A Multilevel Methodology for Analysing Metaheuristic Algorithms for the VRPTW

J. Corstjens[1], B. Depaire[1], A. Caris[1] and K. Sörensen[2]
[1]Hasselt University
jeroen.corstjens@uhasselt.be
[2]University of Antwerp

Abstract
Heuristic algorithms for the vehicle routing problem are commonly assessed by using a competitive evaluation methodology. This may give an indication of which algorithm obtains a better performance, it does not explain, however, why it is better. The appliance of a proper statistical methodology can provide researchers a better understanding of how performance is affected by the different algorithmic parameters and heuristic components and result in a more robust parameter setting and scientific basis for comparison.

Keywords: Algorithm Configuration, Metaheuristics, Statistical Methodology, Vehicle Routing.

1 Problem Statement

Vehicle Routing Problems (VRP) are an extensively studied class of combinatorial optimization problems, with a wide spectrum of real-life applications. In its basic version it is the problem of finding a set of routes in which each customer is visited only once and with the objective of minimizing total costs. The NP-hardness of the problem resulted in an impressive number of heuristic procedures being proposed for VRP problems, ranging from standard route construction and improvement algorithms to powerful metaheuristic algorithms [2][3]. However, no common, agreed-upon methodology is used to analyse heuristic performance on vehicle routing problems. In VRP literature, (meta)heuristics are rarely evaluated by means of statistical techniques and is heuristic performance traditionally studied by evaluating the performance of a specific heuristic on a set of standard test problems.

The benchmark testing approach focuses on being competitive by showing that a new algorithm can outperform existing ones on a standard set of benchmark instances[6]. Although this competitive approach may indicate which algorithms are better, it does not give any explanation why these are better and what factors impact performance the most[1]. The use of the same benchmark set over and over again also risks overfitting the algorithm to the problem set and limits any statements made to the problem instances used in these specific experiments. In order to also generalise conclusions to unseen problem instances, a proper statistical analysis is required[8]. A heuristic algorithm performing well on some set of standard benchmark problems does not necessarily mean it will work well on any problem set.

In order to better understand why an algorithm performs the way it does and to be able to make valid statements that are not limited to a benchmark problem set, a statistical methodology to analyse the relationship between VRP algorithm parameters, components and their performance is necessary and can be of use in designing, optimising and comparing heuristic algorithms. By applying statistical techniques, it can be evaluated whether any observed performance differences are statistically significant, or whether they are simply due to chance[11].

2 State-of-the-art

The deployment of a metaheuristic involves selecting appropriate values for a multitude of parameters. The setting of these parameters influences the effectiveness with which a heuristic algorithm can solve a particular (class of) problem instance(s)[7]. The problem of identifying optimal or good parameter values is referred to as the parameter setting problem or algorithm configuration problem. It is still most common to do this manually, relying on trial-and-error, performing tests on a limited sample of benchmark instances, or simply quoting values from literature without any proper examination of their suitability in the used context. Such an approach makes it difficult to estimate the robustness of proposed algorithms and provides no scientific basis for their comparison. The configuration is rarely based using some rigorous statistical procedure, but usually relies on personal experience and rules of thumb, which often require time consuming experiments. This time and effort spent in manually adjusting the parameters to values that perform good on the used benchmark sets and developing the fastest possible code, is time that could otherwise be spent on learning about the problem and the algorithm under study[6][10].

Although a proper statistical analysis can considerably improve metaheuristic performance [9], the number of published VRP articles that make notion of using either Design of Experiment techniques or statistical tools for obtaining a parameter setting is rather small[4][11][9]. The majority of these papers were published in recent years, indicating the topic is getting increased attention within the research field of vehicle routing. Most of these publications apply a full factorial design in which every combination of parameter values is considered and is then often followed by an Analysis of Variance (ANOVA), a statistical methodology for detecting statistically significant differences among several sample means. Other researchers focused their efforts on more general automatic algorithm configurators such as I/F-Race or SMAC[7]. These configurators are commonly divided into two groups: model-free and model-based procedures. The former try to find parameter configurations that perform well over a large set of instances, while the latter use information from previous evaluated configurations to build a model of the parameter setting space and then select new candidate configurations to be evaluated. Our research is in line with these model-based approaches which have the advantage that the model can identify the relationship between a certain parameter setting and algorithm performance.

All these methods have moved away from the manual approach and introduced a more rigorous way of obtaining parameter values, rather than relying on trial-and-error. A next step would be to look at the methodology that is behind the parameter tuning process, to try to understand why a certain parameter setting obtains better results than another one. Is there a crucial heuristic component that always exerts a positive influence on performance and therefore always needs to be activated? Or perhaps a component that needs to be left out due to a continuous negative performance impact? What happens if we change a certain parameter value from one level to another? How do the problem characteristics influence the relationship between an algorithmic parameter and performance? Answering these questions will provide a better insight and understanding of how performance, the algorithmic parameters and problem characteristics interact.

The aim of this research therefore is to develop a new methodological framework to obtain more insight into metaheuristic performance.

3 Methodology

The proposed methodological framework relies on regression models to obtain complete insights over the full range of algorithmic parameter values and problem characteristics. A regression perspective is preferred over the traditional ANOVA approach due to the fact that ANOVA is limited to categorical variables and therefore only insights into the algorithmic performance can be gained for the levels that are measured. The regression perspective allows statements to be made for the complete range of values. ANOVA, on the other hand, can more easily represent non-linear effects. However, since an ANOVA model can also be expressed as a regression model, the latter approach is chosen with the added benefit of being able to incorporate continuous variables.

The aim is to gain a thorough understanding of the relationship between algorithm performance, algorithmic properties and problem characteristics that will allow us to determine the optimal parameter setting. The chosen approach for understanding these relationships will use a multilevel experimental design consisting of two levels. One level concerns the problem instances. First, the population of problem instances will be defined by specifying the distributions for different instance parameters. Next, from this population, a random sample of artificial instances (level 1) will be drawn. On a second level the algorithmic properties are of interest. Multiple algorithm variants will be created by randomly selecting parameter values and algorithmic components.

Each randomly generated problem instance will be solved by a fixed number of randomly chosen algorithm variants. Such a multilevel data structure is problematic for classical regression models because we can expect observations for parameter setting variants on the same problem instance to be more similar than observations for parameter setting variants on different problem instances. This means that individual observations are not completely independent, an assumption made (about the error terms) in classical regression and, if violated, could lead to inaccurate statistical estimations. Therefore a multilevel regression analysis needs to be applied that takes the hierarchical structure of the data into account [5].

$$Y_i = \alpha_{i[j]} + \beta_{i[j]} Z + \varepsilon_i \tag{1}$$

$$\alpha_j = \mu_0^\alpha + \mu_1^\alpha X_1 + \mu_2^\alpha X_2 + \eta_j^\alpha \tag{2}$$

$$\beta_j = \mu_0^\beta + \mu_1^\beta X_1 + \mu_2^\beta X_2 + \eta_j^\beta \tag{3}$$

An illustration of a multilevel model is given in equations (1) to (3). Equation (1) measures the solution quality of the algorithm variant, represented by Y, based on a single algorithmic parameter, represented by Z. The intercept α represents the problem and has its own regression model based on the problem characteristics (equation (2)), so for each problem instance we have a different intercept. The slope (β) of the algorithmic parameter indicates the effect it has on performance and can also be modelled as is shown in equation (3). The problem characteristics will then act as moderator variables on the relationship between a particular algorithmic parameter and algorithm performance.

What we want to learn from this regression analysis is how a single algorithmic parameter has an impact on performance, under the influence of the problem characteristics. This knowledge will provide us a better understanding of how and when a parameter or a heuristic component works and eventually allow us to determine a robust algorithmic configuration, which is essential to allow fair comparisons of different algorithms on a broad set of optimization problems.

References

[1] Birattari, M. (2009). Tuning Metaheuristics. Berlin, Heidelberg: Springer.

[2] Bräysy, O. and Gendreau, M. (2005a), Vehicle routing problem with time windows, part I: route construction and local search algorithms, Transportation Science, 39(1), pp. 104-118.

[3] Bräysy, O. and Gendreau, M. (2005b), Vehicle routing problem with time windows, part II: metaheuristics, Transportation Science, 39(1), pp. 119-139.

[4] Coy, S. P., Golden, B. L., Runger, G. C. and Wasil, E. A. (2000). Using experimental design to find effective parameter settings for heuristics. Journal of Heuristics, 7, pp. 77-97.

[5] Gelman, A., and Hill, J. (2006). Data Analysis Using Regression and Multilevel/Hierarchical Models. Cambridge University Press.

[6] Hooker, J. (1995). Testing Heuristics: We have it all wrong, Journal of Heuristics, 1(1), pp. 33-42.

[7] Hoos, H. H. (2011). Automated Algorithm Configuration and Parameter Tuning. In: Autonomous Search. s.l.:Springer Berlin Heidelberg, pp. 37-71.

[8] Montgomery, D. C. (2005). Design and Analysis of Experiments. 6th ed. s.l.:John Wiley and Sons Inc.

[9] Palhazi Cuervo, D., Goos, P., Sörensen, K., and Arriz, E. (2014). An iterated local search algorithm for the vehicle routing problem with backhauls. European Journal of Operational Research, 237(2), pp. 454-464.

[10] Rardin, R. and Uzsoy, R. (2001). Experimental evaluation of heuristic optimization algorithms: A tutorial. Journal of Heuristics, 7, pp. 261-304.

[11] Sörensen, K. and Schittekat, P. (2013). Statistical analysis of distance-based path relinking for the capacitated vehicle routing problem. Computer & Operations Research, 40(12), pp. 3197-3205.