Polycopié de cours et exercices corrigés

INTRODUCTION A LA LOGIQUE MATHEMATIQUE

Support destiné aux étudiants de la 1ère année PES (Professeur d'Enseignement Secondaire) informatique, Licence Mathématique et Informatique (MI)

Dr. BENKADDOUR Fatima Zohra
Enseignante au département des sciences exactes
E.N.S d'Oran

Année universitaire 2018-2019

Résumé

Cet ouvrage propose une introduction à la logique mathématique accessible aux étudiants de la première année PES (Professeur d'enseignement Secondaire) à l'ENS d'Oran (E.N.S.O) ainsi qu'aux étudiants du domaine Mathématique et Informatique (MI).

Il aborde les techniques fondamentales de la logique propositionnelle dite d'ordre 0 et la logique des prédicats du premier ordre. Ce document propose une série d'exercices résolus qui conduisent l'étudiant à une connaissance approfondie des notions de base de la logique.

Mots-clés : *Logique propositionnelle, Logique des prédicats, PES, E.N.S.O, MI.*

Table des matières

Table des figures

Liste des tableaux

Avant-propos

Ce présent document se veut un support de cours pour les étudiants de la 1^{re} année PES (Professeur d'enseignement Secondaire) à l'École Normale Supérieure d'Oran (E.N.S.O) [1].

Néanmoins, il décrit aucune spécificité propre à l'environnement informatique de l'ENSO et pourra s'avérer utile à tout étudiant désirant apprendre les bases de la logique mathématique.

Ce document n'est pas exhaustif et survole les notions de base de la logique qui sont généralement les plus utiles.

La structure et le contenu des chapitres de ce document sont synchronisés avec le nouveau contenu du programme établi dans le canevas de la formation. Il est organisé au tour de trois (03) chapitres et trois (03) annexes.

— Le premier chapitre intitulé *Notions de base de la logique mathématique* introduit les notions fondamentales de la logique mathématique qui s'avèrent indispensables aux chapitres suivants.

— Le deuxième chapitre appelé *Logique propositionnelle (d'ordre 0)* ou *Calcul propositionnel* est la première étape dans la construction du calcul des prédicats. Il permet de définir les règles de déduction qui relient les propositions entre elles, sans en examiner le contenu.

— Le troisième chapitre aborde la notion de *Logique des prédicats (d'ordre 1).* La logique du premier ordre est considérée par nature plus expressive que la logique des propositions, et permet de représenter des connaissances relatifs à des environnements complexes. Elle est construite à partir de la logique propositionnelle et s'inspire du langage naturel.

— En annexes, nous proposons une série d'exercices résolus ainsi qu'un examen corrigé.

Ce cours permet aux étudiants de :

- Connaître les principaux opérateurs,quantificateurs universels et leurs propriétés : NON, ET, OU,...
- Comprendre les notions d'implication et d'équivalence ;
- Modéliser un énoncé afin de tester sa validité ;
- Appliquer toutes ces notions à la démonstration mathématique ;
- Structurer proprement un raisonnement ;
- Tester ses connaissances à travers une série d'exercices corrigés.

1. www.ens-oran.dz

Chapitre 1

Notions de base de la logique mathématique

1.1 Introduction

La logique est la base fondamentale de tous les raisonnements mathématiques. Elle est très importante pour l'énonciation de propositions et l'étude de leur valeur de vérité.
Dans ce premier chapitre, nous introduirons les bases de la branche des mathématiques appelée logique. Nous présenterons en particulier les définitions d'assertions, de tautologies et d'antilogies ainsi que les différents connecteurs logiques.
Ce chapitre prévoit les notions apparaissant dans les deux chapitres suivants soient la logique propositionnelle et la logique descriptive du premier ordre.

1.2 Assertion

Une assertion est un énoncé mathématique auquel on attribue l'une des deux valeurs logiques : le vrai (V) ou le faux (F).

Exemple

— L'assertion «$1 + 1 = 2$» est vraie.
— L'assertion «$2 + 2 = 5$» est fausse.

— Les propriétés, théorèmes sont des assertions vraies.

1.3 Tautologies et antilogies

Les assertions (dépendantes de P et Q) qui sont vraies quelle que soit la valeur de vérité de P et Q sont dites des tautologies. Une tautologie est en fait un théorème de logique. Les assertions (dépendantes de P et Q) qui sont fausses quelle que soit la valeur de vérité de P et Q sont dites des antilogies.

1.4 Connecteurs

Ils existent cinq (5) connecteurs logiques, à la base de tout raisonnement mathématique. Soient P et Q deux assertions.

1.4.1 La négation « non » ou « ¬ »

Nous appellerons la négation de P, l'assertion (non P)(not P) et qui sera notée sous forme formalisée ¬P ou $\bar{P}$.

Table de vérité de la négation

Soit la proposition P (voir la table 1.1). La négation d'une proposition P (vraie)

P	$\neg P$
1	0
0	1

TABLE 1.1: Table de vérité de la négation

est une proposition fausse. Si P (fausse) alors $\neg P$ est vraie.

Négation de la négation

En général, une double négation vient souvent renforcer la négation tel que : voulez-vous sortir ? non, non.
En mathématique, une double négation est considérée comme une affirmation.

Exemples

1. Si P est la proposition $x = 0$, $\neg P$ est la proposition $x \neq 0$.
2. Le 05 est non pair donc 05 est impair.

Remarque

Le sens du symbole $\Longleftrightarrow$ qui se lit équivaut, et qui signifie ici que les deux propositions ont toujours la même valeur.

1.4.2 Conjonction « et » ou « $\wedge$»

Nous appellerons conjonction de P et Q, l'assertion (P et Q) (P and Q) et qui sera notée P $\wedge$ Q.

Exemple

P : « La terre est ronde »(vraie) et Q : « Le ciel est bleu » (vraie).
P et Q ou P $\wedge$ Q se lit donc « La terre est ronde **ET** le ciel est bleu ». P $\wedge$ Q est vraie. Nous dirons que l'assertion P $\wedge$ Q est fausse lorsque l'une au moins des deux assertions est fausse. Ainsi « La terre est ronde **ET** le ciel est vert » est une assertion fausse.

Commutativité

(P et Q) $\Longleftrightarrow$ (Q et P)

Table de vérité de la conjonction

Le résultat de la conjonction est démontré par la table de vérité 1.2.

P	**Q**	$P \wedge Q$
1	1	1
1	0	0
0	1	0
0	0	0

TABLE 1.2: Table de vérité de la conjonction

1.4.3 Disjonction « ou » ou « $\vee$ »

Nous appellerons disjonction de P et Q, l'assertion (P ou Q) et qui sera notée P $\vee$ Q.

Remarque

En mathématiques, le «ou» est non-exclusif, c'est à dire qu'il comprend la possibilité que les deux propositions soient vraies. Ainsi la proposition « $xy = 0$ » équivaut à la proposition « $x = 0$ ou $y = 0$ » , elle est vraie quand l'un des deux nombres est nul, elle est aussi vraie quand les deux sont nuls.

Commutativité

$(P$ ou $Q) \Longleftrightarrow (Q$ ou $P)$

Table de vérité de la disjonction

Le résultat de la disjonction est démontré par la table de vérité 1.3.

P	**Q**	$P \vee Q$
1	1	1
1	0	1
0	1	1
0	0	0

TABLE 1.3: Table de vérité de la disjonction

1.4.4 Implication « $\Rightarrow$ »

La proposition notée « P $\Rightarrow$ Q » correspond à la proposition $NonP$ ou Q. P s'appelle alors l'hypothèse et Q la conclusion. $P \Rightarrow Q$ est une proposition qui se nomme implication et que nous pouvons lire de différentes façons :

— Si P alors Q,
— Pour que P il faut Q,
— Pour que Q il suffit P,
— P est une condition suffisante pour Q,
— Q est une condition nécessaire de P.

Table de vérité de l'implication

Le résultat de l'implication est montré à la table 1.4.

P	**Q**	$\neg P$	**P $\Rightarrow$ Q**
1	1	1	1
0	1	1	1
1	0	0	0
0	0	1	1

TABLE 1.4: Table de vérité de la disjonction

L'assertion est vraie dès lors que P est fausse (quelle que soit la vérité de Q). Si P est vraie et (P $\Rightarrow$ Q) vraie alors Q est vraie. De plus l'implication Q $\Rightarrow$ P s'appelle la réciproque de l'implication P $\Rightarrow$ Q.

Exemples

— $P : 2 = 2$ et $Q : 4 = 4$ sont deux assertions vraies, donc $P \Rightarrow Q$ ou ($NonP$ ou Q) est vraie,
— Si $x \in \{5, 6, 9\}$ alors $x \leqslant 9$ est une assertion vraie.

1.4.5 Équivalence « $\Leftrightarrow$ »

Nous dirons que deux assertions sont logiquement équivalentes si elles ont la même valeur de vérité et seront notées $P \Leftrightarrow Q$. En d'autres termes $P \Leftrightarrow Q$ est vraie si P et Q sont toutes les deux vraies ou si toutes les deux sont fausses. La proposition $P \Leftrightarrow Q$ correspond à la proposition $(P \Rightarrow Q)$ et $(Q \Rightarrow P)$. Nous pourrons l'exprimer comme suit :

— P est équivalent à Q,
— Pour P, il faut et il suffit Q,
— P est une condition nécessaire et suffisante pour Q,

— P si et seulement si Q.

Table de vérité de l'équivalence

Le résultat de l'équivalence est montré à la table 1.5.

P	**Q**	$P \Leftrightarrow Q$
1	1	1
0	1	0
1	0	0
0	0	1

TABLE 1.5: Table de vérité de l'équivalence

Exemple

Prenons (P et Q) et (Non (Non P ou Non Q)). Voir le résultat à la table 1.6.

1.5 Lois de Morgan

Les lois de De Morgan permettent de transformer une conjonction en une disjonction (et réciproquement) via la négation.

P	**Q**	**Non P**	**Non Q**	**Non P $\vee$ Non Q**	**Non(Non P ou Non Q)**	**P $\wedge$ Q**
V	V	F	F	F	V	V
F	V	V	F	V	F	F
V	F	F	V	V	F	F
V	F	F	V	V	F	F

TABLE 1.6: Exemple de l'équivalence

Remarque

La méthode des tables de vérité fournit une preuve de ces propriétés. Parmi les règles de calcul logique importantes il y a aussi les règles suivantes.

1. **Règle de double-négation :**

$$\bar{\bar{A}} = A$$

2. **Règle de commutativité :**

$$\text{A} * \text{B} = \text{B} * \text{A}$$

   avec $* = .$ ou $* = +$

3. **Règle de distributivité :**

$$\text{A} + (\text{B} \,.\, \text{C}) = (\text{A} + \text{B}) \,.\, (\text{A} + \text{C})$$

et

$$\text{A} \,.\, (\text{B} + \text{C}) = (\text{A} \,.\, \text{B}) + (\text{A} \,.\, \text{C})$$

4. **Règle d'associativité :**

$$\text{A} * (\text{B} * \text{C}) = (\text{A} * \text{B}) * \text{C}$$

5. **Règle d'idempotence :**

$$\mathrm{A} * \mathrm{A} = \mathrm{A}$$

   avec $* = .$ ou $* = +$

Toute expression booléenne peut être écrite sous une forme particulière.

1. **Forme normale conjonctive (FNC) :** Une expression logique est en FNC si et seulement si elle est une conjonction d'une ou plusieurs disjonctions d'un ou plusieurs littéraux tel que : $(A + \bar{B} + \bar{C}).(\bar{D} + E + F)$
2. **Forme normale disjonctive (FND) :** Une expression logique est en FND si et seulement si elle est une disjonction d'une ou plusieurs conjonctions d'un ou plusieurs littéraux. Par exemple,$(\bar{A}.\bar{B}) + B + (\bar{C}.D.E)$; sachant qu'un littéral est une variable booléenne (une lettre) ou la négation d'une variable.

1.6 Les quantificateurs

Nous considérons dans ce qui suit deux types de quantificateurs.

1.6.1 Le quantificateur universel

Un quantificateur permet de préciser le domaine de validité d'une proposition. Le symbole $\forall$ qui signifie « quel que soit » ou « pour tout » représente le quantificateur universel. Ce symbole représente la lettre « A » renversée qui est l'initiale du mot anglais « All ». Il doit toujours être suivi du signe d'appartenance $\in$.

Exemple

$$\forall x \in R, x^2 \geq 0$$

Qui signifie « quel que soit x appartenant à R, x^2 est positif ou nul ».

1.6.2 Le quantificateur existentiel

Le symbole $\exists$ qui signifie « il existe au moins un ... tel que » représente le quantificateur existentiel. Ce symbole représente la lettre « E » renversée qui est l'initiale du mot anglais « exist ». On peut éventuellement rajouter un point d'exclamation pour montrer l'unicité. On a alors : $\exists!$ qui signifie « *il existe un unique...tel que*».

Exemple

$$\exists! x \in [0,1], x^2 + 4x + 1 = 0$$

Qui signifie « Il existe un unique x appartenant à l'intervalle $[0,1]$ tel que : $x^2 + 4x + 1 = 0$ ».

1.6.3 Propriétés des quantificateurs

Les quantificateurs possèdent un certain nombre de propriétés dont :

— **L'ordre des quantificateurs :** L'ordre dans lequel on écrit les quantificateurs change la signification.
  Exemple : $\forall x \in R, \exists y \in R, y > x$, qui signifie « *Quel que soit le réel x, il existe au moins un réel y tel que y soit supérieur à x* ». On peut toujours trouver un nombre supérieur à un nombre réel donné car l'ensemble R n'est pas borné. La proposition est vraie.
  Inversons maintenant les quantificateurs $\exists x \in R, \forall y \in R, y > x$, « *Il existe au moins un réel x tel que pour tout réel y, y soit supérieur à x* ». Cette proposition cette fois est fausse car on ne peut trouver un réel inférieur à tous les autres. En effet l'ensemble R n'a pas de borne inférieure.

— **Négation d'une proposition universelle :**Une proposition universelle s'énonce : « *Pour tout élément x d'un ensemble E, x possède la proposition P* ». Sa négation sera : « *il existe au moins un élément x de l'ensemble E qui ne possède pas la propriété P* ».
  Exemple : soit la proposition « *Tous les lecteurs de ce chapitre comprennent tout ce qui est écrit*» sa négation sera donc : « *Il existe au moins un lecteur qui ne comprend pas ce chapitre* ».
  Remarque : Pour démontrer qu'une proposition universelle n'est pas vraie,

il suffit donc de trouver un seul x qui ne vérifie pas la proposition P. C'est ce qu'on nomme un « *contre-exemple* ».

— **Négation d'une proposition existentielle :**Une proposition existentielle s'énonce : « *Il existe au moins un élément x de l'ensemble E qui possède la propriété P* ». Sa négation sera : « *Pour tout élément x de l'ensemble E, x ne vérifie pas P* ».
  Exemple : Soit la proposition P : $\exists x \in R, x^2 = -1$. Cette proposition est fausse car un carré ne peut être négatif. Par contre sa négation est vraie : $\forall x \in R, x^2 \neq -1$.

1.7 Théorie des ensembles

La théorie des ensembles se donne comme primitives les notions d'ensemble et d'appartenance, à partir desquelles elle reconstruit les objets usuels des mathématiques : fonctions, relations, etc. Dans ces sous-section, nous présenterons les notions d'ensemble, de sous-ensemble et de relation.

1.7.1 Ensemble

Un ensemble est une collection d'éléments que l'on peut énumérer ou définir par une propriété. Certains ensembles ont des notations particulières (ex. N, Z, D, Q, R). Lorsqu'on énumère les éléments d'un ensemble, on dit que cet ensemble est défini par extension, lorsqu'on définit un ensemble par une propriété, on dit que cet ensemble est défini par compréhension. Un ensemble qui ne contient aucun élément s'appelle : l'ensemble vide noté $\emptyset$.

Exemple

N ensemble des entiers naturels, $N = \{1, 2, 3, ...\}$
Lorsque le nombre des éléments d'un ensemble devient trop important ou qu'il y a un nombre infini d'éléments, on ne peut le définir par compréhension tel que : $C = \{x \in N/1 \leqslant x \leqslant 49\}$.

1.7.2 Élément

Un ensemble est constitué d'éléments. On représente souvent un élément par une minuscule. On dit qu'un élément « a » appartient à un ensemble A. On écrit alors : $a \in A$. Le symbole $\in$ signifiant « *appartient à* » est initiale de « *élément* ».

1.7.3 Sous-ensemble

On dit qu'un ensemble A est un sous-ensemble de l'ensemble E si et seulement si tout élément de A est élément de E ou si $A = \emptyset$. On dit alors que A est inclus dans E. $A \subset E \Leftrightarrow \forall a \in A, a \in E$ ou $A = \emptyset$. Le symbole $\subset$ signifie « *inclus dans* ».

1.7.4 Complémentaire d'un ensemble

On appelle complémentaire de l'ensemble A dans l'ensemble E, l'ensemble noté $C_E(A)$ composé des éléments de E qui ne sont pas élément de A. Le complémentaire correspond au connecteur NON. On a alors : $a \in C_E(A) \Leftrightarrow a \in E$ et $a \notin A$. Le symbole $\notin$ signifie « *n'appartient pas à* ». Lorsque l'ensemble$E$ est implicite, on note le complémentaire de A : $\bar{A}$ qui se prononce « A *barre* ».

Exemple

Soit E l'ensemble des entiers naturels inférieurs ou égaux à 20 et soit A l'ensemble des nombres premiers inférieurs à 20. On a donc : $A = \{2, 3, 5, 7, 11, 13, 17, 19\}$.
L'ensemble $C_E(A)$ sera donc l'ensemble des entiers naturels inférieurs ou égaux à 20 qui ne sont pas premiers. On a donc : $C_E(A) = \{0, 1, 4, 6, 8, 9, 10, 12, 14, 15, 16, 18, 20\}$.

1.7.5 Intersection de deux ensembles

On appelle intersection de deux sous-ensembles A et B dans un ensemble E, l'ensemble noté : $A \cap B$ (A inter B) constitué des éléments communs à A et B. On a donc : $x \in A \cap B \Leftrightarrow x \in A$ et $x \in B$.

Exemple

Soit A l'ensemble des entiers naturels pairs inférieurs ou égaux à 20 et soit B l'ensemble des entiers naturels multiples de 3 inférieurs ou égaux à 20.
On a donc :

— $A = \{0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20\}$,
— $B = \{0, 3, 6, 9, 12, 15, 18\}$,
— $A \cap B = \{0, 6, 12, 18\}$ qui n'est autre que l'ensemble des entiers naturels multiples de 6 inférieurs ou égaux à 20.

Remarques

— Lorsque l'ensemble A est inclus dans l'ensemble B, on a alors : $A \subset B \Rightarrow A \cap B = A$
— Lorsque les ensembles A et B sont disjoints, ils ne possèdent aucun élément commun, leur intersection est donc vide, on a donc : $A \cap B = \emptyset$. Ce qui signifie que A et B sont disjoints, comme c'est le cas entre l'ensemble A et sont complémentaire $\bar{A}$, on a donc : $A \cap A = \emptyset$.

1.7.6 Union de deux ensembles

On appelle union de deux sous-ensembles A et B dans un ensemble E, l'ensemble noté : $A \cup B$ (A union B) constitué des éléments qui appartiennent à A ou à B (*éventuellement aux deux, le « ou » étant non exclusif*). On peut alors écrire : $x \in A \cup B \Leftrightarrow x \in A$ ou $x \in B$.

Exemple

Soit A l'ensemble des entiers naturels pairs inférieurs ou égaux à 20 et soit B l'ensemble des entiers naturels multiples de 3 inférieurs ou égaux à 20. On a donc :

— $A = \{0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20\}$,
— $B = \{0, 3, 6, 9, 12, 15, 18\}$,
— $A \cap B = \{0, 2, 3, 4, 6, 8, 9, 10, 12, 14, 15, 16, 18, 20\}$.

Remarques

1. Lorsque l'ensemble A est inclus dans l'ensemble B, on a alors : $A \subset B \Rightarrow A \cap B = B$,
2. L'union de l'ensemble A et de son complémentaire $\bar{A}$ donne l'ensemble E, c'est à dire : $A \cap A = E$.

1.8 Relations

Nous considérons dans ce cours deux types de relations citées dans ce qui suit.

1.8.1 Relation d'équivalence

Une relation est une relation d'équivalence si elle est :

- **Symétrique :** $\forall x \in E, \forall y \in E$, x R y $\Rightarrow$ y R x,
- **Réflexive :** $\forall x \in E, xRx$,
- **Transitive :** $\forall x \in E, \forall y \in E, \forall z \in E, (xRy)$ et $(yRz) \Rightarrow xRz$.

Remarques

1. Dans le cas d'une relation d'équivalence, deux éléments en relation sont aussi dits équivalents,
2. Si R est une relation d'équivalence et x est un élément de E, on appelle classe d'équivalence de x l'ensemble des éléments y de E tels que xRy.

1.8.2 Relation d'ordre

Une relation d'ordre est une relation réflexive, anti-symétrique, transitive.

- **Réflexive :** $\forall x \in E, xRx$,
- **Anti-symétrique :** $\forall x \in E, \forall y \in E, (xRyetyRx) \Rightarrow x = y$,
- **Transitive :** $\forall x \in E, \forall y \in E, \forall z \in E, (xRyetyRz) \Rightarrow xRz$.

Remarques

Si R est une relation d'ordre sur E, alors :

- on dit que l'ordre est total si on peut toujours comparer deux éléments de E : pour tous x, $y \in E$, on a xRy ou yRx. Dans le cas contraire, on dit que l'ordre est partiel.
- Si A est une partie de E et M est un élément de E, on dit que M est un majorant de A si, pour tout $x \in A$, on a xRM.

1.9 Ensembles ordonnés et Treillis

1.9.1 Ensembles ordonnés

Représentation graphique d'un ensemble ordonné

Un ensemble est représenté graphiquement que s'il est fini. Les éléments de l'ensemble E seront représentés par des points de sorte que si on a la relation aRb, alors a sera relié à b par une flèche orientée de a vers b (voir la figure 1.1).

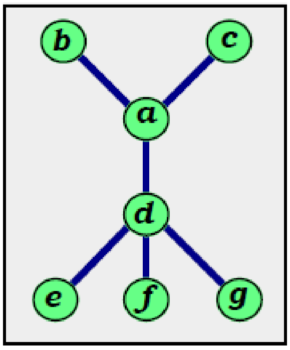

FIGURE 1.1: Exemple d'ensemble ordonné

Exemple

Soit R la relation «*est diviseur de*», $E = \{1, 2, 3, ..., 22, 26\}$ R est transitive (aRb et bRc alors aRc), représentez que les relations directes de type aRb (voir la solution à la figure 1.2).

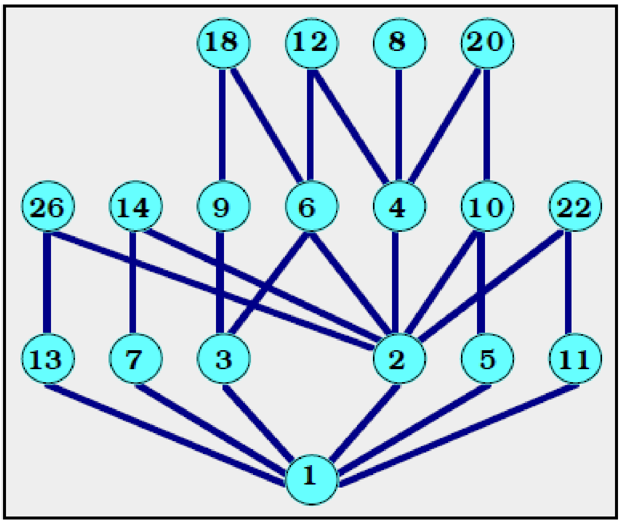

FIGURE 1.2: Solution de l'exemple d'ensemble ordonné

1.9.2 Les Treillis

Un treillis est un ensemble ordonné tel que pour tout couple d'éléments a et b, il existe un plus petit majorant et un plus petit minorant. Le *ppm* noté $a \vee b$ et le *pgm* sera noté $a \wedge b$. Les opérations $\vee$ et $\wedge$ possèdent les propriétés de : commutativité, associativité, idempotence et d'absorption.
L'absorption est définie par : $a \vee (a \wedge b) = a$ et $a \wedge (a \vee b) = a$.

1.10 Conclusion

Dans ce chapitre, nous avons présenté les notions de base de la logique mathématique qui vont nous servir dans la logique descriptive et la logique des prédicats. Nous aborderons dans le prochain chapitre la «*logique d'ordre 0*».

Chapitre 2

Logique propositionnelle (Ordre 0)

2.1 Introduction

Dans la logique propositionnelle, on étudie les relations entre des énoncés, que l'on va appeler propositions ou encore des formules. Ces relations peuvent être exprimées par l'intermédiaire de connecteurs logiques qui permettent, par composition, de construire des formules syntaxiquement correctes. On trouve principalement : la conjonction, la disjonction (inclusive), l'implication, l'équivalence et la négation.

Dans ce chapitre, nous présenterons la manière de construire une formule bien formée (fbf) et déterminer par la suite sa valeur de vérité.

2.2 Syntaxe (Formalisation)

S'intéresser à la syntaxe de la logique propositionnelle, c'est considérer les formules qui sont "*bien écrites*". Pour cela, on se donne un alphabet (langage que l'on notera Ls), i.e. un ensemble de symboles, avec :

— **Un ensemble** $V = \{p, q, r, ...\}$ **:** dénombrable de lettres appelées variables propositionnelles. Il s'agit des propositions atomiques telles que par exemple « 6 est divisible par 2 ».

— **Atomes :** Nous appellerons atomes ou variables propositionnelles ou propositions élémentaires des énoncés dont nous ne connaissons pas la structure interne, et qui gardent leur identité tout au long du calcul propositionnel qui nous occupe. L'ensemble des variables propositionnelles est noté $v(L)$. Elles sont écrites en minuscules $(p, q \dots)$.

— **Un ensemble (fini) de connecteurs logiques :** $\wedge, \vee, \neg, \longrightarrow, \equiv$.

— **Formules :** Nous dénoterons les formules par des lettres majuscules de l'alphabet latin ou grec $(A, B, ...$ ou $\phi \dots)$. L'ensemble des formules, noté $F(L)$, est défini par : les atomes sont des formules $(v(L) \subseteq F(L))$;

Si A et B sont des formules, alors $(A \longleftrightarrow B)$, $(A \longrightarrow B)$, $(A \wedge B)$, $(A \vee B)$ et $(\neg B)$ sont des formules.

Les parenthèses sont un moyen de lever l'ambiguïté. Il en existe un autre qui consiste à donner à chaque opérateur un ordre de priorité. Les connecteurs sont traditionnellement classés de la façon suivante (par priorité décroissante des connecteurs) : $\neg, \wedge, \vee, \longrightarrow, \longleftrightarrow$. Dans le cas où deux connecteurs ont même priorité, et en l'absence de parenthèses, l'associativité se fait de gauche à droite. On peut ainsi se permettre d'omettre les parenthèses. Par exemple, la formule $p \longrightarrow q \longleftrightarrow \neg r$ doit se lire $((p \to q) \longleftrightarrow (\neg r))$. A l'avenir, lorsque nous parlerons de formules bien formées, nous inclurons également les formules dont le parenthèsage est partiellement ou complètement implicite. L'ensemble des formules bien formées ainsi défini forme le langage de la logique propositionnelle.

Parmi les mots que l'on peut écrire avec cet alphabet, on va regarder ceux qui correspondent à des expressions logiques bien formées que l'on définit (inductivement) ainsi :

1. Toutes les propositions atomiques $p, q, r, ...$, sont des expressions bien formées ;

2. Si A est une expression bien formée, alors $\neg A$ est une expression bien formée ;

3. Si A et B sont deux expressions bien formées, alors $(A \wedge B)$, $(A \vee B)$, $(A \to B)$ et $(A \equiv B)$ sont des expressions bien formées ;

4. Il n'y a pas d'autres expressions bien formées que celles des règles précédentes. Par exemple $(((\neg p \equiv q) \vee \neg(r \wedge s)) \to q)$ est une expression bien formée, tandis que $p\neg qr \to t(\equiv$ ne l'est pas. Dans la suite, on ne considère que des expressions bien formées.

   Littéral : Un littéral est un atome (littéral positif) ou la négation d'un atome (littéral négatif).

   Clause : Une clause est une disjonction de littéraux $(p_1 \vee p_2 \vee ... \vee p_n)$, les littéraux pouvant être positifs ou négatifs.

2.3 Sémantique

S'intéresser à la sémantique de la logique propositionnelle, c'est déterminer la valeur de vérité d'un énoncé, c'est-à-dire d'une formule. On parle de l'interprétation d'une formule : il s'agit plus concrètement d'affecter une valeur vraie ou fausse à chacune des variables propositionnelles qui la compose. Pour une formule à n variables, il y a 2^n mondes possibles.

- **Valuation :** On appelle valuation, ou L-Modèle, d'un ensemble de variables propositionnelles $v \subseteq v(L)$, une fonction m de $v(L)$ dans $\{T, F\}$ $(m : v(L) \longrightarrow \{T, F\})$.
  - $v(\neg X) = \neg v(X)$,
  - $v(x \vee Y) = v(x) \vee v(Y)$,
  - $v(x \wedge Y) = v(x) \wedge v(Y)$,
  - $v(X \supset Y) = v(x) \supset v(Y)$.
- **Interprétation :** Une valuation appliquée à une formule dans laquelle est dite interprétation. soit la formule $F = (a \wedge b) \vee \neg b \longrightarrow \neg a$, $\{m(a) = T$, $m(b) = F\}$ est une interprétation de F.
- **Modèle d'une formule :** Une interprétation I est un modèle d'une formule A si elle est vraie (si elle vaut **T**).

2.3.1 Table de vérité

On peut tirer partie de ce que dans certains cas, la connaissance de la valeur de vérité d'une sous-formule peut permettre de déterminer la valeur de la formule. Par exemple si $v(B) = 1$, on sait sans connaître $v(A)$ que $v(A \longrightarrow B) = 1$. On peut donc éviter d'énumérer tous les cas (cependant il devient moins évident que tous les cas sont énumérés et cela doit être vérifié).

A	B	C	$A \wedge B$	$(A \wedge B) \longrightarrow C$	$B \longrightarrow C$	$A \longrightarrow (B \longrightarrow C)$
?	?	1	?	1	1	1
0	?	0	0	1	?	1
1	0	0	0	1	1	1
1	1	0	1	0	0	0

TABLE 2.1: Exemple de table de vérité réduite

On vérifie facilement que tous les cas sont bien énumérés (le " ?" désigne 0 ou 1). On a donc montré que $(A \wedge B) \longrightarrow C \equiv A \longrightarrow (B \longrightarrow C)$.

2.3.2 Validité et consistance

- **Formule valide :** Une formule valide, ou tautologie, est une formule Φ vraie quelles que soient les valeurs de vérité des atomes qui la composent (vraie dans toute interprétation). On la note $\models \Phi$.
- **Formule insatisfiable (falsifiable ou inconsistante) :** Une formule insatisfiable, ou sémantiquement inconsistante, ou encore antilogie, est une formule fausse dans toute interprétation.
- **Formules satisfiables (consistante) :** Une formule satisfiable ou sémantiquement consistante est une formule vraie dans au moins une interprétation.
- **Formules invalides :** Une formule invalide est fausse dans au moins une interprétation.
- **Formules contingentes :** Une formule contingente est vraie dans certaines interprétations et fausse dans d'autres.
- **Conséquence valide :** Soient deux formules A et B. Nous dirons que A est la conséquence valide de B (notée $A \models B$) si tout modèle de A est un modèle de B.

2.3.3 Équivalence des formules bien formées

Deux formules sont équivalentes quand elles ont une même valeur dans toutes interprétation (notation : $A \equiv B$).

Théorème : Règle de substitution uniforme

Soit la formule ϕ contenant les atomes $p_1, p_2, ..., p_n$. Soit la formule ϕ^* obtenue en substituant aux atomes $p_1, p_2, ..., p_n$ les formules $1, 2, ..., n$. Alors si $\models \phi$, on $a \models \phi^*$.

Propriétés des formules équivalentes

Soient A, B et C trois formules bien formées.

- $\models [(A \wedge B) \equiv (B \wedge A)]$ par commutativité
- $\models [(A \vee B) \equiv (B \vee A)]$ par commutativité
- $\models [(A \equiv B) \equiv (B \equiv A)]$ par commutativité

— $\models [((A \wedge B) \wedge C) \equiv ((A \wedge (B \wedge C))]$ par associativité
— $\models [((A \vee B) \vee C) \equiv ((A \vee (B \vee C))]$ par associativité
— $\models [((A \equiv B) \equiv C) \equiv ((A \equiv (B \equiv C))]$ par associativité
— $\models [((A \wedge B) \vee C) \equiv ((A \vee C) \wedge (B \vee C))]$ par distributivité
— $\models [((A \vee B) \wedge C) \equiv ((A \wedge C) \vee (B \wedge C))]$ par distributivité
— $\models [A \wedge (A \vee B)) \equiv A$ par absorption
— $\models (A \vee (A \wedge B)) \equiv A]$ par absorption
— $\models [(A \wedge A) \equiv A]$ par idempotence
— $\models [(A \vee A) \equiv A]$ par idempotence
— $\models [\neg(A \wedge B) \equiv (\neg A \vee \neg B)]$ par la lois de Morgan
— $\models [\neg(A \vee B) \equiv (\neg A \wedge \neg B)]$ par la lois de Morgan
— $\models [(A \longrightarrow B) \equiv (\neg A \wedge B)]$ par implication
— $\models [(A \longrightarrow B) \equiv \neg(A \wedge \neg B)]$ par implication
— $\models [(A \equiv B) \equiv ((A \longrightarrow B) \wedge (B \longrightarrow A))]$ par équivalence
— $\models [(A \equiv B) \equiv ((A \wedge B) \vee (\neg A \wedge \neg B))]$ par équivalence
— $\models [(A \longrightarrow B) \equiv (\neg B \longrightarrow \neg A)]$ par contraposition
— $\models [(A \longrightarrow (B \longrightarrow C)) \equiv ((A \longrightarrow B) \longrightarrow (A \longrightarrow C))]$ autodistributivité
— $\models [(A \longrightarrow (B \longrightarrow C)) \equiv (B \longrightarrow (A \longrightarrow C))]$ par import-export
— $\models [((A \longrightarrow B) \wedge (B \longrightarrow C)) \equiv (A \longrightarrow C)]$ par transitivité

2.3.4 Conséquence logique (satisfaction)

On dit que E est une **conséquence valide** de $A, B, \ldots$ si dans la table de vérité de E est vraie à chaque fois que $A, B, \ldots$ sont vraies.
Notée : $A, B, \ldots \models E$.
Pour vérifier que E est une conséquence valide, il faut :
— Trouver tous les modèles de E,
— Pour chaque modèle M de E, vérifier que $M(f) = v$;
— Si E utilise n propositions atomiques, le nombre de modèles potentiels est 2^n.

Remarques

1. Le temps de vérification devient exponentiel avec le nombre de variables.
2. Si il existe une interprétation où $v(A) = T$, $v(B) = T$,... et $v(E) = F$ alors $A, B, \ldots \not\models E$.

Exemple

Vérifier à l'aide de la table de vérité la conséquence valide suivante :
$A \supset B, B \supset C \models A \supset C$.

A	B	C	$A \supset B$	$B \supset C$	$B \supset C$
T	T	T	**T**	**T**	**T**
T	T	F	T	F	F
T	F	T	F	T	T
T	F	F	F	T	F
F	T	T	**T**	**T**	**T**
F	T	F	T	F	T
F	F	T	**T**	**T**	**T**
F	F	F	**T**	**T**	**T**

TABLE 2.2: Exemple de la conséquence valide

Donc $A \supset C$ est une conséquence valide de $A \supset B$ et $B \supset C$.

2.3.5 Formes normales (Normalisation)

Toute formule peut se réécrire sous la forme :

$$C_1 \wedge C_2 \wedge \ldots \wedge C_n$$

Où chaque C_i (appelé clause) est elle-même de la forme :

$$p_1 \vee p_2 \vee \ldots \vee p_m \vee \neg q_1 \vee \neg q_2 \vee \ldots \vee \neg q_r$$

Où les p_j et q_k sont des atomes.
Cette forme est appelée forme normale conjonctive. Il y a un algorithme pour mettre en FNC.
La FNC est nécessaire pour pouvoir appliquer certains algorithmes.

Exemple

Transformer en forme normale conjonctive la formule donnée ci-dessous.

$$((b \vee c) \implies a \vee d$$

1. Appliquer l'équivalence double-implique et implique-ou pour supprimer les $\implies$ et $\iff$. La formule devient donc : $(\neg(b \vee c) \vee a) \vee d$.
2. Appliquer De Morgan pour "descendre" les négations près des variables. La formule devient : $((\neg b \wedge \neg c) \vee a) \vee d$.
3. Appliquer la distributivité pour descendre les $\vee$ et remonter les $\wedge$, donc $((\neg b \vee a) \wedge (\neg c \vee a)) \vee d \equiv ((\neg b \vee a) \vee d) \wedge ((\neg c \vee a)) \vee d)$.
4. Appliquer l'associativité des $\vee$ et $\wedge \equiv (\neg b \vee a \vee d) \wedge (\neg c \vee a \vee d)$.

Il existe aussi une forme normale disjonctive ϕ^* :

$$D_1 \vee D_2 \vee ... \vee D_m$$

Où chaque D_i est lui-même de la forme :

$$r_1 \wedge r_2 \wedge ... \wedge r_u \wedge \neg s_1 \wedge \neg s_2 \wedge ... \wedge \neg s_v$$

Où les r_j et s_k sont des atomes.

2.4 Théorie de la preuve (démonstration)

Dans le chapitre précédent, nous avons vu la méthode des tables de vérité pour établir la validité et la consistance d'une formule. Cependant, cette méthode réclame, pour une formule contenant n atomes distincts, le calcul d'une table comprenant 2^n lignes. Il faut donc essayer de trouver des méthodes plus efficaces. En conséquence une autre approche, dite déductive, basée sur la formalisation de la théorie est proposée.

L'approche déductive a pour objet de calculer les conséquences logiques par l'application des règles d'inférence. Pour cela on construit des systèmes formels d'inférence composés d'axiomes (formules) et de règles d'inférence.
Il existe plusieurs systèmes d'inférence de ce type pour la logique des propositions : Système de Hilbert, système de déduction naturelle, principe de résolution de Robinson, etc.
Nous verrons dans ce chapitre le système de déduction naturelle ainsi que le principe de résolution de Robinson.

Pour spécifier notre système du calcul propositionnel formel nous avons besoin d'avoir :

1. Un alphabet (l'ensemble des symboles utilisés),
2. Un ensemble de suites finies de ces symboles qui sont appelées formules bien formées. Ce sont les phrases du langage formel,
3. Un ensemble fini de formules appelées axiomes,
4. Un ensemble fini de règles : les règles de déduction (règles d'inférence).

Définition

Le système formel du calcul propositionnel (CPF) est défini par :

1. L'alphabet est composé de :
   (a) $a, b, ..., z$ variables propositionnelles,
   (b) $\neg, \rightarrow$ Symboles primitifs,
   (c) (,) parenthèses.
2. L'ensemble des formules bien formées est défini par induction comme suit :
   (a) Chaque variable propositionnelle est une formule bien formée,
   (b) Si A et B sont des formules bien formées alors, $A \rightarrow B$ et $\neg B$ sont des formules bien formées,
   (c) L'ensemble des formules bien formées est engendré par les clauses a) et b) ci-dessus.

3. Axiomes : Pour toutes formules bien formées (fbfs) A, B et C les formules suivantes sont des axiomes du système formel CPF.
   (a) $(A \to (B \to A))$... (axiome 1),
   (b) $(A \to (B \to C)) \to ((A \to B) \to (A \to C))$...(axiome 2),
   (c) $(\neg B \to \neg A) \to (A \to B)$...(axiome 3).
4. Règles d'inférence appelée modus ponens (MP)$(A \to B, A) \to B$ ou $\frac{A, A \to B}{B}$.
   En d'autres termes cette règle signifie : B est une conséquence directe de A et $(A \to B)$ où A et B sont des formules bien formées (fbfs).
5. Règle appelée modus tollens (MT) : $(A \to B, \neg B) \to \neg A$ ou $\frac{A \to B, \neg B}{\neg A}$
6. Règle de substitution (Sub) : $\frac{B}{[A = p]B}$
   La règle de substitution signifie que à partir d'une formule B contenant une variable p, on peut déduire la formule $[A/p]$ B obtenu en remplaçant toutes les occurrences de p dans B par une formule A.

Remarques

— Les deux symboles $\neg$ et $\to$ ont été introduits sans aucune définition. Ici ils n'ont aucun sens précis. Ce sont de simples marques (des symboles primitifs) intervenant dans la construction des formules.

— La règle Modus Ponens est la seule règle du système qui nous permet d'obtenir, à partir des axiomes, d'autres formules appelées conclusion de déduction.

Important

Un théorème est une affirmation (mathématique ou logique) qui peut être démontrée, c'est-à-dire une assertion qui peut être établie comme vraie au travers d'un raisonnement logique construit à partir des axiomes.

2.4.1 Système de déduction naturelle

Dans ce système, il n'y a pas d'axiomes, uniquement des règles d'inférence. On notera : $\frac{f1, f2, \ldots fn}{g}$

Le fait qu'on peut produire g à partir de $f_1, ..., f_n$ (inférence directe). On note :

$f \rightarrow \ldots \rightarrow g$ représente qu'on peut produire g à partir de f par une succession d'inférences. Dans tout système d'inférence on peut utiliser toutes les équivalences connues pour produire de nouvelles formules. Si on a $f \equiv g$, on a forcément $f \models g$.

Exemple

À partir de la formule suivante $\neg(p \vee q)$ nous pouvons déduire $\neg p \wedge \neg q$.

2.4.1.1 Règles d'inférence

2.4.1.2 Complétude et consistance

Idéalement un système formel doit satisfaire les deux critères de consistance et de complétude. Ces deux notions font le lien entre syntaxe et sémantique.

On se rappelle qu'un théorème est une formule qu'on peut produire par application des règles à partir des axiomes. La «*théorèmité*» d'une formule est donc une notion syntaxique. Par contre la notion de conséquence logique est sémantique, elle est définie en dehors de tout système d'inférence formel. C'est cette notion qui nous servira à vérifier la consistance et la complétude d'un système d'inférence.

Définition

Un système d'inférence est consistant si tous les théorèmes sont des conséquences logiques des formules des axiomes (tout ce qui est démontrable est «vrai »). Ou de manière équivalence, si on peut produire g à partir de l'ensemble de formules F, noté $F \vdash g$, alors g est une conséquence de F ($F \models g$).
Un système d'inférence est complet s'il permet de produire toutes les conséquences logiques des axiomes, c'est-à-dire si toute conséquence des axiomes est un théorème (tout ce qui est vrai est démontrable). Ou, de manière équivalente, $F \models g$ entraîne $F \vdash g$. Le système d'inférence naturel est consistant et complet.

2.4.2 Principe de résolution de Robinson

Le principe de résolution est formé d'une unique règle d'inférence. Sa grande simplicité de mise en œuvre en fait une méthode très utilisée.

Théorème

Toute formule est équivalente à une conjonction de clauses. On notera $C(F)$ un ensemble de clauses dont la conjonction est équivalente à F.

Exemple

$$C(p \longleftrightarrow q) = \neg p \vee q, p \vee \neg q$$

Une interprétation satisfait F si seulement si elle satisfait $C(F)$. En particulier $C(F)$ est contradictoire si seulement si F est une antilogie.

2.4.2.1 Règle de résolution

Si u est un littéral d'une clause C, on notera C/u la clause dont tous les littéraux sont ceux de C sauf u.

Exemples

1. Soit la formule suivante : $(p \vee q \vee r)\ q = p \vee r$
   La clause C est une résolvante des clauses C_1 et C_2 si il existe un littéral u tel que :

   — U est un littéral de C_1,
   — $\neg u$ est un littéral de C_2,
   — $C = (C_1/u) \vee (C_2/\neg u)$.
2. $p \vee r$ et $q \vee \neg r$ donnent la résolvante $p \vee q$,
3. $p \vee q \vee r$ et $\neg r \vee s \vee \neg t \vee q$ donnent $p \vee q \vee s \vee \neg t$,
4. $p \vee q \vee r$ et $\neg p \vee \neg q \vee s$ donnent $q \vee r \vee \neg q \vee s$ et $p \vee r \vee \neg p \vee s$, **attention elles ne donnent pas** $r \vee s$.
5. $\neg p \vee q$ et p donnent q,
6. $\neg p \vee p$ donnent la clause vide notée $\square$.

Une preuve par résolution de la clause F à partir de l'ensemble de clauses C est une suite finie de clauses $C_1, C_2, \ldots C_n$ telles que $C_i \in C$ ou bien C_i est une résolvante

de deux C_j précédents, et $Cn = F$.
Notation : $C \vdash F$.

Exemple

$p, \neg p \vee q, \neg q \vee r \vdash r$ en prenant :

- $C_1 = p, C_2 = \neg p \vee q$, dans l'ensemble de départ,
- $C_3 = q$ est une résolvante de C_1 et C_2.
- $C_4 = \neg q \vee r$, dans l'ensemble de départ,
- $C_2 = r$ est une résolvante de C_3 et C_4.

Une réfutation d'un ensemble de clauses C est une preuve de la clause vide à partir de C.
Notation : $C \vdash \Box$.

Exemple de réfutation

De l'ensemble de clauses :

1. $\neg p \vee \neg q \vee r$,
2. $\neg p \vee q$,
3. P,
4. $\neg r$. On obtient successivement :
5. $\neg p \vee \neg q$ (1+4)
6. Q (2+3)
7. $\neg p$ (5+6)
8. $\Box$ (3+7)

2.4.2.2 Validité de la règle de résolution

Propriété

La règle de résolution est saine (Sound), c'est-à-dire toute résolvante est une conséquence sémantique des clauses dont elle est issue.

Corollaire 1

Toute clause F déduite d'un ensemble C de clauses par résolution est une conséquence sémantique de C si $C \vdash F$ alors $C \models F$.

Corollaire 2

Un ensemble de clauses admettant une réfutation si $C \vdash \square$ alors $C \models F$.

2.4.2.3 Complétude de la méthode de résolution

Théorème

La méthode de résolution est complète pour la réfutation, c'est-à-dire, si C est un ensemble de clauses contradictoire, alors il existe une réfutation de C.

Remarques

- Elle n'est pas complète pour la déduction.
- La méthode est non déterministe, le théorème dit que la réfutation existe, il ne dit pas comment la trouver, ni si la recherche se termine (voir la figure 2.1).

$$\begin{array}{ccc}
C \vdash F & \Rightarrow & C \vDash F \\
\Downarrow & & \Updownarrow \\
C \cup C(\neg F)\square & \Leftrightarrow & C \cup C(\neg F) \vDash f
\end{array}$$

Figure 2.1: Complétude de la méthode de résolution

2.4.2.4 Quelques stratégies

Résolution positive

On n'autorise la résolution qu'entre deux clauses dont l'une est positive (n'a aucun littéral négatif).

Remarque

Il y a au moins une clause positive dans un ensemble contradictoire. (Sinon l'ensemble serait satisfait par l'interprétation toujours fausse).

Propriété

La résolution positive est complète pour la réfutation.

Exemple

de l'ensemble des clauses :

1. $\neg p \vee \neg q \vee r$
2. $\neg p \vee q$
3. p
4. $\neg r$
   On obtient successivement
5. $\neg q \vee r$ *(1 + 3)*
6. q *(2 + 3)*
7. r *(5 + 6)*
8. $\square$ *(4 + 7)*

Résolution négative

On n'autorise la résolution qu'entre deux clauses dont l'une est négative (n'a aucun littéral positif).

Remarque

Il y a au moins une clause négative dans un ensemble contradictoire. (Sinon l'ensemble serait satisfait par l'interprétation toujours vraie).

Propriété

La résolution négative est complète pour la réfutation.

Exemple

de l'ensemble des clauses :

1. $\neg p \vee \neg q \vee r$
2. $\neg p \vee q$
3. p
4. $\neg r$

   On obtient successivement
5. $\neg p \vee \neg q$ *(1 + 4)*
6. $\neg p$ *(2 + 5)*
7. $\square$ *(3 + 6)*

Résolution linéaire et résolution linéaire par entrée

Une preuve par résolution linéaire d'une clause F à partir d'un ensemble C de clauses est une suite $C_0, C_1, ...C_n$ telle que $C_0 \in C$ et pour tout i, C_i est une résolvante de C_{i-1} et d'une clause de C ou d'un C_j précédent.
La preuve est linéaire par entrée si on prend toujours une clause C (clause d'entrée).

Exemple

de l'ensemble des clauses :

1. $\neg p \vee \neg q \vee r$
2. $\neg p \vee q$
3. p
4. $\neg r$

   On obtient successivement
5. q *(2 + 3)*
6. $\neg p \vee r$ *(5 + 1)*

7. *r (6 + 3)*
8. $\square$ *(7 + 4)*

Propriété

La résolution linéaire est complète pour la réfutation. La résolution linéaire par entrée n'est pas complète. La méthode de résolution est, dans le cas général, inefficace. Il existe cependant des cas particuliers où cette méthode est rapide.

Nous allons nous intéresser dans ce qui suit à un type particulier de forme normale, les formes normales composées de clause de Horn.

Propriété

La résolution linéaire par entrée est complète pour la réfutation des clauses de Horn.

Définition de la clause de Horn

Une clause de Horn est une clause comportant au plus un littéral positif. Il existe donc trois types de clauses de Horn :

- Celles comportant un littéral positif et au moins un littéral négatif, appelées clauses de Horn strictes ;
- Celles comportant un littéral positif et aucun littéral négatif, appelées clauses de Horn positives ;
- Celles ne comportant que des littéraux négatifs, appelées clauses de Horn négatives (et dont fait partie la clause vide ϕ).

Algorithme de résolution adapté aux clauses de Horn

L'algorithme présenté ci-dessous constitue une amélioration de l'algorithme de résolution appliqué à une forme normale N ne contenant que des clauses de Horn.

1. Si ϕ est dans N, l'ensemble est inconsistant et la résolution est terminée ;
2. Sinon, choisir une clause C et une clause P telles que P soit une clause de Horn positive réduite à p et C une clause contenant $\neg p$;

3. Calculer la résolvante R de P et C ;
4. Reprendre l'algorithme en remplaçant N par $(N \setminus \{C\}) \bigcup R$.

Cet algorithme peut se terminer de deux façons :

1. Nous avons abouti à $\emptyset$ et l'ensemble N est alors inconsistant ;
2. Il est impossible de poursuivre, car nous ne parvenons plus à trouver P et C vérifiant les conditions requises ; la formule est alors consistante et le modèle composé de chacune des clauses positives restantes (ou plus exactement le modèle composé de chacun des atomes de ces clauses), lorsque l'algorithme se termine, est un modèle de N (on appelle d'ailleurs ce modèle le modèle canonique de N).

Exemple

Nous allons utiliser cette méthode pour démontrer l'inconsistance de :

$$\{\neg p \vee r, \neg r \vee s, p, \neg s\}$$

1. Par sélection de p et $\neg p \vee r$, l'ensemble se réduit à : $\{r, \neg r \vee s, p, \neg s\}$.
2. Par sélection de r et $\neg r \vee s$, nous obtenons : $\{r, s, p, \neg s\}$.
3. Enfin, par sélection de s et $\neg s$: $\{r, s, p, \emptyset\}$.

L'ensemble de clauses est bien inconsistant.

Utilisation des clauses de Horn

— Les clauses de Horn positives, par exemple p, sont appelées faits. Il s'agit en effet de l'énoncé de la vérité logique d'un atome.

— Les clauses de Horn strictes $q \vee \neg p_1 \vee ... \vee \neg p_n$ sont équivalentes à $p_1 ..., p_n \models q$ et représentent des règles du type *si. . . alors. . . .* Elles permettent de déduire de nouveaux faits à partir de faits existants.

— Les clauses négatives peuvent se comprendre comme des buts à atteindre. Considérons que nous souhaitions prouver $\{H_1, ..., H_n\} \models (p \wedge q \wedge r)$. La partie $(p \wedge q \wedge r)$ est le but de notre résolution. En appliquant une technique d'inconsistance nous sommes ramenés à $\{H_1, ..., H_n, (\neg p \vee \neg q \vee \neg r)\} \models \emptyset$. La clause $(\neg p \vee \neg q \vee \neg r)$ est une clause de Horn négative qui modélise donc bien le but à atteindre.

Exemple

Soit $F = (p \to (q \to r)) \to ((p \to q) \to (p \to r))$ alors
$\neg F \equiv (\neg p \vee \neg q \vee r) \wedge (\neg p \vee q) \wedge p \wedge \neg r$
$C(\neg F) = \{\neg p \vee \neg q \vee r, \neg p \vee q, p, \neg r\}$
$C(\neg F) \vdash \Box$
Donc F est une tautologie

2.5 Conclusion

Dans ce chapitre, nous avons présenté la logique propositionnelle définie autant que logique sans quantificateurs qui s'intéresse uniquement aux opérations logiques : la négation, la conjonction, la disjonction, l'implication et l'équivalence. Elle permet de construire des raisonnements à partir de ces connecteurs. Nous aborderons dans le prochain chapitre la logique descriptive ou souvent appelée «*logique d'ordre 1*».

Chapitre 3

Logique des prédicats (ordre 1)

3.1 Introduction

Le calcul des prédicats est considéré comme une extension du calcul propositionnel qui donne la possibilité d'introduire en même temps que les variables propositionnelles d'autres variables appartenant à un domaine arbitraire (ensemble d'entiers, de réels ou d'objets quelconques). Cette extension est obtenue grâce à l'introduction des deux quantificateurs $\forall$ et $\exists$.

L'objectif de ce chapitre est de définir la logique du premier ordre. Nous parlerons d'abord de la syntaxe, c'est à-dire comment écrire les formules, puis de leur sémantique.

3.2 Définitions

3.2.1 Prédicat

Un prédicat est une propriété ou relation qui porte sur un ou plusieurs éléments d'un domaine D. C'est une fonction de D dans T, F.
Prenons le problème suivant :
- Tout homme est mortel.
- Socrate est un homme.

— Donc, Socrate est mortel.

Nous avons déjà traduit des énoncés en logique des propositions. Supposons la traduction suivante :

— Tout homme est mortel est traduit par la proposition a.

— Socrate est un homme est traduit par la proposition b.

— Donc, Socrate est mortel est traduit par la proposition c.

Le problème s'écrit alors, en logique des propositions, $a \wedge b \rightarrow c$. Cette traduction est correcte. Mais elle est de piètre qualité.

Ce problème peut être traduit de manière adéquate en logique des prédicats :

— Hypothèse 1 : Quelque soit x appartenant au domaine D, si x est un homme, alors x est mortel ($\forall x(H(x) \rightarrow M(x))$) ;

— Hypothèse 2 : Socrate à la propriété d'être un homme ($H(Socrate)$) ;

— Conclusion : Socrate à la propriété d'être mortel ($M(Socrate)$).

Dans cet exemple, nous avons utilisé deux prédicats (H et M), une constante (Socrate), une variable (x) et le quantificateur universel ($\forall$). Comme nous l'expliquons plus tard, ce raisonnement, ainsi formalisé en logique des prédicats, sera valide.

3.2.2 Formule

Avant de définir le langage du calcul des prédicats, l'ensemble des formules bien formées (correctement écrites) nous allons d'abord préciser :

1. **L'alphabet** du langage du calcul des prédicats est composé de :
   — $x1, x2$: les variables
   — $a1, a2$: les constantes
   — $P1, P2$: les prédicats
   — $f1, f2$: les fonctions
   — $\neg, \rightarrow$: les connecteurs
   — $\forall, \exists$: les quantificateurs
2. **Les termes** du calcul des prédicats ($CP.$) sont définis comme suit :
   (a) Les variables et les constantes sont des termes.
   (b) Si fi est une fonction à n arguments du $CP.$ et $t_1, t_2, \dots t_n$ sont des termes alors $f_i(t_1, t_2, \dots t_n)$ est un terme du $CP.$
   (c) L'ensemble des termes est engendré par les clauses (a) et (b).

3.2.3 Formule atomique

Une formule atomique du CP. est définie par : Si P_i est un prédicat, une fonction définie d'un domaine D vers l'ensemble $\{0; 1\}$ et $t_1, t_2, \ldots t_n$ sont des termes alors $P_i(t_1; t_2; \ldots t_n)$ est une formule atomique.

3.2.4 Formule bien formée

Les formules bien formées (ou formule tout simplement) sont définies par :

1. Chaque formule atomique est une formule bien formée.
2. Si A, B sont des formules bien formées du CP. alors $\neg A$, $(A \rightarrow B)$, $(\forall x)A$, et $(\exists x)$ A sont des formules bien formées.
3. L'ensemble de toutes les formules bien formées est engendré par les clauses (1) et (2).

Exemple

Les formules suivantes sont des formules bien formées :

— $F \equiv \forall x A(x) \rightarrow (\forall y((P(f(x); 0) \rightarrow (P(g(y); b)))$

— $G \equiv (P(f(a); b)) \rightarrow \forall x((\neg P(x; 0)) \wedge P(f(x); g(x; f(x))))$

Parenthèsage

De la même manière qu'en logique des propositions, nous accepterons les formules dont le parenthèsage est partiellement ou complètement implicite. Pour cela, les connecteurs et les quantificateurs sont traditionnellement classés de la façon suivante (par priorité décroissante des connecteurs et quantificateurs) : $\forall, \exists, \neg, \wedge, \vee, \rightarrow, \leftrightarrow$. Dans le cas où deux connecteurs ont même priorité, et en l'absence de parenthèse, l'associativité se fait de gauche à droite. Ainsi, $\neg \forall x A \vee B$ doit se lire $((\neg(\forall x A)) \vee B)$.

3.2.5 Variables liées, variables libres

Dans la formule $\forall x A$ la formule A est dite champ du quantificateur $\forall$. La variable x est dite variable quantifiée par le quantificateur universel $\forall$. Les positions occupées par la variable x dans la formule A sont appelées occurrences de x. Dans la formule précédente F la variable x possède deux occurrences.

Définition 1

Une occurrence d'une variable x dans une formule est dite liée si elle possède une occurrence dans le champ d'un quantificateur $\forall$ (ou $\exists$) dans cette formule. Si une occurrence d'une variable n'est pas liée, elle est libre.

Exemples

1. Soit $F \equiv \forall x A(x) \rightarrow (\forall y((P(f(x), 0) \rightarrow P(g(y), b)))$. La première occurrence de la variable x est liée par le quantificateur $\forall x$, mais la deuxième occurrence de la variable x est libre.
2. $\forall x, \exists y P(x, y) \supset Q(z, y), \forall x, \exists y P(x, y) \supset Q(z, y)$ la portée de $\exists y$, $x, \exists y P(x, y) \supset Q(z, y)$ la portée de $\forall x$. x est liée, y est liée, z est libre. (Une variable est liée à un quantificateur si elle se trouve dans sa portée).

Définition 2

Soit A une formule du calcul des prédicats. Un terme t est libre pour la variable x dans A si aucune occurrence libre de x dans A n'appartient à un champ d'un quantificateur $(\forall y)$ où y est une variable dans t.

Exemple

Le terme $f(x, y)$ n'est pas libre pour y dans la formule : $\forall x P(x, y) \rightarrow \forall z Q(z, x)$.

Remarques

— Un terme t est dit clos s'il ne contient pas de variables.
— Une formule A est dite close si elle ne contient pas de variables libres.

3.3 Théorie des modèles

Comme en calcul propositionnel, nous allons nous efforcer de construire un modèle permettant de dégager une interprétation sémantique de nos formules.
Cependant, en calcul des prédicats, il n'est pas possible d'appliquer une méthode des tables de vérité directement dérivée du calcul propositionnel, en raison des domaines de valeur des variables de chacun des prédicats. En fait, pour une formule atomique $p(x_1, ..., x_n)$, nous allons avoir besoin d'une fonction (appelée fonction d'interprétation) chargée de donner un sens au symbole p, et donc de calculer sa valeur de vérité selon la valeur des $x_1, ..., x_n$.

3.3.1 Interprétation

On attribue un *sens* (une valeur de vérité) à chacune des formules en interprétant les différents symboles (fonctions, prédicats), les constantes et les variables.

Exemple

Soient $x, y \in N$, $\forall x \exists y (x \preceq y)$. Dans ce cas, on peut voir si cette formule est une proposition vraie ou une proposition fausse.

3.3.1.1 Interprétation des termes

Soit une formule F et I une interprétation de cette formule. On peut étendre l'interprétation I aux termes de F :
- à chaque symbole de constante, on associe sa valeur selon I ;
- à chaque variable, on associe la variable elle-même ;
- à chaque terme $f(t_1, ..., t_n)$, on associe le terme $f'(t'_1, ..., t'_n)$ où $t'_1, ..., t'_n$ sont les interprétations des $t_1, ..., t_n$ et f' est l'interprétation de f.

3.3.1.2 Interprétation des formules

Elle est définie par :
- Si F est un atome $p(t_1, ..., t_n)$, $I(F)$ est la fonction $p\prime(t\prime_1, ..., t\prime_n)$ où $p\prime$ est l'interprétation de p et où chaque $t\prime_i$ est l'interprétation de t_i ;

— Si F est de la forme $(\neg G)$, $(G \rightarrow H)$, $(G \leftrightarrow H)$, $(G \wedge H)$, $(G \vee H)$, $I(F)$ est définie par les mêmes lois fonctionnelles que celles définies pour le calcul propositionnel ;
— Si F est de la forme $\forall x G(x, y_1, ..., y_n)$, pour tout n-uplet $(a_1, ..., a_n) \in D_i$, $I(F)(a_1, ..., a_n) = T$ si pour toute valeur $a \in D$, $I(G)(a, a_1, ..., a_n) = T$, $I(F)(a_1, ..., a_n) = F$ sinon ;
— si F est de la forme $\exists x G(x, y_1, ..., y_n)$, pour tout n-uplet $(a_1, ..., a_n) \in D_i$, $I(F)(a_1, ..., a_n) = T$ s'il existe une valeur $a \in D$, $I(G)(a, a_1, ..., a_n) = T$, $I(F)(a, a_1, ..., a_n) = F$ sinon ;

Définitions

— Une formule A est vraie dans une interprétation I si $v(A) = T$ dans I.
— Une formule A est dite satisfaite s'il existe une interprétation I telle que A est vraie dans I.
— Une formule du calcul des prédicats est dite valide si elle est vraie pour toute interprétation. Elle est dite non valide s'il existe au moins une interprétation pour laquelle la formule n'est pas vraie (fausse) ou possède un contre-modèle.
— Une formule du calcul des prédicats est dite insatisfaisable si elle est fausse pour toute interprétation.

Notation : $I \models A$ signifie que A est vraie dans I.

3.3.2 Validité et consistance

Définitions

— **Formule valide :** Une formule F est dite valide ssi, pour toute interprétation I, on a $I(F) = T$.
— **Formule satisfiable :** Une formule A sera dite satisfiable, ou sémantiquement consistante, s'il existe une interprétation I telle que $I(A) = T$. L'interprétation est alors un modèle de A.
— **Formule invalide :** Une formule invalide est fausse dans au moins une interprétation.

— **Formule insatisfiable :** Une formule insatisfiable, ou sémantiquement inconsistante, ou encore anti-tautologie, est une formule fausse dans toute interprétation.
— **Formule contingente :** Une formule contingente est vraie dans certaines interprétations et fausse dans d'autres.
— **Conséquence logique :** Soit une formule B et une famille de n formules A_i ; On dit que B est conséquence des A_i si pour toute interprétation I telle que $\forall A_i, I(A_i) = T$, on a aussi $I(B) = T$. On note alors $A_1, A_2, ..., A_n \models B$.

3.3.3 Équivalence des formules bien formées

3.3.3.1 Formules équivalentes

Deux formules sont équivalentes quand elles ont même valeur dans toutes interprétation (notation : $A \equiv B$). Soient $A(x)$ et $B(x)$ deux formules atomiques bien formées. Les formules équivalentes de la logique des propositions demeurent équivalentes en logique des prédicats.
— $\forall x A(x) \wedge \forall x B(x) \equiv \forall x(A(x) \wedge B(x))$,
— $\exists x A(x) \vee \exists x B(x) \equiv \exists(A(x) \vee B(x))$,
— $\neg(\forall x A(x)) \equiv \exists \neg A(x)$,
— $\neg(\exists x A(x)) \equiv \forall x \neg A(x)$.

Remarques

— $\forall x A(x) \vee \forall x B(x) \not\equiv \forall x(A(x) \vee B(x))$,
— $\exists x A(x) \wedge \exists x B(x) \not\equiv \exists x(A(x) \wedge B(x))$.

3.3.3.2 Forme prénexe

Elle consiste à **accumuler les quantificateurs au début de la formule**. L'utilité de cette forme est qu'elle permet de mettre en évidence certaines relations logiques qui ne sont pas évidentes à voir sous les formes habituelles des formules.
— Une matrice est une formule du calcul des prédicats ne contenant aucun quantificateur.
— Toute formule admet une forme prénexe équivalente.

— Une forme prénexe est une formule du calcul des prédicats de la forme $Q_1x_1...Q_nx_nM$, où Q désigne $\exists$ ou $\forall$ et M est une matrice.

Théorème

Toute formule du calcul des prédicats peut être transformée en une formule équivalente sous forme normale prénexe conjonctive.

Définition

Une formule est dite sous forme normale prénexe conjonctive si elle est de la forme :

$$(Qv_1)(Qv_2)...(Qv_n)[A_{11} \vee A_{12}... \vee A_{1n}] \vee ... \vee [A_{m1} \wedge A_{m2} \vee ... \vee A_{mq}]$$

Où Q est soit le quantificateur $\forall$ ou soit le quantificateur $\exists$, les v_i sont des variables distinctes qui ont une occurrence libre dans les A_{ij}. Chaque A_{ij} est une formule atomique ou une négation d'une formule atomique.

Algorithme de construction

Pour construire la forme prénexe, il faut :

1. Supprimer les connecteurs d'équivalence et d'implication.
2. Renommer certaines variables liées de manière à n'avoir plus de variable quantifiée deux fois.
3. Supprimer les quantificateurs inutiles (dont la variable quantifiée n'apparaît pas dans leur portée) s'il y en a.
4. Transférer toutes les occurrences de la négation devant les atomes en utilisant les règles de réécriture vues pour la logique des propositions et les règles supplémentaires suivantes :
   — $\neg\forall xA \equiv \exists x\neg A$;
   — $\neg\exists xA \equiv \forall x\neg A$.
5. Faire passer les quantificateurs en tête en utilisant les règles de réécriture suivantes (et en utilisant l'associativité, la commutativité ou le renommage de variable si nécessaire) :

— $(\forall x A \wedge B) = \forall x(A \wedge B)$ si B ne contient pas x ;
— $(\exists x A \wedge B) = \exists x(A \wedge B)$ si B ne contient pas x ;
— $(\forall x A \vee B) = \forall x(A \vee B)$ si B ne contient pas x ;
— $(\exists x A \vee B) = \exists x(A \vee B)$ si B ne contient pas x.

Exemple 01

Transformer la formule suivante sous forme prénexe :

$$\forall x p(x) \wedge \exists y q(y) \rightarrow \exists y(p(y) \wedge q(y))$$

1. $\neg(\forall x p(x) \wedge \exists y q(y)) \vee \exists y(p(y) \wedge q(y))$ par suppression de $\rightarrow$;
2. $\neg(\forall x p(x) \wedge \exists y q(y)) \vee \exists z(p(z) \wedge q(z))$ par renommage des variables ;
3. $(\exists x \neg p(x) \vee \forall y \neg q(y)) \vee \exists z(p(z) \wedge q(z))$ par transfert de la négation ;
4. $\exists x \forall y \exists z(\neg p(x) \vee \neg q(y) \vee (p(z) \wedge q(z)))$ par déplacement des quantificateurs.

Exemple 02

Transformer la formule suivante en forme prénexe conjonctive :

$$\forall x[(\forall y p(x) \vee \forall z q(z;y)) \rightarrow \neg \forall y r(x;y)]$$

1. $(\forall x)[(p(x) \wedge (\forall z)q(z;y)) \rightarrow \neg(\forall y)r(x;y)]$;
2. $\forall x[(p(x) \wedge (\forall z)q(z;y)) \rightarrow \neg \forall y_1 r(x;y_1)]$;
3. $\forall x[\neg(p(x) \wedge \exists z \neg q(z;y)) \vee \exists y_1 \neg r(x;y_1)]$;
4. $\forall x \exists z \exists y_1[\neg(p(x) \wedge \neg q(z;y)) \vee \neg r(x;y_1)]$;
5. $\forall x \exists z \exists y_1[\neg(p(x) \vee \neg r(x;y_1)) \wedge (\neg q(z;y) \vee \neg r(x;y_1)]$.

3.3.4 Forme de Skolem

La forme Skolem est constituée que des quantifications universelles. La procédure de construction est comme suit :

1. Prendre la fermeture existentielle de D (c.à.d quantifier les variables libres de D par le quantificateur $\exists$) ;
2. Éliminer dans D tous les quantificateurs redondants ;
3. Renommer chaque variable quantifiée dans D plus d'une fois ;
4. Éliminer les occurrences des connecteurs autres que $\neg$, $\vee$ et $\wedge$.
5. Pousser à droite tous les connecteurs en remplaçant :
   $\neg(\forall xA)$ par $\exists(\neg A)$;
   $\neg(\exists xA)$ par $\forall x(\neg A)$;
   $\neg(A \vee B)$ par $\neg A \wedge \neg B$;
   $\neg(A \wedge B)$ par $\neg A \vee \neg B$;
   $\neg(\neg A)$ par A.
   Jusqu'à ce que toutes les occurrences de $\neg$ précède immédiatement une formule atomique.
6. Pousser les quantificateurs à droite en remplaçant :
   - $\exists(A \vee B)$ par
     - $A \vee \exists B$ si x n'est pas libre dans A ;
     - $\exists A \vee B$ si x n'est pas libre dans B.
   - $\exists(A \wedge B)$ par
     - $A \wedge \exists B$ si x n'est pas libre dans A ;
     - $\exists A \wedge B$ si x n'est pas libre dans B.
   - $\forall x(A \wedge B)$ par
     - $A \wedge \forall xB$ si x n'est pas libre dans A ;
     - $\forall xA \wedge B$ si x n'est pas libre dans B.
7. Éliminer les quantificateurs existentiels. Prendre la formule la plus à gauche de la forme $\exists yB(y)$ et la remplacer par $B(f(x_1, ..., x_n))$ où
   - $x_1, ..., x_n$ sont des variables distinctes de $\exists yB(y)$ quantifiées universellement à gauche de $\exists yB(y)$.
   - f est fonction à n arguments qui n'occure pas dans la formule. Répéter ce processus jusqu'à l'élimination complète des tous les quantificateurs extensionnels. Dans le cas particulier $n = 0$, remplacer $\exists yB(y)$ par $B(a)$ où a est une nouvelle constante qui n'occure pas dans la formule.
8. Déplacer les quantificateurs $\forall$ à gauche ;
9. Distribuer $\wedge$ et $\vee$;
10. Simplifier en utilisant les règles préservant la satisfaction.

Exemple

Transformer la formule suivante sous forme Skolem.

$$\forall x(p(x) \rightarrow \exists z(\neg\forall y(q(x,y) \rightarrow p(f(t))) \wedge \forall y(q(x,y) \rightarrow p(x))))$$

1. Prendre la fermeture existentielle de D et éliminer le quantificateur redondant $\exists z$.
   $\exists t\forall x(p(x) \rightarrow (\neg\forall y(q(x,y) \rightarrow p(f(t))) \wedge \forall y(q(x,y) \rightarrow p(x))))$
2. Renommer la variable y, elle est quantifiée deux fois.
   $\exists t\forall x(p(x) \rightarrow (\neg\forall y(q(x,y) \rightarrow p(f(t))) \wedge \forall z(q(x,z) \rightarrow p(x))))$
3. Éliminer toutes les occurrences de $\rightarrow$.
   $\exists t\forall x(\neg p(x) \vee (\neg\forall y(\neg q(x,y) \vee p(f(t))) \wedge \forall z(\neg q(x,z) \vee p(x))))$
4. Déplacer $\neg$.
   $\exists t\forall x(\neg p(x) \vee (\exists y(q(x,y) \wedge \neg p(f(t))) \wedge \forall z(\neg q(x,z) \vee p(x))))$
5. Pousser les quantificateurs $\exists y$ et $\forall z$ à droite.
   $\forall x(\neg p(x) \vee ((q(x,g(x)) \wedge \neg p(f(a))) \wedge (\forall z(\neg q(x,z) \vee p(x))))$
6. Éliminer les quantificateurs $\exists t$ et $\exists y$.
   $\forall x(\neg p(x) \vee ((q(x,g(x)) \wedge \neg p(f(a))) \wedge (\forall z(\neg q(x,z) \vee p(x))))$
7. Déplacer le quantificateur $\forall$ à gauche.
   $\forall x\forall z(\neg p(x) \vee ((q(x,g(x)) \wedge \neg p(f(a))) \wedge ((\neg q(x,z) \vee p(x))))$
8. Après distribution de $\vee$ et $\wedge$ simplification.
   $\forall x\{(\neg p(x) \vee ((q(x,g(x))) \wedge \neg p(f(a))\}$

3.3.5 Forme Clausale

Un littéral est défini comme étant une formule atomique ou une négation d'une formule atomique. Une clause est une disjonction de plusieurs littéraux. Une formule du calcul des prédicats est dite sous forme clausale si elle est sous la forme :

$$\forall x_1 ... \forall x_n(C_1 \wedge C_2 \wedge ... \wedge C_k)$$

Algorithme de construction

Pour construire la forme clausale, il faut :

1. Prendre la fermeture existentielle de D (c.à.d. quantifier les variables libres de D par le quantificateur $\exists$),
2. Éliminer dans D tous les quantificateurs redondants,
3. Renommer chaque variable quantifiée dans D plus d'une fois,
4. Éliminer les occurrences des connecteurs autres que : $\neg, \vee$ et $\wedge$.
5. Pousser à droite tous les connecteurs en remplaçant :
   $\neg(\forall x A)$ par $\exists x(\neg A)$;
   $\neg(\exists x A)$ par $\forall x(\neg A)$;
   $\neg(A \vee B)$ par $\neg A \wedge \neg B$;
   $\neg(A \wedge B)$ par $\neg A \vee \neg B$;
   $\neg(\neg A)$ par A.

Exemple

Si nous partons de la formule skolémisée :

$$sk(A_2) = \forall x \forall y (P(x, f(x)) \rightarrow (Q(f(x), y) \wedge R(y, g(x, y)))$$

1. l'élimination des quantificateurs universels, donne :
   $\neg P(x, f(x)) \vee (Q(f(x), y) \wedge R(y, g(x, y)))$
2. l'application de la transformation $\rightarrow$, $\vee$, on obtient :
   $\neg P(x, f(x)) \vee (Q(f(x), y) \wedge R(y, g(x, y)))$
3. Puis par la distributivité, on obtient :
   $(\neg P(x, f(x)) \vee Q(f(x), y)) \wedge (\neg P(x, f(x)) \vee R(y, g(x, y)))$
4. Nous obtenons à la fin les deux clauses :

   (a) $\neg P(x, f(x)) \vee Q(f(x), y)$
   (b) $\neg P(x, f(x)) \vee R(y, g(x, y))$

Remarque

Toute formule est équivalente à une formule prénexe, Skolem et clausale.

3.4 Conclusion

Dans ce chapitre, nous avons présenté la logique des prédicats considérée plus riche que la logique propositionnelle par l'inclusion des quantificateurs universels au même temps que les variables propositionnelles. A travers les prochaines annexes, nous proposons une suite d'exercices corrigés.

Annexe A

Série d'exercices

Exercice 1

Parmi les assertions suivantes, lesquelles sont vraies, lesquelles sont fausses et pourquoi ?

1. Si Napoléon était chinois alors $3 - 2 = 2$;
2. Soit Cléopâtre était chinoise, soit les grenouilles aboient ;
3. Soit les roses sont des animaux, soit les chiens ont 4 pattes ;
4. Si l'homme est un quadrupède, alors il parle ;
5. Les roses ne sont ni des animaux, ni des fleurs ;
6. Alger est en Algérie ou Madrid est en chine.

Exercice 2

Soient (P), (Q) et (R) trois propositions, donner la négation de

1. (P) et $(non(Q)$ ou $(R))$,
2. $((P)et(Q)) \rightarrow (R)$

Exercice 3

Démontrer les règles suivantes à l'aide d'une table de vérité.

1. $(A.B)(A.B).C = (B.C)A.(B.C)$,
2. $\overline{A+B} = \overline{A}.\overline{B}$,
3. $\overline{A.B} = \overline{A} + \overline{B}$

Exercice 4

Écrire l'expression $\overline{A + \overline{B}.C}$ sous forme normale conjonctive et puis sous forme normale disjonctive.

Exercice 5

Soient les quatre assertions suivantes :

1. $\exists x \in R \forall y \in R x + y > 0$;
2. $\forall x \in R \exists y \in R x + y > 0$;
3. $\forall x \in R \forall y \in R x + y > 0$;
4. $\exists x \in R \forall y \in R y^2 > x$.

Les assertions 1, 2, 3, 4 sont-elles vraies ou fausses ? Donner leur négation.

Exercice 6

Traduisez les énoncés suivants en formule logique :

1. Quand il fait beau, Jean est heureux ; il fait soleil ; donc, Jean est heureux.
2. Quand il fait beau, Jean est heureux ; or, il fait mauvais ; donc Jean est malheureux.
3. Quand il fait beau, Jean est heureux ; or, Jean est malheureux ; donc il fait mauvais.

Exercice 7

Les « formules » suivantes (voir la table A.1) sont-elles des formules bien formées ?

$a \vee b \wedge c$	$\neg a \vee \neg b \wedge c$	$a \vee \neg b \wedge c$	$a \neg \vee b \wedge c$
(a)	$(a)b$	$\neg b(a)$	$a \vee \neg(b \wedge c)$
$a \neg(\vee b \wedge c)$	$a \rightarrow b$	$a \leftarrow a$	$a \rightarrow b \leftrightarrow c$
$a \rightarrow \neg(b \leftrightarrow c)$	$a \neg \rightarrow b$	$a \vee (b \leftrightarrow c) \neg \rightarrow d$	$a \vee (b \leftrightarrow c) \rightarrow \neg c \rightarrow d$

TABLE A.1: Table des formules

Exercice 8

Explicitez le parenthèsage implicite des formules suivantes :

1. $a \rightarrow b \rightarrow c$;
2. $a \vee b \wedge c$;
3. $a \vee b \wedge c \leftrightarrow d \rightarrow \neg e \vee f \wedge g$

Simplifiez au maximum le parenthèsage des formules suivantes (voir la table A.2) :

(a)	$((a \vee b))$	$((a) \wedge (b))$
$\neg(((a) \wedge b))$	$a \vee (b \wedge c)$	$(a \vee b) \wedge c$
$(a \wedge (b \rightarrow c))$	$((a \vee b) \wedge c) \leftrightarrow e$	$(((a \vee b) \wedge c) \leftrightarrow e) \rightarrow f$
$(((a \rightarrow b) \rightarrow c) \rightarrow d)$	$(a \wedge (b \wedge c))$	$(a \rightarrow (b \rightarrow c))$
$(\neg(a \vee b))$	$((a \wedge b) \rightarrow c)$	$((a \wedge b) \vee c) \leftrightarrow (e \rightarrow f)$

TABLE A.2: Table des formules à simplifier

Exercice 9

A l'aide de l'arbre de construction (décomposition), vérifier si les chaines de symboles suivantes sont des Fbf.

$((\neg(P \vee Q) \rightarrow \neg\neg\neg Q \leftrightarrow R)$	$((P \supset Q) \supset R) \vee S) \vee T))$
$A \supset B \supset C \equiv ((D \equiv E) \wedge A \vee B)$	$(B \supset D) \vee (A \vee C) \equiv E \wedge \wedge \neg B \vee \neg D)$

TABLE A.3: Table des formules à vérifier

Exercice 10

Vérifier au moyen des tables de vérité si les conséquences sémantiques sont valides :

1. $A, (A \supset B) \models B$;
2. $A, (B \supset C), (A \supset (B \supset C)) \models C$;
3. $(A \supset C), (C \supset B), \neg A \models \neg B$.

Exercice 11

Donner les formes normales conjonctives et disjonctives des propositions suivantes à l'aide de tables de vérité :

1. $(((P \wedge \neg Q) \vee (R \vee \neg P)) \supset (R \equiv \neg P))$;
2. $(((\neg P \vee R) \wedge (Q \wedge P)) \wedge (\neg R \vee Q))$.

Exercice 12

Soit le raisonnement suivant :
— Quand il fait soleil, je mets mes lunettes ou je ne sors pas.
— Je ne reste à la maison que sans lunettes et par temps gris.
Donc si je ne mets pas mes lunettes, c'est qu'il fait gris.

1. Formaliser ce raisonnement en utilisant les variables suivantes :
   s : il fait soleil, l : je mets mes lunettes, m : je reste à la maison.
2. Montrer que le raisonnement ci-dessus est correct (valide) :
   (a) en utilisant la table de vérité ;
   (b) en utilisant une mise en forme normale par le calcul (algébriquement).

Exercice 13

Considérons l'extension du langage du calcul propositionnel (CP) par les deux symboles

— Construire les tables de vérité des formules propositionnelles :

— Montrer que dans cette extension, les ensembles

Exercice 14

Montrez que les formules suivantes sont des théorèmes :

1. $(p \to (q \to r)) \to (q \to (p \to r))$;
2. $(p \to q) \to ((r \to p) \to (r \to q))$.

Exercice 15

Soient les trois formules $f1$, $f2$ et $f3$ suivantes :
$f1 \equiv (A \to (A \to B)) \to (A \to B)$;
$f2 \equiv ((A \to B) \to C) \to (B \to C)$;
$f3 \equiv ((A \to B) \to (B \to C)) \to (A \to ((A \to B) \to C))$.

1. Montrer, à l'aide du théorème de déduction, que $f1$, $f2$ et $f3$ sont des théorèmes.
2. Montrer, maintenant, que $f1$, $f2$ et $f3$ sont des théorèmes ; et cela sans utiliser d'hypothèses.

Exercice 16

Utiliser la méthode de résolution propositionnelle pour montrer si la formule C est une conséquence logique des formules A et B ;
Où : $A = (p \to (q \vee r))$; $B = \neg r$ et $C = (\neg q \to \neg p)$.

Exercice 17

1. En utilisant la méthode de résolution propositionnelle, démontrer que la formule suivante est une tautologie :
   $F = (a \wedge \neg b) \vee (\neg a \wedge \neg(b \vee c)) \vee (\neg c \wedge b) \vee (b \wedge c \wedge a) \vee (c \wedge \neg a)$.
2. En déduire ce qu'il en est de la validité de la formule A suivante :
   $A = (a \to b) \wedge (\neg a \to (b \vee c)) \wedge (\neg c \to \neg b) \wedge ((b \wedge c) \to \neg a) \wedge (c \to a)$.

Exercice 18

Formalisez les phrases suivantes dans le langage des prédicats, en conservant autant de structure que possible.

1. Pierre marche et Jean court.
2. Si Pierre court, alors Pierre sera fatigué.
3. Marie viendra, mais pas Jeanne.
4. Si le président de la République ne répond pas aux questions, alors l'éditorialiste écrira un article ravageur.
5. Si cet homme ou son ami reviennent dans le quartier, je ferai signe.

Exercice 19

Dans les formules suivantes, dites si les variables utilisées sont des variables libres ou des variables liées.

$p(x) \vee q(y)$	$\forall x p(x) \vee q(y)$
$\exists z(\forall x p(x,z) \vee q(y,z))$	$\forall x p(x) \vee q(y,z)$
$\forall x p(x) \vee q(x)$	$\exists z(p(x,z) \vee \exists y q(y,z))$
$p(x,y) \vee p(y,x) \wedge p(x,x) \rightarrow p(x,y)$	$\forall x \forall y(p(x,y) \vee p(y,x)) \wedge p(x,x) \rightarrow p(x,y)$
$\forall x \forall y(p(x,y) \vee p(y,x)) \wedge p(x,x) \rightarrow \exists x \forall y p(x,y)$	$\forall x \forall y(p(x,y) \vee p(y,x)) \wedge \forall x p(x,x) \rightarrow \exists x \forall y p(x,y)$

Table A.4: Table des variables

Exercice 20

Traduisez les phrases suivantes dans le langage des prédicats. Conservez autant de structure que possible.

1. Tous les plombiers sont des hommes ;
2. Pierre est riche ;
3. Si Pierre est un plombier, Pierre est riche ;
4. Quelques plombiers ne sont pas riches ;

5. Aucun plombier n'est riche.

Utilisez les traductions suivantes pour les prédicats :

— P_x : x est plombier ;

— H_x : x est un homme ;

— R_x : x est riche.

Exercice 21

Traduisez les formules en français. Utilisez les traductions des prédicats ci-dessous (table A.5), ainsi que :

— Q_x : x habite à Quimper ;

— a : Antoine ;

— b : Béatrice ;

— c : Christine.

P_a	Q_c	R_b
$\exists x Q_x$	$\forall x(P_x \rightarrow Q_x)$	$\exists x(Q_x \wedge \neg P_x)$

TABLE A.5: Table des prédicats

Éliminer les parenthèses dans ce qui suit :$\forall z(\exists y(\forall x(R(x, y) \wedge P(z))))$.

Annexe B

Corrigés des exercices

Exercice 1

1. Il s'agit, ici d'une implication. « Napoléon est chinois » est faux et « $3-2=2$ » est faux, or la seule possibilité pour qu'une implication soit fausse est qu'une assertion vraie implique une assertion fausse, donc l'assertion 1 **est vraie**.
2. Une phrase, en français, du genre « soit..., soit...» se traduit mathématiquement par «... ou... » « Cléopâtre était chinoise » est faux et « les grenouilles aboient » est faux donc l'assertion 2 **est fausse**.
3. « Les roses sont des animaux » est faux et « les chiens ont 4 pattes » est vrai, donc l'assertion 3 **est vraie**.
4. « L'homme est un quadrupède » est faux et « il parle » est vrai, donc l'assertion 4 **est vraie**.
5. « Les roses ne sont ni des animaux, ni des fleurs » peut se traduire par « les roses ne sont pas des animaux et les roses ne sont pas des fleurs ». « les roses ne sont pas des animaux » est vrai et « les roses ne sont pas des fleurs » est faux donc « les roses ne sont ni des animaux, ni des fleurs » , donc l'assertion 5 est fausse.
6. « Alger est en Algérie » est vrai et « Madrid est en chine » est faux, donc « Alger est en Algérie ou Madrid est en chine » est vraie.

Exercice 2

1. $non((P)$ et $(non(Q)$ ou $(R))) \equiv (non(P)$ ou $non(non(Q)$ ou $(R))) \equiv (non(P)$ ou $((Q)$ et $non(R)) \equiv (non(P)$ ou $(Q))$ et $(non(P)$ ou $non(R)) \equiv (non(P)$ ou $(Q))$ et $non((P)$ et $(R))$
   Les deux dernières équivalences logiques me paraissent acceptables, parce qu'il y a souvent différentes façon d'exprimer une négation, ensuite il faut voir dans les exercices comment se présentent les propositions (P), (Q) et (R).
2. $non(((P)$ et $(Q)) \rightarrow (R)) \equiv ((P)et(Q))$ et $non(R) \equiv (P)$ et $(Q)etnon(R)$.

Exercice 5

1. est fausse. Car sa négation qui est $\forall x \in R \exists y \in R x+y \leq 0$ est vraie. Étant donné $x \in R$ il existe toujours un $y \in R$ tel que $x+y \leq 0$, par exemple on peut prendre $y = -(x+1)$ et alors $x+y = x-x-1$.
2. est vraie, pour un x donné, on peut prendre (par exemple) $y = -x+1$ et alors $x+y = 1 > 0$. La négation de (2) est $\exists x \in R \forall y \in R x+y \leq 0$.
3. $\forall x \in R \forall y \in R x+y > 0$ est fausse, par exemple $x = -1$, $y = 0$. La négation est $\exists x \in R \exists y \in R x+y \leq 0$.
4. est vraie, on peut prendre $x = -1$. La négation est : $\forall x \in R \exists y \in R y^2 \leq x$.

Exercice 6

Ci-dessous une suite de propositions élémentaires qui nous aiderons à formaliser les énoncés.

1. $b \equiv$ il fait beau.
   $h \equiv$ Jean est heureux.
   Hypothèse 1 : $H_1 \equiv b \rightarrow h$ (Quand il fait beau, Jean est heureux).
   Hypothèse 2 : $H_2 \equiv b$ (il fait soleil).
   Conclusion : $C \equiv h$ (Jean est heureux).
   Formulation : $H_1 \wedge H_2 \rightarrow C \equiv (b \rightarrow h) \wedge b \rightarrow h$.
2. $b \equiv$ il fait beau.
   $h \equiv$ Jean est heureux.

Hypothèse 1 : $H_1 \equiv b \rightarrow h$ (Quand il fait beau, Jean est heureux).
Hypothèse 2 : $H_2 \equiv \neg b$ (il fait mauvais).
Conclusion : $C \equiv \neg h$ (Jean est malheureux).
Formulation : $H_1 \wedge H_2 \rightarrow C \equiv (b \rightarrow h) \wedge \neg b \rightarrow \neg h$.

3. $b \equiv$ il fait beau.
   $h \equiv$ Jean est heureux.
   Hypothèse 1 : $H_1 \equiv b \rightarrow h$ (Quand il fait beau, Jean est heureux).
   Hypothèse 2 : $H_2 \equiv \neg h$ (Jean est malheureux).
   Conclusion : $C \equiv \neg b$ (il fait mauvais).
   Formulation : $H_1 \wedge H_2 \rightarrow C \equiv (b \rightarrow h) \wedge \neg h \rightarrow \neg b$.

Exercice 8

Explicitez le parenthèsage implicite des formules suivantes :

1. $a \rightarrow b \rightarrow c \equiv ((a \rightarrow b) \rightarrow c)$
2. $a \vee b \wedge c \equiv (a \vee (b \wedge c))$
3. $a \vee b \wedge c \leftrightarrow d \rightarrow \neg e \vee f \wedge g \equiv ((a \vee (b \wedge c)) \leftrightarrow (d \rightarrow ((\neg e) \vee (f \wedge g))))$

Simplifiez au maximum le parenthèsage des formules suivantes (voir la table B.1) :

$(a) \equiv a$	$((a \vee b)) \equiv a \vee b$	$((a) \wedge (b)) \equiv a \wedge b$
$\neg(((a) \wedge b)) \equiv \neg(a \vee b)$	$a \vee (b \wedge c) \equiv \neg(a \wedge b)$	$(a \vee b) \wedge c \equiv a \vee b \wedge c$
$(a \wedge (b \rightarrow c)) \equiv a \wedge (b \rightarrow c)$	$((a \vee b) \wedge c) \leftrightarrow e \equiv (a \vee b) \wedge c \leftrightarrow e$	$(((a \vee b) \wedge c) \leftrightarrow e) \rightarrow f \equiv ((a \vee b) \wedge c \leftrightarrow e) \rightarrow f$
$(((a \rightarrow b) \rightarrow c) \rightarrow d) \equiv a \rightarrow b \rightarrow c \rightarrow d$	$(a \wedge (b \wedge c)) \equiv a \wedge b \wedge c$	$(a \rightarrow (b \rightarrow c)) \equiv a \rightarrow (b \rightarrow c)$
$(\neg(a \vee b)) \equiv \neg(a \vee b)$	$((a \wedge b) \rightarrow c) \equiv a \wedge b \rightarrow c$	$((a \wedge b) \vee c) \leftrightarrow (e \rightarrow f) \equiv a \wedge b \vee c \leftrightarrow e \rightarrow f$

TABLE B.1: Simplification des formules

Exercice 12

1. Avec les variables s, l et m dénotant : s : il fait soleil ; l : je mets mes lunettes ; m : je reste à la maison ; on peut formaliser le raisonnement donné avec une

formule du calcul propositionnel suivant :
$((s \to (l \vee m)) \wedge (m \to (\neg l \wedge \neg s))) \to (\neg l \to \neg s)$

2. Le raisonnement considéré est valide car la formule (A) est une tautologie ; montré à la table de vérité (Figure B.1) : soient le sous formules : $I = s \to (l \vee m)$, $II = m \to (\neg l \wedge \neg s)$, $III = (\neg l \to \neg s)$ avec $(A) = (I \wedge II) \to III$. Par une méthode algébrique : $I = \neg S \wedge l \wedge m = (l \vee \neg s) \vee m) = (\neg l \to \neg s) \vee m$ puisque $m \to (\neg l \wedge \neg s)$ alors $I \to (\neg l \to \neg s) \wedge (\neg l \wedge \neg s)$.
Or $(\ negl \to \neg s) \vee (\neg l \wedge \neg s) = (\neg l \to \neg s)$ car $(A\ toB) \vee (A \wedge B) = (A \to B))$
On aura donc $(I \wedge II) \to (\neg l \to \neg s) \vee (\neg l \wedge \neg s) = (\neg l \to \neg s)$; c'est à dire $(I \wedge II) \to III$ est une tautologie.

s	l	m	l ∨ m	I	¬l ∧ ¬s	II	III	I ∧ II	(A)
0	0	0	0	1	1	1	1	1	1
0	0	1	1	1	1	1	1	1	1
0	1	0	1	1	0	1	1	1	1
0	1	1	1	1	0	0	1	0	1
1	0	0	0	0	0	1	0	0	1
1	0	1	1	1	0	0	0	0	1
1	1	0	1	1	0	1	1	1	1
1	1	1	1	1	0	0	1	0	1

FIGURE B.1: Table de vérité de la formule A

Exercice 13

Rappel

Un système complet de connecteurs C est un ensemble de connecteurs logiques tel que toute formule propositionnelle est équivalente à une formule n'utilisant que les connecteurs de C.

1. Table de vérité B.2
2. $\{\vee, \to\}$ est un ensemble complet de connecteurs car on peut exprimer les autres connecteurs usuels en fonction des éléments de cet ensemble :
$\neg A = A \to \bot$

A	B	T	⊥	B ∧ T	A→(B∧ T)	B ∨ ⊥	A→(B∨⊥)	B → ⊥	A→(B→⊥)
0	0	1	0	0	1	0	1	1	1
0	1	1	0	1	1	1	1	0	1
1	0	1	0	0	0	0	0	1	1
1	1	1	0	1	1	1	1	0	0

FIGURE B.2: Table de vérité des formules propositionnelles

$$A \wedge B = \neg(\neg A \vee \neg B) = (\neg A \vee \neg B)) \to \bot = ((A \to \bot) \vee (B \to \bot)) \to \bot$$
$$A \leftrightarrow B = (A \to B) \wedge (B \to A) = (((A \to B) \to \bot) \vee ((B \to A)\bot)) \to \bot$$

$\{\to\}$ constitue à lui seul un ensemble complet complet de connecteurs : en effet il suffit de remarquer que $A \vee B = (A \to B) \to B$.

Exercice 16

On a $A, B \models C$ si et seulement si $A, B, \neg C \models \Box$.
On écrit A, B et $\neg C$ sous forme clausale :
$A \equiv \neg P \vee q \vee r$ (une clause),
$B \equiv \neg r$ (une clause),
$\neg C \equiv \neg q \wedge p$ (deux clauses).
On a donc :
$C_1 = \neg p \vee q \vee r$
$C_2 = \neg r$
$C_3 = \neg q$
$C_4 = p$
$C_5 = q \vee r$ Res(C_1, C_4)
$C_6 = r$ $Res(C_3, C_5)$
$C_7 = \Box$ $Res(C_2, C_6)$
Donc C est une conséquence logique de A et B.

Exercice 17

1. Pour montrer que F est une tautologie, on démontre que $\neg F$ est une contradiction.

$F = (a \wedge \neg b) \vee (\neg a \wedge \neg(b \vee c)) \vee (\neg c \wedge b) \vee (b \wedge c \wedge a) \vee (c \wedge \neg a)$.

On a donc :

$C_1 = \neg a \vee b$

$C_2 = a \vee b \vee c$

$C_3 = c \vee \neg b$

$C_4 = \neg b \vee \neg c \vee \neg a$

$C_5 = \neg c \vee a$

A partir de cela, on obtient $\neg F$.

$C_6 = b \vee c$ Res(C_1, C_2)

$C_7 = c$ Res(C_3, C_6)

$C_8 = \neg c \vee \neg a$ Res(C_1, C_4)

$C_9 = \neg c$ Res(C_5, C_8)

$C_{10} = \square$ Res(C_7, C_9)

Donc $\neg F \models \square$, d'où : $\models F$.

2. En effectuant le calcul, on obtient $A \equiv \neg F$; donc A est une contradiction.

Exercice 20

1. Tous les plombiers sont des hommes.
   $\forall x(P_x \Rightarrow H_x)$
2. Pierre est riche.
   a : Pierre, R_a
3. Si Pierre est un plombier, Pierre est riche.
   a : Pierre, $P_a \Rightarrow R_a$
4. Quelques plombiers sont riches.
   $\exists(P_x \wedge R_x)$
5. Aucun plombier n'est riche.
   $\neg\exists(P_x \wedge R_x)$

Annexe C

Examen corrigé

Questions de cours

1. Dire si l'affirmation suivante est correcte ou non, justifier votre réponse.
   — Soit **valid** un algorithme qui répond vrai pour une formule P si la formule P est valide et faux sinon. A partir de **valid**, il est possible de construire un algorithme **sat** qui répond vrai si son entrée est une formule satisfiable et faux sinon.
2. Écrire à l'aide de quantificateurs les propositions suivantes :
   — Le carré de tout réel est positif.
   — Certains réels sont strictement supérieurs à leur carré.

Correction

1. Vrai : en effet une formule A est insatisfiable si et seulement si la formule $\neg A$ est valide donc on teste si $\neg A$ est valide auquel cas auquel cas A n'est pas satisfiable sinon A est satisfiable.
2. — $\forall x \in R, x^2 \geq 0$;
   — $\exists x \in R/x \geq x^2$.

Exercice 1

Trois collègues, Albert, Bernard et Charles déjeunent ensemble chaque jour ouvrable. Les affirmations suivantes sont vraies :

1. Si Albert commande un dessert, Bernard en commande un aussi.
2. Chaque jour, soit Bernard, soit Charles, mais pas les deux, commandent un dessert.
3. Albert ou Charles, ou les deux, commandent chaque jour un dessert.
4. Si Charles commande un dessert, Albert fait de même.

Question **:***Exprimer les données du problème comme des formules propositionnelles.*

Correction

On introduit des variables propositionnelles a, b, c qui représentent le fait que Albert (a), Bernard (b) et Charles (c) prennent un dessert. On traduit ainsi le problème :

1. $a \rightarrow b$;
2. $(b \wedge \neg c) \vee (\neg b \vee c)$ ou $(b \vee c) \wedge (\neg(b \wedge c))$;
3. $a \vee c$;
4. $c \rightarrow a$.

Exercice 2

On définit le connecteur de Sheffer noté $|$ (barre de Sheffer, ou encore NAND) par :$p|q = \neg(p \wedge q)$.

1. Donner la table de vérité de la formule $(p|q)$.
2. Donner la table de vérité de la formule $((p|q)|(p|q))$.
3. On veut maintenant exprimer les connecteurs usuels en utilisant la barre de Sheffer, et rien qu'elle.
   (a) Donner la table de vérité de la formule $(p|p)$ et en déduire que le connecteur $\neg$ peut être défini en n'utilisant que la barre de Sheffer.

(b) Trouver une formule équivalente à $p \vee q$, qui n'utilise que la barre de Sheffer (éventuellement plusieurs fois).

(c) Trouver une formule équivalente à $p \to q$, qui n'utilise que la barre de Sheffer (éventuellement plusieurs fois).

4. D'après les questions précédentes quel système complet peut-on former ? Expliquer.

Correction

1. Voir la table C.1.

p	q	$p\|q$
V	V	F
V	F	V
F	V	V
F	V	V

TABLE C.1: Table de vérité $(p|q)$

2. On retrouve la table C.2 de vérité de $p \wedge q$.

p	q	$(p\|q)\|(p\|q)$
V	V	V
V	F	F
F	V	F
F	V	F

TABLE C.2: Table de vérité $(p|q)|(p|q)$

3. (a) On peut poser $\neg p \equiv (p|p)$

p	$(p\|p)$
V	F
F	V

TABLE C.3: Table de vérité $(p|p)$

(b) On a $p \vee q \equiv (\neg p \wedge \neg q) \equiv (\neg p | \neg q)$ on en déduit un codage de $p \vee q$ est : $(p|p)|(q|q)$.

(c) On a $p \to q \equiv \neg(p \wedge \neg q) \equiv p|\neg q$ et donc en déduit un codage de $p \to q$ qui est : $p|(q|q)$.

Exercice 3

Soit la formule P définie comme $(p \to (q \to r)) \to (r \lor \neg p)$

1. Donner la table de vérité de la formule P ;
2. Dire si la formule est valide, satisfiable, insatisfiable ?
3. La formule P a-t-elle un modèle ? Si oui lequel ?
4. Donner la FNC et la FND de la formule P.

Correction

1. Table de vérité :

p	q	r	$q \to r$	$p \to (q \to r)$	$(p \to (q \to r)) \to (r \lor \neg p)$
		V			V
F		F			V
V	V	F	F	F	V
V	F	F	V	V	F

TABLE C.4: Table de vérité de P

2. La formule n'est pas valide (une ligne de la table où elle est fausse). Elle est satisfiable (une ligne où elle est vraie) et donc elle n'est pas insatisfiable.
3. La formule a plusieurs modèles, par exemple $\{p \to V; q \to V; r \to V\}$.
4. Forme normale conjonctive qui exprime que l'on n'est pas sur la ligne où la valeur de la formule est fausse : $\neg p \lor q \lor r$. C'est aussi une forme normale disjonctive (parmi d'autres).

Ouvrages de références

Champavère, J. (2007). Logique des propositions et logique des prédicats. *Note de cours*.

de La Guillonnière, G. S. (2012). Logique.

Gilles, F. (s. d.). *Logique propositionnelle.* Consulté sur http://cui.unige.ch/isi/icle-wiki/_media/cours:ffsi:4-logique-propositionnelle.pdf

Laurent, A. (2009). *Infz21, logiques du raisonnement valide.* Consulté sur https://laurent-audibert.developpez.com/Cours-Logique/?page=annexe-correction-travaux-diriges

Lucas, T., Berlanger, I., & Degauquier, V. (2014). *Initiation à la logique formelle. avec exercices et corrigés.*

Marie-Pierre, G. (2002). *Systèmes de preuves en logique des propositions.* Consulté sur http://spiralconnect.univ-lyon1.fr/webapp/course/course.html?id=129847&viewMode=visu&idChapter=136499

Rozière, P. (2004). Logique mathématique : introduction.