 BOOLEAN ALGEBRAS AND LOGIC

HAO CHENG

Abstract. In this article we investigate the notion and basic properties of Boolean algebras and prove the Stone's representation theorem. The relations of Boolean algebras to logic and to set theory will be studied and, in particular, a neat proof of completeness theorem in propositional logic will be given using Stone's theorem from Boolean algebra. We mention here that the method we used can also be extended to first order logic, yet we will not go for it in this paper.

1. Introduction

The history of Boolean algebras goes back to George Boole (Boole [2]). Boole stated a list of algebraic identities governing the “laws of thought”, i.e. of classical propositional logic. Boole had in mind two interpretations for his identities. One is from logical systems and the other from the “algebra of classes”. Boole’s observation amounted, in algebraic language, to saying that his identities held true under both interpretations. Some 50 years later, the completeness theorem for propositional logic, saying that every identity valid in the two-element Boolean algebra is derivable from Boole’s axioms, and Stone’s representation theorem, which asserts that every Boolean algebra is isomorphic to an algebra of sets, together proved that Boole’s identities give in fact a complete axiomatization for both of his interpretations. As we shall see in this article, the proofs of both results are closely connected.

The article is divided into 4 sections. Section 2 will provide the necessary definitions and notations, and will familiarize the readers with the algorithms through some examples. In section 3 and 4, we will prove Stone’s representation theorem (set-theoretical version) and the completeness theorem for propositional logic in terms of Boolean algebra respectively. We assume the readers with some backgrounds in propositional logic.

For a more detailed history of Boole and Boolean algebras, see Machale [3].

2. Examples and arithmetic of Boolean algebras

In this section, we will give the definition of Boolean algebras and study their arithmetics through some examples.

2.1. Definitions and notations.

Definition 1. A Boolean algebra is a structure \((B, +, \cdot, -, 0, 1)\) with two binary operations + and \(\cdot\), a unary operation \(-\), and two distinguished elements 0 and 1 such that for all \(x, y, \) and \(z\) in \(B\),

(associativity) \((B1)\) \(x + (y + z) = (x + y) + z\), \(x \cdot (y \cdot z) = (x \cdot y) \cdot z\),

(commutativity) \((B2)\) \(x + y = y + x\), \(x \cdot y = y \cdot x\).
These laws might be easier to understand through the examples below:

\[ (\text{absorption}) \quad (B3) \ x + (x \cdot y) = x, \quad x \cdot (x + y) = x, \]
\[ (\text{distributivity}) \quad (B4) \ x \cdot (y + z) = (x \cdot y) + (x \cdot z), \quad x + (y \cdot z) = (x + y) \cdot (x + z), \]
\[ (\text{complementation})(B5) \ x + (\neg x) = 1, \quad x \cdot (\neg x) = 0, \]

**Definition 2.** A structure \((A, +_A, \cdot_A, -_A, 0_A, 1_A)\) is a subalgebra of the Boolean algebra \((B, +_B, \cdot_B, -_B, 0_B, 1_B)\) if \(A \subseteq B\), \(0_A = 0_B, 1_A = 1_B\) and the operations \(+_A, \cdot_A, -_A\) are the restrictions of \(+_B, \cdot_B, -_B\) to \(A\).

When dealing with different Boolean algebras \((A, +_A, \cdot_A, -_A, 0_A, 1_A)\) and \((B, +_B, \cdot_B, -_B, 0_B, 1_B)\), we prefer dropping the subscripts if no confusion arises.

**Definition 3.** A homomorphism from a Boolean algebra \(A\) into a Boolean algebra \(B\) is a map \(f : A \to B\) such that
\[ f(0) = 0, f(1) = 1 \]
and for all \(x, y\) in \(A\),
\[ f(x + y) = f(x) + f(y), \quad f(x \cdot y) = f(x) \cdot f(y) \]
\[ f(\neg x) = \neg f(x) \]

\(f\) is an isomorphism from \(A\) onto \(B\) if it is a bijective homomorphism from \(A\) onto \(B\). \(A\) and \(B\) are isomorphic if there exists an isomorphism from \(A\) onto \(B\).

### 2.2. Arithmetic of Boolean algebras.

We now prove several arithmetic properties of Boolean algebras which we shall encounter later in the article.

**Proposition 1.**
1. \(-x\) is the unique complement of \(x\).
2. (de Morgan’s laws) \(- (x + y) = \neg x \cdot \neg y\) and \(- (x \cdot y) = \neg x + \neg y\).

**Proof.**

(1) By (B5) in Definition 1, \(-x\) is a complement of \(x\). If \(y\) and \(z\) are both complements of \(x\), then
\[
\begin{align*}
z &= z \cdot 1 \\
&= z \cdot (x + y) \quad \text{by } x + y = 1 \\
&= z \cdot x + z \cdot y \quad \text{by } (B4) \\
&= 0 + z \cdot y \quad \text{by } x \cdot z = 0 \\
&= z \cdot y.
\end{align*}
\]

In a similar way, we derive that \(y = y \cdot z\) and \(z = z \cdot y = y \cdot z = y\).

(2) By distributivity and absorption,
\[
\begin{align*}
(x + y) \cdot \neg x \cdot \neg y &= x \cdot \neg x \cdot \neg y + y \cdot \neg x \cdot \neg y \\
&= 0 + 0 \\
&= 0
\end{align*}
\]

and
\[
\begin{align*}
(x + y) + \neg x \cdot \neg y &= x \cdot (y + \neg y) + y + \neg x \cdot \neg y \\
&= x \cdot y + x \cdot \neg y + y + \neg x \cdot \neg y \\
&= y + x \cdot \neg y + x \cdot \neg y \\
&= y + \neg y \\
&= 1,
\end{align*}
\]

Thus \(-x \cdot \neg y\) is the complement of \(x + y\), and by (1), \(- (x + y) = \neg x \cdot \neg y\).

The second part follows in exactly the same way by exchanging the positions of \(+\) and \(\cdot\) and the positions of \(1\) and \(0\) in the proof above.

\( \square \)

These laws might be easier to understand through the examples below:
2.3. Examples of Boolean algebras. Two standard examples of Boolean algebras, algebras of sets and Lindenbaum-Tarski algebras, arise in set theory and logic. The operations +, ·, and − of a Boolean algebra are therefore often written as ∪, ∩, − or ∨, ∧, ¬ and called union, intersection, complement or disjunction, conjunction, negation. Every Boolean algebra is, by Stone’s representation theorem, isomorphic to an algebra of sets, which we shall prove in section 3. The Lindenbaum-Tarski algebras exemplify the connections between Boolean algebras and logic and, it’s also proved that every Boolean algebra is isomorphic to a Lindenbaum-Tarski algebra. For interested readers, the proof can be found in Chapter 9, vol 2, Monk[1]. Nevertheless, it is beyond of our scope in this article.

Example 1. (power set algebras). Let $X$ be any set and $\mathcal{P}(X)$ its power set. The structure $(\mathcal{P}(X), \cup, \cap, -, \phi, X)$ with $-Y$ the complement $X \setminus Y$ of $Y$ with respect to $X$, is a Boolean algebra: the axioms (B1) through (B5) simply state elementary laws of set theory. $\mathcal{P}(X)$ is called the power set algebra of $X$.

Definition 4. A subalgebra of a power set algebra $\mathcal{P}(X)$ is called an algebra of subsets of $X$ or an algebra of sets over $X$. A Boolean algebra is an algebra of sets if it is an algebra of sets over $X$, for some set $X$.

Example 2. (the trivial Boolean algebra). For $X$ the empty set, $\mathcal{P}(X)$ reduces to the Boolean algebra $\mathcal{A} = (\mathcal{P}(X),\cup,\cap,-,0_A,1_A)$ with $0_A = 1_A$, the trivial or one-element Boolean algebra.

Notice that in the example above, the two distinguished elements 0 and 1 are not necessarily distinct.

Example 3. (Lindenbaum-Tarski algebras). Let $L$ be a language for propositional logic and $T$ a theory, i.e. an arbitrary set of sentences, in $L$. For formulas $\alpha, \beta$ of $L$, define

$$\alpha \sim \beta \iff T \vdash \alpha \leftrightarrow \beta$$

i.e. iff $\alpha \leftrightarrow \beta$ is formally provable from the axioms consisting of all tautologies and theorems of $T$ in classical propositional calculus. We can easily tell from propositional logic that this is indeed an equivalence relation. Let $[\alpha]$ be the equivalence class of $\alpha$ with respect to $\sim$ and put

$$\mathcal{B}_L(T) = \{[\alpha] : \alpha \text{ a formula of } L\}$$

Define

$$[\alpha] + [\beta] = [\alpha \lor \beta], \quad [\alpha] \cdot [\beta] = [\alpha \land \beta],$$

$$-[\alpha] = [-\alpha],$$

$$1 = [\alpha_0 \lor -\alpha_0], \quad 0 = [\alpha_0 \land -\alpha_0],$$

where $\alpha_0$ is an arbitrary formula. Again from propositional logic, it can be shown that the operations are well defined. Thus we made $\mathcal{B}_L(T)$ into a Boolean algebra, the Lindenbaum-Tarski algebra of $T$.

Particularly, if we take $T = \phi$, then $\alpha \sim \beta$ iff $\alpha, \beta$ are logically equivalent. By their very definition, the Lindenbaum-Tarski algebras are not algebras of sets. Nonetheless, we will prove the following:
3. Stone’s theorem

The principal result of this section is the following theorem.

**Theorem 1.** (Stone’s representation theorem, set-theoretical version). *Every Boolean algebra is isomorphic to an algebra of sets.*

For the proof of the theorem, we need some further definitions.

**Definition 5.** A filter in a Boolean algebra is a subset \( p \) of \( A \) such that

- \( 1 \in p \),
- for all \( x, y \in A \), \( x \cdot y \in p \) iff \( x \in p \) and \( y \in p \).

For example, for each \( a \in A \), the set \( \{ x \in A : x \cdot a = a \} \) is a filter in \( A \), called the principal filter generated by \( a \).

**Definition 6.** A filter \( p \) of \( A \) is an *ultrafilter* if, for each \( x \in A \), \( x \in p \) or \( -x \in p \) but not both.

From the definition, an ultrafilter must be proper, i.e. \( p \neq A \), since if \( 0 \in p \), then by the definition of a filter, \( x \cdot -x = 0 \) implies that both \( x \) and \( -x \) belong to \( p \), which contradicts \( p \) being an ultrafilter. Actually, if \( 0 \in p \), \( p \) a filter, then \( x \in p \) for every \( x \in A \), that is, \( p = A \).

**Lemma 1.** Every ultrafilter is maximal and vice versa. Here by a filter \( p \) being maximal we mean that it’s proper and there’s no proper filter of \( A \) having \( p \) as a proper subset.

**Proof.** If \( p \) is an ultrafilter of \( A \), \( q \) a filter containing \( p \) as a proper subset, we want to show that \( q \) must be equal to \( A \). Since \( p \) is proper in \( q \), there is an element \( x \) of \( q \) which does not belong to \( p \). It follows from \( p \) being an ultrafilter, \( -x \) must belong to \( p \), and hence to \( q \). Since \( q \) is a filter and \( 0 = x \cdot -x, 0 \) belongs to \( q \) and therefore \( q = A \).

Conversely, if \( p \) is maximal, we prove that \( p \) is an ultrafilter. Let \( x \in A \). Since \( p \) is proper, \( x \) and \( -x \) cannot both be in \( p \). Suppose \( x \notin p \). By maximality of \( p \), the filter generated by \( p \cup \{ x \} \) contains \( 0 \), hence by the definition of filters, \( a \cdot x = 0 \) for some \( a \in p \). Thus, \( a \cdot -x = a \) and \( -x \in p \).

**Lemma 2.** If \( p \) is an ultrafilter, then for all \( x, y \in A \),

1. \( x + y \in p \) implies that either \( x \in p \) or \( y \in p \),
2. \( x \cdot y \in p \) implies that both \( x \in p \) and \( y \in p \).

**Proof.**

1. Assuming \( x \notin p \) and \( y \notin p \), we have to show that \( x + y \notin p \). This follows from \( -x \in p \), \( -y \in p \) and by Proposition 1, \( -x \cdot -y = -(x + y) \in p \)
2. It follows directly from the definition of a filter.

**Proposition 2.** An element \( a \) of a Boolean algebra is contained in an ultrafilter if \( a \neq 0 \).

**Proof.** Assume \( a \neq 0 \), then it’s easy to see that the filter generated by \( a \), say \( p_0 \), is proper, since \( 0 \notin p_0 \). The set \( P \) of all proper filters of \( A \) including \( p_0 \) is non-empty and partially ordered by inclusion, moreover each non-empty chain \( C \) in \( P \) has \( \cup C \) as an upper bound in \( P \). By Zorn’s lemma, let \( p \) be a maximal element of \( P \). Then \( p \) is a maximal filter and includes \( a \); by Lemma 1, it’s an ultrafilter.
Remark. The proof given above uses Zorn’s lemma, an equivalent of the axiom of choice. In fact, it has been proved that the proposition can’t be derived from the axiom system ZF without the axiom of choice. See FEFERMAN [5].

Definition 7. For $A$ a Boolean algebra,

$$\text{Ult} A = \{ p \subseteq A : p \text{ an ultrafilter of } A \} \subset \mathcal{P}(A)$$

is the set of all ultrafilters of $A$. The map $s : A \to \mathcal{P}(\text{Ult} A)$ defined by

$$s(x) = \{ p \in \text{Ult} A : x \in p \}$$

is the Stone map of $A$.

proof of Stone’s representation theorem. We only need to prove that the Stone map defined above is an embedding, i.e. a 1-1 homomorphism.

Obviously $s(0) = \phi$ since every ultrafilter is proper, and $s(1) = \text{Ult} A$ since every filter contains 1. By Lemma2,

$$s(x + y) = \{ p : x + y \in p \} = \{ p : x \in p \} \cup \{ p : y \in p \} = s(x) \cup s(y)$$

$$s(x \cdot y) = \{ p : x \cdot y \in p \} = \{ p : x \in p \} \cap \{ p : y \in p \} = s(x) \cap s(y)$$

and $s(-x) = \{ p : -x \in p \} = \{ p : x \notin p \} = -s(x)$

holds for all $x, y \in A$. Thus $s$ is a homomorphism.

To prove that $s$ is 1-1, let $x \neq y$ in $A$, without loss of generality, suppose $x \cdot -y \neq 0$; let $p$ be an ultrafilter containing $x \cdot -y$, then $x \in p$ and $y \notin p$ which gives $p \in s(x) \setminus s(y)$, so $s(x) \neq s(y)$. For the case $y \cdot -x \neq 0$, just exchange the positions of $x$ and $y$ in the above. This completes the proof.

□

Hence, we have proved that the Stone map $s$ embeds $A$ into $\text{Ult} A$ for arbitrary $A$. In other words, every Boolean algebra $A$ is isomorphic to an algebra of sets. For an exposition of topological version of Stone’s representation theorem, we refer the readers to XU [4].

4. Completeness theorem for propositional logic

Recall the completeness theorem for propositional logic:

Theorem 2. Let $L$ be a language for propositional logic with $V$ as its set of propositional variables and $T$ a theory in $L$. Then $T$ being consistent is equivalent to $T$ having a model, i.e. there is an assignment $h : V \to 2 = \{ \text{false}, \text{true} \}$ under which every formula in $T$ is true.

Let’s see what it means in terms of Boolean algebras. For a propositional logic language $L$ and $T$ a theory in $L$, there arises naturally two Boolean algebras $B_L(\phi)$ and $B_L(T)$ (see Example3) and a homomorphism of projection:

$$\pi : B_L(\phi) \to B_L(T)$$

From the definition in Example 3 it’s easy to see that $T$ being consistent is equal to say that not every formula in $L$ is equivalent, i.e. the Boolean algebra $B_L(T)$ is not trivial. An assignment

$$h : V \to 2 = \{ \text{false}, \text{true} \}$$

is equivalent to a homomorphism:

$$h : B_L(\phi) \to 2 = \{ 0, 1 \}$$
where \(2 = \{0, 1\}\) will be regarded as a Boolean algebra in usual way and, we denote both of the mappings \(h\) since they mean the same. Base on the fact that for every formula \(\alpha\) in \(T\), \([\alpha] = 1\) in \(B_L(T)\), we deduce that every formula in \(T\) is true under \(h\) is equivalent to that there exists \(\tilde{h}\) such that the following diagram commutes:

\[
\begin{array}{ccc}
B_L(\phi) & \xrightarrow{h} & 2 \\
\pi & \downarrow & \\
B_L(T) & \xrightarrow{\tilde{h}} & \\
\end{array}
\]

Now let’s prove the completeness theorem for propositional logic in terms of Boolean algebras by using the machinery of ultrafilters. For the proof, we need the following lemma:

**Lemma 3.** If \(p\) is an ultrafilter of a Boolean algebra \(A\), then its characteristic function \(\chi_p : A \rightarrow 2 = \{0, 1\}\) is a homomorphism from \(A\) into the two-element Boolean algebra, the characteristic homomorphism.

**Proof.** A subset \(p\) of \(A\) is, by Definition 5, a proper filter iff \(\chi_p(1) = 1\), \(\chi_p(0) = 0\) and \(\chi_p\) preserves the operation \(\cdot\). By Lemma 2, if \(p\) is an ultrafilter, then \(\chi_p\) also preserves the operations \(+\) and \(−\). □

**proof of the theorem.** We prove the theorem in two directions:

1. (Soundness) If \(T\) has a model, then there exists homomorphisms \(h : B_L(\phi) \rightarrow 2 = \{0, 1\}\) and \(\tilde{h} : B_L(T) \rightarrow 2 = \{0, 1\}\) such that the diagram(4.1) commutes. Since \(\tilde{h}\) is a homomorphism from \(B_L(T)\) to \(2 = \{0, 1\}\), \(\tilde{h}(0) = 0\), \(\tilde{h}(1) = 1\), there are at least two elements in \(B_L(T)\). Therefore, \(B_L(T)\) is not trivial and \(T\) is consistent.

2. (Completeness) If \(T\) is consistent, i.e. \(B_L(T)\) is not trivial, then by Proposition 2, there exists an ultrafilter \(p \subset B_L(T)\). By Lemma 3, the characteristic function \(\chi_p : B_L(T) \rightarrow 2 = \{0, 1\}\). Take \(h = \chi_p \circ \pi : B_L(\phi) \rightarrow 2 = \{0, 1\}\), then the following diagram commutes:

\[
\begin{array}{ccc}
B_L(\phi) & \xrightarrow{h} & 2 = \{0, 1\} \\
\pi & \downarrow \quad \quad & \\
B_L(T) & \xrightarrow{\tilde{h}} & \\
\end{array}
\]

i.e. \(h\) is an assignment under which every formula in \(T\) is true. □

**Acknowledgements**

I’m glad here to thank my teacher Dr. Shi for his generosity of spending so much time to explain the subjects and give instructions to me, which finally made this article possible.
References

[1] Monk J.D. and Bonnet R., *Handbook of Boolean Algebras Vol.1,2,3*, Amsterdam, Northen-Holland, 1989.

[2] Boole G., *An Investigation of the Laws of Thought*, Cambridge, 1854.

[3] MacHale, Desmond., *George Boole: His Life and Work*, Boole Press, Dublin, 1985

[4] Xu L., *Duality Between Boolean Algebras and Boolean Spaces*, remains unpublished.

[5] Feferman, S., Some applications of the notion of forcing and generic sets, *Fund. Math.*, 56, 325-345.

Department of Mathematics, Beijing Normal University
E-mail address: haocheng05@mail.bnu.edu.cn