



Contribution ID: 83

Type: **Poster (sponsored)**

Automated Design of Search Methods

Search problems are encountered in robotics, bioinformatics, and many other industrial applications. Search problems of academic and practical relevance are highly complex; for instance “if all atoms in the universe had been computing chess moves at a picosecond rate since the big bang (if any), the analysis would be just getting started” [1]. Consequently, the design of well-thought search methods has been at the heart of the artificial intelligence research [2]. In particular, hybrid metaheuristics [3] have proved to be very effective for solving hard search problems [3].

The design of hybrid metaheuristics is not a “free lunch” as it comes with a high price: it requires expertise in many areas and a lot of man-hours. Recently, there has been a trend in automating the design of search methods [4]. However, the automated design of hybrid metaheuristics has not received the attention it deserves. In this study, we present an automated approach which combines a meta-genetic algorithm with orthogonal arrays to design hybrid metaheuristics out of standard metaheuristics (iterated local search, simulated annealing, and memetic algorithm). The meta-genetic algorithm evolves sequences of the metaheuristics and uses the orthogonal arrays to sample the parameter space in order to perform the parameter tuning as a subtask of the design process.

Our research findings have confirmed that the automatically designed solvers (ADS) perform competitively with the best manually designed methods (from the literature) for the aircraft landing problem which is of an academic and practical interest. Further, the ADS outperforms the standard metaheuristics that are configured automatically by irace [5].

The high-performance computing (HPC) is performed using the Java programming language (java.util.concurrent package) and the concurrency functionality of irace. The cluster is used: two nodes are selected; each with 24 cores (totaling 48 cores). The memory usage is insignificant as our application is not memory-intensive; however, it is computationally demanding. The computational challenge relies on the fact that a large number of configurations need to be evaluated in order to design a high-quality hybrid solver. Each configuration requires a significant amount of time to be evaluated on a set of training instances. Moreover, the underlying problem is NP-hard which scales exponentially with the input size. On the cluster, the jobs take from a few hours to more than 40 hours to complete and without the cluster, one job can take up to several days.

References

- [1] Whitley, D. (1994). A genetic algorithm tutorial. *Statistics and computing*, 4(2), 65-85.
- [2] Burke, E. K., & Kendall, G. (2005). *Search methodologies* (pp. 1-17). Springer Science+Business Media, Incorporated.
- [3] Talbi, E. G. (2002). A taxonomy of hybrid metaheuristics. *Journal of heuristics*, 8(5), 541-564.
- [4] Hutter, F., Hoos, H. H., Leyton-Brown, K., & Stützle, T. (2009). ParamILS: an automatic algorithm configuration framework. *Journal of Artificial Intelligence Research*, 36, 267-306.
- [5] López-Ibáñez, M., Dubois-Lacoste, J., Stützle, T., & Birattari, M. (2011). The irace package, iterated race for automatic algorithm configuration. Technical Report TR/IRIDIA/2011-004, IRIDIA, Université Libre de Bruxelles, Belgium.

Presenter Biography

Ahmed Hassan finished his B.Sc (Hons) in Mathematics with First Class at Sudan University of Science & Technology. In 2011, he joined African Institute for Mathematical Sciences and completed his postgraduate diploma in 2012. In 2014, He awarded M.Sc. in Mathematics with Distinction from Stellenbosch University. Now, he is doing his Ph.D. at the University of KwaZulu-Natal under the supervision of Prof. Nelishia Pillay.

Primary authors: HASSAN, Ahmed (University of KwaZulu-Natal); PILLAY, Nelishia (University of Pretoria)

Presenter: HASSAN, Ahmed (University of KwaZulu-Natal)

Session Classification: Poster session

Track Classification: HPC Techniques and Computer Science