SAT Solvers: CMUSAT, CMUSAT-AIG, NFLSAT-AIG

Himanshu Jain
CMU SCS, Pittsburgh, PA 15213

ABSTRACT

This articles describes the main features of three Boolean satisfiability (SAT) solvers CMUSAT, CMUSAT-AIG, and NFLSAT-AIG.

1. CMUSAT

CMUSAT is a Boolean satisfiability checker for formulas expressed in conjunctive normal form (CNF). Details about CMUSAT are as follows:

- Written in C++ and submitted as a 64-bit binary.
- Written from scratch using STL.
- Uses pre-processing. We use MiniSat2 code [2] for pre-processing.
- The new ideas used in CMUSAT are described in our SAT competition 2007 description of CMUSAT [4]. The CMUSAT version submitted to SAT-Race 2008 adds RSAT style phase saving and restarts [3].

2. CMUSAT-AIG

CMUSAT-AIG is a Boolean satisfiability checker for formulas expressed in And Inverter Graph (AIG) format. Details about CMUSAT-AIG are as follows:

- Written in C++ and submitted as a 64-bit binary.
- The input AIG is converted to an equi-satisfiable CNF formula using `aigtocnf` [1]. The resulting CNF formula is then checked for satisfiability by CMUSAT solver.

3. NFLSAT-AIG

NFLSAT-AIG is a Boolean satisfiability checker for formulas expressed in And Inverter Graph (AIG) format. Details about NFLSAT-AIG are as follows:

- Written in C++ and submitted as a 64-bit binary.
- Written from scratch.
- Does not use any pre-processing.
- Converts the input AIG to Negation Normal Form (NNF). The NNF of a formula is usually more succinct than the Conjunctive Normal Form (CNF) of the formula. The DPLL algorithm is then applied to the graph based representations of NNF formulas. A technical report describing NFLSAT-AIG will be made available soon.

4. REFERENCES

[1] AIGER format, http://fmv.jku.at/aiger/.
[2] MiniSat, www.cs.chalmers.se/cs/research/formalmethods/minisat/.
[3] RSAT, http://reasoning.cs.ucla.edu/rsat/.
[4] SAT 2007 CMUSAT Description. `http://www.cs.cmu.edu/~hjain/papers/cmusat-solvers.pdf`.