Logic formulae transformations

Stéphane Devismes Pascal Lafourcade Michel Lévy
Course given by Jean-François Monin
(jean-francois.monin@imag.fr)

Université Joseph Fourier, Grenoble I

16 January 2015

Previous lecture

- Introduction and history
- Propositional logic
- Syntax
- Meaning of formulae
- Important Equivalences

Our example with a truth table

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) : Mary is the sister of John, or Peter is old.

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

p	j	m	$A = p \Rightarrow \neg j$	$B = \neg p \Rightarrow j$	$C = j \Rightarrow m$	$A \wedge B \wedge C$	$m \vee p$	$A \wedge B \wedge C \Rightarrow m \vee p$
0	0	0	1	0	1	0	0	1
0	0	1	1	0	1	0	1	1
0	1	0	1	1	0	0	0	1
0	1	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	1	0	0	1	0	0	1	1
1	1	1	0	1	1	0	1	1

Plan

Preamble

How to prove that a formula is valid ?

- Truth table
  - Problem : for a formula having 100 variables, the truth table will contain 2^{100} lines (unable to be computed, even by a computer !).
- Idea :
  - Simplify the formula using substitutions, replacements, or normal form transformations (disjunctive or conjunctive)
  - Then, solve the simplified formula using truth tables or a logic reasoning (for example : important equivalences)

Substitution

Definition 1.3.1

A substitution σ is a function mapping variables to formulae.

$A\sigma$ = the formula A where all variables x are replaced by the formula $\sigma(x)$.

Example : $A = \neg(p \wedge q) \Leftrightarrow (\neg p \vee \neg q)$

- Let σ the following substitution : $\sigma(p) = (a \vee b), \sigma(q) = (c \wedge d)$
- $A\sigma = \neg((a \vee b) \wedge (c \wedge d)) \Leftrightarrow (\neg(a \vee b) \vee \neg(c \wedge d))$

Finite support substitution

Definition 1.3.2 The support of a substitution σ

- The set of variables x such as $x\sigma \neq x$.
- A substitution σ which has finite support is denoted $< x_1 := A_1, \ldots, x_n := A_n >$, where $A_1, \ldots, A_n$ are formulae, $x_1, \ldots, x_n$ are distinct variables and the substitution verifies :
  - $\forall i, i \in 1, \ldots n : x_i\sigma = A_i$
  - $\forall y, y \notin \{x_1, \ldots, x_n\} : y\sigma = y$

Example 1.3.3

$A = x \vee x \wedge y \Rightarrow z \wedge y$ and $\sigma =< x := a \vee b, z := b \wedge c >$
$A\sigma = (a \vee b) \vee (a \vee b) \wedge y \Rightarrow (b \wedge c) \wedge y$

Properties of substitutions

Property 1.3.4

Let A be a formula, v a truth assignment and σ a substitution, we have $[A\sigma]_v = [A]_w$ where for every variable x, $w(x) = [\sigma(x)]_v$.

Example 1.3.5 :

Let $A = x \vee y \vee d$
Let $\sigma = < x := a \vee b, y := b \wedge c >$
Let v so that $v(a) = 1$, $v(b) = 0$, $v(c) = 0$, $v(d) = 0$
$A\sigma = (a \vee b) \vee (b \wedge c) \vee d$
$[A\sigma]_v = max(max(max(1,0), min(0,0), 0)) = 1$
$w(x) = [\sigma(x)]_v = [a \vee b]_v = max(1,0) = 1$
$w(y) = [\sigma(y)]_v = [b \wedge c]_v = min(0,0) = 0$
$w(d) = [\sigma(d)]_v = [d]_v = 0$
$[A]_w = max(max(1,0), 0) = 1$

Proof.

Let A a formula, v a truth assignment and σ a substitution.
Proof by induction on the height (or better : the structure) of A. □

Initial step : $|A| = 0$

Induction

Hypothesis : Suppose the property is true for all formula of height less or equal to n.

Let A a formula of height $n+1$; there are two possible cases :

Induction

Hypothesis : Suppose the property is true for all formula of height less or equal to n. Let A a formula of height $n+1$; there are two possible cases :

Substitution of a valid formula

Theorem 1.3.6

The application of a substitution to a valid formula gives a valid formula.

Proof.

Let A be a valid formula and σ a substitution.
Let v be any truth assignment.

□

Examples

Example

Using substitution, prove that $F = (a \wedge b) \vee \neg a \vee \neg b$ is valid.

Examples

Example 1.3.7

Let A the formula $\neg(p \land q) \Leftrightarrow (\neg p \lor \neg q)$. This formula is valid, it is an important equivalence. Let σ the following substitution : $< p := (a \lor b), q := (c \land d) >$. The formula $A\sigma =$

Replacement

Replace a formula by another formula.

Definition 1.3.8

Let A, B, C, D formulae.
The formula D is obtained by replacing in C certain occurences of A by B
if there exist a formula E and a variable x so that, $C = E < x := A >$ and $D = E < x := B >$.

Examples

Example 1.3.9

Consider the formula $C = ((a \Rightarrow b) \vee \neg(a \Rightarrow b))$.

- The formula obtained by replacing all occurences of $(a \Rightarrow b)$ by $(a \wedge b)$ in C is

  it is obtained considering the formula $E = (x \vee \neg x)$ and the following substitutions $< x := (a \wedge b) >$ et $< x := (a \Rightarrow b) >$.

- The formula obtained by replacing the first occurrence of $(a \Rightarrow b)$ by $(a \wedge b)$ in C is

  it is obtained considering the formula $E = (x \vee \neg(a \Rightarrow b))$ and the following substitution $< x := (a \wedge b) >$ and $< x := (a \Rightarrow b) >$.

Properties of the replacements (1/2)

Theorem 1.3.10

Let C a formula and D the formula obtained by replacing, in C, the occurences of formula A by formula B. We have :
$(A \Leftrightarrow B) \models (C \Leftrightarrow D)$.

Proof.

By definition of the replacement, there is a formula E and a variable x so that, $C = E < x := A >$ et $D = E < x := B >$. Suppose that v is a model truth assignment of $(A \Leftrightarrow B)$. We therefore have $[A]_v = [B]_v$.
According to property 1.3.4 :

- $[C]_v = [E]_w$ where w is identical to v except that $w(x) = [A]_v$
- $[D]_v = [E]_{w'}$ where w' is identical to v except that $w'(x) = [B]_v$

Since $[A]_v = [B]_v$, the truth assignments w and w' are identical, therefore $[C]_v = [D]_v$. Consequently, v is a model of $(C \Leftrightarrow D)$. □

Application of the theorem (Example 1.3.12)

$$p \Leftrightarrow q \models (p \vee (\boxed{p} \Rightarrow r)) \Leftrightarrow (p \vee (\boxed{q} \Rightarrow r)).$$

Properties of the replacements (2/2)

Corollary 1.3.11

Let C a formula and D the formula obtained by replacing, in C, one occurence of formula A by formula B. We have : if $A \equiv B$ then $C \equiv D$.

Proof.

If $A \equiv B$, then the formula $(A \Leftrightarrow B)$ is valid (property 1.2.10), hence the formula $(C \Leftrightarrow D)$ is also valid since, according to theorem 1.3.10, the consequence of $(A \Leftrightarrow B)$. Consequently $C \equiv D$. □

Example 1.3.12

$(\neg(p \vee q) \Rightarrow (\boxed{\neg(p \vee q)} \vee r)) \equiv (\neg(p \vee q) \Rightarrow (\boxed{(\neg p \wedge \neg q)} \vee r))$,
since $\neg(p \vee q) \equiv (\neg p \wedge \neg q)$.

Definitions

Definition 1.4.1

- A literal is a variable or its negation.
- A monomial is a conjunction of literals.
- A clause is a disjunction of literals.

Example 1.4.2

Normal form

Definition 1.4.3

A formula is in normal form if it only contains the operators $\wedge, \vee, \neg$ and the negations are only applied to variables.

Example 1.4.4

The formula $\neg a \vee b$ is in normal form, while the formula $a \Rightarrow b$ is not in normal form, even if it is equivalent to the first one.

1. Equivalence elimination
2. Implication elimination
3. Shifting negations such that they only apply to variables

1. Eliminating an equivalence

Replacing an occurrence of $A \Leftrightarrow B$ by one of the sub-formulae

(a) $(\neg A \vee B) \wedge (\neg B \vee A)$

(b) $(A \wedge B) \vee (\neg A \wedge \neg B)$

Eliminating an implication

Replacing an occurrence of $A \Rightarrow B$ by $\neg A \vee B$

Shifting negations

Replacing an occurrence of

(a) $\neg\neg A$ by A

(b) $\neg(A \vee B)$ by $\neg A \wedge \neg B$

(c) $\neg(A \wedge B)$ by $\neg A \vee \neg B$

Remark 1.4.5 : simplifications

Simplify as soon as possible :

1. Replace a sub-formula of the form $\neg(A \Rightarrow B)$ by $A \wedge \neg B$.
2. Replacing a conjunction by $\perp$ if it contains
   - either a formula and its negation,
   - or a $\perp$
3. Replace a disjunction by $\top$, if it contains
   - either a formula and its negation,
   - or a $\top$
4. Replace $\neg\top$ by $\perp$ and $\neg\perp$ by $\top$
5. Eliminate the $\perp$ from the disjunctions and the $\top$ from the conjunctions
6. Apply the simplifications :
   - $x \vee (x \wedge y) \equiv x$,
   - $x \wedge (x \vee y) \equiv x$,
   - $x \vee (\neg x \wedge y) \equiv x \vee y$
7. Apply the idempotence of the conjunction and the disjunction.

Disjunctive normal form (DNF)

Definition 1.4.6

A formula is in disjunctive normal form (DNF) if and only if it is a disjunction (sum) of monomials.

Distribution of conjuctions over disjuctions
$x \wedge (y \vee z) \equiv (x \wedge y) \vee (x \wedge z)$
The interest of DNFs is to highlight their models.

Example 1.4.7

$(x \wedge y) \vee (\neg x \wedge \neg y \wedge z)$ is a DNF, which has two models

Conjunctive normal form (CNF)

Definition 1.4.11

A formula is a conjunctive normal form (CNF) if and only if it is a conjunction (product) of clauses.

Applying distributivity (unusual) of disjunction over conjunction :

- $A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C)$
- $(B \wedge C) \vee A \equiv (B \vee A) \wedge (C \vee A)$.

The interest of CNF is to highlight their counter-models.

Example 1.4.12

$(x \vee y) \wedge (\neg x \vee \neg y \vee z)$ is a CNF, which has two counter-models.

Example

Transform the formula $(a \Rightarrow b) \Leftrightarrow (\neg b \Rightarrow \neg a)$ in disjunction of monomials (*DNF*) :

Example 1.4.8 et 1.4.13

Transformation in DNF of the following :

$$(a \vee b) \wedge (c \vee d \vee e) \equiv$$

Transformation in CNF of the following :

$$(a \wedge b) \vee (c \wedge d \wedge e) \equiv$$

Utilisation of disjunctions of monomials (DNF)

PURPOSE

Determine if a formula is valid or not.

Let A be a formula whose validity we wish to check :

We transform $\neg A$ in an equivalent disjunction of monomials B

- If $B \equiv \bot$ then $\neg A \equiv \bot$, hence $A \equiv \top$, that is, A is valid
- Otherwise B is equal to a disjunction of nonzero monomials equivalent to $\neg A$, which gives us models of $\neg A$, hence counter-models of A.

Example 1.4.9

Let $A = (p \Rightarrow (q \Rightarrow r)) \Rightarrow (p \wedge q \Rightarrow r)$

Determine if A is valid.

Example 1.4.10

Let $A = (a \Rightarrow b) \wedge c \vee (a \wedge d)$.

Determine if A is valid.

Définition 1.5.1

A Boolean Algebra is a set of at least two elements, 0, 1, and three operations, complement (negation) ($\overline{x}$), sum (disjunction) ($+$) and product (conjunction) (.), which verify the following axioms :

1. the sum is :
   - associative : $x+(y+z)=(x+y)+z$,
   - commutative : $x+y=y+x$,
   - 0 is the neutral element for sum : $0+x=x$,
2. the product is :
   - associative : $x.(y.z)=(x.y).z$,
   - commutative : $x.y=y.x$,
   - 1 is the neutral element for product : $1.x=x$,
3. the product is distributive over the sum : $x.(y+z)=(x.y)+(x.z)$,
4. the sum is distributive over the product : $x+(y.z)=(x+y).(x+z)$,
5. negation laws :
   - $x+\overline{x}=1$,
   - $x.\overline{x}=0$.

Propositional logic is a Boolean Algebra

The axioms can be proven by truth tables.

Another example :

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

FIGURE : Figure 1.1

Property of a Boolean Algebra

Property 1.5.3

- For all x, there is one and only one y such that $x + y = 1$ and $xy = 0$, in other words, the negation is unique.
  (proof can be found in the course support (poly))
- 1. $\overline{1} = 0$
  2. $\overline{0} = 1$
  3. $\bar{\bar{x}} = x$
  4. Product idempotence : $x.x = x$
  5. Sum idempotence : $x + x = x$
  6. 1 is an absorbing element for the sum : $1 + x = 1$
  7. 0 is an absorbing element for the product : $0.x = 0$
  8. De Morgan laws :
     - $\overline{xy} = \bar{x} + \bar{y}$
     - $\overline{x + y} = \bar{x}.\bar{y}$

Proof

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

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

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

Proof

- Product idempotence : $x.x = x$.

Proof

- Sum idempotence : $x + x = x$

Proof

- 1 is an absorbing element of the sum : $1 + x = 1$.
- 0 is an absorbing element for the product : $0.x = 0$.

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

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

Definition

Definition 1.5.5

We denote A^* the dual formula of A, inductively defined as :

- $x^* = x$,
- $0^* = 1$,
- $1^* = 0$,
- $(\neg A)^* = (\neg A^*)$,
- $(A \vee B)^* = (A^* \wedge B^*)$,
- $(A \wedge B)^* = (A^* \vee B^*)$.

Example 1.5.6

$(a.(\bar{b} + c))^* =$

Definition and properties

Theorem 1.5.7

If two formulae are equivalent, their duals are also equivalent.

Corollary 1.5.8

If a formula is valid, its dual is inconsistent.

For the proofs, see exercise 29.

Definition1.5.9 : Boolean equality

A formula A is equal to a formula B in a Boolean Algebra iff :

- A and B are syntactically identical,
- A and B constitute the two members of an axiom of Boolean Algebra,
- B equals A (the equality is symmetrical),
- there is a formula C such that A equals C and C equals B (transitivity of equality),
- there are two formulae C and D such that C equals D and B is obtained by replacing in A an occurrence of C by D.

Theorem 1.5.10

If two formulae are equal in a Boolean Algebra, then their duals are also equal.

Definition 1.6.1 : Boolean function

A boolean function is a function whose arguments and the results belong to the set $\mathbb{B}$ defined as $\{0, 1\}$.

Example 1.6.2

- 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

For every variable x, we set $x^0 = \bar{x}$ and $x^1 = x$.

Let f be a boolean function of n arguments. This function is represented using n variables $x_1, \ldots, x_n$. Let A the following formula :

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

a_i are boolean values and A is the sum of the monomials $x_1^{a_1} \ldots x_n^{a_n}$ such that $f(a_1, \ldots, a_n) = 1$. By agreement, if function f always maps to 0 then $A = 0$.

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

Example 1.6.4

The function *maj* with 3 arguments maps to 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

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

Let us verify the theorem 1.6.3 on example 1.6.4

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

Proof of Theorem 1.6.3

Let v be any assignment.

Boolean functions and product of clauses

Theorem 1.6.5

For every variable x, we set $x^0 = \bar{x}$ and $x^1 = x$.

Let f a boolean function of n arguments. This function is represented using n variables $x_1, \ldots, x_n$. Let A the following formula :

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

Les a_i are boolean values and A is the product of the clauses $x_1^{\overline{a_1}} + \ldots + x_n^{\overline{a_n}}$ such that $f(a_1, \ldots, a_n) = 0$. By agreement, if function f always maps to 1 then $A = 1$.

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

Proof of theorem 1.6.5

The proof of the theorem is a homework.

Let v any assignment. Note that for every variable x, $v(x^a) = 0$ if and only if $v(x) \neq a$. From this remark, we deduce the following property :

$$v(x_1^{\overline{a_1}} + \dots x_n^{\overline{a_n}}) = 0 \quad \Leftrightarrow \quad v(x_1) \neq \overline{a_1}, \dots v(x_n) \neq \overline{a_n} \tag{2}$$

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

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 equals 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

Let us verify theorem 1.6.5 on the example 1.6.6

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

BDDC (*Binary Decision Diagram based Calculator*)

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

http://www-verimag.imag.fr/~raymond/tools/bddc-manual/bddc-manual-pages.html.

Conclusion : Today

- Substitution and replacement
- Normal forms
- Boolean Algebra
- Boolean function
- The BDDC tool

Plan of the Semester

TODAY

- Propositional logic *
- Propositional resolution
- Natural propositional deduction
- First order logic

MIDTERM EXAM

- Basis for the automatic proof (≪ first order resolution ≫)
- First order natural deduction

EXAM

Conclusion : Next course

- Resolution

Conclusion

Thank you for your attention.

Questions ?

Prove by formula simplification our example

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