w05-Lec

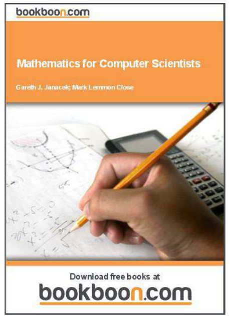
Mathematics and Computer Science: Boolean Algebra

Assembled for 204111

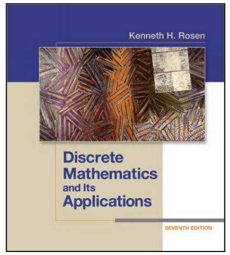
by Areerat Trongratsameethong

Boolean Algebra

- **Mathematics for Computer Scientists – Janacek and Close**
  - **Introduction to Logical Value**
  - **Logical Operators**



- **Discrete Mathematics and Its Applications – K.H. Rosen**
  - **Introduction to Boolean Algebra**
  - **Rules of Precedence for Boolean Operators**
  - **Boolean Properties**
  - **Boolean Expression Simplification**



Introduction to Logical Value

- **A statement has its logical value, either true (T) or false (F)**
  - **"it is raining": may be true or false**
  - **"it is sunny": may be true or false**
- **Many statements can be combined with *and* and *or* to be a compound statement.**
  - **"it is raining and it is sunny"**
  - **"it is raining or it is sunny"**

The logical values of the compound statements depend on the logical value of each combined statement and what connective ("and" or "or") is used.

Logical Operators

- **Three logical operators are used in Boolean Algebra**
  - **Negation (not)**
  - **Conjunction (and)**
  - **Disjunction (or)**

Logical Operators (2)

- **Symbolic Notation: is used to make things shorter**
  - **Negation denoted by ¬**
  - **"and" denoted by ∧**
  - **"or" denoted by ∨**
- **A symbolic can be also used for a statement**
  - ***p* can be used for "it is raining"**

Logical Operators (3)

- **Negation ¬**
  - **The negation of a statement is false when the statement is true.**
  - **The negation of a statement is true when the statement is false.**
- **Example**
  - **Let *p* = "It is raining",**

    then ¬*p* is "it is not raining"
  - **If the logical value of *p* is F**

    then logical value of ¬*p* is T

Logical Operators (4)

- **The logical values can be represented in form of truth table as an example below.**

Example

Let *p* = "All computer scientists are men"

Two possible logical values of *p* are T and F

p	¬*p*
T	F
F	T

Table 2.1: Truth table for negation (¬)

Logical Operators (5)

- **Conjunction ∧**
- **If *p* and *q* are statements, then**

  ***p* ∧ *q* is read as "*p* and *q*".**
  - **Let**
    - ***p* = "It is green",**
    - ***q* = "It is an apple" then**

    ***p* ∧ *q* = "It is green and It is an apple"**
  - **The logical value of *p* ∧ *q* depends on each logical value of *p* and *q* as shown in Table 2.2.**

p	*q*	*p* ∧ *q*
T	T	T
T	F	F
F	T	F
F	F	F

Table 2.2: Truth table for ∧

Logical Operators (3)

- **Disjunction ∨**
- **If *p* and *q* are statements, then**

  ***p* ∨ *q* is read as "*p* or *q*".**
  - **Let**
    - ***p* = "It is green",**
    - ***q* = "It is an apple" then**

    ***p* ∨ *q* = "It is green or It is an apple"**
  - **The logical value of *p* ∨ *q* depends on each logical value of *p* and *q* as shown in Table 2.3.**

p	*q*	*p* ∨ *q*
T	T	T
T	F	T
F	T	T
F	F	F

Table 2.3: Truth table for ∨

Introduction to Boolean Algebra

- **A Symbolic form of Aristotle's system of logic sought by George Boole (1815-1864) - The English mathematician**
- **Mathematical language dealing with the questions of logic**
- **An Investigation of the Laws of Thought (Boole 1854),**
  - **Theories of Logic and Probabilities**
  - **Mathematical Relationship Quantities Rule**
    - **true or false**
    - **1 or 0**

Introduction to Boolean Algebra (2)

- **The mathematical system written by Boole became known as Boolean algebra.**
- **All Boolean quantities have two possible outcomes: 1 or 0.**
- **There is no such thing as "2" or "-1" or "1/2" in the Boolean world.**

Introduction to Boolean Algebra (3)

- **Boolean algebra as on-and-off circuits Control**
- **All signals are characterized as either "high" (1) or "low" (0).**
- **A Symbolic Analysis of Relay and Switching Circuits –MIT Thesis (Shannon 1938)**
  - **Mathematical tool for designing and analyzing digital circuits.**
  - **Defined the circuits in all electronic devices as 1 or 0 referring 'on' or in 'off' position.**

Introduction to Boolean Algebra (4)

- **Boolean algebra provides the operations and the rules for working with the set {0,1}.**
- **Operation for a circuit is called Boolean Function.**
- **Boolean Function produce output for each set of inputs.**
- **This function is built using Boolean expressions and operations.**

Rules of Precedence for Boolean Operators

- **Order of Boolean Operators**
  1. **Complement (or Negation ¬) denoted by ~**
  2. **Boolean Product denoted by ∧**
  3. **Boolean Sum denoted by ∨**

~~**Note: Unless parentheses () are used, operations in the parentheses are done first.**~~

Rules of Precedence for Boolean Operators (2)

Example: Find the value of 1 ∧ 0 ∨ ~ (0 ∨ 1)

$1 \wedge 0 \vee \sim (\underline{0 \vee 1})$

$1 \wedge 0 \vee \underline{\sim 1}$

$\underline{1 \wedge 0} \vee 0$

$0 \vee 0$

0

Boolean Properties

- **The simpler that we can make a Boolean function, the smaller the circuit that will result.**
- **Simpler Circuits**
  - **Cheaper to build**
  - **Consume less power**
  - **Run faster than the complex circuits**
- **With this in mind, we always want to reduce our Boolean functions to their simplest form.**
- **There are a number of Boolean identities that help us to do this.**

Rules of Boolean Algebra

- **Most Boolean identities have an AND (product) form as well as an OR (sum) form.**
- **We give our identities using both forms.**

Identity Name	AND	OR
Identity Law	$1 \land x = x$	$0 \lor x = x$
Null Law	$0 \land x = 0$	$1 \lor x = 1$
Idempotent Law	$x \land x = x$	$x \lor x = x$
Inverse Law	$x \land \sim x = 0$	$x \lor \sim x = 1$

Note: These laws can be proved by truth table.

x	1	$1 \land x$
0	1	0
1	1	1

Truth table of $1 \land x$

How about the others?

Rules of Boolean Algebra(2)

Identity Name	AND
Commutative Law	$x \land y = y \land x$
Associative Law	$(x \land y) \land z = x \land (y \land z)$
Distributive Law	$x \lor (y \land z) = (x \lor y) \land (x \lor z)$

Identity Name	OR
Commutative Law	$x \lor y = y \lor x$
Associative Law	$(x \lor y) \lor z = x \lor (y \lor z)$
Distributive Law	$x \land (y \lor z) = (x \land y) \lor (x \land z)$

Remark: The above identities can be translated to logical equivalences about propositions and to identities about sets.

Boolean Properties (3)

Identity Name	AND	OR
Absorption Law	$x \land (x \lor y) = x$	$x \lor (x \land y) = x$
Demorgan's Law	$\sim(x \land y) = \sim x \lor \sim y$	$\sim(x \lor y) = \sim x \land \sim y$
Double Complement Law	$\sim(\sim x) = x$	

Proof – Absorption Law:

$x \land (x \lor y) = x$

Rewrite: $\land \rightarrow \cdot$, $\lor \rightarrow +$

$x \cdot (x + y) = x \cdot x + (x \cdot y)$

$= x + (x \cdot y)$

$= (x \cdot 1) + (x \cdot y)$

$= x \cdot (1 + y)$

$= x \cdot 1$

$= x$

Proof – Absorption Law:

$x \lor (x \land y) = x$

Rewrite: $\land \rightarrow \cdot$, $\lor \rightarrow +$

$x + (x \cdot y) = (x \cdot 1) + (x \cdot y)$

$= x \cdot (1 + y)$

$= x \cdot 1$

$= x$

Boolean Expression Simplification

Example 1: $(x \lor y) \land (x \lor \sim y) \land \sim(x \land \sim z)$

$(x \lor y) \land (x \lor \sim y) \land \sim(x \land \sim z)$ — DeMorgan's Law

$(x \lor y) \land (x \lor \sim y) \land (\sim x \lor z)$ — Distributive Law

$(x \land x) \lor (x \land \sim y) \lor (y \land x) \lor (y \land \sim y) \land (\sim x \lor z)$

Idempotent and Inverse Laws

$x \lor (x \land \sim y) \lor (y \land x) \lor 0 \land (\sim x \lor z)$

Absorption and Identity Laws

$x \lor (y \land x) \land (\sim x \lor z)$ — Absorption Law

$x \land (\sim x \lor z)$ — Distributive Law

$(x \land \sim x) \lor (x \land z)$ — Inverse Law

$0 \lor (x \land z)$ — Identity Law

$x \land z$

Boolean Expression Simplification (2)

- **Example 2: Find the complement of the Boolean expression below. (Demorgan's Law)**

  (x∧y) ∨ (~x∧z) ∨ (y ∧ ~z)

  ~((x∧y) ∨ (~x∧z) ∨ (y ∧ ~z))

  ~(x∧y) ∧ ~ (~x∧z) ∧ ~ (y ∧ ~z)

  (~x ∨ ~y) ∧ (x ∨ ~z) ∧ (~y ∨ z)

Summary

- **Introduction to Logical Value**
- **Logical Operators**
- **Introduction to Boolean Algebra**
- **Rules of Precedence for Boolean Operators**
- **Boolean Properties**
- **Boolean Expression Simplification**

Reference

- **Mathematics for Computer Scientists, Janacek and Close**
- **Discrete Mathematics and Its Applications, K.H. Rosen**