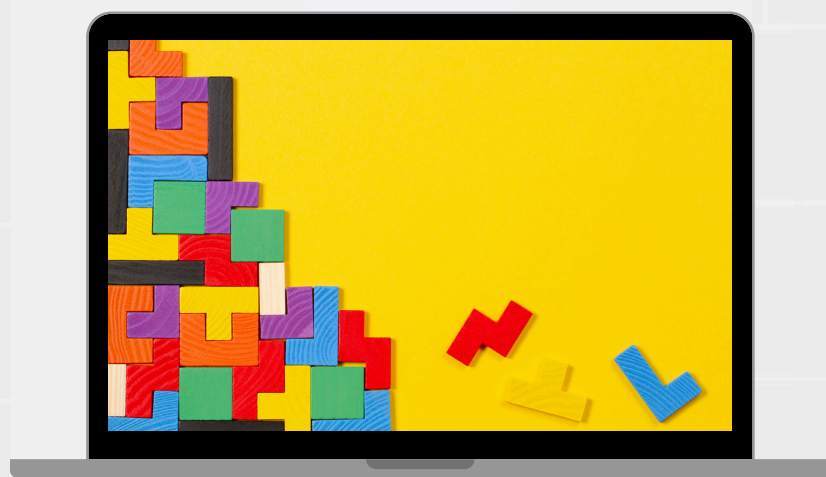




Logique Mathématique

M317



POLYCOPIÉ PÉDAGOGIQUE

Auteur : Fatima Bourebaa

Niveau: 3ème Année
mathématiques

Année universitaire 2025-2026



Logique Mathématique M317

Polycopié de Cours

Niveau : 3ème Année Mathématiques (PES-PEM)

Préparé par : Fatima Bourebaa
Grade : Maître Assistant A docteur

Syllabus du Module : Logique Mathématique – M317

Nom du Syllabus	Logique Mathématique – M317
Niveau	3 ^e Année Mathématiques (Bac + 5 / Bac + 4)
Volume horaire hebdomadaire (V.H.H)	Cours : 1h30
Volume horaire annuel (V.H.A)	45 heures
Système	Annuel
Coefficient	1

Préface

La logique mathématique est une branche fondamentale des mathématiques consacrée à l'étude de la validité des raisonnements [1]. Elle intervient aussi bien dans la formulation et la vérification de démonstrations que dans la conception de systèmes informatiques fiables. Dans ce module, nous cherchons, à travers la logique propositionnelle et la logique du premier ordre, à apporter des réponses formelles à des questions essentielles telles que :

- Qu'est-ce qu'un énoncé logique ?
- Comment déterminer si un énoncé est valide ?
- Qu'est-ce qu'une démonstration correcte ?

Destiné aux étudiants de troisième année Mathématiques (PES–PEM), ce cours vise à développer une pensée logique rigoureuse, indispensable aux études mathématiques avancées. Nous commencerons par la logique propositionnelle (ou logique d'ordre 0), qui traite d'énoncés simples ou composés, chacun doté d'une valeur de vérité — vraie ou fausse. Ces énoncés, appelés propositions et notés généralement p, q, r , peuvent être combinés à l'aide de connecteurs logiques pour former des expressions plus complexes.

La logique des prédicats prolonge cette approche en introduisant des variables parcourant un domaine, des prédicats, des fonctions ainsi que les quantificateurs universel et existentiel. Elle permet d'exprimer des énoncés plus riches, tels que « tout entier possède un successeur » ou « il existe un nombre premier pair ». Cette logique est dite du premier ordre car les quantificateurs portent uniquement sur les éléments du domaine. Si l'on autorise la quantification sur les prédicats et les fonctions, on obtient alors des logiques d'ordre supérieur, où ces entités deviennent elles-mêmes des objets manipulables [2].

D'autres familles de logiques, que nous ne développerons pas dans ce module, méritent toutefois d'être mentionnées pour situer le champ disciplinaire : les logiques temporelles (raisonnement sur l'évolution d'un système), les logiques modales (nécessité et possibilité), ou encore les logiques floues (degrés de vérité continus entre 0 et 1).

Toute logique se caractérise par trois composantes fondamentales [3] :

- **la syntaxe** : définit les règles de formation des formules bien écrites ;
- **la sémantique** : précise l'interprétation des formules et leurs valeurs de vérité ;
- **le système de preuve** : regroupe les axiomes et règles d'inférence permettant d'établir des démonstrations valides.

Ce polycopié est organisé comme suit : le premier chapitre introduit les notions fondamentales du raisonnement formel : modèles, arbres de Beth, formes normales et résolution. Le deuxième chapitre est consacré aux principales méthodes de démonstration, notamment le système de Hilbert et la déduction naturelle. Le troisième chapitre porte sur la logique

des prédicats. Le dernier chapitre est dédié à la théorie des ensembles, qui constitue le cadre fondamental de la construction mathématique moderne. On y présente les axiomes essentiels du système de Zermelo–Fraenkel et leur rôle dans la formalisation rigoureuse des objets mathématiques.

Table des matières

Préface	2
1 Logique propositionnelle	9
1.1 Rappel : Types de raisonnement	10
1.1.1 Les entiers naturels et l'induction	10
1.1.2 Raisonnement par Récurrence (Induction)	11
1.1.3 Raisonnement par l'absurde	13
1.1.4 Raisonnement par contraposée	14
1.1.5 Raisonnement par contre-exemple	14
1.2 Opérations fondamentales de la logique propositionnelles	15
1.3 Tableaux de vérité	17
1.4 Syntaxe de la logique propositionnelle	18
1.4.1 Alphabet	18
1.4.2 Formules bien formées	18
1.4.3 Règles de priorité	19
1.4.4 Arbre syntaxique	19
1.5 Sémantique de la logique propositionnelle	20
1.5.1 Valuation booléenne	20
1.5.2 Tautologie, contradiction et contingence	20
1.5.3 Implication et équivalence logique	21
1.5.4 Modèles et satisfaction	22
1.5.5 Satisfaisabilité et conséquence logique d'un ensemble de formules	23
1.5.6 Théorème de remplacement	25
1.6 Forme Normale Conjonctive (FNC)	26
1.6.1 Définitions	26
1.6.2 Intérêt	27
1.6.3 Techniques de mise sous FNC	27
1.7 Graphes et tautologies (Méthode de Beth)	29
1.8 Connecteurs logiques et langage naturel	33
1.9 Logique propositionnelle et circuits électroniques	35
1.9.1 Portes Logiques Fondamentales	35
1.9.2 Méthodologie de synthèse d'un circuit combinatoire	35
1.9.3 Optimisation et simplification : méthode de Karnaugh	36
1.9.4 Exemple 1 : Circuit d'Alarme	36
1.9.5 Exemple 2 : Vote majoritaire	37
1.10 Exercices	38

1.11	Solutions	40
2	Système formel de la la logique des propositions	48
2.1	Introduction	48
2.2	Système Formel Propositionnel	48
2.3	Approche à la Hilbert	49
2.3.1	Démonstration	50
2.3.2	Déduction	50
2.4	Déduction Naturelle	51
2.4.1	Règles de Conjonction	51
2.4.1.1	Règle d'Introduction.	51
2.4.1.2	Règles d'Élimination.	51
2.4.2	Règles de Disjonction	52
2.4.2.1	Règles d'Introduction.	52
2.4.2.2	Règle d'Élimination.	52
2.4.3	Règles d'Implication	52
2.4.3.1	Règle d'Introduction.	52
2.4.3.2	Règle d'Élimination.	53
2.4.4	Règles de Négation	53
2.4.4.1	Règle d'Introduction.	53
2.4.4.2	Règle d'Élimination.	53
2.4.5	Règles Spéciales	53
2.4.5.1	EFQ (Ex Falso Quodlibet).	53
2.4.5.2	RAA (reductio ad absurdum ou Réduction à l'absurde).	53
2.5	Outil de construction de preuve	56
2.6	Correction et complétude du calcul propositionnel	58
2.7	Exercices	59
3	Calcul des prédicats	65
3.1	De la logique propositionnelle à la logique des prédicats (LP0 à LP1)	67
3.2	Prédicats à une variable	67
3.3	Prédicats à deux ou plusieurs variables	68
3.3.1	Alphabet	69
3.3.2	Signature, Termes, Prédicats, Formules atomiques et Formules	70
3.3.2.1	Règles de priorité	71
3.3.3	Arbre syntaxique	71
3.3.4	Variables libres et variables liées	73
3.3.5	Substitution	74
3.3.6	Unification	75
3.4	Sémantique de la logique des prédicats	76
3.4.1	Interprétation	76
3.4.2	Interprétation des formules	77
3.5	Formulation dans la logique des prédicats	78
3.6	Preuve par Résolution	79
3.6.1	Règle de résolution	79
3.6.2	Étapes d'une preuve par résolution	80
3.6.3	Forme prénexe	81
3.6.4	Forme de Skolem	82

3.7	Exercices	86
3.8	Solutions	88
4	Théorie des ensembles	93
4.1	Introduction	94
4.2	Théorie naïve et paradoxe de Russell	94
4.2.1	Compréhension illimitée	94
4.2.2	Limite de la définition naïve : le paradoxe de Russell	95
4.3	Concept de déduction formelle	95
4.3.1	Composants d'un système formel	95
4.3.1.1	Langage de la théorie des ensembles	95
4.3.1.2	Règles d'inférence principales	96
4.4	Axiomes de Zermelo-Fraenkel (ZF)	96
4.4.1	Ontologie de la théorie ZF	96
4.4.2	Les Axiomes Fondamentaux	97
4.4.3	Axiomes de Construction	98
4.4.3.1	Schéma de séparation (compréhension restreinte)	98
4.4.3.2	Schéma de remplacement	99
4.4.4	Axiome de Régularité (ou de Fondation)	100
4.5	Propriétés métatéoriques	101
4.5.1	Consistance	101
4.5.2	Complétude	101
4.5.2.1	Incomplétude de ZF	101
4.5.3	Catégoricité	101
4.5.3.1	Théorème de Löwenheim-Skolem	102
4.6	Axiome du Continu	102
4.6.1	L'hypothèse du continu	102
4.6.2	Indépendance de l'hypothèse du continu	102
4.7	Exemples de modèles de la théorie ZF	103
4.8	Récapitulation (Rôles des Axiomes)	104
4.9	Exercices	104

Table des figures

1.1	Règles de Représentation.	30
2.1	Schéma de preuve en déduction naturelle.	55
2.2	Capture montrant la preuve automatique de $\neg p \wedge \neg q \rightarrow \neg(p \vee q)$	58
3.1	Arbre syntaxique d'une formule en LP1	72
3.2	Arbre syntaxique d'une <i>fbf</i> complexe	73
3.3	Portée des quantificateurs illustrée.	74
3.4	Substitution sur un littéral	75
3.5	Illustration du processus d'unification	75
3.6	Schéma d'une interprétation.	77

Liste des tableaux

1.1	Équivalences logiques classiques	22
1.2	Tables de vérité des portes logiques fondamentales	35
2.1	Comparaison entre la preuve manuelle et la preuve automatique produite par l'outil DN.	59

Chapitre 1

Logique propositionnelle

À la fin de ce chapitre, l'étudiant sera capable de :

- Comprendre la syntaxe et la sémantique des formules propositionnelles.
- Construire et interpréter des tables de vérité.
- Appliquer les lois et équivalences logiques pour simplifier des expressions.
- Modéliser un problème de logique et le traduire en énoncé formel.
- Analyser sémantiquement des formules à l'aide de la méthode des tableaux de Beth (arbres sémantiques).
- Traduire une expression logique en circuit électronique (portes AND, OR, NOT, etc).

1.1 Rappel : Types de raisonnement

Nous commençons ce chapitre par rappeler les principaux types de raisonnement : l'induction, le raisonnement par l'absurde, la contraposée et le contre-exemple.

1.1.1 Les entiers naturels et l'induction

On note les entiers naturels par :

$$1, 2, 3, \dots$$

Les trois points indiquent que l'ensemble des entiers naturels est infini.

On peut également donner une définition formelle particulièrement utile dans de nombreuses applications mathématiques où chaque entier naturel est construit à partir de l'entier 0 et d'une fonction S , appelée *fonction successeur*, définie par :

$$\forall n \in \mathbb{N}, \quad S(n) = n + 1.$$

Ainsi, les entiers naturels sont générés de manière inductive :

$$1 = S(0), \quad 2 = S(1), \quad 3 = S(2), \quad \dots$$

Définition 1.1 (Entiers naturels). Les entiers naturels sont définis comme suit [2]. :

1. $0 \in \mathbb{N}$
2. Si $n \in \mathbb{N}$ alors $S(n) \in \mathbb{N}$
3. L'ensemble $\mathbb{N}$ est défini exclusivement par les clauses 1 et 2.

Principe d'induction. Cette définition constitue un exemple de définition par induction. Le principe d'induction sert à démontrer des propriétés portant sur des objets mathématiques isomorphes à l'ensemble des entiers naturels.

Dans ce cadre, on distingue généralement trois règles :

- **La règle de base** : elle permet de définir un premier élément, appelé *élément initial* ou *noyau*, à partir duquel la construction commence.
- **La règle d'induction** : elle décrit la manière de générer un nouvel élément de l'ensemble à partir d'un élément déjà construit. C'est cette règle qui assure la progression inductive.
- **La règle de fermeture** : elle stipule que les seuls éléments appartenant à l'ensemble sont ceux obtenus à partir de l'élément de base par application répétée de la règle d'induction. Elle exclut donc tout élément ne pouvant être construit ainsi.

1.1.2 Raisonnement par Récurrence (Induction)

Principe. Pour démontrer qu'une propriété $P(n)$ est vraie pour tout élément d'un ensemble E isomorphe à $\mathbb{N}$ (ou à une partie de $\mathbb{N}$), on procède en deux étapes :

1. *Initialisation* : montrer que $P(n_0)$ est vraie, où n_0 est le plus petit élément de E .
2. *Hérédité* : montrer que pour tout $n \in E$,

$$P(n) \Rightarrow P(n+1).$$

Exemple 1.1. Montrer que :

1. $\forall n \in \mathbb{N}^* : 1 + 3 + 5 + \dots + (2n-1) = n^2$.
2. $\forall n \in \mathbb{N}^* : 1 + 4 + 9 + \dots + n^2 = \frac{n(n+1)(2n+1)}{6}$.

$$1. \forall n \in \mathbb{N}^* : 1 + 3 + 5 + \dots + (2n-1) = n^2.$$

Preuve. [Démonstration par récurrence sur n] Posons pour chaque $n \in \mathbb{N}^*$ la propriété

$$\mathcal{P}(n) : \sum_{k=1}^n (2k-1) = n^2.$$

Initialisation ($n = 1$).

$$\sum_{k=1}^1 (2k-1) = 2 \cdot 1 - 1 = 1 = 1^2.$$

La propriété $\mathcal{P}(1)$ est vraie.

Hérédité. Supposons $\mathcal{P}(n)$ vraie pour un $n \geq 1$, c'est-à-dire

$$\sum_{k=1}^n (2k-1) = n^2.$$

Montrons alors que $\mathcal{P}(n+1)$ est vraie :

$$\sum_{k=1}^{n+1} (2k-1) = \left(\sum_{k=1}^n (2k-1) \right) + (2(n+1)-1)$$

(en isolant le dernier terme). Par l'hypothèse de récurrence,

$$\sum_{k=1}^{n+1} (2k-1) = n^2 + (2n+2-1) = n^2 + (2n+1).$$

Or

$$n^2 + (2n+1) = (n+1)^2.$$

Ainsi,

$$\sum_{k=1}^{n+1} (2k-1) = (n+1)^2,$$

ce qui prouve $\mathcal{P}(n+1)$.

Conclusion. La propriété est vraie pour $n=1$ et héréditaire de n à $n+1$. Par le principe de récurrence, $\mathcal{P}(n)$ est donc vraie pour tout $n \in \mathbb{N}^*$. $\square$

$$2. \forall n \in \mathbb{N}^* : 1 + 4 + 9 + \dots + n^2 = \frac{n(n+1)(2n+1)}{6}.$$

Preuve. [Démonstration par récurrence sur n] Posons, pour chaque $n \in \mathbb{N}^*$, la propriété

$$\mathcal{P}(n) : \sum_{k=1}^n k^2 = \frac{n(n+1)(2n+1)}{6}.$$

Initialisation ($n=1$).

$$\sum_{k=1}^1 k^2 = 1^2 = 1 \quad \text{et} \quad \frac{1 \cdot 2 \cdot (2 \cdot 1 + 1)}{6} = \frac{1 \cdot 2 \cdot 3}{6} = 1.$$

Donc $\mathcal{P}(1)$ est vraie.

Hérédité. Supposons $\mathcal{P}(n)$ vraie pour un $n \geq 1$, c'est-à-dire

$$\sum_{k=1}^n k^2 = \frac{n(n+1)(2n+1)}{6}.$$

Montrons $\mathcal{P}(n+1)$:

$$\sum_{k=1}^{n+1} k^2 = \left(\sum_{k=1}^n k^2 \right) + (n+1)^2 = \frac{n(n+1)(2n+1)}{6} + (n+1)^2$$

(par hypothèse de récurrence, puis en isolant le dernier terme). Factorisons $(n+1)$:

$$= (n+1) \left(\frac{n(2n+1)}{6} + (n+1) \right) = (n+1) \left(\frac{2n^2+n}{6} + \frac{6n+6}{6} \right)$$

$$= (n+1) \cdot \frac{2n^2 + 7n + 6}{6}.$$

Or $2n^2 + 7n + 6 = (2n+3)(n+2)$ (en effet $(2n+3)(n+2) = 2n^2 + 4n + 3n + 6$). Ainsi,

$$\sum_{k=1}^{n+1} k^2 = (n+1) \cdot \frac{(2n+3)(n+2)}{6} = \frac{(n+1)(n+2)(2(n+1)+1)}{6},$$

ce qui est exactement la formule annoncée avec n remplacé par $n+1$. Donc $\mathcal{P}(n+1)$ est vraie.

Conclusion. La propriété est vraie pour $n=1$ et héréditaire de n à $n+1$. Par le principe de récurrence, $\mathcal{P}(n)$ est vraie pour tout $n \in \mathbb{N}^*$. $\square$

1.1.3 Raisonnement par l'absurde

Principe. Pour démontrer qu'une proposition P est vraie, on peut raisonner par l'absurde. Il s'agit de supposer que sa négation $\neg P$ est vraie et de montrer que cette hypothèse conduit à une contradiction $\perp$. Dans ce cas, l'hypothèse $\neg P$ étant impossible, on conclut que P est vraie.

Exemple 1.2. Montrer que :

1. $\frac{a}{b+1} = \frac{b}{a+1} \Rightarrow a=b, a, b \geq 0$.
2. Montrer qu'il n'y a pas d'entier naturel entre n et $n+1$.

Preuve. [Démonstration par l'absurde de $\frac{a}{b+1} = \frac{b}{a+1} \Rightarrow a=b, a, b \geq 0$.] Supposons

$$\frac{a}{b+1} = \frac{b}{a+1} \quad \text{et} \quad a \neq b,$$

avec $a, b \geq 0$.

(1) *Justification du produit en croix.* Comme $a, b \geq 0$, on a $a+1 > 0$ et $b+1 > 0$, donc le produit en croix est licite :

$$a(a+1) = b(b+1).$$

(2) *Mise sous forme factorisée.* Développons et regroupons :

$$a^2 + a = b^2 + b \iff a^2 - b^2 + a - b = 0 \iff (a-b)(a+b) + (a-b) = 0.$$

Factorisons par $(a-b)$:

$$(a-b)((a+b)+1) = 0.$$

(3) *Contradiction*. Deux cas se présentent :

$$(i) \ a - b = 0 \implies a = b \quad (\text{contraire à l'hypothèse } a \neq b),$$

$$(ii) \ (a + b) + 1 = 0.$$

Or $a, b \geq 0$ implique $a + b \geq 0$, donc $(a + b) + 1 \geq 1 > 0$, ce qui rend le cas (ii) impossible. Dans tous les cas on aboutit à une contradiction $\perp$ avec $a \neq b$.

Ainsi, l'hypothèse $a \neq b$ est intenable. Donc nécessairement $a = b$ (c.q.f.d). $\square$

Preuve. [Preuve par l'absurde]

$$\begin{aligned} \exists m \in \mathbb{N} \quad n < m < n + 1 & \quad (\text{hyp.}) \\ \implies \exists k \in \mathbb{N}^* \quad (m = n + k) & \quad (m > n) \\ \implies \exists \ell \in \mathbb{N}^* \quad (n + 1 = m + \ell) & \quad (n + 1 > m) \\ \implies 1 = (n + 1) - n = (m + \ell) - n = (n + k + \ell) - n = k + \ell \\ \implies 1 = k + \ell \geq 1 + 1 = 2 & \implies \perp (\text{c.q.f.d}). \end{aligned}$$

$\square$

1.1.4 Raisonnement par contraposée

Principe. Si on souhaite prouver une proposition $p \Rightarrow q$, le raisonnement par contraposée consiste à prouver $\neg q \Rightarrow \neg p$.

Exemple 1.3. Montrer que : n^2 est pair $\Rightarrow n$ est pair.

Montrons la contraposée : Si n est impair, alors n^2 est impair.

Preuve. [Preuve par contraposée]

$$\begin{aligned} (\exists k \in \mathbb{N} \quad n = 2k + 1) & \implies n^2 = (2k + 1)^2 \implies n^2 = 4k^2 + 4k + 1 \implies n^2 = 2(2k^2 + 2k) + 1 \\ & \implies \exists m \in \mathbb{N} \quad (n^2 = 2m + 1) \quad (\text{avec } m = 2k^2 + 2k) \implies n^2 \text{ est impair.} \end{aligned}$$

La contraposée étant vraie, on conclut bien n^2 pair $\Rightarrow n$ pair. $\square$

1.1.5 Raisonnement par contre-exemple

Principe. Pour montrer qu'une propriété universelle de la forme $\forall x P(x)$ est fausse, il suffit de trouver un élément x_0 tel que $\neg P(x_0)$. Autrement dit, la négation de la propriété, $\exists x_0 \neg P(x_0)$, est alors vérifiée.

Exemple 1.4. Considérons la propriété suivante :

$$\forall n \in \mathbb{N}, n^2 \geq n + 5.$$

Pour déterminer si cette propriété est fausse, cherchons un contre-exemple, c'est-à-dire un entier n_0 tel que :

$$n_0^2 < n_0 + 5.$$

Essayons quelques valeurs :

$$0^2 = 0 \quad \text{et} \quad 0 + 5 = 5 \quad \Rightarrow \quad 0 < 5.$$

Ainsi, pour $n_0 = 0$, la propriété est fausse. On a donc :

$$\neg P(0) : \quad 0^2 < 0 + 5.$$

Puisqu'il existe un entier $n_0 = 0$ pour lequel la propriété est fausse, la proposition universelle

$$\forall n \in \mathbb{N}, n^2 \geq n + 5$$

est fausse.

1.2 Opérations fondamentales de la logique propositionnelles

La logique des propositions constitue le premier niveau de la logique mathématique. Elle s'intéresse à l'étude des énoncés simples — appelés *propositions* — auxquels on peut associer une valeur de vérité : *vrai* ou *faux* [5]. Son objectif est d'analyser la structure logique des raisonnements sans tenir compte des détails des énoncés, mais uniquement de leur forme. Dans ce cadre, les propositions sont considérées comme des unités atomiques que l'on peut combiner grâce à des *connecteurs logiques* (négation, conjonction, disjonction, implication, etc.) pour former des propositions plus complexes.

Définition 1.2. Une *proposition* est un énoncé auquel on peut attribuer une valeur de vérité, qui est soit **vrai** (V), soit **faux** (F).

Toute expression ne pouvant être évaluée en vrai ou faux (comme un ordre, une question ou un énoncé ambigu) n'est pas une proposition au sens logique.

Exemple 1.5. Soient les énoncés suivants :

- « Il pleut » : proposition (vraie ou fausse selon la situation).
- « $2 + 6 = 9$ » : proposition fausse.
- « Révises tes leçons » : ce n'est pas une proposition (pas de valeur de vérité).
- « Quelle belle journée ! » : : ce n'est pas une proposition

En logique des propositions, *les opérations fondamentales (ou connecteurs logiques)* sont les opérations de base utilisées pour construire des propositions complexes à partir de propositions simples. Voici les principales opérations fondamentales :

1. Négation ($\neg$)

- **Définition** : Inverse la valeur de vérité d'une proposition.
- **Symbole** : $\neg$ (ou parfois $\sim$).
- **Exemple** : Si P est "x est Premier", alors $\neg P$ est "x n'est pas premier".

2. Conjonction (ET, $\wedge$)

- **Définition** : Combine deux propositions et est vraie seulement si les deux propositions sont vraies.
- **Symbole** : $\wedge$.
- **Exemple** : $P \wedge Q$ signifie "x est premier ET x est pair".

3. Disjonction (OU, $\vee$)

- **Définition** : Combine deux propositions et est vraie si au moins l'une des deux propositions est vraie.
- **Symbole** : $\vee$.
- **Exemple** : $P \vee Q$ signifie "x est premier OU x est pair".

4. Implication (SI...ALORS, $\rightarrow$)

- **Définition** : Exprime une relation conditionnelle où la vérité de la seconde proposition dépend de la première.
- **Symbole** : $\rightarrow$.
- **Exemple** : $P \rightarrow Q$ signifie "SI x est premier, ALORS x est pair".

5. Équivalence (SI ET SEULEMENT SI, $\leftrightarrow$)

- **Définition** : Indique que deux propositions ont la même valeur de vérité.
- **Symbole** : $\leftrightarrow$.
- **Exemple** : $P \leftrightarrow Q$ signifie "w est premier SI ET SEULEMENT SI x est pair".

1.3 Tableaux de vérité

Une table de vérité permet de représenter toutes les valeurs possibles d'une expression logique, en fonction des valeurs de vérité de ses variables (Vrai ou Faux). Elle constitue un outil fondamental de la sémantique propositionnelle [1]. Elle sert à vérifier qu'une expression est une tautologie, contradiction ou contingence. Ci-après sont récapitulées les tables complètes pour les connecteurs fondamentaux :

p	$\neg p$	p	q	$p \wedge q$	p	q	$p \vee q$
		V	V	V	V	V	V
V	F	V	F	F	V	F	V
F	V	F	V	F	F	V	V
		F	F	F	F	F	F

p	q	$p \rightarrow q$	p	q	$p \leftrightarrow q$
V	V	V	V	V	V
V	F	F	V	F	F
F	V	V	F	V	F
F	F	V	F	F	V

Remarques

- D'autres connecteurs, comme le OU exclusif (XOR) ou le NON-ET (NAND), peuvent être dérivés de ces opérations fondamentales.

Pour Tracer une table de vérité :

1. Identifier les variables de la formule.
2. Créer 2^n lignes, où n est le nombre de variables.
3. Évaluer la formule en ajoutant des colonnes intermédiaires (sous formules) si nécessaire.

Exemple pour la formule $(P \wedge Q) \rightarrow R$:

3 variables donc $2^3 = 8$ lignes

P	Q	R	$P \wedge Q$	$(P \wedge Q) \rightarrow R$
V	V	V	V	V
V	V	F	V	F
V	F	V	F	V
V	F	F	F	V
F	V	V	F	V
F	V	F	F	V
F	F	V	F	V
F	F	F	F	V

1.4 Syntaxe de la logique propositionnelle

La syntaxe est l'aspect grammatical de la logique. Elle décrit comment on construit des formules correctes, indépendamment de leur sens.

1.4.1 Alphabet

Définition 1.3 (Alphabet). En logique propositionnelle, l'alphabet est constitué de [1]. :

- symboles de variables propositionnelles notées : $p, q, r, \dots$
- symboles de connecteurs logiques : $\neg$ (négation), $\wedge$ (conjonction), $\vee$ (disjonction), $\rightarrow$ (implication), $\leftrightarrow$ (équivalence)
- Parenthèse ouvrante '(' et parenthèse fermante ')'

La logique propositionnelle ne comporte ni symboles de variables, ni symboles de quantifications.

1.4.2 Formules bien formées

Définition 1.4 (fbfs). Une formule Bien Formée (*fbf*) est une formule définie inductivement par [6, 1]. :

1. Les variables propositionnelles sont des formules bien formées (*fbfs*)
2. Si A est une fbf alors $\neg A$ est une *fbf*.
3. Si A et B sont des *fbfs* alors $(A \wedge B)$, $(A \vee B)$, $(A \rightarrow B)$ et $(A \leftrightarrow B)$ sont des *fbfs*.
4. Règle de fermeture. Les *fbfs* sont exclusivement définies par les clauses 1, 2 et 3.

Exemple 1.6. $p \rightarrow a \leftrightarrow s$ et p sont des formules en logique propositionnelle.

En revanche, $\forall w (p \wedge q)$ n'est pas une formule de LP0.

Remarque 1.1

On peut ignorer les parenthèses les plus externes.

1.4.3 Règles de priorité

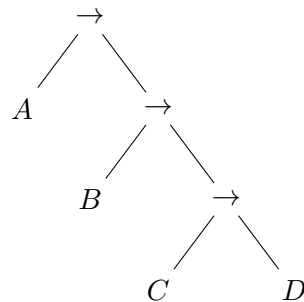
L'utilisation des parenthèses permet de retrouver sans ambiguïté la structure des formules bien formées. Cependant, leur présence parfois excessive, dans les formules complexes, rend leur utilisation contraignante. La définition de règles de priorité sur les connecteurs logiques permet de résoudre ce problème/ L'ordre par priorité décroissante est le suivant :

- La négation $\neg$,
- ensuite la conjonction $\wedge$,
- puis la disjonction $\vee$,
- l'implication $\rightarrow$,
- et enfin l'équivalence $\leftrightarrow$

A priorité égale, on considère prioritaire le connecteur à gauche, sauf pour le cas du connecteur implique $\rightarrow$ qui est associatif à droite. Ainsi, la formule $A \rightarrow B \rightarrow C$ est équivalente à $(A \rightarrow (B \rightarrow C))$

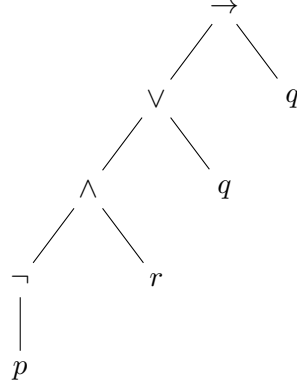
1.4.4 Arbre syntaxique

Les arbres sont une représentation usuelle pour mettre en exergue la structure d'une formule bien formée. La racine de l'arbre représente le connecteur principal de la formule, les branches sont les sous formules et les feuilles correspondent aux variables propositionnelles. L'arbre correspondant à la formule $A \rightarrow B \rightarrow C \rightarrow D$ est présenté ci-après :



Exemple 1.7. Trouver l'arbre syntaxique correspondant à la formule : $\neg p \wedge r \vee q \rightarrow q$

$\neg p \wedge r \vee q \rightarrow q$ est équivalente à $((\neg p) \wedge r) \vee q \rightarrow q$



1.5 Sémantique de la logique propositionnelle

La sémantique précise comment les formules reçoivent une valeur de vérité [1]. On fixe une *valuation booléenne* qui assigne V ou F aux variables et s'étend aux connecteurs de manière compositionnelle. Dans ce qui suit nous adoptons la convention suivante : Noter le faux par F ou 0 et le vrai par V ou 1 .

1.5.1 Valuation booléenne

Définition 1.5. Une valuation booléenne v est une application de l'ensemble $\mathcal{F}$ des formules bien formées (*fbfs*) vers l'ensemble $\{F, V\}$ vérifiant les conditions suivantes :

1. $v(\neg x) = \neg v(x)$
2. $v(x \wedge y) = v(x) \wedge v(y), v(x \vee y) = v(x) \vee v(y), v(x \rightarrow y) = v(x) \rightarrow v(y)$

Exemple 1.8. Calculer $v(\neg p \vee q)$ avec $v(p) = V$ et $v(q) = F$.

$$\begin{aligned}
 v(\neg p \vee q) &= v(\neg p) \vee v(q) \quad (\text{par } v(x \vee y) = v(x) \vee v(y)) \\
 &= \neg v(p) \vee v(q) \quad (\text{par } v(\neg x) = \neg v(x)) \\
 &= \neg V \vee F \quad (\text{hypothèses}) \\
 &= F \vee F \quad (\text{table de } \neg) \\
 &= F \quad (\text{table de } \vee).
 \end{aligned}$$

1.5.2 Tautologie, contradiction et contingence

Définition 1.6 (Tautologie, contradiction, contingence). Soit $A \in \mathcal{F}$ une *fbf* et $v : \mathcal{F} \rightarrow \{V, F\}$ une valuation booléenne.

- Une formule bien formée A est une *tautologie* ssi elle prend une valeur de vérité égale à 1 pour toutes les valeurs prises par les variables propositionnelles contenues dans cette formule, autrement dit, ssi $\forall v. v(A) = V$.

- Une formule bien formée A est une *contradiction* (ou *insatisfaisable*), notée $\perp$ si elle prend une valeur de vérité égale à 0 pour toutes les valeurs prises par les variables propositionnelles contenues dans cette formule, autrement dit, si $\forall v.v(A) = F$.
- une formule bien formée A est *contingente* s'il existe v tel que $v(A) = V$ et il existe v' tel que $v'(A) = F$.

On dit que A est *satisfaisable* s'il existe v tel que $v(A) = V$.

Exemple 1.9. 1. $A \vee \neg A$ est une tautologie (loi du tiers exclu).

2. $A \wedge \neg A$ est une contradiction.

3. $A \rightarrow (A \vee B)$ est une tautologie.

4. $A \leftrightarrow B$ est contingente (vraie ssi A et B ont même valeur).

Proposition 1.1 (Caractéristiques utiles). *Pour toute formule A :*

1. A est une tautologie ssi $\neg A$ est une contradiction.

2. A est satisfaisable ssi $\neg A$ n'est pas une tautologie.

Preuve. (1) et (2) sont immédiats par $v(\neg A) = \neg v(A)$. □

1.5.3 Implication et équivalence logique

Définition 1.7 (Implication logique). Soient ϕ, φ deux *fbfs*, on dit que ϕ implique logiquement φ ssi $\phi \rightarrow \varphi$ est une tautologie

Définition 1.8 (Équivalence logique). on dit que ϕ équivaut logiquement φ ou ϕ et φ sont logiquement équivalentes ssi $\phi \leftrightarrow \varphi$ est une tautologie.

Elles sont aussi dites logiquement équivalentes ssi pour toute valuation v , on a $v(\phi) = v(\varphi)$, on note cette équivalence $\phi \equiv \varphi$.

Quelques Équivalences logiques.

TABLE 1.1 – Équivalences logiques classiques

Nom	Formule	Équivalence
Double négation	$\neg(\neg p)$	$\equiv p$
Lois de De Morgan (pour $\wedge$)	$\neg(p \wedge q)$	$\equiv \neg p \vee \neg q$
Lois de De Morgan (pour $\vee$)	$\neg(p \vee q)$	$\equiv \neg p \wedge \neg q$
Élimination de l'implication	$p \rightarrow q$	$\equiv \neg p \vee q$
Idempotence (conjonction)	$p \wedge p$	$\equiv p$
Idempotence (disjonction)	$p \vee p$	$\equiv p$
Commutativité (conjonction)	$p \wedge q$	$\equiv q \wedge p$
Commutativité (disjonction)	$p \vee q$	$\equiv q \vee p$
Associativité (conjonction)	$(p \wedge q) \wedge r$	$\equiv p \wedge (q \wedge r)$
Associativité (disjonction)	$(p \vee q) \vee r$	$\equiv p \vee (q \vee r)$
Distributivité de $\wedge$ sur $\vee$	$p \wedge (q \vee r)$	$\equiv (p \wedge q) \vee (p \wedge r)$
Distributivité de $\vee$ sur $\wedge$	$p \vee (q \wedge r)$	$\equiv (p \vee q) \wedge (p \vee r)$

1.5.4 Modèles et satisfaction

Dans le cadre propositionnel, une *valuation* $v : \mathcal{F} \rightarrow \{V, F\}$ (cf. section précédente) associe à chaque formule bien formée une valeur de vérité. Une valuation est dite *modèle* d'une formule (ou d'un ensemble de formules) lorsque cette (ces) formule(s) est (sont) vraie(s) sous v .

Définition 1.9 (Modèles et satisfaction). Soit $A \in \mathcal{F}$ et $\Gamma \subseteq \mathcal{F}$.

— L'ensemble des modèles de A est

$$\llbracket A \rrbracket := \{ v : \mathcal{F} \rightarrow \{V, F\} \mid v(A) = V \}.$$

— L'ensemble des modèles de Γ est

$$\llbracket \Gamma \rrbracket := \{ v : \mathcal{F} \rightarrow \{V, F\} \mid \forall B \in \Gamma, v(B) = V \} = \bigcap_{B \in \Gamma} \llbracket B \rrbracket.$$

— On dira que v *satisfait* A (resp. Γ) et l'on écrira $v \in \llbracket A \rrbracket$ (resp. $v \in \llbracket \Gamma \rrbracket$) [4].

Définition 1.10 (Conséquence sémantique et équivalence logique). Soit Γ un ensemble de fbf et φ une formule propositionnelle. On dit que φ est une *conséquence sémantique* de Γ si toute valuation qui satisfait Γ satisfait aussi φ , ce que l'on note :

$$\Gamma \models \varphi$$

. Pour $\Gamma \subseteq \mathcal{F}$ et $A \in \mathcal{F}$:

— A est une *conséquence sémantique* de Γ , noté

$$\Gamma \models A \quad \text{ssi} \quad \llbracket \Gamma \rrbracket \subseteq \llbracket A \rrbracket.$$

— Deux formules A, B sont *logiquement équivalentes*, noté

$$A \equiv B \quad \text{ssi} \quad \llbracket A \rrbracket = \llbracket B \rrbracket.$$

Remarque 1.2

En logique propositionnelle, un *modèle* est simplement une valuation qui rend vraie la (les) formule(s) considérée(s).

Exemple 1.10. Soit $\Gamma = \{\neg p, p \vee q\}$ et $A = q$. Vérifions par table de vérité que $\Gamma \models q$.

p	q	$\neg p$	$p \vee q$	Γ satisfaite ?	q
V	V	F	V	non	V
V	F	F	V	non	F
F	V	V	V	oui	V
F	F	V	F	non	F

On observe que la seule valuation qui satisfait simultanément $\neg p$ et $p \vee q$ est celle où $p = F$ et $q = V$. Dans ce modèle, q est également vraie. Ainsi :

$$\llbracket \Gamma \rrbracket = \{ v \mid p = F, q = V \} \subseteq \llbracket q \rrbracket$$

d'où $\Gamma \models q$.

1.5.5 Satisfaisabilité et conséquence logique d'un ensemble de formules

Soit Γ un ensemble de formules bien formées (fbfs) :

1. Γ est *satisfait* par une valuation v si, pour toute formule $\varphi \in \Gamma$, on a $v(\varphi) = 1$. On note $\llbracket \Gamma \rrbracket$ l'ensemble des valuations qui satisfont Γ .
2. Γ est *satisfaisable* s'il existe une valuation qui le satisfait, i.e. $\llbracket \Gamma \rrbracket \neq \emptyset$.
3. Γ est *contradictoire* s'il n'existe aucune valuation qui le satisfait, i.e. $\llbracket \Gamma \rrbracket = \emptyset$.
4. Deux ensembles Γ_1 et Γ_2 sont *équivalents* si $\llbracket \Gamma_1 \rrbracket = \llbracket \Gamma_2 \rrbracket$.

Exemple 1.11. Considérons l'ensemble de formules :

$$\Gamma = \{ p \rightarrow q, p \}.$$

Recherche des valuations qui satisfont Γ . Construisons la table de vérité :

p	q	$p \rightarrow q$	p	Γ satisfait ?
V	V	V	V	oui
V	F	F	V	non
F	V	V	F	non
F	F	V	F	non

On observe que Γ est satisfaite uniquement pour la valuation $p = V$ et $q = V$. Ainsi :

$$\llbracket \Gamma \rrbracket = \{ v \mid p = V, q = V \}.$$

Comme cet ensemble n'est pas vide, Γ est *satisfaisable*.

Exemple de conséquence logique. Soit $A = q$. Toutes les valuations qui satisfont Γ satisfont aussi q , car dans le seul modèle de Γ on a $q = V$. Donc :

$$\llbracket \Gamma \rrbracket \subseteq \llbracket q \rrbracket \quad \Rightarrow \quad \Gamma \models q.$$

Exemple d'équivalence. Considérons

$$\Gamma_1 = \{ p \leftrightarrow q \}, \quad \Gamma_2 = \{ (p \rightarrow q) \wedge (q \rightarrow p) \}.$$

On construit la table de vérité :

p	q	$p \rightarrow q$	$q \rightarrow p$	$p \leftrightarrow q$	$(p \rightarrow q) \wedge (q \rightarrow p)$	Modèle commun ?
V	V	V	V	V	V	oui
V	F	F	V	F	F	non
F	V	V	F	F	F	non
F	F	V	V	V	V	oui

On observe que les valuations qui satisfont Γ_1 (celles pour lesquelles $p \leftrightarrow q$ est vraie) sont exactement les mêmes que celles qui satisfont Γ_2 (où $(p \rightarrow q) \wedge (q \rightarrow p)$ est vraie). Ainsi

$$\llbracket \Gamma_1 \rrbracket = \llbracket \Gamma_2 \rrbracket,$$

et Γ_1 et Γ_2 sont *équivalents*.

1.5.6 Théorème de remplacement

Proposition 1.2 (Substitution dans une tautologie). *Si nous remplaçons toutes les occurrences d'une variable propositionnelle dans une tautologie par une formule quelconque, nous obtenons encore une tautologie.*

Exemple 1.12. La formule $A \leftrightarrow B \leftrightarrow A$ est une tautologie. Remplaçons A par $(r \wedge s) \leftrightarrow t$ et B par $(r \vee s)$. On obtient :

$$\left((r \wedge s) \leftrightarrow t \right) \leftrightarrow (r \vee s) \leftrightarrow \left((r \wedge s) \leftrightarrow t \right),$$

qui est encore une tautologie.

Théorème 1.1. *Soit A une formule propositionnelle contenant les variables $p_1, p_2, \dots, p_n$ et soient $A_1, A_2, \dots, A_n$ des formules propositionnelles quelconques. Si A est une tautologie, alors la formule B , obtenue en remplaçant dans A chaque p_i par A_i ($1 \leq i \leq n$), est aussi une tautologie.*

Preuve. Notons $A = F(p_1, p_2, \dots, p_n)$ et $B = F(A_1, A_2, \dots, A_n)$. Pour toute valuation v , notons v' la valuation qui attribue à chaque p_i la valeur $v(A_i)$. Par construction, on a $v(B) = v'(A)$. Comme A est une tautologie, $v'(A) = V$ pour toute valuation v' . Ainsi $v(B) = V$ pour toute valuation v , donc B est aussi une tautologie. $\square$

Exemple 1.13. Prenons la tautologie suivante :

$$(p \rightarrow q) \vee p.$$

Remplaçons p par $(r \vee s)$ et q par $\neg t$. On obtient :

$$\left((r \vee s) \rightarrow \neg t \right) \vee (r \vee s).$$

r	s	t	$r \vee s$	$\neg t$	$(r \vee s) \rightarrow \neg t$	$((r \vee s) \rightarrow \neg t) \vee (r \vee s)$
1	1	1	1	0	0	1
1	1	0	1	1	1	1
1	0	1	1	0	0	1
1	0	0	1	1	1	1
0	1	1	1	0	0	1
0	1	0	1	1	1	1
0	0	1	0	0	1	1
0	0	0	0	1	1	1

La table de vérité confirme que cette nouvelle formule est vraie pour toute valuation, donc c'est aussi une tautologie.

1.6 Forme Normale Conjonctive (FNC)

La mise en *forme normale* consiste à réécrire une formule propositionnelle dans un format qui simplifie l'expression en vue de son analyse et qui facilite le calcul de sa valeur de vérité. Les deux formats usuels sont la *forme normale conjonctive* (FNC), qui est une $\wedge$ -conjonction de *clauses* (chacune étant une $\vee$ -disjonction de littéraux), et la *forme normale disjonctive* (FND), qui est une $\vee$ -disjonction de *monômes* (produits de littéraux) [7]. En pratique, on obtient une FNC par les étapes suivantes : (i) éliminer $\Rightarrow$ et $\Leftrightarrow$, (ii) pousser les négations jusqu'aux variables à l'aide des lois de De Morgan et de la double négation, (iii) distribuer $\vee$ sur $\wedge$ puis simplifier. La FNC ainsi obtenue n'est pas unique, mais elle est logiquement équivalente à la formule de départ. Nous détaillons ces points dans les sous-sections suivantes :

1.6.1 Définitions

- **Littéral** : une variable propositionnelle ou sa négation. Exemple : p , $\neg q$.
- **Produit** : une conjonction (souvent notée $\cdot$). Exemple : $p \cdot \neg q$ ou $p \wedge \neg q$.
- **Somme** : une disjonction (souvent notée $+$ ou $\vee$). Exemple : $p \vee \neg q$ ou $p + \neg q$.
- **Monôme** : un produit de littéraux. Exemple : $a \cdot \neg b \cdot c$.
- **Clause** : une somme de littéraux. Exemple : $\neg a + b + c$.
- **Forme Normale Disjonctive (FND)** : une somme de monômes. Exemple : $(a \wedge \neg b \wedge c) \vee (\neg a \wedge b \wedge c)$ ou $(a \cdot \neg b \cdot c) + (\neg a \cdot b \cdot c)$.
- **Forme Normale Conjonctive (FNC)** : produit de clauses. Exemple : $(a \vee \neg b \vee c) \wedge (\neg a \vee b \vee c)$ ou $(a + \neg b + c)(\neg a + b + c)$.

1.6.2 Intérêt

Pour montrer la validité d'une formule. Si $F = C_1 \wedge C_2 \wedge \dots \wedge C_n$, alors F est valide si et seulement si chacune de ces clauses vaut 1. Les C_i doivent être mises sous la forme : $p \vee \dots \vee \neg p$, c'est-à-dire contenir un littéral et sa négation.

1.6.3 Techniques de mise sous FNC

On cherche à mettre une formule F sous la forme :

$$(\dots \vee \dots \vee \dots) \wedge (\dots \vee \dots \vee \dots) \wedge (\dots \vee \dots \vee \dots).$$

Deux étapes :

1. **Après élimination de l'implication et de l'équivalence :**

— Mise sous forme de littéraux par application des lois de De Morgan :

$$\neg(A \wedge B) \equiv \neg A \vee \neg B$$

$$\neg(A \vee B) \equiv \neg A \wedge \neg B$$

$$\neg\neg A \equiv A$$

2. **Distribution de $\vee$ sur $\cdot$ et simplification.**

Rappels

$$\text{— } A \leftrightarrow B \equiv (A \rightarrow B) \wedge (B \rightarrow A)$$

$$\text{— } A \rightarrow B \equiv \neg A \vee B$$

$$\text{— } A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C)$$

Exemple 2.4.1. Mettre sous FNC les formules suivantes :

$$1. F_1 = (a \vee b \vee c) \wedge (a \vee \neg b)$$

$$2. F_2 = (a \wedge \neg b \wedge c) \vee (\neg a \wedge b)$$

$$3. F_3 = a \vee (\neg b \wedge c) \vee (\neg a \wedge b)$$

Mise sous forme normale conjonctive (FNC) — Étapes détaillées

On note $\cdot$ pour la conjonction et $+$ pour la disjonction.

1) **Formule F_1**

$$F_1 = (a + b + c) \cdot (a + \neg b)$$

Cette formule est déjà en FNC (conjonction de clauses) :

$$\boxed{(a + b + c) \cdot (a + \neg b)}$$

2) Formule F_2

$$F_2 = (a \cdot \neg b \cdot c) + (\neg a \cdot b)$$

Distribution de $+$ sur $\cdot$:

$$\begin{aligned} F_2 &\equiv (a + (\neg a \cdot b)) \cdot (\neg b + (\neg a \cdot b)) \cdot (c + (\neg a \cdot b)) \\ &\equiv ((a + \neg a) \cdot (a + b)) \cdot ((\neg b + \neg a) \cdot (\neg b + b)) \cdot ((c + \neg a) \cdot (c + b)) \\ &\equiv \underbrace{(a + \neg a)}_{\top} \cdot (a + b) \cdot (\neg a + \neg b) \cdot \underbrace{(\neg b + b)}_{\top} \cdot (c + \neg a) \cdot (c + b) \\ &\equiv (a + b) \cdot (\neg a + \neg b) \cdot (c + \neg a) \cdot (c + b). \end{aligned}$$

Ainsi :

$$\boxed{F_2 \equiv (a + b) \cdot (\neg a + \neg b) \cdot (c + \neg a) \cdot (c + b)}$$

3) Formule F_3

$$F_3 = a + (\neg b \cdot c) + (\neg a \cdot b)$$

On regroupe :

$$F_3 \equiv (a + (\neg b \cdot c)) + (\neg a \cdot b)$$

Distribution de $a + (\neg b \cdot c)$:

$$\equiv (a + \neg b) \cdot (a + c) + (\neg a \cdot b)$$

On distribue à nouveau :

$$\equiv ((a + \neg b) + (\neg a \cdot b)) \cdot ((a + c) + (\neg a \cdot b))$$

Distribution de chaque terme :

$$\begin{aligned} (a + \neg b) + (\neg a \cdot b) &\equiv (a + \neg b + \neg a) \cdot (a + \neg b + b) \\ &\equiv \top \cdot \top \equiv \top, \\ (a + c) + (\neg a \cdot b) &\equiv (a + c + \neg a) \cdot (a + c + b) \\ &\equiv \top \cdot (a + b + c) \equiv (a + b + c). \end{aligned}$$

Ainsi :

$$F_3 \equiv \top \cdot (a + b + c) \equiv (a + b + c)$$

$$\boxed{F_3 \equiv (a + b + c)}$$

Remarque Lors de la distribution, les clauses tautologiques comme $(p + \neg p)$ sont remplacées par $\top$ et éliminées dans la conjonction.

1.7 Graphes et tautologies (Méthode de Beth)

La méthode des **arbres sémantiques**, appelée aussi **méthode des tableaux sémantiques** [2], est une méthode non linéaire permettant de déterminer si une formule bien formée est une *tautologie* ou non. Cette méthode est aussi appelée *méthode de Beth*, en honneur du logicien néerlandais *Willem Beth* (1908–1964).

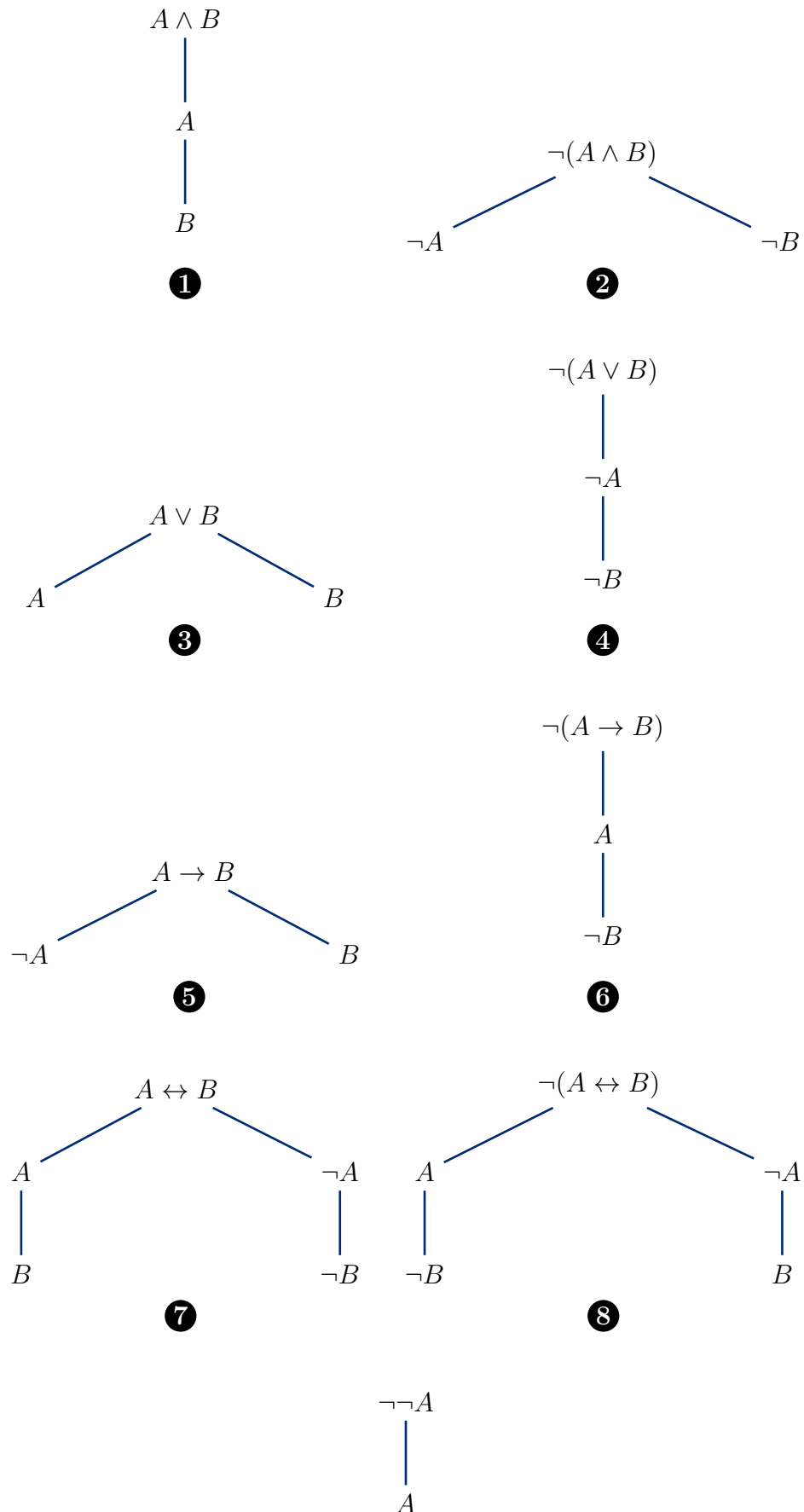
Règles de représentation. Les règles des tableaux sémantiques (Beth) décomposent une formule en sous-formules selon deux familles complémentaires : les *règles α* (non branchantes) et les *règles β* (branchantes). Les règles α s'appliquent sur une même branche : une formule est alors remplacée par plusieurs sous-formules ajoutées séquentiellement sur cette branche. Les règles β ouvrent au contraire deux branches alternatives : la formule est remplacée par un *choix* entre deux possibilités.

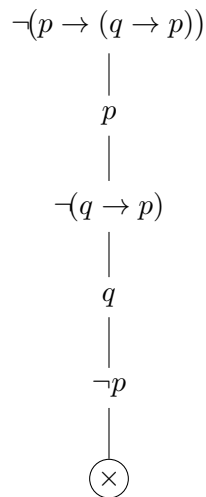
Une branche est dite *fermée* dès qu'elle contient un littéral et sa négation (p et $\neg p$) ; si *toutes* les branches se ferment, la formule placée à la racine (respectivement, sa négation si l'on teste une tautologie) est valide ; à l'inverse, toute *branche ouverte* fournit une valuation contre-exemple (satisfaisabilité)[10].

Principe de la méthode.

1. Placer la négation de la formule à vérifier à la **racine**.
2. Décomposer chaque formule en appliquant les règles de représentation. Les formules sont traitées l'une après l'autre.
3. Une branche est dite **fermée** si elle comporte un atome (littéral) et sa négation ; on inscrit alors une croix sur cette branche.
4. Si toutes les branches sont fermées, la formule en question est **contradictoire**.

FIGURE 1.1 – Règles de Représentation.





Exemple 1.14 (Arbres de Beth — Application des règles.). Soient les formules :

1. $(p \wedge (p \rightarrow q)) \rightarrow q$
2. $((p \rightarrow q) \wedge (q \rightarrow r)) \rightarrow (p \rightarrow r)$
3. $(p \rightarrow (q \rightarrow r)) \leftrightarrow ((p \wedge q) \rightarrow r)$

Pour chacune des formules :

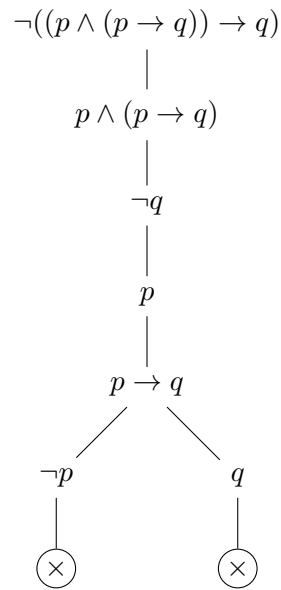
- placez **la négation de la formule** à la racine du tableau sémantique (méthode de Beth);
- appliquez les **règles** α (non-branchantes) et β (branchantes);
- signalez la **fermeture des branches** dès qu'un littéral et sa négation apparaissent sur la même branche;
- concluez si toutes les branches se ferment, la formule est **valide**; sinon fournissez un **contre exemple** à partir d'une branche ouverte.

Formule 1 :

$$(p \wedge (p \rightarrow q)) \rightarrow q$$

Négation :

$$\neg((p \wedge (p \rightarrow q)) \rightarrow q) \equiv (p \wedge (p \rightarrow q)) \wedge \neg q$$



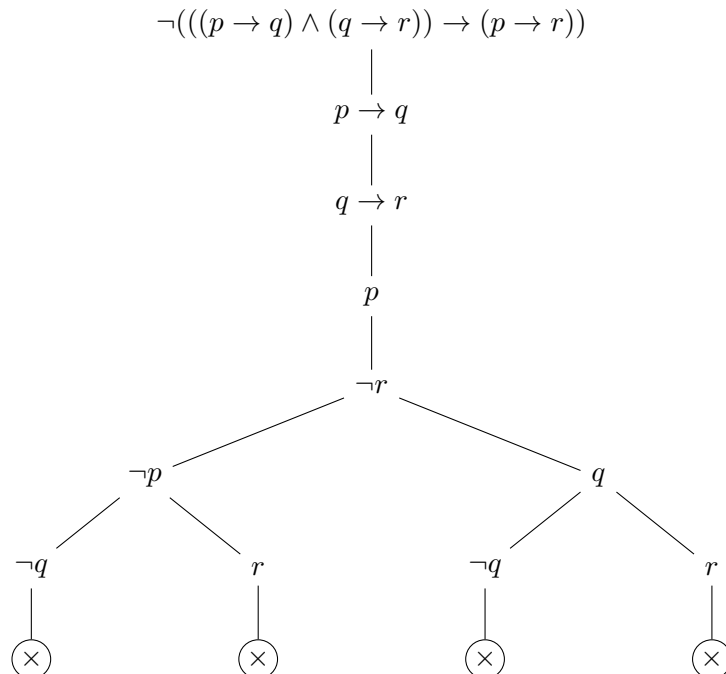
Toutes les branches sont fermées donc la formule 1 est une tautologie.

Formule 2 :

$$((p \rightarrow q) \wedge (q \rightarrow r)) \rightarrow (p \rightarrow r)$$

Négation :

$$\neg(((p \rightarrow q) \wedge (q \rightarrow r)) \rightarrow (p \rightarrow r)) \equiv (p \rightarrow q) \wedge (q \rightarrow r) \wedge \neg(p \rightarrow r) \equiv (p \rightarrow q) \wedge (q \rightarrow r) \wedge (p \wedge \neg r)$$



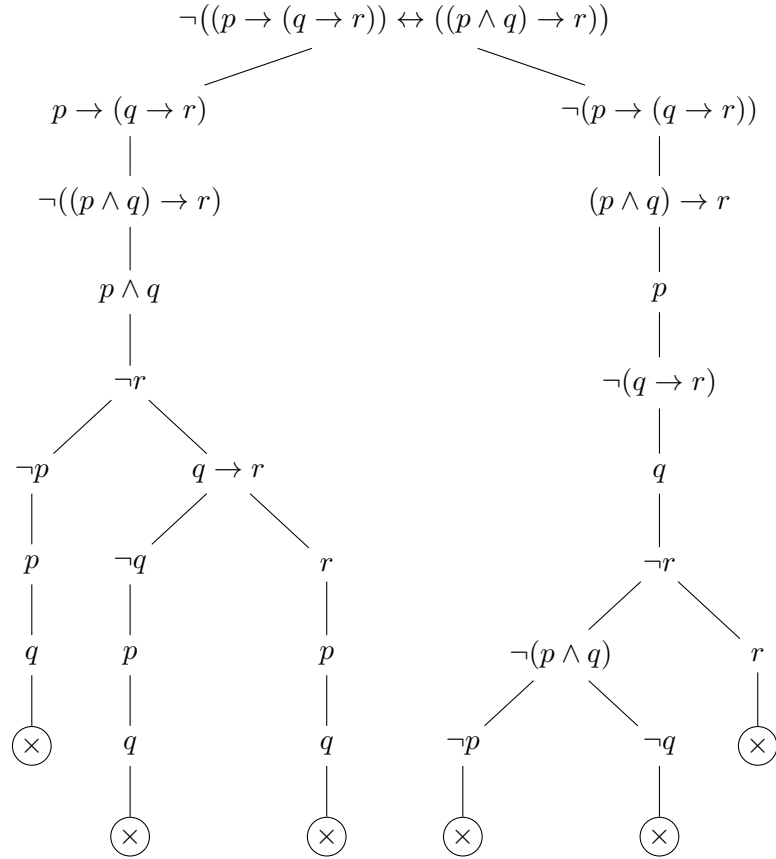
Toutes les branches sont fermées donc $\neg(((p \rightarrow q) \wedge (q \rightarrow r)) \rightarrow (p \rightarrow r))$ est contradictoire d'où la formule 2 est une tautologie.

Formule 3 :

$$(p \rightarrow (q \rightarrow r)) \leftrightarrow ((p \wedge q) \rightarrow r)$$

Négation : $\neg((p \rightarrow (q \rightarrow r)) \leftrightarrow ((p \wedge q) \rightarrow r))$

$$\equiv (p \rightarrow (q \rightarrow r)) \wedge \neg((p \wedge q) \rightarrow r) \quad \vee \quad \neg(p \rightarrow (q \rightarrow r)) \wedge ((p \wedge q) \rightarrow r)$$



Toutes les branches sont fermées donc $\neg((p \rightarrow (q \rightarrow r)) \leftrightarrow ((p \wedge q) \rightarrow r))$ est une contradiction, d'où la formule 3 est une Tautologie.

1.8 Connecteurs logiques et langage naturel

Le langage naturel est souvent ambigu ; la traduction en logique doit lever ces ambiguïtés. Dans ce cours, sauf mention contraire, “ou” est pris au sens *inclusif*.

Correspondances usuelles.

Français	Formel	Commentaire
“ p et q ”	$p \wedge q$	Conjonction.
“ p ou q ”	$p \vee q$	<i>Ou</i> inclusif.
“ p ou bien q (mais pas les deux)”	$(p \vee q) \wedge \neg(p \wedge q)$	Ou exclusif (XOR).
“si p alors q ”	$p \rightarrow q$	Faux ssi p vraie et q fausse.
“ p seulement si q ”	$p \rightarrow q$	q est condition <i>nécessaire</i> de p .
“ q si p ”	$p \rightarrow q$	Même lecture que “si p alors q ”.
“ q est suffisant pour p ”	$q \rightarrow p$	q suffit à garantir p .
“ q est nécessaire pour p ”	$p \rightarrow q$	Sans q , p impossible.
“ p ssi q ”	$p \leftrightarrow q$	Équivalence logique.
“ p à moins que q ”	$\neg q \rightarrow p$	Équivaut à $q \vee p$.
“ni p ni q ”	$\neg p \wedge \neg q$	$\neg(p \vee q)$.

Pièges fréquents.

- “ou” peut être *exclusif* dans la langue courante ; en cas de doute, préciser la lecture.
- “parce que”, “donc” n’induisent pas *automatiquement* une implication formelle valide.
- “ p seulement si q ” $\neq$ “ p si q ” : le premier est $p \rightarrow q$, le second est $q \rightarrow p$.

Exemples (traduction).

1. “Il pleut ou il vente, mais pas les deux.” $\rightsquigarrow (P \vee V) \wedge \neg(P \wedge V)$.
2. “Accès autorisé seulement si authentifié.” $\rightsquigarrow A \rightarrow H$.
3. “Si l’alarme sonne, alors la porte se verrouille.” $\rightsquigarrow S \rightarrow L$.

Vérification sémantique (exemple). Vérifier $p \Rightarrow (p \vee q)$:

p	q	$p \vee q$	$p \rightarrow (p \vee q)$
V	V	V	V
V	F	V	V
F	V	V	V
F	F	F	V

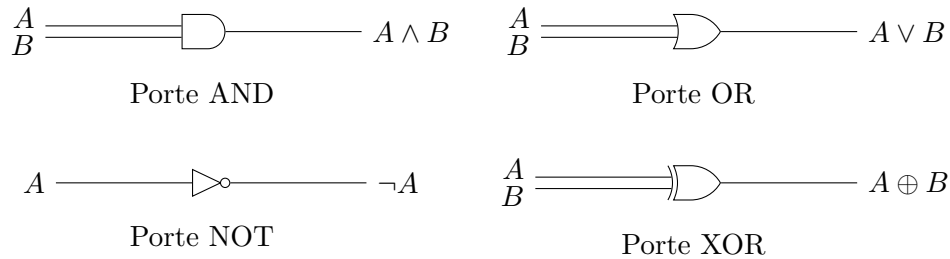
La formule est une tautologie.

1.9 Logique propositionnelle et circuits électroniques

La logique propositionnelle trouve une application remarquable dans la conception des circuits électroniques. Cette correspondance entre formules logiques et circuits numériques constitue le fondement de l'informatique moderne et de l'électronique numérique.

1.9.1 Portes Logiques Fondamentales

Chaque connecteur logique correspond à une porte électronique spécifique :



A	B	$A \wedge B$	$A \vee B$	$\neg A$	$A \oplus B$
0	0	0	0	1	0
0	1	0	1	1	1
1	0	0	1	0	1
1	1	1	1	0	0

TABLE 1.2 – Tables de vérité des portes logiques fondamentales

1.9.2 Méthodologie de synthèse d'un circuit combinatoire

La synthèse d'un circuit combinatoire consiste à concevoir le circuit à partir d'un énoncé décrivant son rôle ou les fonctions à réaliser. Pour mener cette synthèse de manière structurée, les étapes suivantes doivent être suivies :

1. **Analyser le fonctionnement du système** afin de comprendre le comportement attendu.
2. **Identifier les variables d'entrée et les variables de sortie** du circuit.
3. **Établir la table de vérité** en fonction des relations logiques entre entrées et sorties.
4. **Déduire les expressions algébriques des sorties** à partir de la table de vérité.
5. **Simplifier les expressions logiques** (en utilisant l'algèbre de Boole ou les cartes de Karnaugh).
6. **Réaliser le schéma logique (logigramme)** avec un nombre minimal de portes logiques.

1.9.3 Optimisation et simplification : méthode de Karnaugh

La méthode de Karnaugh permet de simplifier une fonction logique en regroupant les minterms (lignes où la fonction vaut 1) dans une carte organisée suivant le code de Gray. Le code de Gray est un codage binaire dans lequel deux mots successifs ne diffèrent que par un seul bit. Il est utilisé dans les cartes de Karnaugh afin de garantir que deux cases adjacentes ne diffèrent que par une seule variable logique, ce qui rend possible la simplification des fonctions logiques par regroupement.

Les groupements doivent être rectangulaires, adjacents et de taille puissance de deux (1, 2, 4, 8, ...). Chaque groupe correspond à un terme simplifié obtenu en éliminant les variables qui changent de valeur à l'intérieur du groupe.

Exemple : $F(A, B, C) = \sum m(1, 2, 3, 5, 7)$.

		BC				
		00	01	11	10	
A	0	0	1	1	1	C $\overline{AB}$
	1	0	1	1	0	

À partir des groupements :

- le groupe de quatre cases ($m1, m3, m5, m7$) correspond à C (seul C est constant),
- le groupe de deux cases ($m2, m3$) correspond à $\overline{AB}$ (avec $A = 0, B = 1$).

Expression simplifiée :

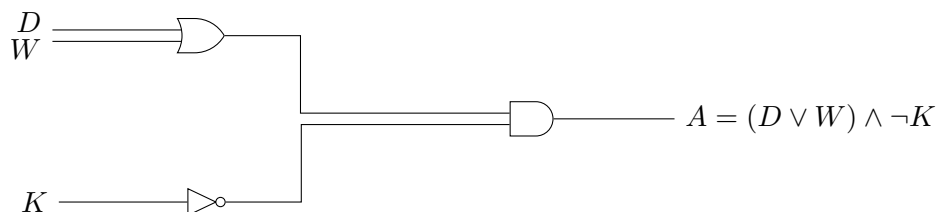
$$F = C + \overline{AB}.$$

1.9.4 Exemple 1 : Circuit d'Alarme

Considérons un système d'alarme qui se déclenche selon la logique suivante :

- D : Détecteur de mouvement activé
- W : Fenêtre ouverte détectée
- K : Clé d'activation présente
- A : Alarme (sortie)

L'alarme se déclenche si (le détecteur OU la fenêtre sont activés) ET que la clé n'est PAS présente.



1.9.5 Exemple 2 : Vote majoritaire

Problème : Concevoir un circuit de vote majoritaire à 3 entrées. Le circuit doit retourner 1 si au moins 2 des 3 entrées valent 1.

Étape 1 - Table de vérité :

A	B	C	Sortie S
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Étape 2 - Forme normale disjonctive : $S = (\neg A \wedge B \wedge C) \vee (A \wedge \neg B \wedge C) \vee (A \wedge B \wedge \neg C) \vee (A \wedge B \wedge C)$

Étape 3 - Simplification par la méthode de Karnaugh :

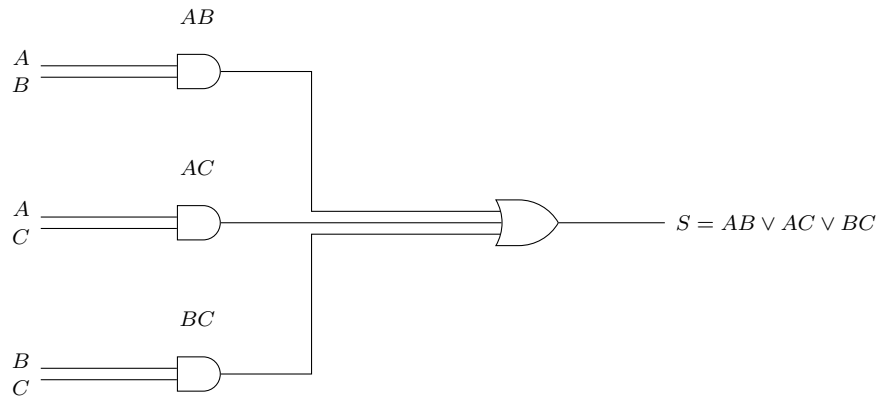
Les minterms correspondant à $S = 1$ sont : m_3, m_5, m_6, m_7 . On les place dans une carte de Karnaugh à trois variables (A, B, C).

		00	01	11	10	
		BC				
	0	0	0	1	0	AC
A						AB
	1	0	1	1	1	BC

On obtient ainsi les trois termes simplifiés :

$$S = AB \vee AC \vee BC.$$

Étape 4 - Circuit résultant :



1.10 Exercices

Exercice 1.1:

1. Montrer que $p \leftrightarrow q \equiv (p \rightarrow q) \wedge (q \rightarrow p)$.
2. Donner la FNC de $(p \vee q) \rightarrow r$.

Exercice 1.2:

Trouver une formule logique φ dont la table de vérité est :

p	q	r	φ
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Exercice 1.3:

Sachant que x et y sont vrais, et z est faux, déterminer à l'aide de l'arbre syntaxique les valeurs de vérité des propositions suivantes :

1. $(x \vee (y \wedge z)) \wedge (y \vee z)$
2. $((y \rightarrow x) \vee \neg(x \leftrightarrow y)) \wedge (z \wedge \neg x)$

Bonne révision

Exercice 1.4:

1. Soit le connecteur logique **barre** ($|$) défini par la table de vérité suivante :

a	b	$a b$
0	0	1
0	1	1
1	0	1
1	1	0

Donner une formule en **forme normale conjonctive (FNC)** et une formule en **forme normale disjonctive (FND)** de $a | b$.

2. Le connecteur $|$ est-il **commutatif** ? Est-il **associatif** ?
 3. Donner les tables de vérité des expressions suivantes :

$$a | a, \quad (a | a) | a, \quad (a | b) | (b | a)$$

4. Donner des formules équivalentes à :

$$\top \text{ (vrai)}, \quad \perp \text{ (faux)}, \quad \neg a, \quad a \cdot b, \quad a \vee b, \quad a \rightarrow b$$

Exercice 1.5:

1. Construire l'arbre syntaxique de la formule :

$$p \vee b \leftrightarrow \neg p \vee \neg q$$

2. Existe-t-il une **valuation** rendant cette formule vraie ? Si oui, laquelle ? Sinon, justifier.
 3. **Utiliser la méthode des arbres sémantiques** (tableaux de vérité analytiques) pour déterminer si la formule suivante est valide :

$$\neg(p \vee q) \leftrightarrow \neg p \cdot \neg q$$

Exercice 1.6:

Considérons le raisonnement suivant :

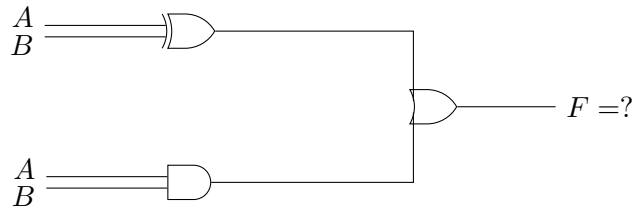
- Si le meurtre a eu lieu le jour, alors le meurtrier est l'ami de la victime.
- Le médecin légiste affirme que le meurtre a eu lieu la nuit.
- Donc, l'ami de la victime ne peut pas être le meurtrier.

Questions :

1. Reformuler les trois énoncés en langage logique et formaliser le raisonnement.
2. Le raisonnement est-il **logiquement valide** ? Justifier.

Exercice 1.7:

Analyser le circuit suivant et établir sa fonction logique :



1.11 Solutions

Exercice 1.1

1. Table de vérité de $(p \leftrightarrow q) \leftrightarrow ((p \rightarrow q) \wedge (q \rightarrow p))$

p	q	$p \leftrightarrow q$	$p \rightarrow q$	$q \rightarrow p$	$(p \rightarrow q) \wedge (q \rightarrow p)$	$(p \leftrightarrow q) \leftrightarrow ((p \rightarrow q) \wedge (q \rightarrow p))$
V	V	V	V	V	V	V
V	F	F	F	V	F	V
F	V	F	V	F	F	V
F	F	V	V	V	V	V

La dernière colonne est toujours vraie : la formule est donc une tautologie, ce qui établit l'équivalence selon la définition

- 2.

$$(p \vee q) \rightarrow r$$

On élimine l'implication :

$$(p \vee q) \rightarrow r \equiv \neg(p \vee q) \vee r$$

On applique la loi de De Morgan :

$$\neg(p \vee q) \equiv \neg p \wedge \neg q$$

Donc :

$$\neg(p \vee q) \vee r \equiv (\neg p \wedge \neg q) \vee r$$

On distribue $\vee$ sur $\wedge$:

$$(\neg p \wedge \neg q) \vee r \equiv (\neg p \vee r) \wedge (\neg q \vee r)$$

Ainsi, la forme normale conjonctive de $(p \vee q) \rightarrow r$ est :

$$\boxed{(\neg p \vee r) \wedge (\neg q \vee r)}.$$

Exercice 1.2

On repère dans la table les lignes où $\varphi = 1$:

p	q	r	φ	
0	0	0	1	$\longrightarrow \neg p \wedge \neg q \wedge \neg r$
0	0	1	1	$\longrightarrow \neg p \wedge \neg q \wedge r$
0	1	0	0	
0	1	1	1	$\longrightarrow \neg p \wedge q \wedge r$
1	0	0	0	
1	0	1	0	
1	1	0	0	
1	1	1	1	$\longrightarrow p \wedge q \wedge r$

On obtient la somme des mintermes :

$$\begin{aligned} \varphi \equiv & (\neg p \wedge \neg q \wedge \neg r) \vee (\neg p \wedge \neg q \wedge r) \\ & \vee (\neg p \wedge q \wedge r) \vee (p \wedge q \wedge r). \end{aligned}$$

Simplification :

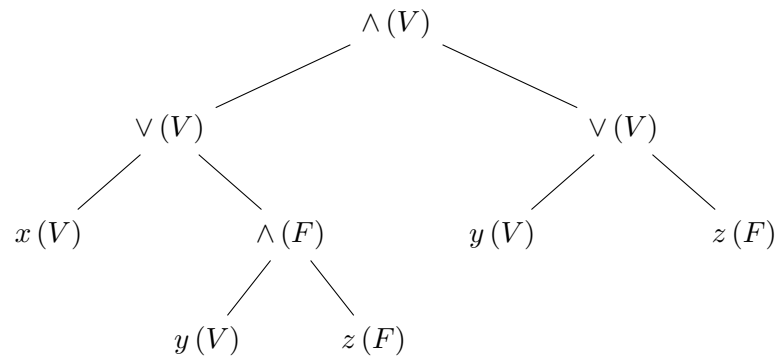
$$\varphi \equiv (\neg p \wedge \neg q) \vee (q \wedge r).$$

$$\boxed{\varphi(p, q, r) = (\neg p \wedge \neg q) \vee (q \wedge r)}.$$

Exercice 1.3

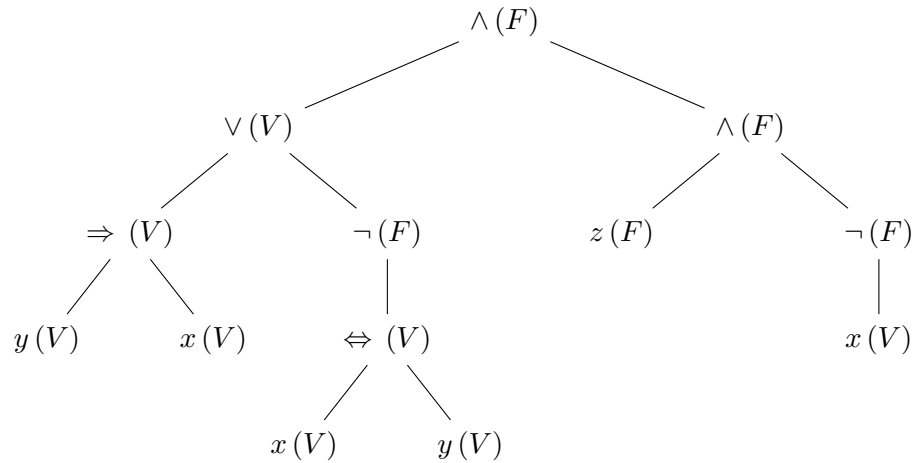
On suppose x et y vraies, et z fausse : $x = V$, $y = V$, $z = F$.

1. Formule $(x \vee (y \wedge z)) \wedge (y \vee z)$.



Ainsi, $x \vee (y \wedge z) = V$ et $y \vee z = V$, donc la formule vaut $V \wedge V = \text{Vrai}$.

2. Formule $((y \Rightarrow x) \vee \neg(x \Leftrightarrow y)) \wedge (z \wedge \neg x)$.



la partie droite $z \wedge \neg x$ vaut F, donc la formule entière vaut Faux.

Exercice 1.4

1. Formulation de $a \mid b$ en formes normales

D'après sa table de vérité, $a \mid b$ est vrai sauf lorsque $a = 1$ et $b = 1$. Ainsi :

$$a \mid b \equiv \neg(a \wedge b)$$

Forme Normale Conjonctive (FNC) : $a \mid b \equiv (\neg a \vee \neg b)$

Forme Normale Disjonctive (FND) :

$$a \mid b \equiv (\neg a \wedge \neg b) \vee (\neg a \wedge b) \vee (a \wedge \neg b)$$

2. Propriétés du connecteur

$$a \mid b \equiv \neg(a \wedge b) \equiv \neg(b \wedge a) \equiv b \mid a \quad \Rightarrow \quad \text{commutatif}$$

Non-associativité du connecteur $\mid$.

Par définition, le connecteur $\mid$ vérifie :

$$a \mid b \equiv \neg(a \wedge b).$$

On calcule d'abord :

$$\begin{aligned} (a \mid b) \mid c &= \neg((a \mid b) \wedge c) \\ &= \neg(\neg(a \wedge b) \wedge c) \\ &\equiv (a \wedge b) \vee \neg c. \end{aligned}$$

De même,

$$\begin{aligned} a \mid (b \mid c) &= \neg(a \wedge (b \mid c)) \\ &= \neg(a \wedge \neg(b \wedge c)) \\ &\equiv \neg a \vee (b \wedge c). \end{aligned}$$

Prenons la valuation :

$$a = 1, \quad b = 1, \quad c = 0.$$

Alors :

$$(a \mid b) \mid c = 1 \quad \text{et} \quad a \mid (b \mid c) = 0.$$

Ainsi,

$$(a \mid b) \mid c \neq a \mid (b \mid c),$$

donc le connecteur $\mid$ **n'est pas associatif**.

3. Tables de vérité simplifiées

(a) Tables de vérité

— Pour $a \mid a$:

a	$a \mid a$
0	1
1	0

$$a \mid a \equiv \neg a$$

— Pour $(a \mid a) \mid a$:

a	$a \mid a$	$(a \mid a) \mid a$
0	1	1
1	0	1

$$(a \mid a) \mid a \equiv \top$$

— Pour $(a \mid b) \mid (b \mid a)$:

a	b	$a \mid b$	$b \mid a$	$(a \mid b) \mid (b \mid a)$
0	0	1	1	0
0	1	1	1	0
1	0	1	1	0
1	1	0	0	1

(b) Expressions à l'aide du connecteur $\mid$ uniquement

On rappelle que

$$a \mid b \equiv \neg(a \wedge b).$$

On peut alors construire les connecteurs usuels comme suit :

$$\top \equiv a \mid (a \mid a)$$

$$\perp \equiv \neg\top \equiv (a \mid (a \mid a)) \mid (a \mid (a \mid a))$$

$$\neg a \equiv a \mid a$$

$$a \wedge b \equiv (a \mid b) \mid (a \mid b)$$

$$a \vee b \equiv \neg(\neg a \wedge \neg b) \equiv \neg a \mid \neg b \equiv (a \mid a) \mid (b \mid b)$$

$$a \rightarrow b \equiv \neg a \vee b \equiv \neg(a \wedge \neg b) \equiv a \mid \neg b \equiv a \mid (b \mid b)$$

Exercice 1.5

(a) La formule est :

$$p \vee b \leftrightarrow \neg p \vee \neg q$$

(b) Exemple de valuation satisfaisante :

Soit $p = 1$, $q = 0$ et $b = 1$.

$$p \vee b = 1 \vee 1 = 1 \quad \neg p \vee \neg q = 0 \vee 1 = 1 \Rightarrow 1 \leftrightarrow 1 = 1$$

La formule est donc satisfaisable.

(c) Validité de la formule :

$$\neg(p \vee q) \leftrightarrow (\neg p \wedge \neg q)$$

Il s'agit de l'une des lois de De Morgan. Cette équivalence est toujours vraie. Il s'agit donc d'une tautologie.

Exercice 1.6

Formalisation logique :

$$\{J \rightarrow A, \neg J\} \models \neg A$$

Méthode 1 : par les modèles.

V	J	A	$J \rightarrow A$	$\neg J$	$\neg A$
V1	0	0	1	1	1
V2	0	1	1	1	0
V3	1	0	0	0	1
V4	1	1	1	0	0

On constate que :

$$\|\{J \rightarrow A, \neg J\}\| = \{V1, V2\}$$

et

$$\|\neg A\| = \{V1, V3\}$$

il est clair que : $\|\{J \rightarrow A, \neg J\}\| \not\subseteq \|\neg A\|$ d'où le raisonnement est non valable.

Méthode 2 : par tableaux de Beth (arbres sémantiques).

Pour tester la validité du raisonnement

$$J \rightarrow A, \neg J \models \neg A,$$

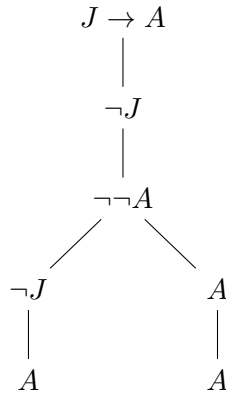
On vérifie la validité de l'implication $J \rightarrow A \wedge \neg J \rightarrow \neg A$ on construit un tableau sémantique sur l'ensemble :

$$\{J \rightarrow A, \neg J, \neg \neg A\}$$

(c'est-à-dire : les prémisses vraies et la conclusion niée).

L'implication se décompose selon la règle β :

$$J \rightarrow A \Rightarrow \neg J \mid A.$$



Chaque branche contient un ensemble de littéraux cohérent :

$$\{\neg J, A\} \quad \text{dans les deux cas,}$$

et ne comporte aucune contradiction du type X et $\neg X$.

Ainsi, aucune branche ne se ferme : le tableau est **ouvert**, ce qui signifie que l'ensemble

$$\{J \rightarrow A, \neg J, \neg\neg A\}$$

n'est pas contradictoire et donc la formule initiale n'est pas valide, d'où

Le raisonnement n'est pas logiquement valide.

Exercice 1.7

On note les sorties intermédiaires :

$$X = A \oplus B \quad \text{et} \quad Y = A \wedge B.$$

La sortie du circuit est donc :

$$F = X \vee Y = (A \oplus B) \vee (A \wedge B).$$

On rappelle que l'addition exclusive (XOR) s'écrit :

$$A \oplus B \equiv (A \wedge \neg B) \vee (\neg A \wedge B).$$

Ainsi,

$$F = (A \wedge \neg B) \vee (\neg A \wedge B) \vee (A \wedge B).$$

On simplifie en regroupant les termes contenant A :

$$\begin{aligned} F &= (A \wedge \neg B) \vee (A \wedge B) \vee (\neg A \wedge B) \\ &= A \wedge (\neg B \vee B) \vee (\neg A \wedge B) \\ &= A \wedge \top \vee (\neg A \wedge B) \\ &= A \vee (\neg A \wedge B). \end{aligned}$$

On utilise ensuite la distributivité (ou facteur commun) :

$$A \vee (\neg A \wedge B) \equiv (A \vee \neg A) \wedge (A \vee B) \equiv \top \wedge (A \vee B) \equiv A \vee B.$$

On en conclut que la fonction réalisée par le circuit est :

$$\boxed{F(A, B) = A \vee B.}$$



Chapitre 2

Système formel de la la logique des propositions

À l'issue de ce chapitre, l'étudiant sera capable de :

- Comprendre la notion de **système formel propositionnel** et ses composantes (alphabet, axiomes, règles d'inférence).
- Appliquer le **système de Hilbert** pour établir des démonstrations logiques.
- Mettre en œuvre la **déduction naturelle** pour prouver la validité d'une formule à partir de prémisses.
- Utiliser un outil de vérification de preuves pour analyser automatiquement la dérivabilité d'une formule.

2.1 Introduction

Jusqu'au chapitre précédent, nous nous sommes occupés de la logique propositionnelle : la syntaxe et la sémantique. Le premier volet permet de déterminer les expressions logiques qui sont syntaxiquement correctes. Le deuxième volet permet de déterminer la valeur de vérité de ces expressions bien formées. Dans ce chapitre, nous abordons le troisième volet de la logique : les systèmes de preuves, ou *théorie des preuves*.

2.2 Système Formel Propositionnel

Définition 2.1 (Système formel propositionnel). Un *système formel propositionnel* (CFP) [1] est un quadruplet $(\Sigma, \mathcal{F}, A, \mathcal{R})$ où :

- Σ : l'alphabet, composé de variables propositionnelles $p, q, r, s, \dots$
- $\mathcal{F}$: l'ensemble des *formules bien formées* (fbfs), défini inductivement par :
 - (a) Toute variable propositionnelle est une fbf.
 - (b) Si A et B sont deux formules bien formées, alors $\neg A$, $(A \wedge B)$, $(A \vee B)$, $(A \leftrightarrow B)$ et $(A \rightarrow B)$ sont aussi des fbfs.
 - (c) Seules les formules engendrées par les règles (1) et (2) sont des fbfs.
- A : un ensemble d'axiomes.
- $\mathcal{R}$: un ensemble de *règles d'inférence* de la forme :

$$\frac{\text{prémisse}}{\text{conclusion}}$$

stipulant que si la formule prémisse est vraie, alors la formule conclusion est vraie.

Il existe plusieurs *calculs propositionnels*, appelés aussi *systèmes de preuves*, qui se distinguent par la forme des axiomes et des règles d'inférence. Nous étudierons dans les sections suivantes deux d'entre eux : le *système à la Hilbert* et la *déduction naturelle* (DN).

2.3 Approche à la Hilbert

Définition 2.2 (Système à la Hilbert). Un **système à la Hilbert** est un calcul propositionnel formel défini par :

Axiomes : 3 axiomes :

$$\text{Ax1} : a \rightarrow (b \rightarrow c)$$

$$\text{Ax2} : (a \rightarrow (b \rightarrow c)) \rightarrow ((a \rightarrow b) \rightarrow (a \rightarrow c))$$

$$\text{Ax3} : (\neg a \rightarrow \neg b) \rightarrow (b \rightarrow a)$$

Règle d'inférence : Un calcul à la Hilbert possède une seule règle d'inférence, dite **règle du Modus Ponens** :

$$\frac{A \quad A \rightarrow B}{B} \quad [\text{MP}]$$

Cette règle stipule que si A est valide et que l'implication $A \rightarrow B$ est valide, alors on peut déduire que B est valide.

Règle de substitution. Pour des fins de preuve, on utilise aussi une règle spéciale dite *règle de substitution*. La sémantique derrière cette règle est la suivante : à partir d'une formule valide B (tautologie), contenant une variable propositionnelle p , on

obtient aussi une formule valide en remplaçant les occurrences de p par une formule quelconque A .

2.3.1 Démonstration

Définition 2.3 (Démonstration). Une **démonstration** est une suite $A_1, A_2, \dots, A_n$, telle que chaque A_i est soit un axiome, soit une conclusion d'une règle d'inférence, obtenue à partir de deux éléments A_j et A_k avec $j, k \leq i$ [1, 3].. La dernière formule A_n de la suite est appelée **théorème**.

Exemple 2.1. Problème : Est-ce que $\vdash A \rightarrow A$?

- (a) $A \rightarrow (A \rightarrow A) \rightarrow A$ (Ax1 + Substitution)
- (b) $(A \rightarrow (A \rightarrow A) \rightarrow A) \rightarrow ((A \rightarrow (A \rightarrow A)) \rightarrow (A \rightarrow A))$ (Ax2 + Substitution)
- (c) $(A \rightarrow (A \rightarrow A)) \rightarrow (A \rightarrow A)$ (MP 1,2)
- (d) $A \rightarrow (A \rightarrow A)$ (Ax1 + Substitution)
- (e) $A \rightarrow A$ (MP 3,4)

2.3.2 Dédution

Définition 2.4 (Définition 3.2.3). Une $\mathcal{F}$ -dédution est une suite $A_1, A_2, \dots, A_n$, telle que chaque A_i est soit :

- (a) un axiome,
- (b) une hypothèse (appartenant à $\mathcal{F}$),
- (c) ou une conclusion obtenue par application d'une règle d'inférence à partir de deux éléments A_j, A_k avec $j, k \leq i$.

On dit que A_n est *déductible* à partir de $\mathcal{F}$ et on note :

$$\mathcal{F} \vdash A_n$$

Exemple 2.2 (Exemple 3.2.2). Vérifions si $A, B \rightarrow (A \rightarrow C) \vdash B \rightarrow C$.

- (a) $(B \rightarrow (A \rightarrow C)) \rightarrow ((B \rightarrow A) \rightarrow (B \rightarrow C))$ (Ax2 + Sub)
- (b) $B \rightarrow (A \rightarrow C)$ (Prémisse)
- (c) $A \rightarrow (B \rightarrow A)$ (Ax1 + Sub)
- (d) A (Prémisse)
- (e) $B \rightarrow A$ (MP 3,4)
- (f) $(B \rightarrow A) \rightarrow (B \rightarrow C)$ (MP 1,2)
- (g) $B \rightarrow C$ (MP 5,6)

2.4 Dédution Naturelle

La méthode des tableaux sémantiques ne convient pas pour étudier la satisfaisabilité d'une formule si la taille de la formule ou le nombre de variables propositionnelles est important. La **dédution naturelle**, proposée par *Gerhard Gentzen* (1909–1945), est une formalisation des règles évidentes permettant de faire du raisonnement mathématique pas à pas.

Elle permet de répondre à la question de la *dérivabilité* d'une formule F , appelée *conclusion*, à partir d'un ensemble de formules $F_1, \dots, F_n$, appelées *prémisses*, que nous notons :

$$F_1, \dots, F_n \vdash F$$

Cette dernière expression est appelée **séquent**.

Remarque 3.3.1.

- $\perp$: faux, $\top$: vrai.
- $\top \equiv \perp \rightarrow \perp$, $\neg A \equiv A \rightarrow \perp$.
- $A \leftrightarrow B \equiv (A \rightarrow B) \wedge (B \rightarrow A)$.

Remarque 3.3.2. Pour chaque symbole de connecteur, il existe une *règle d'introduction* et une *règle d'élimination*.

2.4.1 Règles de Conjonction

2.4.1.1 Règle d'Introduction.

Cette règle, notée $\wedge I$, signifie que si nous avons une preuve de A et une preuve de B , alors nous avons une preuve de $A \wedge B$.

$$\frac{A \quad B}{A \wedge B} (\wedge I)$$

2.4.1.2 Règles d'Élimination.

Les règles d'élimination $\wedge E_1$ et $\wedge E_2$ permettent de déduire, à partir d'une conjonction de deux formules, que l'une ou l'autre est valide.

$$\frac{A \wedge B}{A} (\wedge E_1) \qquad \frac{A \wedge B}{B} (\wedge E_2)$$

2.4.2 Règles de Disjonction

2.4.2.1 Règles d'Introduction.

La disjonction (notée $\vee$) exprime qu'au moins une des deux propositions est vraie. Les deux règles d'introduction, notées $\vee I_1$ et $\vee I_2$, stipulent que :

- Si l'on dispose d'une preuve de A , alors on peut conclure $A \vee B$ ($\vee I_1$).
- Si l'on dispose d'une preuve de B , alors on peut conclure $A \vee B$ ($\vee I_2$).

Formellement :

$$\frac{A}{A \vee B} (\vee I_1) \qquad \frac{B}{A \vee B} (\vee I_2)$$

2.4.2.2 Règle d'Élimination.

La règle d'élimination de la disjonction (notée $\vee E$) formalise le principe de **preuve par cas** couramment utilisé en mathématiques. Pour déduire une conclusion C à partir d'une disjonction $A \vee B$:

- (a) On suppose A vraie et, à partir de cette hypothèse, on démontre C .
- (b) On suppose B vraie et, à partir de cette hypothèse, on démontre C .

Si C est prouvée dans les deux cas, on peut conclure que C est vraie sans condition.

Formellement :

$$\frac{A \vee B \quad [A] \dots C \quad [B] \dots C}{C} (\vee E)$$

où les crochets $[A]$ et $[B]$ indiquent que A et B sont des hypothèses temporaires, valides uniquement dans leur sous-preuve respective.

2.4.3 Règles d'Implication

2.4.3.1 Règle d'Introduction.

Cette règle, notée $\rightarrow I$, est fondamentale : elle exprime que si, à partir de l'hypothèse A , nous pouvons déduire B , alors nous pouvons conclure que l'implication $A \rightarrow B$ est valide, en levant l'hypothèse A .

Formellement :

$$\frac{[A] \dots B}{A \rightarrow B} (\rightarrow I)$$

où $[A]$ indique que A est une hypothèse temporaire, valable uniquement dans la sous-preuve menant à B .

2.4.3.2 Règle d'Élimination.

La règle d'élimination de l'implication, notée $\rightarrow E$, correspond à la célèbre *règle de Modus Ponens* du système à la Hilbert. Elle stipule que si A est vrai et que $A \rightarrow B$ est vrai, alors B est nécessairement vrai.

Formellement :

$$\frac{A \quad A \rightarrow B}{B} (\rightarrow E)$$

2.4.4 Règles de Négation

2.4.4.1 Règle d'Introduction.

Cette règle, notée $\neg I$, exprime que si, à partir de l'hypothèse A , on arrive à une contradiction (notée $\perp$), alors on peut conclure que $\neg A$ est vraie. Elle formalise l'idée que si l'hypothèse A mène à l'absurde, alors A doit être fausse.

Formellement :

$$\frac{[A] \dots \perp}{\neg A} (\neg I)$$

2.4.4.2 Règle d'Élimination.

Cette règle, notée $\neg E$, traduit le principe de non-contradiction : on ne peut pas avoir simultanément A et $\neg A$. Si ces deux formules apparaissent ensemble, on en déduit directement une contradiction.

Formellement :

$$\frac{A \quad \neg A}{\perp} (\neg E)$$

2.4.5 Règles Spéciales

2.4.5.1 EFQ (Ex Falso Quodlibet).

« Du faux, tout. » Cette règle exprime que, dès qu'une contradiction est obtenue (notée $\perp$), toute formule A peut être dérivée.

$$\frac{\perp}{A} (\text{EFQ})$$

2.4.5.2 RAA (reductio ad absurdum ou Réduction à l'absurde).

Cette règle élimine la double négation : si $\neg\neg A$ est dérivable, alors A l'est également.

$$\frac{\neg\neg A}{A} (\text{RAA})$$

Exemple 2.3 (Exemple 3.3.1). **Problème** : Peut-on démontrer que C est une conséquence logique des prémisses suivantes :

$$A \vee B, \quad A \rightarrow C, \quad B \rightarrow C \vdash C ?$$

Démonstration par déduction naturelle :

- (a) $A \vee B$ (Prémisse)
- (b) $A \rightarrow C$ (Prémisse)
- (c) $B \rightarrow C$ (Prémisse)
- (d) Supposons A (Hypothèse)
- (e) C ($\Rightarrow$ E sur 2 et 4)
- (f) Supposons B (Hypothèse)
- (g) C ($\Rightarrow$ E sur 3 et 6)
- (h) C ($\vee$ E sur 1, 4–5 et 6–7)

Ainsi, C est bien une conséquence logique des trois prémisses.

Exemple 2.4. Travail à faire : Utilisez les règles de *déduction naturelle* pour prouver chacune des conséquences logiques suivantes. Pour chaque question, commencez par analyser les prémisses, identifiez le but à atteindre, et justifiez chaque étape par une règle d'introduction ou d'élimination adaptée.

- (a) Montrer que :

$$(p \rightarrow q \rightarrow r) \vdash (p \wedge q \rightarrow r)$$

Quels sous-objectifs intermédiaires pourriez-vous formuler pour atteindre r ?

- (b) Montrer que :

$$(p \rightarrow (q \wedge r)) \vdash (p \rightarrow q) \wedge (p \rightarrow r)$$

Comment utiliserez-vous la règle d'élimination de la conjonction ici ?

- (c) Montrer que :

$$(p \rightarrow q) \vdash (\neg q \rightarrow \neg p)$$

Reconnaissez-vous la forme d'une contraposée dans cet énoncé ?

- (d) Montrer que :

$$(p \wedge q) \vee (p \wedge r) \vdash p \wedge (q \vee r)$$

Par quelle règle pouvez-vous passer d'une disjonction de conjonctions à une conjonction contenant une disjonction ?

Idée générale de la preuve et schématisation. Dans cette preuve, nous cherchons à établir la validité de la conclusion à partir d'un ensemble de prémisses en appliquant les règles de la déduction naturelle. La structure logique de l'argument est représentée sous forme d'un arbre orienté vers le bas : les **nœuds bleus** correspondent aux prémisses de départ (formules acceptées comme vraies dès le début), les **nœuds rouges** indiquent les hypothèses temporaires introduites pour appliquer certaines règles (par exemple, l'élimination de la disjonction), et les **nœuds verts** représentent les conclusions obtenues à chaque étape par application des règles de déduction. La lecture de l'arbre se fait du haut vers le bas : à chaque nœud, on applique une règle logique pour produire de nouvelles formules, jusqu'à atteindre la conclusion finale située en bas de l'arbre. Cette représentation visuelle permet de suivre pas à pas le raisonnement, de repérer les sous-preuves et de distinguer clairement les étapes qui reposent sur des hypothèses locales de celles qui s'appuient uniquement sur les prémisses initiales.

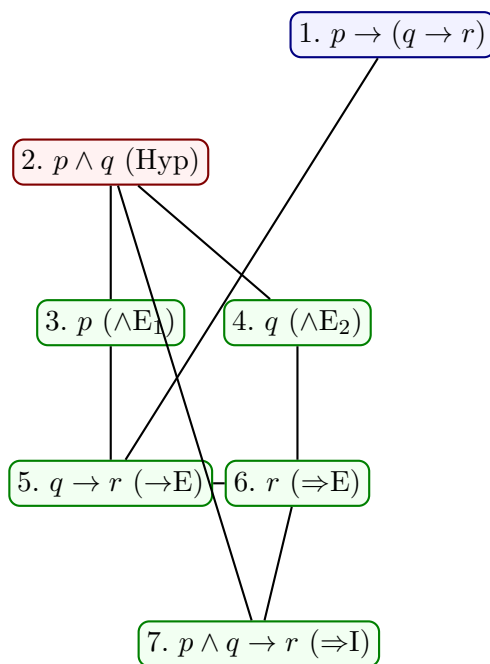


FIGURE 2.1 – Schéma de preuve en déduction naturelle.

Preuve. 1. $p \rightarrow (q \rightarrow r)$	(Prémisse)
2. Supposons $p \wedge q$	(Hypothèse pour $\Rightarrow I$)
3. p	($\wedge E_1$ sur 2)
4. q	($\wedge E_2$ sur 2)
5. $q \rightarrow r$	($\Rightarrow E$ sur 1 et 3)
6. r	($\Rightarrow E$ sur 5 et 4)
7. $p \wedge q \rightarrow r$	($\Rightarrow I$ sur 2-6) □

Preuve. 1. $p \rightarrow (q \wedge r)$ (Prémisse)
 2. Supposons p (Hypothèse pour $\Rightarrow$ I)
 3. $q \wedge r$ ($\Rightarrow$ E sur 1 et 2)
 4. q ($\wedge$ E₁ sur 3)
 5. $p \rightarrow q$ ($\Rightarrow$ I sur 2–4)
 6. Supposons p (Hypothèse pour $\Rightarrow$ I)
 7. $q \wedge r$ ($\Rightarrow$ E sur 1 et 6)
 8. r ($\wedge$ E₂ sur 7)
 9. $p \rightarrow r$ ($\Rightarrow$ I sur 6–8)
 10. $(p \rightarrow q) \wedge (p \rightarrow r)$ ($\wedge$ I sur 5 et 9) $\square$

Preuve. 1. $p \rightarrow q$ (Prémisse)
 2. Supposons $\neg q$ (Hypothèse pour $\Rightarrow$ I)
 3. Supposons p (Hypothèse pour $\neg$ I)
 4. q ($\Rightarrow$ E sur 1 et 3)
 5. $\perp$ ($\neg$ E sur 2 et 4)
 6. $\neg p$ ($\neg$ I sur 3–5)
 7. $\neg q \rightarrow \neg p$ ($\Rightarrow$ I sur 2–6) $\square$

Preuve. 1. $(p \wedge q) \vee (p \wedge r)$ (Prémisse)
 2. Supposons $p \wedge q$ (Hypothèse pour $\vee$ E)
 3. p ($\wedge$ E₁ sur 2)
 4. q ($\wedge$ E₂ sur 2)
 5. $q \vee r$ ($\vee$ I₁ sur 4)
 6. $p \wedge (q \vee r)$ ($\wedge$ I sur 3 et 5)
 7. Supposons $p \wedge r$ (Hypothèse pour $\vee$ E)
 8. p ($\wedge$ E₁ sur 7)
 9. r ($\wedge$ E₂ sur 7)
 10. $q \vee r$ ($\vee$ I₂ sur 9)
 11. $p \wedge (q \vee r)$ ($\wedge$ I sur 8 et 10)
 12. $p \wedge (q \vee r)$ ($\vee$ E sur 1, 2–6, 7–11) $\square$

2.5 Outil de construction de preuve

Un outil implémenté en OCAML¹ permet de *saisir des formules* à prouver et de *vérifier/corriger* des preuves en déduction naturelle. Il effectue un contrôle syntaxique,

1. <http://teachinglogic.liglab.fr/DN/>

cherche une preuve (classique ou, lorsque c'est possible, *intuitionniste*—c'est-à-dire sans utiliser la règle RAA), et fournit un *contre-modèle* lorsque la formule n'est pas démontrable. L'interface affiche les preuves dans une fenêtre dédiée, avec lignes *numérotées* et *justifications* de règles, ce qui facilite la comparaison avec la présentation adoptée dans ce cours.

Fonctionnalités principales.

- **Prove Formula** (données : une formule).

Si la formule est mal formée, un message d'erreur est affiché. Si elle est prouvable, la preuve apparaît dans la *Proof Window*. Si une preuve intuitionniste existe (sans RAA), elle est fournie. Si la formule n'est pas prouvable, l'outil produit un *contre-modèle*.

- **Annotate Proof** (données : une formule et une preuve).

En cas d'erreur de syntaxe dans la formule, un message d'erreur est affiché. Si la preuve fournie comporte des erreurs ou ne prouve pas la formule, l'outil signale le problème et ouvre une fenêtre indiquant le *numéro de la ligne* fautive dans la preuve numérotée. Si la preuve est correcte et prouve bien la formule, une nouvelle fenêtre affiche la *preuve justifiée* (règles explicitées).

La figure 2.2 présente une capture d'écran de l'interface de l'outil de déduction naturelle, montrant la preuve automatique de la formule :

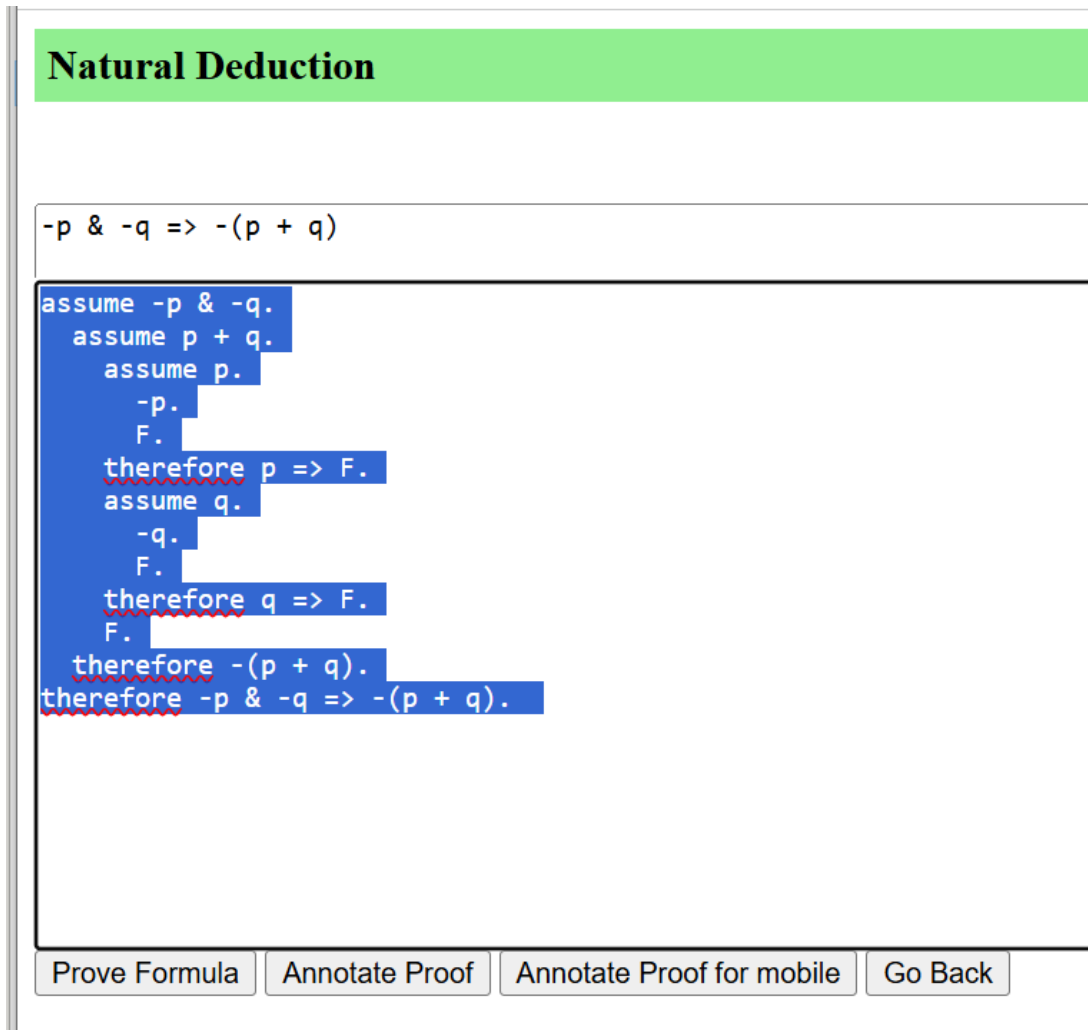
$$\neg p \wedge \neg q \Rightarrow \neg(p \vee q)$$

Cette formule exprime qu'« avoir simultanément la négation de p et la négation de q implique la négation de la disjonction $(p \vee q)$ ».

Le tableau 2.1 met en parallèle la preuve manuelle, rédigée dans le style standard de la déduction naturelle, et la preuve générée automatiquement par l'outil.

Remarque sur les connecteurs utilisés :

- Le symbole « - » désigne la **négation** ($\neg$).
- Le symbole « & » désigne la **conjonction** ($\wedge$).
- Le symbole « + » désigne la **disjonction** ($\vee$).
- La flèche « => » désigne l'**implication** ($\rightarrow$).
- F représente la **contradiction** (faux).

FIGURE 2.2 – Capture montrant la preuve automatique de $\neg p \wedge \neg q \rightarrow \neg(p \vee q)$.

2.6 Correction et complétude du calcul propositionnel

Après avoir introduit les systèmes de preuve formelle, notamment le système de Hilbert et la déduction naturelle, il est essentiel d'établir le lien fondamental entre la preuve syntaxique et la vérité sémantique. Ce lien est garanti par les deux théorèmes suivants.

Théorème 2.1 (Correction). *Pour toute base de formules Γ et toute formule φ , si $\Gamma \vdash \varphi$, alors $\Gamma \models \varphi$.*

Ce théorème signifie que toute formule démontrable dans un système de preuve est nécessairement vraie dans tous les modèles qui satisfont les hypothèses Γ .

Théorème 2.2 (Complétude). *Pour toute base de formules Γ et toute formule φ , si $\Gamma \models \varphi$, alors $\Gamma \vdash \varphi$.*

Preuve manuelle		Preuve automatique (outil DN)
1. $\neg p \wedge \neg q$	Hyp.	assume $\neg p \ \& \ \neg q$
2. $\neg p$	$\wedge E_1$	assume $p \ + \ q$
3. $\neg q$	$\wedge E_2$	assume p
4. Supposons $p \vee q$	Hyp.	$\neg p$
5. Si p , contradiction	$\neg E$	F
6. Si q , contradiction	$\neg E$	therefore $p \Rightarrow F$
7. Donc $\neg(p \vee q)$	$\neg I$	assume q
8. $\neg p \wedge \neg q \rightarrow \neg(p \vee q)$	$\Rightarrow I$	$\neg q$
		F
		therefore $q \Rightarrow F$
		F
		therefore $\neg(p \ + \ q)$
		therefore $\neg p \ \& \ \neg q \Rightarrow \neg(p \ + \ q)$

TABLE 2.1 – Comparaison entre la preuve manuelle et la preuve automatique produite par l'outil DN.

Ce théorème signifie que toute formule vraie dans tous les modèles admet une démonstration formelle dans les systèmes de preuve présentés.

On en déduit l'équivalence fondamentale suivante entre preuve et vérité :

$$\Gamma \vdash \varphi \iff \Gamma \models \varphi.$$

2.7 Exercices

Exercice 2.1:

Utilisez les règles de *déduction naturelle* pour prouver chacune des conséquences logiques suivantes. Pour chaque question, commencez par analyser les prémisses, identifiez le but à atteindre, et justifiez chaque étape par une règle d'introduction ou d'élimination adaptée [8].

- (a) $(A \vee B), (A \rightarrow C), (B \rightarrow C) \vdash C$
- (b) $(B \rightarrow (D \rightarrow E)), ((E \wedge F) \rightarrow G), (J \rightarrow \neg(F \Rightarrow G)) \vdash B \rightarrow (D \rightarrow \neg J)$
- (c) $\neg(C \rightarrow B), (\neg B \rightarrow D) \vdash D$
- (d) $A \wedge (B \vee C) \vdash (A \wedge B) \vee (A \wedge C)$
- (e) $\vdash (A \rightarrow B) \rightarrow [A \rightarrow (A \vee C)]$

(f) $\vdash [((A \wedge B) \rightarrow C) \wedge \neg C] \rightarrow \neg(A \wedge B)$

(g) $A \rightarrow (B \rightarrow C) \vdash (A \wedge B) \rightarrow C$

(h) $(A \vee B) \wedge \neg B \vdash A$

(i) $A \vee \neg A \vdash \neg\neg(A \vee \neg A)$

(j) $\neg\neg A \vdash A$

(k) $\vdash A \rightarrow (B \rightarrow A)$

Réaliser les preuves dans l'outil de construction de preuves.

Corrections

$(A \vee B), (A \rightarrow C), (B \rightarrow C) \vdash C$

Preuve. 1. $A \vee B$ (Prémisse)
 2. $A \rightarrow C$ (Prémisse)
 3. $B \rightarrow C$ (Prémisse)
 4. Supposons A (Hypothèse pour $\vee E$)
 5. C ($\rightarrow E$ sur 2 et 4)
 6. Supposons B (Hypothèse pour $\vee E$)
 7. C ($\rightarrow E$ sur 3 et 6)
 8. C ($\vee E$ sur 1, 4–5 et 6–7) $\square$

$(B \rightarrow (D \rightarrow E)), ((E \wedge F) \rightarrow G), (J \rightarrow \neg(F \rightarrow G)) \vdash B \rightarrow (D \rightarrow \neg J)$

Preuve. 1. $B \rightarrow (D \rightarrow E)$ (Prémisse)
 2. $(E \wedge F) \rightarrow G$ (Prémisse)
 3. $J \rightarrow \neg(F \rightarrow G)$ (Prémisse)
 4. Supposons B (Hypothèse pour $\Rightarrow I$)
 5. $D \rightarrow E$ ($\rightarrow E$ sur 1 et 4)
 6. Supposons D (Hypothèse pour $\rightarrow I$)
 7. E ($\Rightarrow E$ sur 5 et 6)
 8. Supposons J (Hypothèse pour $\neg I$)
 9. $\neg(F \rightarrow G)$ ($\Rightarrow E$ sur 3 et 8)
 10. Supposons F (Hypothèse pour $\rightarrow I$)

11. $E \wedge F$	($\wedge$ I sur 7 et 10)
12. G	($\rightarrow$ E sur 2 et 11)
13. $F \rightarrow G$	($\Rightarrow$ I sur 10–12)
14. $\perp$	($\neg$ E sur 9 et 13)
15. $\neg J$	($\neg$ I sur 8–14)
16. $D \rightarrow \neg J$	($\Rightarrow$ I sur 6–15)
17. $B \rightarrow (D \rightarrow \neg J)$	($\Rightarrow$ I sur 4–16) $\square$

$\neg(C \rightarrow B), (\neg B \rightarrow D) \vdash D$

Preuve. 1. $\neg(C \rightarrow B)$	(Prémisse)
2. $\neg B \rightarrow D$	(Prémisse)
3. Supposons B	(Hypothèse pour $\neg$ I)
4. Supposons C	(Hypothèse pour $\Rightarrow$ I)
5. $C \rightarrow B$	($\Rightarrow$ I sur 4–3)
6. $\perp$	($\neg$ E sur 1 et 5)
7. $\neg B$	($\neg$ I sur 3–6)
8. D	($\rightarrow$ E sur 2 et 7) $\square$

$A \wedge (B \vee C) \vdash (A \wedge B) \vee (A \wedge C)$

Preuve. 1. $A \wedge (B \vee C)$	(Prémisse)
2. A	($\wedge$ E ₁ sur 1)
3. $B \vee C$	($\wedge$ E ₂ sur 1)
4. Supposons B	(Hypothèse pour $\vee$ E)
5. $A \wedge B$	($\wedge$ I sur 2 et 4)
6. $(A \wedge B) \vee (A \wedge C)$	($\vee$ I ₁ sur 5)
7. Supposons C	(Hypothèse pour $\vee$ E)
8. $A \wedge C$	($\wedge$ I sur 2 et 7)
9. $(A \wedge B) \vee (A \wedge C)$	($\vee$ I ₂ sur 8)
10. $(A \wedge B) \vee (A \wedge C)$	($\vee$ E sur 3, 4–6 et 7–9) $\square$

$\vdash (A \rightarrow B) \rightarrow [A \rightarrow (A \vee C)]$

Preuve. 1. Supposons $A \rightarrow B$ (Hypothèse pour $\rightarrow$ I)
 2. Supposons A (Hypothèse pour $\rightarrow$ I)
 3. $A \vee C$ ($\vee$ I₁ sur 2)
 4. $A \rightarrow (A \vee C)$ ($\Rightarrow$ I sur 2–3)
 5. $(A \rightarrow B) \rightarrow [A \rightarrow (A \vee C)]$ ($\rightarrow$ I sur 1–4) $\square$

$\vdash [((A \wedge B) \rightarrow C) \wedge \neg C] \rightarrow \neg(A \wedge B)$

Preuve. 1. Supposons $((A \wedge B) \rightarrow C) \wedge \neg C$ (Hypothèse pour $\rightarrow$ I)
 2. $(A \wedge B) \rightarrow C$ ($\wedge$ E₁ sur 1)
 3. $\neg C$ ($\wedge$ E₂ sur 1)
 4. Supposons $A \wedge B$ (Hypothèse pour $\neg$ I)
 5. C ($\rightarrow$ E sur 2 et 4)
 6. $\perp$ ($\neg$ E sur 3 et 5)
 7. $\neg(A \wedge B)$ ($\neg$ I sur 4–6)
 8. $[((A \wedge B) \rightarrow C) \wedge \neg C] \rightarrow \neg(A \wedge B)$ ($\rightarrow$ I sur 1–7) $\square$

$A \rightarrow (B \rightarrow C) \vdash (A \wedge B) \rightarrow C$

Preuve. 1. $A \rightarrow (B \rightarrow C)$ (Prémisse)
 2. Supposons $A \wedge B$ (Hypothèse pour $\rightarrow$ I)
 3. A ($\wedge$ E₁ sur 2)
 4. B ($\wedge$ E₂ sur 2)
 5. $B \rightarrow C$ ($\rightarrow$ E sur 1 et 3)
 6. C ($\rightarrow$ E sur 5 et 4)
 7. $(A \wedge B) \rightarrow C$ ($\Rightarrow$ I sur 2–6) $\square$

$(A \vee B) \wedge \neg B \vdash A$

Preuve. 1. $(A \vee B) \wedge \neg B$ (Prémisse)
 2. $A \vee B$ ($\wedge$ E₁ sur 1)
 3. $\neg B$ ($\wedge$ E₂ sur 1)
 4. Supposons $\neg A$ (Hypothèse pour $\neg\neg$ E / preuve par l'absurde)

- | | |
|------------------|---|
| 5. Supposons A | (Hypothèse pour $\forall E$) |
| 6. $\perp$ | (Contradiction entre 4 et 5) |
| 7. Supposons B | (Hypothèse pour $\forall E$) |
| 8. $\perp$ | ($\neg E$ sur 3 et 7) |
| 9. $\perp$ | ($\forall E$ sur 2, 5–6 et 7–8) |
| 10. A | ($\neg E$ sur 4–9) $\square$ |

$A \vee \neg A \vdash \neg\neg(A \vee \neg A)$

- Preuve.**
- | | |
|------------------------------------|---|
| 1. $A \vee \neg A$ | (Prémisse) |
| 2. Supposons $\neg(A \vee \neg A)$ | (Hypothèse pour $\neg I$) |
| 3. $\perp$ | ($\neg E$ sur 2 et 1) |
| 4. $\neg\neg(A \vee \neg A)$ | ($\neg I$ sur 2–3) $\square$ |

$\neg\neg A \vdash A$

- Preuve.**
- | | |
|-----------------------|---|
| 1. $\neg\neg A$ | (Prémisse) |
| 2. Supposons $\neg A$ | (Hypothèse pour $\neg I$) |
| 3. $\perp$ | ($\neg E$ sur 1 et 2) |
| 4. A | ($\neg I$ sur 2–3) $\square$ |

$\vdash A \rightarrow (B \rightarrow A)$

- Preuve.**
- | | |
|--------------------------------------|--|
| 1. Supposons A | (Hypothèse pour $\rightarrow I$) |
| 2. Supposons B | (Hypothèse pour $\rightarrow I$) |
| 3. A | (Réitération de 1) |
| 4. $B \rightarrow A$ | ($\rightarrow I$ sur 2–3) |
| 5. $A \rightarrow (B \rightarrow A)$ | ($\rightarrow I$ sur 1–4) $\square$ |



Chapitre 3

Calcul des prédicats

Compétences visées :

- Différencier la logique propositionnelle (LP0) de la logique des prédicats (LP1) [10].
- Comprendre les notions fondamentales : domaine de discours, constantes, variables, symboles de fonction, symboles de prédicat, arité, termes, formules atomiques et formules bien formées (FBF).
- Identifier et distinguer les concepts de variables libres et variables liées, interpréter leur portée, et détecter les erreurs de liaison.
- Construire l'arbre syntaxique d'une formule bien formée, en respectant la hiérarchie des quantificateurs, connecteurs, fonctions et prédicats.
- Modéliser un problème concret en logique des prédicats et le **traduire en énoncé formel** correct (formalisation).
- Comprendre la notion **d'interprétation** d'un langage du premier ordre, évaluer la vérité d'une formule dans un modèle.
- Employer le principe de résolution pour prouver l'insatisfaisabilité ou la dérivabilité d'un ensemble de formules.

La logique des prédicats constitue une extension naturelle et essentielle de la logique propositionnelle [1]. Alors que cette dernière permet de raisonner sur des propositions considérées comme des blocs indivisibles, la logique des prédicats introduit la notion de *variable*, de *prédicat* et de *quantification*, permettant ainsi de formaliser des propriétés générales et des relations entre objets.

Ce formalisme joue un rôle fondamental en mathématiques, notamment dans la définition rigoureuse des structures, des relations, des classes d'équivalence, ainsi que dans l'expression précise des énoncés d'algèbre, d'analyse et de topologie.

Dans ce chapitre, nous présentons successivement la syntaxe et la sémantique de la

logique du premier ordre, puis les principaux outils de manipulation formelle des formules, aboutissant à la méthode de preuve par résolution.

3.1 De la logique propositionnelle à la logique des prédicats (LP0 à LP1)

En logique propositionnelle (LP0), chaque énoncé est traité comme une entité atomique, sans structure interne. Par exemple, les phrases suivantes sont traduites par de simples lettres propositionnelles :

- « L’homme est mortel » est noté p ,
- « Mohamed est l’enseignant de Ali » est noté q .

Cette approche ne permet cependant pas d’exprimer la structure interne des énoncés, ni de généraliser les propriétés à des ensembles d’objets. La logique des prédicats (LP1) remédie à cette limitation en introduisant :

- des *variables*,
- des *prédicats*,
- des *quantificateurs*.

Ainsi, les objets et les relations entre objets deviennent explicitement représentables.

Exemple 1 : « L’homme est mortel »

Cet énoncé exprime une propriété valable pour l’ensemble des individus considérés comme des hommes. On introduit un prédicat unaire $\text{Homme}(x)$ et un prédicat unaire $\text{Mortel}(x)$. L’énoncé se formalise alors par :

$$\forall x (\text{Homme}(x) \Rightarrow \text{Mortel}(x)).$$

Exemple 2 : « Mohamed est l’enseignant de Ali »

Cet énoncé met en jeu une relation entre deux individus. On introduit un prédicat binaire $E(x, y)$ signifiant « x est l’enseignant de y ». La formalisation est :

$$E(M, A),$$

où M et A désignent les constantes associées aux individus *Mohamed* et *Ali*.

3.2 Prédicats à une variable

Soit D un ensemble non vide appelé *domaine de discours*. Un *prédicat à une variable* (ou prédicat unaire) représente une propriété susceptible d’être vraie ou fausse pour chaque élément de D .

Formellement, un prédicat unaire P est une application :

$$P : D \longrightarrow \{0, 1\},$$

où $P(a) = 1$ signifie que la propriété P est vérifiée par l'élément $a \in D$.

On associe à chaque prédicat son *ensemble de vérité* :

$$\llbracket P \rrbracket = \{a \in D \mid P(a) = 1\},$$

de sorte que $P(a)$ est vraie si et seulement si $a \in \llbracket P \rrbracket$.

Exemples sur $\mathbb{N}$

$\text{Carré}(n)$ ssi $\exists k \in \mathbb{N}, n = k^2$, $\text{Petit}(n)$ ssi $n \leq 10$.

On obtient alors : $\llbracket \text{Carré} \rrbracket = \{0, 1, 4, 9, \dots\}$, $\llbracket \text{Petit} \rrbracket = \{0, 1, \dots, 10\}$.

3.3 Prédicats à deux ou plusieurs variables

Les prédicats à une variable permettent d'exprimer des propriétés portant sur un seul objet. Dans de nombreuses situations mathématiques, il est cependant nécessaire de modéliser des *relations entre plusieurs objets*. Cette nécessité conduit naturellement à la notion de prédicat à plusieurs variables, également appelé prédicat n -aire.

Définition. Soient $D_1, \dots, D_n$ des ensembles non vides. Un prédicat n -aire $R(x_1, \dots, x_n)$ est interprété comme une relation au sens ensembliste, c'est-à-dire comme un sous-ensemble du produit cartésien :

$$\llbracket R \rrbracket \subseteq D_1 \times \dots \times D_n.$$

Pour tout $(a_1, \dots, a_n) \in D_1 \times \dots \times D_n$, on a

$$R(a_1, \dots, a_n) \text{ vraie} \quad \text{ssi} \quad (a_1, \dots, a_n) \in \llbracket R \rrbracket.$$

Le cas particulier $n = 2$, correspondant aux prédicats binaires, joue un rôle fondamental : il permet de formaliser les relations entre deux objets (ordre, égalité, parenté, adjacence, etc.).

Exemples de prédicats binaires.

— Sur $\mathbb{N}$:

- $\text{Div}(x, y) : \ll x \text{ divise } y \gg$,
- $x \leq y : \text{relation d'ordre usuel}$.
- Sur un ensemble d'individus D :

$\text{Ens}(t, s)$ signifie « t est l'enseignant de s »

avec $\llbracket \text{Ens} \rrbracket \subseteq D \times D$.

Propriétés classiques des relations binaires. Soit $R \subseteq D \times D$ une relation binaire. On définit :

- Réflexive : $\forall x R(x, x)$,
- Symétrique : $\forall x \forall y (R(x, y) \Rightarrow R(y, x))$,
- Antisymétrique : $\forall x \forall y ((R(x, y) \wedge R(y, x)) \Rightarrow x = y)$,
- Transitive : $\forall x \forall y \forall z ((R(x, y) \wedge R(y, z)) \Rightarrow R(x, z))$.

Ces propriétés permettent de caractériser des notions fondamentales telles que les *relations d'ordre*, les *relations d'équivalence* et les *structures de graphe*.

Combinaisons logiques et interprétation ensembliste. Soient $R, S \subseteq D_1 \times \dots \times D_n$ deux relations n -aires. On a :

$$\llbracket R \wedge S \rrbracket = R \cap S, \quad \llbracket R \vee S \rrbracket = R \cup S, \quad \llbracket \neg R \rrbracket = (D_1 \times \dots \times D_n) \setminus R.$$

Ces correspondances traduisent le calcul logique des prédicats en opérations ensemblistes sur les relations.

Image d'un ensemble par une relation. Soit $R \subseteq D \times E$ et $X \subseteq D$. L'image de X par R est définie par :

$$R[X] = \{ y \in E \mid \exists x \in X, R(x, y) \}.$$

Cette notion joue un rôle central dans l'étude des relations et dans la modélisation des systèmes dynamiques, des graphes et des correspondances fonctionnelles.

3.3.1 Alphabet

L'alphabet de la logique des prédicats est composé des symboles suivants :

- un ensemble de **symboles de prédicats** à 0 ou plusieurs arguments, notés $p, q, r, \dots$ ou encore *Mortel*, *Homme*, $\dots$

- un ensemble dénombrable de **variables d'individus**, notées $x, y, z, \dots$
- un ensemble dénombrable de **symboles de fonctions** à 0 ou plusieurs arguments, notés $f, g, h, \dots$
- les **quantificateurs** : universel $\forall$ et existentiel $\exists$
- les **connecteurs** : *False*, $\neg$, $\wedge$, $\vee$, $\Rightarrow$
- les parenthèses : $(\)$

3.3.2 Signature, Termes, Prédicats, Formules atomiques et Formules

Définition 3.1 (Signature). Une **signature** Σ d'un langage du premier ordre est un triplet (C, F, R) composé de :

- C : l'ensemble des symboles de constantes,
- F : l'ensemble des symboles de fonctions,
- R : l'ensemble des symboles de prédicats.

Exemple 3.1. $\Sigma = (\{0, 1\}, \{s, +\}, \{Impair, Premier, =, <\})$

Définition 3.2 (Terme). Un **terme** est défini inductivement par :

- (a) Les variables et les constantes sont des termes.
- (b) Si f est un symbole de fonction à n arguments et $t_1, t_2, \dots, t_n$ sont des termes, alors $f(t_1, t_2, \dots, t_n)$ est un terme.
- (c) Seules les expressions respectant les règles 1 et 2 sont des termes.

Les applications de fonctions aux termes sont donc elles-mêmes des termes. Par exemple : $Pere(Pere(x))$.

Un **terme clos** est un terme sans variables.

Exemple 3.2. Pour $\Sigma = (\{0, 1\}, \{s, +\}, \{Impair, Premier, =, <\})$, l'expression $+(x, s(y))$ est un exemple de terme, et $+(a, s(b))$ est un terme clos où a et b sont deux constantes de la signature.

Définition 3.3 (Prédicat). Un **prédicat** est une fonction définie d'un domaine D^n vers $\{0, 1\}$, vérifiant :

Si P est un symbole de prédicat n -aire et $t_1, t_2, \dots, t_n$ sont des termes, alors $P(t_1, t_2, \dots, t_n)$ est un prédicat ou **formule atomique** (formule de base).

Exemple 3.3. $Impair(s(n))$, $Premier(n)$ sont des prédicats.

Définition 3.4 (Formule bien formée). Une **formule bien formée** (FBF) est définie par :

- (a) Chaque formule atomique est une formule bien formée.

(b) Si A et B sont des formules bien formées alors :

$$\neg A, \quad (A \wedge B), \quad (A \vee B), \quad (A \rightarrow B), \quad \forall x A, \quad \exists x B$$

sont des formules bien formées.

(c) L'ensemble des formules bien formées est engendré exclusivement par les règles (1) et (2).

Exemple 3.4.

$$\forall x P(x) \rightarrow \left(\exists y \left(Q(f(y), s(0)) \rightarrow P(g(b)) \right) \right), \quad P(g(a))$$

sont des prédicats sur la signature

$$\Sigma = (\{0, 1\}, \{f, g\}, \{P, Q\}).$$

3.3.2.1 Règles de priorité

L'ordre de priorité des opérateurs en logique des prédicats est le suivant :

$$\forall, \exists \quad (\text{plus prioritaire})$$

puis

$$\neg, \quad \wedge, \quad \vee, \quad \rightarrow$$

sauf si la présence de parenthèses indique un autre ordre d'évaluation.

Pour éviter toute ambiguïté, on introduit les parenthèses de la même manière qu'en logique propositionnelle.

Exemple 3.5. — $\neg A \wedge B$ est équivalente à $(\neg A) \wedge B$ et non pas à $\neg(A \wedge B)$.

— Un quantificateur s'applique uniquement à la formule située immédiatement à sa droite ; par exemple :

$$\forall x P(x) \wedge R(x) \equiv (\forall x P(x)) \wedge R(x)$$

3.3.3 Arbre syntaxique

Une formule bien formée (*fbf*) peut être représentée de manière *structurée* à l'aide d'un arbre syntaxique, qui met en évidence la construction hiérarchique de la formule à partir des règles de formation.

Définition 3.5 (Arbre syntaxique). L'**arbre syntaxique** d'une formule bien

formée est un graphe orienté et enraciné où :

- chaque nœud interne correspond à un connecteur logique ou à un quantificateur,
- chaque feuille correspond à une formule atomique.

La racine représente la formule entière et les sous-arbres représentent les sous-formules.

Remarque 3.1: Construction d'un arbre syntaxique

La construction d'un arbre syntaxique suit directement les règles de formation des FBF :

- (a) On part des formules atomiques comme feuilles.
- (b) On applique récursivement les connecteurs et quantificateurs comme opérations sur les sous-formules.

Exemple 3.6. Considérons la formule :

$$\forall x (P(x) \rightarrow \exists y Q(x, y))$$

Son arbre syntaxique est donné par la figure 3.1.

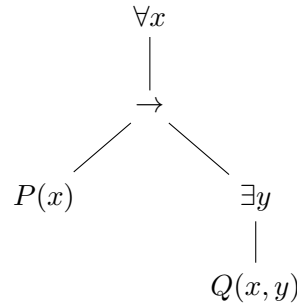


FIGURE 3.1 – Arbre syntaxique d'une formule en LP1

Exemple 3.7. Considérons la formule :

$$\forall x ((P(x) \wedge Q(f(x))) \rightarrow \exists y (R(y) \vee \neg P(y)))$$

Cette formule exprime que, pour tout x , si $P(x)$ et $Q(f(x))$ sont vrais, alors il existe un y tel que $R(y)$ soit vrai ou bien $P(y)$ soit faux.

L'arbre syntaxique de cette formule mettra en évidence :

- Le quantificateur universel $\forall x$ en racine,
- L'implication comme connecteur principal sous le quantificateur,
- La conjonction $(P(x) \wedge Q(f(x)))$ dans l'antécédent,
- Le quantificateur existentiel $\exists y$ dans le conséquent,

— La disjonction $R(y) \vee \neg P(y)$ comme contenu de $\exists y$.

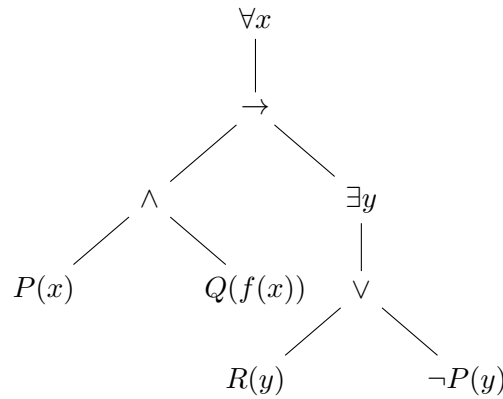


FIGURE 3.2 – Arbre syntaxique d'une *fbf* complexe

3.3.4 Variables libres et variables liées

Définition 3.6. Une occurrence d'une variable x dans une formule F est dite **liée** si elle apparaît dans la **portée** d'un quantificateur $\forall$ ou $\exists$ sur x . Si :

$$F = \forall x A \quad \text{ou} \quad F = \exists x A$$

alors A est appelée la *portée* du quantificateur et toutes les occurrences de x dans A sont liées. Sinon, l'occurrence est dite **libre**.

Définition 3.7. — Une variable est **libre** dans une formule si elle possède au moins une occurrence libre dans la formule.

— Une formule est dite **close** (ou *fermée*) si elle ne possède aucune variable libre.

Les variables liées sont des *variables muettes* : remplacer le nom d'une variable liée par un autre nom de variable ne modifie pas la sémantique de la formule. Par exemple :

$$\forall x A \quad \text{et} \quad \forall h A[h/x]$$

sont sémantiquement équivalentes (où $A[h/x]$ désigne la formule obtenue en remplaçant x par h dans A).

Exemple 3.8. Considérons la formule :

$$(R(x, z) \Rightarrow \forall z (R(y, z) \vee y = z))$$

On observe :

- L'occurrence de x est **libre**.
- La première occurrence de z (dans $R(x, z)$) est **libre**, les deuxième et troisième occurrences (dans $\forall z (R(y, z) \vee y = z)$) sont **liées**. Donc z est libre (car elle a

au moins une occurrence libre).

— La variable y est **libre**.

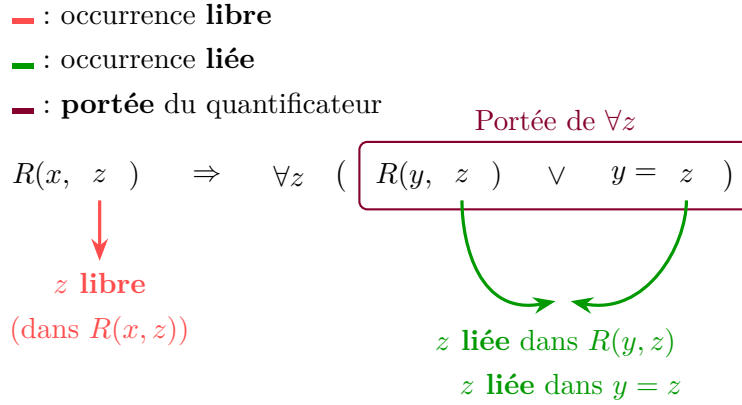


FIGURE 3.3 – Portée des quantificateurs illustrée.

3.3.5 Substitution

Dans les définitions qui suivent, un *littéral* désigne un prédicat ou la négation d'un prédicat. Par exemple : $R(x, f(y))$, $\neg P(x)$.

Définition 3.8 (Substitution). Une **substitution** θ est un ensemble

$$\theta = \{x_1 = t_1, \dots, x_n = t_n\}$$

qui, appliquée à un littéral α , donne un littéral $\alpha\theta$ obtenu en remplaçant simultanément toute occurrence de x_i par t_i dans α .

Une substitution est donc un remplacement d'une variable par un terme (variable ou fonction). On utilise les lettres $\lambda, \mu, \sigma, \theta$ pour désigner une substitution et la lettre ε pour désigner la **substitution vide**.

Notation alternative. On utilise parfois la notation $F[x/y]$ pour indiquer que, dans la formule F , on remplace x par y .

Intérêt. La substitution est particulièrement utile pour :

- la mise sous *forme prénexe* ;
- la *skolémisation* (voir définitions ultérieures).

Exemple 3.9. Soit :

$$\alpha = R(x, g(y), f(c)), \quad \theta = \{y = x, x = b\}$$

alors :

$$\alpha\theta = R(b, g(x), f(c))$$

Autre exemple :

$$\forall x \exists y P(x, y)[x/z] \equiv \forall z \exists y P(z, y)$$

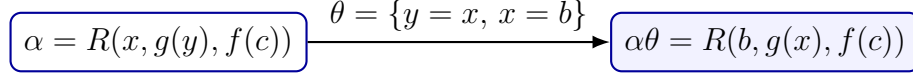


FIGURE 3.4 – Substitution sur un littéral

3.3.6 Unification

Définition 3.9 (Unification). Un **unificateur** de deux littéraux r_1 et r_2 est une substitution θ qui *unifie* r_1 et r_2 , c'est-à-dire qui rend $r_1\theta = r_2\theta$.

Exemple 3.10.

- Les littéraux $P(g(x), z)$ et $P(y, c)$ sont unifiables par $\theta = \{y \mapsto g(x), z \mapsto c\}$.
- Par contre, $P(g(x), a)$ et $P(y, c)$ ne sont pas unifiables.

Exemple 3.11. Les termes $p(x, x, a)$ et $p(b, y, y)$ sont-ils unifiables ?

Réponse : on cherche une substitution θ telle que

$$p(x, x, a)\theta = p(b, y, y)\theta$$

On remarque que pour unifier :

$$x \mapsto b, \quad a \mapsto y$$

mais cela impliquerait $a = y$ et $y = a$ avec a constante, ce qui ne résout pas la répétition de x dans les deux premières positions. Donc ces termes **ne sont pas unifiables**.

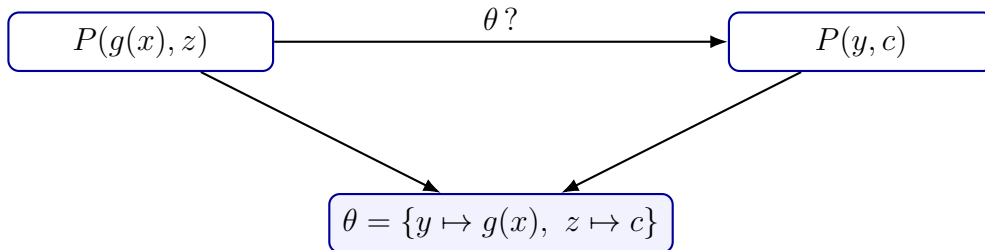


FIGURE 3.5 – Illustration du processus d'unification

3.4 Sémantique de la logique des prédicats

La sémantique d'un langage du premier ordre repose sur la notion *d'interprétation* de sa signature [7]. Celle-ci précise le sens des symboles et la manière d'évaluer les termes et formules.

3.4.1 Interprétation

Définition 3.10 (Interprétation). Une **interprétation** $\mathcal{I}$ d'un langage du premier ordre est un quadruplet :

- (a) Un **domaine de discours** D (ou univers d'interprétation), ensemble non vide d'individus.
- (b) Une **fonction d'évaluation des variables** I_V : ensemble des variables $\rightarrow D$.
- (c) Une **fonction d'interprétation des prédicats** I_P : à chaque prédicat p de n arguments, associe une application $D^n \rightarrow \{0, 1\}$.
- (d) Une **fonction d'interprétation des fonctions et constantes** I_F :
 - à chaque fonction f de n arguments, associe une application $D^n \rightarrow D$;
 - à chaque constante c , associe un élément de D .

Interprétation des termes. Une interprétation $\mathcal{I}$ s'étend naturellement aux termes :

$$I(f(t_1, \dots, t_n)) = I_F(f)(I(t_1), \dots, I(t_n))$$

Un **terme clos** est un terme ne contenant aucune variable.

Exemple 3.12. Soit $\Sigma = (\{0, 1\}, \{s, +\}, \{\text{Impair}, \text{Premier}, =, <\})$. Alors $+(x, s(y))$ est un terme, et $+(a, s(b))$ est un terme clos si a, b sont des constantes de Σ .

Notation. Comme en logique propositionnelle, on note $I \models A$ (ou $I(A) = 1$) pour indiquer que I est un modèle de A .

Exemple 3.13. Soit :

$$A = (\forall x \exists y R(x, y)) \wedge (\forall x \neg R(x, x)) \wedge (\forall x \forall y \forall z ((R(x, y) \wedge R(y, z)) \Rightarrow R(x, z)))$$

et $\Sigma = (\emptyset, \emptyset, \{R^2\})$. Un modèle est donné par :

- $D = \mathbb{N}$ (entiers naturels),
- $I_P(R)(d_1, d_2) = 1$ ssi $d_1 < d_2$.

Cette interprétation satisfait A pour toute affectation des variables I_V .

Exemple 3.14. Pour $F = \forall x P(a, x)$:

- Dans $I_1 = (\mathbb{N}, I_V, \{\leq\}, \{0\})$, F est vraie.

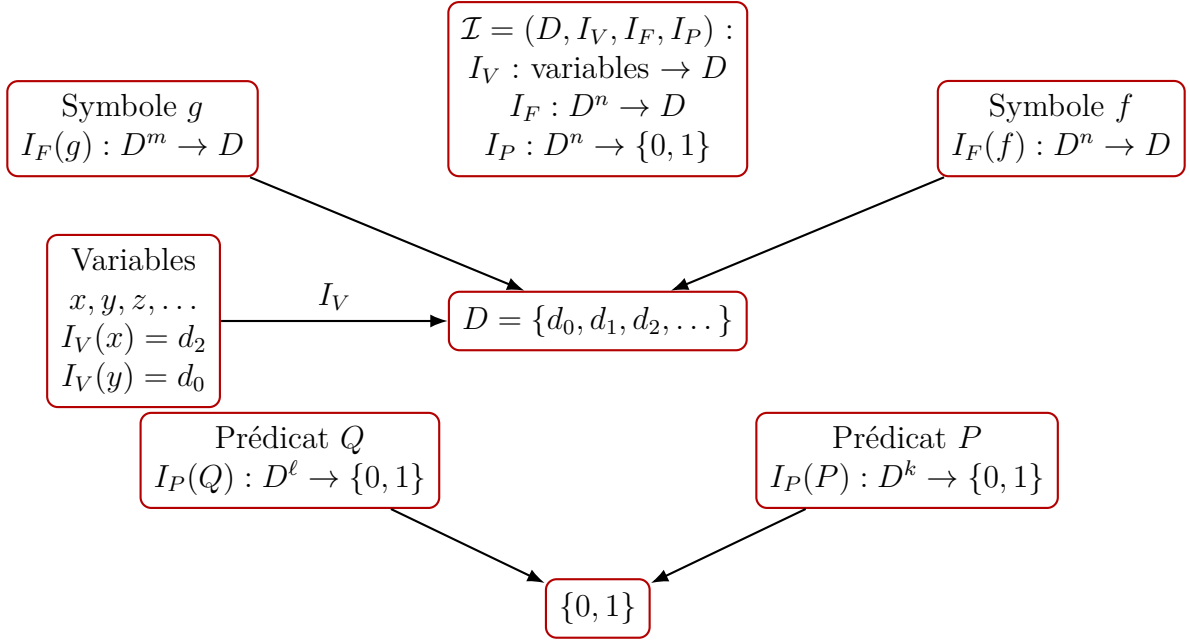


FIGURE 3.6 – Schéma d'une interprétation.

— Dans $I_2 = (\mathbb{Z}, I_V, \{\leq\}, \{0\})$, F est fausse.

3.4.2 Interprétation des formules

L'interprétation $\mathcal{I}$ s'étend aux formules par induction :

$$\begin{aligned}
 I(p(t_1, \dots, t_n)) &= I_P(p)(I(t_1), \dots, I(t_n)) \\
 I(\forall x A) &= 1 \quad \text{sii} \quad \forall \mathcal{I}' \text{ variant de } \mathcal{I} \text{ sur } x, I'(A) = 1 \\
 I(\exists x A) &= 1 \quad \text{sii} \quad \exists \mathcal{I}' \text{ variant de } \mathcal{I} \text{ sur } x, I'(A) = 1
 \end{aligned}$$

Vocabulaire utile.

- **Terme clos** : aucun variable (ex. $f(a) * b$).
- **Formule close** : aucune variable libre (ex. $\forall x P(x)$).
- **Formule valide** : vraie pour toutes interprétations (ex. $\exists x (P(x) \vee \neg P(x))$).
- **Formule satisfiable** : vraie pour au moins une interprétation.
- **Formule insatisfiable (antilogie)** : fausse pour toutes interprétations.

La figure 3.6 présente le schéma général d'une **interprétation** $\mathcal{I} = (D, I_V, I_F, I_P)$ dans un langage du premier ordre.

- **Domaine** D : ensemble non vide d'éléments $(d_0, d_1, \dots)$ constituant l'univers

de discours.

- **Variables** $(x, y, z, \dots)$: interprétées via I_V , qui associe à chaque variable un élément de D .
- **Symboles de fonction** $(f, g, \dots)$: interprétés par I_F comme des applications $D^n \rightarrow D$.
- **Symboles de prédicat** $(P, Q, \dots)$: interprétés par I_P comme des applications $D^n \rightarrow \{0, 1\}$, déterminant la vérité ou la fausseté d'une relation.

3.5 Formulation dans la logique des prédicats

Pour formuler un énoncé en logique des prédicats, il faut distinguer clairement entre *variables*, *constantes*, *fonctions* et *prédicats* :

- Une **variable** représente des objets (éléments) sur lesquels portent les prédicats, et non pas des propositions atomiques.
- Une **constante** représente un objet particulier d'un domaine donné. Par exemple : **Mohamed** et **Amina** peuvent être des constantes dans le domaine des personnes, et π est une constante dans le domaine des réels.

On distingue deux types d'énoncés :

- **Énoncés universels** : les variables représentent tous les objets d'un domaine.
Exemple : Si x est une personne alors elle est mortelle.

$$\forall x(\text{personne}(x) \Rightarrow \text{mortel}(x))$$

- **Énoncés existentiels** : la variable représente un objet particulier du domaine.
Exemple : On peut trouver au moins un entier naturel n tel que $n^2 > 4$.

$$\exists n(\text{entier}(n) \wedge n^2 > 4)$$

Ci-après, quelques formulations d'énoncés en logique du premier ordre :

- (a) Amina est la fille de Nadia :

$$\text{fille}(\text{amina}, \text{nadia})$$

- (b) Tout le monde a une mère :

$$\forall x(\text{personne}(x) \Rightarrow \exists y(\text{personne}(y) \wedge \text{mere}(y, x)))$$

(c) Amina respecte tout le monde :

$$\forall x (\text{personne}(x) \Rightarrow \text{respecte}(\text{amina}, x))$$

(d) Personne ne respecte Amina :

$$\forall x (\text{personne}(x) \Rightarrow \neg \text{respecte}(x, \text{amina}))$$

(e) Une porte est ouverte ou fermée :

$$\forall x (\text{porte}(x) \Rightarrow \text{ouvert}(x) \vee \text{ferme}(x))$$

(f) Pour tout entier, il existe un entier plus grand :

$$\forall x (\text{entier}(x) \Rightarrow \exists y (\text{entier}(y) \wedge \text{plus-grand}(y, x)))$$

(g) Il existe un plus grand entier :

$$\exists x (\text{entier}(x) \wedge \forall y (\text{entier}(y) \Rightarrow \text{plus-grand}(x, y)))$$

3.6 Preuve par Résolution

La résolution [9] est une méthode de preuve ou d'inférence correcte et complète sous certaines conditions (non détaillées ici). Elle constitue une procédure *décidable* permettant de vérifier la satisfaisabilité d'une formule, et repose sur le **principe de réfutation** :

Pour prouver $\Gamma \vdash F$, on montre que $\Gamma \wedge \neg F$ est insatisfaisable.

Autrement dit, on transforme le problème en un ensemble de *clauses*, puis on applique la règle de résolution de façon répétée jusqu'à obtention d'une *clause vide* (notée $\square$ ou $\perp$), ce qui prouve l'insatisfaisabilité.

3.6.1 Règle de résolution

En logique propositionnelle, la règle de *modus ponens* est :

$$a, \quad \neg a \vee b \quad \Rightarrow \quad b$$

Elle peut se reformuler en termes de clauses :

$$\begin{aligned} p \vee L_1 \vee \dots \vee L_n, \quad \neg p \vee M_1 \vee \dots \vee M_k \\ \Rightarrow L_1 \vee \dots \vee L_n \vee M_1 \vee \dots \vee M_k \end{aligned}$$

En logique des prédicats, la règle devient :

$$P(a), \quad \neg P(x) \vee Q(x) \Rightarrow Q(a)$$

Plus généralement :

$$\begin{aligned} A \vee L_1 \vee \dots \vee L_n, \quad \neg B \vee M_1 \vee \dots \vee M_k \\ \Rightarrow L_1\sigma \vee \dots \vee L_n\sigma \vee M_1\sigma \vee \dots \vee M_k\sigma \end{aligned}$$

où σ est un **unificateur** de A et $\neg B$ (substitution rendant les deux littéraux identiques).

Exemple 3.15.

(a) Clauses : $\neg d(x)$ et $d(a)$ Substitution : $x \mapsto a$ Après résolution :

$$\neg d(a), \quad d(a) \Rightarrow \square$$

On obtient la clause vide, donc l'ensemble est insatisfaisable.

3.6.2 Étapes d'une preuve par résolution

1. Mise sous forme universelle close On transforme toutes les formules en :

$$\forall x_1 \dots \forall x_n \varphi(x_1, \dots, x_n)$$

où φ est sans quantificateurs. Cela se fait en deux étapes :

(a) **Forme normale prénexe** : réécriture de la formule sous la forme

$$Q_1x_1 Q_2x_2 \dots Q_nx_n \varphi$$

avec $Q_i \in \{\forall, \exists\}$ et φ sans quantificateur.

(b) **Skolémisation** : élimination des quantificateurs existentiels en introduisant de nouvelles constantes ou fonctions.

2. Mise sous forme clausale On met φ sous *forme normale conjonctive* (FNC). Comme $\forall x(\psi \wedge \psi') \equiv (\forall x.\psi) \wedge (\forall x.\psi')$, on obtient une **conjonction de clauses**, chacune étant une disjonction de littéraux universellement quantifiés. Les variables de clauses différentes sont renommées si nécessaire.

Exemple 3.16. Formules initiales :

$$\forall x (p(x, f(x)) \Rightarrow q(x)), \quad \forall x \forall y P(f(x), f(y)), \quad \neg q(f(a))$$

Forme clausale :

$$\neg p(x, f(x)) \vee q(x), \quad p(f(y), f(z)), \quad \neg q(f(a))$$

3. Application répétée de la règle de résolution On résout les clauses jusqu'à obtenir :

- $\square$: l'ensemble est insatisfaisable, donc $\Gamma \vdash F$ est prouvé.
- Pas de clause vide : la conclusion ne peut pas être inférée.

Nous détaillons les principales étapes de résolution dans les sections suivantes :

3.6.3 Forme prénexe

Définition 3.11. Une formule φ est sous **forme prénexe** si elle est de la forme :

$$Q_1 x_1 Q_2 x_2 \dots Q_n x_n \psi$$

où chaque Q_i est un quantificateur ($\forall$ ou $\exists$), tous les x_i sont distincts, et ψ est sans quantificateur. Toute formule du premier ordre possède une forme prénexe équivalente.

Rappel d'équivalences utiles

$$\exists x \varphi \equiv \exists y \varphi(x/y) \quad (y \text{ nouveau})$$

$$\forall x \varphi \equiv \forall y \varphi(x/y) \quad (y \text{ nouveau})$$

$$\forall x(\varphi \wedge \psi) \equiv (\forall x \varphi) \wedge \psi \quad (x \text{ non libre dans } \psi)$$

Algorithme de mise en forme prénexe

- (a) Éliminer $\Rightarrow, \Leftrightarrow, \text{FALSE}$.
- (b) Appliquer les équivalences de négation et de déplacement de quantificateurs.
- (c) Renommer les variables liées si nécessaire.

(d) Amener tous les quantificateurs en tête de formule.

Exemple 3.17. Formule : $\neg[\forall x \exists y F(u, x, y) \Rightarrow \exists x (\neg \forall y G(y, v) \Rightarrow H(x))]$ Mise en forme prénexe : à calculer suivant les étapes précédentes.

3.6.4 Forme de Skolem

Définition 3.12. Une formule est **universelle** si elle est prénexe et ne contient que des quantificateurs universels. La **Skolémisation** consiste à supprimer les quantificateurs existentiels en introduisant de nouveaux symboles de fonction ou de constante, tout en conservant la satisfaisabilité.

Intuition (Skolémisation)

Idée générale. Dans la formule $\forall x \exists y P(x, y)$, pour *chaque* x il existe *au moins* un y tel que $P(x, y)$ soit vraie. On peut donc supposer qu'il existe une **fonction de choix** f qui fournit un témoin y pour chaque x . Ainsi, on réécrit

$$\forall x P(x, f(x)).$$

Cas général.

$$\forall x_1 \cdots \forall x_n \exists y \varphi(x_1, \dots, x_n, y) \quad \rightsquigarrow \quad \forall x_1 \cdots \forall x_n \varphi(x_1, \dots, x_n, f(x_1, \dots, x_n)).$$

Exemple 3.18. Énoncé. « Tout étudiant est inscrit dans une faculté. »

Modélisation.

$$\forall x \exists y (\text{etudiant}(x) \Rightarrow \text{fac}(y) \wedge \text{inscrit}(x, y)).$$

Skolémisation. On introduit f : pour chaque x , $f(x)$ désigne *une* faculté associée à x :

$$\forall x (\text{etudiant}(x) \Rightarrow \text{fac}(f(x)) \wedge \text{inscrit}(x, f(x))).$$

Exemple 3.19 (Exemple 1.13). $\exists x \forall y (x > y)$ devient, après skolémisation,

$$\forall y (a > y),$$

où a est une **constante de Skolem**.

Exemple 3.20 (Exemple 1.14).

$$\forall x \exists y \forall z \exists u \left(R(x, y) \Rightarrow S(y, z, u) \right).$$

Étapes.

- $\exists y$ dépend de x : on remplace y par $f(x)$.
- $\exists u$ dépend de (x, z) : on remplace u par $g(x, z)$.

On obtient la formule universelle

$$\forall x \forall z \left(R(x, f(x)) \Rightarrow S(f(x), z, g(x, z)) \right).$$

1.7 Forme normale conjonctive

Définition 3.13 (Clause, Forme clausale). Une clause est une disjonction finie de littéraux. Une formule du calcul des prédicats est dite sous forme clausale si elle s'écrit

$$\forall x_1 \cdots \forall x_n \left(C_1 \wedge C_2 \wedge \cdots \wedge C_k \right),$$

où chaque C_i ($1 \leq i \leq k$) est une clause et les x_j sont des variables, renommées au besoin.

Exercice. Mettre sous forme normale *clausale* la formule :

$$\left(\exists x P(x) \Rightarrow \forall y P(y) \right) \wedge \left(\forall x (P(x) \vee Q(x)) \right) \wedge \left(\forall z (\exists x \neg Q(x) \Rightarrow \forall y P(y)) \right).$$

1.7.1 Équivalences utiles pour la mise en forme

Fermetures.

$$\models A \equiv \models \forall x A \quad (\text{fermeture universelle})$$

$$A \text{ est satisfiable} \equiv \exists x A \quad (\text{fermeture existentielle})$$

Quantification répétée sur la même variable.

$$\vdash (Q_1 x. Q_2 x. A) \equiv Q_2 x. A.$$

Quantification sans occurrence libre. Si A n'a aucune occurrence libre de x :

$$\models (Qx. A) \equiv A, \quad \models A[x/t] \equiv A.$$

Renommage de variables liées.

$$\models (Qx.A) \equiv Qy.A[x/y] \quad (y \text{ nouvelle}).$$

Deux cas de distribution (autorisés).

$$\models (\forall x A) \wedge (\forall x B) \equiv \forall x (A \wedge B), \quad \models (\exists x A) \vee (\exists x B) \equiv \exists x (A \vee B).$$

Élargissement de portée (variable non libre). Si x n'est pas libre dans B :

$$\models (Qx.A) \wedge B \equiv Qx.(A \wedge B), \quad \models (Qx.A) \vee B \equiv Qx.(A \vee B).$$

Attention. Ces équivalences *échouent* si x a des occurrences libres dans B . Par exemple,

$$(\exists x p(x)) \wedge \neg p(x) \not\equiv \exists x (p(x) \wedge \neg p(x)).$$

1.8 Exemple

La formule suivante est-elle satisfaisable ?

$$\forall x (p(x, f(x)) \Rightarrow q(x)) \wedge \forall x \forall y p(f(x), f(y)) \wedge \exists x \neg q(f(x)) ?$$

Solution

$$\varphi = \forall x (p(x, f(x)) \Rightarrow q(x)) \wedge \forall x \forall y p(f(x), f(y)) \wedge \exists x \neg q(f(x)).$$

Nous allons étudier la satisfaisabilité de φ en utilisant la méthode de *résolution*.

Étape 1 : élimination de l'implication

On remplace $A \Rightarrow B$ par $\neg A \vee B$. La première conjonction devient :

$$\forall x (\neg p(x, f(x)) \vee q(x)).$$

Ainsi, la formule φ est équivalente à :

$$\varphi_1 = \forall x (\neg p(x, f(x)) \vee q(x)) \wedge \forall x \forall y p(f(x), f(y)) \wedge \exists x \neg q(f(x)).$$

Étape 2 : mise sous forme prénexe et Skolemisation

Les deux premières composantes sont déjà universellement quantifiées, la troisième contient un quantificateur existentiel.

On standardise les variables pour éviter toute confusion :

$$\forall u (\neg p(u, f(u)) \vee q(u)) \wedge \forall v \forall w p(f(v), f(w)) \wedge \exists z \neg q(f(z)).$$

On applique ensuite la *Skolemisation* sur le quantificateur existentiel $\exists z$ en introduisant une constante de Skolem a :

$$\exists z \neg q(f(z)) \rightsquigarrow \neg q(f(a)).$$

Après suppression des quantificateurs universels (implicites en forme clausale), on obtient le système de formules suivant :

$$\neg p(u, f(u)) \vee q(u), \quad p(f(v), f(w)), \quad \neg q(f(a)).$$

Étape 3 : mise sous forme clausale

Chaque formule est déjà sous forme disjonctive. On la réécrit sous forme de clauses :

- $C_1 : \neg p(u, f(u)) \vee q(u)$
- $C_2 : p(f(v), f(w))$
- $C_3 : \neg q(f(a))$

Les variables u, v, w sont universelles (implicites).

Étape 4 : résolution

L'objectif est de dériver une contradiction (la clause vide $\square$).

1. Instanciation de la clause C_2 . À partir de $C_2 : p(f(v), f(w))$, on choisit une instanciation adaptée :

$$v := a, \quad w := f(a),$$

ce qui donne la clause (instance de C_2) :

$$C'_2 : p(f(a), f(f(a))).$$

2. Instanciation de la clause C_1 . À partir de $C_1 : \neg p(u, f(u)) \vee q(u)$, on choisit :

$$u := f(a),$$

ce qui donne :

$$C'_1 : \neg p(f(a), f(f(a))) \vee q(f(a)).$$

3. Résolution entre C'_2 et C'_1 . On résout C'_2 et C'_1 sur le littéral $p(f(a), f(f(a)))$:

$$\begin{array}{l} C'_2 : p(f(a), f(f(a))) \\ C'_1 : \neg p(f(a), f(f(a))) \vee q(f(a)) \\ \hline C_4 : q(f(a)) \end{array}$$

4. Résolution entre C_4 et C_3 . On dispose aussi de :

$$C_3 : \neg q(f(a)).$$

On résout C_4 et C_3 sur $q(f(a))$:

$$\begin{array}{l} C_4 : q(f(a)) \\ C_3 : \neg q(f(a)) \\ \hline \square \end{array}$$

L'obtention de la *clause vide* $\square$ signifie que l'ensemble des clauses est **insatisfaisable**.

3.7 Exercices

Exercice 3.1:

Soit la signature $\Sigma = (\{a, b\}, \{f^1, g^2\}, \{p^1, q^2\})$. Parmi les expressions suivantes, lesquelles sont des **formules du premier ordre** ?

- (a) $p(f(a)) \wedge f(a)$
- (b) $p(g(a, y)) \vee q(g(x), f(b))$
- (c) $\forall y (q(g(x, y), f(a)) \rightarrow p(f(y)))$
- (d) $\forall y q(g(x, y), f(b)) \rightarrow (\exists x p(g(f(a), h(x, y))) \wedge q(a, a))$

Exercice 3.2:

Pour chaque symbole utilisé dans la formule du premier ordre suivante, préciser s'il s'agit d'une **fonction** ou d'un **prédicat** et donner son **arité**. On suppose que x , y et z sont des variables.

$$((j(x, y) \rightarrow (\neg i(x, y, b(z, y)) \vee d(x))) \wedge e(f(x, g(y))))$$

Exercice 3.3:

- (a) Soit f la fonction unaire représentant « le frère de », et $F(x, y)$ le prédicat binaire « x est le frère de y ». Donner le sens de l'expression :

$$F(f(Ahmed), Ahmed)$$

- (b) Donner la négation de la proposition suivante : « Les étudiants assimilent bien les cours ». En utilisant les prédicats suivants :

$$a(x, y) : x \text{ assimile le cours } y, \quad e(x) : x \text{ est un étudiant}, \quad c(y) : y \text{ est un cours.}$$

Exercice 3.4:

Traduire les énoncés suivants en logique des prédicats (LP1) :

- (a) « Pour tout nombre réel x , il existe un réel y tel que $y = x^2$ »
- (b) « Pour tout nombre naturel n , n est soit pair, soit impair »
- (c) « Pour tout rationnel, il existe un entier qui est plus grand »
- (d) « Il existe un entier qui est le plus grand entier »

Exercice 3.5:

Pour chaque formule, identifier la nature de chaque symbole (fonction, prédicat, constante, variable).

- (a) $P(a)$
- (b) $Q(f(x), b)$
- (c) $R(g(a, x), h(b, y))$
- (d) $\forall x P(x) \rightarrow Q(f(a))$
- (e) $\exists y (R(g(a, y)) \wedge \neg S(h(b)))$
- (f) $\forall x P(f(g(x), h(a, b)), i(j(x), k(c)))$

Exercice 3.6:

Dans la formule suivante, déterminer les occurrences libres et liées :

$$\forall x(P(x, y, z) \rightarrow \exists y(Q(y) \wedge \forall z(R(x, y, z) \vee S(a, b, z))))$$

Exercice 3.7:

En utilisant les prédicats :

$$\text{Parent}(x, y), \quad \text{Homme}(x), \quad \text{Femme}(x),$$

écrire les contraintes suivantes :

- (a) La relation de parenté est unidirectionnelle.
- (b) Un individu ne peut pas être à la fois homme et femme.

Exercice 3.8:

Déterminer si les termes suivants sont unifiables :

$$R(x, y, a) \quad \text{et} \quad R(b, f(y), y).$$

Exercice 3.9:

Formaliser les énoncés suivants :

- (a) a est une personne.
- (b) b est un chat.
- (c) a est propriétaire de b .
- (d) Toute personne qui possède un chat est une personne.
- (e) Toute personne possédant un chat l'aime.
- (f) a est allergique aux chats.
- (g) Toute personne allergique à un chat ne l'aime pas.

3.8 Solutions

Exercice 1

On rappelle la signature :

$$\Sigma = (\{a, b\}, \{f^1, g^2\}, \{p^1, q^2\})$$

- (a) $p(f(a)) \wedge f(a)$ **Incorrecte.** $p(f(a))$ est une formule, mais $f(a)$ est un terme, et non une formule. L'opérateur $\wedge$ doit relier deux formules.
- (b) $p(g(a, y)) \vee q(g(x), f(b))$ **Incorrecte.** $g(x)$ est incorrect car g attend deux arguments (g^2). Il manque une variable ou constante.
- (c) $\forall y (q(g(x, y), f(a)) \rightarrow p(f(y)))$ **Correcte.** - $g(x, y) \rightarrow$ terme (fonction binaire), - $f(a) \rightarrow$ terme, - $q(g(x, y), f(a)) \rightarrow$ formule, - $p(f(y)) \rightarrow$ formule, - donc l'implication et la quantification sont bien formées.
- (d) $\forall y q(g(x, y), f(b)) \rightarrow (\exists x p(g(f(a), h(x, y))) \wedge q(a, a))$ **Incorrecte.** Le symbole h n'appartient pas à la signature (non défini dans Σ).

Exercice 2

Formule :

$$((j(x, y) \rightarrow (\neg i(x, y, b(z, y)) \vee d(x))) \wedge e(f(x, g(y))))$$

Symbole	Type	Arité
j	Prédicat	2
i	Prédicat	3
d	Prédicat	1
e	Prédicat	1
f	Fonction	1
g	Fonction	1
b	Fonction	2
x, y, z	Variables	—

Exercice 3

- (a) f désigne la fonction unaire « le frère de », et $F(x, y)$ le prédicat « x est le frère de y ».

$$F(f(\text{Ahmed}), \text{Ahmed})$$

Se lit : *Le frère de Ahmed est le frère de Ahmed.* Autrement dit, « *Le frère d'A Ahmed est bien son frère.* »

(b) Proposition à nier :

« Les étudiants assimilent bien les cours »

Utilisation des prédicats :

$$\forall x \forall y ((e(x) \wedge c(y)) \rightarrow a(x, y))$$

Négation :

$$\neg \forall x \forall y ((e(x) \wedge c(y)) \rightarrow a(x, y))$$

$$\equiv \exists x \exists y \neg ((e(x) \wedge c(y)) \rightarrow a(x, y))$$

$$\equiv \exists x \exists y (e(x) \wedge c(y) \wedge \neg a(x, y))$$

Il existe au moins un étudiant et au moins un cours tels que cet étudiant n'assimile pas ce cours.

Exercice 4

- (a) $\forall x (\text{Réal}(x) \rightarrow \exists y (\text{Réal}(y) \wedge y = x^2))$
- (b) $\forall n (\text{Nat}(n) \rightarrow (\text{Pair}(n) \vee \text{Impair}(n)))$
- (c) $\forall x (\text{Rat}(x) \rightarrow \exists y (\text{Entier}(y) \wedge y > x))$
- (d) $\exists n (\text{Entier}(n) \wedge \forall m (\text{Entier}(m) \rightarrow m \leq n))$

Exercice 5

- (a) P prédicat unaire ; a constante.
- (b) Q prédicat ; f fonction unaire ; x variable ; b constante.
- (c) R prédicat ; g, h fonctions ; a, b constantes ; x, y variables.
- (d) x variable ; P, Q prédicats ; f fonction ; a constante.
- (e) y variable ; R, S prédicats ; g, h fonctions ; a, b constantes.
- (f) x variable ; P prédicat ; f, g, h, i, j, k fonctions ; a, b, c constantes.

Exercice 6

$$\forall x(P(x, y, z) \rightarrow \exists y(Q(y) \wedge \forall z(R(x, y, z) \vee S(a, b, z)))).$$

Occurrences liées.

- $\forall x$ lie toutes les occurrences de x dans la formule : x est lié dans $P(x, y, z)$ et dans $R(x, y, z)$.
- $\exists y$ lie les occurrences de y dans $Q(y)$ et dans $R(x, y, z)$.
- $\forall z$ lie les occurrences de z dans $R(x, y, z)$ et $S(a, b, z)$.

Occurrences libres.

- L'occurrence de y dans $P(x, y, z)$ est libre.
- L'occurrence de z dans $P(x, y, z)$ est libre.

Ainsi :

Variables liées : x , Variables libres : y, z .

Exercice 7

- (a) $\forall x \forall y (\text{Parent}(x, y) \rightarrow \neg \text{Parent}(y, x))$
- (b) $\forall x (\text{Homme}(x) \rightarrow \neg \text{Femme}(x))$

Exercice 8

Aucune substitution ne peut satisfaire simultanément $x = b$, $y = f(y)$ et $y = a$. La contrainte $y = f(y)$ est récursive, donc les deux termes ne sont pas unifiables.

Exercice 9

Prédicats : P, C, O, L, A.

- $P(x)$: x est une personne.
- $C(x)$: x est un chat.
- $O(x, y)$: x est le propriétaire de y .
- $L(x, y)$: x aime y .
- $A(x)$: x est allergique aux chats.

1. $P(a)$
2. $C(b)$
3. $O(a, b)$
4. $\forall x \forall y (\neg O(x, y) \vee \neg C(y) \vee P(x))$
5. $\forall x \forall y (\neg P(x) \vee \neg O(x, y) \vee \neg C(y) \vee L(x, y))$
6. $A(a)$
7. $\forall x \forall y (\neg P(x) \vee \neg A(x) \vee \neg L(x, y))$



Chapitre 4

Théorie des ensembles

À l'issue de ce chapitre, l'étudiant sera capable de :

- Comprendre le rôle fondamental de la théorie des ensembles comme langage unificateur des mathématiques modernes.
- Distinguer la théorie naïve des ensembles (Cantor) de la théorie axiomatique (ZF), et expliquer comment cette dernière évite les paradoxes (notamment celui de Russell).
- Comprendre la notion d'ontologie en théorie axiomatique : «tout est ensemble».
- Énoncer et expliquer précisément les axiomes ZF :
- Traduire des propriétés ensemblistes en formules logiques et rédiger une preuve formelle à l'aide des axiomes ZF et des règles de PL1.
- Expliquer les concepts métalogiques : Consistance (absence de contradiction), Complétude, Catégoricité et lien avec les modèles

4.1 Introduction

Ce chapitre présente les fondements axiomatiques de la théorie des ensembles et examine leurs principales propriétés métatéoriques : consistance, complétude et catégoricité.

La théorie des ensembles fournit le langage de base des mathématiques modernes [11]. Elle permet de formaliser tous les objets mathématiques (nombres, fonctions, espaces, structures algébriques, etc.) à partir d’une seule notion primitive : *l’ensemble*.

Historiquement développée par George Cantor à la fin du XIX^e siècle, la théorie des ensembles naïve, où tout énoncé du type “les x tels que $P(x)$ ” définit un ensemble, a conduit à des contradictions notoires, comme le paradoxe de Russell. Pour éviter ces paradoxes, la théorie axiomatique des ensembles, notamment celle de **Zermelo–Fraenkel** (ZF), a été formulée.

4.2 Théorie naïve et paradoxe de Russell

4.2.1 Compréhension illimitée

Définition 4.1 (Principe de compréhension illimitée). Dans la théorie naïve des ensembles, on suppose que *pour toute propriété* $P(x)$, il existe un ensemble

$$\{x \mid P(x)\}$$

contenant exactement les objets x qui satisfont $P(x)$.

Exemple 4.1 (Définitions naïves d’ensembles). — L’ensemble des nombres premiers :

$$\{x \mid x \text{ est un nombre premier}\}.$$

— L’ensemble des planètes : $\{x \mid x \text{ est une planète}\}$.

— L’ensemble vide : $\{x \mid \text{faux}\}$.

Dans cette approche, **tout objet mathématique est un ensemble** et il n’existe aucune restriction sur la propriété P utilisée pour définir un ensemble.

Remarque 4.1

Cette vision est simple et unificatrice : elle semble permettre de construire toute collection imaginable comme ensemble [11].

4.2.2 Limite de la définition naïve : le paradoxe de Russell

Si l'on applique le principe de compréhension illimitée à la propriété :

$$P(x) : x \notin x$$

on obtient :

$$R = \{ x \mid x \notin x \}.$$

Selon la théorie naïve, cet ensemble devrait exister, même s'il paraît étrange.

Or :

- Si $R \in R$, alors par définition de R , on doit avoir $R \notin R$.
- Si $R \notin R$, alors par définition de R , on doit avoir $R \in R$.

On obtient ainsi une contradiction : la théorie naïve est **inconsistante** [11].

Remarque 4.2

Pour éviter de tels paradoxes, la théorie moderne des ensembles (Zermelo–Fraenkel) **restreint les règles de formation des ensembles** grâce à des axiomes précis, au lieu d'autoriser toute propriété.

Avant d'examiner les axiomes ZF, il convient de rappeler les principes de déduction formelle qui permettront d'en exploiter la richesse.

4.3 Concept de déduction formelle

4.3.1 Composants d'un système formel

Nous rappelons qu'un système formel comprend :

- un **langage** (syntaxe) ;
- des **axiomes** (formules de base) ;
- des **règles d'inférence**.

4.3.1.1 Langage de la théorie des ensembles

Le langage $\mathcal{L}_\in$ est un langage du premier ordre comprenant :

- un **symbole de relation binaire** : $\in$ (appartenance) ;
- un **symbole d'égalité** : $=$;
- des **variables** : $x, y, z, x_1, x_2, \dots$;
- les **connecteurs logiques** : $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$;
- les **quantificateurs** : $\forall, \exists$.

Définition 4.2 (Formules bien formées). Les formules de $\mathcal{L}_\in$ sont définies inductivement :

- (a) Si x, y sont des variables, alors $x \in y$ et $x = y$ sont des formules atomiques ;
- (b) Si ϕ, ψ sont des formules, alors $\neg\phi, \phi \wedge \psi, \phi \vee \psi, \phi \rightarrow \psi, \phi \leftrightarrow \psi$ sont des formules ;
- (c) Si ϕ est une formule et x une variable, alors $\forall x\phi$ et $\exists x\phi$ sont des formules.

4.3.1.2 Règles d'inférence principales

Les règles d'inférences principale utilisées en déduction naturelle sont énumérées ci-après

- (a) **Modus Ponens** : $\frac{\phi, \phi \rightarrow \psi}{\psi}$;
- (b) **Généralisation universelle** : $\frac{\phi}{\forall x \phi}$ (si x n'est pas libre dans les hypothèses) ;
- (c) **Instantiation universelle** : $\frac{\forall x \phi(x)}{\phi(t)}$ (où t est substituable à x) ;
- (d) **Règle existentielle** : $\frac{\phi(t)}{\exists x \phi(x)}$.

Définition 4.3 (Démonstration). Une démonstration de ϕ à partir d'un ensemble d'axiomes Γ est une suite finie $\phi_1, \dots, \phi_n$ telle que :

- (a) $\phi_n = \phi$;
- (b) chaque ϕ_i est soit un axiome de Γ , soit un axiome logique, soit obtenu par une règle d'inférence à partir de formules précédentes.

On note $\Gamma \vdash \phi$ lorsque ϕ est prouvable à partir de Γ .

Théorème 4.1 (Théorème de déduction). $\Gamma, \phi \vdash \psi$ si et seulement si $\Gamma \vdash \phi \rightarrow \psi$.

4.4 Axiomes de Zermelo-Fraenkel (ZF)

Le système axiomatique ZF a été développé pour éviter les paradoxes de la théorie naïve des ensembles. Ces axiomes définissent quels ensembles existent et comment ils peuvent être construits.

Ainsi, avant d'énoncer les axiomes de Zermelo–Fraenkel (ZF), on fixe l'ontologie :

4.4.1 Ontologie de la théorie ZF

Le mot *ontologie* vient de la philosophie. Dans son sens général, il désigne l'étude de ce qui existe et des catégories fondamentales de l'être. Autrement dit, l'ontologie

cherche à répondre à une question très simple : **quels types de choses existent dans un domaine donné ?**

En mathématiques, on reprend ce mot dans un sens plus restreint et opérationnel : il ne s'agit plus de parler de l'existence au sens large, mais simplement de préciser **quels sont les objets de base** que la théorie manipule.

Dans la théorie des ensembles de Zermelo-Fraenkel (ZF), **tout est ensemble**. Il n'existe pas de distinction primitive entre “ensembles” et “éléments” : il n'y a qu'un seul type d'objet mathématique, à savoir l'ensemble. Tous les autres objets mathématiques (entiers, réels, fonctions, relations, structures) sont ensuite reconstruits à partir de cette brique fondamentale.

Conséquences pratiques Lorsque nous écrivons $x \in A$, nous avons :

- x est un ensemble,
- A est un ensemble,
- la relation exprime que l'ensemble x est membre de l'ensemble A .

Exemple 4.2. Les nombres naturels. Dans la construction de von Neumann :

$$0 = \emptyset, \quad 1 = \{\emptyset\} = \{0\}, \quad 2 = \{\emptyset, \{\emptyset\}\} = \{0, 1\}, \quad 3 = \{0, 1, 2\}, \quad \dots$$

Les couples ordonnés. Un couple ordonné est défini par :

$$(a, b) = \{\{a\}, \{a, b\}\}.$$

Les fonctions. Une fonction est un ensemble de couples ordonnés.

Cette approche peut sembler contre-intuitive :

- Dans $3 \in \mathbb{N}$, le nombre 3 est un ensemble, et $\mathbb{N}$ est un ensemble contenant 3.
- Dans $0 \in 3$, le nombre 0 est un ensemble, et 3 est un ensemble contenant 0.

Remarque 4.3

Cette unification est ce qui rend ZF suffisamment simple pour servir de fondement à l'ensemble des mathématiques modernes.

4.4.2 Les Axiomes Fondamentaux

Axiome d'Extensionnalité (Ext) :

$$\forall A \forall B (\forall x (x \in A \leftrightarrow x \in B) \rightarrow A = B)$$

Deux ensembles sont égaux s'ils contiennent exactement les mêmes éléments.

Exemple : $\{1, 2, 3\} = \{3, 2, 1\}$.

Axiome de l'Ensemble Vide (Vide) :

$$\exists A \forall x (x \notin A)$$

Il existe au moins un ensemble sans éléments, noté $\emptyset$. *Exemple* : $\emptyset = \{\}$.

Axiome de la Paire (Paire) :

$$\forall a \forall b \exists A \forall x (x \in A \leftrightarrow (x = a \vee x = b))$$

Pour deux objets a et b , on peut former l'ensemble $\{a, b\}$. *Exemple* : $\{2, 5\}$ est un ensemble contenant exactement 2 et 5.

Axiome de l'Union (Union) :

$$\forall \mathcal{F} \exists A \forall x (x \in A \leftrightarrow \exists B (B \in \mathcal{F} \wedge x \in B))$$

L'union d'une famille d'ensembles existe. *Exemple* : Si $\mathcal{F} = \{\{1, 2\}, \{2, 3\}\}$, alors $\bigcup \mathcal{F} = \{1, 2, 3\}$.

Axiome de l'Ensemble des Parties (Parties) :

$$\forall A \exists P \forall x (x \in P \leftrightarrow x \subseteq A)$$

ou en utilisant uniquement la relation d'appartenance :

$$\forall A \exists P \forall x \left(x \in P \leftrightarrow \forall u (u \in x \Rightarrow u \in A) \right).$$

Pour tout ensemble A , il existe un ensemble $\mathcal{P}(A)$ qui contient tous ses sous-ensembles. *Exemple* : $\mathcal{P}(\{1, 2\}) = \{\emptyset, \{1\}, \{2\}, \{1, 2\}\}$.

4.4.3 Axiomes de Construction

4.4.3.1 Schéma de séparation (compréhension restreinte)

Idée de base. Dans la théorie naïve, on admettait : «pour toute propriété $P(x)$, l'ensemble $\{x \mid P(x)\}$ existe». Cette liberté mène au paradoxe de Russell. En ZF, on *restreint* la construction : on ne filtre que dans un ensemble déjà existant.

Formulation. Pour toute propriété logique $\varphi(x)$, et pour tout ensemble A , il existe un ensemble composé exactement des éléments de A qui vérifient φ .

Énoncé formel. Pour toute formule $\varphi(x, p_1, \dots, p_n)$ à paramètres arbitraires :

$$\forall A \exists B \forall x \left(x \in B \leftrightarrow (x \in A \wedge \varphi(x, p_1, \dots, p_n)) \right).$$

Exemple. Soit $A = \{1, 2, 3, 4, 5\}$ et $\varphi(x) = \text{«}x \text{ est pair}\text{»}$. Alors la séparation garantit que

$$\{x \in A \mid x \text{ est pair}\} = \{2, 4\}$$

est bien un ensemble.

Remarque 4.4

On ne parle jamais de $\{x \mid x \notin x\}$ en ZF, car il n'existe pas d'«ensemble universel» U contenant tout. La séparation n'autorise qu'un **filtrage interne** à partir d'un A donné — c'est précisément ce qui évite le paradoxe de Russell.

4.4.3.2 Schéma de remplacement

Idée générale. Alors que le schéma de séparation permet d'extraire un sous-ensemble d'un ensemble existant, le *schéma de remplacement* autorise la construction d'un ensemble obtenu en *transformant* les éléments d'un ensemble donné selon une règle bien définie. Plus précisément, si pour chaque $x \in A$ il existe **un unique** y tel que $\varphi(x, y, \vec{p})$, on peut considérer la collection de toutes les « images » de x par cette relation fonctionnelle. Le schéma de remplacement garantit que cette collection forme bien un ensemble.

Formulation. Soit $\varphi(x, y, \vec{p})$ une formule du langage, où $\vec{p}$ désigne un n -uplet de paramètres $(p_1, \dots, p_n)$. Si φ définit une fonction sur A , c'est-à-dire si

$$\forall x \in A \exists! y \varphi(x, y, \vec{p}),$$

alors il existe un ensemble B constitué exactement des valeurs ainsi obtenues :

$$\forall A \left[\left(\forall x \in A \exists! y \varphi(x, y, \vec{p}) \right) \Rightarrow \exists B \forall y \left(y \in B \leftrightarrow \exists x \in A \varphi(x, y, \vec{p}) \right) \right].$$

On note classiquement cet ensemble sous la forme :

$$B = \{y \mid \exists x \in A, \varphi(x, y, \vec{p})\}.$$

Les *paramètres* sont les variables libres autres que celles quantifiées explicitement dans une formule. Leur valeur est supposée fixée dans le contexte. Si une formule $\varphi(x, y, p_1, \dots, p_n)$ contient de tels paramètres, ceux-ci peuvent être utilisés dans le schéma de séparation ou de remplacement.

Exemples.

- **Ensemble fini.** Si $A = \{1, 2, 3\}$ et que la règle donnée est $y = 2x$, alors le schéma de remplacement garantit l'existence de l'ensemble $\{2, 4, 6\}$.
- **Ensemble infini.** À partir de $A = \mathbb{N}$ et de la règle $y = x^2$, le remplacement permet de former l'ensemble des carrés :

$$\{n^2 \mid n \in \mathbb{N}\} = \{0, 1, 4, 9, 16, \dots\}.$$

Remarque 4.5

Le remplacement est crucial pour développer des constructions «grandes» (ordinaux, cardinaux, ...). Sans lui, ZF ne suffirait pas pour formaliser confortablement l'analyse et de nombreuses parties modernes des maths.

4.4.4 Axiome de Régularité (ou de Fondation)

Définition 4.4 (Axiome de Fondation). L'axiome de régularité affirme que tout ensemble non vide contient un élément qui ne partage aucun élément avec lui. Formellement :

$$\forall A (A \neq \emptyset \rightarrow \exists x (x \in A \wedge x \cap A = \emptyset)).$$

Intuition. Cet axiome impose une structure “bien fondée” à la relation d'appartenance. Il interdit :

- les boucles d'appartenance telles que $x \in x$;
- les cycles d'ensemble : $A \in B$ et $B \in A$;
- les chaînes descendantes infinies :

$$\dots \in x_3 \in x_2 \in x_1 \in x_0.$$

Ainsi, tout ensemble “repose” sur des éléments plus simples et possède un élément minimal pour la relation $\in$.

Exemples d'ensembles impossibles.

$$A = \{A\}, \quad A = \{B\}, \quad B = \{A\}, \quad x_0 \ni x_1 \ni x_2 \ni \dots$$

Tous ces objets sont exclus en ZF grâce à la fondation.

4.5 Propriétés métatéoriques

4.5.1 Consistance

Définition 4.5. Une théorie T est *consistante* si elle ne prouve pas à la fois ϕ et $\neg\phi$ pour une même formule ϕ .

4.5.2 Complétude

Définition 4.6 (Complétude sémantique). Une théorie T est (sémantiquement) complète si pour toute formule ϕ , soit $T \models \phi$ soit $T \models \neg\phi$.

Définition 4.7 (Complétude syntaxique). Une théorie T est (syntaxiquement) complète si pour toute formule ϕ , soit $T \vdash \phi$ soit $T \vdash \neg\phi$.

Théorème 4.2 (Théorème de complétude de Gödel). *Pour la logique du premier ordre : $T \vdash \phi$ si et seulement si $T \models \phi$.*

4.5.2.1 Incomplétude de ZF

Théorème 4.3 (Incomplétude de ZF). *ZF n'est pas complète. Il existe des énoncés indécidables dans ZF, notamment :*

- (a) *L'hypothèse du continu CH*
- (b) *L'axiome du choix AC*
- (c) *L'hypothèse généralisée du continu GCH*

4.5.3 Catégoricité

Définition 4.8 (Catégoricité). Une théorie T est catégorique dans la cardinalité κ si tous les modèles de T de cardinalité κ sont isomorphes.

Théorème 4.4 (Non-catégoricité de ZF). *ZF n'est catégorique dans aucune cardinalité infinie. En particulier :*

- (a) *Il existe des modèles dénombrables non isomorphes de ZF*
- (b) *Pour toute cardinalité infinie κ , il existe des modèles de ZF de cardinalité κ non isomorphes*

Esquisse. Utilise le théorème de Löwenheim-Skolem et des techniques de forcing pour construire des modèles avec des propriétés différentes. $\square$

4.5.3.1 Théorème de Löwenheim-Skolem

Théorème 4.5 (Löwenheim-Skolem descendant). *Si une théorie du premier ordre a un modèle infini, alors elle a un modèle dénombrable.*

Théorème 4.6 (Löwenheim-Skolem ascendant). *Si une théorie du premier ordre a un modèle infini, alors pour toute cardinalité $\kappa \geq |\mathcal{L}|$, elle a un modèle de cardinalité κ .*

4.6 Axiome du Continu

Le cardinal d'un ensemble est sa « taille ». Pour les ensembles finis, il s'agit du nombre d'éléments. Pour les ensembles infinis, deux ensembles ont le même cardinal s'il existe une bijection entre eux.

Le plus petit cardinal infini est

$$\aleph_0 = |\mathbb{N}|,$$

et un ensemble ayant ce cardinal est dit *dénombrable*. Par exemple, $\mathbb{Z}$ et $\mathbb{Q}$ sont tous deux de cardinal $\aleph_0$.

Cantor a montré que la droite réelle a un cardinal strictement supérieur :

$$|\mathbb{R}| = 2^{\aleph_0},$$

appelé *le continu*. La question naturelle est de savoir s'il existe un cardinal intermédiaire entre $\aleph_0$ et $2^{\aleph_0}$; c'est l'objet de l'hypothèse du continu.

4.6.1 L'hypothèse du continu

Définition 4.9 (Hypothèse du continu (CH)). Il n'existe pas d'ensemble dont la cardinalité est strictement comprise entre celle de $\mathbb{N}$ et celle de $\mathbb{R}$:

$$\text{CH} : \neg \exists X (\aleph_0 < |X| < 2^{\aleph_0})$$

Équivalamment : $2^{\aleph_0} = \aleph_1$.

4.6.2 Indépendance de l'hypothèse du continu

Idée. Une question naturelle est de savoir si l'hypothèse du continu (CH) peut être déduite des axiomes usuels de la théorie des ensembles (ZFC), ou si au contraire elle peut être réfutée à partir de ces axiomes.

Les travaux de Gödel puis de Cohen ont montré que les axiomes de ZFC ne permettent ni de prouver CH, ni de prouver sa négation. On peut donc construire un univers mathématique où CH est vraie, et un autre univers où CH est fausse, sans contredire ZFC.

On dit alors que CH est *indépendante* de ZFC.

Théorème 4.7 (Gödel, 1938). *Si ZF est consistante, alors $ZF + CH$ est consistante.*

Idée. Gödel construit un modèle particulier de la théorie des ensembles, appelé l'univers *constructible* et noté L , dans lequel il montre que CH est vraie. Ainsi, l'ajout de CH aux axiomes ne produit pas de contradiction.

Théorème 4.8 (Cohen, 1963). *Si ZF est consistante, alors $ZF + \neg CH$ est consistante.*

Idée. Cohen développe une nouvelle méthode, le *forcing*, qui permet d'« ajouter » des sous-ensembles de $\mathbb{N}$ tout en respectant les axiomes de ZF. Dans le modèle obtenu, la cardinalité du continu devient strictement supérieure à $\aleph_1$, et donc CH est fausse.

Corollaire 4.1. *L'hypothèse du continu est indépendante de ZFC.*

4.7 Exemples de modèles de la théorie ZF

Un *modèle* de la théorie ZF est une structure $(M, \in)$ telle que tous les axiomes de ZF y soient satisfaits.

Exemple 1 : Le modèle standard $\mathcal{V}$. L'univers usuel des mathématiques, constitué de l'ensemble de tous les ensembles, est un modèle informel de ZF construit par la hiérarchie :

$$V_0 = \emptyset, \quad V_{\alpha+1} = \mathcal{P}(V_\alpha), \quad V = \bigcup_{\alpha \in \text{Ord}} V_\alpha.$$

Exemple 2 : Modèle fini (non standard). Soit $M = \{\emptyset, \{\emptyset\}\}$ muni de la relation d'appartenance usuelle. Certaines propriétés de ZF y sont satisfaites (extensionnalité, vide, paire), mais l'axiome de l'infini y est faux. Il s'agit donc d'un *modèle partiel* des axiomes de ZF.

4.8 Récapitulation (Rôles des Axiomes)

La théorie des ensembles Zermelo–Fraenkel (ZF) est fondée sur un ensemble fini d’axiomes formalisant les propriétés fondamentales de l’appartenance $\in$. Chaque axiome joue un rôle précis dans la construction des objets mathématiques.

Axiome	Rôle fondamental
Extensionnalité	Deux ensembles sont égaux s’ils ont les mêmes éléments.
Ensemble vide	Garantit l’existence de $\emptyset$.
Paire	Permet de construire $\{a, b\}$.
Union	Permet de construire $\bigcup A$.
Puissance	Permet de construire $\mathcal{P}(A)$.
Infini	Assure l’existence d’un ensemble infini.
Séparation	Permet de définir un sous-ensemble par propriété.
Remplacement	Garantit l’image d’un ensemble par une fonction définissable.
Fondation	Exclut les chaînes d’appartenance infinies descendantes.

Ainsi, tous les objets mathématiques (entiers, relations, fonctions, espaces, etc.) sont construits à partir de $\emptyset$ par itérations successives de ces axiomes.

4.9 Exercices

Exercice 4.1: Unicité de l’ensemble vide

Dans la théorie ZF et la logique du premier ordre, prouvez formellement que

$$\exists! E \forall x (x \notin E).$$

Exercice 4.2

Montrer que pour tout ensemble A , on a $A \subseteq \bigcup \mathcal{P}(A)$.

Exercice 4.3

À partir de l’axiome de la paire, montrer l’existence de l’ensemble $\{a\}$.

Exercice 4.4

Construire explicitement les entiers naturels 0, 1, 2 dans le cadre de ZF.

Solutions

Exercice 1

- Preuve.** 1. $\exists A \forall x (x \notin A)$ (ZF — Axiome du Vide)
2. Supposons A, B tels que $\forall x (x \notin A)$ et $\forall x (x \notin B)$ (Hypothèse pour l'unicité)
3. Soit x arbitraire. De $\forall x (x \notin A)$, on obtient $x \notin A$ ($\forall$ -élimination sur A)
4. De $\forall x (x \notin B)$, on obtient $x \notin B$ ($\forall$ -élimination sur B)
5. De $x \notin A$, on déduit $(x \in A \rightarrow x \in B)$ (Tautologie $\neg P \rightarrow (P \rightarrow Q)$ + MP)
6. De $x \notin B$, on déduit $(x \in B \rightarrow x \in A)$ (Tautologie analogue + MP)
7. Des étapes (5) et (6), on conclut $(x \in A \leftrightarrow x \in B)$ ($\leftrightarrow$ -intro à partir de $\rightarrow$)
8. $\forall x (x \in A \leftrightarrow x \in B)$ ($\forall$ -intro sur x)
9. $\forall A \forall B (\forall x (x \in A \leftrightarrow x \in B) \rightarrow A = B)$ (ZF — Axiome d'Extensionnalité)
10. $\forall B (\forall x (x \in A \leftrightarrow x \in B) \rightarrow A = B)$ ($\forall$ -élimination sur A)
11. $(\forall x (x \in A \leftrightarrow x \in B) \rightarrow A = B)$ ($\forall$ -élimination sur B)
12. $A = B$ (MP avec (8) et (11))
13. Conclusion : il existe un A vide (1) et tout B vide coïncide avec A (12), donc $\exists! E \forall x (x \notin E)$ $\square$

Exercice 2

Montrer que pour tout ensemble A , on a $A \subseteq \bigcup \mathcal{P}(A)$.

- Preuve.** 1. $A \subseteq B \iff \forall x (x \in A \Rightarrow x \in B)$ (Définition de l'inclusion)
2. $\mathcal{P}(A) = \{ X \mid X \subseteq A \}$ (Définition de l'ensemble des parties)
3. $\bigcup \mathcal{P}(A) = \{ x \mid \exists X (X \in \mathcal{P}(A) \wedge x \in X) \}$ (Définition de l'union)
4. Soit x tel que $x \in A$ (Hypothèse arbitraire pour $\forall$ -intro)

5. De $A \subseteq A$, on déduit $A \in \mathcal{P}(A)$ (Définition de $\mathcal{P}(A)$)
6. Comme $x \in A$, on a $\exists X (X \in \mathcal{P}(A) \wedge x \in X)$ (en prenant $X = A$) (Introduction de $\exists$)
7. De (3) et (6), on obtient $x \in \bigcup \mathcal{P}(A)$ (Définition de l'union)
8. Donc $\forall x (x \in A \Rightarrow x \in \bigcup \mathcal{P}(A))$ ($\forall$ -intro)
9. Par (1) et (8), on conclut $A \subseteq \bigcup \mathcal{P}(A)$ (Définition de l'inclusion) $\square$

Exercice 3

Montrer que pour tout ensemble A , on a $\emptyset \subseteq A$.

- Preuve.** 1. $\forall x \neg(x \in \emptyset)$ (ZF — Axiome du Vide)
2. Soit A un ensemble arbitraire. (Hypothèse arbitraire sur A)
3. Pour montrer $\emptyset \subseteq A$, il suffit de prouver

$$\forall x (x \in \emptyset \Rightarrow x \in A)$$

- (Définition de l'inclusion)
4. Soit x un objet arbitraire. (Hypothèse pour $\forall$ -intro)
5. De (1), on obtient $\neg(x \in \emptyset)$. ($\forall$ -élimination sur x)
6. De $\neg(x \in \emptyset)$, on déduit
- $$x \in \emptyset \Rightarrow x \in A$$
- (Tautologie $\neg P \Rightarrow (P \Rightarrow Q)$)
7. Donc $\forall x (x \in \emptyset \Rightarrow x \in A)$. ($\forall$ -introduction sur x)
8. Par (3) et (7), on a $\emptyset \subseteq A$. (Définition de l'inclusion)
9. Comme A était arbitraire, on conclut

$$\forall A (\emptyset \subseteq A).$$

($\forall$ -introduction sur A) $\square$

Exercice 4

Construire explicitement les entiers naturels 0, 1, 2 dans le cadre de ZF.

Dans l'approche usuelle (Von Neumann), les entiers sont définis par :

$$0 = \emptyset, \quad 1 = \{0\} = \{\emptyset\}, \quad 2 = \{0, 1\} = \{\emptyset, \{\emptyset\}\}.$$

- L'axiome de l'ensemble vide assure l'existence de $0 = \emptyset$.
- L'axiome de la paire appliqué à $(0, 0)$ donne l'existence de $\{0\}$, que l'on note 1.
- L'axiome de la paire appliqué à $(0, 1)$ donne l'existence de $\{0, 1\}$, que l'on note 2.

On peut ensuite poursuivre la construction des entiers

$$3 = \{0, 1, 2\}, \quad 4 = \{0, 1, 2, 3\}, \quad \dots$$

en utilisant de nouveau la paire et l'union.



Bibliographie

- [1] E. Mendelson. *Introduction to Mathematical Logic*. 6th ed., Taylor & Francis, 2015.
- [2] S. C. Kleene. *Introduction to Metamathematics*. Ishi Press International, 2009.
- [3] P. S. Novikov. *Éléments de logique mathématique*. Dunod, Paris, 1964.
- [4] G. Kreisel, J.-L. Krivine. *Éléments de logique mathématique : Théorie des modèles*. Dunod, Paris, 1967.
- [5] A. A. Stolyar. *Introduction to Elementary Mathematical Logic*. Dover Publications, New York, 1984.
- [6] R. Fraïssé. *Cours de logique mathématique*. Gauthier-Villars, Paris, 1967.
- [7] A. Fuchs, G. Reeb. *Logique*. Office des Publications Universitaires (O.P.U.), Alger, 1986.
- [8] Université de Nantes, Département IFAC. *Quelques exercices corrigés de déduction naturelle*. [En ligne]. Disponible : http://www.ifac.univ-nantes.fr/IMG/pdf/exercices_corriges_de_deduction_naturelle-1.pdf. Consulté le 14 août 2025.
- [9] J. A. Robinson. A Machine-Oriented Logic Based on the Resolution Principle. *Journal of the ACM*, 12(1) :23–41, 1965. doi :10.1145/321250.321253.
- [10] R. M. Smullyan. *First-Order Logic*. Dover Publications, 1995.
- [11] C. C. Pinter. *A Book of Set Theory*. Dover Publications, New York, 2014.