CS1382 Discrete Computational Structures

Lecture 08: The Foundations: Logic and Proofs

Spring 2019

Richard Matovu

TEXAS TECH UNIVERSITY

References

- The materials of this presentation is mostly from the following:
    - Discrete Mathematics and Its Applications (Text book and Slides)

      By Kenneth Rosen, 7th edition

The Foundations: Logic and Proofs

- Logic defines the ground rules for establishing truths.
    - Basis for mathematical reasoning and automated reasoning
- Mathematical logic spells out these rules in complete detail, defining what constitutes a formal proof.
- Learning mathematical logic is a good way to learn logic because it puts you on a firm foundation.
- Writing formal proofs in mathematical logic is a lot like computer programming. The rules of the game are clearly defined.

- Uses of Proofs in Computer Science
    - Verify that computer programs produce the correct output for all possible input values
    - Algorithms always produce the correct result
    - Establish the security of a system
    - Create Artificial Intelligence

CS1382 Discrete Computational Structures

Propositional Logic

Spring 2019

Richard Matovu

TEXAS TECH UNIVERSITY

Propositional Logic

- Propositional logic is a language that **abstracts away from content** and focuses on the **logical connectives**.
- **Proposition** is a declarative sentence
  - Either True or False, but not both.
- Examples of propositions
  - Java is case-sensitive (True)
  - It is rainy today (False)
  - 0 + 1 = 1 (True)
  - 0 + 0 = 3 (False)
- Not propositions
  - Questions
  - Imperatives
- Examples that aren't propositions
  - Sit down!
  - Read this carefully
  - What time is it?
  - x + 1 = 2
  - x + y = z

Propositional Logic

- Constructing Propositions
  - Propositional variables: **p, q, r, s, …**
    - Variables are propositions
    - Model true or false
  - The proposition that is always true is denoted by T and the proposition that is always false is denoted by F.
  - Compound Propositions
    - Constructed from logical connectives and other propositions

- **Connectives**
  - Negation - $\neg$
  - Conjunction - $\land$
  - Disjunction - $\lor$
  - Implication - $\rightarrow$
  - Biconditional - $\leftrightarrow$

Compound Propositions: Negation

- Let p be a proposition.
  - The negation of p, denoted by ¬p is the statement:

    "It is not the case that p."
- The proposition ¬p is pronounced "not p."
- The negation of a proposition p is denoted by ¬p and has this truth table:

p	¬p
T	F
F	T

- Example:

  If p denotes
  - "The earth is round"
    - ¬p denotes "It is not the case that the earth is round" or more simply "The earth is not round."
  - "Michael's PC runs Linux"
    - ¬p denotes "It is not the case that Michael's PC runs Linux" or more simply "Michael's PC doesn't run Linux"

Compound Propositions: Conjunction

- Let p and q be propositions.
  - The **conjunction** of p and q, denoted **p ∧ q**, is true when p and q are both true and false otherwise.
- The conjunction of propositions p and q is denoted by p ∧ q and has this truth table:

p	q	p ∧ q
T	T	T
T	F	F
F	T	F
F	F	F

- Example: If
  - p denotes “I am at home”
  - q denotes “It is raining.”
  - p ∧ q denotes “I am at home and it is raining.”

Compound Propositions: Disjunction

- Let p and q be propositions.
  - The **disjunction** of p and q, denoted **p V q**, is false when p and q are false and true otherwise.
- The disjunction of propositions p and q is denoted by p V q and has this truth table:

p	q	p V q
T	T	T
T	F	T
F	T	T
F	F	F

- Example: If
  - p denotes “I am at home”
  - q denotes “It is raining.”
  - p Vq denotes “I am at home or it is raining.”

The Connective Or in English

- **“Inclusive Or”**
  - “Students who have taken CS202 or Math120 may take this class”
  - We assume that students need to have taken one of the prerequisites, but may have taken both.
  - This is the meaning of disjunction. For $p \vee q$ to be true, either one or both of p and q must be true.

- **“Exclusive Or”**
  - “2GB or 4GB of RAM comes with this smartphone”
  - We do not expect to be able to get both 2GB and 4GB.
  - This is the meaning of Exclusive Or (Xor).
  - In $p \oplus q$, one of p and q must be true, but not both.

Exclusive Or

- Let p and q be propositions. The **exclusive or** of p and q denoted **p ⊕ q**,
  - Proposition that is true exactly when one of p and q is true.
- The exclusive or has this truth table:

p	**q**	**p ⊕ q**
T	T	F
T	F	T
F	T	T
F	F	F

Conditional Statements / Implication

- Let p and q be propositions.
  - $p \rightarrow q$ is a conditional statement or implication which is read as " **if p, then q** " and has this truth table:

p	q	$p \rightarrow q$
T	T	T
T	F	F
F	T	T
F	F	T

- In $p \rightarrow q$, p is the hypothesis (antecedent or premise) and q is the conclusion (or consequence).

- Example: If
  - p denotes "I am at home"
  - q denotes "It is raining."
  - $p \rightarrow q$ denotes "If I am at home, then it is raining."

Understanding Implication

- In $p \rightarrow q$, there does not need to be any connection between the hypothesis and the consequence.

- The "meaning" of $p \rightarrow q$ depends only on the truth values of p and q.

- These implications are perfectly fine, but would not be used in ordinary English.
  - "If the moon is made of green cheese, then I'm on welfare."
  - "If 1 + 1 = 3, then your grandma wears combat boots."

- One way to view the logical conditional is to think of an obligation or contract.
  - "If I am elected, then I will lower taxes."
  - "If you get 100% on the final, then you will get an A."
  - *If the politician is elected and does not lower taxes, then the voters can say that he or she has broken the campaign pledge. Something similar holds for the professor. This corresponds to the case where p is true and q is false.*

Different ways of expressing $p \rightarrow q$

- if p, then q
- If p, q
- q unless ¬p
- q if p
- p is sufficient for q
- q is necessary for p
- a sufficient condition for q is p
- p implies q
- p only if q
- q when p
- q whenever p
- q follows from p
- a necessary condition for p is q

Converse, Contrapositive, and Inverse

- From $p \to q$ we can form new conditional statements .
  - $q \to p$ is the converse of $p \to q$
  - $\neg q \to \neg p$ is the contrapositive of $p \to q$
  - $\neg p \to \neg q$ is the inverse of $p \to q$
- Example:

  Find the converse, inverse, and contrapositive of

  "It raining is a sufficient condition for my not going to town."

- Solution
  - Converse:
    If I do not go to town, then it is raining.
  - Inverse:
    If it is not raining, then I will go to town.
  - Contrapositive:
    If I go to town, then it is not raining

Converse, Contrapositive, and Inverse - Exercise

- The home team wins whenever it is raining
  - ***q whenever p***
  - **Condition:** If it is raining, then the home team wins
  - **Contrapositive:** If the home team does not win, then it is not raining
  - **Converse:** If the home team wins, then it is raining.
  - **Inverse:** If it is not raining, then the home team does not win

Compound Propositions: Biconditional

- Let p and q be propositions.
  - The **biconditional** proposition **p ↔ q**, read as " **p if and only if q** "
  - The biconditional p ↔ q denotes the proposition with this truth table:

p	q	p ↔ q
T	T	T
T	F	F
F	T	F
F	F	T

- Example: If
  - p denotes "I am at home"
  - q denotes "It is raining."
  - p ↔ q denotes "I am at home if and only if it is raining."
- Some alternative ways "p if and only if q" is expressed in English:
  - p is necessary and sufficient for q
  - if p then q , and conversely
  - p iff q

Truth Tables For Compound Propositions

Construction of a truth table:

Rows

- Need a row for every possible combination of values for the atomic propositions.

Columns

- Need a column for the compound proposition (usually at far right)
- Need a column for the truth value of each expression that occurs in the compound proposition as it is built up.
  - This includes the atomic propositions

Example Truth Table

Construct a truth table for $p \lor q \rightarrow \neg r$

p	q	r	¬ r	p ∨ q	p ∨ q → ¬ r
T	T	T	F	T	F
T	T	F	T	T	T
T	F	T	F	T	F
T	F	F	T	T	T
F	T	T	F	T	F
F	T	F	T	T	T
F	F	T	F	F	T
F	F	F	T	F	T

Equivalent Propositions

Two propositions are ***equivalent*** if they always have the same truth value.

Example:

Show using a truth table that the conditional is equivalent to the contrapositive.

Solution:

p	q	¬ p	¬ q	p → q	¬ q → ¬ p
T	T	F	F	T	T
T	F	F	T	F	F
F	T	T	F	T	T
F	F	T	T	T	T

Using a Truth Table to Show Non-Equivalence

Show using truth tables that neither the converse nor inverse of an implication are not equivalent to the implication.

Solution:

p	q	$\neg p$	$\neg q$	$p \rightarrow q$	$\neg p \rightarrow \neg q$	$q \rightarrow p$
T	T	F	F	T	T	T
T	F	F	T	F	T	T
F	T	T	F	T	F	F
F	F	T	T	T	T	T

Problem

- How many rows are there in a truth table with *n* propositional variables?

- **Solution**:

  2^n *(We will see how to do this in Chapter 6)*

- Note that this means that with n propositional variables, we can construct 2^n distinct

  (i.e., not equivalent) propositions.

Precedence of Logical Operators

Operator	Precedence
$\neg$	1
$\wedge$	2
$\vee$	3
$\rightarrow$	4
$\leftrightarrow$	5

$p \vee q \rightarrow \neg r$ is equivalent to $(p \vee q) \rightarrow \neg r$

If the intended meaning is $p \vee (q \rightarrow \neg r)$,

then parentheses must be used.

CS1382 Discrete Computational Structures

Applications of Propositional Logic

Spring 2019

Richard Matovu

TEXAS TECH UNIVERSITY

Translating English Sentences

- Steps to convert an English sentence to a statement in propositional logic
  - Identify atomic propositions and represent using propositional variables.
  - Determine appropriate logical connectives
- Example
  - "If I go to Harry's or Mary's apartment, I will not go shopping."

- Solution
  - p: I go to Harry's apartment
  - q: I go to Mary's apartment
  - r: I will go shopping
  - If p or q, then not r

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

Exercise

- Translate the following sentence into propositional logic:
  - "If you can access the Internet from campus, then you are a computer science major or you are not a freshman."

- Solution
  - a: "You can access the internet from campus,"
  - c: "You are a computer science major,"
  - f: "You are a freshman."

$$\mathbf{a \rightarrow (c \lor \neg f)}$$

System Specifications

- System and Software engineers take requirements in English and express them in a precise specification language based on logic.
- Example:

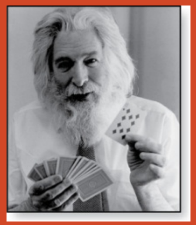
  Express in propositional logic:

  "The automated reply cannot be sent when the file system is full"

- One possible solution
  - p: "The automated reply can be sent"
  - q: "The file system is full"

$$\mathbf{q \rightarrow \neg p}$$

Logic Puzzles



Raymond Smullyan (Born 1919)

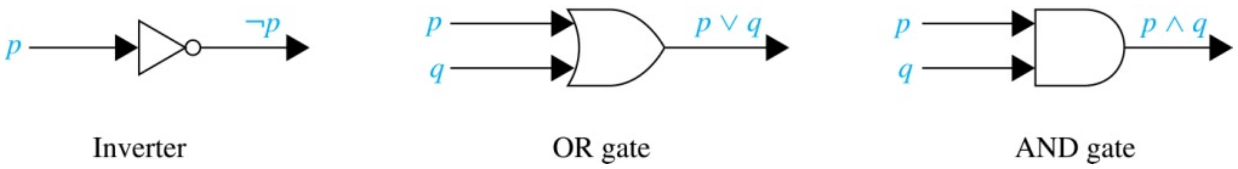
- An island has two kinds of inhabitants, ***knights***, who always tell the truth, and ***knaves***, who always lie.
- You go to the island and meet A and B.
  - A says “B is a knight.”
  - B says “The two of us are of opposite types.”
- Example: What are the types of A and B?

Solution

- p: A is a knight, ¬ p: A is a knave
- q: B is a knight, ¬ q: B is a knave
- If A is a knight, then p is true.
  Since knights tell the truth, q must also be true. Then $(p \wedge \neg q) \vee (\neg p \wedge q)$ would have to be true, but it is not. So, A is not a knight and therefore ¬ p must be true.
- If A is a knave, then B must not be a knight since knaves always lie. So, then both ¬ p and ¬ q hold since both are knaves.

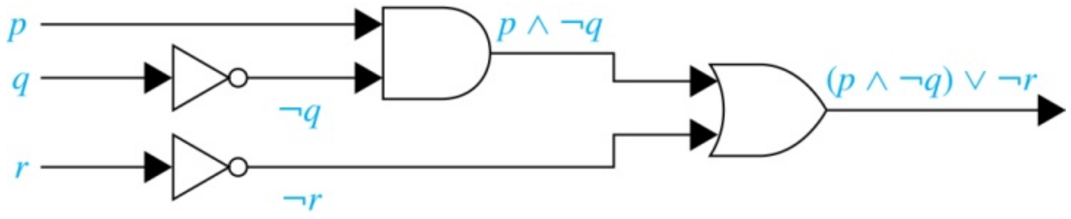
Logic Circuits

- Electronic circuits; each input/output signal can be viewed as a 0 or 1.
  - 0 represents False
  - 1 represents True
- Complicated circuits are constructed from three basic circuits called **gates**.
  - The inverter (**NOT gate**) takes an input bit and produces the negation of that bit.
  - The **OR gate** takes two input bits and produces the value equivalent to the disjunction of the two bits.



- The **AND gate** takes two input bits and produces the value equivalent to the conjunction of the two bits.

- More complicated digital circuits can be constructed by combining these basic circuits to produce the desired output given the input signals by building a circuit for each piece of the output expression and then combining them. For example:



CS1382 Discrete Computational Structures

Lecture 09: Propositional Equivalences

Spring 2019

Richard Matovu

TEXAS TECH UNIVERSITY

Tautologies, Contradictions, and Contingencies

- A ***tautology*** is a proposition which is always true.
  - Example: $p \vee \neg p$
- A ***contradiction*** is a proposition which is always false.
  - Example: $p \wedge \neg p$
- A ***contingency*** is a proposition which is neither a tautology nor a contradiction, such as p

p	$\neg p$	$p \vee \neg p$	$p \wedge \neg p$
T	F	T	F
F	T	T	F

Logically Equivalent

- Two compound propositions p and q are logically equivalent if $p \leftrightarrow q$ is a tautology.
- We write this as $p \Leftrightarrow q$ or as $p \equiv q$ where p and q are compound propositions.
- Two compound propositions p and q are equivalent if and only if the columns in a truth table giving their truth values agree.

- This truth table shows that ¬p ∨ q is equivalent to p → q.

p	q	$\neg p$	$\neg p \vee q$	$p \rightarrow q$
T	T	F	T	T
T	F	F	F	F
F	T	T	T	T
F	F	T	T	T

De Morgan's Laws

Augustus De Morgan

1806-1871

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

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

This truth table shows that De Morgan's Second Law holds.

p	q	¬ p	¬ q	(p ∨ q)	¬ (p ∨ q)	¬ p ∧ ¬ q
T	T	F	F	T	F	F
T	F	F	T	T	F	F
F	T	T	F	T	F	F
F	F	T	T	F	T	T

Key Logical Equivalences

Identity Laws: $p \wedge T \equiv p$ $\quad$ $p \vee F \equiv p$

Domination Laws: $p \vee T \equiv T$ $\quad$ $p \wedge F \equiv F$

Idempotent laws: $p \vee p \equiv p$ $\quad$ $p \wedge p \equiv p$

Double Negation Law: $\neg(\neg p) \equiv p$

Negation Laws: $p \vee \neg p \equiv T$ $\quad$ $p \wedge \neg p \equiv F$

Key Logical Equivalences (*cont*)

Commutative Laws: $p \vee q \equiv q \vee p \quad p \wedge q \equiv q \wedge p$

Associative Laws:
$$(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$$
$$(p \vee q) \vee r \equiv p \vee (q \vee r)$$

Distributive Laws:
$$(p \vee (q \wedge r)) \equiv (p \vee q) \wedge (p \vee r)$$
$$(p \wedge (q \vee r)) \equiv (p \wedge q) \vee (p \wedge r)$$

Absorption Laws: $p \vee (p \wedge q) \equiv p \quad p \wedge (p \vee q) \equiv p$

More Logical Equivalences

TABLE 7 Logical Equivalences Involving Conditional Statements.

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

TABLE 8 Logical Equivalences Involving Biconditional Statements.

$$p \leftrightarrow q \equiv (p \to q) \wedge (q \to p)$$
$$p \leftrightarrow q \equiv \neg p \leftrightarrow \neg q$$
$$p \leftrightarrow q \equiv (p \wedge q) \vee (\neg p \wedge \neg q)$$
$$\neg(p \leftrightarrow q) \equiv p \leftrightarrow \neg q$$

Equivalence Proofs - Example

Show that $\neg(p \vee (\neg p \wedge q))$ is logically equivalent to $\neg p \wedge \neg q$

Solution:

$$
\begin{aligned}
\neg(p \vee (\neg p \wedge q)) &\equiv \neg p \wedge \neg(\neg p \wedge q) && \text{by the second De Morgan law} \\
&\equiv \neg p \wedge [\neg(\neg p) \vee \neg q] && \text{by the first De Morgan law} \\
&\equiv \neg p \wedge (p \vee \neg q) && \text{by the double negation law} \\
&\equiv (\neg p \wedge p) \vee (\neg p \wedge \neg q) && \text{by the second distributive law} \\
&\equiv F \vee (\neg p \wedge \neg q) && \text{because } \neg p \wedge p \equiv F \\
&\equiv (\neg p \wedge \neg q) \vee F && \text{by the commutative law} \\
& && \text{for disjunction} \\
&\equiv (\neg p \wedge \neg q) && \text{by the identity law for } \mathbf{F}
\end{aligned}
$$

Equivalence Proofs

Show that $(p \wedge q) \to (p \vee q)$ is a tautology.

Solution:

$$
\begin{aligned}
(p \wedge q) \to (p \vee q) &\equiv \neg(p \wedge q) \vee (p \vee q) && \text{by truth table for } \to \\
&\equiv (\neg p \vee \neg q) \vee (p \vee q) && \text{by the first De Morgan law} \\
&\equiv (\neg p \vee p) \vee (\neg q \vee q) && \text{by associative and} \\
& && \text{commutative laws} \\
& && \text{laws for disjunction} \\
&\equiv T \vee T && \text{by truth tables} \\
&\equiv T && \text{by the domination law}
\end{aligned}
$$

Propositional Satisfiability

- A compound proposition is ***satisfiable*** if there is an assignment of truth values to its variables that make it true.
- When no such assignments exist, the compound proposition is ***unsatisfiable***.

Example

- Determine the satisfiability of the following compound propositions:

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

- **Solution**:

Satisfiable. Assign **T** to *p, q,* and *r*

Exercise

- Determine the satisfiability of the following compound propositions:

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

- **Solution**:

  Satisfiable. Assign **T** to p and F to q

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

- **Solution:**

  Not satisfiable. Check each possible assignment of truth values to the propositional variables and none will make the proposition true.

Questions?

Thank You!