



HAL
open science

Logique formelle et modélisation du raisonnement — Notions de base

Denis Roegel

► **To cite this version:**

Denis Roegel. Logique formelle et modélisation du raisonnement — Notions de base. École d'ingénieur.
France. 1995. hal-02340515

HAL Id: hal-02340515

<https://inria.hal.science/hal-02340515>

Submitted on 30 Oct 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Logique formelle et modélisation du raisonnement
Notions de base

Support de cours — ESSTIN

Denis Roegel

Novembre 1995

Mises à jour: 18 février 1996, 25 septembre 1997 et 2 octobre 1997

Table des matières

Introduction : Formalisation du raisonnement — les logiques	4
1 Systèmes formels : exemples introductifs	6
1.1 Exemple 1 : génération de théorèmes de l'arithmétique	6
1.2 Exemple 2 : calcul d'intégrales	7
1.3 Exemple 3 : arithmétique de Peano	8
2 Calcul des propositions	10
2.1 Introduction	10
2.1.1 Les formules atomiques	10
2.1.2 Les formules composées	11
2.1.3 Le substrat de la logique	11
2.1.4 But du calcul des propositions	12
2.2 Syntaxe	12
2.3 Vérité d'une formule	13
2.3.1 Introduction	13
2.3.2 Algèbres de Boole	13
2.3.3 Interprétation des formules dans une algèbre de Boole	14
2.3.4 Tautologies et équivalences	14
2.3.5 Tables de vérité	15
2.4 Équivalences classiques	16
2.5 Systèmes axiomatiques	17
2.5.1 Emploi d'un système axiomatique pour dériver des formules valides	17
2.5.2 Preuve à partir d'hypothèses	19
2.6 Calcul des séquents	20
2.6.1 Présentation	20
2.6.2 Réversibilité des règles	24
2.6.3 Récapitulation des règles	24
2.6.4 Validité d'un séquent	25
2.6.5 Théorème de correction (ou consistance)	25
2.6.6 Théorème de complétude	25
2.6.7 Exemples de preuves	26
2.6.8 Exercices	27

3	Calcul des prédicats : exemples introductifs	28
3.1	Le syllogisme ou les catégories d'Aristote	28
3.2	Diagrammes de Venn	29
3.3	Limites du calcul des propositions	31
3.4	Limites des diagrammes de Venn	31
4	Calcul des prédicats (logique du premier ordre)	33
4.1	Introduction	33
4.2	Syntaxe	34
4.2.1	Lexique	34
4.2.2	Grammaire	34
4.3	Interprétation des formules	35
4.3.1	Limites de l'interprétation, nécessité d'un système de preuve	37
4.4	Équivalences classiques	37
4.5	Systèmes axiomatiques	38
4.6	Calcul des séquents	39
4.6.1	Présentation	39
4.6.2	Récapitulation des règles	41
4.6.3	Théorèmes de correction et de complétude	41
4.6.4	Exemple de preuve	41
	Glossaire et définitions	42
	Bibliographie	44

Avertissement

Ce document en est à sa première mouture. Malgré des soins particuliers consacrés à sa conception, l'auteur ne doute pas de la présence d'erreurs. Il prie le lecteur d'être indulgent, et le remercie de lui signaler les erreurs potentielles et des améliorations éventuelles. L'adresse électronique de l'auteur est `roegel@loria.fr`.

Remerciements

L'auteur n'ayant pas inventé la logique, il s'est contenté d'exposer ce qu'il pense devoir être les notions fondamentales de la logique, en s'inspirant des travaux et œuvres de ses aînés. Sans qu'il en soit toujours fait mention explicitement, l'auteur a été guidé par les ouvrages cités en bibliographie, en particulier ceux de Lalement, Gochet et Gribomont, et Quine. L'auteur s'est aussi inspiré d'un cours de Denis Lugiez, chercheur au CRIN (Centre de Recherche en Informatique de Nancy). Enfin, l'auteur remercie Daniel Briaud pour de nombreuses et fructueuses discussions autour de ce cours.

Introduction : Formalisation du raisonnement — les logiques

« Logique » est un mot provenant du grec *logos* qui signifie « science de la raison ». La logique étudie le discours, et plus particulièrement le(s) raisonnement(s).

De tous temps, les hommes se sont disputés et la force a souvent triomphé sur la raison. Les discours eussent souvent pu éviter des drames, pourvu qu'ils aient été bien compris. Malheureusement, on s'est aperçu il y a bien longtemps de la difficulté qu'il y avait à exprimer des choses sûres et vraies dans notre langue. Celle-ci est source de malentendus.

Pour convaincre ou pour se laisser convaincre, il paraît indispensable de supprimer les *ambiguïtés de la langue parlée ou écrite*. Nous souhaitons pouvoir comprendre sans équivoque une phrase. En somme, nous aimerions que la *sémantique* d'une phrase soit clairement déterminée.

Plus encore, même si chaque phrase a un sens bien précis, il paraît nécessaire de codifier les enchaînements de phrases, tout comme dans un jeu où certains coups ne sont pas autorisés.

Cette *codification*, ces règles, ces restrictions, créent une nouvelle langue, beaucoup plus rigide que la « vraie » langue, mais cette rigidité est ce qui nous intéresse, car c'est elle qui va nous permettre de découvrir des propriétés nouvelles.

Une logique, c'est une telle codification. C'est une *description* d'un certain type de réalité et cette description a pour but de nous aider à trouver la vérité. Nous voulons par exemple savoir si une affirmation est vraie ou fausse ou si quelqu'un a raison ou tort, relativement à la logique considérée.

La logique n'est pas unique parce que la langue que l'on cherche à codifier ne l'est pas non plus. Certes, on peut supposer qu'on ne s'intéressera qu'à des discours en français, mais ces discours n'en peuvent pas moins avoir une *structure* très différente. Pour caricaturer, on pourrait imaginer les deux discours extrêmes suivants, l'un où seules des phrases affirmatives sont autorisées, telles que « Il fait beau » ou « il pleut », l'autre où des hypothèses peuvent être émises, telles que « s'il pleut, le sol est mouillé ».

On peut voir une logique comme une *restriction* d'un langage mais aussi comme une *formalisation* de ce même langage. Dans ce cas, on imagine bien qu'à chaque type de raisonnement *correspond* une formalisation, d'où l'existence de diverses logiques.

Il y a le raisonnement et il y a ce sur quoi on raisonne. La logique ne s'intéresse pas à l'objet du raisonnement. Le logicien se place dans un monde « logique » où les faits¹ sont des entités abstraites *correspondant* aux entités concrètes du monde. Le processus est le

1. *Le monde se décompose en faits* (Ludwig Wittgenstein, Tractatus logico-philosophicus, 1922, § 1.2.1).

suivant : nous partons de faits concrets et nous leur faisons correspondre des faits abstraits. La logique manipule ces faits abstraits et en déduit d'autres. La manipulation des faits abstraits est censée *correspondre* à celle des faits concrets. Nous disons « censée » parce que la correspondance reflète une certaine image du monde, laquelle n'est pas nécessairement figée et peut dépendre des connaissances physiques du moment. Aux nouveaux faits abstraits, nous faisons correspondre *réciroquement* de nouveaux faits concrets.

L'un des intérêts de la formalisation du raisonnement découle du fait que la structure du raisonnement varie peu. Nous raisonnons souvent de la même manière. Si nous prouvons qu'un certain raisonnement est correct ou faux, nous pourrions *réutiliser* cette connaissance avec d'autres discours. L'analyse du raisonnement nous permet aussi de réfuter des discours, de pointer sur des contradictions. Enfin, la mécanisation du raisonnement a bien sûr des applications de choix en informatique et dans toutes les disciplines où la machine doit aider, voire remplacer l'homme.

Chapitre 1

Systèmes formels : exemples introductifs

Nous donnons dans cette partie quelques exemples pouvant servir d'introduction aux systèmes formels. Tous ces exemples font appel à certaines notions qui ne seront vues que dans les chapitres ultérieurs. Nous suggérons au lecteur de se faire une première idée des théories exposées ici, sans pour autant vouloir tout comprendre, puis d'y revenir après avoir lu le chapitre consacré au calcul des prédicats.

1.1 Exemple 1 : génération de théorèmes de l'arithmétique

Voici un premier exemple repris à Quine¹.

Admettons comme axiomes les deux équations :

$$\begin{aligned} (A) \quad & x = x - (y - y) \\ (B) \quad & x - (y - z) = z - (y - x) \end{aligned}$$

et comme règles d'inférence :

- substitution d'une expression quelconque pour toutes les occurrences d'une variable (*substitution*);
- si $\alpha = \beta$, remplacement de β par α n'importe où (*remplacement des égaux par les égaux*).

Voici quelques théorèmes que nous pouvons déduire :

1. W.V. Quine: *A Method for Generating Part of Arithmetic Without Intuitive Logic*, 1934.

$$\begin{array}{lll}
(1) & x = x & (A),(A) \\
(2) & x = x - (z - z) & (A) \\
(3) & y - (x - z) = z - (x - y) & (B) \\
(4) & y - (x - (z - z)) = z - z - (x - y) & (B) \\
(5) & z - (z - z - (x - y)) = x - y - (z - z - z) & (B) \\
(6) & y - y - (x - x - x) = y - y - (x - x - x) & (1) \\
(7) & y - x = z - z - (x - y) & (2),(4) \\
(8) & x - z - y = w - w - (y - (x - z)) & (7) \\
(9) & x - y - z = w - w - (z - (x - y)) & (7) \\
(10) & z - (y - x) = x - y - (z - z - z) & (7),(5) \\
(11) & x - (y - y) = y - y - (x - x - x) & (10) \\
(12) & x - y - z = w - w - (y - (x - z)) & (3),(9) \\
(13) & x - (y - z) = x - y - (z - z - z) & (B),(10) \\
(14) & x - (y - y) = x - y - (y - y - y) & (13) \\
(15) & x - y - z = x - z - y & (8),(12) \\
(16) & z - z - (x - y) = z - (x - y) - z & (15) \\
(17) & y - x = z - (x - y) - z & (7),(16) \\
(18) & y - x = y - (x - z) - z & (3),(17) \\
(19) & x = x - y - (y - y - y) & (A),(14) \\
(20) & x = y - y - (x - x - x) & (A),(11) \\
(21) & y - y - (x - x - x) = x & (20),(6)
\end{array}$$

Les numéros à gauche des équations sont les numéros des théorèmes. Les numéros à droite indiquent comment un théorème a été obtenu. Lorsqu'il ne figure qu'un nombre ou une lettre à droite, cela signifie que le théorème a été obtenu à partir du théorème indiqué, en y faisant une substitution (première règle d'inférence). Par exemple (8) est obtenu de (7) en y remplaçant z par w , x par y et y par $x - z$ (en parallèle).

Lorsque deux nombres figurent à droite, cela signifie que la partie droite de l'équation représentée par le premier nombre est remplacée par sa partie gauche dans l'équation représentée par le second nombre. Par exemple, (17) est obtenu de (16) en remplaçant $z - z - (x - y)$ par $y - x$, conformément à l'équation (7).

Quine a montré que tous les théorèmes constitués uniquement de soustractions et de parenthèses peuvent être obtenus à partir des deux axiomes et des règles d'inférence.

1.2 Exemple 2 : calcul d'intégrales

On considère l'ensemble des fonctions polynomiales dont les monômes ont un degré positif ou nul. Ces fonctions sont intégrables sur \mathbb{R} .

On note $I(f,a,b)$ la valeur de l'intégrale $\int_a^b f(x) dx$.

On se donne les axiomes suivants, où $A \supset B$ doit être lu « si A alors B » :

- (1) $(f = f_1 + f_2) \supset (I(f,a,b) = I(f_1,a,b) + I(f_2,a,b))$
- (2) $(f = -f_1) \supset (I(f,a,b) = -I(f_1,a,b))$
- (3) $(f = c \cdot f_1) \supset (I(f,a,b) = c \cdot I(f_1,a,b))$
- (4) $(f = x^n) \supset (I(f,a,b) = \frac{b^{n+1}}{n+1} - \frac{a^{n+1}}{n+1})$
- (5) $(a < b < c) \supset (I(f,a,c) = I(f,a,b) + I(f,b,c))$
- (6) $I(f,a,b) = -I(f,b,a)$

ainsi que les règles de *modus ponens* (cf. §2.5.1), de remplacement d'égaux par des égaux (si $f = f_1 + f_2$, on peut remplacer f par $f_1 + f_2$ où on le souhaite) et de substitution dans les axiomes ou les théorèmes. Ces deux dernières règles sont analogues à celles du premier exemple.

À partir de ces axiomes et de ces règles d'inférence, on peut alors déduire tous les théorèmes concernant les valeurs des intégrales de fonctions polynomiales.

Par exemple, si $f(x) = x^3 - 7x^2$, on obtient

$$\begin{aligned}
 I(f,0,1) &= I(x^3,0,1) + I(-7x^2,0,1) \text{ par (1)} \\
 &= 1/4 + I(-7x^2,0,1) \text{ par (4)} \\
 &= 1/4 - 7I(x^2,0,1) \text{ par (3)} \\
 &= 1/4 - 7 \cdot 1/3 \text{ par (4)}
 \end{aligned}$$

1.3 Exemple 3 : arithmétique de Peano

L'*arithmétique de Peano* (du nom du mathématicien italien qui l'a introduite) est une des grandes théories étudiées par les logiciens, et utilisée tant par les mathématiciens que par les informaticiens.

Il s'agit d'un système pour l'arithmétique usuelle sur les entiers naturels. Les axiomes de ce système comportent les symboles $<$ et $=$ pour représenter les relations « plus petit » et « égal ». Ils comprennent d'autre part les symboles 0 pour le nombre zéro et les symboles s , $+$ et \cdot pour les opérations « successeur », « addition » et « multiplication ». Dans ce système, 1 est noté $s(0)$ (« successeur de zéro »), 2 est noté $s(s(0))$ et ainsi de suite.

Ce système admet les axiomes suivants :

- (1) $(\forall x) \neg(s(x) = 0)$, i.e., le successeur d'un entier naturel n'est jamais égal à zéro;
- (2) $(\forall x,y) (s(x) = s(y)) \supset (x = y)$, i.e. si deux successeurs sont égaux, ils sont les successeurs d'un même nombre;
- (3) $(\forall x) (x + 0 = x)$, i.e. 0 est élément neutre de l'addition;
- (4) $(\forall x,y) (x + s(y) = s(x + y))$
- (5) $(\forall x) (x \cdot 0 = 0)$,
- (6) $(\forall x,y) (x \cdot s(y) = x \cdot y + x)$, i.e., distributivité de la multiplication par rapport au successeur;
- (7) $(\forall x) \neg(x < 0)$, i.e. x est positif ou nul;
- (8) $(\forall x,y) ((x < s(y)) \equiv (x < y) \vee (x = y))$, i.e. définition de $x < y$;

(9) $\varphi(0) \wedge (\forall x) (\varphi(x) \supset \varphi(s(x))) \supset \forall x \varphi(x)$, i.e., principe de récurrence.

Il s'agit d'un exemple de système incomplet, c'est-à-dire que certaines affirmations vraies de ce système ne sont pas démontrables. (cf. Gödel)

Chapitre 2

Calcul des propositions

2.1 Introduction

On s'intéresse à des affirmations concernant des choses particulières, ces affirmations pouvant être vraies ou fausses. Ces affirmations sont des formules et celles que nous considérons sont de deux sortes :

1. Les formules dites atomiques, ou encore *propositions*, sont des expressions que l'on considère indécomposables; certaines de ces propositions sont vraies, certaines sont fausses; quelquefois la vérité (ou la fausseté) d'une proposition nous est donnée, quelquefois nous devons la déterminer.
2. Les formules composées, ou non atomiques, sont des formules obtenues à partir d'autres formules plus petites en appliquant des opérations « logiques ». Dans la pratique, ces opérations sont la négation (« non »), la conjonction (« et »), la disjonction (« ou ») et le conditionnel¹ (« si ... alors »). On verra par la suite que l'ensemble des opérations peut être différent, car certaines opérations sont équivalentes à l'emploi combiné de plusieurs autres opérations.

2.1.1 Les formules atomiques

On peut décider que les phrases qui suivent sont des propositions (formules atomiques) :

- « le tableau est propre »;
- « j'ai écrit sur le tableau »;
- « personne ne vient d'effacer le tableau »;

Qu'est-ce qui fait qu'une formule est atomique et qu'une autre ne l'est pas ?

La *seule chose* qui caractérise une formule atomique, c'est le fait que nous nous interdisons de la décomposer ou d'émettre — au sein du langage formel — des considérations relatives

1. Nous distinguons dans ce document le « conditionnel » de l'« implication ». Nous considérerons des formules conditionnelles, mais parlerons de l'implication entre deux formules. Le terme employé dépend du niveau où l'on se place. Pour plus de détails sur cette distinction, nous renvoyons à l'ouvrage de Quine cité en bibliographie.

à certains constituants d'une formule atomique. Une telle formule est toujours prise comme un tout. On peut par exemple supposer qu'elle est vraie et voir ce qu'il en résulte. On peut supposer qu'elle est fausse. On peut — en faisant d'autres hypothèses — essayer de prouver qu'une formule atomique est vraie. Ou qu'elle est fausse. Exemple : si «la climatisation est branchée et il fait frais» est considéré comme une formule atomique vraie, si «la climatisation est branchée» et «il fait frais» sont des formules atomiques dont on cherche à savoir si elles sont vraies ou fausses, on ne peut pas déduire la vérité de ces deux dernières formules de la première. Il n'est pas impossible que cette vérité puisse être déduite grâce à d'autres informations, mais pas grâce à celle-là.

Il faut aussi noter que si «la climatisation est branchée» n'est pas considérée comme une proposition, alors la déduction ne pourra pas être faite.

2.1.2 Les formules composées

On peut voir les affirmations suivantes comme des formules composées :

- «le tableau n'est pas propre»;
- «je n'ai pas écrit sur le tableau»;
- «le tableau n'est pas propre et je n'ai pas écrit sur le tableau»;
- «si le tableau est propre, alors je n'ai pas écrit sur le tableau»;
- «si le tableau n'est pas propre, alors personne ne vient de l'effacer».

En fait, ces affirmations traduisent l'application de la négation et du conditionnel, ou de leur combinaison aux formules atomiques données plus haut. Toutefois, il est clair que nous avons commis ici un abus de langage, puisque nous avons *apparemment pénétré* dans certaines formules atomiques. En toute rigueur, nous aurions dû dire que les formules composées sont les suivantes :

- non(«le tableau est propre»);
- non(«j'ai écrit sur le tableau»);
- non(«le tableau est propre») et non(«j'ai écrit sur le tableau»);
- (si «le tableau est propre», alors non(«j'ai écrit sur le tableau»));
- (si non(«le tableau est propre»), alors «personne ne vient de l'effacer»).

2.1.3 Le substrat de la logique

Le langage qui sous-tend les propositions (c'est-à-dire le langage employé entre les guillemets) est donc séparé du langage «logique» proprement dit (celui des connecteurs «et», «ou», «si», etc.). En quelque sorte, on peut imaginer que l'on a choisi un certain nombre (éventuellement infini) de briques de base (les propositions) et que ces briques peuvent être combinées suivant un nombre restreint de règles de formation parfaitement codifiées.

Le substrat peut être un langage plus codifié que le français. Il pourrait par exemple s'agir de formules d'arithmétique (cf. §1.3).

2.1.4 But du calcul des propositions

Le calcul des propositions a pour objectif de décider de la vérité de telle ou telle affirmation, et en particulier de vérifier tel ou tel raisonnement en formalisant précisément les connaissances (quelles sont les propositions vraies) et les moyens de déduire de nouvelles connaissances à partir de ce qui est déjà connu. Cette formalisation passe par l'abstraction du substrat propositionnel. En effet, comme les formules atomiques sont « fermées » à notre perspicacité, nous noterons celles-ci par des lettres a, b, \dots , appelées *variables propositionnelles*. a représentera une certaine affirmation, par exemple « le tableau est propre », mais sa nature exacte ne nous intéressera pas. Nous partirons en général d'un certain nombre de *faits* (par exemple : a est vrai, c est faux, (si a est vrai alors b est vrai, etc.) et essaierons de déterminer si une affirmation particulière est vraie, cette affirmation pouvant être une formule atomique (est-ce que b est vrai?) ou une formule composée (est-ce que a est vrai et b est faux?).

Les déductions se feront formellement et indépendamment du « contenu » des propositions. Par exemple, si « la vache est ou deviendra maigre », « la vache n'a rien à manger », « la vache vit dans un désert sans herbe », « la vache va mourir de faim » et « la vache a ou aura faim » sont des formules atomiques, et (si « la vache vit dans un désert sans herbe », alors « la vache a ou aura faim ») est une formule composée, on ne pourra pas déduire « la vache va mourir de faim » de « la vache vit dans un désert sans herbe » parce que la *seule* information dont nous pourrions nous servir—à savoir ce que nous savons de la vie dans un désert—est inutilisable ici car *nous n'avons pas le droit* d'entrer à l'intérieur des propositions. Ceci peut sembler être une limitation, mais en réalité ceci nous permet de ne pas nous préoccuper des problèmes d'ambiguïté de la langue du substrat et de formaliser précisément le raisonnement ou une partie d'un raisonnement. Si nous ne pouvons pas déduire un fait que nous aimerions déduire, nous pouvons ajouter des informations à l'ensemble de nos hypothèses. Par exemple, dans le cas précédent, on pourrait ajouter des règles (formules composées « si ... alors ... ») expliquant ce qui arrive à une vache dans un désert. Il se pose donc ici le problème de l'adéquation des hypothèses au raisonnement que nous souhaitons formaliser. Ce problème d'adéquation est très important, mais nous ne nous y intéresserons pas. Nous resterons constamment dans un cadre formel, tout en sachant que les hypothèses formelles que nous avons considérées correspondent à une certaine réalité et que si ces correspondances s'étendent à une description de l'évolution de la réalité, les déductions faites de manière purement formelles auront une correspondance dans la réalité. La plupart du temps, nous oublierons cette volonté de reproduire la réalité, même si c'est ce qui a motivé l'introduction à la logique depuis l'*Organon* d'Aristote.

2.2 Syntaxe

Nous décrivons maintenant formellement l'ensemble des formules de la logique propositionnelle :

Définition 1 Une variable propositionnelle a est une variable dont la valeur est une proposition élémentaire et non une composition de propositions élémentaires.

Les variables propositionnelles forment les constituants ultimes, les atomes, d'une formule de la logique propositionnelle.

Définition 2 Soit $\mathcal{P} = \{a, b, c, \dots\}$ un ensemble (éventuellement infini) de variables propositionnelles. L'ensemble F des formules de la logique propositionnelle est défini comme étant le plus petit ensemble S tel que :

- Si $p \in \mathcal{P}$ alors $p \in S$,
- Si A et B sont des éléments de S , alors $(\neg A)$, $(A \wedge B)$, $(A \vee B)$ et $(A \supset B)$ sont des éléments de S .

F contient donc a , b , $(\neg a)$, $(a \wedge b)$, $(a \wedge a)$, $(b \wedge a)$, a , $(a \supset (\neg b))$, etc. En pratique, nous n'indiquerons pas toujours les parenthèses.

Intuition La définition précédente collecte inductivement (ou récursivement) toutes les formules abstraites correspondant aux opérations concrètes suivantes :

- Négation (« non »), noté formellement \neg ;
- Conjonction (« et »), noté formellement \wedge ;
- Disjonction (« ou »), noté formellement \vee ;
- Groupement (parenthèses) ;
- Conditionnel matériel (« si ... alors »), noté formellement \supset ;

Les variables propositionnelles correspondent à certaines affirmations du monde concret.

Conditionnel matériel. Ce n'est pas la causalité. $A \supset B$ dit que B n'est pas plus faux que A . La causalité se traduit par le conditionnel. La tradition philonienne définit le conditionnel dans tous les cas de figure et ce uniquement par rapport à la vérité de ses constituants (et non en fonction des liens entre les constituants). (cf. Gochet et Gribomont, tome 1, page 32)

2.3 Vérité d'une formule

2.3.1 Introduction

Il faut un moyen de déterminer ce qui est vrai et ce qui est faux. L'interprétation qui nous servira de référence est l'interprétation structurelle, parce que qu'elle est la plus naturelle.

2.3.2 Algèbres de Boole

Une algèbre de Boole est une structure $\mathcal{B} = (B, \leq, \wedge, \vee, 0, 1)$ satisfaisant les propriétés suivantes :

1. $\{0, 1\} \subseteq B$, où $0 \neq 1$;
2. \vee et \wedge sont des fonctions de B^2 dans B ;
3. (B, \leq) est un ordre partiel tel que, pour chaque $x, y \in B$, $x \vee y \in B$ et $x \wedge y \in B$;
4. 1 est l'élément maximum de B et 0 est l'élément minimum de B , c'est-à-dire que pour chaque $x \in B$, $0 \leq x \leq 1$;

5. Pour chaque $x \in B$, il existe un élément de B , noté \bar{x} , tel que $x \wedge \bar{x} = 0$ et $x \vee \bar{x} = 1$.
On appelle \bar{x} le complément de x . - peut être vu comme une fonction de B dans B .
6. \vee est distributive par rapport à \wedge et \wedge est distributive par rapport à \vee , c'est-à-dire que pour chaque $x, y, z \in B$, $(x \vee y) \wedge z = (x \wedge z) \vee (y \wedge z)$ et $(x \wedge y) \vee z = (x \vee z) \wedge (y \vee z)$.

Exemple : d'après ce qui précède, on a les relations suivantes : $0 \wedge 0 = 0 \wedge 1 = 1 \wedge 0 = 0$, $1 \wedge 1 = 1$, $0 \vee 1 = 1 \vee 0 = 1 \vee 1 = 1$, $0 \vee 0 = 0$.

2.3.3 Interprétation des formules dans une algèbre de Boole

Pour déterminer si une formule est vraie ou fausse, nous lui faisons correspondre un élément de $\{0,1\}$, où 0 correspondra à « faux » et 1 à « vrai ». Plus précisément, nous établissons une correspondance entre deux structures, d'une part celle des formules du calcul des propositions avec les règles de formation \neg , \wedge , \vee et \supset , d'autre part l'algèbre de Boole $(\{0,1\}, \leq, \wedge, \vee, 0,1)$, structure souvent notée **2**.

Pour interpréter une formule dans l'algèbre de Boole **2**, on associe soit 0 soit 1 à l'ensemble des variables propositionnelles. On se donne donc une fonction de *valuation* ν :

$$\nu : \mathcal{P} \mapsto \{0,1\}$$

Les variables auxquelles on associe 0 correspondront à des propositions fausses, alors que les variables auxquelles on associe 1 correspondront à des propositions vraies.

Cette fonction ν n'est initialement définie que pour les variables propositionnelles. On l'étend à l'ensemble des formules de la manière suivante :

$$\nu(\neg A) := \overline{\nu(A)} \tag{2.1}$$

$$\nu(A \vee B) := \nu(A) \vee \nu(B) \tag{2.2}$$

$$\nu(A \wedge B) := \nu(A) \wedge \nu(B) \tag{2.3}$$

$$\nu(A \supset B) := \overline{\nu(A)} \vee \nu(B) \tag{2.4}$$

Intuition Ces définitions de ν correspondent au sens que l'on veut donner aux formules du calcul propositionnel. La première dit que $\neg A$ est faux lorsque A est vrai et réciproquement. La seconde dit que $A \vee B$ est vrai quand au moins A ou B est vrai. La troisième dit que $A \wedge B$ est vrai lorsque A et B est vrai. Enfin la dernière dit que $A \supset B$ est vrai si ou bien A est faux, ou bien B est vrai, ou bien les deux. Il est important de réaliser que les opérations \wedge et \vee à droite sont différentes des opérations de gauche. Ce n'est que par commodité que nous employons les mêmes.

La fonction ν est déterminée de manière unique par les équations précédentes, connaissant la valeur de ν pour les variables propositionnelles.

2.3.4 Tautologies et équivalences

On appelle tautologie une formule dont l'interprétation est 1, quelle que soit la valeur des variables propositionnelles. On parlera aussi de formule *valide*. Un exemple de formule valide

du calcul des propositions est $\varphi \vee \neg\varphi$, c'est-à-dire le *tiers exclus*. Que la valeur associée à φ soit 0 (fausse) ou 1 (vraie), l'interprétation de la formule est 1.

On dit que φ est équivalent à ψ et on le note $\varphi \equiv \psi$ si $(\varphi \supset \psi) \wedge (\psi \supset \varphi)$ est une tautologie. Nous donnons plus loin un certain nombre d'équivalences classiques, pouvant facilement être vérifiées au moyen de ce qui précède.

Par la suite, nous écrirons souvent «V» (pour Vrai) au lieu de 1 et «F» (pour Faux) au lieu de 0.

2.3.5 Tables de vérité

On peut déterminer la validité d'une formule du calcul propositionnel en considérant toutes les valeurs de vérité possibles pour les variables de la formules. Ces données s'arrangent dans un tableau que l'on appelle table de vérité. Les tables de vérité semblent avoir été introduites par Ludwig Wittgenstein.

Tables de vérité des connecteurs

Les tables de vérité des connecteurs logiques \neg , \wedge , \vee et \supset sont des représentations des valeurs de vérité des formules $\neg A$, $A \wedge B$, $A \vee B$ et $A \supset B$ en fonction de A et B . Ces tables correspondent directement à la définition de la fonction ν un peu plus haut (cf. §2.3.3).

– Négation :

A	$\neg A$
F	V
V	F

– Conjonction :

A	B	$A \wedge B$
V	V	V
V	F	F
F	V	F
F	F	F

– Disjonction :

A	B	$A \vee B$
V	V	V
V	F	V
F	V	V
F	F	F

– Conditionnelle :

A	B	$A \supset B$
V	V	V
V	F	F
F	V	V
F	F	V

Exemple

Voici par exemple la table de vérité de la formule F égale à $((p \wedge q) \supset \neg r) \equiv (p \supset (q \supset \neg r))$:

p	q	r	$\neg r$	$p \wedge q$	$q \supset \neg r$	$(p \wedge q) \supset \neg r$	$p \supset (q \supset \neg r)$	F
V	V	V	F	V	F	F	F	V
V	V	F	V	V	V	V	V	V
V	F	V	F	F	V	V	V	V
V	F	F	V	F	V	V	V	V
F	V	V	F	F	F	V	V	V
F	V	F	V	F	V	V	V	V
F	F	V	F	F	V	V	V	V
F	F	F	V	F	V	V	V	V

Si la dernière colonne de la table ne contient que des «V», cela signifie que la formule est vraie quelles que soient les valeurs des variables p , q et r , donc que c'est une formule valide du calcul des propositions.

Simplification d'une formule

On a vu que deux formules du calcul propositionnel peuvent être équivalentes. On peut chercher à trouver la «formule la plus petite» équivalente à une formule donnée. Ceci a des applications très importantes pour la conception des circuits digitaux, puisque cela permet de réduire le nombre de composants.

Inconvénient des tables de vérité

S'il y a trop de variables, l'emploi de tables de vérité pour déterminer la validité d'une formule n'est plus adapté.

2.4 Équivalences classiques

Les équivalences suivantes peuvent facilement être vérifiées par les méthodes vues précédemment. Ici A , B et C sont des formules quelconques du calcul des propositions.

Identité	$A \equiv A$
Lois de De Morgan	$\neg(A \wedge B) \equiv (\neg A \vee \neg B)$ $\neg(A \vee B) \equiv (\neg A \wedge \neg B)$
Commutativité	$(A \wedge B) \equiv (B \wedge A)$ $(A \vee B) \equiv (B \vee A)$
Associativité	$(A \wedge (B \wedge C)) \equiv ((A \wedge B) \wedge C)$ $(A \vee (B \vee C)) \equiv ((A \vee B) \vee C)$
Distributivité	$(A \wedge (B \vee C)) \equiv ((A \wedge B) \vee (A \wedge C))$ $(A \vee (B \wedge C)) \equiv ((A \vee B) \wedge (A \vee C))$
Contraposition	$(A \supset B) \equiv (\neg B \supset \neg A)$
Conditionnel matériel	$(A \supset B) \equiv (\neg A \vee B) \equiv \neg(A \wedge \neg B)$
Équivalence matérielle	$(A \equiv B) \equiv (A \supset B) \wedge (B \supset A) \equiv (A \wedge B) \vee (\neg A \wedge \neg B)$
Idempotence	$A \equiv (A \wedge A)$ $A \equiv (A \vee A)$
Exportation- importation	$((A \wedge B) \supset C) \equiv (A \supset (B \supset C))$
Double négation	$A \equiv \neg\neg A$
Absorption	$(A \wedge (A \vee B)) \equiv A$ $(A \vee (A \wedge B)) \equiv A$

2.5 Systèmes axiomatiques

L'application de la logique à l'une ou l'autre théorie scientifique, telle que l'arithmétique ou une quelconque branche de la physique, est quelquefois rendue explicite par la définition de ce qu'on appelle un *système axiomatique*. Certaines affirmations de la théorie sont pris comme points de départ sous le nom d'*axiomes*, et puis, d'autres affirmations, appelées *théorèmes*, sont engendrées en montrant qu'elles découlent logiquement des axiomes.

Les systèmes axiomatiques sont des systèmes formels.

2.5.1 Emploi d'un système axiomatique pour dériver des formules valides

On peut utiliser un système axiomatique pour obtenir toutes les formules valides dans le calcul des propositions. Nous allons sommairement présenter deux systèmes, chacun étant constitué d'axiomes et utilisant deux règles d'inférence particulières, le *modus ponens* et la *substitution*. Il existe de nombreux autres systèmes d'axiomes. Examinons les deux règles :

Le *modus ponens* Si on a prouvé A et $A \supset B$, alors on peut déduire B . A est appelé la *prémisse majeure* et $A \supset B$ la *prémisse mineure* de la règle du *modus ponens*.
Exemple : de $x > y$ et $(x > y) \supset (y < x)$ on peut déduire $y < x$.

La *substitution* On peut dans un schéma d'axiome remplacer une lettre par une formule quelconque, pourvu que toutes les lettres identiques soient remplacées par des formules identiques.

Axiomes de Whitehead et Russell (1910)²

Le système de Whitehead et Russell adopte comme symboles primitifs \neg et \vee et définit \supset , \wedge et \equiv à partir de ces derniers de la manière suivante :

1. $A \supset B$ est une abréviation de $\neg A \vee B$;
2. $A \wedge B$ est une abréviation de $\neg(\neg A \vee \neg B)$;
3. $A \equiv B$ est une abréviation de $(A \supset B) \wedge (B \supset A)$.

Ce système comprend cinq axiomes, assez évidents en soi, et les deux règles d'inférence précédentes. Les axiomes sont donnés ici en utilisant des symboles non primitifs, comme le faisaient Whitehead et Russell :

$$\begin{array}{ll}
 \text{A1} & (A \vee A) \supset A \\
 \text{A2} & B \supset (A \vee B) \\
 \text{A3} & (A \vee B) \supset (B \vee A) \\
 \text{A4} & (A \vee (B \vee C)) \supset (B \vee (A \vee C)) \\
 \text{A5} & (B \supset C) \supset ((A \vee B) \supset (A \vee C))
 \end{array}$$

(ces cinq axiomes ne sont pas indépendants les uns des autres; le quatrième peut être obtenu à partir des quatre autres)

Exemple de preuve: pour prouver que $(B \supset C) \supset ((A \supset B) \supset (A \supset C))$, on peut partir de l'axiome A5 :

$$\begin{array}{ll}
 (B \supset C) \supset ((A \vee B) \supset (A \vee C)) & \text{Axiome A5} \\
 (B \supset C) \supset ((\neg A \vee B) \supset (\neg A \vee C)) & \text{Idem, en remplaçant } A \text{ par } \neg A \\
 \neg A \vee B \equiv A \supset B & \text{définition} \\
 \neg A \vee C \equiv A \supset C & \text{définition} \\
 (B \supset C) \supset ((A \supset B) \supset (A \supset C)) & \text{substitution}
 \end{array}$$

Pour prouver que $\neg A \vee A$, on peut procéder ainsi :

$$\begin{array}{ll}
 (1) & B \supset (A \vee B) & \text{Axiome A2} \\
 (2) & A \supset (A \vee A) & (1) \text{ et subst.} \\
 (3) & (B \supset C) \supset ((A \vee B) \supset (A \vee C)) & \text{Axiome A5} \\
 (4) & (B \supset C) \supset ((\neg A \vee B) \supset (\neg A \vee C)) & (3) \text{ et subst.} \\
 (5) & (B \supset C) \supset ((A \supset B) \supset (A \supset C)) & (4) \text{ et déf. de } \supset \\
 (6) & ((A \vee A) \supset A) \supset ((A \supset (A \vee A)) \supset (A \supset A)) & (5) \text{ et subst.} \\
 (7) & (A \vee A) \supset A & \text{Axiome A1} \\
 (8) & (A \supset (A \vee A)) \supset (A \supset A) & (6) \text{ et (7) (modus ponens)} \\
 (9) & A \supset A & (2) \text{ et (8) (modus ponens)} \\
 (10) & \neg A \vee A & (9) \text{ et déf. de } \supset
 \end{array}$$

Plus loin nous donnons les axiomes de Lukasiewicz. Voici des preuves des deux premiers axiomes, dans le système de Whitehead et Russell. Ce sont les formules (8) et (18) de la

2. Il s'agit du système exposé dans l'ouvrage *Principia Mathematica*, 1910–1913.

dérivation suivante :

(1)	$(A \vee (B \vee C)) \supset (B \vee (A \vee C))$	Axiome A4
(2)	$(\neg(B \supset C) \vee (\neg(A \vee B) \vee (A \vee C)))$ $\supset (\neg(A \vee B) \vee (\neg(B \supset C) \vee (A \vee C)))$	(1) et substitution
(3)	$(B \supset C) \supset ((A \vee B) \supset (A \vee C))$	Axiome A5
(4)	$\neg(B \supset C) \vee ((A \vee B) \supset (A \vee C))$	(3) et déf. de \supset
(5)	$\neg(A \vee B) \vee (\neg(B \supset C) \vee (A \vee C))$	(2) et (4) (modus ponens)
(6)	$(A \vee B) \supset ((B \supset C) \supset (A \vee C))$	(5) par déf. de \supset
(7)	$(\neg A \vee B) \supset ((B \supset C) \supset (\neg A \vee C))$	(6) et subst.
(8)	$(A \supset B) \supset ((B \supset C) \supset (A \supset C))$	(7) et déf. de \supset
(9)	$(B \supset (A \vee B)) \supset (((A \vee B) \supset (B \vee A)) \supset (B \supset (B \vee A)))$	(8) et subst.
(10)	$((A \vee B) \supset (B \vee A)) \supset (B \supset (B \vee A))$	Axiome A2, (9) et modus ponens
(11)	$B \supset (B \vee A)$	Axiome A3, (10) et modus ponens
(12)	$\neg B \supset (\neg B \vee A)$	(11) et subst.
(13)	$\neg\neg B \vee (\neg B \vee A)$	(12) et déf. de \supset
(14)	$(\neg\neg B \vee (\neg B \vee A)) \supset (\neg B \vee (\neg\neg B \vee A))$	Axiome A4 et subst.
(15)	$\neg B \vee (\neg\neg B \vee A)$	(13), (14) et modus ponens
(16)	$B \supset (\neg\neg B \vee A)$	(15) et déf. de \supset
(17)	$B \supset (\neg B \supset A)$	(16) et déf. de \supset
(18)	$A \supset (\neg A \supset B)$	(17) et subst.

Nous laissons à titre d'exercice la preuve du troisième axiome de Lukasiewicz.

Axiomes de Lukasiewicz (1929)

Ce système comprend les trois axiomes suivants, plus les deux règles vues plus haut :

$$\begin{aligned} \text{A1} \quad & (A \supset B) \supset ((B \supset C) \supset (A \supset C)) \\ \text{A2} \quad & A \supset (\neg A \supset B) \\ \text{A3} \quad & (\neg A \supset A) \supset A \end{aligned}$$

Montrons comment prouver $A \supset A$ avec ces axiomes :

$A \supset (\neg A \supset A)$	(substitution de B par A dans A2)
$(A \supset B) \supset ((B \supset A) \supset (A \supset A))$	(substitution de C par A dans A1)
$(A \supset (\neg A \supset A)) \supset (((\neg A \supset A) \supset A) \supset (A \supset A))$	(substitution de B par $\neg A \supset A$)
$((\neg A \supset A) \supset A) \supset (A \supset A)$	(modus ponens)
$A \supset A$	(modus ponens)

2.5.2 Preuve à partir d'hypothèses

Définition 3 Une suite finie de formules B_1, B_2, \dots, B_m est appelée preuve à partir des hypothèses A_1, A_2, \dots, A_n si pour chaque i :

- B_i est l'une des hypothèses A_1, A_2, \dots, A_n ,
- ou B_i est une variante d'un axiome,,
- ou B_i est inférée (par application de la règle du modus ponens) à partir de la prémisse majeure B_j et de la prémisse mineure B_k où $j < i, k < i$,

- ou B_i est inférée (par application de la règle de substitution) à partir d'une prémisse antérieure B_j , la variable remplacée n'apparaissant pas dans A_1, A_2, \dots, A_n .

Une telle suite de formules — B_m étant la formule finale de la suite — est appelée plus explicitement preuve de B_n à partir des hypothèses A_1, A_2, \dots, A_n , ce qu'on note par $A_1, A_2, \dots, A_n \vdash B_m$.

On notera que lorsque l'on essaie de prouver un résultat à partir d'un certain nombre d'hypothèses, on n'essaie pas de prouver les hypothèses elles-mêmes.

2.6 Calcul des séquents

L'objectif d'un système de preuve est de fournir des règles pour *vérifier* qu'une formule est *vraie*. Un système de preuve *adapté* à une logique comporte des règles qui ne doivent nous permettre de ne trouver que du vrai et si possible tout ce qui est vrai.

Si donc l'on possède un moyen de déterminer la vérité d'une formule, l'emploi d'un système de preuve aura comme condition préalable la (méta)preuve que les règles préservent le vrai et ne permettent pas de prouver quelque chose de faux. Il s'agit d'une (méta)preuve de consistance que l'on verra plus loin.

Nous allons présenter un système de preuve qui nous permettra de prouver celles des formules de la forme $(\Gamma_1 \wedge \dots \wedge \Gamma_n) \supset (\Delta_1 \vee \dots \vee \Delta_m)$ qui sont vraies quelle que soit la valuation de variables propositionnelles. Ce système de preuve nous permettra en particulier de vérifier si une formule est vraie ou non, selon l'interprétation booléenne.

Il s'agit de réaliser que le système de preuve que l'on utilise détermine la *manière* de faire les preuves.

2.6.1 Présentation

Le calcul des séquents³ est un système de preuve manipulant des *séquents*, c'est-à-dire des listes de symboles de la forme $\varphi_1, \dots, \varphi_n \rightarrow \psi_1, \dots, \psi_m$, où les φ_i et ψ_i sont des formules du calcul des propositions. L'antécédent d'un séquent est ce qui précède \rightarrow et le succédent ce qui le suit.

Intuition Un séquent $\varphi_1, \dots, \varphi_n \rightarrow \psi_1, \dots, \psi_m$ signifie que de φ_1 et $\varphi_2, \dots, \varphi_n$ on peut déduire ψ_1 ou ψ_2, \dots, ψ_m , dans le même sens que $p \supset q$ représente la déduction de q à partir de p . Un séquent est une représentation d'un ensemble $\Psi = \{\psi_1, \dots, \psi_m\}$ de connaissances qui découlent toutes d'un ensemble $\Phi = \{\varphi_1, \dots, \varphi_n\}$. On montrera plus loin que si $\varphi_1, \dots, \varphi_n \rightarrow \psi_1, \dots, \psi_m$ est un séquent valide (c'est-à-dire déductible des axiomes par application des règles d'inférence) alors $(\varphi_1 \wedge \dots \wedge \varphi_n) \supset (\psi_1 \vee \dots \vee \psi_m)$ est une formule vraie du calcul des propositions, et réciproquement (théorèmes de correction et de complétude). Ceci veut donc dire que le fait d'employer les règles du calcul des séquents revient à se forcer à ne manipuler que des formules propositionnelles d'une certaine forme. Il sera donc

3. Le système présenté ici est une variante du système LK (K pour Klassische) introduit par Gerhard Gentzen dans les années 1930.

nécessaire de montrer que cette «restriction» n'en est pas une réellement, c'est-à-dire qu'elle ne nous empêche pas de montrer la validité d'une formule propositionnelle particulière.

Une règle d'inférence en calcul des séquents va donc naturellement être un moyen d'obtenir un nouveau séquent à partir de un ou plusieurs autres séquents. Les règles d'inférence du calcul des séquents sont soit de la forme $\frac{A}{B}$ soit de la forme $\frac{A}{C} \frac{B}{C}$ où A , B et C sont des séquents.

Il est important de s'arrêter sur les symboles utilisés dans le calcul des séquents. Nous avons vu que les formules du calcul propositionnel contiennent des variables propositionnelles (a , b , ...) ainsi que les symboles \wedge , \vee , \neg , \supset et les parenthèses. Le calcul des séquents ne manipule pas des formules du calcul propositionnel, mais des séquents et ces séquents comportent un symbole supplémentaire: « \rightarrow ». Nous avons choisi ce symbole pour plusieurs raisons. D'une part, c'est ce symbole qu'utilise Gentzen lui-même. D'autre part, nous voulons éviter autant que possible de créer une confusion. Aussi, nous n'avons pas suivi Gochet et Gribomont qui emploient « \Rightarrow » là où nous employons « \rightarrow ». « \Rightarrow » est en effet un symbole très souvent utilisé pour représenter le conditionnel. Or nous avons préféré utiliser le symbole \supset de Peano. D'autres ouvrages emploient « \vdash » là où nous avons utilisé « \rightarrow », mais nous ne l'utilisons pas ici parce que c'est un symbole pouvant prêter à confusion. En effet, en général, le symbole « \vdash » est réservé aux déductions et est en cela similaire à la barre horizontale dans les règles qui suivent. Les ouvrages qui emploient « \vdash » emploient souvent « \Rightarrow » là où nous employons « \supset ». Certains ouvrages (comme celui de Lalemant) emploient « $:$ » à la place de notre « \rightarrow ». Et certains ouvrages utilisent « \rightarrow » là où nous employons « \supset ». Il est important de ne pas se laisser induire en erreur par ces variantes syntaxiques. Le contexte permet en général de résoudre les problèmes d'interprétation.

Nous allons maintenant donner les règles d'inférence du calcul des séquents. Ces règles diffèrent de celles données par Gentzen par le fait qu'elles sont réversibles. Ce point sera détaillé plus loin. Nous donnerons pour chacune des règles une justification intuitive, correspondant à ce que la règle est censée représenter. *Ces intuitions ne sont pas des démonstrations.* Leur but est de montrer que ces règles ne sont pas si artificielles que cela. Nous noterons des formules particulières avec des lettres grecques minuscules (par exemple φ) et des listes de formules par des lettres grecques majuscules (par exemple Γ). Des lettres différentes peuvent correspondre à la même formule ou au même ensemble de formule (tout comme en arithmétique, x et y peuvent avoir la même valeur).

Les règles d'inférence du calcul des séquents se répartissent en plusieurs groupes : axiomes, règles structurelles, règles logiques et règle de coupure.

Axiomes

Les axiomes du calcul des séquents sont les séquents de la forme $\Gamma_1, \varphi, \Gamma_n \rightarrow \Delta_1, \varphi, \Delta_m$, c'est-à-dire des séquents où l'une des formules (φ) apparaît à la fois dans l'antécédent et dans le succédent.

Intuition *Cet axiome correspond à l'identité : à partir d'un certain nombre de formules, incluant φ , on peut déduire entre autres φ .*

Règles structurelles

Les règles structurelles sont des règles permettant d'ignorer l'ordre ou la répétition des formules dans un séquent.

- Règle de permutation dans l'antécédent, notée G_P (pour Gauche Permutation):

$$\frac{\Gamma_1, \varphi, \psi, \Gamma_2 \rightarrow \Delta}{\Gamma_1, \psi, \varphi, \Gamma_2 \rightarrow \Delta} \quad (G_P)$$

- Règle de permutation dans le succédent, notée D_P (pour Droite Permutation):

$$\frac{\Gamma \rightarrow \Delta_1, \varphi, \psi, \Delta_2}{\Gamma \rightarrow \Delta_1, \psi, \varphi, \Delta_2} \quad (D_P)$$

- Règle de contraction dans l'antécédent, notée G_C (pour Gauche Contraction):

$$\frac{\Gamma, \varphi, \varphi \rightarrow \Delta}{\Gamma, \varphi \rightarrow \Delta} \quad (G_C)$$

- Règle de contraction dans le succédent, notée D_C (pour Droite Contraction):

$$\frac{\Gamma \rightarrow \Delta, \varphi, \varphi}{\Gamma \rightarrow \Delta, \varphi} \quad (D_C)$$

- Règle d'atténuation (ou d'affaiblissement) dans l'antécédent, notée G_A (pour Gauche Atténuation):

$$\frac{\Gamma \rightarrow \Delta}{\Gamma, \varphi \rightarrow \Delta} \quad (G_A)$$

- Règle d'atténuation (ou d'affaiblissement) dans le succédent, notée D_A (pour Droite Atténuation):

$$\frac{\Gamma \rightarrow \Delta}{\Gamma \rightarrow \Delta, \varphi} \quad (D_A)$$

Règles logiques

Dans la liste qui suit, nous justifions intuitivement le passage de la (ou des) prémisses(s) à la conclusion. Nous ne justifions pas ici le passage inverse qui est aussi vrai.

- Règle d'introduction de la négation dans l'antécédent, notée G_{\neg} :

$$\frac{\Gamma \rightarrow \varphi, \Delta}{\Gamma, \neg \varphi \rightarrow \Delta} \quad (G_{\neg})$$

Intuition Cette règle correspond à un raisonnement par cas: si de Γ on peut déduire φ ou Δ , alors si φ n'est pas vrai, Δ l'est nécessairement.

- Règle d'introduction de la négation dans le succédent, notée D_{\neg} :

$$\frac{\Gamma, \varphi \rightarrow \Delta}{\Gamma \rightarrow \neg \varphi, \Delta} \quad (D_{\neg})$$

Intuition Si de Γ et φ on peut déduire Δ , alors de Γ seul on peut déduire soit $\neg \varphi$ soit Δ . En effet, s'il n'en était pas ainsi $\neg \varphi$ et Δ seraient faux, φ serait donc vrai; et l'on aurait donc à la fois Γ et φ vrais, mais Δ faux, ce qui contredit la prémisse.

- Règle d'introduction de la conjonction dans l'antécédent, noté G_{\wedge} :

$$\frac{\Gamma, \varphi, \psi \rightarrow \Delta}{\Gamma, \varphi \wedge \psi \rightarrow \Delta} \quad (G_{\wedge})$$

Intuition De Γ et $\varphi \wedge \psi$, on peut déduire tout ce que l'on peut déduire de Γ , φ et ψ , puisque $\varphi \wedge \psi$ correspond intuitivement à la conjonction, tout comme l'apposition de φ et ψ dans l'antécédent d'un séquent.

- Règles d'introduction de la conjonction dans le succédent, notée D_\wedge :

$$\frac{\Gamma \rightarrow \varphi, \Delta \quad \Gamma \rightarrow \psi, \Delta}{\Gamma \rightarrow \varphi \wedge \psi, \Delta} \quad (D_\wedge)$$

Intuition Si de Γ nous ne pouvons déduire Δ , nous pouvons déduire φ et ψ , ce qui est équivalent — intuitivement — avec la déduction de $\varphi \wedge \psi$ de Γ .

- Règle d'introduction de la disjonction dans l'antécédent, notée G_\vee :

$$\frac{\Gamma, \varphi \rightarrow \Delta \quad \Gamma, \psi \rightarrow \Delta}{\Gamma, \varphi \vee \psi \rightarrow \Delta} \quad (G_\vee)$$

Intuition De Γ et $\varphi \vee \psi$, on peut déduire φ ou ψ parce que la correspondance voulue de $\varphi \vee \psi$ est « φ ou ψ ». Si on peut déduire φ , on pourra déduire Δ par hypothèse. Sinon, on peut déduire ψ et donc Δ par hypothèse. Donc, dans tous les cas, on peut déduire Δ de Γ et $\varphi \vee \psi$.

- Règle d'introduction de la disjonction dans le succédent, notée D_\vee :

$$\frac{\Gamma \rightarrow \varphi, \psi, \Delta}{\Gamma \rightarrow \varphi \vee \psi, \Delta} \quad (D_\vee)$$

Intuition Pouvoir déduire $\varphi \vee \psi$ ou Δ est équivalent — en ce sens que nous voulons donner à ces expressions la même correspondance — au fait de pouvoir déduire ou bien φ ou bien ψ ou bien Δ .

- Règle d'introduction du conditionnel dans l'antécédent, notée G_\supset

$$\frac{\Gamma_1, \Gamma_2 \rightarrow \Delta_1, \varphi, \Delta_2 \quad \Gamma_1, \psi, \Gamma_2 \rightarrow \Delta_1, \Delta_2}{\Gamma_1, \varphi \supset \psi, \Gamma_2 \rightarrow \Delta_1, \Delta_2} \quad (G_\supset)$$

Intuition La règle D_\supset est justifiée intuitivement comme suit. Selon les hypothèses, de Γ_1 et Γ_2 , on déduit soit φ , soit Δ_1 ou Δ_2 . Le second cas justifie immédiatement la règle. Dans le premier cas, où l'on déduit φ de Γ_1 et Γ_2 , on va pouvoir déduire ψ de Γ_1 , $\varphi \supset \psi$ et Γ_2 , puisque $\varphi \supset \psi$ correspond aussi au fait de pouvoir déduire ψ de φ . La seconde hypothèse permet alors de déduire Δ_1 et Δ_2 .

- Règle d'introduction du conditionnel dans le succédent, notée D_\supset

$$\frac{\Gamma, \varphi \rightarrow \psi, \Delta}{\Gamma \rightarrow \varphi \supset \psi, \Delta} \quad (D_\supset)$$

Intuition On a vu (règle D_\neg) que l'hypothèse implique que de Γ , on peut déduire soit $\neg\varphi$, soit ψ , soit Δ . Si l'on ne peut déduire Δ de Γ , on peut donc déduire soit $\neg\varphi$, soit ψ , ce qui correspond justement à la définition du conditionnel $\varphi \supset \psi$. (si φ alors ψ — ou bien de φ on peut « déduire » ψ — si, ou bien ψ est vrai, ou bien φ est faux.)

On notera que toutes les règles de ce groupe sont des règles dites d'introduction, parce qu'elles font apparaître un connecteur. Ces règles ne peuvent faire disparaître un connecteur.

Règle de coupure

$$\frac{\Gamma_1 \rightarrow \varphi, \Delta_1 \quad \Gamma_2, \varphi \rightarrow \Delta_2}{\Gamma_1, \Gamma_2 \rightarrow \Delta_1, \Delta_2} \quad (C)$$

Intuition La règle de coupure correspond au raisonnement suivant : si de Γ_1 on peut déduire φ ou Δ_1 , on peut en faire de même à partir de Γ_1 et Γ_2 (cf. règle d'atténuation). De même, si de Γ_2 et φ , on peut déduire Δ_2 , on peut en faire de même à partir de Γ_1 , Γ_2 et φ . Par conséquent, de Γ_1 et Γ_2 on peut déduire soit φ , soit Δ_1 (soit les deux). Dans le premier cas, on déduit φ ce qui d'après ce que nous venons de dire nous permet de déduire Δ_2 . Dans le second cas, on déduit Δ_1 . En somme, à partir de Γ_1 et Γ_2 on déduit soit Δ_1 soit Δ_2 .

2.6.2 Réversibilité des règles

Toutes les règles d'introduction sont réversibles, y compris les règles D_\wedge et G_\vee , et c'est ce fait que nous utilisons dans les preuves (cf. §2.6.7).

2.6.3 Récapitulation des règles

Nous récapitulons ici l'ensemble des règles. On notera en particulier leur caractère très symétrique.

Axiomes

$$\boxed{\Gamma_1, \varphi, \Gamma_n \rightarrow \Delta_1, \varphi, \Delta_m \quad (Ax)}$$

Règles structurelles

$$\boxed{\begin{array}{ll} \frac{\Gamma_1, \varphi, \psi, \Gamma_2 \rightarrow \Delta}{\Gamma_1, \psi, \varphi, \Gamma_2 \rightarrow \Delta} \quad (G_P) & \frac{\Gamma \rightarrow \Delta_1, \varphi, \psi, \Delta_2}{\Gamma \rightarrow \Delta_1, \psi, \varphi, \Delta_2} \quad (D_P) \\ \frac{\Gamma, \varphi, \varphi \rightarrow \Delta}{\Gamma, \varphi \rightarrow \Delta} \quad (G_C) & \frac{\Gamma \rightarrow \Delta, \varphi, \varphi}{\Gamma \rightarrow \Delta, \varphi} \quad (D_C) \\ \frac{\Gamma \rightarrow \Delta}{\Gamma, \varphi \rightarrow \Delta} \quad (G_A) & \frac{\Gamma \rightarrow \Delta}{\Gamma \rightarrow \Delta, \varphi} \quad (D_A) \end{array}}$$

Règles logiques

$$\boxed{\begin{array}{ll} \frac{\Gamma \rightarrow \varphi, \Delta}{\Gamma, \neg \varphi \rightarrow \Delta} \quad (G_\neg) & \frac{\Gamma, \varphi \rightarrow \Delta}{\Gamma \rightarrow \neg \varphi, \Delta} \quad (D_\neg) \\ \frac{\Gamma, \varphi, \psi \rightarrow \Delta}{\Gamma, \varphi \wedge \psi \rightarrow \Delta} \quad (G_\wedge) & \frac{\Gamma \rightarrow \varphi, \psi, \Delta}{\Gamma \rightarrow \varphi \vee \psi, \Delta} \quad (D_\vee) \\ \frac{\Gamma \rightarrow \varphi, \Delta \quad \Gamma \rightarrow \psi, \Delta}{\Gamma \rightarrow \varphi \wedge \psi, \Delta} \quad (D_\wedge) & \frac{\Gamma, \varphi \rightarrow \Delta \quad \Gamma, \psi \rightarrow \Delta}{\Gamma, \varphi \vee \psi \rightarrow \Delta} \quad (G_\vee) \\ \frac{\Gamma_1, \Gamma_2 \rightarrow \Delta_1, \varphi, \Delta_2 \quad \Gamma_1, \psi, \Gamma_2 \rightarrow \Delta_1, \Delta_2}{\Gamma_1, \varphi \supset \psi, \Gamma_2 \rightarrow \Delta_1, \Delta_2} \quad (G_\supset) & \frac{\Gamma, \varphi \rightarrow \psi, \Delta}{\Gamma \rightarrow \varphi \supset \psi, \Delta} \quad (D_\supset) \end{array}}$$

Règle de coupure

$$\boxed{\frac{\Gamma_1 \rightarrow \varphi, \Delta_1 \quad \Gamma_2, \varphi \rightarrow \Delta_2}{\Gamma_1, \Gamma_2 \rightarrow \Delta_1, \Delta_2} \quad (C)}$$

2.6.4 Validité d'un séquent

Nous allons définir la *validité* d'un séquent. Les théorèmes qui suivent font le lien entre la notion de validité et celle de prouvabilité.

Pour définir la validité d'un séquent, nous étendons la fonction de valuation ν aux séquents : par définition, $\nu(\Gamma \rightarrow \Delta) \equiv \nu((\phi_1 \wedge \dots \wedge \phi_n) \supset (\psi_1 \vee \dots \vee \psi_m))$, où $\Gamma = \phi_1, \dots, \phi_n$ et $\Delta = \psi_1, \dots, \psi_m$. On définit aussi les cas particuliers $\nu(\rightarrow \Delta) \equiv \nu(\Delta)$, $\nu(\Gamma \rightarrow) \equiv \neg\nu(\Gamma)$ et $\nu(\rightarrow) \equiv 0$.

Le séquent $\Gamma \rightarrow \Delta$ est alors dit valide si $\nu(\Gamma \rightarrow \Delta) = 1$ pour tout ν .

Le fait que le séquent $\Gamma \rightarrow \Delta$ est valide est noté $\Gamma \models \Delta$.

2.6.5 Théorème de correction (ou consistance)

Théorème 1 *S'il existe une preuve de $\Gamma \rightarrow \Delta$ dans le calcul des séquents, alors $\Gamma \models \Delta$.*

Démonstration 1 *Nous démontrons ce théorème par récurrence sur la longueur de la preuve de $\Gamma \rightarrow \Delta$, où la longueur est le nombre maximal d'applications de règles entre un séquent et un axiome. Les axiomes sont des preuves de longueur 0.*

1. *Pas de base : les preuves les plus courtes sont des applications des axiomes. Il faut donc montrer que $\frac{}{\Gamma_1, \varphi, \Gamma_2 \rightarrow \Delta_1, \varphi, \Delta_2}$ (Ax) implique $\nu(\Gamma_1, \varphi, \Gamma_2 \rightarrow \Delta_1, \varphi, \Delta_2) = 1$ pour toute valuation ν . Nous ne détaillons pas plus la preuve.*
2. *Supposons que le théorème soit vrai pour les preuves de longueur inférieure ou égale à n . Une preuve de longueur $n + 1$ finira donc par l'application d'une règle de la forme $\frac{H_1, \dots, H_k}{C}$ où H_1, \dots, H_k et C sont des séquents. On montre alors que si pour tout i , $\nu(H_i) = 1$, alors $\nu(C) = 1$. Nous ne détaillons pas plus la preuve.*

2.6.6 Théorème de complétude

Théorème 2 *Si $\Gamma \models \Delta$, alors il existe une preuve de $\Gamma \rightarrow \Delta$ dans le calcul des séquents.*

Démonstration 2 *La démonstration se fait par récurrence sur le nombre de connecteurs qui apparaissent dans le séquent $\Gamma \rightarrow \Delta$.*

1. *Cas de base : aucun connecteur n'apparaît dans $\Gamma \rightarrow \Delta$. Alors $\Gamma \rightarrow \Delta = \varphi_1, \dots, \varphi_n \rightarrow \psi_1, \dots, \psi_m$, où chaque φ_i et ψ_j est une variable propositionnelle. Supposons que pour tout $i \in \llbracket 1, n \rrbracket$ et pour tout $j \in \llbracket 1, m \rrbracket$, $\varphi_i \neq \psi_j$. Ceci nous permet de choisir la valuation ν telle que $\nu(\varphi_1) = \dots = \nu(\varphi_n) = 1$ et $\nu(\psi_1) = \dots = \nu(\psi_m) = 0$ qui entraîne $\nu(\Gamma \rightarrow \Delta) = 0$. Ceci contredit l'hypothèse selon laquelle $\Gamma \models \Delta$. Par conséquent, il existe $i \in \llbracket 1, n \rrbracket$ et $j \in \llbracket 1, m \rrbracket$ tels que $\varphi_i = \psi_j$. Et $\Gamma \rightarrow \Delta$ est donc un axiome du calcul des séquents.*

2. Supposons maintenant que le théorème soit prouvé pour tous les séquents $\Gamma \rightarrow \Delta$ avec n connecteurs. Pour montrer que le théorème est vrai avec $n + 1$ connecteurs, on examine toutes les formes que peut prendre le séquent $\Gamma \rightarrow \Delta$. Quelques unes des formes possibles sont:

- (a) $\Gamma_1, \neg\varphi, \Gamma_2 \rightarrow \Delta_1, \Delta_2$
- (b) $\Gamma_1, \Gamma_2 \rightarrow \Delta_1, \neg\varphi, \Delta_2$
- (c) $\Gamma_1, \varphi \wedge \psi, \Gamma_2 \rightarrow \Delta$
- (d) $\Gamma \rightarrow \Delta_1, \varphi \wedge \psi, \Delta_2$
- (e) $\Gamma_1, \varphi \vee \psi, \Gamma_2 \rightarrow \Delta$
- (f) $\Gamma \rightarrow \Delta_1, \varphi \vee \psi, \Delta_2$
- (g) $\Gamma_1, \Gamma_2 \rightarrow \Delta_1, \varphi \supset \psi, \Delta_2$
- (h) $\Gamma_1, \varphi \supset \psi, \Gamma_2 \rightarrow \Delta_1, \Delta_2$

Traisons par exemple le dernier cas. Par hypothèse, pour chaque valuation ν , on a $\nu(\Gamma_1, \varphi \supset \psi, \Gamma_2 \rightarrow \Delta_1, \Delta_2) = 1$. Donc au moins une des possibilités suivantes est vérifiée :

- (a) il existe $\gamma \in \Gamma_1$ tel que $\nu(\gamma) = 0$,
- (b) $\nu(\varphi \supset \psi) = 0$,
- (c) il existe $\gamma \in \Gamma_2$ tel que $\nu(\gamma) = 0$,
- (d) il existe $\delta \in \Delta_1$ tel que $\nu(\delta) = 1$,
- (e) il existe $\delta \in \Delta_2$ tel que $\nu(\delta) = 1$.

Toutes ces conditions garantissent que $\nu(\Gamma_1, \psi, \Gamma_2 \rightarrow \Delta_1, \Delta_2) = 1$ et $\nu(\Gamma_1, \Gamma_2 \rightarrow \Delta_1, \varphi, \Delta_2) = 1$.

Il y a moins de connecteurs dans ces séquents que dans le séquent de départ. Alors, par hypothèse de récurrence, ces deux séquents sont prouvables et on obtient une preuve de $\Gamma_1, \varphi \supset \psi, \Gamma_2 \rightarrow \Delta_1, \Delta_2$ en appliquant la règle G_{\supset} .

On procède de manière analogue dans les autres cas.

2.6.7 Exemples de preuves

Que faire avec ces règles et comment prouver avec elles une formule quelconque de la logique propositionnelle?

Si l'on veut prouver la formule φ , on cherche à écrire une preuve du séquent $\rightarrow \varphi$.

Exemple 1 Une preuve du séquent $\rightarrow A \vee \neg A$ est :

$$\frac{\frac{\frac{}{A \rightarrow A} (Ax)}{\rightarrow A, \neg A} (D_{\neg})}{\rightarrow A \vee \neg A} (D_{\vee})$$

Exemple 2 Une preuve du séquent $\rightarrow A \supset (B \supset (A \wedge B))$ est :

$$\frac{\frac{\frac{\overline{A, B \rightarrow A} (Ax)}{A, B \rightarrow A \wedge B} (D_{\wedge})}{A \rightarrow B \supset (A \wedge B)} (D_{\supset})}{\rightarrow A \supset (B \supset (A \wedge B))} (D_{\supset})$$

Ces exemples devraient convaincre le lecteur de la manière purement mécanique utilisée pour faire les preuves. C'est — nous le répétons — l'un des principaux avantages du calcul des séquents. Nous invitons le lecteur à prouver $A \vee \neg A$ ou $A \supset (B \supset (A \wedge B))$ avec les systèmes axiomatiques proposés plus haut pour le convaincre de la réflexion qu'ils nécessitent.

2.6.8 Exercices

Donner des preuves des séquents suivants :

1. $\rightarrow \neg\neg A \supset A$
2. $A \supset B \rightarrow (A \supset \neg B) \supset \neg A$
3. $\rightarrow (A \supset B) \supset ((A \supset (B \supset C)) \supset (A \supset C))$
4. $A \supset B \rightarrow (C \supset A) \supset (C \supset B)$
5. $(A \wedge B) \supset C \rightarrow A \supset (B \supset C)$
6. $A \wedge (B \vee C) \rightarrow (A \wedge B) \vee (A \wedge C)$
7. $(A \vee B) \wedge (A \vee C) \rightarrow A \vee (B \wedge C)$

Chapitre 3

Calcul des prédicats : exemples introductifs

3.1 Le syllogisme ou les catégories d'Aristote

Aristote présente dans l'*Organon* un type de raisonnement appelé *syllogisme* dont voici un exemple bien connu :

Tous les hommes sont mortels
Socrate est un homme
Donc, Socrate est mortel

Le syllogisme introduit l'idée de quantification : on parle de choses qui existent, ou vraies pour certains individus et pas d'autres.

Un syllogisme est toujours constitué de deux prémisses et d'une conclusion. Les prémisses et la conclusion adoptent l'une des quatre formes (catégories) suivantes :

1. Tout A est B
2. Nul A n'est B
3. Quelque A est B
4. Quelque A n'est pas B

Dans l'ensemble du syllogisme, les trois prédicats qui interviennent apparaissent tous chacun deux fois. C'est donc du rapport de deux quantités à une même troisième que l'on déduit leur rapport réciproque.

Le syllogisme introductif ne semble pas employer les catégories standard. En fait si. Il faut lire le syllogisme de la manière suivante :

Tous les hommes sont mortels
Tout (individu qui est Socrate) est un homme
Donc, tout (individu qui est Socrate) est mortel

C'est la forme la plus simple du syllogisme :

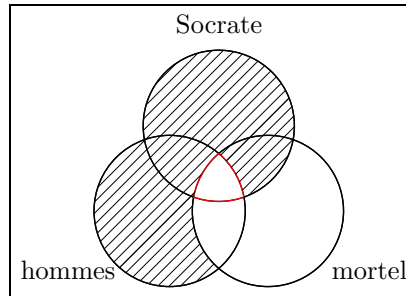
1. Tout A est B

2. Tout C est A
3. Donc, tout C est B

3.2 Diagrammes de Venn

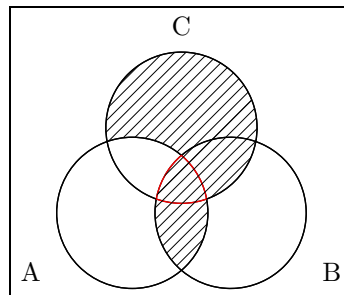
Les diagrammes de Venn¹ sont des diagrammes permettant de vérifier tel ou tel syllogisme. Les formes « Tout ... » ou « Nul ... » conduisent à griser certaines zones (celles où il n'y a rien), alors que les formes « Quelque ... » conduisent à mettre des croix. Nous donnons quelques exemples et nous les expliquons brièvement.

- (1) Tout homme est mortel; (2) (Tout) Socrate est un homme;



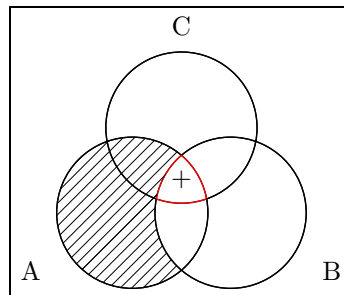
Par (1), tous les hommes non mortels sont grisés. Par (2), tous les Socrate(s) qui ne sont pas des hommes sont grisés aussi. On peut alors lire : (Tout) Socrate est mortel.

- (1) Nul A n'est B; (2) Tout C est A;



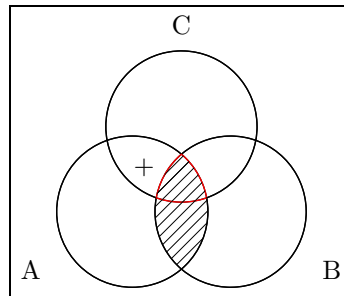
Par (1), la partie commune à A et B est grisée. Par (2), la partie de C en dehors de A est grisée. On peut alors lire : Nul B n'est C.

- (1) Tout A est B; (2) Quelque C est A;

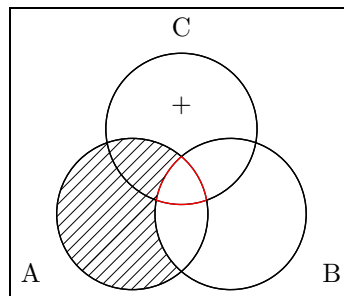


1. Cette méthode a été proposée par John Venn dans un article de 1880 intitulé *On the Diagrammatic and Mechanical Representation of Propositions and Reasonings*.

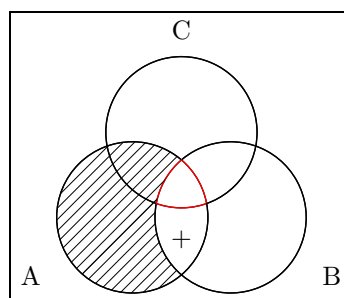
- Par (1), toute la partie de A qui est en dehors de B est barrée. Par (2), il y a un élément de C dans A et on place une croix dans la partie commune à C et A non grisée². On peut alors lire : Quelque C est B . (on notera qu'il y a plusieurs conclusions possibles)
- (1) Nul A n'est B ; (2) Quelque C est A ;



- Par (1), toute la partie de A qui est dans B est barrée. Par (2), il y a un élément de C dans A et on place une croix dans la partie commune à C et A non grisée. On peut alors lire : Quelque A n'est pas B . (par exemple)
- (1) Tout A est B ; (2) Quelque C n'est pas B ;



- Par (1), toute la partie de A qui est en dehors de B est barrée. Par (2), il y a un élément de C dans la partie en dehors de B ; on place une croix dans la partie non barrée de cette zone. On peut alors lire : Quelque C n'est pas A .
- (1) Tout A est B ; (2) Quelque A n'est pas C ;



Par (1), toute la partie de A qui est en dehors de B est barrée. Par (2), il y a un élément de A dans la partie en dehors de C ; on place une croix dans la partie non barrée de cette zone. On peut alors lire : Quelque A est B .

2. On voit au passage qu'il est très important de d'abord interpréter les affirmations universelles, avant les affirmations particulières.

3.3 Limites du calcul des propositions

Le calcul des propositions n'est pas adapté à la manipulation des syllogismes. Les syllogismes quant à eux ne permettent pas d'exprimer le calcul des propositions. Il nous faut pouvoir exprimer des faits sur *tous* les individus d'une classe, ou sur au moins un d'entre eux, mais sans pour autant savoir lequel. Le calcul des prédicats permet de formaliser et d'analyser naturellement les syllogismes d'Aristote.

3.4 Limites des diagrammes de Venn

L'exemple suivant est repris de Lewis Carroll :

1. Les seuls animaux dans cette maison sont des chats ;
2. Tout animal qui aime contempler la lune est apte à devenir un animal familier ;
3. Quand je déteste un animal, je l'évite soigneusement ;
4. Aucun animal n'est carnivore, à moins qu'il n'aille rôder dehors la nuit ;
5. Aucun chat ne manque jamais de tuer les souris ;
6. Aucun animal ne s'attache jamais à moi, excepté ceux qui sont dans la maison ;
7. Les kangourous ne sont pas aptes à devenir des animaux familiers ;
8. Aucun animal non carnivore ne tue de souris ;
9. Je déteste les animaux qui ne s'attachent pas à moi ;
10. Les animaux qui vont rôder dehors la nuit aiment toujours contempler la lune ;

Conclusion: J'évite toujours soigneusement les kangourous.

Pour vérifier ce raisonnement à l'aide des diagrammes de Venn, il faut le découper en parties. Ainsi, de (1) et (5), on conclut (11) : Tous les animaux dans cette maison tuent des souris. Pour cette étape on peut utiliser un diagramme de Venn. De même, de (8) et (11), on déduit : Tous les animaux de cette maison sont carnivores. Et ainsi de suite.

Les déductions dans le calcul des prédicats sont cependant beaucoup plus simples. Voici une transcription un peu plus formelle de l'exemple précédent, où la variable libre x est *implicitement* quantifiée universellement. Les déductions se font aisément à l'aide du *modus ponens*.

1. $e(x) \supset c(x)$
2. $l(x) \supset n(x)$
3. $d(x) \supset a(x)$
4. $m(x) \supset b(x)$
5. $c(x) \supset k(x)$
6. $r(x) \supset e(x)$
7. $h(x) \supset \neg n(x)$
8. $\neg b(x) \supset \neg k(x)$

9. $\neg r(x) \supset d(x)$
10. $m(x) \supset l(x)$

Conclusion : $h(x) \supset a(x)$

($e(x) \rightarrow$ « dans cette maison », $c(x) \rightarrow$ « chat », $l(x) \rightarrow$ « qui aime contempler la lune »,
 $n(x) \rightarrow$ « apte à devenir un animal familier », $d(x) \rightarrow$ « je déteste », $a(x) \rightarrow$ « je évite
soigneusement », $m(x) \rightarrow$ « rôder dehors la nuit », $b(x) \rightarrow$ « carnivore », $k(x) \rightarrow$ « tue les
souris », $r(x) \rightarrow$ « s'attache à moi », $h(x) \rightarrow$ « est un kangourou »)

Chapitre 4

Calcul des prédicats (logique du premier ordre)

4.1 Introduction

La logique propositionnelle ne nous permet pas d'affirmer des choses générales sur les éléments d'un ensemble par exemple. Dans ce sens, la logique propositionnelle ne reflète qu'une partie du raisonnement. Le calcul des prédicats au contraire permet de manipuler formellement des affirmations telles que «il existe un x tel que [x a une voiture américaine]» ou «pour tous les x [si x est un teckel, alors x est petit]»; en somme, nous étendons les formules composées afin de pouvoir affirmer des quantifications existentielles («il existe ...») et des quantifications universelles («pour tout ...»). Les exemples que nous venons de donner font intervenir des propositions un peu particulières comme « x a une voiture américaine». Il s'agit ici de propositions comportant une *variable*. Ces propositions sont en fait l'application d'une *fonction* à x . Cette fonction, c'est celle qui associe « x a une voiture américaine» à x . Nous dénoterons quelquefois cette fonction par «_ a une voiture américaine» et nous dirons que c'est une *fonction propositionnelle*, car c'est une fonction dont la *valeur* est une *proposition*. Ou encore un *prédicat*.

Les quantifications existentielles et universelles vont donc de pair avec l'emploi de fonctions propositionnelles.

Le calcul des prédicats est cependant limité dans les formules existentielles et universelles. Ainsi, on s'interdit des formules comme «il existe une affirmation de x telle que ...». En fait, on ne s'autorise à quantifier que des «individus». C'est pour cela que l'on dit que le calcul des prédicats est une logique du premier ordre.

Nous allons présenter le calcul des prédicats qui est une extension de la logique propositionnelle incluant les quantifications existentielles et universelles. Il existe d'autres extensions de la logique propositionnelle, intermédiaires entre celle-ci et le calcul des prédicats, par exemple la logique équationnelle ou la logique de Horn, toutes deux des logiques du premier ordre. Nous présentons le calcul des prédicats car c'est la logique du premier ordre qui est présentée classiquement.

L'apport du calcul des prédicats est de donner une *représentation plus fine* du discours. Les énoncés sans structure du calcul des propositions ont ici une structure que l'on peut

manipuler et qui sert à construire de nouvelles formules.

4.2 Syntaxe

Nous construirons la langue formelle du calcul des prédicats en définissant d'abord un *lexique*, c'est-à-dire un ensemble de *symboles simples* rangés en catégories syntaxiques et ensuite une *grammaire* dont les règles permettent de former des *symboles composés* (termes ou formules). Nous verrons plus loin quel sens attribuer aux expressions de ce langage.

4.2.1 Lexique

Le lexique comprend :

1. un ensemble infini dénombrable de *symboles de prédicat* à 1-place $\{P_1(-), P_2(-), \dots, P_k(-), \dots\}$, à 2-places $\{P_1(-,-), P_2(-,-), \dots, P_k(-,-), \dots\}$, \dots , à n -places, \dots . Les *variables propositionnelles* p, q, r, \dots , sont considérées comme des symboles de prédicat à 0 place.
2. un ensemble infini dénombrable de symboles de fonction ou *foncteurs* à 1-place $\{f_1(-), f_2(-), \dots, f_k(-), \dots\}$, à 2-places $\{f_1(-,-), f_2(-,-), \dots, f_k(-,-), \dots\}$, \dots , à n -places, \dots . Les foncteurs à 0-place sont des constantes d'individu $\{a_1, a_2, \dots, a_n, \dots\}$. Par exemple « $(-)^2$ » est un foncteur à une place, « $(-) \cdot (-)$ » est un foncteur à deux places et « 120 » est une constante d'individu.
3. un ensemble infini dénombrable de *variables* individuelles $\{u, v, w, x, y, z, u', v', w', x', y', z', u'', \dots\}$.
4. les connecteurs $\{\neg, \wedge, \vee, \supset\}$.
5. les quantificateurs $\{\forall, \exists\}$.

Exemple 3 Dans l'arithmétique de Peano (cf. section 1.3), les symboles de fonction sont : 0 (constante), s (foncteur à une place), et $+$, \cdot (foncteurs à deux places). Les symboles de relation sont $<$ et $=$. Les axiomes de ce système sont une bonne illustration des formules de ce langage.

4.2.2 Grammaire

Le calcul des prédicats avec symboles de fonctions comporte quatre constructions et distingue les formules et les termes.

– La formation des termes :

1. x_1, \dots, x_n, \dots et $f_1^0, \dots, f_n^0, \dots$ sont des termes (variables d'individus et foncteurs à 0 place).
2. Si t_1, \dots, t_n sont des termes, $f_i^n(t_1, \dots, t_n)$ est un terme (f_i^n est le i -ième foncteur à n places).

- La *prédication* qui consiste à insérer n termes dans les n places d'un prédicat à n places :
Si t_1, \dots, t_n sont des termes, alors p et $P_i^n(t_1, \dots, t_n)$ sont des formules (variables propositionnelles et prédicats à n places). En particulier, les symboles propositionnels p, q, r, \dots , sont des formules atomiques.
- L'introduction de *connecteurs* :
 1. Si A est une formule, $\neg A$ l'est aussi.
 2. Si A et B sont des formules, $(A \wedge B)$, $(A \vee B)$, $(A \supset B)$ le sont aussi.
- La *quantification* qui consiste à préfixer une formule par un quantificateur :
Si A est une formule, et si x_i est une variable d'individu, $(\forall x_i)A$ et $(\exists x_i)A$ sont des formules. On écrira quelquefois $\forall x_i A$ et $\exists x_i A$ lorsque cela ne prête pas à confusion.

4.3 Interprétation des formules

Les formules sont constituées de symboles, sans signification *a priori*. Il faut précisément définir le sens des symboles que l'on utilise, sans quoi une preuve n'aura aucune valeur. Considérons par exemple la formule $(\forall x)(x \geq 0 \supset ((\exists y)R(x,y)))$. Dans cette formule, R est un symbole de relation à deux places, mais si nous ne disons de cette relation que son nom, cela n'est pas suffisant — en général — pour savoir précisément ce que cette relation exprime. Dans certains cas, la signification peut être plus explicite que d'autres, par exemple si l'on écrit $x = \cos y$ ou bien $(x = y) \vee (x = 2 \cdot y)$. Cependant, pour lever l'ambiguïté, apparente ou non, on associe à tous les symboles de relation, et à tous les foncteurs, des relations ou des fonctions mathématiques parfaitement définies, à la fois en ce qui concerne leur « effet » qu'en ce qui concerne le « type » de leurs arguments. Si l'on souhaite par exemple exprimer par la formule donnée plus haut une affirmation concernant des réels, on choisira \mathbb{R} comme domaine d'interprétation. À R , l'*interprétation* associe alors une relation sur $\mathbb{R} \times \mathbb{R}$.

La nécessité de cette interprétation est évidente si l'on note qu'une formule est vraie pour certaines interprétations et fautive pour d'autres. $(\forall x)(x \geq 0 \supset ((\exists y)x = y^2))$ est vrai si x et y sont des éléments de \mathbb{R} , mais pas s'ils sont éléments de \mathbb{N} (2 et $=$ ayant les interprétations usuelles).

Cela étant dit, nous pouvons définir plus précisément ce qu'est une interprétation :

Définition 4 Une interprétation \mathcal{I} se compose d'un domaine, noté $\mathcal{D}_{\mathcal{I}}$, tel que

1. Pour chaque symbole de constante c ,

$$\mathcal{I}(c) = c_{\mathcal{I}} \in \mathcal{D}_{\mathcal{I}}$$

2. Pour chaque symbole de fonction f , d'arité n , ($n \geq 0$),

$$\mathcal{I}(f) : \mathcal{D}_{\mathcal{I}}^n \mapsto \mathcal{D}_{\mathcal{I}}$$

3. Pour chaque symbole de relation R , d'arité m ,

$$\mathcal{I}(R) \subset \mathcal{D}_{\mathcal{I}}^m$$

Exemple 4 Soit F la formule

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

Si l'on choisit l'interprétation \mathcal{I} donnée par :

- $\mathcal{D}_{\mathcal{I}} = \mathbb{N}$
- $\mathcal{I}(a) = 1$
- $\mathcal{I}(S)(n) = n + 1$
- $\mathcal{I}(P)(m,n) \Leftrightarrow m|n$ (« m divise n »)
- $\mathcal{I}(Q)(m,n) \Leftrightarrow m = n$
- $\mathcal{I}(R)(m,n) \Leftrightarrow m > n$

alors la formule F signifie « il existe un nombre premier supérieur à 2 ».

Une interprétation donne une sémantique à chaque formule qui n'a pas de variables libres. On introduit la notion d'une *valuation* qui permet de parler de la sémantique d'une formule avec variables libres :

Définition 5 Une valuation ν est une application de l'ensemble des variables dans $\mathcal{D}_{\mathcal{I}}$. Cette définition d'une valuation s'étend à une valuation ν^* des termes de la manière suivante :

$$\nu^*(c) = c_{\mathcal{I}} \tag{4.1}$$

$$\nu^*(x) = \nu(x) \tag{4.2}$$

$$\nu^*(f(t_1, \dots, t_n)) = \mathcal{I}(f)(\nu^*(t_1), \dots, \nu^*(t_n)) \tag{4.3}$$

Les définitions précédentes permettent maintenant de définir la « vérité » d'une formule quelconque du calcul des prédicats. La définition suivante est due à Alfred Tarski.

Définition 6 Soit \mathcal{I} une interprétation, ν une valuation et A une formule. On définit la relation $\models_{\mathcal{I}}^{\nu} A$, signifiant que A est vraie dans l'interprétation \mathcal{I} avec la valuation ν , par récurrence sur la longueur de A :

1. $\models_{\mathcal{I}}^{\nu} R(t_1, \dots, t_n)$ si et seulement si $(\nu(t_1), \dots, \nu(t_n)) \in \mathcal{I}(R)$
2. $\models_{\mathcal{I}}^{\nu} \neg A$ si et seulement si $\not\models_{\mathcal{I}}^{\nu} A$
3. $\models_{\mathcal{I}}^{\nu} A \vee B$ si et seulement si $\models_{\mathcal{I}}^{\nu} A$ ou $\models_{\mathcal{I}}^{\nu} B$
4. $\models_{\mathcal{I}}^{\nu} A \wedge B$ si et seulement si $\models_{\mathcal{I}}^{\nu} A$ et $\models_{\mathcal{I}}^{\nu} B$
5. $\models_{\mathcal{I}}^{\nu} A \supset B$ si et seulement si $\not\models_{\mathcal{I}}^{\nu} A$ ou $\models_{\mathcal{I}}^{\nu} B$
6. $\models_{\mathcal{I}}^{\nu} (\exists x)A$ si et seulement s'il existe ν' tel que $\models_{\mathcal{I}}^{\nu'} A$ et $y \neq x$ implique que $\nu'(y) = \nu(y)$
7. $\models_{\mathcal{I}}^{\nu} (\forall x)A$ si et seulement si $\models_{\mathcal{I}}^{\nu'} A$ pour tout ν' tel que $y \neq x$ implique $\nu'(y) = \nu(y)$.

- Définition 7**
1. On dit que A est vraie dans l'interprétation \mathcal{I} , et on écrit $\models_{\mathcal{I}} A$, si et seulement si pour toute valuation ν , $\models_{\mathcal{I}}^{\nu} A$.
 2. Soit Γ un ensemble de formules. Lorsque \mathcal{I} est une interprétation dans laquelle chaque formule dans Γ est vraie, on dit que \mathcal{I} est un modèle de Γ .
 3. Lorsque Γ un ensemble de formules et A une formule, nous disons que A est une conséquence logique de Γ si A est vrai dans chaque modèle de Γ . Dans ce cas nous écrivons $\Gamma \models A$.

Théorème 3 *La validité d'une formule du calcul des prédicats est indécidable en général. Ceci découle de l'existence de formules qui ne sont ni vraies, ni fausses dans l'ensemble des interprétations et valuations. Par exemple, ni $\forall x P(x)$, ni $\exists x \neg P(x)$ ne sont des formules valides, alors que l'une est la négation de l'autre. (on peut trouver un P tel que $\forall x P(x)$ soit faux et un autre P tel que $\exists x \neg P(x)$ soit faux)*

4.3.1 Limites de l'interprétation, nécessité d'un système de preuve

L'interprétation des formules est fastidieuse si le domaine $\mathcal{D}_{\mathcal{I}}$ est important et impossible s'il est infini.

4.4 Équivalences classiques

Les équivalences données dans le cadre du calcul propositionnel restent vraies. Nous en donnons d'autres ici, sans les prouver. Elles peuvent par exemple être prouvées en utilisant l'appareil du calcul des séquents (voir plus loin).

- Lois de conversion des quantificateurs

$$\exists \nu \varphi \equiv \neg \forall \nu \neg \varphi \quad (4.4)$$

$$\forall \nu \varphi \equiv \neg \exists \nu \neg \varphi \quad (4.5)$$

$$\neg \forall \nu \varphi \equiv \exists \nu \neg \varphi \quad (4.6)$$

$$\neg \exists \nu \varphi \equiv \forall \nu \neg \varphi \quad (4.7)$$

- Lois de distribution des quantificateurs

$$\forall \nu (\varphi \wedge \psi) \equiv (\forall \nu \varphi \wedge \forall \nu \psi) \quad (4.8)$$

$$\exists \nu (\varphi \vee \psi) \equiv (\exists \nu \varphi \vee \exists \nu \psi) \quad (4.9)$$

- Lois de permutation des quantificateurs de même sorte

$$\forall \nu \forall \mu \varphi \equiv \forall \mu \forall \nu \varphi \quad (4.10)$$

$$\exists \nu \exists \mu \varphi \equiv \exists \mu \exists \nu \varphi \quad (4.11)$$

- Lois de réalphabetisation (renommage) des variables

On peut toujours renommer une variable liée et la variable du quantificateur au sein d'une formule. Cependant, le nouveau nom doit pas être un nom déjà utilisé pour une variable libre ou liée de la formule. Par exemple, dans $\exists x(\forall xFxy \supset (Gx \vee q))$, on peut opérer deux renommages : $\exists u(\forall vFvy \supset (Gu \vee q))$.

- Lois de passage

Si ν ne figure pas à titre d'occurrence libre dans ψ , on a les lois suivantes, dont une instance figure immédiatement en-dessous de chaque formule :

$$\forall \nu (\varphi \wedge \psi) \equiv (\forall \nu \varphi) \wedge \psi \quad (4.12)$$

$$\forall x (P(x) \wedge q) \equiv (\forall x P(x)) \wedge q \quad (4.13)$$

$$\exists \nu (\varphi \wedge \psi) \equiv (\exists \nu \varphi) \wedge \psi \quad (4.14)$$

$$\exists x (P(x) \wedge q) \equiv (\exists x P(x)) \wedge q \quad (4.15)$$

$$\forall \nu (\varphi \vee \psi) \equiv (\forall \nu \varphi) \vee \psi \quad (4.16)$$

$$\forall x (P(x) \vee q) \equiv (\forall x P(x)) \vee q \quad (4.17)$$

$$\exists \nu (\varphi \vee \psi) \equiv (\exists \nu \varphi) \vee \psi \quad (4.18)$$

$$\exists x (P(x) \vee q) \equiv (\exists x P(x)) \vee q \quad (4.19)$$

$$\forall \nu (\varphi \supset \psi) \equiv (\exists \nu \varphi) \supset \psi \quad (4.20)$$

$$\forall x (P(x) \supset q) \equiv (\exists x P(x)) \supset q \quad (4.21)$$

$$\exists \nu (\varphi \supset \psi) \equiv (\forall \nu \varphi) \supset \psi \quad (4.22)$$

$$\exists x (P(x) \supset q) \equiv (\forall x P(x)) \supset q \quad (4.23)$$

$$\forall \nu (\psi \supset \varphi) \equiv \psi \supset (\forall \nu \varphi) \quad (4.24)$$

$$\forall x (q \supset P(x)) \equiv q \supset (\forall x P(x)) \quad (4.25)$$

$$\exists \nu (\psi \supset \varphi) \equiv \psi \supset (\exists \nu \varphi) \quad (4.26)$$

$$\exists x (q \supset P(x)) \equiv q \supset (\exists x P(x)) \quad (4.27)$$

4.5 Systèmes axiomatiques

Nous présentons ici un système axiomatique pour le calcul des prédicats. Il se compose de cinq schémas d'axiomes et de deux règles d'inférence.

Les trois premiers schémas d'axiomes sont ceux d'un système axiomatique du calcul des propositions :

$$\begin{array}{ll} \text{A1} & \varphi \supset (\psi \supset \varphi) \\ \text{A2} & (\varphi \supset (\psi \supset \xi)) \supset ((\varphi \supset \psi) \supset (\varphi \supset \xi)) \\ \text{A3} & (\neg \psi \supset \neg \varphi) \supset ((\neg \psi \supset \varphi) \supset \psi) \end{array}$$

À cela, on ajoute les définitions de \wedge , \vee et \equiv , ainsi que la règle du *modus ponens*.

Deux nouveaux axiomes sont propres au calcul des prédicats :

A4 $\forall \nu (\psi \supset \varphi) \supset (\psi \supset \forall \nu \varphi)$ où ν ne figure pas à titre d'occurrence libre dans ψ .

A5 $\forall \nu \varphi \supset \varphi_{\tau/\nu}$ où ν est libre pour tout τ dans φ . ($\varphi_{\tau/\nu}$ représente φ où l'on remplace les occurrences libres de ν par τ)

Les deux règles d'inférence sont le *modus ponens* étendu aux formules ouvertes (contenant des variables libres), et d'autre part la règle de généralisation universelle, notée GU:

$$\frac{\varphi}{\forall \nu \varphi}$$

Cette dernière règle signifie que si l'on déduit φ à l'aide des axiomes et des règles d'inférence, alors on peut déduire $\forall \nu \varphi$. Si φ contient la variable libre ν , cela signifie que ν est variable libre dans l'un des axiomes, ce qui signifie que l'axiome est vrai pour un ν quelconque, donc pour tout ν , ce qui justifie la règle.

Cette règle ne doit pas être confondue avec la déduction de $\forall \nu \varphi$ à partir de φ pris comme hypothèse. En effet, si ν figure comme variable libre dans une hypothèse, il ne s'agit pas d'un ν quelconque, mais d'un ν bien précis, et on ne peut alors pas quantifier universellement. Cette nuance est très importante. On trouvera des explications complémentaires à ce sujet dans l'ouvrage de Gochet & Gribomont cité en bibliographie.

4.6 Calcul des séquents

4.6.1 Présentation

Le système de preuve reprend le calcul des séquents de la logique propositionnelle et le complète avec les règles suivantes :

- Règle d'« introduction » de \forall dans l'antécédent, notée G_{\forall}

$$\frac{\Gamma, \varphi(r), (\forall x) \varphi(x) \rightarrow \Delta}{\Gamma, (\forall x) \varphi(x) \rightarrow \Delta} \quad (G_{\forall})$$

Intuition Si de Γ , $\varphi(r)$ et $(\forall x) \varphi(x)$ on peut déduire Δ , alors puisque de $(\forall x) \varphi(x)$ on peut déduire $\varphi(r)$, on peut aussi déduire la conclusion. Cette règle aussi est réversible.

- Règle d'introduction de \forall dans le succédent, notée D_{\forall}

$$\frac{\Gamma \rightarrow \varphi(r), \Delta}{\Gamma \rightarrow (\forall x) \varphi(x), \Delta} \quad (D_{\forall}) \quad (r \text{ ne doit être libre ni dans } \Gamma \text{ ni dans } \Delta)$$

Intuition Si à partir de Γ ne dépendant pas de r , on peut déduire $\varphi(r)$, il est clair que l'on peut déduire $\varphi(r)$ pour n'importe quel r . Dire que de Γ on peut déduire $(\forall x) \varphi(x)$ est la formalisation de cette idée. Inversement, on peut particulariser x .

- Règle d'introduction de \exists dans l'antécédent, notée G_{\exists}

$$\frac{\Gamma, \varphi(r) \rightarrow \Delta}{\Gamma, (\exists x) \varphi(x) \rightarrow \Delta} \quad (G_{\exists}) \quad (r \text{ ne doit être libre ni dans } \Gamma \text{ ni dans } \Delta)$$

Intuition De Γ et $(\exists x) \varphi(x)$ on peut déduire Γ et $\varphi(r)$, en supposant que r n'est pas libre dans Γ , on peut donc en déduire Δ . La règle réciproque est aussi vraie.

– Règle d'« introduction » de \exists dans le succédent, notée D_{\exists}

$$\frac{\Gamma \rightarrow \varphi(r), (\exists x) \varphi(x), \Delta}{\Gamma \rightarrow (\exists x) \varphi(x), \Delta} \quad (D_{\exists})$$

Intuition *Si de Γ , on peut déduire $\varphi(r)$, on peut naturellement aussi en déduire $(\exists x) \varphi(x)$ ce qui justifie intuitivement la règle. Réciproquement, il s'agit d'un cas particulier de la règle d'atténuation.*

Note : On peut être surpris de voir que ni \forall ni \exists ne sont réellement introduits dans la première et la dernière de ces quatre règles. En fait, deux choses doivent être réalisées. La première, c'est que si l'on part de la conclusion pour aller vers les prémisses, on particularise un x , mais comme on ne donne aucune contrainte¹ sur ce x particulier, on doit conserver la formule quantifiée initiale. La seconde, c'est que ces formules quantifiées finiront par disparaître dans les axiomes qui ne manqueront pas d'apparaître au fur et à mesure que l'on avance la preuve.

1. Dans d'autres variantes du calcul des séquents, ces règles ne reprennent pas les formules quantifiées mais introduisent à la place des contraintes sur r .

4.6.2 Récapitulation des règles

$$\boxed{\begin{array}{cc} \frac{\Gamma_1, \varphi, \psi, \Gamma_2 \rightarrow \Delta}{\Gamma_1, \psi, \varphi, \Gamma_2 \rightarrow \Delta} (G_P) & \frac{\Gamma \rightarrow \Delta_1, \varphi, \psi, \Delta_2}{\Gamma \rightarrow \Delta_1, \psi, \varphi, \Delta_2} (D_P) \\ \frac{\Gamma, \varphi, \varphi \rightarrow \Delta}{\Gamma, \varphi \rightarrow \Delta} (G_C) & \frac{\Gamma \rightarrow \Delta, \varphi, \varphi}{\Gamma \rightarrow \Delta, \varphi} (D_C) \\ \frac{\Gamma \rightarrow \Delta}{\Gamma, \varphi \rightarrow \Delta} (G_A) & \frac{\Gamma \rightarrow \Delta}{\Gamma \rightarrow \Delta, \varphi} (D_A) \end{array}}$$

$$\boxed{\begin{array}{cc} \frac{\Gamma \rightarrow \varphi, \Delta}{\Gamma, \neg \varphi \rightarrow \Delta} (G_{\neg}) & \frac{\Gamma, \varphi \rightarrow \Delta}{\Gamma \rightarrow \neg \varphi, \Delta} (D_{\neg}) \\ \frac{\Gamma, \varphi, \psi \rightarrow \Delta}{\Gamma, \varphi \wedge \psi \rightarrow \Delta} (G_{\wedge}) & \frac{\Gamma \rightarrow \varphi, \psi, \Delta}{\Gamma \rightarrow \varphi \vee \psi, \Delta} (D_{\vee}) \\ \frac{\Gamma \rightarrow \varphi, \Delta \quad \Gamma \rightarrow \psi, \Delta}{\Gamma \rightarrow \varphi \wedge \psi, \Delta} (D_{\wedge}) & \frac{\Gamma, \varphi \rightarrow \Delta \quad \Gamma, \psi \rightarrow \Delta}{\Gamma, \varphi \vee \psi \rightarrow \Delta} (G_{\vee}) \\ \frac{\Gamma_1, \Gamma_2 \rightarrow \Delta_1, \varphi, \Delta_2 \quad \Gamma_1, \psi, \Gamma_2 \rightarrow \Delta_1, \Delta_2}{\Gamma_1, \varphi \supset \psi, \Gamma_2 \rightarrow \Delta_1, \Delta_2} (G_{\supset}) & \frac{\Gamma, \varphi \rightarrow \psi, \Delta}{\Gamma \rightarrow \varphi \supset \psi, \Delta} (D_{\supset}) \\ \frac{\Gamma, \varphi(r), (\forall x) \varphi(x) \rightarrow \Delta}{\Gamma, (\forall x) \varphi(x) \rightarrow \Delta} (G_{\forall}) & \frac{\Gamma \rightarrow \varphi(r), \Delta}{\Gamma \rightarrow (\forall x) \varphi(x), \Delta} (D_{\forall})(*) \\ \frac{\Gamma, \varphi(r) \rightarrow \Delta}{\Gamma, (\exists x) \varphi(x) \rightarrow \Delta} (G_{\exists})(*) & \frac{\Gamma \rightarrow \varphi(r), (\exists x) \varphi(x), \Delta}{\Gamma \rightarrow (\exists x) \varphi(x), \Delta} (D_{\exists}) \end{array}}$$

(*) r non libre dans Γ et Δ .

$$\boxed{\frac{\Gamma_1 \rightarrow \varphi, \Delta_1 \quad \Gamma_2, \varphi \rightarrow \Delta_2}{\Gamma_1, \Gamma_2 \rightarrow \Delta_1, \Delta_2} (C)}$$

4.6.3 Théorèmes de correction et de complétude

Nous ne donnons pas ici la preuve de ces théorèmes. La démonstration du théorème de complétude peut être trouvée par exemple dans *Introduction à la logique* de François Rivenc.

4.6.4 Exemple de preuve

Exemple 5 Une preuve du séquent $\rightarrow \exists y(Fy \wedge \forall x(Fx \supset Gxy)) \supset \exists x(Fx \wedge Gxx)$ est :

$$\frac{\frac{\frac{\frac{\frac{\forall x(Fx \supset Gxa), Fa \rightarrow Gaa, Fa, \exists x(Fx \wedge Gxx) \quad \forall x(Fx \supset Gxa), Fa, Gaa \rightarrow Gaa, \exists x(Fx \wedge Gxx)}{\forall x(Fx \supset Gxa), Fa, Fa \supset Gaa \rightarrow Gaa, \exists x(Fx \wedge Gxx)} (G_{\supset})}{Fa, \forall x(Fx \supset Gxa) \rightarrow Fa, \exists x(Fx \wedge Gxx)} \quad Fa, \forall x(Fx \supset Gxa) \rightarrow Gaa, \exists x(Fx \wedge Gxx)} (D_{\wedge})}{Fa, \forall x(Fx \supset Gxa) \rightarrow Fa \wedge Gaa, \exists x(Fx \wedge Gxx)} (G_{\wedge})}{Fa \wedge \forall x(Fx \supset Gxa) \rightarrow Fa \wedge Gaa, \exists x(Fx \wedge Gxx)} (D_{\exists})}{Fa \wedge \forall x(Fx \supset Gxa) \rightarrow \exists x(Fx \wedge Gxx)} (G_{\exists})}{\exists y(Fy \wedge \forall x(Fx \supset Gxy)) \rightarrow \exists x(Fx \wedge Gxx)} (D_{\supset})}{\rightarrow \exists y(Fy \wedge \forall x(Fx \supset Gxy)) \supset \exists x(Fx \wedge Gxx)}$$

(pour éviter la surcharge de parenthèses, nous avons écrit Fxx au lieu de $F(x)$ et Gxy au lieu de $G(x, y)$)

Glossaire et définitions

arité L'arité d'une fonction est le nombre de paramètres qu'elle attend. Exemple : l'arité de la fonction f telle que $f(x,y,z,n) = x^n + y^n - z^n$ est 4. L'arité d'une constante est zéro.

axiome C'est une expression qui sert de point de départ. On la prend vraie par hypothèse et on ne cherche pas à la démontrer. Si on a plusieurs axiomes, ceux-ci ne doivent pas être contradictoires entre eux. Cependant, un système de plusieurs axiomes peut être redondant, certains axiomes pouvant être déduits à partir d'autres axiomes. On essaie en général d'avoir le plus petit nombre d'axiomes, ceci ayant en particulier l'avantage de réduire les risques de contradictions. Il est important de noter qu'un axiome ne contient pas de variables de formules. C'est ce qui le différencie des *schémas d'axiome* (voir ce mot). Par exemple, l'un des axiomes de l'arithmétique de Peano est $(\forall x)(x + 0 = x)$.

formel Un *système formel* est un ensemble d'axiomes et de règles d'inférence, définissant un *langage formel*.

libre cf. **lié**

lié Une variable liée est une variable qui est sous l'*effet* d'un quantificateur. Une variable libre est une variable qui n'est pas liée. Exemple : dans $\forall x \varphi(x) \vee y$, x est une variable liée (par le quantificateur \forall). y n'y est pas liée. Dans certaines expressions une variable peut être liée en certaines occurrences et libre en d'autres. Exemple : dans $(\forall x (\psi(x) \vee y)) \wedge (x = 5)$, la première occurrence de x (dans $\psi(x)$) est liée (par \forall), alors que la seconde (dans $x = 5$) ne l'est pas. Ces deux x n'ont rien à voir entre eux.

occurrence Dans une formule, une variable peut avoir plusieurs *occurrences*. Il s'agit des apparitions de cette variable, sans compter celles dans les quantificateurs. Dans $\forall x,y,z (x = y) \vee (x = z)$ il y a deux occurrences de x , une de y et une de z .

règle Une règle est un moyen de transformer une expression en une autre. Les règles sont notées $\frac{E_1}{E_2}$ et signifient que l'expression E_1 est transformée en l'expression E_2 . Il s'agit de règles de déduction ou règles d'inférence. Le nom de la règle est souvent indiqué à droite ou à gauche de la barre.

Exemple : si les expressions sont des égalités entre entiers naturels, on peut avoir la règle $\frac{a=b}{a+1=b+1}$.

règles dérivées À partir de certaines règles primitives, on peut obtenir des règles plus complexes. Si par exemple $\frac{a=b}{a+1=b+1}$ est une règle d'inférence, on peut en déduire une nouvelle règle *dérivée* : $\frac{a=b}{a+2=b+2}$

schémas d'axiomes Un schéma d'axiome est une formule décrivant un *ensemble* d'axiomes. Ainsi, si l'on prend l'axiomatique du calcul des prédicats (cf. §4.5), on admet comme point de départ possible l'expression : $\varphi \supset (\psi \supset \varphi)$. Il ne s'agit pas là d'un axiome car

φ et ψ sont des variables et on n'obtient un axiome que lorsque l'on remplace φ et ψ par des formules particulières. Ainsi, si le substrat de la logique (le langage des termes) est l'arithmétique, un des axiomes sera : $(x = 5) \supset ((x = 6) \supset (x = 5))$.

substitution Opération par laquelle dans une formule donnée une variable est remplacée par une expression, et ce pour toutes ses occurrences. Par exemple, la substitution de x par $x + y$ dans $x^n + y^n = z^n$ donne $(x + y)^n + y^n = z^n$.

système formel cf. formel.

Bibliographie

Les ouvrages sur la logique sont légion. Nous nous contenterons ici de citer des ouvrages en français. Pour des références plus complètes, nous suggérons de consulter la bibliographie de Gochet et Gribomont. Nous avons très brièvement commenté ces ouvrages.

- François Rivenc : *Introduction à la logique*, Petite bibliothèque Payot, 1989. Cet ouvrage s'adresse plutôt aux étudiants de philosophie.
- René Lalement : *Logique, réduction, résolution*, Masson, 1990. Seules quelques parties concernent le traitement classique de la logique. Sans doute un peu difficile d'abord.
- Paul Gochet, Pascal Gribomont : *Logique*, 2 tomes, Hermes, 1991 (tome 1). Très complet, mais aussi très touffu.
- Jean Largeault : *La logique*, Que sais-je?, n° 225, PUF, 1993. Bref, mais utile pour se faire une idée et s'ouvrir des horizons.
- Willard V. O. Quine : *Méthodes de logique*, PUF, 1973. Ouvrage très didactique, recommandé pour une première introduction à la logique.