

# Analyse

Arnaud Durand et Pierre Gervais

September 18, 2016

## Contents

<b>I</b>	<b>Calcul propositionnel</b>	<b>1</b>
<b>1</b>	<b>Syntaxe</b>	<b>1</b>
1.1	Raisonnements . . . . .	3
1.2	Définition alternative de $\mathcal{F}_{\mathcal{P}}$ . . . . .	3
1.2.1	Sous-formule, hauteur, arbre de décomposition . . . . .	3
<b>2</b>	<b>Sémantique</b>	<b>4</b>
<b>3</b>	<b>Exemples de formalisation</b>	<b>5</b>
3.1	Contraintes de compatibilité/exclusion . . . . .	5

## Part I

# Calcul propositionnel

## 1 Syntaxe

Le *calcul propositionnel* est un langage *inductivement et librement engendré* par un ensemble de règles.

C'est à dire qu'une formule ne peut pas être obtenu de deux façons différentes.

**Définition 1.** Soit  $\mathcal{P}$  un ensemble de constantes propositionnelles, on définit  $\mathcal{F}_{\mathcal{P}}$  le calcul propositionnel sur  $\mathcal{P}$  obtenu par les règles suivantes :

- si  $p \in \mathcal{P}$ , alors  $p \in \mathcal{F}_{\mathcal{P}}$
- $\perp \in \mathcal{F}_{\mathcal{P}}$
- si  $F \in \mathcal{F}_{\mathcal{P}}$ , alors  $(\neg F) \in \mathcal{F}_{\mathcal{P}}$
- si  $F, G \in \mathcal{F}_{\mathcal{P}}$  alors  $(F \vee G), (F \wedge G), (F \rightarrow G) \in \mathcal{F}_{\mathcal{P}}$

*Notation 1.* S'il n'y a pas d'ambiguïté, on notera  $\mathcal{F}_{\mathcal{P}} = \mathcal{F}$

**Définition 2.** Une définition alternative de  $\mathcal{F}$  est  $\mathcal{F} = \bigcup_{n \geq 0} \mathcal{F}_n$  où

- $\mathcal{F}_0 = \mathcal{P}$
- $\mathcal{F}_{n+1} = \mathcal{F}_n \cup \{(\neg F) \mid F \in \mathcal{F}_n\} \cup \{(F \star G) \mid F, G \in \mathcal{F}_n, \star \in \{\wedge, \vee, \rightarrow\}\}$ , avec  $n \geq 0$

On définit la *hauteur* d'une formule  $F$  par le plus petit  $n$  tel que  $F \in \mathcal{F}_n$ .

*Remarque 1.* Ce langage est fortement parenthésé et toute formule peut être représentée par un arbre de décomposition.

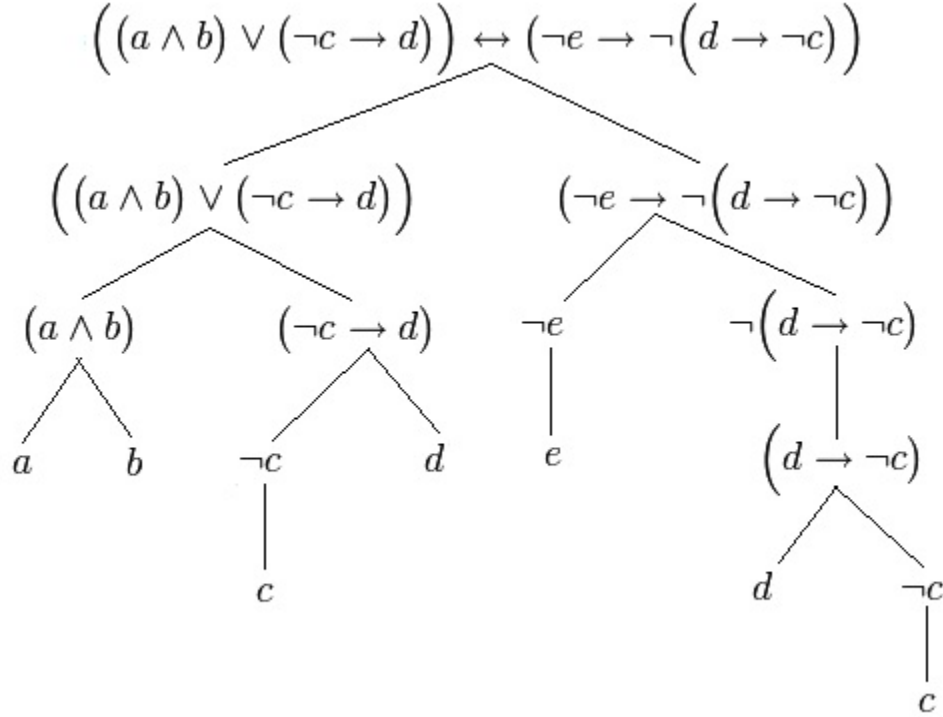


Figure 1: Arbre de décomposition

**Propriété 1.** *Propriété de lecture unique*

Pour tout  $F \in \mathcal{F}$ , un seul de ces cas est vrai :

1.  $F \in \mathcal{P}$
2. Il existe un unique  $G \in \mathcal{F}$  tel que  $F = (\neg G)$
3. Il existe d'uniques  $G, H \in \mathcal{F}$  et  $\star \in \{\vee, \wedge, \rightarrow\}$  tels que  $F = G \star H$

C'est-à-dire que toute formule ne peut se décomposer que d'une seule façon.

## 1.1 Raisonnements

On démontrera généralement les propriétés s'appliquant à  $\mathcal{F}$  par induction : pour démontrer une proposition  $A$  s'appliquant à  $\mathcal{F}$ , on la démontre sur  $\mathcal{P}$  et pour tout  $(F \star G)$  et  $(\neg F)$  où on suppose que  $F, G \in \mathcal{F}$  vérifient  $A$  et  $\star \in \{\wedge, \vee, \rightarrow\}$ .

## 1.2 Définition alternative de $\mathcal{F}_{\mathcal{P}}$

Soit  $\Sigma = \mathcal{P} \cup \{(\,, \neg, \wedge, \vee, \rightarrow, \perp\}$ ,  $\Sigma^*$  est l'ensemble des mots sur  $\Sigma$ .

*Exemple 1.*

- $F = (\wedge \neg x_1)(\in \Sigma^*$
- $F = (\neg x_1) \in \Sigma^*$

**Définition 3.**  $\mathcal{F}$  est le plus petit sous-ensemble de  $\Sigma^*$  contenant  $\mathcal{P} \cup \{\perp\}$  et **clos** par les opérations

1.  $(F, G) \mapsto (F \vee G)$
2.  $(F, G) \mapsto (F \wedge G)$
3.  $(F, G) \mapsto (F \rightarrow G)$

*Remarque 2.* On peut montrer que les deux définitions correspondent.  $\mathcal{F}$  satisfait la propriété de lecture unique (voir TD).

### 1.2.1 Sous-formule, hauteur, arbre de décomposition

**Définition 4.** Soit  $F \in \mathcal{F}$ , on définit  $\mathcal{S}(F)$  l'ensemble des *sous-formules* de  $F$  telles que

- si  $F \in \mathcal{P}$ ,  $\mathcal{S}(F) = \{F\}$
- si  $F = (\neg G)$  alors  $\mathcal{S}(F) = \{F\} \cup \mathcal{S}(G)$
- si  $F = (G \star H)$  où  $\star \in \{\wedge, \vee, \rightarrow\}$ , alors  $\mathcal{S}(F) = \{F\} \cup \mathcal{S}(G) \cup \mathcal{S}(H)$

TODO : vérifier dernier point

**Définition 5.** Soit  $F \in \mathcal{F}$  on définit la *hauteur*  $h(F)$  de  $F$  par

- $h(F) = 0$ , si  $F \in \mathcal{P}$
- si  $F = (\neg G)$ , alors  $h(F) = 1 + h(G)$
- si  $F = (G \star H)$ , alors  $h(F) = 1 + \max\{h(G), h(H)\}$

**Définition 6.** Soit  $F \in \mathcal{F}$ , l'*arbre de décomposition* de  $F$   $arb(F)$  est un graphe étiqueté défini par

1. si  $F \in \mathcal{P}$ ,  $arb(F)$  est réduit à un sommet étiqueté par  $F$ .
2. si  $F = (\neg G)$ , alors  $arb(F) = \neg - arb(G)$
3. si  $F = (G \star H)$ , alors  $arb(F) = G - \star - H$

*Notation 2.* Soit  $F$  une formule,  $var(F)$  est l'ensemble des variables de  $F$ ,  $occ(F)$  est le multi-ensemble des variables de  $F$  et  $arb(F)$  est le graphe

- dont les sommets sont  $V$
- et muni d'une fonction d'étiquetage  $\lambda : V \longrightarrow \{\neg, \perp, \vee, \wedge, \rightarrow\} \cup var(F)$ .

*Remarque 3.* Toutes les définitions sont univoques par la propriété de lecture unique.

*Remarque 4.* On définit la hauteur d'une formule par la hauteur de son arbre de décomposition, c'est-à-dire la distance maximum entre les feuilles et la racine.

*Notation 3.*

- $\top$  comme abréviation pour  $(\perp \rightarrow \perp)$
- $(p \longleftrightarrow q)$  pour  $(p \leftarrow q) \wedge (p \rightarrow q)$
- $\bigwedge_{i=1}^n A_i = (((A_1 \wedge A_2) \wedge A_3) \dots \wedge A_n)$

## 2 Sémantique

On s'intéresse à des propositions dont la valeur de vérité est soit vrai soit faux. On a besoin d'une **interprétation** (en terme de vrai ou faux) de ces constantes propositionnelles.

**Définition 7.** Une *valuation* est une fonction  $v : \mathcal{P} \longrightarrow \{0, 1\}$  (on l'appelle aussi interprétation). Étant donné une valuation  $v$ , on définit l'*interprétation*  $\bar{v} : \mathcal{F} \longrightarrow \{0, 1\}$  comme ceci

- si  $F = p \in \mathcal{P}$  alors  $\bar{v} = v(p)$
- si  $F = (\neg G) \in \mathcal{P}$  alors  $\bar{v}(F) = 1$  si et seulement si  $\bar{v}(G) = 0$
- $\bar{v}(\perp) = 0$
- $\bar{v}(F \wedge G) = 1$  si et seulement si  $\bar{v}(F) = \bar{v}(G) = 1$

On peut décrire l'interprétation d'une formule par sa *table de vérité* :

$F$	$G$	$\neg G$	$F \wedge G$	$F \rightarrow G$
0	0	1	0	1
0	1	1	0	1

On définit formellement la *table de vérité* par une fonction  $v : \{0, 1\}^{\mathcal{P}} \longrightarrow \{0, 1\}$

**Définition 8.**

- $F \in \mathcal{F}$  est dit *satisfaisable* s'il existe une valuation  $v$  de  $\mathcal{P}$  tel que  $\bar{v}(F) = 1$
- $F$  est dit *valide* si pour toute valuation  $v$  de  $\mathcal{P}$ ,  $\bar{v}(F) = 1$ , on dit aussi que  $F$  est une *tautologie*.
- $F$  et  $G$  sont dites *équivalentes*, notées  $F \equiv G$ , si pour toute valuation  $v$ ,  $\bar{v}(F) = \bar{v}(G)$

*Exercice 1.* Vérifier que  $F \equiv G$  si et seulement si  $F \leftrightarrow G$  est valide.

**Proposition 1.** Pour tout  $F \in \mathcal{F}$ ,  $F$  est satisfaisable si et seulement si  $(\neg F)$  n'est pas valide.

### 3 Exemples de formalisation

#### 3.1 Contraintes de compatibilité/exclusion

**Problème :** On possède  $n$  produits chimiques à ranger dans  $k \leq n$  conteneurs. Certains produits ne peuvent pas être stockés ensemble dans un conteneur.

La contrainte est donnée sous la forme d'un ensemble  $\mathcal{L} \subseteq [n]$  tel que  $I = \{i_1, \dots, i_k\} \subseteq \mathcal{L}$  si et seulement si les produits  $i_1, \dots, i_k$  ne peuvent pas être stockés ensemble.

**Enjeu :** Écrire une formule propositionnelle  $F$  telle que  $F$  est satisfaisable si le problème a une solution.

Les variables propositionnelles  $\mathcal{P} = p(i, j)$ ,  $i \leq n$ ,  $j \leq k$  sont interprétées par "le produit chimique  $i$  est dans le camion  $j$ ".

On exprime deux propositions :

$$\begin{aligned}
 & \text{- Chaque produit se trouve dans un unique conteneur : } F = \underbrace{\left( \bigwedge_{i \leq n} \left( \bigvee_{j \leq k} p(i, j) \right) \right)}_{\text{Chaque produit } i \text{ est stocké dans au moins un camion } j} \wedge \underbrace{\left( \bigwedge_{i \leq n} \left( \bigwedge_{\substack{j, j' \leq k \\ j \neq j'}} (\neg(p(i, j) \wedge p(i, j'))) \right) \right)}_{\substack{\text{Pour chaque produit } i \\ \text{et chaque paire de camions } j \neq j' \\ \text{il est faux que } i \text{ est à la fois dans } j \text{ et } j'}} \\
 & \text{- On respecte les incompatibilités : } G = \underbrace{\bigwedge_{I \subseteq \mathcal{L}} \left( \bigwedge_{j \leq k} \neg \left( \bigwedge_{i \in I} p(i, j) \right) \right)}_{\substack{\text{Pour chaque ensemble } I \text{ de produits} \\ \text{ne pouvant pas être stockés ensemble} \\ \text{et pour chaque camion } j, \\ \text{aucun produit de } I \text{ n'est présent dans le camion}}}
 \end{aligned}$$