Support de Cours
Logique Mathématique

Enseignant Responsable : Hatem HADJ KACEM

Département d'Informatique de Gestion

FSEG, Université de Sfax, Tunisie

Février 2018

Table des matières

1 Logique Propositionnelle **3**
1.1 Introduction . . . 3
1.2 Syntaxe du langage . . . 4
1.2.1 Alphabet ou vocabulaire . . . 4
1.2.2 Langage . . . 5
1.2.3 Règles de suppression des parenthèses . . . 6
1.3 Sémantique des propositions . . . 6
1.3.1 Interprétation classique de la logique des propositions . 6
1.3.2 Formules équivalentes en logique des propositions . . . 8
1.4 Représentation des connaissances . . . 10
1.5 Systèmes de preuve en logique des propositions . . . 11
1.5.1 Preuve par les tables de vérité . . . 11
1.5.2 Preuve par les lois d'équivalence . . . 13
1.5.3 Forme Normale . . . 13

2 Exercices **17**

3 Logique des prédicats du Premier Ordre **26**
3.1 Introduction . . . 26
3.2 Syntaxe du langage . . . 27
3.2.1 Alphabet . . . 27
3.2.2 Termes . . . 28
3.2.3 Atomes (ou formules atomiques) . . . 29
3.2.4 Formules bien formées (fbf) . . . 29
3.2.5 Variable libre et variable liée . . . 30
3.3 Règles d'inférence . . . 31
3.4 Sémantique de la logique des prédicats . . . 32
3.4.1 Interprétations . . . 32
3.4.2 Evaluation d'une fbf dans une interprétation I . . . 33
3.4.3 Validité (et invalidité), inconsistance (et consistance) d'une fbf . . . 34

3.4.4 Formules équivalentes 34

4 Exercices **36**

5 Principe de résolution **43**
5.1 Les clauses 43
5.2 Forme Normale Prenex d'une fbf 43
5.3 Passage sous forme clausale 44
5.4 Le principe de résolution appliqué à des clauses concrètes . . . 46
5.5 L'unification 47
5.5.1 Substitutions 47
5.5.2 Unifieurs 47
5.5.3 Algorithme d'unification 48

6 Exercices **50**

Chapitre 1

Logique Propositionnelle

La logique est définie traditionnellement comme la science qui étudie la validité des propositions et des démonstrations par des procédés qui analysent la structure formelle des objets et non leur contenu (signification). C'est une théorie de l'inférence formelle du raisonnement, c'est-à-dire des relations d'implication entre un ensemble d'hypothèses et de conclusions indépendamment de la véracité de telle ou telle hypothèse.
Nous étudions ici la logique mathématique en tant qu'informaticien, c'est-à-dire que nous l'utilisons comme formalisme (système) de traitement de l'information.
Trois notions fondamentales en logique :

- ***Langage*** *: construction et formation des formules logiques à partir d'un alphabet de symboles ; construction des formules.*
- ***Vérité*** *: sémantique et signification des formules ; signification des formules.*
- ***Preuve*** *: enchaînement des formules en preuves et démonstration ; démonstration des formules.*

1.1 Introduction

Une proposition est un énoncé déclaratif. On traite ici les connexions ou relations entre propositions.

Exemple 1

- *0 est l'élément neutre de l'addition.*
- *S'il fait beau alors je me promènerai.*
- $3 > 5$.
- $x < Succ(x)$ *pour tout* x *dans* N.

- $5 > 3$ *et* $24 < 17$.
- *Il fait soleil*

Chacun de ces énoncés est une proposition. Les propositions sont vraies ou fausses. Certaines sont formées à partir de propositions élémentaires reliées par des opérateurs ou connecteurs, par exemple : *si ... alors .... et ....*
La forme de l'énoncé est prise en compte ainsi que la sémantique.

- Il fait soleil.
- Il ne fait pas gris.
- $X >= 3$.
- $3 <= X$.

Parmi les énoncés précédents certains sont vrais, d'autres faux. "Vrai (V) ou (T)" et "Faux (F)" sont appelés **valeurs de vérité**. Un énoncé qui n'est pas vrai est faux et réciproquement.

Exemple 2

— *"7 est un nombre impair" est un énoncé vrai.*
— *"2+2 = 5" est un énoncé faux.*

Proposition

Une proposition est un énoncé qu'on peut juger sans ambiguïté Vrai ou Faux.

Les propositions sont généralement notées par des lettres majuscules.

Assertion

Une assertion est une proposition à laquelle on a attribué une valeur de vérité.

1.2 Syntaxe du langage

Pour étudier la syntaxe d'un langage, il faut donner un alphabet (un ensemble de symboles) et des règles de constructions syntaxiques d'expressions à partir de ces symboles.

1.2.1 Alphabet ou vocabulaire

C'est un ensemble dénoté par VT et composé des sous-ensembles suivants :

- Ensemble dénombrable d'objets appelés lettres propositionnelles (ou variables propositionnelles ou atomes) dénotées par des lettres latines majuscules éventuellement indicées $\{A, B, ..., A_1, B_2, ...\}$.
- Ensemble de connecteurs $\{\neg, \vee, \wedge, \rightarrow, \leftrightarrow, \}$ avec :
  $\neg$: connecteur unaire de négation,
  $\wedge$: connecteur binaire de conjonction,
  $\vee$: connecteur binaire de disjonction,
  $\rightarrow$: connecteur binaire d'implication et
  $\leftrightarrow$: connecteur binaire d'équivalence.
- Ensemble de délimiteurs $\{(,)\}$.

D'où l'alphabet du langage VT est :

$$VT = \{A, B, ..., A_1, B_2, ...\} \cup \{\neg, \wedge, \vee, \rightarrow, \leftrightarrow\} \cup \{(,)\}$$

Un connecteur binaire est une manière de composer deux propositions pour en obtenir une troisième. Un connecteur unaire est une manière d'obtenir une autre proposition à partir d'une proposition donnée.

1.2.2 Langage

Le langage est constitué de l'ensemble des Formules Bien Formées (appelées aussi *fbf* ou well formed formula *wff*) ou expressions bien formées défini comme suit :

Définition 1 *Les fbf sont définies inductivement comme suit :*
- **Base :** *Tout atome est une fbf.*
- **Induction :** *Si A et B sont des fbf alors $(\neg A)$, $(A \wedge B)$, $(A \vee B)$, $(A \rightarrow B)$ et $(A \leftrightarrow B)$ sont des fbf.*
- **Clôture :** *Toute fbf est obtenue par un nombre fini d'applications des règles ci-dessus.*

On note par $L = \{fbf\}$.

Définition 2 *Soient S l'axiome de la grammaire, $VN = \{S, SA\}$ l'ensemble des symboles non terminaux et $V = VT \cup VN$.*

$$G : \begin{cases} S \rightarrow SA|(\neg S)|(S \wedge S)|(S \vee S)|(S \rightarrow S)|(S \leftrightarrow S) \\ SA \rightarrow A|B|C|\ldots|A_1|B_1|\ldots \end{cases}$$

Nous pouvons montrer aisément que $L = L(G)$.

Exemple 3
$((A \wedge (\neg B)) \rightarrow (C \vee D)) \in L$
$(A \wedge B)(\neg \vee A) \notin L$

1.2.3 Règles de suppression des parenthèses

Les règles de suppression des parenthèses d'une formule bien formée fbf sont les suivantes :

1. Priorité décroissante des connecteurs dans l'ordre : $\{\neg, \wedge, \vee, \rightarrow, \leftrightarrow\}$.
2. On omet les parenthèses les plus externes.
3. Quand il y a un seul connecteur, l'association se fait de gauche à droite.

Exemple 4

$A \vee \neg B \wedge \neg C$	*s'écrit*
$A \vee B$	*s'écrit*
$A \rightarrow B \rightarrow C \rightarrow D$	*s'écrit*

1.3 Sémantique des propositions

La sémantique attribue une signification aux expressions. Elle est compositionnelle : la signification d'une formule est fonction de celle de ses constituants.

A chaque proposition, c'est-à-dire à chaque fbf, on veut attribuer une valeur sémantique. La définition des fbf est faite de façon inductive à partir des atomes et des connecteurs. Il faut donc donner à tout atome d'une fbf une valeur dans un domaine D à préciser et donner un sens aux connecteurs dans ce domaine.

Définition 3 *Une interprétation du calcul des propositions consiste en la donnée d'un domaine D, la définition des connecteurs par des applications de D dans D pour $\neg$, et de $D \times D$ dans D pour $\wedge, \vee, \rightarrow, \leftrightarrow$ et une valuation v des atomes dans D.*

1.3.1 Interprétation classique de la logique des propositions

D est un domaine binaire : $\{T, F\}$, noté aussi $\{Vrai, Faux\}$ ou $\{0, 1\}$, c'est-à-dire, que tout atome est T ou F.

La définition des connecteurs est décrite par des tables de vérité :

A	$\neg A$
T	F
F	T

A	B	$A \wedge B$	$A \vee B$	$A \to B$	$A \leftrightarrow B$
T	T	T	T	T	T
T	F	F	T	F	F
F	T	F	T	T	F
F	F	F	F	T	T

Définition 4 *Pour une fbf G dont les atomes distincts sont $A_1, A_2, ..., A_n$, une interprétation de la fbf est une assignation v des valeurs de vérité T ou F à chacun des A_i pour $i \in \{1, 2, ..., n\}$. Si G comporte n atomes, il y a 2^n interprétations. Pour les examiner de façon exhaustive, on utilise des tables de vérité.*

Remarque 1 *Dans une interprétation d'une fbf A, on note $A = T$ (resp F), l'assignation de A à T (resp F) au lieu de $v(A) = T$ (resp F).*

Définition 5 *la fbf G est Vrai dans une interprétation si sa valeur est T. La fbf G est Faux dans une interprétation si sa valeur est F.*

Exemple 5 *G_1 : $P \to (Q \vee \neg R)$. G_1 est F dans $I_1(P = T, Q = F, R = T)$ et T dans $I_2(P = F, Q = T, R = F)$.*

P	Q	R	$\neg R$	$Q \vee \neg R$	$P \to (Q \vee \neg R)$
T	F	T	F	F	F
F	T	F	?	?	T

$G_2 = A \vee (\neg A \wedge B)$

A	B	$\neg A$	$\neg A \wedge B$	G_2
T	T	F	F	T
T	F	F	F	T
F	T	T	T	T
F	F	T	F	F

G_2 est T dans $I_1(A = T, B = T)$, $I_2(A = T, B = F)$ et $I_3(A = F, B = T)$ et F dans $I_4(A = F, B = F)$.

Définition 6 *Deux fbf G_1 et G_2 sont équivalentes si et seulement si les valeurs de vérité de G_1 et de G_2 sont les mêmes dans toute interprétation. On notera $G_1 \equiv G_2$.*

1.3.2 Formules équivalentes en logique des propositions

Soient A, B et C des fbf.

Implication

$A \to B \equiv \neg A \vee B$

Équivalence

$A \leftrightarrow B \equiv (A \to B) \wedge (B \to A)$

Commutativité

a) $A \vee B \equiv B \vee A$
b) $A \wedge B \equiv B \wedge A$

Associativité

a) $(A \vee B) \vee C \equiv A \vee (B \vee C)$
b) $(A \wedge B) \wedge C \equiv A \wedge (B \wedge C)$

Distributivité

a) $A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C)$
b) $A \wedge (B \vee C) \equiv (A \wedge B) \vee (A \wedge C)$

Complémentarité

a) $A \vee \neg A = T$
b) $A \wedge \neg A = F$

De Morgan

a) $\neg(A \vee B) \equiv \neg A \wedge \neg B$
b) $\neg(A \wedge B) \equiv \neg A \vee \neg B$
c) $A \vee (\neg A \wedge B) \equiv A \vee B$
d) $A \wedge (\neg A \vee B) \equiv A \wedge B$
e) $A \vee (A \wedge B) \equiv A$
f) $A \wedge (A \vee B) \equiv A$

Involution

a) $\neg\neg A \equiv A$

Identité

a) $(A \vee A) \equiv A$
b) $(A \wedge A) \equiv A$

Exemple 6 *G_2 se simplifie en $A \vee B$ selon le tableau précédent.*

Définition 7

- *Une fbf est valide (tautologie) si et seulement si elle est vraie dans toute interprétation.*
- *Une fbf est invalide si et seulement si elle n'est pas valide.*
- *Une fbf est inconsistante (contradiction) si et seulement si elle est Fausse dans toute interprétation.*
- *Une fbf est consistante si et seulement si elle n'est pas inconsistante.*

Remarque 2

1. *Une fbf peut être à la fois invalide et consistante.*
2. *G est valide si et seulement si $\neg G$ est inconsistante.*
3. *Si G est valide alors G est consistante ; Si G est inconsistante alors G est invalide.*
4. *Il existe une procédure effective pour déterminer si une fbf est valide : la table de vérité.*

Définition 8 *Si G est T dans l'interprétation I on dit que I satisfait G et que I est un modèle de G.*

Définition 9 *Soient F_1, F_2, ..., F_n et G des $fbfs$. On dit que G est une conséquence logique des F_i pour $i \in \{1, 2, ..., n\}$ (ou G découle logiquement de F_i) si et seulement si toute interprétation qui satisfait toutes les F_i satisfait aussi G.*

Théorème 1 *Soient F_1, F_2, ..., F_n et G des $fbfs$, G est une conséquence logique des F_i pour $i \in \{1, 2, ..., n\}$ si et seulement si $F_1 \wedge F_2 \wedge ... \wedge F_n \rightarrow G$ est valide.*

Corollaire 1 *G est une conséquence logique des F_i pour $i \in \{1, 2, ..., n\}$ si et seulement si $F_1 \wedge F_2 \wedge ... \wedge F_n \wedge \neg G$ est inonsistante.*

1.4 Représentation des connaissances

La représentation des connaissances en Intelligence Artificielle consiste à faire une correspondance entre le monde extérieur et un système symbolique manipulable par un ordinateur. La représentation des connaissances comporte un aspect passif : il faut mémoriser mais aussi un aspect actif : il faut inférer, manipuler ces connaissances, effectuer un raisonnement.

Le sens des connecteurs n'est pas exactement le même que celui exprimé par le langage naturel. La capacité d'expression en logique propositionnelle est beaucoup moins riche qu'en langage naturel. Néanmoins les connecteurs logiques ont des correspondances dans la langue naturelle.

Exemple :

$A \wedge B$ peut se traduire par :
- A et B
- B et A
- à la fois A et B
- Non seulement A mais B
- ...

$A \vee B$ peut se traduire par :
- A ou B
- ou B ou A
- Soit A soit B
- A ou B ou les deux à la fois
- ...

$A \rightarrow B$ peut se traduire par :
- Si A alors B
- B alors A
- B si A
- B lorsque A
- ...

$A \leftrightarrow B$ peut se traduire par :
- A si et seulement si B
- A si B et B si A
- ...

1.5 Systèmes de preuve en logique des propositions

La logique consiste à examiner la validité des raisonnements. Un raisonnement (ou inférence) est le passage d'une proposition ou d'un groupe de propositions (prémisses) vers une autre proposition (conclusion).

1.5.1 Preuve par les tables de vérité

La preuve à l'aide des tables de vérité permet de manière mécanique de dire si une formule est valide, consistante, inconsistante ou invalide. Pour cela il faut décomposer la formule en sous-formules en procédant de l'intérieur vers l'extérieur. Ensuite associer à chaque atome présent dans la formule une colonne, une colonne par sous-formule et une pour la formule.

Exemple 7

Il est à noter qu'il n'est pas nécessaire de calculer la valeur de vérité pour toutes les interprétations. Tant qu'on obtient que des vrais (respectivement que des faux), il faut calculer l'interprétation suivante, mais dès qu'on obtient un faux (respectivement un vrai) on s'arrête. La valeur logique vrai suffit pour dire qu'une formule est consistante et la valeur logique faux suffit pour dire qu'elle est invalide.

1.5.2 Preuve par les lois d'équivalence

Chacun des théorèmes d'équivalence permet un acte d'inférence. Une autre façon pour prouver qu'une formule est valide consiste à raisonner en utilisant les lois d'équivalence. Il suffit de montrer par équivalence que la formule de départ est équivalente à vrai.

Exemple 8

1.5.3 Forme Normale

Les formes normales d'une formule bien formée permettent d'écrire la formule de départ sous une forme donnée.

Définition 10

- *Un littéral est un atome ou la négation d'un atome.*
- *Une clause est une disjonction de littéraux.*
- *Un ensemble de clauses est une conjonction de clauses.*

Définition 11

- *Une fbf est mise sous forme normale conjonctive (fnc) si et seulement si elle est de la forme $F_1 \wedge F_2 \wedge ... \wedge F_n$ où chaque F_i pour $i \in \{1, 2, ..., n\}$ est une disjonction de littéraux (de la forme $A_1 \vee A_2 \vee ... \vee A_m$). Une fnc est une conjonction de disjonction de littéraux.*
- *Une fbf est mise sous forme normale disjonctive (fnd) si et seulement si elle est de la forme $F_1 \vee F_2 \vee ... \vee F_n$ où chaque F_i pour $i \in \{1, 2, ..., n\}$ est une conjonction de littéraux (de la forme $A_1 \wedge A_2 \wedge ... \wedge A_m$). Une fnd est une disjonction de conjonction de littéraux.*

Exemple 9

Remarque 3 *Une fnc $F_1 \wedge F_2 \wedge ... \wedge F_n$ peut être représentée (ou notée) par l'ensemble de clauses $S = \{F_1, F_2, ..., F_n\}$. On dira que S est insatisfiable ou inconsistant si et seulement si S est faux pour toute interprétation.*

Exemple 10
La formule $(A \vee B) \wedge (\neg C \vee \neg B) \wedge D$ est représentée par $S = \{A \vee B, \neg C \vee \neg B, D\}$.
La formule $A \vee \neg B \vee \neg C$ est représentée par $S = \{A \vee \neg B \vee \neg C\}$

a) Recherche d'une FNC à partir d'une fbf

Pour mettre une fbf sous forme normale conjonctive il faut :
- Éliminer les connecteurs $\rightarrow$ et $\leftrightarrow$.
- Développer le $\neg$ en utilisant les lois de Morgan et éliminer $\neg\neg$
- Regrouper les $\vee$ par distributivité.

Exemple 11

b) Principe de résolution

On sait qu'une fbf G est valide si et seulement si $\neg G$ est inconsistante. Etant donné une fbf G, on va donc chercher à résoudre le problème "$\neg G$ est-elle inconsistante ?". Pour déterminer efficacement l'inconsistance de la fbf $\neg G$ on peut utiliser une méthode basée sur le principe de résolution. Pour cela, nous allons travailler sur la forme normale conjonctive de la fbf $\neg G$ représentée par l'ensemble de clauses S.

Définition 12 *Soient deux clauses C_1 et C_2 en logique des propositions. Si un littéral L_1 de C_1 est complémentaire d'un littéral L_2 $(= \neg L_1)$ de C_2, on construit C, la disjonction des clauses restantes après suppression des littéraux L_1 et L_2 ; C est appelée clause résolvante ou résolvant de C_1 et C_2. L_1 et L_2 sont les littéraux résolus. Si $C_1 = L_1 \vee C_1'$ et $C_2 = L_2 \vee C_2'$ où C_1' et C_2' sont des clauses et $L_2 = \neg L_1$ alors $C = C_1' \vee C_2'$.*

Exemple 12
$C_1 = \neg q \vee p$; $C_2 = r \vee \neg p \vee s$ *alors*
$C_1 = \neg q \vee p$; $C_2 = q$ *alors*
$C_1 = \neg p \vee \neg q \vee r$; $C_2 = p \vee s \vee \neg r$ *alors*
$C_1 = p \vee q \vee r$; $C_2 = p \vee r$ *alors*
$C_1 = \neg p$; $C_2 = p$ *alors* $C = \square$. *En effet,* $C_1 = \neg p \vee \square$ *et* $C_2 = p \vee \square$ *d'ou* $C = \square \vee \square = \square$

$\square$ est appelée clause vide.

Théorème 2 *Le résolvant C de deux clauses C_1 et C_2 est une conséquence logique de C_1 et C_2.*

Remarque 4 *Le principe de résolution est une généralisation de deux règles d'inférence (ou modes de raisonnement) bien connues.*

1. *Modus Ponens : Si A et $A \rightarrow B$ alors B. Or $C_1 = A$ et $C_2 = A \rightarrow B \equiv \neg A \vee B$ ont pour résolvant B.*
2. *Enchaînement : Si $A \rightarrow B$ et $B \rightarrow D$ alors $A \rightarrow D$. Or $C_1 = \neg A \vee B$ et $C_2 = \neg B \vee D$ ont pour résolvant $\neg A \vee D \equiv A \rightarrow D$.*

Le principe de résolution est une règle d'inférence; Elle produit une clause qui est la conséquence logique des clauses dont elle est issue.

Définition 13 *Soit S un ensemble de clauses. Une réduction (ou résolution) d'une clause C à partir de S est une séquence finie $R_1, R_2, ..., R_n$ de clauses tels que $R_n = C$ et chaque R_i pour $i \in \{1, 2, ..., n-1\}$ est soit une clause de S soit un résolvant de clauses le précédant.*

Exemple 13
Soit $S = \{r \vee q, \neg r, \neg q \vee p, \neg p \vee r\}$ *un ensemble de clauses.*
$R_1 = r \vee q \in S$
$R_2 = \neg r \in S$
$R_3 = q$ *est le résolvant de* R_1 *et* R_2. $R_3 = q$ *est une déduction à partir de* S.
$R_4 = \neg q \vee p \in S$
$R_5 = p$ *est le résolvant de* R_3 *et* R_4
$R_6 = \neg p \vee r \in S$
$R_7 = \neg p$ *est le résolvant de* R_2 *et* R_6
$R_8 = \Box$ *est le résolvant de* R_5 *et* R_7
$R_1, R_2, R_3, R_4, R_5, R_6, R_7, R_8 = \Box$ *est une déduction de* $\Box$ *à partir de* S.

b) Propriété

S'il existe une déduction de $\Box$ à partir de S alors S est insatisfiable ou inconsistant.

Définition 14 *Une déduction de* $\Box$ *à partir de* S *est appelée réfutation.*

Chapitre 2

Exercices

Exercice 1 On considère les énoncés suivants :

L'utilisateur appuie sur la touche Cancel. L'utilisateur appuie sur la touche OK. Le programme se plante. Le fichier est effacé

On définit les trois propositions suivantes :

— A : "Si l'utilisateur appuie sur la touche OK, alors le programme ne se plante pas."

— B : "Le fichier est effacé si le programme se plante ou l'utilisateur appuie sur la touche Cancel."

1. Formaliser les énoncés ci-dessus en logique propositionnelle en précisant le sens des symboles utilisés.

2. Exprimer A et B en fonction des symboles utilisés dans la question précédente et des connecteurs logiques.

3. Écrire la négation de A et B.

Exercice 2 On suspecte Elise, Fred et Gaétan d'avoir commis un vol. Nous avons à leur sujet les informations suivantes :

— F_1 : Si Gaétan n'est pas coupable alors Fred est coupable
— F_2 : Si Elise n'est pas coupable alors Gaétan est coupable
— F_3 : Si Gaétan est coupable alors Elise l'est aussi
— F_4 : Si Elise est coupable alors Fred ne l'est pas

1. Formaliser ces énoncés dans le langage de la logique propositionnelle en précisant le sens des propositions atomiques utilisées.

2. Vérifier, selon le principe de résolution si

— Gaétan est coupable
— Fred est coupable
— Elise ou Fred est coupable
— Elise est coupable

Exercice 3 Nathalie confie à une amie d'enfance les renseignements suivants :

— Si je suis en vacances alors je fais du sport
— Si je ne suis pas en vacances alors je ne fais pas de régime
— Je suis détendue ou je ne fais pas de sport
— Je fais un régime

1. Formaliser ces énoncés dans le langage de la logique propositionnelle en précisant le sens des propositions atomiques que vous utilisez.

2. Vérifier, selon le principe de résolution si

— Nathalie fait du sport
— Nathalie n'est pas détendue
— Nathalie n'est pas en vacances
— Si Nathalie n'est pas détendue alors elle n'est pas en vacances

Exercice Sur la cour de l'école, Aubin donne les informations suivantes à son copain Kamel :

- Si je fais du vélo alors je fais du tennis
- Si je ne fais pas de vélo alors je ne fais pas de foot
- Je ne fais pas de tennis ou je fais de la trottinette
- Je fais du foot

1. Formaliser ces énoncés dans le langage de la logique propositionnelle en précisant le sens des propositions atomiques que vous utilisez.
2. Vérifier, selon le principe de résolution si

- Aubin fait du vélo
- Aubin ne fait pas de tennis
- Aubin ne fait pas de trottinette
- Si Aubin ne fait pas de trottinette alors il ne fait pas de vélo

Exercice 4 Soit l'énoncé suivant : La musique n'est ni triste ni rythmée. Il baille et il n'est pas joyeux. Quand il écoute de la musique rythmée, il est joyeux et il danse. Il danse, sauf s'il n'est pas joyeux. S'il danse en baillant, c'est qu'il n'est pas joyeux.

1. Formaliser cet énoncé dans le langage de la logique propositionnelle en précisant le sens des propositions atomiques que vous utilisez.
2. Montrer, selon le principe de résolution, que le fait "qu'il n'écoute que la musique non rythmée" est une conséquence logique des formules associées à la question précédente.

Exercice 5 Lors de ses aventures au pays des merveilles rapportées par Lewis Carroll, Alice est souvent accompagnée par le chat de Cheshire. Ce félin énigmatique s'exprime sous la forme d'affirmations logiques qui sont toujours vraies. Alice se trouve dans un corridor dont toutes les portes à sa taille sont fermées. La seule porte ouverte est nettement trop petite pour qu'elle puisse l'emprunter. Une étagère est fixée au-dessus de cette porte. Le chat dit alors à Alice : *L'un des flacons posés sur cette étagère contient un liquide qui te permettra de prendre une taille plus adéquate. Mais attention, les autres flacons peuvent contenir un poison fatal.*

Trois flacons sont effectivement posés sur l'étagère. Le premier est rouge, le second jaune, le troisième bleu. Une étiquette est collée sur chaque flacon. Alice lit l'inscription figurant sur chaque étiquette :

- **Flacon rouge** : "Le flacon jaune contient un poison ; le bleu n'en contient pas".
- **Flacon jaune** : "Si le flacon rouge contient un poison, alors le bleu aussi".
- **Flacon bleu** : "Je ne contiens pas de poison, mais au moins l'un des deux autres si".

Nous noterons R, J et B les variables propositionnelles correspondant au fait que les flacons rouge, jaune et bleu contiennent un poison.

Nous noterons IR, IJ et IB les propositions correspondant aux inscriptions sur les flacons rouge, jaune et bleu.

1. Formaliser les différentes inscriptions dans la logique propositionnelle en précisant le sens des propositions atomiques utilisées.
2. Les inscriptions sur les trois flacons sont-elles compatibles.
3. Supposons que les trois inscriptions sont vraies, quel flacon sera empoisonné ?

Chapitre 3

Logique des prédicats du Premier Ordre

3.1 Introduction

La logique des propositions est insuffisante pour traduire la plupart des concepts logiques intuitifs. Outre la notion de quantificateurs, il faut introduire celle de prédicats (qui décrivent des propriétés et des relations) et celle de fonctions au sens classique du terme.

Exemple 14 *Luc réside à Tunis et Max vit à Sfax (hypothèse). Luc et Max sont des hommes mais Bell n'est pas un homme (hypothèse). Tout homme est mortel (hypothèse). Il existe un mortel qui vit à Tunis (Conclusion).*

On ne peut, au niveau du calcul des propositions, traduire les relations contenues dans cet énoncé. Il s'agit de faire intervenir :

- *Des constantes : l, m, b, t, g représentant respectivement Luc, Max, Bell, Tunis et Sfax.*
- *Des propriétés : $Homme(X)$: X est un homme et $Mortel(X)$: X est mortel*
- *Des relations : $Reside(X,Y)$: X réside à Y où X et Y sont des variables.*

$Homme(X)$, $Mortel(X)$, $Reside(X,Y)$ sont appelés symboles de prédicats ; ils sont respectivement unaire, unaire et binaire. Pour traduire tout homme est mortel et il existe un mortel, on utilise les quantificateurs $\forall$ et $\exists$. On retrouve aussi les connecteurs : $\neg$, $\wedge$ et $\rightarrow$. Intuitivement on peut traduire :

1. *Luc réside à Tunis et Max vit à Sfax*
   $\Longrightarrow$

2. Luc et Max sont des hommes mais Bell n'est pas un homme
$\Longrightarrow$

3. Tout homme est mortel
$\Longrightarrow$

4. Conclusion : Il existe un mortel qui vit à Tunis
$\Longrightarrow$

Dans ce chapitre nous présenterons un langage formel : le langage des prédicats du premier ordre, qui permet d'exprimer des connaissances complexes avec rigueur, ainsi que des méthodes de combinaison de ces connaissances qui à leur tour en engendrent de nouvelles.

3.2 Syntaxe du langage

Les expressions correctes du langage, appelées *termes*, *atomes*, *littéraux* et *formules bien formées*, sont bâties à partir d'un alphabet de symboles.

3.2.1 Alphabet

Il comprend :

- Un ensemble de symboles appelés *séparateurs* : $\{(,),,\}$
- Un ensemble de symboles appelés *constantes.*
Ce sont les lettres minuscules de l'alphabet latin et les concaténations de telles lettres.
- Un ensemble de symboles appelés *variables.*
Ce sont les lettres majuscules de l'alphabet latin et les concaténations de telles lettres.
- Un ensemble de symboles appelés *prédicats.*
Ce sont, comme les variables, des chaînes en majuscules de l'alphabet latin telles que $P, Q, ENTRE, ...$ Pour manipuler un prédicat on devra convenir de son *arité* ou nombre d'arguments. L'arité est un nombre entier positif. Par exemple, on peut assigner les arités 1, 0, 2 respectivement pour les prédicats P, Q et $ENTRE$. Lorsque l'arité est fixé à zéro le prédicat est aussi appelé *proposition.*
- Un ensemble de symboles appelés *fonctions.*
Ce sont, comme les constantes, des chaînes de lettres minuscules telles que f, g, *poids*, *succ*, ... Tout comme pour les prédicats chaque symbole de fonction aura une *arité* fixée. Nous pouvons admettre que une fonction d'arité zéro est une *constante.* Exemple : *age* : fonction à un

paramètre qui fournit l'âge d'une personne, *distance* : fonction à deux paramètres qui fournit la distance entre deux points.

- Un ensemble de symboles appelés *connecteurs* : $\{\neg, \vee, \wedge, \rightarrow, \leftrightarrow\}$.
- Deux symboles appelés *quantificateurs* : $\forall$ quantificateur universel et $\exists$ quantificateur existentiel.

Exemple 15

— *“ Tout est relatif. ”*
$\Longrightarrow$
— *“ Une porte est ouverte ou fermée. ”*
$\Longrightarrow$
— *“ Tout ce qui brille n'est pas or. ”*
$\Longrightarrow$
— *“ Il y a des peines, il y a des plaisirs, mais aucune peine n'est un plaisir. ”*
$\Longrightarrow$
— *“ Tous les chemins mènent à Rome. ”*
$\Longrightarrow$
— *“ Pour tout entier il existe un entier plus grand. ”*
$\Longrightarrow$
— *“ Il existe un plus grand entier. ”*
$\Longrightarrow$

3.2.2 Termes

Définition 15 *Les termes sont définies inductivement comme suit :*

1. *Une constante est un terme.*
2. *Une variable est un terme.*
3. *Si f est un symbole de fonction d'arité n et si $t_1, t_2, ..., t_n$ sont des termes alors $f(t_1, t_2, ..., t_n)$ est un terme.*
4. *Les termes sont définis par un nombre fini d'applications des trois règles ci-dessus.*

Exemple 16 $b, Y_2, succ(X), f(ass_2, f(g(X_1), toto))$ *sont-ils des termes ?*

3.2.3 Atomes (ou formules atomiques)

Définition 16 *Les atomes sont définis par application des deux lois suivantes :*

1. *Les propositions (prédicats d'arité zéro) sont des atomes*
2. *Si P est un prédicat d'arité n ($n >= 1$) et si $t_1, t_2, ..., t_n$ sont des termes alors $P(t_1, t_2, ..., t_n)$ est un atome.*

Exemple 17
$P(X, bleu)$, $VIDE$, $ENTRE(table, X, appui(fenetre))$, $succ(X, Y)$ *ou* $appui(fenetre)$ *sont-ils des atomes ?*

3.2.4 Formules bien formées (fbf)

Définition 17 *Toute formule bien formée (fbf) est engendrée par application des trois lois suivantes :*

1. *Les atomes sont des fbf*
2. *Si G et H représentent des fbf, alors $(\neg G)$, $(G \vee H)$, $(G \wedge H)$, $(G \rightarrow H)$ et $(G \leftrightarrow H)$ sont des fbf.*
3. *Si G est une fbf et X une variable, alors $(\forall X)G(X)$ et $(\exists X)G(X)$ sont des fbf.*

Exemple 18 $(\exists X)(\forall Y)((P(X, Y) \vee Q(X, Y)) \rightarrow R(X))$, $((\neg P(a) \rightarrow P(b))) \rightarrow \neg P(b))$, $\neg f(a)$ *et* $f(P(a))$ *sont-elles des fbf ?*

Remarque 5

- *Une fbf qui est un atome ou qui est de la forme $(\neg G)$, G étant un atome, est appelé un littéral.*
- *Dans l'appellation logique des prédicats du premier ordre l'expression premier ordre est associée à la définition précédente des fbf qui interdit de quantifier des symboles de prédicats ou de fonctions.*

3.2.5 Variable libre et variable liée

Définition 18 *Une occurrence de la variable X est liée dans une fbf si elle est dans le champ d'un quantificateur $(\forall X)$ ou $(\exists X)$ ou si elle suit immédiatement le symbole $\forall$ ou $\exists$; sinon cette occurrence est dite libre.*

Exemple 19

$$\underset{Li\acute{e}e}{\overset{Libre}{(\forall X)((\exists Y)P(X,Y,Z) \vee Q(X,Z)) \vee R(X)}}$$

$$\underset{Li\acute{e}e}{\overset{Libre}{(\forall X)(((\exists Y)Q(X,Y)) \vee P(X,Y,Z)).}}$$

3.3 Règles d'inférence

Une *règle d'inférence* est la représentation d'un procédé pour, à partir d'une ou plusieurs fbf, dériver d'autres fbf. Ainsi :

- La règle d'inférence appelée ***Modus ponens***, à partir de deux fbf respectivement de la forme G et $(G \to H)$ dérive la fbf H.
- La règle d'inférence ***Spécialisation universelle***, à partir d'une fbf de la forme $(\forall X)G(X)$ et de n'importe quelle constante, soit a, dérive la fbf $G(a)$: toutes les occurrences de X dans G sont remplacées par a.
- La règle d'inférence appelée ***Modus tollens***, à partir de deux fbf respectivement de la forme $(\neg H)$ et $(G \to H)$, dérive la fbf $(\neg G)$

Les fbf choisies initialement sont appelées *axiomes*. Les fbf obtenues par application des règles d'inférence sont appelées *théorèmes*. Une chaîne d'applications de règles d'inférence conduisant, depuis les axiomes, à un théorème, constitue une preuve du théorème.

3.4 Sémantique de la logique des prédicats

On va examiner maintenant les bases qui permettent d'utiliser les fbf pour représenter (et raisonner sur) les valeurs de vérité que l'on accorde à des connaissances déjà disponibles en vue d'établir la valeur de vérité d'autres connaissances.

3.4.1 Interprétations

Comme en logique propositionnelle, on fixe d'abord un ensemble de modèles (aussi appelés ou interprétations). Ensuite, pour un modèle donné, on stipule des conditions de vérité permettant d'établir pour n'importe quelle formule A du langage si A est vraie ou fausse dans ce modèle. Pour donner un 'sens' aux variables, constantes et fonctions du langage, il faut un domaine d'objets (appelé aussi univers de discours). À chaque variable et constante sera associé un élément du domaine. À chaque fonction sera associé une application dans le domaine.

Ensuite, pour pouvoir donner un 'sens' aux formules, il faut une fonction associant à chaque symbole de prédicat d'arité n l'ensemble des n-uplets d'éléments du domaine qui le rendent vrai.

Finalement, l'interprétation des formules se fait comme en logique propositionnelle, avec comme interprétation de la quantification universelle $(\forall\, X)A(X)$ 'A est vrai pour toute interprétation de X'.

Définition 19 *Une interprétation I d'une fbf G est définie par les cinq étapes suivantes :*

- *Le choix d'un domaine d'interprétation non vide D dans lequel les variables individuelles prennent leurs valeurs.*
- *L'assignation à chaque constante de G un élément de D.*
- *L'assignation à chaque proposition (prédicat d'arité 0) d'un élément de l'ensemble particulier : $\{T, F\}$.*
- *L'assignation à chaque symbole de prédicat P d'arité n $(n >= 1)$ d'une application de $D^n \to \{T, F\}$.*
- *L'assignation à chaque symbole de fonction f d'arité n $(n >= 1)$ d'une application de $D^n \to D$.*

On dit qu'on a ainsi défini une interprétation de G sur D.

Exemple 20 *Soient les fbf suivantes :*

$$\begin{cases} G_1 : (\forall X)P(X) \\ G_2 : (\forall X)(\exists Y)Q(X,Y) \\ G_3 : (\forall X)(R(X) \to T(f(X),a)) \end{cases}$$

Ci-après nous définissons les interprétations I_1, I_2, I_3 de, respectivement, G_1, G_2, G_3 sur, respectivement, D_1, D_2, D_3. Nous notons F pour l'élément False de $\{True, False\}$ et T pour l'élément True.

$I_1 : D_1 = \{1, 2\}$

$P(1)$	$P(2)$
F	T

$I_2 : D_2 = \{1, 3\}$

$Q(1,1)$	$Q(1,3)$	$Q(3,1)$	$Q(3,3)$
F	T	F	F

$I_3 : D_3 = \{4, 5\}$

a	$f(4)$	$f(5)$	$R(4)$	$R(5)$	$T(4,4)$	$T(5,4)$
4	5	4	T	F	T	T

Remarque 6 *Il peut arriver parfois que l'on considère des interprétations dites incomplètes, dans lesquelles les assignations nécessaires ne sont que partiellement spécifiées.*

3.4.2 Evaluation d'une fbf dans une interprétation I

Soit une interprétation I, de domaine D, d'une formule G.

- Si G est une proposition : alors la valeur qui lui est assignée par définition de I est appelée : *valeur de G selon I.*
- Si G est un littéral non propositionnel : alors, pour chaque choix C de valeurs dans D pour les variables de G on obtiendra une valeur *vrai* ou *faux* en suivant la définition de I. Cette valeur est dite : *valeur de G selon I pour le choix C des valeurs des variables.*
- Si G est de la forme : $(\forall X)G'$, on définira la valeur de G selon I comme T (vrai) si la valeur de G' selon I pour toutes valeurs de la variable X (dans D) est T, sinon comme F (faux).
- Si G est de la forme : $(\exists X)G'$, on définira la valeur de G selon I comme T si la valeur de G' selon I pour au moins une valeur de X (dans D) est T, sinon comme F.
- Si G est de la forme $(G' \vee G")$ ou $(G' \wedge G")$ ou $(G' \rightarrow G")$ ou $(G' \leftrightarrow G")$, on définira la valeur de G selon I, lorsque les valeurs de G' et $G"$ selon I sont définies, au moyen des tables de vérités.
- Si G est de la forme : $(\neg G')$, on définira la valeur de $(\neg G')$ selon I, lorsque celle de G' selon I est définie, au moyen de la table suivante :

Valeur de G' selon I	Valeur de $\neg G$ selon I
T	F
F	T

Quand une formule G est T selon une interprétation I, on dit que I est *un modèle* de G ou I *satisfait* G.

3.4.3 Validité (et invalidité), inconsistance (et consistance) d'une fbf

Définition 20 *Une formule est valide si et seulement si sa valeur est T selon toute interprétation. Sinon elle est invalide. Une formule est inconsistante si et seulement si sa valeur est F selon toute interprétation. Sinon elle est consistante.*

Exemple 21

$$\textit{Soient} \begin{cases} G_1 : (\forall X)(P(X) \vee (\neg Q(X))) \\ G_2 : (\forall X)(P(X) \vee (\neg P(X))) \\ G_3 : ((\forall X)P(X)) \wedge ((\exists Y)(\neg P(Y))) \end{cases}$$

Soit $I'_1 : D = \{1\}; P(1)/T; Q(1)/T$. La formule G_1 est consistante car l'interprétation I'_1 suivante lui assigne la valeur T.
Soit $I" : D = \{2\}; P(2)/F; Q(2)/T$. La formule G_1 est invalide car l'interprétation $I"$ lui assigne la valeur F.

Montrons que la formule G_2 est valide
Soit I une interprétation du domaine qui falsifie G_2 alors il existe donc une valeur a de X, prise dans D, telle que $(P(a) \vee (\neg P(a)))$ soit F, ce qui est impossible par les tables de vérité des connecteurs $\vee$ et $\neg$; donc G_2 est valide.

Montrons que la formule G_3 est inconsistante
Soit I un modèle de G, de domaine D. I doit satisfaire $(\exists Y)(\neg P(Y))$ donc il existe a dans D tel que $P(a)$ vaut F ; mais alors $(\forall X)P(X)$ ne peut être satisfaite sur D. Donc G_3 est inconsistante.

3.4.4 Formules équivalentes

Deux fbf G et H sont équivalentes si et seulement si elles prennent les mêmes valeurs (T ou F) pour toute interprétation et nous la notons par : Pour toute interprétation I, $I(G) = I(H)$.

On peut vérifier, par exemple, que les formules $P(a) \rightarrow Q(b)$ et $(\neg P(a) \vee Q(b))$ sont équivalentes grâce aux tables de vérité.

Nous donnons une liste d'équivalences d'usage courant (à part celle développée dans la logique des propositions). Soient G et H des fbf et, $G(X)$ et $H(X)$ des fbf où X est une variable.

Formules equivalentes	**Appellations**
$(\forall X)(G(X)) \equiv (\forall Y)(G(Y))$ $(\exists X)(G(X)) \equiv (\exists Y)(G(Y))$	*Lois des variables muettes*
$\neg(\exists X)(G(X)) \equiv (\forall X)(\neg G(X))$ $\neg(\forall X)(G(X)) \equiv (\exists X)(\neg G(X))$	
$(\forall X)(G(X) \wedge H(X)) \equiv (\forall X)G(X) \wedge (\forall Y)H(Y)$ $(\exists X)(G(X) \wedge H(X)) \equiv (\exists X)G(X) \wedge (\exists Y)H(Y)$	

Chapitre 4

Exercices

Exercice Formaliser en Calcul des prédicats les phrases suivantes :

1. Les baleines sont des mammifères.
2. Les entiers sont pairs ou impairs.
3. Il existe un entier pair

Exercice Soient R et F deux symboles de prédicats d'arité 1 définis respectivement comme suit :

$R(X)$: signifie “X est une rose”.
$F(X)$: signifie “X est une fleur”.

Représenter dans le langage des prédicats de premier ordre les énoncés suivants :

1. Toutes les roses sont des fleurs.
2. Aucune rose n'est une fleur.
3. Quelques roses sont des fleurs.
4. Quelques roses ne sont pas des fleurs.

Exercice On veut formaliser un tournoi de tennis en utilisant la logique des prédicats de premier ordre. Pour cela on introduit les données suivantes :

- Deux constantes a et b qui représentent deux joueurs nommés Ali et Bilel.
- Deux symboles de prédicats d'arité 1 de nom I et E tels que :
  - $I(X)$: signifie "X est inscrit au tournoi".
  - $E(X)$: signifie "X est éliminé du tournoi".
- Deux symboles de prédicats d'arité 2 de nom A et B tels que :
  - $A(X, Y)$: signifie "X a joué avec Y".
  - $B(X, Y)$: signifie "X a battu Y".

1. Traduire en formules logiques les assertions suivantes :
   - Ali et Bilel sont inscrits au tournoi.
   - Tout joueur battu est éliminé.
   - Bilel a battu tous les joueurs inscrits qui ont joué contre Ali.
2. Exprimer en langue naturelle les formules suivantes :
   - $\neg(\forall X)(I(X) \rightarrow A(X, a))$
   - $(\exists X)(\forall Y)(I(X) \wedge B(b, X) \wedge (I(Y) \wedge A(Y, a) \rightarrow B(X, Y)))$
   - $(\forall X)(\exists Y)(I(X) \wedge B(X, b) \rightarrow (I(Y) \wedge B(Y, a) \wedge B(X, Y)))$
3. Soient les trois formules suivantes :
   - F_1 : $(\exists X)(I(X) \wedge \neg E(X))$
   - F_2 : $(\forall X)(I(X) \rightarrow ((\exists Y)B(Y, X) \rightarrow E(X)))$
   - F_3 : $(\forall X)(\forall Y)(B(X, Y) \rightarrow \neg B(Y, X))$

   3.1 Trouver la forme normale prenex et la forme normale clausale pour les formules F_1, F_2 et F_3.

   3.2 Chercher, à l'aide du principe de résolution, une conséquence logique pour F_1, F_2 et F_3.

Exercice

1. Représenter dans le langage de prédicats de premier ordre les énoncés suivants :
   - E_1 : Tous les étudiants qui ont choisi le cours Intelligence artificielle ont déjà suivi le cours systèmes logiques.
   - E_2 : Pas tous les étudiants suivent le cours Intelligence artificielle et systèmes logiques.
   - E_3 : Les étudiants qui suivent le cours Intelligence artificielle sont de bons informaticiens.
2. Chercher une conséquence logique à partir de E_1, E_2 et E_3.

Exercice On admet les énoncés suivants :

— les chevaux sont plus rapides que les chiens
— il existe un lévrier plus rapide que tout lapin
— les lévriers sont des chiens
— Harry est un cheval
— Ralph est un lapin

1. Formaliser les énoncés précédents en logique de prédicats de premier ordre en précisant le sens des prédicats utilisés.
2. Peut-on déduire que "Harry est plus rapide que Ralph" ?

Exercice On veut formaliser un tournoi de tennis en utilisant la logique des prédicats de premier ordre. Pour cela on introduit les données suivantes :

- Deux constantes a et b qui représentent deux joueurs nommés Ali et Bilel.
- Deux symboles de prédicats d'arité 1 de nom I et E tels que :
  - $I(X)$: signifie "X est inscrit au tournoi".
  - $E(X)$: signifie "X est éliminé du tournoi".
- Deux symboles de prédicats d'arité 2 de nom A et B tels que :
  - $A(X, Y)$: signifie "X a joué avec Y".
  - $B(X, Y)$: signifie "X a battu Y".

1. Traduire en formules logiques les assertions suivantes :
   - Ali et Bilel sont inscrits au tournoi.
   - Tout joueur battu est éliminé.
   - Bilel a battu tous les joueurs inscrits qui ont joué contre Ali.
2. Exprimer en langue naturelle les formules suivantes :
   - $\neg(\forall X)(I(X) \rightarrow A(X, a))$
   - $(\exists X)(\forall Y)(I(X) \wedge B(b, X) \wedge (I(Y) \wedge A(Y, a) \rightarrow B(X, Y)))$
   - $(\forall X)(\exists Y)(I(X) \wedge B(X, b) \rightarrow (I(Y) \wedge B(Y, a) \wedge B(X, Y)))$
3. Soient les trois formules suivantes :
   - F_1 : $(\exists X)(I(X) \wedge \neg E(X))$
   - F_2 : $(\forall X)(I(X) \rightarrow ((\exists Y)B(Y, X) \rightarrow E(X)))$
   - F_3 : $(\forall X)(\forall Y)(B(X, Y) \rightarrow \neg B(Y, X))$

   3.1. Trouver, pour les formules F_1, F_2 et F_3, la forme normale prenex et la forme normale clausale.

   3.2. Montrer en utilisant le principe de résolution que le fait "Il y a un joueur qui n'a pas été battu" est une conséquence logique des formules F_1, F_2 et F_3.

Exercice Soient M, H, V, C et L des symboles de prédicat de premier ordre définis respectivement comme suit :

$M(X,Y)$: signifie " X mange Y ".
$H(X)$: signifie " X est un herbivore ".
$V(X)$: signifie " X est un végétal ".
$C(X)$: signifie " X est une carotte ".
$L(X)$: signifie " X est un lapin ".

3.1. Représenter dans le langage des prédicats de premier ordre les énoncés suivants :
- F_1 : Les herbivores mangent des végétaux.
- F_2 : Les herbivores ne mangent que des végétaux.
- F_3 : Aucun herbivore ne mange tout type de végétal.
- F_4 : Certains herbivores ne mangent pas de carottes.
- F_5 : Les lapins sont des herbivores qui ne consomment que des carottes.

3.2. Trouver, pour les formules F_1, F_2 et F_4, la forme normale prenex et la forme normale clausale.

3.3. En utilisant le principe de résolution, trouver une conséquence logique des formules F_1, F_2 et F_4.

Chapitre 5

Principe de résolution

C'est une règle d'inférence qui s'applique à une famille particulière de fbf appelées *clauses.*

5.1 Les clauses

On appelle clauses toutes fbf qui ont la forme d'une disjonction de littéraux. Par exemple : $R(Z, a, g(X)) \vee (\neg T(U)) \vee (\neg V(b, k(c))$

5.2 Forme Normale Prenex d'une fbf

C'est une formule dérivée d'une fbf en appliquant la séquence des quatre transformations ci-après.

1. **Éliminer les connecteurs** $\rightarrow$ et $\leftrightarrow$. Pour ce faire, il faut utiliser les lois d'équivalence entre :
   $(G \rightarrow H) \equiv$
   $(G \leftrightarrow H) \equiv$
2. **Accoler les connecteurs** $\neg$ aux atomes concernés. Pour ce faire, utiliser les lois d'équivalence entre :
   $(\neg(\neg G)) \equiv$
   $(\neg(G \vee H)) \equiv$
   $(\neg(G \wedge H)) \equiv$
   $\neg((\exists X)G(X)) \equiv$
   $\neg((\forall X)G(X)) \equiv$
3. **Distinguer les variables** (de sorte que chaque quantificateur gouverne une variable originale). Pour ce faire, utiliser les lois d'équivalence entre :

$(\forall X)G(X) \equiv$
$(\exists X)G(X) \equiv$

4. **Déplacer tous les quantificateurs à gauche de la formule sans changer leur ordre**. Ce déplacement conduit à une nouvelle fbf équivalente à la précédente car, après l'étape 3, il n'y avait plus de conflit possible entre étiquettes de variables quantifiées.

Au terme de ces quatre étapes on obtient une *forme normale prenex* de la fbf initiale, qui lui est équivalente.

Exemple 22 *Soit la* fbf $G : [(\forall X)((P(X) \wedge Q(X,a)) \to R(X,b))] \vee ((\forall X)S(X))$

Après l'étape 1, on obtient :

Après les étapes 2 :

Après les étapes 3 :

Après l'étape 4, on obtient une forme normale prenex de G

5.3 Passage sous forme clausale

A partir d'une forme normale prenex G' d'une fbf on peut produire une forme de clauses G" de G par les cinq transformations 1, 2, 3, 4 et 5 ci-après décrites. Soulignons immédiatement que si les formules G et G' sont toujours équivalentes il n'en n'est pas de même pour G et G".

1. **Éliminer les quantificateurs existentiels :**

   Soit une formule de type $(\exists X)G(X)$ et supposons que cette formule soit une sous-formule d'une ou plusieurs formules quantifiées universellement par rapport à $Y_1, ..., Y_p$. On supprimera $(\exists X)$ et on remplacera chaque occurrence de X dans $G(X)$ par une fonction telle que $f(Y_1, ..., Y_p)$; Notons bien que cette fonction comportera autant d'arguments qu'il existe de quantificateurs universels à gauche de la formule $(\exists X)G(X)$. De telles fonctions sont dites *fonctions de Skolem.* Lorsqu'il n'existe aucun quantificateurs universels à gauche de l'existentiel considéré, la fonction de Skolem introduite n'aura pas d'argument et par la suite elle sera considérée comme une nouvelle constante appelée *constante de Skolem*.

Les différents cas possibles pour cette étape sont les suivants :

1^{er} **cas :** $(\forall X)(\exists Y)[G(X) \wedge P(Y)]$

2^{me} **cas :** $(\forall X)(\forall Y)(\exists Z)[G(X) \wedge P(Y) \wedge Q(Z)]$

3^{me} **cas :** $(\exists Y)(\forall X)[G(X) \wedge P(Y)]$

4^{me} **cas :** $(\exists Y)(\exists Z)(\forall X)[G(X) \wedge P(Y) \wedge Q(Z)]$

Exemple 23
$(\exists X)P(X)$ *devient ;*
$(\forall X)(\exists Y)SUIT(Y, X)$ *devient ;*

La Forme Normale Prenex de **l'exemple** 22 est :

Après l'étape 1 :

2. **Éliminer tous les quantificateurs :**
   A la fin de l'étape précédente il ne reste que des quantificateurs universels. On allège la notation en les supprimant.

   Après l'étape 2 :

3. **Passer sous forme normale conjonctive :**
   Il s'agit de passer sous la forme d'une conjonction de littéraux, c'est à dire une conjonction de clauses. A cet effet on utilise les lois d'associativité et de distributivité des connecteurs $\wedge$ et $\vee$.

   Après l'étape 3 :

4. **Éliminer les connecteurs $\wedge$:**
   La conjonction de clauses obtenues à la fin de l'étape précédente est traditionnellement considérée comme un ensemble de clauses en supprimant le connecteur $\wedge$.

   Après l'étape 4 :

5. **Distinguer les variables de clauses distinctes**.

   Après l'étape 5 :

5.4 Le principe de résolution appliqué à des clauses concrètes

On dira qu'un littéral est concret s'il ne comporte aucune variable. Par exemple $P(a)$, $Q(a, f(b))$ sont des littéraux concrets tandis que $P(X)$ ou $Q(a, Y)$ n'en sont pas. Une clause concrète est une disjonction de littéraux concrets.

Soient deux clauses concrètes $G = G_1 \vee G_2 \vee ... \vee G_n$ et $H = \neg G_1 \vee H_2 \vee ... \vee H_m$ où, les G_i et H_j sont des littéraux concrets. Les littéraux G_1 et $\neg G_1$, dans G et

H respectivement, sont dits littéraux complémentaires. La règle d'inférence appelée *principe de résolution* produit, à partir de G et H la clause $K = G_2 \vee ... \vee G_n \vee H_2 \vee ... \vee H_m$ appelée *clause résolvante* ou *résolvant* de G et H. On dit aussi que G et H *se résolvent* en K. Un résolvant est défini par élimination de littéraux complémentaires et disjonction de tous les autres littéraux des clauses parentes.

La généralisation du principe de résolution à des clauses quelconques conduit à chercher au sein des clauses parentes des littéraux complémentaires dans un sens plus large que précédemment. Cette recherche utilise une opération appelée *unification* que nous allons présenter maintenant.

5.5 L'unification

Considérons, par exemple, les clauses $\neg G(X) \vee H(X)$ et $G(f(Y))$. Si la première clause était remplacée par $\neg G(f(Y)) \vee H(f(Y))$, le principe de résolution pour clauses concrètes serait aisément étendu. En effet, après élimination des littéraux complémentaires $\neg G(f(Y))$ et $G(f(Y))$ on obtiendrait la clause $H(f(Y))$. Justement, l'opération d'unification permet de *transformer des clauses en vue de faire apparaître des littéraux complémentaires, par application de substitutions.*

5.5.1 Substitutions

Une substitution est un ensemble fini de couples notés t_i/V_i où les t_i sont des termes et les V_i sont des variables. L'application d'une substitution $s = \{t_i/V_i\}_i$ à une expression quelconque E est noté E_s et s'appelle une *instantiation* de E selon s. Elle consiste à remplacer toutes les occurrences initiales de chaque variable V_i dans E par t_i.

Exemple 24 *soient* $E = \{G(f(X), a, Y)\}$ *et* $s_1 = \{Z/X, U/Y\}$, $s_2 = \{b/X\}$, $s_3 = \{Y/X, g(X)/Y\}$, $s_4 = \{a/X, k(c)/Y\}$.
On a $E_{s_1} = \{G(f(Z), a, U)\}$, $E_{s_2} = \{G(f(b), a, Y)\}$, $E_{s_3} = \{G(f(Y), ag(X))\}$, $E_{s_4} = \{G(f(a), a, k(c))\}$.

5.5.2 Unifieurs

On dit qu'un ensemble $\{E_i\}_i$ d'expressions (termes ou formules) est *unifiable* par s ou que s est un *unifieur* de $\{E_i\}_i$ si et seulement si tous les E_{is} sont identiques. Nous noterons l'expression produite par un unifieur s : $\{E_i\}_s$.

Exemple 25 *$s = \{a/X, c/Y, c/V, b/Z, b/U, g(b)/W\}$ est un unifieur de $\{E_i\}_i =$*

$$\{G(X, f(Y), g(b)), G(X, f(c), g(Z)), G(X, f(c), g(U)), G(X, f(V), W)\}$$

puisque toutes les expressions sont instanciées en $G(a, f(c), g(b))$. Il peut exister plusieurs unifieurs pour un ensemble d'expressions donné.

5.5.3 Algorithme d'unification

L'algorithme récursif, présenté informellement ci-après, produit un unifieur pour un ensemble E fini d'expressions unifiables. Si E n'est pas unifiable, l'algorithme s'arrête en le déclarant. On appelle initialement l'algorithme par $UNIFIER(E, \sigma)$ où σ dénote la substitution vide.

L'algorithme utilise la notion d'*ensemble de discordance* noté $\mathcal{D}$ d'un ensemble E d'expressions. Cet ensemble est construit en balayant de gauche à droite simultanément tous les éléments de E jusqu'à la première position de symbole qui fait apparaître une différence entre ces éléments, puis en extrayant de chaque élément de E l'expression qui débute en cette position discordante de symbole. L'ensemble de ces expressions constitue $\mathcal{D}$.

Algorithme UNIFIER (E, σ)

Début
01. **Si** (E est un singleton (tous les éléments identiques)) **alors**
    arrêter en éditant σ comme unifieur
  Fin_si
02. Former l'ensemble $\mathcal{D}$ de discordance de E
03. **Si** (il existe deux éléments V et t de $\mathcal{D}$ tel que V est une variable, t est un terme et V ne figure pas dans t) **alors**
    $\sigma = \sigma \circ \{t/V\}$
    $E = E_{\{t/V\}}$
    $UNIFIER(E, \sigma)$
  Fin_si
10. arrêter : l'ensemble de départ n'est pas unifiable
Fin.

Exemple 26 *Soit $E = \{P(a, X, f(g(Y))), P(Z, f(Z), f(U))\}$. Appliquer l'algorithme d'unification pour trouver un unifieur pour E.*

Définition 21 *Soient C_1 et C_2 deux clauses qui n'ont aucune variable commune. Soient L_1 et L_2 deux littéraux de C_1 et C_2 respectivement. Si L_1 et $\neg L_2$ ont un unifieur s alors la clause $(C_1.s - L_1.s) \cup (C2.s - L2.s)$ est appelée résolvant binaire de C_1 et C_2.*

Exemple 27 $C_1 = P(a, X) \vee R(X)$ *et* $C_2 = \neg P(T, f(T)) \vee Q(T, T)$
$\{P(a, X), P(T, f(T))\}$ *possède un unifieur* $s = \{a/T, f(a)/X\}$
$C_1.s = P(a, f(a)) \vee R(f(a))$
$C_2.s = \neg P(a, f(a)) \vee Q(a, a)$
$C = R(f(a)) \vee Q(a, a)$ *est le résolvant de* C_1 *et* C_2.

Théorème 3 *le résolvant C de deux clauses C_1 et C_2 est la conséquence logique de ces deux clauses.*

Lemme 1 *Si C'_1 et C'_2 sont des instances de C_1 et C_2 respectivement et si C' est un résolvant de C'_1 et C'_2 alors il existe un résolvant C de C_1 et C_2 dont C' est une instance.*

Théorème 4 *Un ensemble S de clauses est inconsistant si et seulement si il existe une déduction par résolution de la clause vide.*

Chapitre 6

Exercices

Exercice Soient les formules suivantes en logique des prédicats de premier ordre dans lesquels F et G sont des prédicats, f une fonction, X une variable et a une constante :

$$\begin{cases} F_1 & : & F(a) \\ F_2 & : & (\forall X)(F(X) \rightarrow G(f(X))) \\ F_3 & : & (\forall X)(G(X) \rightarrow F(f(X))) \\ C & : & P(f(f(f(f(a))))) \end{cases}$$

1. Donner la forme clausale de chacune des formules F_1, F_2, F_3 et C.
2. Montrer que C est une conséquence logique de l'ensemble des clauses de F_1, F_2 et F_3 en utilisant le principe de résolution