CS 512 Formal Methods, Spring 2017 Instructor: Assaf Kfoury

Lecture 7 Resolution in Propositional Logic & SAT Solver

February 14th, 2017 Tian Zhang

(These lecture notes are **not** proofread and proof-checked by the instructor.)

Genral Notes:

- Assignment 3 due Feb 15
- Assignment 3 posted Feb 14
- Be prepared to give a progress report before the Spring break
- Due to the cancelled classes, Prof. Kfoury decides to use take-home exam for midterm.

Founders:

- Martin Davis (1928 -), New York City
- Hilary Putnam (1926 - 2016), Harvard University
- George Logemann(1938 - 2012), New York City

Resolution Example:

Ex. $\dfrac{D \vee \neg P \qquad C \vee P}{D \vee C}$

$\neg C \to P \qquad P \to D;$

$\neg C \to D;$

$\neg\neg C \vee D;$

$C \vee D.$

Two important heuristics

- **Unit clause**: Prefer a resolution involving a unit clause (a clause with one literal), because it produces a shorter clause as a resolvent.

  Ex. $\dfrac{C \vee \neg x, \qquad x}{C}$

- **set-of-support rule**: Use the so-called set-of-support rule , i.e., prefer a resolution involving the negated goal or any clause derived from the negated goal, because we are trying to produce a contradiction that follows from the negated goal and these are the most relevant clauses.

Completeness

- **Strong Completeness**: If $\Gamma \vDash \psi$, then $\Gamma \vdash \psi$ (Natural Deduction, Tableau, Resolution)
- **Refutation Completeness**: If $\Gamma \vDash \perp$, then $\Gamma \vdash \perp$ (Tableau, Resolution)

3-SAT & 2-SAT

- **2-SAT** is a computational problem of assigning values to variables, each of which has two possible values, in order to satisfy a system of constraints on pairs of variables[1].

- **3-SAT**: NP-complete, it is used as a starting point for proving that other problems are also NP-hard.This is done by polynomial-time reduction from 3-SAT to the other problem. An example of a problem where this method has been used is the clique problem: given a CNF formula consisting of c clauses, the corresponding graph consists of a vertex for each literal, and an edge between each two non-contradicting literals from different clauses, cf. picture. The graph has a c-clique if and only if the formula is satisfiable[2].

Horn Formula

Deciding the truth of quantified Horn formulas can be done in polynomial time.

EX. $(x_1 \vee \neg x_2) \wedge (\neg x_1 \vee x_2 \vee x_3) \wedge \neg x_1$ is not a Horn Formula.By introducing y_3 as negation of x_3, it can be renamed to the Horn formula:

$(x_1 \vee \neg x_2) \wedge (\neg x_1 \vee x_2 \vee \neg y_3) \wedge \neg x_1$

In contrast, $(x_1 \vee \neg x_2 \vee \neg x_3) \wedge (\neg x_1 \vee x_2 \vee x_3) \wedge \neg x_1$ leads to Horn Formula.

XOR Satisfiability

XOR Truth Table

p	q	$p \quad XOR \quad q$
T	T	F
T	F	T
F	T	T
F	F	F

Two Main Approach to SAT Solver

- **Stochastic search**:
- **Exhaustive search**: SAT solvers based on exhaustive search use what is known as the DPLL procedure, or a refined and more efficient version of the original DPLL procedure.