Boolean Arithmetics

Anatoly Weinstein

January 19, 2024

In this document we will contain dry theory of boolean equations and arithmetics. For the rest of the paper: $0 \in \mathbb{N}$ holds true.

1 Axioms

Axiom of Binary Choice. All unknowns x in a *boolean system* can take at least one of two values.

$$\forall x : x \in \mathbb{B} = \{\top, \bot\}$$

Boolean Systems. A *boolean system* is a structure, which can be mapped to one and only one function $f : \mathbb{N} \mapsto \mathbb{B}$.

A boolean system is a *tautology* if its' mapping is $f(x) = \top$, a *contradiction*, if $f(x) = \bot$

2 Boolean Functions

We will call Σ the *environment*. Abstractly speaking, the environment of any *boolean system* is the set of all inputs. An element $x \in \Sigma$ is an *interpretation*. A *finite boolean function* $f : \Sigma \to \mathbb{B}$ with $\Sigma = \mathbb{B}^n$, a degree $n \in \mathbb{N}$ and the set of truthy interpretations $I \subseteq \Sigma$ is defined as following:

$$x \in \Sigma : f(x) = \begin{cases} \top, & x \in I \\ \bot, & x \notin I \end{cases}$$

We say the function is *satisfiable* if $|I| > 0$, *unsatisfiable* or a *contradiction* if $|I| = 0$ and a *tautology* if $I = \mathcal{P}(X) = \{Y : Y \subseteq X\}$. An interpretation x is *truthy* if $f(x) = \top$, and *falsy* otherwise. Two boolean functions f_1, f_2 are equal if and only if their truthy interpretations sets I_1, I_2 are equal.

An *infinite boolean function* $f : [0,1]_2 \to \mathbb{B}$ is a *boolean function* of infinite degree. $[0,1]_2$ is the set of all irrational numbers between 0 and 1 in base 2.

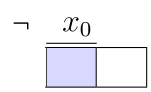
Proof of Truthy. $\top$ (tautalogy) is *truthy*.

$$f : \varnothing \to \mathbb{B} \mapsto \top \Rightarrow I_f = \{\varnothing\} \Rightarrow |I_f| = 1 \Rightarrow \top \text{ is truthy.}$$

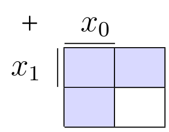
Proof of Falsy. $\bot$ (contradiction) is *falsy*.

$$f : \varnothing \to \mathbb{B} \mapsto \bot \Rightarrow I_f = \varnothing \Rightarrow |I_f| = 0 \Rightarrow \bot \text{ is falsy.}$$

In this document we will mainly work with the 1st-degree function *not* (noted as $\neg$ or overline), and the 2nd-degree functions *and* (noted as $\cdot$) and *or* (noted as $+$) defined as:

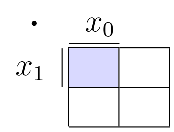


$\neg(x) \in \mathbb{B} \backslash \{x\}$



$+(x_0 x_1) = \top \Leftrightarrow$

$x_0 = \top \vee x_1 = \top$



$\cdot(x_0 x_1) = \top \Leftrightarrow$

$x_0 = x_1 = \top$

3 Boolean Equations

A *boolean equation* is a *boolean system* with *tuple* of variables X, the degree $n = |X|$ and the set of truthy interpretations $I \subseteq \Sigma = \mathbb{B}^n$.

Boolean Equation. A *boolean equation* with environment Σ and binary functions $\mathcal{F} \subseteq \{f : \mathbb{B}^2 \to \mathbb{B}\}$ is a string defined by the following context-free grammar with macro $\mathcal{M}_X$ creating productions for every element of X.

$$S \quad \to \quad (S) \mid S \; \mathcal{M}_{\mathcal{F}} \; S \mid \neg(S) \mid \mathcal{M}_{\mathbb{B}} \mid \mathcal{M}_{\Sigma}$$

The boolean function for this equation is defined by propagating the syntax tree, calling the binary functions while iterating over the branches. For every boolean equation S we assign $\chi_S(X)$

For the rest of the document we say that $\mathcal{F} = \{+, \cdot\}$.

Conjunctive normal form. A boolean equation is in conjunctive normal form, when the following grammar, too, applies.

$$\begin{array}{rcc} S & \rightarrow & (D) \mid S \cdot (D) \\ D & \rightarrow & L \mid D + L \\ L & \rightarrow & \mathcal{M}_\Sigma \mid \neg \mathcal{M}_\Sigma \end{array}$$

A boolean equation in CNF consists of disjunctions D, a disjunction consists of literals L. The notation $(a + \overline{b}) \cdot (b + \overline{c} + \overline{d}) \cdot (\overline{a} + \overline{c})$ will be used as type "set of literal sets" $\{\{a, \neg b\}, \{b, \neg c, \neg d\}, \{\neg a, \neg b\}\}$ in this paper.

Notation of CNF. We will now define some macros for working with CNF-SAT equations. Be S a boolean equation in CNF-SAT, consisting of disjunctions D, $d_i \in D, D_i \subseteq D$, over the environment Σ.

$$D_i =_S D_j \Leftrightarrow f_{D_i \cdot D} = f_{D_j \cdot D}$$

Two disjunctions are equivalent over the equation S if under union with every disjunction from S, the equations are equivalent.