CISC 204 Class 11

Conjunctive Normal Form, or CNF

Text Correspondence: pp. 57–65

Main Concepts:

- *Literal expression: atom or negation of an atom*
- *Disjunctive caluse: disjunction of literals*
- *CNF: conjunction of disjunctive clauses*

We will now define CNF formally.

Definition: Literal Expression

We say that L is a *literal expression*, or simply a *literal*, if and only if it is an atomic proposition or the negation of an atomic proposition.

For example, p and $\neg q$ are literals. The formula $\neg\neg p$ is not a literal because it is neither an atom or the negation of an atom. It is, however, semantically equivalent to p so if we see double negation in a formula we can safely substitute for it.

Symbolically, we can write the definition of a literal using a kind of assignment statement that is common in symbolic processing:

$$L \quad ::= \quad p \mid \neg p$$

The next term we need is a way of describing exactly the kind of disjuncts that we will allow.

Definition: Disjunctive clause

We say that D is a *disjunctive clause* if it is a literal or if it is the disjunction of a literal and a disjunctive clause.

Symbolically, we can write the definition of a disjunctive clause as

$$D \quad ::= \quad L \mid L \vee D$$

Examples of disjuncts include:

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

Counterexamples of disjuncts include:

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

We can now specify the conjunctive normal form in terms of literals and disjunctive clauses.

<u>Definition:</u> Conjunctive normal form (CNF)

We say that C is in *conjunctive normal form* (CNF) if it is a conjunction of disjunctive clauses.

Symbolically, we can write the definition of CNF as

$$C \quad ::= \quad D \;\mid\; (D) \wedge (C)$$

Examples of formulas in conjunctive normal form:

(i) $(\neg\, q \vee p \vee r) \wedge (\neg\, r \vee p) \wedge q$

(ii) $(p \vee r) \wedge (\neg\, p \vee r) \wedge (p \vee \neg\, r)$

Counterexample formula, not in CNF:

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

Example *(ii)* has 3 disjuncts so its form is $D_1 \wedge D_2 \wedge D_3$. It evaluates to true only if there is a valuation, or model, such that all D_i are *simultaneously* true.

Recall that ϕ is satisfiable if and only if $\neg\,\phi$ is not valid. We can use this property to extract the CNF for any formula ϕ from its truth table. A simple algorithm is:

- For each row in the truth table where ϕ is **F**, take the disjunction of all of the negations of the atoms
- Form a conjunction of all of these disjunctive formulas

For example, consider the truth table

p	q	$(p \vee \neg p) \to q$
T	**T**	**T**
T	**F**	**F**
F	**T**	**T**
F	**F**	**F**

From this table, CNF for $(p \vee \neg\, p) \to q$ is

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

Is this the only CNF, or the minimal CNF? By inspection of the truth table, we can see that a simpler way of writing the implication is just the formula q. We can also use the rule $\vee$e to syntactically deduce q.

The CNF derived from the truth table is not guaranteed to be minimal, but it is always a correct way of writing the original formula.

We now have a way of finding the CNF for any given formula:

- Expand the literals and the formula into a truth table, then
- Write the **F** rows of the truth table as a disjunction of conjunctions, then
- Apply De Morgan's law to find a conjunction of disjunctions

Being computer scientists, we might wonder if there is a process for directly converting any given formula into CNF. The answer is that yes there is a process, and it is quite involved (taking many pages of description in the textbook).

To transform any given formula into CNF, it is useful to apply some simplifications and expansion. Because semantic equivalence corresponds to syntactic derivation, we can safely replace some complicated expressions with expressions that are easier to work with. Important equivalences that make this process easier include:

$$\begin{aligned}
\neg\neg\phi &\equiv \phi \\
(\phi \to \psi) &\equiv (\neg\,\phi \vee \psi) \\
\neg(\phi \wedge \psi) &\equiv (\neg\phi \vee \neg\psi) \\
\neg(\phi \vee \psi) &\equiv (\neg\phi \wedge \neg\psi) \\
((\phi \wedge \psi) \vee \chi)) &\equiv ((\phi \vee \chi) \wedge (\psi \vee \chi)) \\
((\phi \vee \psi) \wedge \chi)) &\equiv ((\phi \wedge \chi) \vee (\psi \wedge \chi))
\end{aligned}$$

How to translate a formula into CNF:

1. Remove all implications by replacing sub-formulas of the form $\phi \to \psi$ into the form $\neg\phi \vee \psi$.
2. Cancel out double negations
3. Distribute all negations to atomic formula and cancel double negations
4. Distribute all disjunctions of conjunctions to conjunction of disjunctions

CNF can be used to check validity with a simple algorithm. Each disjunctive clause has literals l_1, l_2, $\cdots$. The clause is valid if and only if it includes a disjunction $(\phi \vee \neg\phi)$. A brief way of writing this is:

A clause is valid if and only if there exist literals l_i and l_j such that $l_i \equiv \neg l_j$

We can use this property to conclude:

A CNF formula is valid if and only if every disjunctive clause is valid

Self-Study: Put the following formulas into CNF, both by using truth tables and by using the above algorithm for translating formulas. Using only the CNF of the formulas, determine whether the original formulas are valid. Sample CNF versions are on the next page of these notes.

Formula:

$p \to (q \to p)$

$p \to (q \wedge p)$

$(p \vee \neg r) \to (q \to r)$

$(\neg p \vee q) \to (p \to q)$

Self-Study: Which of these CNF formulas are valid?

CNF

$$\neg p \vee q \vee p$$

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

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

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