Tutorial on SAT Solvers

Combinatorial Problem Solving (CPS)

Enric Rodríguez-Carbonell

May 28, 2019

SAT Solvers

- SAT solvers take as input a CNF formula F and return:
  - `sat`(+ model): if F is satisfiable
  - `unsat`: if F is unsatisfiable
- We will be using `lingeling` (developed by Armin Biere)
- Usage: `lingeling [ <option> ... ] <input>`
- Some options:
  - `-s <seed>`: set random seed
  - `-o <output>`: set output file
  - `-h`: help, shows all options

Input Format: DIMACS (I)

- First some optional lines: `c␣<comment>`
- Then a line: `p␣cnf␣<num_vars>␣<num_clauses>`
- Then clauses:
  - Each variable is represented with an integer ≥ 1
  - Negated literals are negative integers
  - Literals in a clause separated by blank spaces
  - `0` marks the end of a clause

Input Format: DIMACS (II)

- $(x_1 \vee x_2) \wedge \neg x_3$

```
c This is an example of SAT formula
p cnf 3 2
1  2 0
-3 0
```

- $(x_1 \vee x_2) \wedge (x_1 \vee \neg x_2) \wedge (\neg x_1 \vee x_2) \wedge (\neg x_1 \vee \neg x_2)$

```
c This is an example of UNSAT formula
p cnf 2 4
 1  2 0
 1 -2 0
-1  2 0
-1 -2 0
```

Output Format

- 1st line is one of:
  - `s SATISFIABLE`
  - `s UNSATISFIABLE`

- If satisfiable, then comes a list of true literals.

  Each following line is of the form `v <list of lits>`

  Example: output for formula $(x_1 \vee x_2) \wedge \neg x_3$

```
s SATISFIABLE
v 1 2 -3 0
```

  Interpretation I with $I(x_1) = I(x_2) = 1, I(x_3) = 0$ is model