Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)

Solving Hard Subgraph Problems in Parallel

Ciaran McCreesh*
University of Glasgow, Glasgow, Scotland
c.mccreesh.1@research.gla.ac.uk

Abstract

We look at problems involving finding subgraphs in larger graphs, such as the maximum clique problem, the subgraph isomorphism problem, and the maximum common subgraph problem. We investigate variable and value ordering heuristics, different inference strategies, intelligent backtracking search (backjumping), and bit- and thread-parallelism to exploit modern hardware.

1 Overview

We are looking at problems which involve finding subgraphs inside larger graphs. The *maximum clique* problem is to find a largest complete subgraph inside a given graph. More generally, the *subgraph isomorphism* family of problems involve finding a specified pattern graph inside a larger target graph. More generally still, the *maximum common subgraph* problem is to find a largest subgraph common to two or more given graphs. These problems are NP-hard, but we can often solve instances quickly in practice, even for graphs with up to tens of thousands of vertices. Algorithms for these problems have been used successfully in areas including bioinformatics and chemistry, computer vision, law enforcement, model checking, compilers, and pattern recognition.

Ultimately, all of these problems are solved using some form of backtracking search: we try to build up an assignment of values to variables, satisfying a set of constraints. To obtain the best results, we need to select the right kind of preprocessing, inference, learning, and search.

2 Our Contributions

When branching, selecting a good variable and value can have a huge impact on search: with good heuristics, most instances that occur in practice are reasonably easy to solve.

Modern clique algorithms often use some variation of an integrated colouring-based bound and ordering heuristic [Tomita *et al.*, 2010]. We have determined why this particular branching strategy works so well in practice [McCreesh and Prosser, 2014b]. Our results suggest that it is because this strategy tends to branch on small colour classes first, which is similar to the widely-used smallest domain first heuristic (and this contradicts the explanation given previously in the literature). This knowledge allowed us to design a better branching strategy, which explicitly considered the sizes of colour classes.

For subgraph isomorphism, we observed that many existing benchmark suites contain large numbers of very easy satisfiable instances. We have shown how to generate "really hard" random instances for subgraph isomorphism problems [McCreesh *et al.*, 2016], and explained how this lets us design better variable and value ordering heuristics. This work builds upon the satisfiable-unsatisfiable phase transition phenomena commonly seen in NP-complete problems, but because of the larger number of instance generation parameters, much richer behaviour is observed. Interestingly, our results suggest that constrainedness [Gent *et al.*, 1996], not proximity to a phase transition, determines whether an instance is hard in practice.

Since we are looking for good performance, it is important to design algorithms with modern hardware features in mind. Current CPUs have several (or dozens of) processing cores, and it is wasteful not to exploit this. We have shown that it is possible to get very good speedups by using thread-parallel tree search for the maximum clique problem [McCreesh and Prosser, 2013]. To explain why our approach worked so well (and why randomised work stealing does not), we then looked in more detail at how parallel search strategies interact with search. We show that there is a connection between parallel tree-search and value ordering heuristic behaviours: by explicitly stealing work early in search, we can offset poor early value ordering heuristic choices [McCreesh and Prosser, 2015c], giving an alternative to discrepancy searches. An additional benefit of this approach is that it guarantees that adding more processing cores will never make behaviour worse [Trienekens, 1990], and that parallel runtimes are reproducible.

We brought all of this together to develop a new algorithm for the subgraph isomorphism problem [McCreesh and Prosser, 2015a]. We combined bit-parallel filtering, a new all-different propagator, a stronger form of inference via preprocessing, and a different way of performing backjumping that does not need to maintain conflict sets or modify propagators. We showed how backjumping may be parallelised safely and effectively, by treating the branching as a lazy fold with left-zero elements.

Our results indicated that different instances would benefit

*This work was supported by the Engineering and Physical Sciences Research Council [grant number EP/K503058/1]

from different strengths of filtering. We demonstrated that it is possible to use using machine learning to select, on an instance-by-instance basis, a good choice from a portfolio of subgraph isomorphism algorithms [Kotthoff *et al.*, 2016].

We have also investigated generalisations of these problems: the balanced induced biclique problem [McCreesh and Prosser, 2014a] requires symmetry breaking, and the maximum labelled clique problem is a two-pass multi-objective problem with additional side constraints [McCreesh and Prosser, 2015b].

We are currently investigating the maximum weight clique (or independent set) problem. The colouring bound used in the maximum clique problem is not usable with large weights. We have seen favourable initial results using decision diagram bounds [Bergman *et al.*, 2014b], although variable ordering heuristics behave unexpectedly compared to in conventional tree-search. We also believe that parallel search is likely to be beneficial here, and that existing work on safe and reproducible parallel search should generalise to parallel decision diagrams [Bergman *et al.*, 2014a].

We have preliminary results on the maximum common subgraph problem. The best existing approaches fall into one of two categories: backtracking search with strong inference based around an enhanced all-different propagator, [Ndiaye and Solnon, 2011], or by reduction to the maximum clique problem. We believe a hybrid approach is possible.

Finally, we intend to investigate stronger conflict analysis for these algorithms, and to move towards a clause-learning type approach. Combining SAT-style clause learning with graph-based search has been shown to be effective for colouring problems [Zhou *et al.*, 2014]. We would like to understand why, and how to generalise this to subgraph isomorphism problems (ideally without having to reduce to CNF, which loses high level structural information).

References

[Bergman *et al.*, 2014a] David Bergman, André A. Ciré, Ashish Sabharwal, Horst Samulowitz, Vijay A. Saraswat, and Willem Jan van Hoeve. Parallel combinatorial optimization with decision diagrams. In *Integration of AI and OR Techniques in Constraint Programming - 11th International Conference, CPAIOR 2014, Cork, Ireland, May 19-23, 2014. Proceedings*, pages 351–367, 2014.

[Bergman *et al.*, 2014b] David Bergman, André A. Ciré, Willem Jan van Hoeve, and John N. Hooker. Optimization bounds from binary decision diagrams. *INFORMS Journal on Computing*, 26(2):253–268, 2014.

[Gent *et al.*, 1996] Ian P. Gent, Ewan MacIntyre, Patrick Prosser, and Toby Walsh. The constrainedness of search. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence and Eighth Innovative Applications of Artificial Intelligence Conference, AAAI 96, IAAI 96, Portland, Oregon, August 4-8, 1996, Volume 1.*, pages 246–252, 1996.

[Kotthoff *et al.*, 2016] Lars Kotthoff, Ciaran McCreesh, and Christine Solnon. Portfolios of subgraph isomorphism algorithms. 2016. To appear at LION.

[McCreesh and Prosser, 2013] Ciaran McCreesh and Patrick Prosser. Multi-threading a state-of-the-art maximum clique algorithm. *Algorithms*, 6(4):618–635, 2013.

[McCreesh and Prosser, 2014a] Ciaran McCreesh and Patrick Prosser. An exact branch and bound algorithm with symmetry breaking for the maximum balanced induced biclique problem. In *Integration of AI and OR Techniques in Constraint Programming - 11th International Conference, CPAIOR 2014, Cork, Ireland, May 19-23, 2014. Proceedings*, pages 226–234, 2014.

[McCreesh and Prosser, 2014b] Ciaran McCreesh and Patrick Prosser. Reducing the branching in a branch and bound algorithm for the maximum clique problem. In *Principles and Practice of Constraint Programming - 20th International Conference, CP 2014, Lyon, France, September 8-12, 2014. Proceedings*, pages 549–563, 2014.

[McCreesh and Prosser, 2015a] Ciaran McCreesh and Patrick Prosser. A parallel, backjumping subgraph isomorphism algorithm using supplemental graphs. In *Principles and Practice of Constraint Programming - 21st International Conference, CP 2015, Cork, Ireland, August 31 - September 4, 2015, Proceedings*, pages 295–312, 2015.

[McCreesh and Prosser, 2015b] Ciaran McCreesh and Patrick Prosser. A parallel branch and bound algorithm for the maximum labelled clique problem. *Optimization Letters*, 9(5):949–960, 2015.

[McCreesh and Prosser, 2015c] Ciaran McCreesh and Patrick Prosser. The shape of the search tree for the maximum clique problem and the implications for parallel branch and bound. *TOPC*, 2(1):8, 2015.

[McCreesh *et al.*, 2016] Ciaran McCreesh, Patrick Prosser, and James Trimble. Heuristics and really hard instances for subgraph isomorphism problems. 2016. To appear at IJCAI.

[Ndiaye and Solnon, 2011] Samba Ndojh Ndiaye and Christine Solnon. CP models for maximum common subgraph problems. In *Principles and Practice of Constraint Programming - CP 2011 - 17th International Conference, CP 2011, Perugia, Italy, September 12-16, 2011. Proceedings*, pages 637–644, 2011.

[Tomita *et al.*, 2010] Etsuji Tomita, Yoichi Sutani, Takanori Higashi, Shinya Takahashi, and Mitsuo Wakatsuki. A simple and faster branch-and-bound algorithm for finding a maximum clique. In Md.Saidur Rahman and Satoshi Fujita, editors, *WALCOM: Algorithms and Computation*, volume 5942 of *Lecture Notes in Computer Science*, pages 191–203. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[Trienekens, 1990] Harry W.J.M. Trienekens. *Parallel Branch and Bound Algorithms*. PhD thesis, Erasmus University Rotterdam, 1990.

[Zhou *et al.*, 2014] Zhaoyang Zhou, Chu-Min Li, Chong Huang, and Ruchu Xu. An exact algorithm with learning for the graph coloring problem. *Computers & Operations Research*, 51:282 – 301, 2014.