Lecture 8, 9

Propositional Logic:
Conjunctive Normal Form & Disjunctive Normal Form

Instructor: Yu Zhen Xie
Conjunctive Normal Form (CNF)

- Resolution works best when the formula is of the special form: it is an $\land$ of $\lor$s of (possibly negated, $\neg$) variables (called literals).

- This form is called a Conjunctive Normal Form, or CNF. 
  - $(y \lor \neg z) \land (\neg y) \land (y \lor z)$ is a CNF
  - $(x \lor \neg y \lor z)$ is a CNF. So is $(x \land \neg y \land z)$.
  - $(x \lor \neg y \land z)$ is not a CNF

- An AND ($\land$) of CNF formulas is a CNF formula.
  - So if all premises are CNF and the negation of the conclusion is a CNF, then AND of premises AND NOT conclusion is a CNF.
CNF

• To convert a formula into a CNF.
  – Open up the implications to get ORs.
  – Get rid of double negations.
  – Convert $F \lor (G \land H)$ to $(F \lor G) \land (F \lor H)$ //distributivity

• Example: $A \rightarrow (B \land C)$
  $\equiv \neg A \lor (B \land C)$
  $\equiv (\neg A \lor B) \land (\neg A \lor C)$

• In general, CNF can become quite big, especially when have $\leftrightarrow$. There are tricks to avoid that ...
What is the relation between propositional logic and logic circuits?

– View a formula as computing a function (called a Boolean function),
  • inputs are values of variables,
  • output is either true (1) or false (0).

– For example, $\text{Majority}(x, y, z) = true$ when at least two out of $x, y, z$ are true, and false otherwise.

– Such a function is fully described by a truth table of its formula (or its circuit: circuits have truth tables too).
Boolean functions and circuits

• What is the relation between propositional logic and logic circuits?
  – So both formulas and circuits “compute” Boolean functions – that is, truth tables.
  – In a circuit, can “reuse” a piece in several places, so a circuit can be smaller than a formula.
    • Still, most circuits are big!
  – Majority($x, y, z$) is $(x \land y) \lor (x \land z) \lor (y \land z)$
CNF and DNF

• Every truth table (Boolean function) can be written as either a conjunctive normal form (CNF) or disjunctive normal form (DNF)

• **CNF** is an $\land$ of $\lor$s, where $\lor$ is over variables or their negations (literals); an $\lor$ of literals is also called a **clause**.

• **DNF** is an $\lor$ of $\land$s; an $\land$ of literals is called a **term**.
Notations

• \( \neg p, x, s \) are examples of literals
• Neither \( \neg \neg p \) nor \( (x \lor y) \) is a literal
• \( (x \lor \neg y \lor z), (\neg p) \) are clause
• \( (x \land \neg y \land z), (\neg p) \) are terms
• \( (x \lor \neg z \lor y) \land (\neg x \lor \neg y) \land (\neg y) \) is a CNF
• \( (x \land z) \lor (\neg y \land z \land x) \lor (\neg x \land z) \) is a DNF
• \( (x \land \neg (y \lor z) \lor u) \) is neither a CNF nor DNF, but is equivalent to DNF \( ((x \land \neg y \land \neg z) \lor u) \)
Facts about CNF and DNF

• Any propositional formula is tautologically equivalent to some formula in disjunctive normal form.

• Any propositional formula is tautologically equivalent to some formula in conjunctive normal form.
Why CNF and DNF?

• Convenient normal forms
• **Resolution** works best for formulas in CNF
• Useful for constructing formulas given a **truth table**
  
  – **DNF**: take a disjunction (that is, $\lor$) of all **satisfying** truth assignments
  
  – **CNF**: take a conjunction ($\land$) of **negations** of **falsifying** truth assignments
From truth table to DNF and CNF

• A **minterm** is a conjunction of literals in which each variable is represented exactly once
  – If a Boolean function (truth table) has the variables $(p, q, r)$ then $p \land \neg q \land r$ is a minterm but $p \land \neg q$ is not.

• Each minterm is true for exactly one assignment.
  – $p \land \neg q \land r$ is true if $p$ is true (1), $q$ is false (0) and $r$ is true (1).
  – Any deviation from this assignment would make this particular minterm false.

• A disjunction of minterms is true only if at least one of its constituents minterms is true.
From truth table to DNF

• If a function, e.g. $F$, is given by a truth table, we know exactly for which assignments it is true.

• Consequently, we can select the minterms that make the function true and form the disjunction of these minterms.

• $F$ is true for three assignments:
  o $p, q, r$ are all true, $(p \land q \land r)$
  o $p, \neg q, r$ are all true, $(p \land \neg q \land r)$
  o $\neg p, \neg q, r$ are all true, $(\neg p \land \neg q \land r)$

• DNF of $F$: $(p \land q \land r) \lor (p \land \neg q \land r) \lor (\neg p \land \neg q \land r)$
From truth table to CNF

- **Complementation** can be used to obtain *conjunctive normal forms* from truth tables.
- If $A$ is a formula containing only the connectives $\neg$, $\lor$ and $\land$, then its complement is formed by
  - replacing all $\lor$ by $\land$
  - replacing all $\land$ by $\lor$
  - replacing all atoms by their complements.
    - The complement of $q$ is $\neg q$
    - The complement of $\neg q$ is $q$

- **Example:** Find the complement of the formula
  - $(p \land q) \lor \neg r$
    - $(\neg p \lor \neg q) \land r$
From truth table to CNF

- **Solution:** \( \neg G \) is **true** for the following assignments.
  
  \[
  \begin{align*}
  p = 1; & \quad q = 0; \quad r = 1 \\
  p = 1; & \quad q = 0; \quad r = 0 \\
  p = 0; & \quad q = 0; \quad r = 1 
  \end{align*}
  \]

- The **DNF** of \( \neg G \) is therefore:
  \[
  (p \land \neg q \land r) \lor (p \land \neg q \land \neg r) \lor (\neg p \land \neg q \land r)
  \]

- The formula has the **complement**:
  \[
  (\neg p \lor q \lor \neg r) \land (\neg p \lor q \lor r) \land (p \lor q \lor \neg r)
  \]

- It is the desired **CNF** of \( G \)
Canonical CNF and DNF

• So for every formula, there is a unique canonical CNF (and a truth table, and a Boolean function).

• And for every possible truth table (a Boolean function), there is a formula (the canonical CNF).

• Recall, to make a canonical DNF from a truth table:
  – Take all satisfying assignments.
  – Write each as an AND of literals, as before.
  – Then take an OR of these ANDs.
Complete set of connectives

• CNFs only have $\neg, V, \land$, yet any formula can be converted into a CNF
  – Any truth table can be coded as a CNF

• Call a set of connectives which can be used to express any formula a **complete set** of connectives.

  – In fact, $\neg, V$ is already complete. So is $\neg, \land$.

    • By DeMorgan, $(A \lor B) \equiv \neg(\neg A \land \neg B)$ No need for $V$!

  – But $\land, V$ is not: cannot do $\neg$ with just $\land, V$.

    • Because when both inputs have the same value, both $\land, V$ leave them unchanged.
Complete set of connectives

• How many connectives is enough?
  – Just one: **NAND** (NotAND), also called the Sheffer stroke, written as $|$ 

    $\neg A \equiv A \mid A$

    $A \lor B \equiv \neg (\neg A \land \neg B)$
    $\equiv (\neg A \mid \neg B)$
    $\equiv (A \mid A) \mid (B \mid B)$

  – In practice, most often stick to $\land, \lor, \neg$
Puzzle

• Susan is 28 years old, single, outspoken, and very bright. She majored in philosophy. As a student she was deeply concerned with issues of discrimination and social justice and also participated in anti-nuke demonstrations.

Please rank the following possibilities by how likely they are. List them from least likely to most likely. Susan is:

1. a kindergarden teacher
2. works in a bookstore and takes yoga classes
3. an active feminist
4. a psychiatric social worker
5. a member of an outdoors club
6. a bank teller
7. an insurance salesperson
8. a bank teller and an active feminist