SAT-Solver

Tao Lin, EECS, PKU, 1600012773

July 17, 2017

DPLL Algorithm

In every step, continuously apply BCP and SET_PURE_TRUE to simplying the formula, then choose one variable, respectively set it to True and False and recursively move to next step, until a conflict occurs or the formula is satisfied.

Data Structure

1. a list F of sets to represent the CNF. $F[j]$ is the jth clause, which is a set of literals.
2. a set I of indexes to record the left clauses. $I = \{j|F[j] \neq \top\}$.
3. a hash map S which maps literal x to the clauses x is in, implemented by a dict. $S[x] = \{j|x \in F[j]\}$
4. a set T of literals that can be True. In BCP, every unit clause provides such a literal, $T_1 = \{x|\exists j, |F[j]| = 1, x \in F[j]\}$; in SPT, consider x and its negation $-x$, $T_2 = \{x||S[x]| > 0, |S[-x]| = 0\}$.$T = T_1 \cup T_2$.

Key Ideas

1. **See BCP and SPT as a whole.** Except for choosing variables, they treat variables the same way: set to True or False, until no variable can be determined. T is shared by them.
2. **Choose variables greedily and randomly.** Greedily: clauses of smaller size are more decisive and variables occuring more often are more influential. Therefore, from the smallest clauses, choose a literal with the greatest $\alpha|S[-x]|+\beta|S[x]|$. α and β are weights, respectively set to 1.0 and 1.0 based on experiment. Randomly: experiment reveals that when there are too many clauses sharing the smallest size, greedy method fails, so randomly pick a literal instead. Set the chosen literal to True first.
3. **Modify F, I, S, T dynamically in assignment.** When setting literal x_0 to True, remove all the clauses which contain x_0, by deleting indexes from I and from $S[x]$ for x in deleted clauses; remove $-x_0$ from its clauses. Use the size of $F[j]$ to decide unit clause, use the sizes of $S[x]$ and $S[-x]$ to find pure literal.
4. **Undo assignment.** The opposite of assignment. Undo assignment to regain F, I, S instead of to reconstruct them. After assigning x_0, $F[j]$ that contain x_0, $S[x_0]$ and $S[-x_0]$ will never change, so they can be used when undoing. After a failing new step, undo the decision. BCP and SPT should record the sequence of chosen literals, and undo in opposite order before return.
5. **Deal with T.** The set T of literals which can be True is ignored when undoing. After BCP and SPT, T becomes empty. After assigning x_0 and in the new step, T changes. When assigning $-x_0$, T is useless, thus, empty T before assigning $-x_0$.
6. **Return.** Return Unsat if $F[j] = \{\}$ is found when setting literals; or contradictory assignment is found in BCP and SPT. Return Sat if BCP and SPT return an empty I.