A statistical methodology for analysing heuristic algorithms

J. Corstjens, A. Caris, B. Depaire
Hasselt University, Research Group Logistics
e-mail : jeroen.corstjens@uhasselt.be

K Sörensen
Antwerp University

Heuristic experimentation commonly entails running an algorithm on the instances of some standard benchmark problem set and measuring its performance — solution quality, run time or both. These performance results are then compared with the results other heuristic algorithms obtained on this benchmark problem set. It is a type of evaluation that ensues a competition with state-of-the-art methods in the literature. The goal is to obtain a better solution quality and/or a faster running time on the benchmark instances than other existing algorithm and claim first place in the "horse race". This approach, however, does not seek to explain why one method performs better than another one [2]. Which elements of the heuristic algorithm have contriubted to a greater or less extent to this superior performance ? Is it mainly due to a certain (combination of) operator(s) employed within the algorithm ? Or fixing certain parameters at specific values ? Or maybe it is due to a researcher's superior coding skills leading to a more efficient implementation of an existing algorithm ? Do all components significantly contribute to the performance of the algorithm, or can certain elements be left out, thereby possibly increasing the efficiency of the method ? These are all questions that often remain unanswered when a new method is presented. Even though some competition between researchers might spur innovation, it has been noted that true innovation builds on the understanding of how a heuristic algorithm behaves, and not on proof of competitiveness [5]. A competitive focus works when considering a specific setting [4], but when the objective is to learn how the different heuristic elements contribute to performance and make statements beyond a specific problem setting, a statistical evaluation methodology has to be applied.

We propose a statistical methodology with the principal aim of gaining a thorough understanding of the relationship between algorithm performance, algorithmic properties, and problem instance characteristics. We wish to identify how the algorithmic properties impact algorithm performance, positively or negatively, and how these effects vary across different parts of the problem space. The proposed methodology relies on multilevel models that enables to study how algorithmic parameter effects vary given different problem conditions.

In a first application of the methodology a number of randomly generated instances for the vehicle routing problem with time windows are solved using a simplified version of the Adaptive Large Neighbourhood Search algorithm [3] that considered less operators and also removed the adaptive mechanism used to assign weights to the operators after each iteration. The results showed that including more operators to an algorithm does not necessarily lead to a better performance in terms of solution quality. We often observed better results for configurations with only one repair operator and one or two destroy operators. Furthermore, the characteristics of a specific instance influence these effects in such a way that conclusions differ, for example, between instances with a small number of instances and instances with many customers [1]. For a second experiment, we include the adaptive mechanism for assigning weights to the operators per iteration, compare the findings with our first experiment and seek to expose the contribution of the adaptive process.

Acknowledgements

This work is supported by the National Bank of Belgium and the Interuniversity Attraction Poles Programme initiated by the Belgian Science Policy Office (research project COMEX, Combinatorial Optimization : Metaheuristics & Exact Methods

The computational resources and services used in this work were provided by the VSC (Flemish Supercomputer Center), funded by the Research Foundation - Flanders (FWO) and the Flemish Government âĂŞ department EWI.

Références

[1] Corstjens, J., Depaire, B., Caris, A., & Sörensen, K. (2016). Analysing metaheuristic algorithms for the vehicle routing problem with time windows. In Verolog 2016 proceedings, page 89.

[2] Hooker, J. N. (1995). Testing heuristics : We have it all wrong. Journal of heuristics, 1(1), 33-42.

[3] Pisinger, D., & Ropke, S. (2007). A general heuristic for vehicle routing problems. Computers & operations research, 34(8), 2403-2435. Relations.

[4] Rardin, R. L., & Uzsoy, R. (2001). Experimental evaluation of heuristic optimization algorithms : A tutorial. Journal of Heuristics, 7(3), 261-304.

[5] Sörensen, K. (2015). Metaheuristics - the metaphor exposed. International Transactions in Operational Research, 22(1), 3-18.