Extending time-to-target plots to test sets with multiple problem instances: m-tttplots

Alberto Reyes
Universidade Federal Fluminense, Institute of Computing, Niterói, Brazil.
celso@ic.uff.br

Celso C. Ribeiro
Universidade Federal Fluminense, Institute of Computing, Niterói, Brazil.
albertord84@gmail.com

April 12, 2016

Metaheuristics are general high-level procedures that coordinate simple heuristics and rules to find good (often optimal) approximate solutions to computationally difficult combinatorial optimization problems. Some metaheuristics (such as GRASP and VNS) rely on randomization to sample the search space. Randomization can be used to break ties, so as that different trajectories can be followed from the same initial solution in multistart methods or to sample fractions of large neighborhoods. Greedy randomized algorithms make use of randomization to build different solutions at different runs.

Time-to-target plots (or, more simply, tttplots) display on the ordinate axis the probability that an algorithm will find a solution at least as good as a given target value within a given running time, shown on the abscissa axis. They provide a very useful tool to characterize the running times of stochastic algorithms for optimization problems and to compare different algorithms or strategies for solving a given problem. They are widely used as a tool for algorithm design and comparison.

Aiex et al. [2] developed a Perl program to create time-to-target plots for measured times that are assumed to fit a shifted exponential distribution, following closely [1]. Such plots are very useful and have been widely used in the comparison of different algorithms or strategies for solving a given problem.

Later, Ribeiro et al. [3, 4] proposed a numerical method to compare the runtime distributions obtained by two algorithms for the same problem instance. Given two algorithms $\mathcal{A}$ and $\mathcal{B}$ for solving the same problem $\mathcal{P}$, their approach computes the probability X that algorithm $\mathcal{A}$ finds a target solution value *look4* for an instance $\mathcal{I}$ of problem $\mathcal{P}$ in less time than algorithm $\mathcal{B}$. Ribeiro and Rosseti [5] developed a Perl code that implements these computations.

However, tttplots can only consider only one instance at a time and conclusions are limited to the instance that generated each plot. To overcome this limitation, we present in this work the extension of tttplots to multiple instances and we propose a new tool for the evaluation and comparison of randomized heuristics: m-tttplots.

Given a set of problem instances and their corresponding targets values, the multiple time-to-target plot (or, more simply, m-tttplot) of a given algorithm for this test set displays on the ordinate axis the probability that this algorithm will find a solution at least as good as the corresponding target value within a given total running time, shown on the abscissa axis, for all instances.

m-tttplots may be numerically computed by integration for $n = 2$, following the same ideas in [3, 4]. They can also be recursively computed by numerical integration for arbitrarily many variables. In addition, they can also be constructed by Monte Carlo simulation from the individual tttplots of each instance. These computations will be presented in detail in the final, extended version of this work, together with a public domain code that will be made available and with extensive numerical results for different problems, illustrating the applicability and usefulness of the newly proposed tool.

References

[1] R.M. Aiex, M.G.C. Resende, and C.C. Ribeiro, "Probability distribution of solution time in GRASP: An experimental investigation", *Journal of Heuristics*, 8:343-373, 2002.

[2] R.M. Aiex, M.G.C. Resende, and C.C. Ribeiro, "TTTPLOTS: A Perl program to create time-to-target plots", *Optimization Letters*, 1:355-366, 2007.

[3] C.C. Ribeiro, I. Rosseti, and R. Vallejos, "On the use of run time distributions to evaluate and compare stochastic local search algorithms", *Lecture Notes in Computer Science*, 5752:16-30, 2009.

[4] C.C. Ribeiro, I. Rosseti, and R. Vallejos, "Exploiting run time distributions to compare sequential and parallel stochastic local search algorithms", *Journal of Global Optimization*, 54:405-429, 2012.

[5] C.C. Ribeiro and I. Rosseti, "TTTplots-COMPARE: A Perl program to compare time-to-target plots or general runtime distributions of randomized algorithms", *Optimization Letters*, 9:601-614, 2015.