Switching Algebra and Its Applications

Zvi Kohavi and Niraj K. Jha

Switching Algebra

Basic postulate: existence of two-valued switching variable that takes two distinct values 0 and 1

Switching algebra: algebraic system of set {0,1}, binary operations OR and AND, and unary operation NOT

OR operation	AND operation
0 + 0 = 0	0 . 0 = 0
0 + 1 = 1	0 . 1 = 0
1 + 0 = 1	1 . 0 = 0
1 + 1 = 1	1 . 1 = 1

NOT operation (complementation): 0' = 1 and 1' = 0

OR: also called logical sum

AND: also called logical product

Basic Properties

Idempotency: $x + x = x$

$x \cdot x = x$

Perfect induction: proving a theorem by verifying every combination of values that the variables may assume

Proof of $x + x = x$: $1 + 1 = 1$ and $0 + 0 = 0$

If x is a switching variable, then: $x + 1 = 1$

$x \cdot 0 = 0$

$x + 0 = x$

$x \cdot 1 = x$

Commutativity: $x + y = y + x$

$x \cdot y = y \cdot x$

3

Basic Properties (Contd.)

Associativity: $(x + y) + z = x + (y + z)$

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

Complementation: $x + x' = 1$

$x \cdot x' = 0$

Distributivity: $x \cdot (y + z) = x \cdot y + x \cdot z$

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

Proof by perfect induction using a truth table:

x	y	z	xy	xz	$y+z$	$x(y+z)$	$xy+xz$
0	0	0	0	0	0	0	0
0	0	1	0	0	1	0	0
0	1	0	0	0	1	0	0
0	1	1	0	0	1	0	0
1	0	0	0	0	0	0	0
1	0	1	0	1	1	1	1
1	1	0	1	0	1	1	1
1	1	1	1	1	1	1	1

4

Basic Properties (Contd.)

Principle of Duality:

- Preceding properties grouped in pairs
- One statement can be obtained from the other by interchanging operations OR and AND and constants 0 and 1
- The two statements are said to be dual of each other
- This principle stems from the symmetry of the postulates and definitions of switching algebra w.r.t. the two operations and constants
- Implication: necessary to prove only one of each pair of statements

5

Switching Expressions and Their Manipulation

Switching expression: combination of finite number of switching variables and constants via switching operations (AND, OR, NOT)

- Any constant or switching variable is a switching expression
- If T_1 and T_2 are switching expressions, so are T_1', T_2', T_1+T_2 and T_1T_2
- No other combination of constants and variables is a switching expression

Absorption law: $x + xy = x$

$x(x + y) = x$

Proof: $x + xy = x1 + xy$ [basic property]

$= x(1 + y)$ [distributivity]

$= x1$ [commutativity and basic property]

$= x$ [basic property]

6

Laws of Switching Algebra

Another important law: $x + x'y = x + y$

$x(x' + y) = xy$

Proof: $x + x'y = (x + x')(x + y)$ [distributivity]

$= 1(x + y)$ [complementation]

$= x + y$ [commutativity and basic property]

Consensus theorem: $xy + x'z + yz = xy + x'z$

$(x + y)(x' + z)(y + z) = (x + y)(x' + z)$

Proof: $xy + x'z + yz = xy + x'z + yz1$

$= xy + x'z + yz(x+x')$

$= xy(1 + z) + x'z(1 + y)$

$= xy + x'z$

7

Switching Expression Simplification

Literal: variable or its complement

Redundant literal: if value of switching expression is independent of literal x_i then x_i is said to be redundant

Example: Simplify $T(x,y,z) = x'y'z + yz + xz$

$x'y'z + yz + xz = z(x'y' + y + x)$

$= z(x' + y + x)$

$= z(y + 1)$

$= z1$

$= z$

Thus, literals x and y are redundant in $T(x,y,z)$

Important note: Since no inverse operations are defined in Switching Algebra, cancellations are not allowed

- $A + B = A + C$ does not imply $B = C$
- Counterexample: $A = B = 1$ and $C = 0$
- Similarly, $AB = AC$ does not imply $B = C$

8

De Morgan's Theorems

Involution: $(x')' = x$

De Morgan's theorem for two variables:

$(x + y)' = x' . y'$

$(x . y)' = x' + y'$

Proof by perfect induction:

x	y	x'	y'	$x+y$	$(x+y)'$	$x'y'$
0	0	1	1	0	1	1
0	1	1	0	1	0	0
1	0	0	1	1	0	0
1	1	0	0	1	0	0

De Morgan's theorems for *n* variables:

$[f(x_1, x_2, \ldots, x_n, 0, 1, +, .)]' = f(x_1', x_2', \ldots, x_n', 1, 0, ., +)$

9

Simplification Examples

Example: Simplify $T(x,y,z) = (x + y)[x'(y' + z')]' + x'y' + x'z'$

$$\begin{aligned}(x + y)[x'(y' + z')]' + x'y' + x'z' &= (x + y)(x + yz) + x'y' + x'z' \\ &= (x + xyz + yx + yz) + x'y' + x'z' \\ &= x + yz + x'y' + x'z' \\ &= x + yz + y' + z' \\ &= x + z + y' + z' \\ &= x + y' + 1 \\ &= 1\end{aligned}$$

Thus, $T(x,y,z) = 1$, independently of the values of the variables

Example: Prove $xy + x'y' + yz = xy + x'y' + x'z$

- From consensus theorem, $x'z$ can be added to LHS
- Consensus theorem can be applied again to first, third and fourth terms in $xy + x'y' + yz + x'z$ to eliminate yz and reduce it to RHS

10

Switching Functions

Let $T(x_1, x_2, \ldots, x_n)$ be a switching expression:

- Since each variable can assume 0 or 1, 2^n combinations are possible

Determining the value of an expression for an input combination:

Example: $T(x,y,z) = x'z + xz' + x'y'$

$T(0,0,1) = 0'1 + 01' + 0'0' = 1$

Truth table for T

x	y	z	T
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

Switching Function (Contd.)

Switching function $f(x_1, x_2, \ldots, x_n)$: values assumed by an expression for all combinations of variables $x_1, x_2, \ldots, x_n$

Complement function: $f'(x_1, x_2, \ldots, x_n)$ assumes value 0 (1) whenever $f(x_1, x_2, \ldots, x_n)$ assumes value 1 (0)

Logical sum of two functions: $f(x_1, x_2, \ldots, x_n) + g(x_1, x_2, \ldots, x_n) = 1$ for every combination in which either f or g or both equal 1

Logical product of two functions: $f(x_1, x_2, \ldots, x_n) . g(x_1, x_2, \ldots, x_n) = 1$ for every combination for which both f and g equal 1

Switching Function (Contd.)

Illustrating sum, product and complementation of functions:

x	y	z	f	g	f'	$f+g$	fg
0	0	0	1	0	0	1	0
0	0	1	0	1	1	1	0
0	1	0	1	0	0	1	0
0	1	1	1	1	0	1	1
1	0	0	0	1	1	1	0
1	0	1	0	0	1	0	0
1	1	0	1	1	0	1	1
1	1	1	1	0	0	1	0

Simplification of Expressions

Example: Simplify *T*(*x*,*y*,*z*) = *A'C'* + *ABD* + *BC'D* + *AB'D'* + *ABCD'*

- Apply consensus theorem to first three terms -> *BC'D* is redundant
- Apply distributive law to last two terms -> *AD'*(*B'* + *BC*) -> *AD'*(*B'* + *C*)
- Thus, *T* = *A'C'* + *A*[*BD* + *D'*(*B'* + *C*)]

Example: Simplify *T*(*A*,*B*,*C*,*D*) = *A'B* + *ABD* + *AB'CD'* + *BC*

- *A'B* + *ABD* = *B*(*A'* + *AD*) = *B*(*A'* + *D*)
- *AB'CD'* + *BC* = *C*(*B* + *AB'D'*) = *C*(*B* + *AD'*)
- Thus, *T* = *A'B* + *BD* + *ACD'* + *BC*
- Expand *BC* to (*A* + *A'*)*BC* to obtain *T* = *A'B* + *BD* + *ACD'* + *ABC* + *A'BC*
- From absorption law: *A'B* + *A'BC* = *A'B*
- From consensus theorem: *BD* + *ACD'* + *ABC* = *BD* + *ACD'*
- Thus, *T* = *A'B* + *BD* + *ACD'*

Canonical Forms

Deriving an expression from a truth table:

- Find the sum of all terms that correspond to combinations for which function is 1
- Each term is a product of the variables on which the function depends
- Variable x_i appears in uncomplemented (complemented) form in the product if has value 1 (0) in the combination

Truth table for $f = x'y'z' + x'yz' + x'yz + xyz' + xyz$

Decimal code	*x*	*y*	*z*	*f*
0	0	0	0	1
1	0	0	1	0
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	1
7	1	1	1	1

Canonical Sum-of-products

Minterm: a product term that contains each of the *n* variables as factors in either complemented or uncomplemented form

- It assumes value 1 for exactly one combination of variables

Canonical sum-of-products: sum of all minterms derived from combinations for which function is 1

- Also called disjunctive normal expression

Compact representation of switching functions: $\sum(0,2,3,6,7)$

Decimal code	*x*	*y*	*z*	*f*
0	0	0	0	1
1	0	0	1	0
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	1
7	1	1	1	1

Canonical Product-of-sums

Maxterm: a sum term that contains each of the *n* variables in either complemented or uncomplemented form

- It assumes value 0 for exactly one combination of variables
- Variable x_i appears in uncomplemented (complemented) form in the sum if it has value 0 (1) in the combination

Canonical product-of-sums: product of all maxterms derived from combinations for which function is 0

- Also called conjunctive normal expression

Compact representation of switching functions: $\prod(1,4,5)$

$$f = (x + y + z')(x' + y + z)(x' + y + z')$$

Decimal code	x	y	z	f
0	0	0	0	1
1	0	0	1	0
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	1
7	1	1	1	1

Shannon's Expansion to Obtain Canonical Forms

Shannon's expansion theorem:

$$f(x_1, x_2, \dots, x_n) = x_1 \,.\, f(1, x_2, \dots, x_n) + x_1' \,.\, f(0, x_2, \dots, x_n)$$

$$f(x_1, x_2, \dots, x_n) = [x_1 + f(0, x_2, \dots, x_n)] \,.\, [x_1' + f(1, x_2, \dots, x_n)]$$

Proof by perfect induction: Plug in $x_1 = 1$ and then $x_1 = 0$ to reduce RHS to LHS

Shannon's expansion around two variables:

$$f(x_1, x_2, \dots, x_n) = x_1x_2f(1, 1, x_3, \dots, x_n) + x_1x_2'f(1, 0, x_3, \dots, x_n) + x_1'x_2f(0, 1, x_3, \dots, x_n) + x_1'x_2'f(0, 0, x_3, \dots, x_n)$$

Similar Shannon's expansion around all *n* variables yields the canonical sum-of-products

Repeated expansion of the dual form yields the canonical product-of-sums

Simpler Procedure for Canonical Sum-of-products

1. Examine each term: if it is a minterm, retain it; continue to next term
2. In each product which is not a minterm: check the variables that do not occur; for each x_i that does not occur, multiply the product by $(x_i + x_i')$
3. Multiply out all products and eliminate redundant terms

Example: $T(x,y,z) = x'y + z' + xyz$

$$= x'y(z + z') + (x + x')(y + y')z' + xyz$$
$$= x'yz + x'yz' + xyz' + xy'z' + x'yz' + x'y'z' + xyz$$
$$= x'yz + x'yz' + xyz' + xy'z' + x'y'z' + xyz$$

Canonical product-of-sums obtained in a dual manner

Example:

$$T = x'(y' + z)$$
$$= (x' + yy' + zz')(y' + z + xx')$$
$$= (x' + y + z)(x' + y + z')(x' + y' + z)(x' + y' + z')(x + y' + z)(x' + y' + z)$$
$$= (x' + y + z)(x' + y + z')(x' + y' + z)(x' + y' + z')(x + y' + z)$$

Transforming One Form to Another

Example: Find the canonical product-of-sums for

$$T(x,y,z) = x'y'z' + x'y'z + x'yz + xyz + xy'z + xy'z'$$

$$T = (T')' = [(x'y'z' + x'y'z + x'yz + xyz + xy'z + xy'z')']'$$

Complement T' consists of minterms not contained in T. Thus,

$$T = [x'yz' + xyz']'$$
$$= (x + y' + z)(x' + y' + z)$$

Canonical forms are unique

Two switching functions are equivalent if and only if their corresponding canonical forms are identical

Functional Properties

Let binary constant a_i be the value of function $f(x_1, x_2, \ldots, x_n)$ for the combination of variables whose decimal code is *i*. Thus,

$$f(x_1, x_2, \ldots, x_n) = a_0 x_1' x_2' \ldots x_n' + a_1 x_1' x_2' \ldots x_n + \ldots + a_r x_1 x_2 \ldots x_n$$

Factor a_i is set to 1 (0) if the corresponding minterm is (is not) in the canonical form

Since there are 2^n coefficients, each of which can have two values, 0 and 1, there are $2^{\wedge}2^n$ possible switching functions of *n* variables

Example: Canonical sum-of-products form for two variables

$$f(x,y) = a_0 x'y' + a_1 x'y + a_2 xy' + a_3 xy$$

There are $2^{\wedge}2^2$ functions corresponding to the 16 possible assignments of 0's and 1's to a_0, a_1, a_2, and a_3

21

List of Functions of Two Variables

a_3	a_2	a_1	a_0	$f(x,y)$	*Name of function*	*Symbol*
0	0	0	0	0	Inconsistency	
0	0	0	1	$x'y'$	NOR(dagger)	$x \downarrow y$
0	0	1	0	$x'y$		
0	0	1	1	x'	NOT	x'
0	1	0	0	xy'		
0	1	0	1	y'		
0	1	1	0	$x'y + xy'$	EXCLUSIVE OR (modulo-2 addition)	$x \oplus y$
0	1	1	1	$x' + y'$	NAND(Sheffer stroke)	$x \mid y$
1	0	0	0	xy	AND	$x \cdot y$
1	0	0	1	$xy + x'y'$	Equivalence	$x \equiv y$
1	0	1	0	y		
1	0	1	1	$x' + y$	Implication	$x \rightarrow y$
1	1	0	0	x		
1	1	0	1	$x + y'$	Implication	$y \rightarrow x$
1	1	1	0	$x + y$	OR	$x + y$
1	1	1	1	1	Tautology	

22

The Exclusive-OR Operation

Exclusive-OR: modulo-2 addition, i.e., $A \oplus B = 1$ if either A or B is 1, but not both

Commutativity: $A \oplus B = B \oplus A$

Associativity: $(A \oplus B) \oplus C = A \oplus (B \oplus C) = A \oplus B \oplus C$

Distributivity: $(AB) \oplus (AC) = A(B \oplus C)$

If $A \oplus B = C$, then

$A \oplus C = B$

$B \oplus C = A$

$A \oplus B \oplus C = 0$

Exclusive-OR of an even (odd) number of elements, whose value is 1, is 0 (1)

23

Functionally Complete Operations

Every switching function can be expressed in canonical form consisting of a finite number of switching variables, constants and operations +, ., '

A set of operations is functionally complete (or universal) if and only if every switching function can be expressed by operations from this set

Example: Set {+, ., '}

Example: Set {+, '} since using De Morgan's theorem, $x \,.\, y = (x' + y')'$. Thus, + and ' can replace the . in any switching function

Example: Set {., '} for similar reasons

Example: NAND since NAND(x,x) = x' and NAND[NAND(x,y),NAND(x,y)] = xy

Example: NOR since NOR(x,x) = x' and NOR[NOR(x,y),NOR(x,y)] = $x + y$

24

Isomorphic Systems

Isomorphism: Two algebraic systems are isomorphic if

- For every operation in one system, there exists a corresponding operation in the second system
- To each element x_i in one system, there corresponds a unique element y_i in the other system, and vice versa
- If each operation and element in every postulate of one system is replaced by the corresponding operation and element in the other system, then the resulting postulate is valid in the second system

Thus, two algebraic systems are isomorphic if and only if they are identical except the labels and symbols used to represent the operations and elements

25

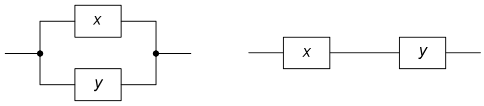
Series-parallel Switching Circuits

Gate: a two-state device capable of switching from one state, which permits flow of information, to another, which blocks it, and vice versa

Primed (unprimed) two-valued variable: denotes flowing (blocked) information

If a gate permits (blocks) the flow of information: literal associated with it takes value 1 (0)

Elementary series-parallel switching circuits



Parallel connection $x + y$ Series connection xy

Series-parallel circuits: any circuit constructed of either a series or parallel connection of two or more elementary series-parallel circuits

26

Transmission Function

Transmission function for a circuit: assumes value 1 (0) when there is (there is not) a path from one terminal of the circuit to the other through which information flows

Definition of transmission functions

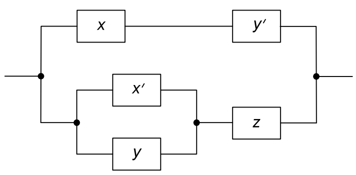
x	y	$x + y$	xy
0	0	0	0
0	1	1	0
1	0	1	0
1	1	1	1

Analogy: OR <-> parallel; AND <-> series

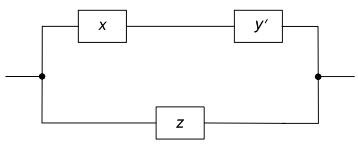
Complement of a given circuit: one that blocks all paths of information flow whenever the given circuit permits any

Thus, algebraic system for switching circuits isomorphic to switching algebra

Switching Circuit Simplification



Circuit realizing $T = xy' + (x'+ y)z$



Simplified circuit realizing

$$\begin{aligned} T &= xy' + x'z + yz \\ &= xy' + x'z + y'z + yz \\ &= xy' + x'z + z \\ &= xy' + z \end{aligned}$$

Important application of theory of switching circuits: CMOS

Propositional Calculus

Proposition: declarative statement which may be either true or false

Example: temperature is 100 degree Celsius
turtle runs faster than the hare
sum of 2 and 3 equals 5

Proposition variable: 1 (0) if proposition is true (false)

Negation *p'* of proposition *p*: 1 (0) if *p* is 0 (1)

Conjunction of propositions *p* and *q* is *pq*: true when both *p* and *q* are true and false whenever either one or both *p* and *q* are false

Disjunction of propositions *p* and *q* is *p* + *q*: true when either *p* or *q* or both are true and false whenever both *p* and *q* are false

29

Propositional Calculus (Contd.)

Definition of conjunction and disjunction of *p* and *q*

p	q	pq	$p+q$
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

Analogy: OR <-> disjunction; AND <-> conjunction

Thus, algebraic system for switching circuits is isomorphic to propositional calculus

30

Propositional Calculus Example

Example: Air-conditioning of a storage warehouse to be turned on if one or more of the following three conditions occurs:

1. Weight of stored material is less than 100 kg, relative humidity is at least 60%, and temperature is above 60 degrees Celsius
2. Weight of stored material is 100 kg or more and temperature is above 60 degrees Celsius
3. Weight of the stored material is less than 100 kg and the barometer stands at 30 inches of mercury or over

A: proposition that air-conditioning is turned on

W: weight of 100 kg or more

H: relative humidity of at least 60%

T: temperature above 60 degrees Celsius

P: barometric pressure is 30 inches of mercury or more

$A = W'HT + WT + W'P$
$= HT + WT + W'P$
$= T(H+W) + W'P$

Thus, air-conditioning is on if the temperature is above 60 degrees Celsius and either the weight is at least 100 kg or the humidity is at least 60%, or if the weight is less than 100 kg and the barometer stands at 30 inches or over

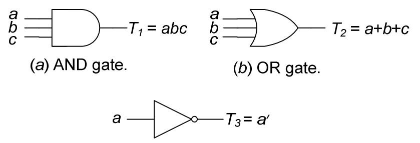
Electronic Gate Networks

Electronic gates: generally receive voltages as inputs and produce output voltages

Precise values of voltages not significant: restricted to value ranges – high (value 1) and low (value 0)

Electronic gates constructed with two-state switching devices: each capable of permitting the flow of current or blocking it

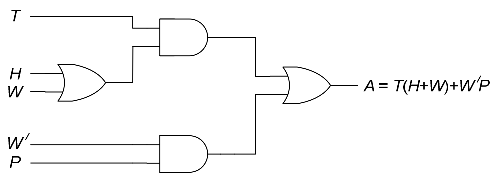
To implement arbitrary switching functions: gates must be able to implement a functionally complete set of operations



(*a*) AND gate. (*b*) OR gate. (*c*) NOT gate.

Functionally complete set

Gate Network for Air-conditioning Function



Boolean Algebras

Boolean algebra *B*: a distributive and complemented lattice

- A set of elements *a*, *b*, *c*, …, and binary operations + and . that satisfy the idempotent, commutative, absorption, and associative laws, and are mutually distributive
- It contains two bounds, 0 and 1, which are the least and greatest elements, respectively
- It has a unary operation of complementation, which assigns to every element in *B* its complement

Complement *a'* of any element *a* in *B* is unique, i.e., there exists only element *a'* such that $a + a' = 1$ and $a \,.\, a' = 0$

- Suppose there exists element *a* which has two complements, b_1 and b_2, i.e., $a + b_1 = 1$, $a \,.\, b_1 = 0$, $a + b_2 = 1$, $a \,.\, b_2 = 0$
- Then $b_1 = b_1 \,.\, 1 = b_1 \,.\, (a + b_2) = b_1 \,.\, a + b_1 \,.\, b_2 = 0 + b_1 \,.\, b_2 = b_1 \,.\, b_2$
- Similar arguments show $b_2 = b_1 \,.\, b_2$.
- Thus, $b_1 = b_2$, proving the uniqueness of the complement

Complements of elements 0 and 1: since by definition 0 + 0' = 1, from the definition of the lub, 0' = 1. Similarly, 1' = 0

De Morgan's Theorem

Prove De Morgan's theorem for two variables:

$(a + b)' = a' . b'$

$(a . b)' = a' + b'$

- We have to show that $(a + b)(a' . b') = 0$ and $(a + b) + a' . b' = 1$
- Applying the distributive law:

  $(a + b) + a'b' = (a + b + a')(a + b + b') = (b + 1)(a + 1) = 1$
- Dual property proved similarly

Definition of a Boolean algebra isomorphic to switching algebra

$+$	0	1	$\cdot$	0	1	
0	0	1	0	0	0	$0' = 1$
1	1	1	1	0	1	$1' = 0$

Example of Boolean algebra:

$+$	0	1	a	b	$\cdot$	0	1	a	b	
0	0	1	a	b	0	0	0	0	0	$0' = 1$
1	1	1	1	1	1	0	1	a	b	$1' = 0$
a	a	1	a	1	a	0	a	a	0	$a' = b$
b	b	1	1	b	b	0	b	0	b	$b' = a$

35