Licence de Sciences Economiques et de Gestion

Introduction à la Logique

Joëlle Cohen

11 juillet 2006

Introduction

Ce résumé de cours ne prétend pas être exhaustif ni se substituer en aucune manière aux ouvrages publiés sur ce sujet. Ce document n'a d'autre but que de fournir aux étudiants un support de cours qui n'est qu'une petite introduction à la logique.

Table des matières

Chapitre 1

Calcul propositionnel

Le calcul propositionnel permet une modélisation du raisonnement mathématique simple : il traite des problèmes où les assertions ne peuvent prendre que deux valeurs possibles. Le calcul propositionnel a une propriété fondamentale : il est complet c'est-à-dire que tout ce qui est vrai est démontrable. Mais ce cadre ne suffit pas à décrire certaines situations mathématiques courantes comme l'existence d'un objet satisfaisant à une propriété donnée.

De façon générale, la logique fait apparaître la différence entre la *syntaxe* – les règles formelles de manipulation des symboles utilisés – et la *sémantique* – l'interprétation des formules.

1.1 Syntaxe

1.1.1 Définition des formules propositionnelles

Soit $\mathcal{A}$ un ensemble de symboles constitué de la façon suivante :
- $\longrightarrow, \neg, (,) \ \in \mathcal{A}$
- $T, F \in \mathcal{A}$
- $\mathcal{V} \subset \mathcal{A}$ où $\mathcal{V}$ est un ensemble dont les éléments sont appelés variables propositionnelles ou atomes

$\mathcal{A} = \{\longrightarrow, \neg, (,), T, F\} \cup \mathcal{V}$

Définition 1.1 *L'ensemble $\mathcal{P}$ des formules propositionnelles construites sur $\mathcal{V}$ est défini inductivement par*
- *$T \in \mathcal{P}$, $F \in \mathcal{P}$, $\mathcal{V} \subset \mathcal{P}$ (les éléments de $\mathcal{V}$ sont alors appelés formules atomiques)*
- *si $\varphi \in \mathcal{P}$ alors $\neg\varphi \in \mathcal{P}$*
- *si $\varphi, \psi \in \mathcal{P}$ alors $(\varphi \longrightarrow \psi) \in \mathcal{P}$*

$\longrightarrow, \neg$ sont des symboles logiques.

***exemple* 1.1** : Soit $\mathcal{V} = \{A, B, C\}$.

$\varphi = (A \longrightarrow (B \longrightarrow \neg C)) \in \mathcal{P}$ et $\psi = (\neg(A \longrightarrow C) \longrightarrow B \in \mathcal{P}$.

1.1.2 Représentation arborescente

On peut considérer que la structure des éléments de $\mathcal{P}$ est arborescente : les feuilles sont les variables propositionnelles et les noeuds internes sont des symboles logiques.

exemple : (suite de l'exemple 1.1)

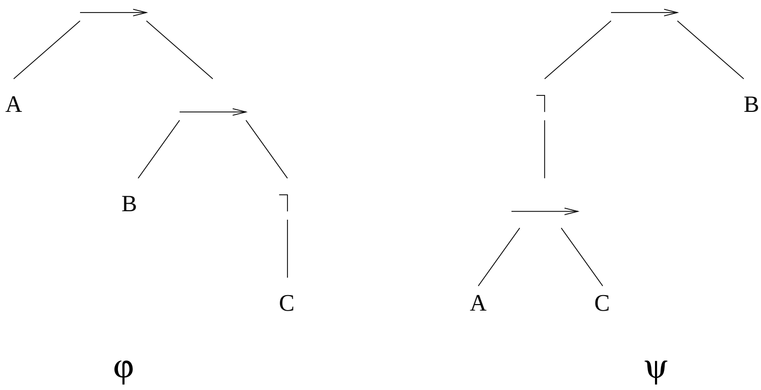

Fig. 1.1 – représentation arborescente de formules propositionnelles

On peut alors parler de la *hauteur* d'une formule propositionnelle comme étant la hauteur de l'arbre la représentant.

exemple : (suite de l'exemple 1.1) $h(\varphi) = h(\psi) = 3$

exemple : $h((A \longrightarrow C)) = 1$ et $h(A) = 0$

On peut aussi définir de façon ascendante l'ensemble $\mathcal{P}$ en construisant les formules de "proche en proche" selon la définition inductive. On définit la suite d'ensembles de formules propositionnelles suivante :

- $\mathcal{P}_0 = \mathcal{V}$
- $\mathcal{P}_{n+1} = \mathcal{P}_n \cup \{\neg\varphi \,;\, \varphi \in \mathcal{P}_n\} \cup \{(\varphi \longrightarrow \psi) \,;\, \varphi, \psi \in \mathcal{P}_n\}$

remarque : on a alors $\mathcal{P}_0 \subset \mathcal{P}_1 \subset \cdots \mathcal{P}_n \cdots$

Théorème 1.1 $\mathcal{P} = \bigcup_{n \in \mathbb{N}} \mathcal{P}_n$

preuve $\mathcal{P}_0 = \mathcal{V} \subset \mathcal{P}$ par définition de $\mathcal{P}$. Supposons que pour un entier $n \geq 0$, on a $\mathcal{P}_n \subset \mathcal{P}$. Soit $\phi \in \mathcal{P}_{n+1}$. On a trois cas

- $\phi \in \mathcal{P}_n$ et donc $\phi \in \mathcal{P}$ par hypothèse
- $\phi = \neg\varphi$ pour une formule $\varphi \in \mathcal{P}_n$; puisque $\varphi \in \mathcal{P}$ par hypothèse, on a, par définition de $\mathcal{P}$, $\varphi \in \mathcal{P}$
- $\phi = \varphi \longrightarrow \psi$ pour deux formules $\varphi, \psi \in \mathcal{P}_n$; puisque $\varphi \in \mathcal{P}$ et $\psi \in \mathcal{P}$ par hypothèse, on a, par définition de $\mathcal{P}$, $\varphi \longrightarrow \psi \in \mathcal{P}$

Donc $\mathcal{P}_{n+1} \subset \mathcal{P}$.

On a bien montré que pour tout entier n, $\mathcal{P}_n \subset \mathcal{P}$ et donc que $\bigcup_{n \in \mathbb{N}} \mathcal{P}_n \subset \mathcal{P}$.

Réciproquement, on va démontrer par induction sur $\mathcal{P}$ que $\mathcal{P} \subset \bigcup_{n\in\mathbb{N}} \mathcal{P}_n$.

Par définition de $\mathcal{P}_0 = \mathcal{V}$, $\mathcal{V} \subset \bigcup_{n\in\mathbb{N}} \mathcal{P}_n$.

Supposons que pour deux formules $\varphi, \varphi' \in \mathcal{P}$ on a $\varphi, \varphi' \in \bigcup_{n\in\mathbb{N}} \mathcal{P}_n$; alors il existe k et k' tels que $\varphi \in \mathcal{P}_k$ et $\varphi' \in \mathcal{P}_{k'}$.

Donc si $K = sup\{k, k'\}$ on a $\varphi \in \mathcal{P}_K$ et $\varphi' \in \mathcal{P}_K$. On en déduit que $\neg\varphi \in \mathcal{P}_{K+1}$ et que $\varphi \longrightarrow \varphi' \in \mathcal{P}_{K+1}$. D'où $\neg\varphi \in \bigcup_{n\in\mathbb{N}} \mathcal{P}_n$ et $\varphi \longrightarrow \varphi' \in \bigcup_{n\in\mathbb{N}} \mathcal{P}_n$. □

1.1.3 Sous-formules

Définition 1.2 *l'ensemble des sous-formules d'une formule propositionnelle φ est l'ensemble des sous-arbres au sens large de φ.*

exemple : (suite de l'exemple 1.1) $Ssf(\varphi) = \{\varphi; (B \longrightarrow \neg C); \neg C; A; B; C\}$

On peut aussi définir l'ensemble des sous-formules d'une formule propositionnelle φ inductivement par

- si $\varphi = A$ où $A \in \mathcal{V}$ alors $Ssf(\varphi) = \{A\}$
- si $\varphi = \neg\psi$ alors $Ssf(\varphi) = \{\neg\psi\} \cup Ssf(\psi)$
- si $\varphi = (\varphi_1 \longrightarrow \varphi_2)$ alors $Ssf(\varphi) = \{(\varphi_1 \longrightarrow \varphi_2)\} \cup Ssf(\varphi_1) \cup Ssf(\varphi_2)$

1.1.4 Substitution

exemple : (suite de l'exemple 1.1) en remplaçant les feuilles de φ par des formules propositionnelles – A par $(B \longrightarrow C)$, B par $\neg C$ et C par $\neg(A \longrightarrow C)$ – on obtient une nouvelle formule propositionnelle.

Définition 1.3 *Soient $A_1, \cdots A_n$ n variables propositionnelles distinctes deux à deux. Soient $\varphi_1, \cdots \varphi_n$ n formules propositionnelles et ψ une formule propositionnelle. On définit $\psi' = \psi_{\varphi_1/A_1, \cdots \varphi_n/A_n}$ par*

- *si $\psi \in \mathcal{V}$ alors $\psi' = \psi$ si $\psi \notin \{A_1, \cdots A_n\}$ et $\psi' = \varphi_k$ si $\psi = A_k$*
- *si $\psi = \neg\phi$ alors $\psi' = \neg\phi_{\varphi_1/A_1, \cdots \varphi_n/A_n}$*
- *si $\psi = (\alpha \longrightarrow \beta)$ alors $\psi' = \alpha_{\varphi_1/A_1, \cdots \varphi_n/A_n} \longrightarrow \beta_{\varphi_1/A_1, \cdots \varphi_n/A_n}$*

Théorème 1.2 *$\psi_{\varphi_1/A_1, \cdots \varphi_n/A_n}$ est une formule propositionnelle.*

On verra, qu'une fois interprétées, ψ et $\psi_{\varphi_1/A_1, \cdots \varphi_n/A_n}$ ont des liens.

1.2 Sémantique

1.2.1 Algèbre de Boole minimale

On rappelle que l'algèbre de Boole minimale, notée $\mathbb{B}$ est l'ensemble $\{0, 1\}$ muni de deux opérations binaires $+$, $\cdot$ décrites par leur table :

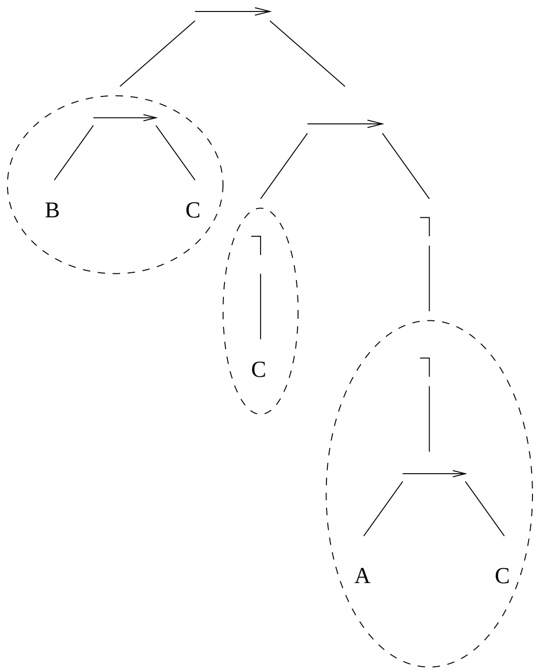

FIG. 1.2 – substitution dans φ

$+$	0	1
0	0	1
1	1	1

$\cdot$	0	1
0	0	0
0	0	1

et d'une opération unaire $\overline{}$ telle que $\overline{0} = 1$ et $\overline{1} = 0$.

Les propriétés de ces opérations sont succinctement décrites par la proposition suivante :

Propriété 1.3 *pour tous $x, y \in \mathbb{B}$, on a*

- $0 + x = x + 0 = x$ *(0 est neutre pour +)*
- $1 + x = x + 1 = 1$ *(1 est absorbant pour +)*
- $1 \cdot x = x \cdot 1 = x$ *(1 est neutre pour ·)*
- $0 \cdot x = x \cdot 0 = 0$ *(0 est absorbant pour ·)*
- $\overline{\overline{x}} = x$, $\overline{x} + x = 1$, $\overline{x} \cdot x = 0$
- $+$ *est commutatif, associatif et distributif par rapport à* $\cdot$
- $\cdot$ *est commutatif, associatif et distributif par rapport à* $+$
- $\overline{x+y} = \overline{x} \cdot \overline{y}$ *et* $\overline{x \cdot y} = \overline{x} + \overline{y}$ *sont les lois de De Morgan.*

Pour éviter l'usage d'un nombre trop important de parenthèses, on admet que $\cdot$ a priorité sur $+$.

1.2.2 Interprétation – valeurs de vérité

Définition 1.4 *une interprétation est une application v de $\mathcal{V}$ dans $\mathbb{B}$ (à chaque variable propositionnelle est associé un élément de $\mathbb{B}$).*

Le résultat suivant permet d'étendre la définition de $v_{\mathcal{V}}$ à $\mathcal{P}$.

Théorème 1.4 *il existe une unique application v de $\mathcal{P}$ dans $\mathbb{B}$ prolongeant $v_{\mathcal{V}}$ telle que*
- $v(A) = v_{\mathcal{V}}(A)$ *pour tout* $A \in \mathcal{V}$,
- $v(T) = 1$ *et* $v(F) = 0$,
- $v(\neg\varphi) = \overline{v(\varphi)}$,
- $v(\varphi \longrightarrow \psi) = \overline{v(\varphi)} + v(\psi)$.

***exemple* 1.2** : (suite de l'exemple 1.1)

$v(\varphi) = \overline{v(A)} + v((B \longrightarrow \neg C) = \overline{v(A)} + \overline{v(B)} + v(\neg C) = \overline{v(A)} + \overline{v(B)} + \overline{v(C)}$

Si $v(A) = 0, v(B) = 1$ et $v(C) = 0$ alors $v(\varphi) = \overline{0} + \overline{1} + \overline{0} = 1 + 0 + 1 = 1$.

Définition 1.5 *une formule φ est dite* satisfiable *s'il existe une interprétation v telle que $v(\varphi) = 1$. On dit alors que v est un modèle de φ.*

***exemple* 1.3** : soit $\alpha = ((A \longrightarrow B) \longrightarrow (A \longrightarrow C))$

$v(\alpha) = \overline{v(A \longrightarrow B)} + v(A \longrightarrow C) = \overline{\overline{v(A)} + v(B)} + \overline{v(A)} + v(C) = v(A) \cdot \overline{v(B)} + \overline{v(A)} + v(C)$.

Si $v(A) = 0, v(B) = 1$ et $v(C) = 1$ alors $v(\alpha) = 1$ donc α est satisfiable.

On dira qu'une formule φ est
- valide si toute interprétation est un modéle de φ (on notera alors $\models \varphi$)
- insatisfiable si aucune interprétation n'est un modéle de φ (on dit aussi que φ est contradictoire)

exemple : (suite de l'exemple 1.3) α n'est pas valide puisque, si $v(A) = 1, v(B) = 1$ et $v(C) = 0$ on a $v(\alpha) = 0$.

1.2.3 De nouveaux connecteurs logiques

Définition 1.6 *on définit la* conjonction *notée $\wedge$ et la* disjonction *notée $\vee$ par*
- $\varphi \wedge \psi = \neg(\varphi \longrightarrow \neg\psi)$
- $\varphi \vee \psi = (\neg\varphi \longrightarrow \psi)$

Propriété 1.5 *pour toute interprétation v et toutes formules propositionnelles φ, ψ*
- $v(\varphi \wedge \psi) = v(\varphi) \cdot v(\psi)$
- $v(\varphi \vee \psi) = v(\varphi) + v(\psi)$

preuve $v(\varphi \wedge \psi) = v(\neg(\varphi \longrightarrow \neg\psi)) = \overline{\overline{v(\varphi)} + v(\neg\psi)} = v(\varphi) \cdot \overline{\overline{v(\psi)}} = v(\varphi) \cdot v(\psi)$ et

$v(\varphi \vee \psi) = v(\neg\varphi \longrightarrow \psi) = \overline{v(\neg\varphi)} + v(\psi) = \overline{\overline{v(\varphi)}} + v(\psi) = v(\varphi) + v(\psi)$. □

remarque : dans $\mathbb{B}$, 0 et 1 représentent aussi les valeurs de vérité $Vrai$ et $Faux$ et les opérations $+$, $\cdot$, $\overline{}$ représentent respectivement les connecteurs logiques $\wedge, \vee, \neg$ *i.e.* le *ou* logique, le *et* logique, et la *négation*.

On peut aussi définir le connecteur $\longleftrightarrow$ par

Définition 1.7 $(\varphi \longleftrightarrow \psi) = (\varphi \longrightarrow \psi) \wedge (\psi \longrightarrow \varphi)$

et on a

Propriété 1.6 *pour toute interprétation* v *et toutes formules propositionnelles* φ, ψ,

$$v(\varphi \longleftrightarrow \psi) = v(\varphi) \cdot v(\psi) + \overline{v(\varphi)} \cdot \overline{v(\psi)}.$$

preuve :

$$\begin{array}{rcl} v(\varphi \longleftrightarrow \psi) & = & v((\varphi \longrightarrow \psi) \wedge (\psi \longrightarrow \varphi)) \\ & = & v((\varphi \longrightarrow \psi)) \cdot v((\psi \longrightarrow \varphi)) \\ & = & (\overline{v(\varphi)} + v(\psi)) \cdot (\overline{v(\psi)} + v(\varphi)) \\ & = & \overline{v(\varphi)} \cdot \overline{v(\psi)} + v(\psi) \cdot \overline{v(\psi)} + \overline{v(\varphi)} \cdot v(\varphi) + v(\varphi) \cdot v(\psi) \\ & = & \overline{v(\varphi)} \cdot \overline{v(\psi)} + v(\varphi) \cdot v(\psi) \end{array}$$

□

Pour éviter trop de parenthèses on admet les priorités suivantes (de la plus élevée à la plus faible) : $\neg > \wedge, \vee > \longrightarrow, \longleftrightarrow$, mais on peut toujours représenter les formules propositionnelles sous forme arborescente.

Il est important de noter pour les connecteurs $\vee, \wedge, \longrightarrow$ que leurs tables de vérité comportent toujours un cas particulier qui correspond à $Faux$ ou $Vrai$ selon le connecteur :

Propriété 1.7 *Soient* φ *et* ψ *deux formules et* v *une interprétation quelconque.*

- $v(\varphi \longrightarrow \psi) = 0$ *si et seulement si* $v(\varphi) = 1$ *et* $v(\psi) = 0$
- $v(\varphi \vee \psi) = 0$ *si et seulement si* $v(\varphi) = 0$ *et* $v(\psi) = 0$
- $v(\varphi \wedge \psi) = 1$ *si et seulement si* $v(\varphi) = 1$ *et* $v(\psi) = 1$

remarque : en particulier, si $v(\varphi) = 0$ alors $v(\varphi \longrightarrow \psi) = 1$ quelque soit la valeur de $v(\psi)$.

1.2.4 Conséquence et équivalence sémantiques

Soit E un ensemble de formules propositionnelles et φ une formule propositionnelle.

Définition 1.8 *On dit* φ *est une* conséquence logique *(ou sémantique) de* E *que l'on note* $E \models \varphi$*, si tout modèle de* E *(c'est-à-dire satisfaisant toutes les formules de* E*) est aussi un modèle de* φ.

Soit $E = \{\varphi_1 \cdots \varphi_n\}$. $E \models \varphi$ *si et seulement si* pour toute interprétation v,

$$v(\varphi_1) = \cdots = v(\varphi_n) = 1 \Rightarrow v(\varphi) = 1.$$

Propriété 1.8 *les propositions suivantes sont équivalentes*

1. $\varphi_1, \cdots, \varphi_n \models \varphi$
2. $\models \varphi_1 \wedge \cdots \wedge \varphi_n \longrightarrow \varphi$
3. $\models \varphi_n \longrightarrow (\varphi_{n-1} \longrightarrow (\cdots(\varphi_1 \longrightarrow \varphi)\cdots))$

preuve : On va montrer (1) $\Leftrightarrow$ (3).

pour $n = 1$ supposons que $\varphi_1 \models \varphi$. Pour une interprétation quelconque v, $v(\varphi_1 \longrightarrow \varphi) = v(\overline{\varphi_1}) + v(\varphi)$. Or, si $v(\varphi_1) = 1$ alors $v(\varphi) = 1$ et donc $v(\varphi_1 \longrightarrow \varphi) = 1$ et si $v(\varphi_1) = 0$ alors $v(\overline{\varphi_1}) = 1$ et $v(\varphi_1 \longrightarrow \varphi) = 1$. Donc on a bien $\models \varphi_1 \longrightarrow \varphi$.

Réciproquement supposons que $\models \varphi_1 \longrightarrow \varphi$. On a donc $v(\varphi_1 \longrightarrow \varphi) = 1$ pour toute interprétation v. En particulier, si $v(\varphi_1) = 1$, puisque $v(\varphi_1 \longrightarrow \varphi) = 1$, on a $v(\varphi_1 \longrightarrow \varphi) = v(\overline{\varphi_1} + v(\varphi) = 0 + v(\varphi) = v(\varphi) = 1$.

Pour $n \geq 2$ quelconque, supposons $\varphi_1, \cdots, \varphi_n \models \varphi$ et soit v une interprétation.

Si v est un modèle de $\varphi_1, \cdots, \varphi_n$ alors on a $v(\varphi_n) = \cdots = v(\varphi_1) = 1$ et donc $v(\varphi) = 1$ par hypothèse d'où $v(\varphi_n \longrightarrow (\cdots(\varphi_1 \longrightarrow \varphi)\cdots)) = 1$.

Sinon soit $k = max\{i = 1 \cdots n | v(\varphi_i) = 0\}$. On a donc $v(\varphi_n) = \cdots = v(\varphi_{k+1}) = 1$ et $v(\varphi_k) = 0$. D'où $v(\varphi_n \longrightarrow (\cdots(\varphi_1 \longrightarrow \varphi)\cdots)) = v(\varphi_k \longrightarrow (\cdots(\varphi_1 \longrightarrow \varphi)\cdots)) = 1$.

Réciproquement supposons que $\models \varphi_n \longrightarrow (\varphi_{n-1} \longrightarrow (\cdots(\varphi_1 \longrightarrow \varphi)\cdots))$ et soit v une interprétation modèle de $\varphi_1, \cdots, \varphi_n$. Alors puisque $v(\varphi_n \longrightarrow (\cdots(\varphi_1 \longrightarrow \varphi)\cdots)) = 1$ et $v(\varphi_n) = \cdots = v(\varphi_1) = 1$ on a nécessairement $v(\varphi) = 1$ et donc $\varphi_1, \cdots, \varphi_n \models \varphi$.

On laissera l'équivalence (1) $\Leftrightarrow$ (2) au lecteur. □

Définition 1.9 *On dira que φ et ψ sont logiquement équivalentes, noté $\varphi \equiv \psi$, si $\varphi \models \psi$ et $\psi \models \varphi$.*

Propriété 1.9 *$\varphi \equiv \psi$ si et seulement si pour toute interprétation v, $v(\varphi) = v(\psi)$.*

preuve : supposons $\varphi \equiv \psi$ et soit v une interprétation quelconque. Si $v(\varphi) = 1$ alors, puisque $\varphi \models \psi$, on a $v(\psi) = 1$. Si $v(\varphi) = 0$ alors on ne peut pas avoir $v(\psi) = 1$ puisque $\psi \models \varphi$, donc $v(\psi) = 0$.

Réciproquement supposons que $v(\varphi) = v(\psi)$ quelque soit l'interprétation v. Soit v une interprétation telle que $v(\varphi) = 1$. Alors $v(\psi) = 1$ donc $\varphi \models \psi$. Maintenant soit v une interprétation telle que $v(\psi) = 1$. Alors $v(\varphi) = 1$ et $\psi \models \varphi$. □

1.2.5 Propriétés de la relation $\equiv$

Propriété 1.10 *$\equiv$ est une relation d'équivalence sur $\mathcal{P}$*

Les trois théorèmes suivants font le lien entre formules et sous-formules et la notion de substitution.

Théorème 1.11 *Soit φ une formule propositionnelle, $\varphi_1, \cdots, \varphi_n$ n formules propositionnelles et $A_1, \cdots, A_n$ n variables propositionnelles distinctes 2 à 2. Si φ est une tautologie alors $\varphi_{\varphi_1/A_1, \cdots \varphi_n/A_n}$ est aussi une tautologie.*

***exemple* 1.4** : Soit $\varphi = A \vee \neg A$. φ est clairement une tautologie donc $A \wedge (B \longrightarrow C) \vee \neg(A \wedge (B \longrightarrow C))$ est aussi une tautologie.

Théorème 1.12 *Soit ψ une sous-formule de φ et soit ψ' une formule équivalente à ψ. Alors la formule obtenue en substituant ψ' à ψ dans φ est logiquement équivalente à φ.*

preuve : on démontrera ce résultat par induction sur la construction des formules propositionnelles après avoir remarqué qu'il suffit de considérer les ces où ψ est une sous-formule propre de φ. Soit v une interprétation quelconque.

- si φ est une variable propositionnelle alors $\psi = \varphi$
- supposons que la propriété soit vraie pour deux formules quelconques ϕ_1 et ϕ_2
  - $\varphi = \neg\phi_1$ alors ψ est une sous-formule de ϕ_1 donc la formule φ' obtenue en substituant ψ' à ψ dans φ est égale à ϕ'_1, formule obtenue en substituant ψ' à ψ dans ϕ_1. Donc $v(\varphi) = \overline{v(\phi_1)} = \overline{v(\phi'_1)} = v(\varphi')$
  - $\varphi = \phi_1 \longrightarrow \phi_2$ alors ψ est une sous-formule de ϕ_1 ou de ϕ_2. Soit ϕ'_i la formule obtenue en substituant ψ' à ψ dans ϕ_i pour $i = 1, 2$. La formule φ' obtenue en substituant ψ' à ψ dans φ est égale à $\phi'_1 \longrightarrow \phi'_2$. Donc $v(\varphi) = \overline{v(\phi_1)} + v(\phi_2) = \overline{v(\phi'_1)} + v(\phi'_2) = v(\phi'_1 \longrightarrow \phi'_2) = v(\varphi')$

□

***exemple* 1.5** : Soit $\varphi = (A \wedge (B \longrightarrow (C \wedge \neg A))) \vee \neg(A \wedge (B \longrightarrow C))$.
La formule $A \wedge (B \longrightarrow (C \wedge \neg A))$ est une sous-formule de φ et est logiquement équivalente à $A \wedge \neg B$ donc $\varphi \equiv (A \wedge \neg B) \vee \neg(A \wedge (B \longrightarrow C))$.

Théorème 1.13 *Soient $\varphi_1, \cdots, \varphi_n$ n formules propositionnelles et $A_1, \cdots, A_n$ n variables propositionnelles distinctes 2 à 2. Soit v une interprétation, on définit l'interprétation v' de la façon suivante*

$$\begin{aligned} v'(A) &= v(A) \textit{ si } A \in \mathcal{V} \textit{ et } A \notin \{A_1, \cdots, A_n\} \\ &= v(\varphi_i) \textit{ si } A = A_i \end{aligned}$$

On a alors $v(\varphi_{\varphi_1/A_1,\cdots\varphi_n/A_n}) = v'(\varphi)$

preuve : on démontrera ce résultat par induction sur la construction des formules propositionnelles. On pose $\varphi' = \varphi_{\varphi_1/A_1,\cdots\varphi_n/A_n}$

- si $\varphi = T$ alors $\varphi' = T$ et $v(\varphi') = v'(\varphi) = 1$.
- si $\varphi = F$ alors $\varphi' = F$ et $v(\varphi') = v'(\varphi) = 0$.
- si $\varphi = A_i$ alors $\varphi' = \varphi_i$ et $v(\varphi') = v(\varphi_i) = v'(A_i) = v'(\varphi)$.
- si $\varphi = A$ et $A \notin \{A_1, \cdots, A_n\}$ alors $\varphi' = \varphi$ et $v(\varphi') = v(A) = v'(A) = v'(\varphi)$.
- supposons que la propriété soit vraie pour deux formules quelconques ϕ et ϕ'.
  - si $\varphi = \neg\phi$ alors $\varphi' = \neg\phi_{\varphi_1/A_1,\cdots\varphi_n/A_n}$. Donc $v(\varphi') = \overline{v(\phi_{\varphi_1/A_1,\cdots\varphi_n/A_n})} = \overline{v'(\phi)}$ par hypothèse sur ϕ. Puisque $v'(\varphi) = \overline{v'(\phi)}$ on a $v(\varphi') = v'(\varphi)$.
  - si $\varphi = \phi \longrightarrow \phi'$ alors $\varphi' = \phi_{\varphi_1/A_1,\cdots\varphi_n/A_n} \longrightarrow \phi'_{\varphi_1/A_1,\cdots\varphi_n/A_n}$ et on a $v(\varphi') = \overline{v(\phi_{\varphi_1/A_1,\cdots\varphi_n/A_n})} + v(\phi'_{\varphi_1/A_1,\cdots\varphi_n/A_n}) = \overline{v'(\phi)} + v'(\phi')$ par hypothèse sur ϕ et ϕ'. On en déduit que $v(\varphi') = v'(\phi \longrightarrow \phi') = v'(\varphi)$.

□

***exemple* 1.6** : on veut calculer la table de vérité de la formule $\varphi = ((A \longrightarrow B) \longrightarrow C) \longrightarrow (A \longrightarrow C)$. Pour toute interprétation v, $v(\varphi) = v'(K \longrightarrow L)$ où $v'(K) = v((A \longrightarrow B) \longrightarrow C)$ et $v'(L) = v(A \longrightarrow C)$. Cela permet d'établir le type de table de vérité suivant :

A	B	C	$(A \longrightarrow B) \longrightarrow C$	$A \longrightarrow C$	φ
0	0	0	0	1	1
0	0	1	1	1	1
0	1	0	0	1	1
0	1	1	1	1	1
1	0	0	1	0	0
1	0	1	1	1	1
1	1	0	0	0	1
1	1	1	1	1	1

1.3 Formes normales

1.3.1 Des tables de vérité aux formules

En établissant une table de vérité on décrit en fait le graphe d'une fonction booléenne c'est-à-dire une fonction de $\mathbb{B}^n$ dans $\mathbb{B}$.

Donc à chauqe formule φ correspond une fonction $f : \mathbb{B}^n \longrightarrow \mathbb{B}$.

On peut alors se demander si la correspondance inverse est vraie et combien y-a-t-il de telles fonctions pour une valeur de n fixée.

On sait que $|\mathbb{B}^n| = 2^n$ donc il y a 2^{2^n} applications de $\mathbb{B}^n$ dans $\mathbb{B}$.

On remarque ensuite que, si $\varphi \equiv \psi$, alors φ et ψ ont la même table de vérité et donc correspondent à la même fonction. On peut donc dire qu'une classe d'équivalence de formules détermine une fonction.

Mais si on choisit une fonction au hasard peut-on trouver une telle classe ou du moins un de ses représentants ?

Théorème 1.14 *Pour toute application $f : \mathbb{B}^n \longrightarrow \mathbb{B}$ il existe au moins une formule dont la table de vérité est le graphe de l'application f.*

preuve : on donnera un exemple ; si l'application f vaut 1 pour les n-uplets suivants et seulement pour ceux là :

$$(0, \cdots, 0, 1), (1, 0, \cdots, 0), (1, 1, 0, \cdots, 0), (0, \cdots, 0, 1, 1)$$

alors une formule φ dont la table de vérité est le graphe de l'application f est $\varphi =$
$(\neg A_1 \wedge \neg A_2 \wedge \cdots \wedge \neg A_{n-1} \wedge A_n) \vee (A_1 \wedge \neg A_2 \wedge \cdots \wedge \neg A_n) \vee (A_1 \wedge A_2 \wedge \neg A_3 \wedge \cdots \wedge \neg A_n) \vee (\neg A_1 \wedge \neg A_2 \wedge \cdots \wedge \neg A_{n-2} \wedge A_{n-1} \wedge A_n)$ □

Parce que cet exemple est généralisable à toute fonction booléenne, cela signifie que toute formule est équivalente à une formule de la même forme que φ. C'est ce qui nous amène à la notion de forme normale.

1.3.2 Formes normales

D'après l'exemple précédent, les formules s'expriment en fonction des variables atomiques ou de leur négation et des connecteurs $\wedge$ et $\vee$.

Définition 1.10 *un* littéral *est une variable atomique ou la négation d'une variable atomique*

Soit n le nombre total de variables atomiques. Soit $\mathcal{V} = \{A_1 \cdots A_n\}$ l'ensemble des variables atomiques. On notera $l_i = A_i$ ou $l_i = \neg A_i$ et $\widetilde{\mathcal{V}}$ l'ensemble des n-uplets $(l_1, \cdots, l_n)$.

Définition 1.11 *une formule φ est sous* forme normale disjonctive canonique *(FNDC) si et seulement si* $\varphi = \bigvee\limits_{(l_1,\cdots,l_n)\in\widetilde{\mathcal{V}}} (l_1 \wedge \cdots \wedge l_n)$

Définition 1.12 *une formule φ est sous* forme normale conjonctive canonique *(FNCC) si et seulement si* $\varphi = \bigwedge\limits_{(l_1,\cdots l_n)\in\widetilde{\mathcal{V}}} (l_1 \vee \cdots \vee l_n)$

remarque : on parle de FND ou de FNC lorsque toutes les variables atomiques ne sont pas nécessairement présentes.

exemple : Soit $\mathcal{V} = \{A, B, C\}$. On définit φ et φ' :
$\varphi = (\neg A \wedge \neg B \wedge C) \vee (\neg A \wedge \neg B \wedge \neg C) \vee (A \wedge B \wedge \neg C)$ est en FNDC.
$\varphi' = (\neg A \wedge \neg B) \vee (A \wedge B \wedge \neg C)$ est en FND.
On a $\varphi \equiv \varphi'$

Toute formule est équivalente à une formule en FNDC ou en FNCC comme on l'a vu dans l'introduction de ce paragraphe. Mais comment déterminer une FNDC ou une FNCC équivalente à une formule donnée ?

***exemple* 1.7** : Soit $\varphi = (A \longrightarrow (((B \wedge \neg A) \vee (\neg C \wedge A)) \longleftrightarrow (A \vee (A \longrightarrow B))))$. On peut établir la table de vérité de φ :

A	B	C	φ
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

φ est satisfait par les interprétations suivantes : $\begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{array}$ et $\neg\varphi$ par : $\begin{array}{ccc} 1 & 0 & 1 \\ 1 & 1 & 1 \end{array}$

On peut alors écrire

$\varphi \equiv (\neg A \wedge \neg B \wedge \neg C) \vee (\neg A \wedge \neg B \wedge C) \vee (\neg A \wedge B \wedge \neg C) \vee (\neg A \wedge B \wedge C) \vee (A \wedge \neg B \wedge \neg C) \vee (A \wedge B \wedge \neg C)$,

et $\neg\varphi \equiv (A \wedge \neg B \wedge C) \vee (A \wedge B \wedge C)$.

On en déduit par les lois de De Morgan que $\varphi \equiv (\neg A \vee B \vee \neg C) \wedge (\neg A \vee \neg B \vee \neg C)$.

On a ainsi les formes FNDC et FNCC de φ.

La méthode employée dans cet exemple est basée sur les remarques suivantes

- si $L = (l_1, \cdots l_n) \in \widetilde{\mathcal{V}}$ alors la formule $l_1 \wedge \cdots \wedge l_n$ est satisfaite par l'interprétation v_L telle que $v_L(A_i) = 1$ si $l_i = A_i$ et $v_L(A_i) = 0$ si $l_i = \neg A_i$, et seulement par cette interprétation.
- si $L_k = (l_{k_1}, \cdots l_{k_n}) \in \widetilde{\mathcal{V}}$ pour $k = 1, \cdots, p$ alors la formule $\bigvee_{1 \leq k \leq p}(l_{k_1} \wedge \cdots \wedge l_{k_n})$ est satisfaite par les interprétations v_{L_k}, $k = 1, \cdots, p$ et seulement par celles-ci.

1.3.3 Système complet de connecteurs

C'est un ensemble de connecteurs logiques qui permet d'engendrer tous les connecteurs de la logique propositionnelle.

exemple : $\{\neg, \longrightarrow\}$ est complet et minimal, ainsi que $\{\neg, \vee\}$ et $\{\neg, \wedge\}$.

$\{\neg, \wedge, \vee\}$ est complet mais pas minimal et $\{\wedge, \vee\}$ n'est pas complet.

Pour les preuves inductives, on peut se contenter de deux étapes en choisissant bien les connecteurs.

1.3.4 Clauses de Horn

Définition 1.13 *une* clause *est une disjonction de littéraux*

Définition 1.14 *une* clause de Horn *est une clause dont au plus un littéral est "positif" c'est-à-dire est une variable atomique.*

***exemple* 1.8** : si $\mathcal{V} = \{A, B, C\}$ alors $A \vee \neg B \vee \neg C$, $\neg A \vee B$ sont des clauses de Horn.

remarque : une clause de Horn (avec un littéral positif) est en fait équivalente à une formule du type $(A_1 \wedge \cdots \wedge A_n) \longrightarrow B$ où $A_1, \cdots, A_n, B$ sont des variables atomiques.

S'il n'y a pas de littéral positif alors la clause de Horn est en fait équivalente à une formule du type $(A_1 \wedge \cdots \wedge A_n) \longrightarrow F$ et s'il n'y a qu'un littéral positif et pas de littéraux négatifs alors la clause de Horn est en fait équivalente à une formule du type $T \longrightarrow B$.

remarque : une clause de Horn peut être vide, elle est alors équivalente à $True$.

On dira qu'une formule est une *formule de Horn* si elle est en FNC et si chaque disjonction contient au plus une variable atomique *i.e.* une formule de Horn est une conjonction de clauses de Horn.

Il existe un algorithme pour tester la satisfiabilité d'une formule de Horn φ.

- s'il existe une sous-formule $T \longrightarrow A$ alors marquer chaque occurrence de A dans φ
- appliquer les règles suivantes jusqu'à ce que l'on ne puisse plus
  - si $(A_1 \wedge \cdots \wedge A_n) \longrightarrow B$ et si $A_1, \cdots, A_n$ sont marqués et B n'est pas marqué alors marquer chaque occurrence de B dans φ
  - si $(A_1 \wedge \cdots \wedge A_n) \longrightarrow F$ et si $A_1, \cdots, A_n$ sont marqués alors **stop** : φ est insatisfiable
- **stop** : φ est satisfiable par l'interprétation v définie par $v(A) = 1$ *si et seulement si* A est marqué dans φ.

Cet algorithme termine en un temps au plus n, où n est le nombre de variables propositionnelles.

***exemple* 1.9** : soit

$$\varphi = (A \vee \neg B) \wedge (\neg A \vee \neg B \vee C) \wedge C.$$

Alors $\varphi \equiv (B \longrightarrow A) \wedge (A \wedge B \longrightarrow C) \wedge (T \longrightarrow C)$ et on marque chaque occurrence de C dans φ

$\varphi \equiv (B \longrightarrow A) \wedge (A \wedge B \longrightarrow \mathbf{C}) \wedge (T \longrightarrow \mathbf{C})$

L'algorithme s'arrête et φ est satisfiable par l'interprétation v avec $v(A) = v(B) = 0, v(C) = 1$

remarque : toute formule n'est pas équivalente à une clause de Horn. Par exemple $A \vee B$.

Chapitre 2

Calcul des prédicats

Le calcul propositionnel permet de modéliser des raisonnements dans lesquels la satisfaisabilité d'une repose simplement sur les valeurs de propositions atomiques.

Mais il ne permet pas de modéliser ce genre d'assertions :

"tous les marins ont le mal de mer" ou encore "si on marche sur la queue d'un chien, il mord" ou bien "il y a des fleurs vertes".

La première assertion sera vérifiée si on peut vérifier que chaque marin du monde entier passé et à venir a le mal de mer.

La seconde assertion sera vérifiée si on peut effectivement marcher sur la queue de tous les chiens et qu'on se fait mordre à chaque fois.

La troisième assertion suppose que l'on trouve parmi les fleurs du monde entier au moins une fleur verte.

On veut donc exprimer des assertions qui vont dépendre d'un certain environnement et le lien avec cet environnement se fera par l'intermédiaire de la notion de *variable* qui sera éventuellement quantifiée par un *quantificateur universel* ou par un *quantificateur existentiel.*

2.1 Syntaxe

La syntaxe du calcul des prédicats est plus complexe que celle du calcul propositionnel. Elle se définit par étapes.

2.1.1 Langage

Un *langage* des prédicats $\mathcal{L}$ est défini par son alphabet constitué par les données suivantes :

- $\neg, \wedge, \vee, \longrightarrow, \longleftrightarrow$ (connecteurs logiques)
- T, F (constantes logiques)
- ( ) , (parenthèses et virgule)
- $\forall$ (quantificateur universel)
- $\exists$ (quantificateur existentiel)
- $\mathcal{X}$ un ensemble dénombrable de symboles de *variables*
- $\mathcal{C}$ un ensemble dénombrable de symboles de *constantes*

– $\mathcal{F}$ un ensemble dénombrable de symboles de *fonctions*, si $f \in \mathcal{F}$ alors l'arité de f sera notée $a(f)$
– $\mathcal{R}$ un ensemble dénombrable de symboles de *relations*, si $R \in \mathcal{R}$ alors l'arité de f sera notée $a(f)$

$\mathcal{C}, \mathcal{F}, \mathcal{R}$ forment la *signature* de $\mathcal{L}$.

Par la suite les lettres $x, y, z, \cdots$ dénoteront des symboles de variables, $a, b, c, \cdots$ dénoteront des symboles de constantes les lettres $f, g, h, \cdots$ dénoteront des symboles de fonctions et les lettres $R, P, Q, \cdots$ dénoteront des symboles de relations.

Pour éviter les parenthèses trop nombreuses, on écrira fxy pour $f(x, y)$ si f est un symbole de fonction d'arité 2.

2.1.2 Termes

L'ensemble des termes définis sur le langage $\mathcal{L}$ est donné récursivement par
– si $x \in \mathcal{X}$ alors x est un terme
– si $c \in \mathcal{C}$ alors c est un terme
– si $t_1, \cdots, t_k$ sont des termes et si $f \in \mathcal{F}$ est d'arité k alors $ft_1 \cdots t_k$ est un terme

2.1.3 Formules atomiques

L'ensemble des formules atomiques est défini par

si $t_1, \cdots, t_n$ sont des termes et si $R \in \mathcal{R}$ est d'arité n alors $Rt_1 \cdots t_n$ est une formule atomique.

2.1.4 Formules

L'ensemble des formules sur le langage $\mathcal{L}$ est défini récursivement par
– toute formule atomique est une formule
– T, F sont des formules
– si φ et ψ sont des formules alors $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \longrightarrow \psi$, $\varphi \longleftrightarrow \psi$ sont des formules.
– si φ est une formule, si x est un symbole de variables alors $\exists x(\varphi)$ et $\forall x(\varphi)$ sont des formules et on dit que φ est la portée du quantificateur.

***exemple* 2.1** : Soit $\mathcal{F} = \{f, g, h\}$ avec $a(f) = a(g) = 2$, $a(h) = 1$. Soit $\mathcal{R} = \{R, P\}$ avec $a(R) = 1$ et $a(P) = 2$. Soit $\mathcal{C} = \{a, b\}$.

$\varphi = Rfxa$ – soit $R(f(x, a))$

$\psi = Phxa$ – soit $P(h(x), a)$

$\phi = \exists xRhx$ – soit $\exists x(R(h(x)))$

$\Psi = \forall y(Rgby \longrightarrow \exists xPxy)$ – soit $\forall y(R(g(b, y)) \longrightarrow \exists xP(x, y))$

sont des formules.

remarque : Attention les deux formules suivantes sont différentes

$\exists xRx \vee Pxx$ – soit $\exists x(R(x)) \vee P(x, x)$

$\exists x(Rx \vee Pxx)$ – soit $\exists x(R(x) \vee P(x,x))$

On peut représenter les formules de façon arborescente ; par exemple la formule Ψ de l'exemple précédent est représenté par l'arbre de la figure 2.1.

Les noeuds internes sont des quantificateurs ou bien des connecteurs logiques et les feuilles sont des formules atomiques.

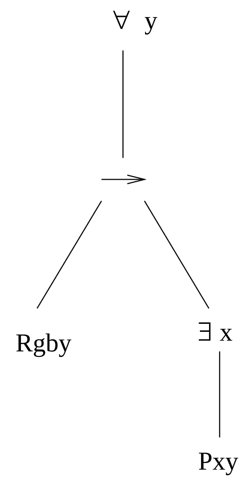

FIG. 2.1 – formule Ψ

On appelle *occurrence* de x dans une formule φ tout noeud étiqueté par x, $\exists x$ ou $\forall x$.

2.1.5 Variables libres ou variables liées

Une occurrence de la variable x est *libre* si aucun de ses ascendants n'est $\exists x$ ou $\forall x$. Les autres occurrences sont *liées*.

***exemple* 2.2** : dans la formule $\exists x Rx \vee Pxx$, x a une occurrence liée et deux occurrences libres.

On peut aussi définir récursivement les ensembles $Vlib(\varphi)$ et $Vlien(\varphi)$ par

– si φ est une formule atomique alors $Vlien(\varphi) = \emptyset$ et $Vlib(\varphi)$ est l'ensemble de toutes les variables ayant au moins une occurrence dans φ.
– si $\varphi = \neg\psi$ alors $Vlien(\varphi) = Vlien(\psi)$ et $Vlib(\varphi) = Vlib(\psi)$
– si $\varphi = \varphi_1$ *connecteur* φ_2 alors $Vlien(\varphi) = Vlien(\varphi_1) \cup Vlien(\varphi_2)$ et
$$Vlib(\varphi) = Vlib(\varphi_1) \cup Vlib(\varphi_2).$$
– si $\varphi = quantificateur\ x\ (\psi)$ alors $Vlien(\varphi) = Vlien(\psi) \cup \{x\}$ et $Vlib(\varphi) = Vlib(\psi) \setminus \{x\}$.

remarque : $Vlib(\varphi)$ et $Vlien(\varphi)$ ne sont pas nécessairement disjoints.

2.2 Sémantique

exemple : la formule $\exists y \forall x Pxy$ est-elle vraie ou fausse ? Tout dépend de la signification donnée au symbole P et du domaine sur lequel cette interprétation sera valable.

Si P est la relation $\leq$ sur l'ensemble $\mathbb{N}$, alors la formule est vraie, on peut trouver un entier naturel – 0 – tel qu'il soit inférieur à tout entier naturel. Mais si P est la relation $\leq$ sur l'ensemble $\mathbb{Z}$, alors la formule est fausse car aucun entier relatif ne peut être inférieur à tout entier relatif.

2.2.1 Structure et interprétation d'un langage

Si $\mathcal{C}, \mathcal{F}, \mathcal{R}$ forment la signature du langage $\mathcal{L}$, une interprétation de $\mathcal{L}$ est réalisée par une structure $\mathcal{U}$ composée de

- un ensemble non vide $\mathcal{D}$ appelé *domaine*
- une application de $\mathcal{C}$ dans $\mathcal{D}$ qui associe à chaque symbole de constante c son interprétation $c^{\mathcal{U}}$
- pour chaque symbole de fonction f de $\mathcal{F}$, une application de $\mathcal{D}^k$ dans $\mathcal{D}$ noté $f^{\mathcal{U}}$ où $a(f) = k$
- pour chaque symbole de relation R de $\mathcal{R}$, une partie de $\mathcal{D}^n$ noté $R^{\mathcal{U}}$ où $a(f) = n$ (cette partie représente les $n-$uplets de $\mathcal{D}^n$ qui sont en relation par $R^{\mathcal{U}}$)

***exemple* 2.3** : Soit $\mathcal{C} = \{c\}$, $\mathcal{F} = \{f, g, h\}$ et $\mathcal{R} = \{R, P\}$ où $a(f) = a(g) = 2$, $a(h) = 1$, $a(R) = a(P) = 2$.

On pose $\mathcal{D} = \mathbb{N}$ et $\mathcal{U} = \mathbb{N}, 0, +, \times, successeur, =, \leq$.

C'est la structure de l'arithmétique de Péano.

2.2.2 Valeur d'une formule

Soit $\mathcal{U}$ une structure interprétant le langage $\mathcal{L}$. On appelle *assignation de variables* toute application d'une partie finie de $\mathcal{X}$ à valeurs dans $\mathcal{D}$ et on notera $x := i$ l'assignation qui à x associe la valeur i de $\mathcal{D}$.

On va tout d'abord évaluer les termes ; soit t un terme alors pour une assignation donnée σ

- si $t = x$ alors $v_{\mathcal{U},\sigma}(t) = \sigma(x)$
- si $t \in \mathcal{C}$ alors $v_{\mathcal{U},\sigma}(t) = c^{\mathcal{U}}$
- si $t = ft_1 \cdots t_k$ alors $v_{\mathcal{U},\sigma}(t) = f^{\mathcal{U}}(v_{\mathcal{U},\sigma}(t_1), \cdots v_{\mathcal{U},\sigma}(t_k))$

Puis on évalue les formules

- si $\varphi = Rt_1 \cdots t_n$ alors $v_{\mathcal{U},\sigma}(\varphi) = 1$ si $(v_{\mathcal{U},\sigma}(t_1), \cdots v_{\mathcal{U},\sigma}(t_n)) \in R^{\mathcal{U}}$ et $v_{\mathcal{U},\sigma}(\varphi) = 0$ sinon
- si $\varphi = T$ alors $v_{\mathcal{U},\sigma}(\varphi) = 1$
- si $\varphi = F$ alors $v_{\mathcal{U},\sigma}(\varphi) = 0$
- si $\varphi = \neg\psi$ alors $v_{\mathcal{U},\sigma}(\varphi) = \overline{v_{\mathcal{U},\sigma}(\psi)}$
- si $\varphi = \varphi_1 \longrightarrow \varphi_2$ alors $v_{\mathcal{U},\sigma}(\varphi) = \overline{v_{\mathcal{U},\sigma}(\varphi_1)} + v_{\mathcal{U},\sigma}(\varphi_2)$
- si $\varphi = \varphi_1 \vee \varphi_2$ alors $v_{\mathcal{U},\sigma}(\varphi) = v_{\mathcal{U},\sigma}(\varphi_1) + v_{\mathcal{U},\sigma}(\varphi_2)$
- si $\varphi = \varphi_1 \wedge \varphi_2$ alors $v_{\mathcal{U},\sigma}(\varphi) = v_{\mathcal{U},\sigma}(\varphi_1) \cdot v_{\mathcal{U},\sigma}(\varphi_2)$
- si $\varphi = \varphi_1 \longleftrightarrow \varphi_2$ alors $v_{\mathcal{U},\sigma}(\varphi) = v_{\mathcal{U},\sigma}(\varphi_1) \cdot v_{\mathcal{U},\sigma}(\varphi_2) + \overline{v_{\mathcal{U},\sigma}(\varphi_1)} \cdot \overline{v_{\mathcal{U},\sigma}(\varphi_2)}$
- si $\varphi = \forall x(\psi)$ alors $v_{\mathcal{U},\sigma}(\varphi) = 1$ *si et seulement si* pour tout élément i de $\mathcal{D}$, on a $v_{\mathcal{U},\sigma'}(\psi) = 1$ où $\sigma' = \sigma[x := i]$ (*i.e.* σ composée avec l'assignation $x := i$)
- si $\varphi = \exists x(\psi)$ alors $v_{\mathcal{U},\sigma}(\varphi) = 1$ *si et seulement si* il existe un élément i de $\mathcal{D}$ vérifiant $v_{\mathcal{U},\sigma'}(\psi) = 1$ où $\sigma' = \sigma[x := i]$.

***exemple* 2.4** : soit φ la formule $\forall y(Rxy)$. On donne $\mathcal{U}$ par $\mathcal{D} = \mathbb{N}$ et $\mathbb{R}^{\mathcal{U}} =\leq$ et on pose $x := 0$ et $y := 3$ pour l'assignation σ. Alors $v_{\mathcal{U},\sigma}(\varphi) = 1$. Mais pour σ' défini par $x := 2$ alors $v_{\mathcal{U},\sigma'}(\varphi) = 0$.

soit ϕ la formule $\forall x \forall y(Rxy)$. On donne $\mathcal{U}$ par $\mathcal{D} = \mathbb{N}$ et $\mathbb{R}^{\mathcal{U}} =\leq$ et on pose $x := 0$ et $y := 3$ pour l'assignation σ. Alors $v_{\mathcal{U},\sigma}(\phi) = 0$. On remarque ici que quelle que soit l'assignation, on a $v_{\mathcal{U},\sigma}(\phi) = 0$

Propriété 2.1 *Soit φ une formule telle que $Vlib(\varphi) = \{x_1, \cdots, x_p\}$. Soit $\mathcal{U}$ une structure et σ, σ' deux assignations* **coïncidant** *sur $Vlib(\varphi)$. Alors $v_{\mathcal{U},\sigma}(\varphi) = v_{\mathcal{U},\sigma'}(\varphi)$. Cela signifie que seules les valeurs attribuées aux variables libres par une assignation sont pertinentes dans l'évaluation de φ par rapport à cette assignation.*

Corollaire 2.2 *Si $Vlib(\varphi) = \emptyset$ (on dit que φ est une formule close) alors $v_{\mathcal{U},\sigma}(\varphi) = v_{\mathcal{U},\sigma'}(\varphi)$ pour toutes assignations σ, σ'. On écrira donc $v_{\mathcal{U},\sigma}(\varphi) = v_{\mathcal{U}}(\varphi)$ pour une formule close φ.*

Définition 2.1 *On dit que $\mathcal{U}$ est un* modèle *d'une formule φ si pour toute assignation σ, $v_{\mathcal{U},\sigma}(\varphi) = 1$. On notera $\mathcal{U} \models \varphi$.*

remarque : si la formule φ n'est pas close alors pour une même structure, φ peut être évalué à vrai ou à faux selon l'assignation choisie.

Définition 2.2 *On dit qu' une formule φ est* valide *si pour toute structure $\mathcal{U}$, $\mathcal{U} \models \varphi$. On notera $\models \varphi$.*

Définition 2.3 *On dit qu' une formule φ est* satisfiable *s'il existe une structure $\mathcal{U}$ telle que $\mathcal{U} \models \varphi$. Sinon φ est insatisfiable.*

Définition 2.4 (conséquence logique) *Soit $\mathcal{L}$ un langage. Soit $E = \{\varphi_1, \cdots, v_n\}$ un ensemble de formules et soit ψ une formule. On dit que ψ est une* conséquence logique *de E si pour toute structure $\mathcal{U}$ et toute assignation σ on a l'implication*

$$v_{\mathcal{U},\sigma}(\varphi_1 \wedge \cdots \wedge \varphi_n) = 1 \Rightarrow v_{\mathcal{U},\sigma}(\psi) = 1$$

On notera $E \models \psi$.

Si $\varphi \models \psi$ et $\psi \models \varphi$ alors on dira que φ et ψ sont logiquement équivalentes et on notera $\varphi \equiv \psi$.

On se reportera à l'annexe B pour une suite d'équivalences utiles.

Annexe A : Formulaire de logique propositionnelle

$\varphi \wedge \varphi \equiv \varphi$ (idempotence)
$\varphi \vee \varphi \equiv \varphi$

$\varphi \wedge \psi \equiv \psi \wedge \varphi$ (commutativité)
$\varphi \vee \psi \equiv \psi \vee \varphi$

$(\varphi \wedge \psi) \wedge \phi \equiv \varphi \wedge (\psi \wedge \phi)$ (associativité)
$(\varphi \vee \psi) \vee \phi \equiv \varphi \vee (\psi \vee \phi)$

$\varphi \vee (\psi \wedge \phi) \equiv (\varphi \vee \psi) \wedge (\varphi \vee \phi)$ (distributivité)
$\varphi \wedge (\psi \vee \phi) \equiv (\varphi \wedge \psi) \vee (\varphi \wedge \phi)$

$\varphi \wedge (\varphi \vee \psi) \equiv \varphi$ (absorption)
$\varphi \vee (\varphi \wedge \psi) \equiv \varphi$

$\neg(\varphi \vee \psi) \equiv (\neg\varphi \wedge \neg\psi)$ (lois de de Morgan)
$\neg(\varphi \wedge \psi) \equiv (\neg\varphi \vee \neg\psi)$

$\varphi \wedge \neg\varphi \equiv F$
$\varphi \vee \neg\varphi \equiv T$

$\varphi \wedge F \equiv F$ $\qquad$ $\varphi \wedge T \equiv \varphi$
$\varphi \vee T \equiv T$ $\qquad$ $\varphi \vee F \equiv \varphi$

$\neg T \equiv F$ $\qquad$ $\neg F \equiv T$

$\varphi \longrightarrow \psi \equiv \neg\varphi \vee \psi$
$\varphi \vee \psi \equiv \neg\varphi \longrightarrow \psi$
$\varphi \wedge \psi \equiv \neg(\varphi \longrightarrow \neg\psi)$

$\varphi \longleftrightarrow \psi \equiv (\varphi \longrightarrow \psi) \wedge (\psi \longrightarrow \varphi)$
$\varphi \longleftrightarrow \psi \equiv (\varphi \wedge \psi) \vee (\neg\varphi \wedge \neg\psi)$

$\varphi \longrightarrow (\psi_1 \wedge \psi_2) \equiv (\varphi \longrightarrow \psi_1) \wedge (\varphi \longrightarrow \psi_2)$
$\varphi \longrightarrow (\psi_1 \vee \psi_2) \equiv (\varphi \longrightarrow \psi_1) \vee (\varphi \longrightarrow \psi_2)$

$(\psi_1 \wedge \psi_2) \longrightarrow \varphi \equiv (\psi_1 \longrightarrow \varphi) \vee (\psi_2 \longrightarrow \varphi)$
$(\psi_1 \vee \psi_2) \longrightarrow \varphi \equiv (\psi_1 \longrightarrow \varphi) \wedge (\psi_2 \longrightarrow \varphi)$

$(\varphi \longleftrightarrow \psi) \longleftrightarrow \phi \equiv \varphi \longleftrightarrow (\psi \longleftrightarrow \phi)$

Annexe B : Formulaire de logique du premier ordre

$\forall x \varphi \equiv \neg \exists x \neg \varphi$
$\neg \forall x \varphi \equiv \exists x \neg \varphi$
$\exists x \varphi \equiv \neg \forall x \neg \varphi$
$\neg \exists x \varphi \equiv \forall x \neg \varphi$

$\forall x(\varphi \wedge \psi) \equiv \forall x \varphi \wedge \forall x \psi$
$\exists x(\varphi \vee \psi) \equiv \exists x \varphi \vee \exists x \psi$
$\exists x(\varphi \longrightarrow \psi) \equiv \forall x \varphi \longrightarrow \exists x \psi$

$\forall x \forall y \varphi \equiv \forall y \forall x \varphi$
$\exists x \exists y \varphi \equiv \exists y \exists x \varphi$

On a aussi les équivalences suivantes suivantes si x n'est pas libre dans ψ

$\forall x \psi \equiv \exists x \psi \equiv \psi$
$\forall x(\varphi \wedge \psi) \equiv \forall x \varphi \wedge \psi$
$\exists x(\varphi \vee \psi) \equiv \exists x \varphi \vee \psi$
$\forall x(\varphi \vee \psi) \equiv \forall x \varphi \vee \psi$
$\exists x(\varphi \wedge \psi) \equiv \exists x \varphi \wedge \psi$
$\exists x(\psi \longrightarrow \varphi) \equiv \psi \longrightarrow \exists x \varphi$
$\forall x(\psi \longrightarrow \varphi) \equiv \psi \longrightarrow \forall x \varphi$
$\exists x(\varphi \longrightarrow \psi) \equiv \forall x \varphi \longrightarrow \psi$
$\forall x(\varphi \longrightarrow \psi) \equiv \exists x \varphi \longrightarrow \psi$

Les trois formules suivantes sont universellement valides

$\exists x(\varphi \wedge \psi) \longrightarrow (\exists x \varphi \wedge \exists x \psi)$
$\forall x \varphi \vee \forall x \psi \longrightarrow \forall x(\varphi \vee \psi)$
$\exists x \forall y \varphi \longrightarrow \forall y \exists x \varphi$

Principe de renommage : si y n'est pas une variable libre de φ

$\forall x \varphi \equiv \forall y \varphi_{x \leftarrow y}$
$\exists x \varphi \equiv \exists y \varphi_{x \leftarrow y}$