCIS) and the DAAD PPP project No. 57314626.

References

1. Bischl, B. et al.: ASlib: a benchmark library for algorithm selection. Artif. Intell. J. **237**, 41–58 (2016). https://doi.org/10.1016/j.artint.2016.04.003
2. Blot, A., Hoos, H., Jourdan, L., Marmion, M., Trautmann, H.: In: Joaquin, V. et al. (ed.) MO-ParamILS: A multi-objective automatic algorithm configuration framework, pp. 32–47. Springer International Publishing, Ischia (2016)
3. Coello Coello, C., Lamont, G.B., van Veldhuizen, D.: Evolutionary Algorithms for Solving Multi-objective Problems. Springer, Berlin (2007)
4. Hansen, N., Auger, A., Finck, S., Ros, R.: Real-Parameter Black-Box Optimization Benchmarking 2009: Experimental Setup. Technical Report RR-6828, INRIA (2009). https://hal.inria.fr/inria-00362649v3/document
5. Kerschke, P., Kotthoff, L., Bossek, J., Hoos, H.H., Trautmann, H.: Leveraging TSP solver complementarity through machine learning. Evol. Comput. **0**(0), 1–24 (2017). https://doi.org/10.1162/evco_a_00215, pMID: 28836836
6. Zitzler, E., Thiele, L.: Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. IEEE Trans. Evol. Comput. **3**(4), 257–271 (1999)