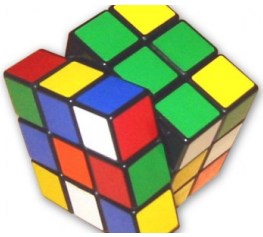

Exploiting Computer Algebra for Efficient Symmetry Breaking

Bart Bogaerts (AI Lab, DINF) & Jan De Beule (DIMA, WIDS)

April 24, 2020

Since, and largely thanks to the introduction of conflict-driven clause learning [2], the field of satisfiability solving (SAT) has seen a tremendous increase both in solver efficiency and application areas. While in practice these solvers perform very well, there are classes of applications on which they get hopelessly lost.

An example of such a problem is the n-pigeonhole problem "do n pigeons fit in $n-1$ holes?". Even for n as small as 10, most out-of-the-box SAT solvers are unable to solve this problem in reasonable time (an hour).

To overcome this limitation, we recently introduced an automatic symmetry breaking preprocessor for SAT [1], which achieved very good performance on SAT competitions. It takes a SAT instance and detects symmetries using a graph automorphism tool. Next, it adds extra constraints to the instance; these extra constraints eliminate the symmetries by forbidding the (lexicographically) smaller of every two symmetric assignments. The main contribution of our symmetry breaker was the introduction of a concept called "row interchangeability". Intuitively, row interchangeability corresponds to interchangeability of a set of objects at a high level of the specificiation (in our example for instance to the interchangeability of pigeons). What we showed was that for this specific type of symmetries, the entire symmetry group (which is exponentially large in terms of the number of generators) can be broken by a linear number of clauses (again, in terms of the number of generators of the symmetry group).

One question that remained open is: "how can we detect row interchangeability?". In our symmetry breaking tool, we developed ad-hoc heuristic algorithms for finding this structure, but this is unsatisfactory both from a theoretical perspective as from a practical perspective (we have no guarantees whatsoever that row interchangeability will be found if present). Our hypothesis is that the use of computer algebra tools would allow for more effective and more theoretically substantiated methods to detect this type of symmetry. The goal of this thesis is to test that hypothesis and, if successful, test the efficiency of these methods.

Additional information This thesis is a collaboration between Bart Bogaerts from the computer science department, who will provide the expertise on

symmetry breaking and Jan De Beule from the mathematics department, who will provide the expertise on computer algebra tools. Feel free to drop by our offices any time for information on this topic.

Promotor & contact
Bart Bogaerts: bart.bogaerts@vub.be
Jan De Beule jan.de.beule@vub.be

References

[1] Jo Devriendt, Bart Bogaerts, Maurice Bruynooghe, and Marc Denecker. Improved static symmetry breaking for SAT. In Nadia Creignou and Daniel Le Berre, editors, *Theory and Applications of Satisfiability Testing - SAT 2016 - 19th International Conference, Bordeaux, France, July 5-8, 2016, Proceedings*, volume 9710 of *Lecture Notes in Computer Science*, pages 104–122. Springer, 2016.

[2] João P. Marques-Silva and Karem A. Sakallah. GRASP: A search algorithm for propositional satisfiability. *IEEE Transactions on Computers*, 48(5):506–521, 1999.