Computational Thinking

Logic

Barnaby Martin

Department of Computer Science

Resolution for Propositional Logic

Durham University

Resolution

- Recall the rule of inference known as resolution

$$\frac{p \vee q \qquad\qquad \neg p \vee r}{q \vee r}$$

  – forms the basis of the proof system for propositional logic known as Resolution.

- However, the basic rule of Resolution is a more general one than that above

$$\frac{p_1 \vee \ldots \vee p_{i\text{-}1} \vee \boxed{x \vee} p_{i+1} \vee \ldots \vee p_m \qquad q_1 \vee \ldots \vee q_{i\text{-}1} \vee \boxed{\neg x} \vee q_{i+1} \vee \ldots \vee q_n}{p_1 \vee \ldots \vee p_{i\text{-}1} \vee p_{i+1} \vee \ldots \vee p_m \vee q_1 \vee \ldots \vee q_{i\text{-}1} \vee q_{i+1} \vee \ldots \vee q_n}$$

  – the p's and the q's are literals
    - that is, variables or negated variables (not necessarily distinct)
  – this is the *only* rule of Resolution.

The proof system Resolution

- Natural Deduction proves theorems starting from scratch, whereas
  - Resolution takes a given formula and works with it
    - in order to decide whether it is a theorem or not.
- In the proof system Resolution, we proceed as follows
  - we are given a propositional formula φ
  - we take $\neg\varphi$ and write it in c.n.f. as $C_1 \wedge C_2 \wedge \ldots \wedge C_m$
  - we start with the clauses $C_1, C_2, \ldots, C_m$
  - we continually apply the resolution rule of inference to infer *new* clauses
    - if ever we infer the empty clause $\emptyset$
      - then we halt and output that φ is a theorem
    - if we get to the point where we have not inferred the empty clause and we cannot infer any *new* clauses
      - then we halt and output that φ is not a theorem.
- We have one minor remark
  - when resolving, we are also allowed to delete repeated literals in any clause.
- Resolution is both sound and complete
  - if Resolution announces that φ is a theorem then φ is a tautology
  - if φ is a tautology then Resolution announces that φ is a theorem.

Resolution in action

- Consider the propositional formula φ

$$(((A \wedge W) \Rightarrow I) \wedge (\neg A \Rightarrow P) \wedge (\neg W \Rightarrow S) \wedge \neg I \ \wedge (D \Rightarrow (\neg P \wedge \neg S))) \Rightarrow \neg D$$

$$(\quad \varphi_1 \quad \wedge \quad \varphi_2 \quad \wedge \quad \varphi_3 \quad \wedge \ \varphi_4 \ \wedge \quad \varphi_5 \quad) \Rightarrow \ \psi$$

$$\neg(\quad \varphi_1 \quad \wedge \quad \varphi_2 \quad \wedge \quad \varphi_3 \quad \wedge \ \varphi_4 \ \wedge \quad \varphi_5 \quad) \vee \ \psi$$

$$\neg(\varphi_1 \wedge \varphi_2 \wedge \varphi_3 \wedge \varphi_4 \wedge \varphi_5) \vee \psi \qquad \varphi$$

- So, $\neg\varphi$ is

$$\neg\varphi = \neg(\neg(\varphi_1 \wedge \varphi_2 \wedge \varphi_3 \wedge \varphi_4 \wedge \varphi_5) \vee \psi) = (\varphi_1 \wedge \varphi_2 \wedge \varphi_3 \wedge \varphi_4 \wedge \varphi_5) \wedge \psi$$

$$\neg\varphi \equiv ((A \wedge W) \Rightarrow I) \wedge (\neg A \Rightarrow P) \wedge (\neg W \Rightarrow S) \wedge \neg I \wedge (D \Rightarrow (\neg P \wedge \neg S))) \wedge D$$

$$\equiv (\neg(A \wedge W) \vee I) \wedge (A \vee P) \wedge (W \vee S) \wedge \neg I \wedge (\neg D \vee (\neg P \wedge \neg S) \wedge D$$

$$\equiv (\neg A \vee \neg W \vee I) \wedge (A \vee P) \wedge (W \vee S) \wedge \neg I \wedge (\neg D \vee \neg P) \wedge (\neg D \vee \neg S) \wedge D$$

- So, the set of clauses to which we apply Resolution is

$$\neg A \vee \neg W \vee I \qquad A \vee P \qquad W \vee S \qquad \neg I \qquad \neg D \vee \neg P \qquad \neg D \vee \neg S \qquad D$$

Resolution in action

- So, we have our set of clauses

  $\neg A \vee \neg W \vee \cancel{I}$ $\quad A \vee P$ $\quad W \vee S$ $\quad \cancel{\neg I}$ $\quad \neg D \vee \neg P$ $\quad \neg D \vee \neg S$ $\quad D$

  True

- Now we start resolving
  - $\neg A \vee \neg W$ $\quad$ $\neg A \vee \neg W \vee T$, $False \vee \neg I \Rightarrow \neg A \vee \neg W \vee False$ $= \neg A \vee \neg W$
  - $P \vee \neg W$ $\quad$ $\neg A \vee \neg W$, $A \vee P \Rightarrow P \vee \neg W$
  - $P \vee S$ $\quad$ $P \vee \neg W$, $W \vee S \Rightarrow P \vee S$
  - $\neg D \vee S$ $\quad$ $\neg D \vee \underline{\neg P}$, $\underline{P} \vee S \Rightarrow \neg D \vee S$
  - $\neg D \vee \neg D$ $\quad$ $\neg D \vee S$, $\neg D \vee \neg S \Rightarrow \neg D \vee \neg D$
  - $\neg D$ True
  - $\varnothing$ $\Uparrow$ no input will make φ to be false

so φ is a theorem

- and so a tautology

Resolution in action

- Let φ be the formula $((p \vee q) \wedge (\neg p \vee \neg q) \wedge (r \Rightarrow (p \wedge q))) \Rightarrow r$.
- So, $\neg\varphi$ is $\neg(((p \vee q) \wedge (\neg p \vee \neg q) \wedge (r \Rightarrow (p \wedge q))) \Rightarrow r)$

  $\equiv ((p \vee q) \wedge (\neg p \vee \neg q) \wedge (r \Rightarrow (p \wedge q))) \wedge \neg r$ $\quad \neg(\neg((p \vee q) \wedge (\neg p \vee \neg q) \wedge (r \Rightarrow (p \wedge q)) \vee r)$

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

  $\equiv (p \vee q) \wedge (\neg p \vee \neg q) \wedge (\neg r \vee p) \wedge (\neg r \vee q) \wedge \neg r$
- Hence, the set of clauses to which we apply Resolution is

  $p \vee q \qquad \neg p \vee \neg q \qquad \neg r \vee p \qquad \neg r \vee q \qquad \neg r$

Resolution in action

- Let φ be the formula $((p \vee q) \wedge (\neg p \vee \neg q) \wedge (r \Rightarrow (p \wedge q))) \Rightarrow r$.
- So, $\neg\varphi$ is $\neg(((p \vee q) \wedge (\neg p \vee \neg q) \wedge (r \Rightarrow (p \wedge q))) \Rightarrow r)$
  $\equiv ((p \vee q) \wedge (\neg p \vee \neg q) \wedge (r \Rightarrow (p \wedge q))) \wedge \neg r$
  $\equiv (p \vee q) \wedge (\neg p \vee \neg q) \wedge (\neg r \vee (p \wedge q)) \wedge \neg r$
  $\equiv (p \vee q) \wedge (\neg p \vee \neg q) \wedge (\neg r \vee p) \wedge (\neg r \vee q) \wedge \neg r$
- Hence, the set of clauses to which we apply Resolution is

  $p \vee q \qquad \neg p \vee \neg q \qquad \neg r \vee p \qquad \neg r \vee q \qquad \neg r$
- Now we start resolving
  - $\cancel{q \vee \neg q}$ we can ignore this clause as it will never yield a new clause
  - $\cancel{p \vee \neg p}$ we can ignore this clause as it will never yield a new clause
  - $\neg q \vee \neg r$
  - $\neg p \vee \neg r$
  - $\cancel{p \vee \neg r}$ we have this clause already
  - $\cancel{\neg r \vee \neg r}$ i.e., $\neg r$ and we have this clause already
  - $\cancel{\neg q \vee \neg r}$ we have this clause already
  - $\cancel{\neg r \vee \neg r}$ i.e., $\neg r$ and we have this clause already
  - no new clauses can be inferred.

so φ is not a theorem
- and not a tautology

Is Resolution the silver bullet?

- Resolution works by taking the negation of a formula φ we wish to prove true
  - and showing that this negation $\neg\varphi$ is unsatisfiable (in essence).

- One might be inclined to think (from our examples) that Resolution will always give a “quick” answer as to whether a formula is a tautology or not.

- However, this is not the case
  - for in the worst case Resolution involves an exponential number of applications.

Satisfiability vs. tautologies

Durham University

- SAT-solvers check whether or not a given formula of propositional logic is satisfiable
  - whereas proof systems, such as Resolution, aim to prove theorems.
- To some extent, these two tasks are different sides of the same coin.
- Let φ be some propositional formula.
  - If φ is satisfiable
    - then there exists a truth assignment making φ *true*
    - i.e., there exists a truth assignment making $\neg\varphi$ *false*
    - i.e., $\neg\varphi$ is not a tautology.
  - Conversely, if $\neg\varphi$ is not a tautology
    - then there exists some truth assignment making $\neg\varphi$ *false*
    - i.e., there exists some truth assignment making φ *true*
    - i.e., φ is satisfiable.
- So, φ is satisfiable if, and only if, $\neg\varphi$ is not a tautology
  - this leads to strong links between SAT-solving and automated theorem proving.

$a \vee b = a \wedge b$

$a=T,\ b=T$

a	b	$a \vee b$	$a \wedge b$
T	T	T	T
T	F	T	F
F	T	T	F
F	F	F	F

$a \wedge b = \neg(a \wedge b)$

$a=T, b=T$ False

a	b	$a \wedge b$	$\neg(a \wedge b)$
T	T	T	F
T	F	F	T
F	T	F	T
F	F	F	T