Speeding up SAT solver by exploring CNF symmetries: Revisited
Arup Kumar Ghosh, University of Central Florida

Boolean Satisfiability solvers have gone through dramatic improvements in their performances and scalability over the last few years by considering symmetries. It has been shown that by using graph symmetries and generating symmetry breaking predicates (SBPs) it is possible to break symmetries in Conjunctive Normal Form (CNF). The SBPs cut down the search space to the non-symmetric regions of the space without affecting the satisfiability of the CNF formula. The symmetry breaking predicates are created by representing the formula as a graph, finding the graph symmetries and using some symmetry extraction mechanism (Crawford et al.). Here in this paper we take one non-trivial CNF and explore its symmetries. Finally, we generate the SBPs and adding it to CNF we show how it helps to prune the search tree, so that SAT solver would take short time. Here we present the pruning procedure of the search tree from scratch, starting from the CNF and its graph representation. As we explore the whole mechanism by a non-trivial example, it would be easily comprehensible. Also we have given a new idea of generating symmetry breaking predicates for breaking symmetry in CNF, not derived from Crawford’s conditions.

Keywords: Symmetry, SBP, Satisfiability, SAT solver