Logic and Computation – CS 2800
Fall 2023

Propositional logic continued

Stavros Tripakis

Northeastern University
Khoury College of
Computer Sciences

Outline

- Truth tables
- Satisfiability and validity
- Stronger, weaker, and equivalent formulas

Homework: build the truth table of this Boolean expression

$$\neg p \lor q \equiv p \Rightarrow q$$

Truth tables of Boolean formulas

- For every Boolean formula (= Boolean expression) we can construct its truth table
  - E.g., for the formula $\neg p \vee q \equiv p \Rightarrow q$ we get the truth table:

p	q	¬p	¬p ∨ q	p ⇒ q	¬p ∨ q ≡ p ⇒ q
T	T	F	T	T	T
T	F	F	F	F	T
F	T	T	T	T	T
F	F	T	T	T	T

Assignment: assigns truth values to the propositional variables of the formula

Subformula: a subexpression (subtree)

Size of truth tables

- How many rows does a truth table have?
  - Answer: 2^n, where n is the number of inputs, i.e., the number of variables
- How many columns does a truth table have?
  - Answer: one column for every subformula
- If n is the number of inputs, how many different last columns can I have? (i.e., how many different n-ary Boolean functions can I have?)
  - Answer: 2^{2^n} (make sure you understand why!)

Syntax vs semantics

- Infinitely many Boolean expressions!
- Finitely many Boolean functions!
- Finitely many truth tables!

- Many Boolean expressions are **equivalent**: they represent the same Boolean function, i.e., they have the same last column in their truth table.
  - E.g., p is equivalent to $p \vee p$, and also to $p \wedge p$, and also to $p \vee p \vee p$, etc.

Satisfiability and validity

Satisfiability and validity

- A Boolean formula (=Boolean expression) is called:
  - **Satisfiable**: when it is **sometimes true**
  - **Unsatisfiable**: when it is **never true**
  - **Valid** or a **tautology**: when it is **always true**
  - **Falsifiable**: when it is **sometimes false**

- “Sometimes”, “always”, “never” refer to the truth table of the formula:
  - Sometimes: exists assignment to make the formula true
  - Never: no assignment makes the formula true
  - Always: all assignments make the formula true

Satisfiability and validity

- Every Boolean formula satisfies exactly two of the previous characterizations
- In particular, a Boolean formula is exactly one of the following:
  - Satisfiable and valid: always true
  - Satisfiable but not valid = satisfiable and falsifiable: sometimes true and sometimes false
  - Unsatisfiable and falsifiable: always false
- Think of it as a *lattice*:

$$\begin{array}{c} \text{valid} \\ | \\ \text{neither} \\ | \\ \text{unsatisfiable} \end{array}$$

Examples – homework

- Think of examples of the three categories

- Valid:

- Satisfiable and falsifiable:

- Unsatisfiable:

Stronger, weaker, and equivalent formulas

- We say that formula P is **stronger** than formula Q if the formula $P \rightarrow Q$ is valid (i.e., a tautology). This is the same as saying that whenever P is true, Q is also true.
- We say that formula P is **weaker** than formula Q if the formula $Q \rightarrow P$ is valid (i.e., a tautology). This is the same as saying that whenever Q is true, P is also true. This is also the same as saying that Q is stronger than P.
- We say that formula P is **equivalent** to formula Q if both formulas $P \rightarrow Q$ and $Q \rightarrow P$ are valid. This the same as saying that P is both stronger and weaker than Q.
- The same terminology applies to non-Boolean formulas, e.g., we can say that $x > 5$ is stronger than $x > 0$.

- Mini (ungraded) quizzes:
  - What is the strongest possible formula?
  - What is the weakest possible formula?

Necessary and sufficient conditions

- In other settings you will hear the terms **necessary** and **sufficient**, usually followed by conditions

- P is a **necessary** condition for Q if Q is stronger than P
- P is a **sufficient** condition for Q if P is stronger than Q

- If P is both necessary and sufficient for Q, then the two are equivalent

- E.g., $x > 0$ is necessary for $x > 1$, and sufficient for $x \geq 0$