Discrete Mathematics

Chapter 1, Part I: Propositional Logic

Reference book: Discrete Mathematics and Its Applications 8th ed. (Rosen, 2019)

Propositional Logic

Section 1.1

Propositions

A *proposition* is a declarative sentence that is either true or false.

Examples of propositions:

a) The Moon is made of green cheese.
b) Trenton is the capital of New Jersey.
c) Toronto is the capital of Canada.
d) $1 + 0 = 1$
e) $0 + 0 = 2$

Examples that are not propositions.

a) Sit down!
b) What time is it?
c) $x + 1 = 2$
d) $x + y = z$

Propositional Logic

Constructing Propositions

- Propositional Variables: *p*, *q*, *r*, *s*, …
- 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
  - Negation $\neg$
  - Conjunction $\wedge$
  - Disjunction $\vee$
  - Implication $\rightarrow$
  - Biconditional $\leftrightarrow$

Compound Propositions: Negation

The *negation* of a proposition p is denoted by $\neg p$ and has this truth table:

p	$\neg p$
T	F
F	T

Example: If p denotes "The earth is round.", then $\neg p$ denotes "It is not the case that the earth is round," or more simply "The earth is not round."

Conjunction

The *conjunction* of propositions *p* and *q* is denoted by $p \land q$ and has this truth table:

p	*q*	$P \land q$
T	T	T
T	F	F
F	T	F
F	F	F

Example: If *p* denotes "I am at home." and *q* denotes "It is raining." then $p \land q$ denotes "I am at home and it is raining."

Disjunction

The *disjunction* of propositions *p* and *q* is denoted by $p \vee q$ and has this truth table:

p	q	$P \vee q$
T	T	T
T	F	T
F	T	T
F	F	F

Example: If *p* denotes "I am at home." and *q* denotes "It is raining." then $p \vee q$ denotes "I am at home or it is raining."

The Connective Or in English

In English “or” has two distinct meanings.

- “Inclusive Or” - In the sentence “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” - When reading the sentence “Soup or salad comes with this entrée,” we do not expect to be able to get both soup and salad. This is the meaning of Exclusive Or (Xor). In $p \oplus q$, one of p and q must be true, but not both. The truth table for $\oplus$ is:

p	q	$P \oplus q$
T	T	F
T	F	T
F	T	T
F	F	F

Implication

If p and q are propositions, then $p \to q$ is a *conditional statement* or *implication* which is read as "if p, then q" and has this truth table:

p	q	$P \to q$
T	T	T
T	F	F
F	T	T
F	F	T

Example: If p denotes "I am at home." and q denotes "It is raining." then $p \to q$ denotes "If I am at home then it is raining."

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

Different Ways of Expressing $p \to q$

if p**, then** q

if p, q

q **unless** $\neg p$

q **if** p

q **whenever** p

q **follows from** p

p **implies** q

p **only if** q

q **when** p

p **is sufficient for** q

q **is necessary for** p

a necessary condition for p **is** q

a sufficient condition for q **is** p

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.

Biconditional

If p and q are propositions, then we can form the *biconditional* proposition $p \leftrightarrow q$, read as "p if and only if q." The biconditional $p \leftrightarrow q$ denotes the proposition with this truth table:

p	q	$P \leftrightarrow q$
T	T	T
T	F	F
F	T	F
F	F	T

If p denotes "I am at home." and q denotes "It is raining." then $p \leftrightarrow q$ denotes "I am at home if and only if it is raining."

Expressing the Biconditional

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* **if** *q*

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	$\neg p$	$\neg q$	$p \rightarrow q$	$\neg q \rightarrow \neg p$
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

Example: 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

Precedence of Logical Operators

Operator	Precedence
$\neg$	1
$\land$	2
$\lor$	3
$\rightarrow$	4
$\leftrightarrow$	5

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

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

then parentheses must be used.

Applications of Propositional Logic

Section 1.2

Applications

- Imprecise or ambiguous statements in mathematics, sciences, natural language.
- Translation into the language of logic to make such statements precise.
- Thus, logic and its rules can be applied.

Applications of Propositional Logic: Summary

Translating English to Propositional Logic

System Specifications

Boolean Searching

Logic Puzzles

Logic Circuits

AI Diagnosis Method (Optional)

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

"If I go to Harry's or to the country, I will not go shopping."

- *p*: I go to Harry's
- *q*: I go to the country.
- *r*: I will go shopping.

If *p* or *q* then not *r*.

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

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"

Solution: One possible solution: Let *p* denote "The automated reply can be sent" and *q* denote "The file system is full."

$$q \to \neg p$$

Consistent System Specifications

Definition: A list of propositions is *consistent* if it is possible to assign truth values to the proposition variables so that each proposition is true.

Exercise: Are these specifications consistent?

- "The diagnostic message is stored in the buffer or it is retransmitted."
- "The diagnostic message is not stored in the buffer."
- "If the diagnostic message is stored in the buffer, then it is retransmitted."

Solution: Let p denote "The diagnostic message is stored in the buffer." Let q denote "The diagnostic message is retransmitted" The specification can be written as: $p \vee q$, $\neg p$, $p \rightarrow q$. When p is false and q is true all three statements are true. So the specification is consistent.

- What if "The diagnostic message is not retransmitted" is added.

  Solution: Now we are adding $\neg q$ and there is no satisfying assignment. So the specification is not consistent.

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: Let p and q be the statements that A is a knight and B is a knight, respectively. So, then $\neg p$ represents the proposition that A is a knave and $\neg q$ that B is a knave.

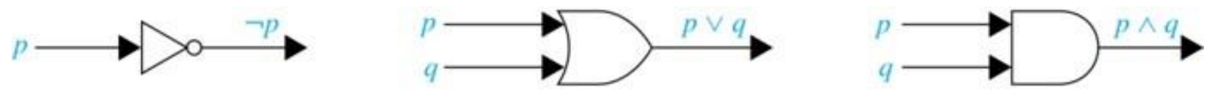
- If A is a knight, then p is true. Since knights tell the truth, q must also be true. Then $(p \wedge \neg q) \vee (\neg p \wedge q)$ would have to be true, but it is not. So, A is not a knight and therefore $\neg p$ must be true.
- If A is a knave, then B must not be a knight since knaves always lie.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

Logic Circuits
(Studied in depth in Chapter 12)

Electronic circuits; each input/output signal can be viewed as a 0 or 1.

- 0 represents **False**
- 1 represents **True**

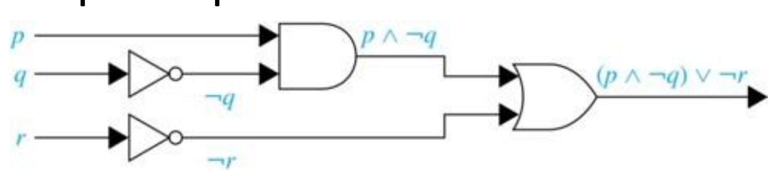
Complicated circuits are constructed from three basic circuits called gates.



Inverter OR gate AND gate

- 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:



Propositional Equivalences

Section 1.3

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 $\neg p \vee q$ is equivalent to $p \rightarrow q$.

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

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

Augustus De Morgan
1806-1871

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

p	q	$\neg p$	$\neg q$	$(p \vee q)$	$\neg(p \vee q)$	$\neg p \wedge \neg q$
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 [1]

Identity Laws: $p \wedge T \equiv p, \qquad p \vee F \equiv p$

Domination Laws: $p \vee T \equiv T, \qquad p \wedge F \equiv F$

Idempotent laws: $p \vee p \equiv p, \qquad p \wedge p \equiv p$

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

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

Key Logical Equivalences 2

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

Constructing New Logical Equivalences

We can show that two expressions are logically equivalent by developing a series of logically equivalent statements.

To prove that $A \equiv B$ we produce a series of equivalences beginning with A and ending with B.

$$\begin{gathered} A \equiv A_1 \\ \vdots \\ A_n \equiv B \end{gathered}$$

Equivalence Proofs [1]

Example: Show that $\neg\left(p \vee (\neg p \wedge q)\right)$

is logically equivalent to $\neg p \wedge \neg q$

Solution:

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

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*.

A compound proposition is unsatisfiable if and only if its negation is a tautology.

Questions on Propositional Satisfiability

Example: Determine the satisfiability of the following compound propositions:

Exercise 1

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

Exercise 2

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

Exercise 3

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

Notation

$$\bigvee_{j=1}^{n} p_j \text{ is used for } p_1 \vee p_2 \vee \ldots \vee p_n$$

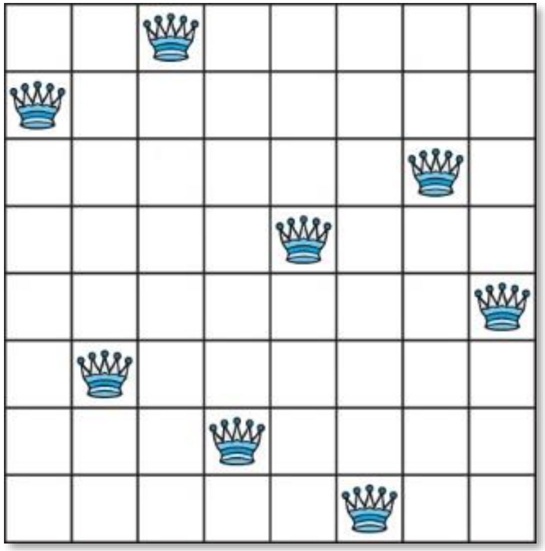
$$\bigwedge_{j=1}^{n} p_j \text{ is used for } p_1 \wedge p_2 \wedge \ldots \wedge p_n$$

Needed for the next example.

N-Queen Problem

Problem

- Place N Queens on a NxN grid, while not placing two Queens on the same vertical, horizontal or diagonal line



Modeling

- Proposition $p_{i,j}$ indicates whether a Queen is placed at the i-row and at the j-th column

N-Queen Problem

Every row contains at least one queen

$$Q_1 = \bigwedge_{i=1..n} \bigvee_{j=1..n} p_{i,j}$$

At most one queen in each row

$$Q_2 = \bigwedge_{i=1..n} \bigwedge_{j=1..n-1} \bigwedge_{k=j+1..n} \neg(p_{i,j} \wedge p_{i,k})$$

At most one queen in each column

$$Q_3 = \bigwedge_{i=1..n} \bigwedge_{j=1..n-1} \bigwedge_{k=j+1..n} \neg(p_{j,i} \wedge p_{k,i})$$

No diagonal contains two queens
(toward the top-right direction)
i-1: limit toward top, n-j: limit toward right

$$Q_4 = \bigwedge_{i=2..n} \bigwedge_{j=1..n-1} \bigwedge_{k=1..\min(i-1,n-j)} \neg(p_{i,j} \wedge p_{i-k,j+k})$$

No diagonal contains two queens
(toward the bottom-right direction)
n-1: limit toward bottom, n-j: limit toward right

$$Q_5 = \bigwedge_{i=1..n-1} \bigwedge_{j=1..n-1} \bigwedge_{k=1..\min(n-i,n-j)} \neg(p_{i,j} \wedge p_{i+k,j+k})$$

$$Q = Q_1 \wedge Q_2 \wedge Q_3 \wedge Q_4 \wedge Q_5$$

Solution is the truth values to $p(i,j)$ that satisfies Q.

Sudoku

A **Sudoku puzzle** is represented by a 9×9 grid made up of nine 3×3 subgrids, known as **blocks**. Some of the 81 cells of the puzzle are assigned one of the numbers 1,2, …, 9.

The puzzle is solved by assigning numbers to each blank cell so that every row, column and block contains each of the nine possible numbers.

Example

	2	9				4		
			5			1		
	4							
				4	2			
6							7	
5								
7			3					5
	1			9				
							6	

Encoding as a Satisfiability Problem [1]

Let $p(i,j,n)$ denote the proposition that is true when the number n is in the cell in the ith row and the jth column.

There are $9 \times 9 \times 9 = 729$ such propositions.

In the sample puzzle $p(5,1,6)$ is true, but $p(5,j,6)$ is false for $j = 2,3,\ldots 9$

Encoding as a Satisfiability Problem 2

For each cell with a given value, assert *p*(*i*,*j*,*n*), when the cell in row *i* and column *j* has the given value.

Assert that every row contains every number.

$$\bigwedge_{i=1}^{9}\bigwedge_{n=1}^{9}\bigvee_{j=1}^{9} p(i,j,n)$$

Assert that every column contains every number.

$$\bigwedge_{j=1}^{9}\bigwedge_{n=1}^{9}\bigvee_{i=1}^{9} p(i,j,n)$$

Encoding as a Satisfiability Problem 3

Assert that each of the 3 × 3 blocks contain every number.

$$\bigwedge_{r=0}^{2}\bigwedge_{s=0}^{2}\bigwedge_{n=1}^{9}\bigwedge_{i=1}^{3}\bigvee_{j=1}^{3} p\left(3r+i,3s+j,n\right)$$

(this is tricky - ideas from chapter 4 help)

Assert that no cell contains more than one number. Take the conjunction over all values of *n*, *n′*, *i*, and j, where each variable ranges from 1 to 9 and n ≠ *n′* , of

$$p\left(i,j,n\right)\rightarrow\neg p\left(i,j,n'\right)$$

Solving Satisfiability Problems

To solve a Sudoku puzzle, we need to find an assignment of truth values to the 729 variables of the form $p(i,j,n)$ that makes the conjunction of the assertions true. Those variables that are assigned T yield a solution to the puzzle.

A truth table can always be used to determine the satisfiability of a compound proposition. But this is too complex even for modern computers for large problems.

There has been much work on developing efficient methods for solving satisfiability problems as many practical problems can be translated into satisfiability problems.