A Brief Intro to the Logic, History, and Applications of Boolean Algebra

20 April 2019

By Joshua Doucet

Introduction

"I visualize a time when we will be to robots what dogs are to humans. And I am rooting for the machines" (Rosen, 2013). This is a quote from 1987 by the famous mathematician Claude E. Shannon who is hugely responsible for laying the foundations of the information age through his work in Boolean Algebra. Boolean algebra is the branch of mathematics that computes true or false outputs from true or false inputs. This branch of mathematics has laid the bedrock of our digital world by combining numerous true or false propositions to represent complex data. Modern societies would not be the same if it were not for some of the brilliant mathematicians of the past that dedicated their time to logic and mathematics.

Applications: Data Storage and Encryption

Boolean algebra has many applications, but the applications that are most prominent today are those that are related to our digital age of information. Without Boolean algebra there would be no computer hardware, no software, and the telecommunications industry would be an unorganized hackable mess. Boolean arguments were used in the mid 1900's to enhances the ciphers being used to encrypt telecommunication messages. This method of cryptography was initially used to mask the details of communication during World War II (Collins, 2002). More information on Boolean algebra in cryptography can be found in the book *Cryptographic Boolean functions and applications* (Cusick, 2017).

Beyond the applications of data transmission, Boolean Algebra has found its most influential application in digital data representation. For example, the number 57 is stored on a computer as 00111001 which is a sequence of 0 and 1 values. The proper sequence of 0 and 1 values is determined through multiple true or false arguments that can be represented as Boolean functions and logic gates. Logic gates are either an idealized or physical gate that implements a Boolean function. In other words, the use of true or false inputs allows logic gates to produces a single true or false output. This is the fundamental concept that allows all computing device to store information. Claude E Shannon is the mathematician often recognized as the founder of information theory, but there are several mathematicians that came before him that discovered the necessary concepts that led to modern computing devices.

Brief History: Aristotle, Boole, and Shannon

George Boole 1815–1864 was an English philosopher and mathematician that was born in Lincoln, England who was primarily self-taught due to financial adversity (Rosen, 2013). Boole studied symbolic logic and authored *The Mathematical Analysis of Logic* in 1848. This titled gave him recognition in the field of mathematics and was the necessary credentials needed to land a job as a professor at Queen's College in Ireland. Boole became famous for his work in the field of logical arguments when he authored the book *The Laws of Thought* in 1854. In this publication he coined the term Boolean Algebra and added concrete equations to Aristotle's ideas on propositional logic. After the publication of this book, Boole married his wife who was also a professor at Queen's university. Nine years later in 1864, Boole passed away due to pneumonia; however, his works in mathematics are still being taught today (Rosen, 2013). His name also remains relevant in modern programming languages. A datatype that can be true or false is often called a Boolean datatype.

Claude E Shannon 1916-2001 took Boole's work to the next level by applying the concepts of Boolean Algebra to data transmission and digital storage. Shannon was born and raised in Michigan and graduated from the University of Michigan in 1936. He continued to study at M.I.T. where he wrote his master's thesis on the differential analyzer, a mechanical computing machine constructed of gears and shifts (Rosen, 2013). Shannon's thesis is a foundational piece on switches in electrical circuits and has become quite famous in the field of Boolean Algebra. Shannon's thesis laid the groundwork for all electronic digital computers. Shannon saw that the binary properties of electrical circuits could be used to represent data electronically (Collins, 2002). In 1940, Shannon obtained his PhD and began to work for Bell Laboratories, the company that would later split into AT&T and Nokia (Nokia Bell Labs, 2019).

The Mathematical Logic behind Boolean Algebra

How does that mathematics behind Boolean Algebra work? The reading that follows assume that the reader has a basic understanding of propositional logic. To begin, Boolean logic is used to determine whether propositions are true or false. When Boolean propositions are combined they can be represented as Boolean functions which also evaluate to true or false. True propositions are typically mapped to the value 1 and false propositions are mapped to 0.

Boolean functions can only have values from the set {0,1} as inputs and outputs. Also, Boolean functions can have multiple inputs, but only a single output, which is in line with the standard definition of a function. In these functions there are three main operators, the Boolean sum "+", Boolean product "·", and the complement "$\overline{C}$" where C represents a Boolean input. The "+" operator in Boolean functions acts as the OR operator, the "·" acts as the AND operator, and the complement operator acts as the negation operator.

Examples using the Boolean sum operator **1+1=1, 1+0=1, 0+1=1, 0+0=0.**

Examples using the Boolean product operator **1·1=1, 1·0=0, 0·1=0, 0·0=0**

In comparing Boolean and propositional arguments, the following symbols are equivalent.

Propositional	Boolean
T	1
F	0
∧ (AND)	.
∨ (OR)	+
¬C (NOT C) where C is a proposition	$\overline{C}$ where C is a Boolean input

Example using equivalent symbols

$1 \cdot 0 + \overline{(0+1)} = 0$ is the Boolean representation of $(\mathbf{T} \wedge \mathbf{F}) \vee \neg(\mathbf{T} \vee \mathbf{F}) \equiv \mathbf{F}$.

Below are 2 tables. The one on the left represents propositional logic equivalences, and the one on the right represents Boolean identities. Notice the similarities.

Equivalence	*Name*
$p \wedge \mathbf{T} \equiv p$ $p \vee \mathbf{F} \equiv p$	Identity laws
$p \vee \mathbf{T} \equiv \mathbf{T}$ $p \wedge \mathbf{F} \equiv \mathbf{F}$	Domination laws
$p \vee p \equiv p$ $p \wedge p \equiv p$	Idempotent laws
$\neg(\neg p) \equiv p$	Double negation law
$p \vee q \equiv q \vee p$ $p \wedge q \equiv q \wedge p$	Commutative laws
$(p \vee q) \vee r \equiv p \vee (q \vee r)$ $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$	Associative laws
$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$ $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	Distributive laws
$\neg(p \wedge q) \equiv \neg p \vee \neg q$ $\neg(p \vee q) \equiv \neg p \wedge \neg q$	De Morgan's laws
$p \vee (p \wedge q) \equiv p$ $p \wedge (p \vee q) \equiv p$	Absorption laws
$p \vee \neg p \equiv \mathbf{T}$ $p \wedge \neg p \equiv \mathbf{F}$	Negation laws

(Source: Rosen, 2013)

Identity	*Name*
$\overline{\overline{x}} = x$	Law of the double complement
$x + x = x$ $x \cdot x = x$	Idempotent laws
$x + 0 = x$ $x \cdot 1 = x$	Identity laws
$x + 1 = 1$ $x \cdot 0 = 0$	Domination laws
$x + y = y + x$ $xy = yx$	Commutative laws
$x + (y + z) = (x + y) + z$ $x(yz) = (xy)z$	Associative laws
$x + yz = (x + y)(x + z)$ $x(y + z) = xy + xz$	Distributive laws
$\overline{(xy)} = \overline{x} + \overline{y}$ $\overline{(x + y)} = \overline{x}\,\overline{y}$	De Morgan's laws
$x + xy = x$ $x(x + y) = x$	Absorption laws
$x + \overline{x} = 1$	Unit property
$x\overline{x} = 0$	Zero property

(Source: Rosen, 2013)

Below are a couple examples of Boolean functions which do not behave identically to standard mathematical functions. Remember, Boolean function only have values from the set {0,1} as both inputs and outputs.

$$F(x, y) = x\overline{y}$$

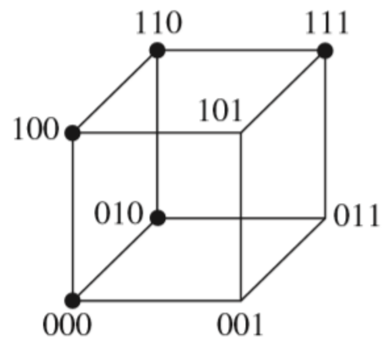
$$F(x, y, z) = xy + \overline{z}.$$

The second function has 3 inputs and 8 possible outputs as shown in the truth table below

x	y	z	xy	$\overline{z}$	$F(x, y, z) = xy + \overline{z}$
1	1	1	1	0	1
1	1	0	1	1	1
1	0	1	0	0	0
1	0	0	0	1	1
0	1	1	0	0	0
0	1	0	0	1	1
0	0	1	0	0	0
0	0	0	0	1	1

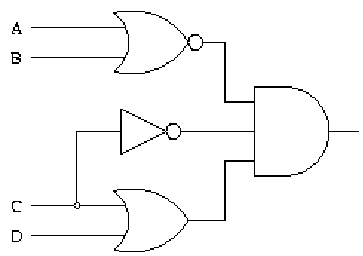
(Source: Rosen, 2013)

When representing Boolean functions graphically, those that have 1, 2, or 3 inputs can simply be drawn on a coordinate grid in 1D, 2D, or 3D. The function $F(x, y, z) = xy + \overline{z}$. graphically is pictured below on an x, y, z coordinate plane. Each vertex represents the output value of the function where a dot represents a true output value of 1 and the lack of a dot represents a false output of 0.

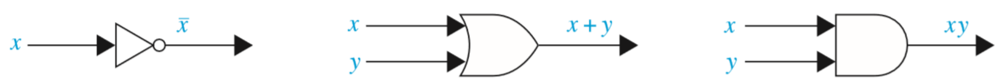


Lastly, functions with more than 3 inputs can be drawn with logic gate flow charts.

$$f(A,B,C,D) = \left(\overline{A+B}\right).(C+D).\overline{C}$$



The three main logic gate symbols are



(a) Inverter
NOT gate

(b) OR gate

(c) AND gate

(Source: Rosen, 2013)

Conclusion

Overall, the age of information and digital storage has invaded societies due to the discoveries by Shannon, Boole, Aristotle, and all the other mathematician's in-between who improved on the concepts of logical arguments. These brilliant minds gave humanity the field of Boolean Algebra which led to improved security in telecommunications, and more importantly the creation of computing devices that use binary bits. These bits have allowed humans as a species to store complex information digitally which can be transmitted worldwide in the blink of an eye. Without Boolean algebra and the great mathematicians behind it, modern digital devices may have never seen the light of day.

References

Boole, G. (1854). *The laws of thought*.

Boole, G. (1848). *The Mathematical Analysis of Logic*.

Collins, G. P. (2002, October 14). Claude E. Shannon: Founder of Information Theory. Retrieved April 30, 2019, from https://www.scientificamerican.com/article/claude-e-shannon-founder/

Cusick, T. W., & Stănică, P. (2017). *Cryptographic Boolean functions and applications*. London: Academic Press.

Nokia Bell Labs. (2019). History. Retrieved April 30, 2019, from https://www.bell-labs.com/about/history-bell-labs/

Rosen, K. H. (2013). *Discrete Mathematics and its Applications*. New York: McGraw-Hill.

Shannon, C. E. (1936). *Switching Circuits and Logical Design* (Master's thesis, MIT, 1936). MIT.