CircleSAT for SAT-Race 2010

Jingchao Chen

School of Informatics, Donghua University
2999 North Renmin Road, Songjiang District, Shanghai 201620, P. R. China
chen-jc@dhu.edu.cn

CircleSAT is a hybrid SAT solver that improves the MoRsat solver [1]. As MoRsat, its basic framework is based on a look-ahead technique. Its core search engine is a conflict-driven DPLL solver. The main difference between MoRsat and CircleSAT is that MoRsat is built on the top of Rsat [2], while CircleSAT is built on the top of the optimized PrecoSAT [3], which is the Gold Medal winner in the application category of the SAT 2009 competition [4]. CircleSAT consists of the following parts.

(1) A preprocessor for simplifying SAT instances.
(2) A switching mechanism for transforming between a look-ahead DPLL search and a conflict-driven DPLL search.
(3) A look-ahead DPLL search engine.
(4) A conflict-driven DPLL search engine.
(5) A middle simplifier for simplifying dynamically the SAT instance via a conflict-driven DPLL solver.

Although SATeLite [5] is a very successful preprocessor, we do not adopt it. This is because SATeLite has not the detection of XOR constraints. The preprocessor of CircleSAT contains the detection of XOR constraints. Like the preprocessor of the March solver [6], it can remove inactive variables in XOR constraints, and shorten XOR constraints. Notice, our XOR constraint detecting borrows the idea of [1], which is the simplest so far.

As a hybrid SAT solver, it is very important to how to switch from a look-ahead DPLL search to a conflict-driven DPLL search and vice versa. The switching mechanism of MoRsat is very rough. Its switching is based on the number of variables and clauses in a SAT instance, and the initial result given by a conflict-driven DPLL search. CircleSAT refines the switching mechanism of MoRsat. It makes use of information on search depth provided by a conflict-driven DPLL procedure to switch between the look-ahead DPLL search and the conflict-driven DPLL search.

The look-ahead DPLL search engine in MoRsat is much simpler than March [6]. So it does not contain any binary resolvent. To be able to improve the efficiency of solving some SAT instances such as the DIMACS 32-bit parity problem, in CircleSAT, we add an implied binary resolving, and maintain dynamically the database of binary resolvents.

The efficiency of CircleSAT depends heavily on a conflict-driven DPLL search engine. Choosing a good conflict-driven DPLL solver is very important. We select PrecoSAT as the conflict-driven DPLL search engine of CircleSAT. In order to meet some special demands of CircleSAT, we optimize PrecoSAT and make some changes. The detailed optimizations include:

(1) PrecoSAT cannot perform XOR reasoning. We incorporate XOR reasoning in it.
(2) PrecoSAT is a pure conflict-driven DPLL search. We add a look-ahead DPLL search in it. The improved PrecoSAT is also a hybrid SAT solver. The look-ahead DPLL search of CircleSAT is divided into two categories: external and internal. What the improved PrecoSAT applies is the internal look-ahead DPLL search, while what the framework of CircleSAT applies is the external look-ahead DPLL search.
(3) The move-to-front policy is adopted extensively in PrecoSAT. However, our improved PrecoSAT does not use fully the move-to-front policy when updating the watch literal. What we use is a half move-to-front policy.

The old PrecoSAT cannot solve instances such as mizh-sha0 [4], while the improved PrecoSAT can do it, though it is slow. In addition, CircleSAT introduces an idea called the group-based solving technology. This technology is to partition the variables into some groups. In each group, we construct a sub-problem. The middle result (not a partial solution) of each sub-problem is obtained by a conflict-driven DPLL solver. By integrating the middle results, we can solve more easily the full solution of the original problem.

References

[1] Chen, J.C.: Building a Hybrid SAT Solver via Conflict-driven, Look-ahead and XOR Reasoning Techniques, SAT 2009.

[2] Pipatssrisawat, K., Darwiche, A.: Rsat 2.0: SAT solver description, in SAT 2007 competition.

[3] Biere A., P{re,i}coSAT@SC'09, SAT 2009 competitive events booklet, p.41.

[4] SAT'09 Competition Homepage. https://www.cril.univ-artois.fr/SAT09

[5] Een, N., Sorensson, N.: An extensible SAT-solver. SAT 2003.

[6] Heule, M., Van Maaren, H.: Aligning CNF- and Equivalence-Reasoning, SAT 2004, LNCS 3542, 145-156, 2005.