CS395T, Spring 2011

The Davis-Putnam-Logemann-Loveland Procedure

A *literal* is an atom or a negated atom. A *clause* is a disjunction of literals (possibly the empty disjunction $\bot$). A formula is said to be in *conjunctive normal form (CNF)* if it is a conjunction of clauses (possibly the empty conjunction $\top$).

Many existing SAT solvers are based on the Davis-Putnam-Logemann-Loveland procedure, or DPLL [Davis and Putnam, 1960, Davis *et al.*, 1962]. It allows us to decide whether a CNF formula is satisfiable, and to find a satisfying interpretation if it is.

For any CNF formula F and atom A, $F|_A$ stands for the formula obtained from F by replacing all occurrences of A by $\top$ and simplifying the result by removing

- all clauses containing the disjunctive term $\top$, and
- the disjunctive terms $\neg\top$ in all remaining clauses.

Similarly, $F|_{\neg A}$ is the result of replacing A in F by $\bot$ and simplifying the result. For instance,

$$(p \vee q \vee \neg r) \wedge (\neg p \vee r)|_{\neg p} \;=\; q \vee \neg r.$$

If a CNF formula F contains a clause that consists of a single literal ("unit clause") then F can be simplified using the procedure called *unit propagation* (Figure 1). In this procedure, U is a set of literals that does not contain complementary pairs A, $\neg A$. To apply unit propagation to a given CNF formula F_0, Unit-Propagate is invoked with $F = F_0$ and

---

Unit-Propagate(F, U)
    while F contains no empty clause but has a unit clause L
        $F \leftarrow F|_L$;
        $U \leftarrow U \cup \{L\}$
    end

Figure 1: Unit propagation

$U = \emptyset$. After every execution of the body of the loop, the conjunction of F with the literals U remains equivalent to F_0.

For instance, to apply unit propagation to

$$p \wedge (\neg p \vee \neg q) \wedge (\neg q \vee r)$$

we invoke UNIT-PROPAGATE with this formula as F and with $\emptyset$ as U. After the first execution of the body of the loop,

$$F = \neg q \wedge (\neg q \vee r) \text{ and } U = \{p\};$$

after the second iteration

$$F = \top \text{ and } U = \{p, \neg q\}.$$

This computation shows that the given formula is equivalent to $p \wedge \neg q$.

There are two cases when the process of unit propagation alone is sufficient for solving the satisfiability problem for F_0. Consider the values of F and U upon the termination of UNIT-PROPAGATE. First, if $F = \top$, as in the example above, then F_0 is satisfiable, and a satisfying interpretation can be easily extracted from U. Second, if F contains the empty clause then F_0 is not satisfiable.

Problem 1. Use unit propagation to decide whether the formula

$$p \wedge (p \vee q) \wedge (\neg p \vee \neg q) \wedge (q \vee r) \wedge (\neg q \vee \neg r)$$

is satisfiable.

The Davis-Putnam-Logemann-Loveland procedure (Figure 2) is an extension of the unit propagation method that can solve the satisfiability

---

DPLL(F, U)
  UNIT-PROPAGATE(F, U);
  if F contains the empty clause **then** return;
  if $F = \top$ **then** exit with a model of U;
  $L \leftarrow$ a literal containing an atom from F;
  DPLL($F|_L, U \cup \{L\}$);
  DPLL($F|_{\overline{L}}, U \cup \{\overline{L}\}$)

Figure 2: Davis-Putnam-Logemann-Loveland procedure

problem for any CNF formula. Like UNIT-PROPAGATE, it is initially invoked with $F = F_0$ and $U = \emptyset$.

Consider, for instance, the application of the DPLL procedure to

$$(\neg p \lor q) \land (\neg p \lor r) \land (q \lor r) \land (\neg q \lor \neg r).$$

First DPLL is called with this formula as F and with $\emptyset$ as U (Call 1). After the call to UNIT-PROPAGATE, the values of F and U remain the same. Assume that the literal selected as L is p. Now DPLL is called recursively with

$$q \land r \land (q \lor r) \land (\neg q \lor \neg r)$$

as F and $\{p\}$ as U (Call 2). After the call to UNIT-PROPAGATE, F turns into the empty clause. Next DPLL is called with

$$(q \lor r) \land (\neg q \lor \neg r)$$

as F and $\{\neg p\}$ as U (Call 3). After the call to UNIT-PROPAGATE, F and U remain the same. Assume that the literal selected as L is q. Then DPLL is called with $\neg r$ as F and $\{\neg p, q\}$ as U (Call 4). After the call to UNIT-PROPAGATE, $F = \top$ and $U = \{\neg p, q, \neg r\}$. The computation produces an interpretation satisfying the given formula:

p	q	r
f	t	f

Problem 2. How would this computation be affected by selecting $\neg p$ as L in Call 1? By selecting $\neg q$ as L in Call 3?

References

[Davis and Putnam, 1960] Martin Davis and Hillary Putnam. A computing procedure for quantification theory. *Journal of ACM*, 7:201–215, 1960.

[Davis *et al.*, 1962] Martin Davis, George Logemann, and Donald Loveland. A machine program for theorem proving. *Communications of the ACM*, 5(7):394–397, 1962.