Resolution Refutation in Proposition Logic

Resolution Refutation in PL

- *Resolution refutation: A*nother simple method to prove a formula by contradiction.
- Here negation of goal is added to given set of clauses.
  - If there is a refutation in new set using resolution principle then goal is proved
- During resolution we need to identify two clauses,
  - one with positive atom (P) and other with negative atom (~ P) for the application of resolution rule.
- Resolution is based on modus ponen inference rule.

Disjunctive & Conjunctive Normal Forms

- *Disjunctive Normal Form* (DNF): A formula in the form $(L_{11} \wedge \ldots\ldots \wedge L_{1n}) \vee \ldots\ldots \vee (L_{m1} \wedge \ldots\ldots \wedge L_{mk})$, where all L_{ij} are literals.
  - Disjunctive Normal Form is disjunction of conjunctions.
- *Conjunctive Normal Form* (CNF): A formula in the form $(L_{11} \vee \ldots\ldots \vee L_{1n}) \wedge \ldots\ldots \wedge (L_{p1} \vee \ldots\ldots \vee L_{pm})$, where all L_{ij} are literals.
  - CNF is conjunction of disjunctions or
  - CNF is conjunction of clauses
- *Clause:* It is a formula of the form $(L_1 \vee \ldots \vee L_m)$, where each L_k is a positive or negative atom.

Conversion of a Formula to its CNF

- Each PL formula can be converted into its equivalent CNF.
- Use following equivalence laws:
  - $P \rightarrow Q \cong \sim P \lor Q$
  - $P \leftrightarrow Q \cong (P \rightarrow Q) \land (Q \rightarrow P)$
  - Double Negation
    - $\sim\sim P \cong P$
  - (De Morgan's law)
    - $\sim(P \land Q) \cong \sim P \lor \sim Q$
    - $\sim(P \lor Q) \cong \sim P \land \sim Q$
  - (Distributive law)
    - $P \lor (Q \land R) \cong (P \lor Q) \land (P \lor R)$

Resolvent of Clauses

- If two clauses C_1 and C_2 contain a complementary pair of literals {L, ~L},
  - then these clauses may be resolved together by deleting L from C_1 and ~ L from C_2 and constructing a new clause by the disjunction of the remaining literals in C_1 and C_2.
- The new clause thus generated is called **resolvent** of C_1 and C_2 .
  - Here C1 and C2 are called parents of resolved clause.
- Inverted binary tree is generated with the last node (root) of the binary tree to be a resolvent.
  - This is also called resolution tree.

Example

- Find resolvent of the following clauses:
  - C_1 = P V Q V R; C_2 = ~ Q V W; C_3 = P V ~ W
- Inverted Resolution Tree

P V Q V R ~ Q V W

{Q, ~ Q}

P V R V W P V ~ W

{W, ~ W}

P V R

- Resolvent(C1,C2, C3) = P V R

Logical Consequence

- **Theorem1**: If C is a resolvent of two clauses C_1 and C_2 , then C is a *logical consequence* of $\{C_1 , C_2\}$.
  - A deduction of an empty clause (or resolvent as contradiction) from a set S of clauses is called a *resolution refutation* of S.
- **Theorem2:** Let S be a set of clauses. A clause C is a *logical consequence* of S iff the set $S' = S \cup \{\sim C\}$ is *unsatisfiable*.
  - In other words, C is a logical consequence of a given set S iff an empty clause is deduced from the set S'.

Example

- Show that C V D is a logical consequence of
  - S ={AVB, ~ AVD, C V~ B} using resolution refutation principle.
- First we will add negation of logical consequence
  - i.e., ~ (C V D) ≅ ~C Λ ~D to the set S.
  - Get S' = {A V B, ~ A V D, C V~ B, ~C, ~D}.
- Now show that S' is unsatisfiable by deriving contradiction using resolution principle.