EECS 1028 M: Discrete Mathematics for Engineers

Suprakash Datta
Office: LAS 3043

Course page: `http://www.eecs.yorku.ca/course/1028`
Also on Moodle

Using the laws - 2

Q: Simplify $(p \to q) \to \neg q$.

- We need to use analytic means simplify:

$$
\begin{aligned}
(p \to q) \to \neg q &\equiv \neg(p \to q) \vee \neg q \text{ equivalent form of } \to \\
&\equiv \neg(\neg p \vee q) \vee \neg q \text{ equivalent form of } \to \\
&\equiv (p \wedge \neg q) \vee \neg q \text{ De Morgan's Law} \\
&\equiv \neg q \text{ Absorption}
\end{aligned}
$$

- Check using truth tables

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

Inference in Propositional Logic

Section 1.6 pages 71-75

- Recall: the reason for studying logic was to formalize derivations and proofs.
- How can we infer facts using logic?
- Simple inference rule (Modus Ponens) : From (a) $p \rightarrow q$ and (b) p is TRUE, we can infer that q is TRUE.
- Many other rules, see page 72.
  - Understanding the rules is crucial, memorizing is not.
  - You should be able to see that the rules make sense and correspond to our intuition about formal reasoning.

Inference Rules

TABLE 1 **Rules of Inference.**

Rule of Inference	*Tautology*	*Name*
p $p \to q$ $\therefore q$	$(p \wedge (p \to q)) \to q$	Modus ponens
$\neg q$ $p \to q$ $\therefore \neg p$	$(\neg q \wedge (p \to q)) \to \neg p$	Modus tollens
$p \to q$ $q \to r$ $\therefore p \to r$	$((p \to q) \wedge (q \to r)) \to (p \to r)$	Hypothetical syllogism
$p \vee q$ $\neg p$ $\therefore q$	$((p \vee q) \wedge \neg p) \to q$	Disjunctive syllogism
p $\therefore p \vee q$	$p \to (p \vee q)$	Addition
$p \wedge q$ $\therefore p$	$(p \wedge q) \to p$	Simplification
p q $\therefore p \wedge q$	$((p) \wedge (q)) \to (p \wedge q)$	Conjunction
$p \vee q$ $\neg p \vee r$ $\therefore q \vee r$	$((p \vee q) \wedge (\neg p \vee r)) \to (q \vee r)$	Resolution

Modus Ponens

- Example:
  (a) if these lecture slides are online then you can print them out
  (b) these lecture slides are online
  Can you print out the slides?

- $((p \to q) \wedge p) \to q$ is a TAUTOLOGY.

- From $p \to q$, $q \to r$ and p is TRUE, we can infer that r is TRUE.

Other Inference Rules

- Modus Tollens and Disjunctive Syllogism can be seen as alternative forms of Modus Ponens
- Hypothetical syllogism is like the chain rule of implications
- Other rules like "From p is true we can infer $p \vee q$" are very intuitive
- Resolution: From
  (a) $p \vee q$ and (b) $\neg p \vee r$, we can infer : $q \vee r$

  Exercise: check that $((p \vee q) \wedge (\neg p \vee r)) \rightarrow (q \vee r)$ is a TAUTOLOGY.

  $\boxed{\text{Very useful in computer generated proofs}}$.

Correctness of Inference, Proofs

- A Propositional Logic statement is correctly inferred iff it is made using one of the rules of inference listed before
- A proof is correct iff it is a sequence of statements that are either
  - axioms,
  - statements inferred earlier, or
  - statements inferred using one of the rules of inference listed before, and

  the last conclusion is the assertion that needed to be proved.

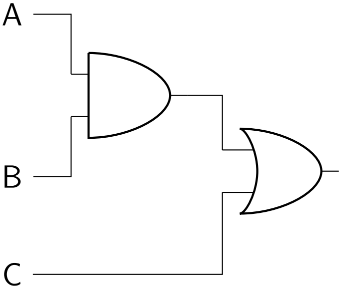
Terminology: An inferred fact is called a *proposition, lemma* or *theorem* depending on its importance; a special case of a proved statement is sometimes stated as a *corollary*.

Practice on Propositional Logic Inference

- Q3c, pg 78
  If it is rainy, then the pool will be closed. It is rainy. Therefore, the pool will be closed.
- Q3e, pg 78
  If I go swimming, then I will stay in the sun too long. If I stay in the sun too long, then I will sunburn. Therefore, if I go swimming, then I will sunburm.
- Q9c, pg 78.
  "I am either clever or lucky", ""I am not lucky", "If I am lucky, then I will win the lottery".

Implementing Propositional Logic Statements in Hardware

Typically assume AND, OR, NOT "logic gates", sometimes NAND, NOR, XOR. E.g.,

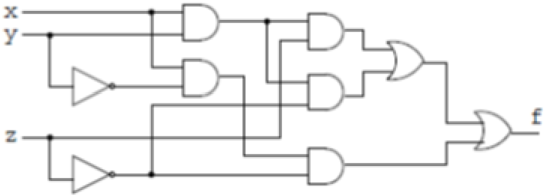


$\boxed{(A \wedge B) \vee C}$

- Often OR and AND are written as $+, \cdot$ respectively

More on Boolean Circuits

- Evaluating Boolean Circuits: Propagate values sequentially by levels, from input to output
- Making a circuit from a propositional logic expression: Disjunctive normal form - ORs of ANDs, e.g., $f(x, y, z) = xyz + xy\overline{z} + x\overline{yz}$



- The function $f(x, y, z)$ simplifies to $x(y + \overline{z})$.

Boolean Circuits from Truth Tables

- Truth Table to DNF:

x	y	z	$f(x, y, z)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

- Disjunction of three terms, one for each 1 entry in the last column
- For each term, put each variable, negated iff it is zero in that row
- So first 1 corresponds to $x\overline{y}\overline{z}$
- So, $f(x, y, z) = xyz + xy\overline{z} + x\overline{y}\overline{z}$

- DNF to circuit: Same as the previous slide

$\boxed{\text{(How) can this circuit be minimized?}}$ More advanced courses

Limitations of Propositional Logic

- What can we NOT express using predicates?
  E.g., : How do you make a statement about all even integers, like "for all integers x, if $x > 2$ then $x^2 > 4$"?

- A more general language: Predicate logic (Sec 1.4)