Popularity and Similarity in SAT

Jordi Levy
IIIA-CSIC
levy@iiia.csic.es

Satisfiability of propositional formulas in disjunctive normal form (SAT) is a fundamental problem in computer science. SAT was the first problem proved to be NP-complete, and defines this class of complexity (roughly speaking, problems that can be solved guessing a possible solution, and checking it in polynomial time). Whether P (the class of problems solvable in polynomial time) and NP are equal is probably the most significant open question in computer science, and it is included the Millennium Prize Problems.

Besides its theoretical interest, SAT has also a practical interest. Many real-world problems in distinct areas as circuit verification, software verification, planning, scheduling, optimization, etc... may be efficiently *reduced* (translated) into SAT. Unless P=NP, the best algorithms to solve SAT will need exponential time, in the worst case. Paradoxically, a community of computer scientist have been able to develop *SAT solvers* able to solve in a few seconds real-world (or industrial) problems with thousands of variables and millions of clauses (constraints). The explanation relies in the structure of these real-world problems. We can represent this structure as a bi-partite graph (or network) where nodes are variables and clauses, and every edge represents the presence of a variable in a clause.

Every year, the SAT community celebrates competitions where SAT solvers are evaluated on hundreds of SAT benchmarks coming from distinct industrial areas. This has contributed to improve the performance of SAT solvers, many times following an strategy of trial-and-error. We have analyzed the structure of SAT problems included in these benchmarks, trying to characterize the properties that make them easy for SAT solvers. We found that most problems have a scale-free structure (in both variable and clause nodes). However, most theoretical result on random SAT assume a uniform probability distribution for variables and clauses. We also found that the modularity of real-world problems is very high. In fact the modularity of a problem is now used as a predictor for the hardness of a problem. Many real-world SAT problems are also self-similar with a low fractal dimension.

Considering the scale-free structure of real-world SAT problems, a natural strategy could be assigning first the variables that occur more frequently. This would reduce the size of the problem significantly. Although, variants of this strategy were used in the past, nowadays modern SAT solvers use strategies that try to focus the search inside an *area* of the problem. We use the hyperbolic geometric network model to generate random SAT problems, and analyze the behavior of most efficient SAT solvers on them. We show that modern SAT solvers exploit, not only the popularity of variables, but also its similarity. This way, they focus the instantiation of variables to those that are closer and most popular, the ones that are more promising in order to reflect before an incoherence and prune the search more efficiently.

Keywords: SAT, Constraint Solving, Hyperbolic Geometric Networks.