

# Chapitre 25

## Logique propositionnelle

### Sommaire.

1	Formules propositionnelles.	1
1.1	Syntaxe.	1
1.2	Sémantique.	1
1.3	Satisfiabilité.	2
2	Formes normales.	2
2.1	Formes normales négatives.	2
2.2	Formes normales conjonctives.	2
2.3	Algorithme de Quine	3
2.4	Forme normale disjonctive.	4

Les propositions marquées de ★ sont au programme de colles.

### 1 Formules propositionnelles.

#### 1.1 Syntaxe.

Définition 1: Logique propositionnelle.

Soit  $\mathbb{V}$  un ensemble fini (ou dénombrable) de symboles appelés variables propositionnelles. On définit inductivement l'ensemble des formules propositionnelles sur  $\mathbb{V}$  :

- $\perp$  et  $\top$  sont des expressions logiques, **Faux** et **Vrai** respectivement.
- $p$  est une variable propositionnelle de  $\mathbb{V}$ .
- À partir de  $\varphi$  et  $\psi$  deux formules, on peut construire :
  - $(\varphi \wedge \psi)$  (conjonction).
  - $(\varphi \vee \psi)$  (disjonction).
  - $(\neg \varphi)$  (négation).

Ici,  $\varphi$  et  $\psi$  désigneront toujours des formules.  
Toute formule propositionnelle peut être représentée par un arbre : avec les variables propositionnelles en tant que feuilles, et les constructeurs en tant que noeuds internes.

#### 1.2 Sémantique.

Définition 2: Valuation.

Une **valuation** sur  $\mathbb{V}$  est une application  $v : \mathbb{V} \rightarrow \{0, 1\}$ .  
On étend cette application aux formules propositionnelles :  
Soient  $\varphi, \psi$  des formules propositionnelles. On définit inductivement  $v(\varphi)$  tel que :

- $v(\perp) = 0$ .
- $v(\top) = 1$ .
- $v(\varphi) = v(\varphi)$  si  $\varphi \in \mathbb{V}$ .
- $v(\varphi \wedge \psi) = v(\varphi) \times v(\psi)$ .
- $v(\varphi \vee \psi) = v(\varphi) + v(\psi) - v(\varphi) \times v(\psi)$ .
- $v(\neg \varphi) = 1 - v(\varphi)$ .

Ici,  $v$  désignera toujours une valuation.

Définition 3: Équivalence logique. ★

Deux formules  $\varphi$  et  $\psi$  sont **sémantiquement équivalentes** si pour toute valuation  $v$  sur  $\mathbb{V}$ ,  $v(\varphi) = v(\psi)$ .  
On note alors  $\varphi \equiv \psi$ . Ainsi,  $\equiv$  est une relation d'équivalence sur les formules.

**Remarque:** Dans la pratique, on compare les tables de vérité de  $\varphi$  et  $\psi$ .

Définition 4: Autres constructeurs.

Il existe des liens logiques qui s'expriment à partir de ceux de base :

- L'implication  $\varphi \rightarrow \psi \equiv \neg \varphi \vee \psi$ .
- L'équivalence  $\varphi \leftrightarrow \psi \equiv \varphi \rightarrow \psi \wedge \psi \rightarrow \varphi$ .
- Vrai :  $\top \equiv \varphi \vee \neg \varphi$ .
- Faux :  $\perp \equiv \varphi \wedge \neg \varphi$ .

**Proposition 5: Lois de De Morgan. ★**

Soient  $\varphi$  et  $\psi$  deux formules logiques. Alors :

- $\neg(\varphi \wedge \psi) \equiv \neg\varphi \vee \neg\psi.$
- $\neg(\varphi \vee \psi) \equiv \neg\varphi \wedge \neg\psi.$

**Preuve :**

On le montre facilement en comparant les tables de vérité.

**1.3 Satisfiabilité.**

**Définition 6: Modèles.**

Soit  $\varphi$  une formule sur  $\mathbb{V}$ . Une valuation  $v : \mathbb{V} \rightarrow \mathcal{B}$  est un **modèle** de  $\varphi$  si  $v(\varphi) = 1$ .

On note alors  $v \models \varphi$ .

On dit alors qu’une formule est **satisfiable** si elle admet un modèle.

Une formule  $\varphi$  pour laquelle toute valuation est un modèle est une tautologie, on note  $\models \varphi$ .

Si aucune valuation n’en est un modèle,  $\varphi$  est une antilogie, on note  $\not\models \varphi$ .

**Définition 7: Conséquence logique.**

Soient  $\varphi, \psi$  deux formules.

On dit que  $\varphi$  est en **conséquence logique** de  $\psi$ , et on note  $\psi \models \varphi$  si tout modèle de  $\psi$  est modèle de  $\varphi$ .

On étend cette notation à un ensemble  $\Gamma$  de formules, dans ce cas, on dit que  $\varphi$  est une **conséquence logique** de  $\Gamma$  si  $\varphi$  est en **conséquence logique** de toute formule de  $\Gamma$ .

**Définition 8: Équisatisfiabilité**

Deux formules  $\varphi$  et  $\psi$  sont **équisatisfiables** si  $\varphi$  est satisfiable si et seulement si  $\psi$  l’est.

**2 Formes normales.**

**2.1 Formes normales négatives.**

**Définition 9: Littéral.**

On appelle **littéral** une variable propositionnelle ou sa négation.

**Définition 10: Construction. ★**

Une formule est dite en **forme normale négative (FNN)** si ses négations ne s’appliquent qu’aux variables. Pour une formule  $\varphi$ , on construit sa forme normale négative  $\text{nnF}(\varphi)$  inductivement de la manière suivante :

- $\text{nnF}(\varphi) = \varphi$  si c’est un littéral.
- $\text{nnF}(\neg\neg\varphi) = \text{nnF}(\varphi)$
- $\text{nnF}(\varphi \wedge \psi) = \text{nnF}(\varphi) \wedge \text{nnF}(\psi)$
- $\text{nnF}(\varphi \vee \psi) = \text{nnF}(\varphi) \vee \text{nnF}(\psi)$
- $\text{nnF}(\neg(\varphi \vee \psi)) = \text{nnF}(\neg\varphi) \wedge \text{nnF}(\neg\psi)$
- $\text{nnF}(\neg(\varphi \wedge \psi)) = \text{nnF}(\neg\varphi) \vee \text{nnF}(\neg\psi)$

**Proposition 11: Existence.**

Pour toute formule  $\varphi$ ,  $\text{nnF}(\varphi)$  est sous forme normale négative et  $\text{nnF}(\varphi) \equiv \varphi$ .

**Preuve :**

Par induction sur les formules propositionnelles.

**Cas de base.** Soit  $\varphi$  un littéral.  $\text{nnF}(\varphi) = \varphi$  sous FNN et  $\text{nnF}(\varphi) \equiv \varphi$ .

**Hérédité:** Soient  $\varphi, \psi$  telles que la propriété soit vraie sur elles-mêmes et leurs négations.

Soit  $v$  une valuation de  $\varphi$  et  $\psi$ .

On a  $\text{nnF}(\varphi \wedge \psi) = \text{nnF}(\varphi) \wedge \text{nnF}(\psi)$  donc c’est bien sous forme normale négative par hypothèse.

De plus,  $v \models \text{nnF}(\varphi \wedge \psi) = \iff v \models \text{nnF}(\varphi) \wedge \text{nnF}(\psi) \iff v \models \varphi \text{ et } v \models \psi \iff v \models \varphi \wedge \psi$ .

On a  $\text{nnF}(\neg(\varphi \wedge \psi)) = \text{nnF}(\neg\varphi) \vee \text{nnF}(\neg\psi)$  donc c’est bien sous forme normale négative par hypothèse.

De plus,  $v \models \text{nnF}(\neg(\varphi \wedge \psi)) \iff v \models \text{nnF}(\neg\varphi) \vee \text{nnF}(\neg\psi) \iff v \models \neg\varphi \text{ ou } v \models \neg\psi \iff v \models \neg\varphi \vee \neg\psi \iff v \models \neg(\varphi \wedge \psi)$ .

Même raisonnement pour la disjonction.

Par théorème d’induction, c’est vrai pour toute formule  $\varphi$ .

**2.2 Formes normales conjonctives.**

**Définition 12: Problème SAT.**

Le problème SAT prend une formule en entrée et répond à la question : "Cette formule est-elle satisfiable ?".

**Définition 13: Clause.**

Une **clause** est une disjonction de littéraux.

**Définition 14: Forme normale conjonctive. ★**

Une formule est en **forme normale conjonctive (FNC)** si elle est une conjonction de clauses.  
On définit inductivement la mise sous FNC de  $\varphi$  en  $\text{cnF}(\varphi)$  par :

- $\text{cnF}(\varphi) = \varphi$  si  $\varphi$  littéral.
- $\text{cnF}(\varphi \vee \psi) = \varphi \vee \psi$  si  $\varphi, \psi$  littéraux.
- $\text{cnF}(\varphi \wedge \psi) = \text{cnF}(\varphi) \wedge \text{cnF}(\psi)$ .
- $\text{cnF}(\varphi \vee (\psi \wedge \psi')) = \text{cnF}(\varphi \vee \psi) \wedge \text{cnF}(\varphi \wedge \psi')$ .
- $\text{cnF}(\varphi \vee (\psi \vee \psi')) = \text{cnF}(\varphi \vee \text{cnF}(\psi \vee \psi'))$ .

**Proposition 15**

Si  $\varphi$  est une formule sous FNN,  $\text{cnF}(\varphi)$  est sous FNC et  $\text{cnF}(\varphi) \equiv \varphi$ .

**Proposition 16**

Si  $\varphi$  est sous FNN, on peut construire une FNC équisatisfiable à  $\varphi$  en temps linéaire.

**2.3 Algorithme de Quine**

**Définition 17: Substitution.**

Soit  $\varphi$  une formule sur un esemble  $\{p_1, \dots, p_n\}$  et soient  $\{\varphi_1, \dots, \varphi_n\}$  des formules.  
La substitution des  $\varphi_i$  aux  $p_i$  est la formule obtenue en remplaçant simultanément chaque  $p_i$  par  $\varphi_i$ .  
On la note  $\varphi[\varphi_1/p_1, \dots, \varphi_n/p_n]$ .

La substitution se définit inductivement :

- $\varphi[\varphi_i/p_i] = \varphi_i$  si  $\varphi = p_i$ .
- $\varphi[\varphi_1/p_1, \dots, \varphi_n/p_n] = \neg\varphi'[\dots]$  si  $\varphi = \neg\varphi'$ .
- $\varphi[\dots] = \varphi_1[\dots] \wedge \varphi_2[\dots]$  si  $\varphi = \varphi_1 \wedge \varphi_2$ .
- $\varphi[\dots] = \varphi_1[\dots] \vee \varphi_2[\dots]$  si  $\varphi = \varphi_1 \vee \varphi_2$ .

**Proposition 18**

Une substitution dans une tautologie donne une tautologie.

**Preuve :**

Soit  $\varphi$  sur  $\{p_1, \dots, p_n\}$  et  $\{\varphi_1, \dots, \varphi_n\}$  des formules sur  $\mathbb{V}$ .  
Soit  $v$  une valuation sur  $\mathbb{V}$  et  $\omega$  sur  $\{p_1, \dots, p_n\} : \omega(p_i) = v(\varphi_i)$ .  
Montrons que  $\omega(\varphi) = v(\varphi[\dots])$ .  
**Cas de base.** Trivial si  $\varphi = \top$  ou  $\varphi = \perp$ .  
Si  $\varphi = p_i$ , alors  $\varphi[\dots] = \varphi_i$  et  $\omega(\varphi) = \omega(p_i) = v(\varphi_i)$ .  
**Hérédité.**  
Si  $\varphi = \neg\varphi'$ ,  $\omega(\varphi) = \omega(\neg\varphi') = \neg\omega(\varphi') = \neg v(\varphi'[\dots]) = v(\neg\varphi'[\dots]) = v(\varphi[\dots])$ .  
Si  $\varphi = \varphi_1 \vee \varphi_2$ ,  $\omega(\varphi) = \omega(\varphi_1 \vee \varphi_2) = \omega(\varphi_1) \vee \omega(\varphi_2) = v(\varphi_1[\dots]) \vee v(\varphi_2[\dots]) = v(\varphi_1[\dots] \vee \varphi_2[\dots]) = v(\varphi)$ .  
De même pour la conjonction, avec  $\varphi_1, \varphi_2$  vérifiant l'hypothèse.  
Par principe d'induction structurelle, la propriété est vérifiée.

Supposons  $\varphi$  une tautologie. Soit  $v$  une valuation de la formule substituée., il existe  $\omega$  telle que  $\omega(\varphi) = v(\varphi[\dots])$ .  
Comme  $\varphi$  est tautologie,  $\omega(\varphi) = 1$  donc  $v(\varphi[\dots]) = 1$  donc  $v \models \varphi[\dots]$ , c'est une tautologie.

**Définition 19: Algorithme de Quine.**

**Entrée:**  $\varphi$  sous FNC.  
**Sortie:** 1 si  $\varphi$  est satisfiable, 0 sinon.  
1. Simplifier les clauses.  
2. Si  $\varphi$  est une conjonction sur  $\emptyset$ , renvoyer 1.  
3. Si  $\varphi$  contient  $\perp$ , renvoyer 0.  
4. Choisir la prochaine variable  $p$  dans l'une des clauses :

- Si  $\text{Quine}(\varphi[\perp/p])$ , renvoyer 1, sinon renvoyer  $\text{Quine}(\varphi[\top/p])$ .

**Étape 1:**

- Si la clause est  $\top$ , la supprimer.
- Tiers-exclu : les clauses contenant des littéraux opposés sont supprimées.
- Fusion : supprimer les doublons de littéraux.
- Si une clause en contient une autre, on la supprime.
- Si une clause contient  $\perp$ , le supprimer.

**Terminaison:** Toutes les opérations s'effectuent en temps fini.  
Il y a un nombre fini d'appels récursifs : variant d'appel donnée par le nombre de variables apparaissant dans la formule.  
**Correction:** assurée par le tiers-exclu.

2.4    Forme normale disjonctive.

Définition 20: Conjonction élémentaire.

Une **conjonction élémentaire** est une formule sans disjonctions.

Définition 21: Forme normale disjonctive. ★

Une formule est une **forme normale disjonctive (FND)** si c’est une disjonction de conjonctions élémentaires. Pour passer de  $\varphi$  sous FNN à  $\text{dnF}(\varphi)$  sous FND, on procède par induction :

- $\text{dnF}(\varphi) = \varphi$  si  $\varphi$  est littéral.
- $\text{dnF}(\varphi) = \varphi$  si  $\varphi = l \wedge l'$  avec  $l, l'$  littéraux.
- $\text{dnF}(\varphi \vee \psi) = \text{dnF}(\varphi) \vee \text{dnF}(\psi)$ .
- $\text{dnF}(\varphi \wedge (\psi \vee \psi')) = \text{dnF}(\varphi \wedge \psi) \vee \text{dnF}(\varphi \wedge \psi')$ .
- $\text{dnF}(\varphi \wedge (\psi \wedge \psi')) = \text{dnF}(\varphi \wedge \text{dnF}(\psi \wedge \psi'))$ .

Proposition 22

Si  $\varphi$  est sous FNN,  $\text{dnF}(\varphi)$  est sous FND et  $\text{dnF}(\varphi) \equiv \varphi$ .

Preuve :

Pour tout modèle  $v$  de  $\varphi$ , on construit :

$$\varphi_v = \bigwedge_{p \in \mathbb{V}} l_p \quad \text{où} \quad l_p = \begin{cases} p & \text{si } v(p) = 1 \\ \neg p & \text{sinon} \end{cases}.$$

On pose alors  $\psi$  la disjonction des  $\varphi_v$  pour tout modèle  $v$  de  $\varphi$ .  
On obtient alors  $\psi$  sous FND et  $\psi \equiv \varphi$ .

Définition 23

Une FND est complète si chaque variable est représentée une unique fois dans chaque conjonction élémentaire.