Online Learning of Deeper Variable Ordering Heuristics for Constraint Optimisation Problems

Doolaard, F.P.; Yorke-Smith, N.

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Online Learning of Deeper Variable Ordering Heuristics for Constraint Optimisation Problems

Floris Doolaard and Neil Yorke-Smith[0000−0002−1814−3515]

`f.p.doolaard@student.tudelft.nl,n.yorke-smith@tudelft.nl`

Delft University of Technology, The Netherlands

Abstract. Solvers for constraint optimisation problems exploit variable and value ordering heuristics. Numerous expert-designed heuristics exist, while recent research uses machine learning to learn novel heuristics. We introduce the concept of *deep heuristics*, a data-driven approach to learn extended versions of a given variable ordering heuristic. We demonstrate deep variable ordering heuristics based on the smallest, anti first-fail, and maximum regret heuristics. The results show that deep heuristics solve 20% more problem instances than classical 'shallow' heuristics.

1 Motivation and Approach

The order in which the variables are chosen can have significant effect on the total runtime of a constraint optimisation problem solver [3]. We address the situation of online solving of unseen optimisation problems. We introduce *deep variable ordering heuristics*, approximation functions that look at multiple levels of a search tree with the aim of generalizing better than classical heuristics.

As summarised in Figure 1, we implement deep heuristics in the open source Gecode solver [5]. Given a problem instance, an initial probing phase employs pseudo-random search to gather a variety of variable-value assignments. This data is then utilised by the machine learning component to acquire a deep heuristic function. Then second, during solving, given the current search state, the solver can predict scores with the learned model and select the variable with the best predicted score. Third, to leverage the pseudo-random nature of the probing data, a restart-based search strategy allows for multiple ML models to be learned, increasing the chance of finding solutions.

Chu and Stuckey [1] use online learning to acquire value heuristics: we learn variable ordering heuristics and we utilise a more complex score function. We use deeper lookaheads than Glankwamdee and Linderoth [4], and exploit ML predictions to circumnavigate the cost of lookaheads during search.

2 Results and Discussion

We test deep heuristics on four representative problem classes from the MiniZinc benchmarks: Resource Constrained Project Scheduling Problem (RCPSP), Evilshop, Amaze, Open Stacks. Instances are run for a maximum time of 4 hours.
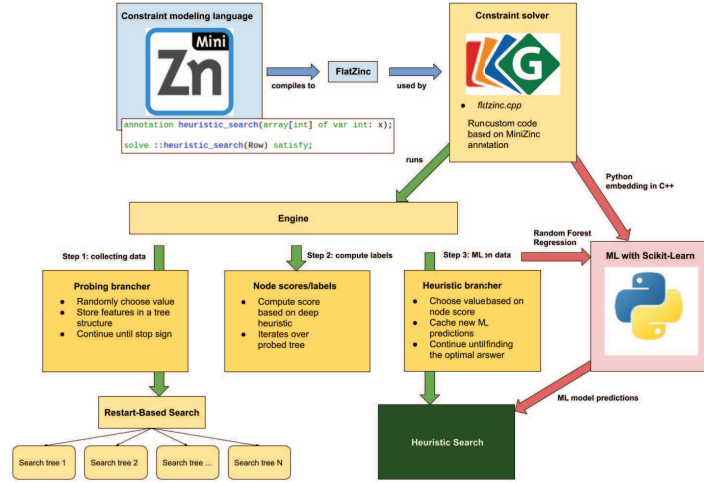
2   Doolaard, Yorke-Smith

Fig. 1: Probing, learning, and heuristic search phases implemented in Gecode.

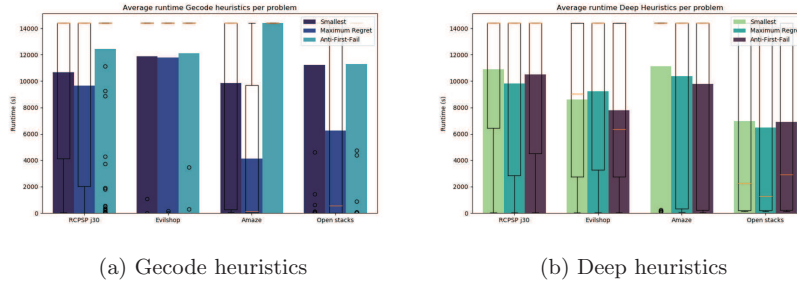(a) Gecode heuristics   (b) Deep heuristics

Fig. 2: Comparison of mean runtime between heuristics

Results, such as shown in Figure 2, indicate that the deep heuristics often – but not always – outperform the 'classical' version of the heuristics. For the deep heuristics, the runtime includes the probing and training time, as well as the solving time. Full results are found in the thesis [2]. Overall we find that deep heuristics solve 20% more problem instances, while improving on total runtime for the Open Stacks and Evilshop benchmark problems.

The thesis provides a novel approach to one-shot learning of search heuristics for constraint optimisation problems. Further experiments are warranted to assess the contribution of each the parts of our approach. In particular, recognising the stochasticity inherent in a learning-based approach, we use restarts with the deep heuristics – but not with their classical counterparts.

References

[1] Chu, G., Stuckey, P.J.: Learning value heuristics for constraint programming. In: Proceedings of the 12th International Conference on the Integration of AI and OR Techniques in Constraint Programming (CPAIOR'15), p. 108–123 (2015)

[2] Doolaard, F.: Deepification: Learning Variable Ordering Heuristics in Constraint Optimization Problems. Masters thesis, Delft University of Technology (2020)

[3] Gent, I.P., MacIntyre, E., Prosser, P., Smith, B.M., Walsh, T.: An empirical study of dynamic variable ordering heuristics for the constraint satisfaction problem. In: Proceedings of 2nd International Conference on Principles and Practice of Constraint Programming (CP'96), pp. 179–193 (1996)

[4] Glankwamdee, W., Linderoth, J.: Lookahead branching for mixed integer programming. Tech. rep., Lehigh University (Oct 2006)

[5] Schulte, C., Tack, G., Lagerkvist, M.Z.: Modeling and Programming with Gecode 6.2.0 (2019), URL www.gecode.org