

Propositional Resolution

A deductive system

Benjamin Wack

Université Grenoble Alpes

January 2022

Last course

- ▶ Important Equivalences
- ▶ Substitutions and replacement
- ▶ Normal Forms

John, Peter and Mary by simplification

$$(p \Rightarrow \neg j) \wedge (\neg p \Rightarrow j) \wedge (j \Rightarrow m) \Rightarrow m \vee p$$

$$\neg((p \Rightarrow \neg j) \wedge (\neg p \Rightarrow j) \wedge (j \Rightarrow m)) \vee m \vee p$$

$$\neg(p \Rightarrow \neg j) \vee \neg(\neg p \Rightarrow j) \vee \neg(j \Rightarrow m) \vee m \vee p$$

$$(p \wedge \neg \neg j) \vee (\neg p \wedge \neg j) \vee (j \wedge \neg m) \vee m \vee p$$

with $x \vee (x \wedge y) \equiv x$

$$(\neg p \wedge \neg j) \vee (j \wedge \neg m) \vee m \vee p$$

with $x \vee (\neg x \wedge y) \equiv x \vee y$

$$\neg j \vee j \vee m \vee p = 1$$

Overview

Boolean Algebra

Boolean functions

The BDDC tool

Introduction to resolution

Some definitions and notations

Conclusion

Definition 1.5.1

A **Boolean Algebra** is a set of:

- ▶ at least two elements 0 and 1
- ▶ and three operations, complement (\bar{x}), sum (+) and product (.)
- ▶ such that :
 1. the sum is associative, commutative, with neutral element 0
 2. the product is associative, commutative, with neutral element 1
 3. the product is distributive over the sum
 4. the sum is distributive over the product
 5. negation laws:
 - ▶ $x + \bar{x} = 1$,
 - ▶ $x.\bar{x} = 0$.

Propositional logic is a Boolean Algebra

The axioms can be proven using the truth tables.

Another example:

Boolean Algebra	$\mathcal{P}(X)$
1	X
0	\emptyset
\bar{p}	$X - p$
$p + q$	$p \cup q$
$p \cdot q$	$p \cap q$

Properties of a Boolean Algebra

Property 1.5.3

- ▶ For any x , there is exactly one y such that $x + y = 1$ and $xy = 0$.
In other words, the complement is unique.
- ▶
 1. $\bar{1} = 0$
 2. $\bar{0} = 1$
 3. $\bar{\bar{x}} = x$
 4. $x.x = x$
 5. $x + x = x$
 6. $1 + x = 1$
 7. $0.x = 0$
 8. De Morgan laws:
 - ▶ $\overline{xy} = \bar{x} + \bar{y}$
 - ▶ $\overline{x + y} = \bar{x}.\bar{y}$

Proof

1. $\bar{1} = 0.$

By definition of negation, $x.\bar{x} = 0$. Hence, $1.\bar{1} = 0$.
Since 1 is neutral for the product, $\bar{1} = 0$.

2. $\bar{0} = 1.$

Ditto : $x + \bar{x} = 1$ hence $0 + \bar{0} = 1$.
Since 0 is neutral, $\bar{0} = 1$.

3. $\overline{\bar{x}} = x.$

By commutativity, $\bar{x} + x = 1$ and $\bar{x}.x = 0$.
Because the complement of \bar{x} is unique, $\overline{\bar{x}} = x$.

Proof

4. Product idempotence: $x \cdot x = x$.

$$\begin{aligned}x &= x \cdot 1 \\ &= x \cdot (x + \bar{x}) \\ &= x \cdot x + x \cdot \bar{x} \\ &= x \cdot x + 0 \\ &= x \cdot x\end{aligned}$$

5. Sum idempotence: $x + x = x$

Ditto, starting from $x = x + 0$.

Proof

6. 1 is an absorbing element of the sum: $1 + x = 1$.

We use sum idempotence.

$$\begin{aligned}1 + x &= (x + \bar{x}) + x \\ &= x + \bar{x} \\ &= 1\end{aligned}$$

7. 0 is an absorbing element for the product: $0.x = 0$.

Ditto from $0.x = (x.\bar{x}).x$

Proof: De Morgan Law: $\overline{xy} = \bar{x} + \bar{y}$

We first show that $xy + (\bar{x} + \bar{y}) = 1$

$$\begin{aligned}x.y + (\bar{x} + \bar{y}) &= (x + \bar{x} + \bar{y}).(y + \bar{x} + \bar{y}) \\ &= (1 + \bar{y}).(1 + \bar{x}) \\ &= 1.1 \\ &= 1\end{aligned}$$

Similarly $x.y.(\bar{x} + \bar{y}) = 0$.

Since negation is unique $\bar{x} + \bar{y}$ is the negation of xy .

Similarly we can prove that $\overline{x + y} = \bar{x}.\bar{y}$ by switching the uses of $.$ and $+$ in this demonstration.

Definition 1.6.1: Boolean function

A **boolean function** is a function whose arguments and result belong to the set $\mathbb{B} = \{0, 1\}$.

Example 1.6.2

Which of these functions are boolean ?

- ▶ The function $f : \mathbb{B} \rightarrow \mathbb{B} : f(x) = \neg x$

- ▶ The function $f : \mathbb{N} \rightarrow \mathbb{B} : f(x) = x \bmod 2$

- ▶ The function $f : \mathbb{B} \rightarrow \mathbb{N} : f(x) = x + 1$

- ▶ The function $f : \mathbb{B} \times \mathbb{B} \rightarrow \mathbb{B} : f(x, y) = \neg(x \wedge y)$

Boolean functions and monomial sums

Theorem 1.6.3

Let $x^0 = \bar{x}$ and $x^1 = x$.

Let f be a boolean function with n arguments, and let:

$$A = \sum_{f(a_1, \dots, a_n)=1} x_1^{a_1} \dots x_n^{a_n}.$$

A is the sum of the monomials $x_1^{a_1} \dots x_n^{a_n}$ such that $f(a_1, \dots, a_n) = 1$.

For any assignment v such that $v(x_1) = a_1, \dots, v(x_n) = a_n$, we have $f(a_1, \dots, a_n) = [A]_v$.

Example 1.6.4

The function *maj* with 3 arguments yields 1 when at least 2 of its arguments equal 1.

Define the equivalent sum of monomials (theorem 1.6.3)

x_1	x_2	x_3	$maj(x_1, x_2, x_3)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

$$\bar{x}_1 x_2 x_3$$

$$x_1 \bar{x}_2 x_3$$

$$x_1 x_2 \bar{x}_3$$

$$x_1 x_2 x_3$$

$$maj(x_1, x_2, x_3) = \bar{x}_1 x_2 x_3 + x_1 \bar{x}_2 x_3 + x_1 x_2 \bar{x}_3 + x_1 x_2 x_3$$

Let us verify the theorem 1.6.3 on example 1.6.4

x_1	x_2	x_3	$\text{maj}(x_1, x_2, x_3)$	$\overline{x_1}x_2x_3$	$x_1\overline{x_2}x_3$	$x_1x_2\overline{x_3}$	$x_1x_2x_3$	$\overline{x_1}x_2x_3 + x_1\overline{x_2}x_3 + x_1x_2\overline{x_3} + x_1x_2x_3$
0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0
0	1	1	1	1	0	0	0	1
1	0	0	0	0	0	0	0	0
1	0	1	1	0	1	0	0	1
1	1	0	1	0	0	1	0	1
1	1	1	1	0	0	0	1	1

$$\text{maj}(x_1, x_2, x_3) = \overline{x_1}x_2x_3 + x_1\overline{x_2}x_3 + x_1x_2\overline{x_3} + x_1x_2x_3$$

Proof of Theorem 1.6.3

Let v be any assignment.

Note that for all variable x , $v(x^a) = 1$ if and only if $v(x) = a$.

Thus:

$$[x_1^{a_1} \dots x_n^{a_n}]_v = 1 \quad \text{if and only if} \quad v(x_1) = a_1, \dots, v(x_n) = a_n. \quad (1)$$

Let v be an assignment such that $v(x_1) = a_1, \dots, v(x_n) = a_n$.

Consider the following two cases:

1. $f(a_1, \dots, a_n) = 1$: According to (1), we have $[x_1^{a_1} \dots x_n^{a_n}]_v = 1$.
According to the definition of A , this monomial is the element of the sum A , so $[A]_v = 1$.
2. $f(a_1, \dots, a_n) = 0$: By definition of A , any monomial $x_1^{b_1} \dots x_n^{b_n}$ in A is such that $a_i \neq b_i$ for at least one subscript i .
Consequently $v(x_i) \neq b_i$, so according to (1), $[x_1^{b_1} \dots x_n^{b_n}]_v = 0$.

Since this is true for every monomial in A , we conclude that $[A]_v = 0$.

Boolean functions and product of clauses

Theorem 1.6.5

Let f a boolean function with n arguments, and:

$$A = \prod_{f(a_1, \dots, a_n)=0} x_1^{\overline{a_1}} + \dots + x_n^{\overline{a_n}}.$$

A is the product of the clauses $x_1^{\overline{a_1}} + \dots + x_n^{\overline{a_n}}$ such that $f(a_1, \dots, a_n) = 0$.

For any assignment v such that $v(x_1) = a_1, \dots, v(x_n) = a_n$, we have $f(a_1, \dots, a_n) = [A]_v$.

Proof of theorem 1.6.5

Similar proof:

- ▶ For every variable x , $v(x^a) = 0$ if and only if $v(x) \neq a$.
- ▶ From this remark, we deduce the following property:

$$[x_1^{\bar{a}_1} + \dots x_n^{\bar{a}_n}]_v = 0 \Leftrightarrow v(x_1) \neq \bar{a}_1, \dots v(x_n) \neq \bar{a}_n \quad (2)$$

$$\Leftrightarrow v(x_1) = a_1, \dots v(x_n) = a_n. \quad (3)$$

- ▶ From the above properties, we deduce as before that $f(x_1, \dots, x_n) = A$.

Example 1.6.6

The function *maj* of 3 arguments yields 1 if at least 2 of its arguments equal 1.

Define the equivalent product of clauses (theorem 1.6.5)

x_1	x_2	x_3	$maj(x_1, x_2, x_3)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

$$x_1 + x_2 + x_3$$

$$x_1 + x_2 + \overline{x_3}$$

$$x_1 + \overline{x_2} + x_3$$

$$\overline{x_1} + x_2 + x_3$$

$$maj(x_1, x_2, x_3) = (x_1 + x_2 + x_3)(x_1 + x_2 + \overline{x_3})(x_1 + \overline{x_2} + x_3)(\overline{x_1} + x_2 + x_3)$$

Let us verify theorem 1.6.5 on the example 1.6.6

x_1	x_2	x_3	$\text{maj}(x_1, x_2, x_3)$	$x_1 + x_2 + x_3$	$x_1 + x_2 + \bar{x}_3$	$x_1 + \bar{x}_2 + x_3$	$\bar{x}_1 + x_2 + x_3$	$(x_1 + x_2 + x_3)$ $(x_1 + x_2 + \bar{x}_3)$ $(x_1 + \bar{x}_2 + x_3)$ $(\bar{x}_1 + x_2 + x_3)$
0	0	0	0	0	1	1	1	0
0	0	1	0	1	0	1	1	0
0	1	0	0	1	1	0	1	0
0	1	1	1	1	1	1	1	1
1	0	0	0	1	1	1	0	0
1	0	1	1	1	1	1	1	1
1	1	0	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1

$$\text{maj}(x_1, x_2, x_3) = (x_1 + x_2 + x_3)(x_1 + x_2 + \bar{x}_3)(x_1 + \bar{x}_2 + x_3)(\bar{x}_1 + x_2 + x_3)$$

BDDC (*Binary Decision Diagram based Calculator*)

BDDC is a tool for manipulating propositional formulae developed by Pascal Raymond and available at the following address:

<http://www-verimag.imag.fr/~raymond/home/tools/bddc/>

Plan of the Semester

- ▶ Propositional logic *
- ▶ Propositional resolution
- ▶ Natural propositional deduction

MIDTERM EXAM

- ▶ First order logic
- ▶ Basis for the automatic proof (“first order resolution”)
- ▶ First order natural deduction

EXAM

Deduction methods

- ▶ Is a formula valid?
- ▶ Is a reasoning correct?

Two methods:

The truth tables and transformations

Problem

If the number of variables increases, these methods are very long

Example

By a truth table, to verify

$a \Rightarrow b, b \Rightarrow c, c \Rightarrow d, d \Rightarrow e, e \Rightarrow f, f \Rightarrow g, g \Rightarrow h, h \Rightarrow i, i \Rightarrow j \models a \Rightarrow j$

we must test $2^{10} = 1024$ lines.

Or, by deduction, this is a correct reasoning:

1. By transitivity of the implication, $a \Rightarrow j \models a \Rightarrow j$.
2. By definition, the formula $a \Rightarrow j$ is a consequence of its own.

Today

- ▶ Formalisation of a **deductive system** (with 1 rule)
- ▶ How to prove a formula by **resolution**
- ▶ Some properties of resolution

David Hilbert (1862-1943)

- ▶ Founder of the **formalism** school : mathematics can and should be formalized to be studied.
- ▶ Hilbert's program (1920):
 “*Wir müssen wissen. Wir werden wissen.*”
 as an answer to “*Ignoramus et ignorabimus*”
 - ▶ choose a finite set of axioms to express all of maths
 - ▶ prove it is consistent
 - ▶ design an algorithm that decides whether a proposition can be proved (*Entscheidungsproblem*)
- ▶ *Hilbert-style* deductive systems: axioms such as $\vdash p \Rightarrow (q \Rightarrow p)$
 and a few deduction rules such as
$$\frac{\vdash p \Rightarrow q \quad \vdash p}{\vdash q}$$
- ▶ proofs are thorough but hard to read and to check



Intuition

Formulas are put into CNF (conjunction of clauses), and then we use:

$$a + \bar{b}, b + c \models a + c$$

Definitions

Definition 2.1.1

A clause is identified to the **set** of its literals, so we may say that:

- ▶ A literal is a **member of a clause**.
- ▶ A clause A is **included in a clause** B (or is a **sub-clause** of B).
- ▶ Two clauses are **equal** if they have the same set of literals.

Example 2.1.2

- ▶ The clauses $p + \bar{q}$, $\bar{q} + p$, and $p + \bar{q} + p$ are equal
- ▶ $p \in \bar{q} + p + r + p$
- ▶ $p + \bar{q} \subseteq \bar{q} + p + r + p$
- ▶ $\bar{q} + p + r + p - p = \bar{q} + r$
- ▶ $p + p + p - p = \perp$
- ▶ Adding the literal r to the clause p yields the clause $p + r$
- ▶ Adding the literal p to the clause \perp yields the clause p

Notation

$s(A)$ the set of literals of the clause A .

By convention \perp is the empty clause and $s(\perp) = \emptyset$.

Example 2.1.3

$$s(\bar{q} + p + r + p + \bar{p}) =$$

$$\{\bar{q}, p, r, \bar{p}\}$$

Complementary literal

Definition 2.1.4

We note L^c the **complementary literal** of a literal L :

If L is a variable, L^c is the negation of L .

If L is the negation of a variable, L^c is obtained by removing the negation of L .

Example 2.1.5

$$x^c = \bar{x} \text{ and } \bar{x}^c = x.$$

Resolvent

Definition 2.1.6

Let A and B be two clauses.

The clause C is a **resolvent** of A and B iff there exists a literal L such that

$$L \in A, \quad L^c \in B, \quad C = (A - \{L\}) \cup (B - \{L^c\})$$

“ C is a resolvent of A and B ” is represented by:

$$\frac{A \quad B}{C}$$

C is generated by A and B

A and B are the parents of clause C .

Examples with resolution

Example 2.1.7

Give the resolvents of:

- ▶ $p+q+r$ and $p+\bar{q}+r$

$$\frac{p+q+r \quad p+\bar{q}+r}{p+r}$$

- ▶ $p+\bar{q}$ and $\bar{p}+q+r$

$$\frac{p+\bar{q} \quad \bar{p}+q+r}{\bar{p}+p+r} \quad \frac{p+\bar{q} \quad \bar{p}+q+r}{\bar{q}+q+r}$$

- ▶ p and \bar{p}

$$\frac{p \quad \bar{p}}{\perp}$$

Property

Property 2.1.8

If one of the parents of a resolvent is valid, the resolvent is valid or contains the other parent.

Proof.

See exercise 39. □

Example

$$\frac{p + \bar{p} + q \quad \bar{q} + r}{p + \bar{p} + r} \quad \frac{p + \bar{p} + q \quad \bar{p} + r}{\bar{p} + q + r}$$

Definition of a proof

Definition 2.1.11

Let Γ be a set of clauses and C a clause.

A **proof** of C starting from Γ is a list of clauses:

- ▶ where every clause of the proof is a member of Γ
- ▶ or is a resolvent of two clauses already obtained
- ▶ ending with C .

The clause C is **deduced** from Γ (Γ **yields** C , or Γ **proves** C), denoted $\Gamma \vdash C$, if there is a proof of C starting from Γ .

The **size** of a proof is the number of lines in it.

Example

Example 2.1.12

Let Γ be the set of clauses $\bar{p} + q$, $p + \bar{q}$, $\bar{p} + \bar{q}$, $p + q$.

We show that $\Gamma \vdash \perp$:

1	$p + q$	Hypothesis
2	$p + \bar{q}$	Hypothesis
3	p	Resolvent of 1, 2
4	$\bar{p} + q$	Hypothesis
5	q	Resolvent of 3, 4
6	$\bar{p} + \bar{q}$	Hypothesis
7	\bar{p}	Resolvent of 5, 6
8	\perp	Resolvent of 3, 7

Proof tree

Example 2.1.12

Let Γ be the set of clauses $\bar{p} + q, p + \bar{q}, \bar{p} + \bar{q}, p + q$.

We show that $\Gamma \vdash \perp$:

$$\begin{array}{c}
 \frac{p+q \quad p+\bar{q}}{p} \quad \bar{p}+q \\
 \hline
 q \quad \bar{p}+\bar{q} \\
 \hline
 \bar{p} \quad \frac{p+q \quad p+\bar{q}}{p} \\
 \hline
 \perp
 \end{array}$$

Monotony and Composition

Property 2.1.14

1. **Monotony:** If $\Gamma \vdash A$ and if $\Gamma \subseteq \Delta$ then $\Delta \vdash A$
2. **Composition:** If $\Gamma \vdash A$ and $\Gamma \vdash B$ and if C is a resolvent of A and B then $\Gamma \vdash C$.

Proof.

Exercise 38



Today

- ▶ **Important equivalences** correspond to computation rules in a **Boolean algebra**
- ▶ Any **boolean function** can be represented by a (normal) formula
- ▶ A **deductive system** is given by a set of **formal rules**
- ▶ A **proof** is a sequence of applications of these rules starting from **hypotheses**.

Next course

- ▶ Correctness and Completeness of the system
- ▶ Comprehensive strategy
- ▶ Davis-Putnam

Homework

Hypotheses:

- ▶ (H1): If Peter is old, then John is not the son of Peter
- ▶ (H2): If Peter is not old, then John is the son of Peter
- ▶ (H3): If John is Peter's son then Mary is the sister of John

Conclusion (C): Either Mary is the sister of John or Peter is old.

Transform into clauses the premises and the negation of the conclusion.

What can you (or should you) **prove** using resolution ?