Logique mathématique : une introduction au calcul des prédicats du premier ordre.

Paul Rozière
Paris 7 – M63010

19 janvier 2011
(1047 –version provisoire)

1 Une première appproche très informelle.

1.1 Les objets, les énoncés, les preuves.

En mathématiques on traite d'*objets* : les nombres, les points, les droites, les ensembles, les fonctions etc. On énonce des propriétés de ces objets de façon organisée. Certains *énoncés* initiaux, les *axiomes* sont admis, ils décrivent les propriétés de base des objet de la théorie. Par exemple la récurrence qui est une propriété fondamentale des entiers, se décrit par des axiomes. On en déduit d'autres énoncés, les *théorèmes* en utilisant certaines règles de raisonnement, souvent implicites, mais que toute personne pratiquant les mathématiques admet. Les théorèmes décrivent des propriétés de moins en moins évidentes des objets considérés. Une *démonstration* d'un énoncé est une construction qui permet de convaincre que l'on a bien utilisé pour déduire l'énoncé les règles de raisonnement communément admises. On dit alors que l'énoncé est *conséquence* des axiomes utilisés pour la démonstration.

Cet exposé est trop court pour présenter la notion formelle de démonstration, mais on peut formaliser celle-ci, et de plus on considère que d'une certaine façon cette formalisation est achevée, au sens où l'on connait toutes les règles de raisonnement de nature logique : celles qui ont une portée tout à fait générale et ne sont pas particulières à un domaine mathématique donné. Par exemple le raisonnement par Modus Ponens, dit que de $A \Rightarrow B$ et de A on déduit B : c'est une règle logique primitive. Le raisonnement par récurrence fait référence aux entiers est n'est pas une règle logique. Si la formalisation est achevée pour la logique pure[1], ce n'est pas le cas pour les entiers : d'une certaine façon elle ne peut l'être de façon satisfaisante[2].

1.1.1 Structures et théories.

Une structure est un ensemble muni de plusieurs fonctions ou opérations, comme l'addition, et de prédicats ou relations, comme l'ordre. C'est celle qui est commune en algèbre, elle généralise les groupes, les anneaux, les corps, les ensembles ordonnés etc., mais aussi les entiers ou les réels munis de leurs opérations habituelles (addition, multiplication etc.).

Un énoncé mathématique est écrit dans un certain langage, et on peut dire qu'il est vrai ou faux dans une structure qui interprète tous les éléments du langage. Par exemple l'énoncé $1+1=0$ est faux pour $(\mathbb{N},+)$, et vrai pour $(\mathbb{Z}/2\mathbb{Z},+)$, de même que l'énoncé « il existe un x différent de 0 tel que $x+x=0$ ».

Un exemple très simple de langage et celui de la théorie des ordres, qui n'utilise qu'un seul prédicat, $\leqslant$, et les symboles logiques, que l'on retrouve dans tous les langages, variables, implication (« si ..., alors ...»), quantificateur universel (pour tout x ...) etc. On peut dire par exemple que la structure $(\mathbb{N},\leqslant)$, possède un plus petit élément, c'est à dire que l'énoncé « il existe x tel que pour tout y, $x \leqslant y$ » est vrai dans les entiers naturels. Ce même énoncé est faux dans $(\mathbb{Z},\leqslant)$.

Une théorie est décrite par un ensemble d'énoncés, ses axiomes, écrits dans un certain langage. Par exemple la théorie des ordres totaux est constituée des quatre axiomes de réflexivité, transitivité, anti-symétrie et totalité. D'après ce qui précède l'énoncé « il existe x tel que pour tout y, $x \leqslant y$ » n'est pas déterminé dans la théorie des ordres totaux : il peut être vrai, par exemple dans la structure $(\mathbb{N},\leqslant)$ ou faux, par exemple dans la structure $(\mathbb{Z},\leqslant)$, ces deux structures vérifiant les axiomes d'ordre total.

Cette remarque évidente a pour but de faire le rapport entre démonstration et validité dans une structure. Comme l'énoncé « il existe x tel que pour tout y, $x \leqslant y$ » est faux dans $(\mathbb{Z},\leqslant)$, on ne peut pas le démontrer dans la théorie des ordres totaux. Comme l'énoncé « il existe x tel que pour tout y, $x \leqslant y$ » est vrai dans $(\mathbb{N},\leqslant)$, on ne peut pas non plus démontrer sa négation dans la théorie des ordres totaux. On s'appuie pour cela sur le fait très intuitif que la validité dans une structure est conservée par une déduction correcte, même si on n'a pas examiné les règles du raisonnement.

La notion de conséquence logique, peut donc aussi se décrire par la validité dans les structures : un énoncé est conséquence d'un système d'axiomes s'il est vrai dans toute structure qui valide ces axiomes.

Le théorème de complétude de Gödel, qui est hors de portée de cet exposé très introductif, énonce que les deux façons d'aborder la conséquence logique, celle que nous n'avons même pas essayé de décrire, par la donnée de règles de raisonnement, et celle par la validité dans les structures, coïncident. Plus précisément, le théorème de complétude dit que si un énoncé A n'est pas conséquence formelle des axiomes d'une théorie, on pourra toujours construire une structure dans laquelle tous les axiomes de la théorie sont vrais, mais dans laquelle l'énoncé A est faux.

[1] c'est une version très informelle du théorème de complétude de Gödel

[2] c'est une version très informelle du théorème d'incomplétude de Gödel

1.1.2 Premier et second ordre.

Pour pouvoir définir de façon précise la notion de validité dans une structure, il est nécessaire de décrire de façon précise les énoncés que l'on va interpréter, le langage dans lequel on va exprimer ceux-ci. Ce seront les langages du calcul des prédicats du premier ordre. Cela signifie essentiellement que les seules variables autorisées sont celles qui sont astreintes aus éléments de l'ensemble de base d'une structure, mais pas aux sous-ensembles de cet ensemble de base, non plus aux fonctions définies sur celui-ci, etc. En fait le plus important et que l'on ne peut pas écrire de quantificateurs autres que portant sur ces variables.

Par exemple l'axiome de bon ordre n'est pas un énoncé du premier ordre. En effet, si $(E, \leqslant)$ vérifie déjà les axiomes d'ordre, on dit qu'il est bien ordonné si :

$$\forall A \subset E \ (A \neq \emptyset \Rightarrow \exists x \in A \ \forall y \in A \ x \leqslant y)$$

La quantification « $\forall A \subset E \ldots$ » n'est pas du premier ordre. On dit que c'est un *énoncé du second ordre* de la théorie des ensembles ordonnés.

Cela signifie bien-sûr pas que l'on refuse en général d'écrire un tel énoncé, mais que celui-ci est considéré comme un énoncé d'une théorie plus large, par exemple la théorie des ordres plus la théorie des ensembles. On n'est pas obligé en fait de faire appel forcément à toute la théorie des ensembles de Zermelo, mais il y aura toujours une forme de l'axiome (ou schéma d'axiomes) de compréhension. Les démonstrations dans ces théories ne font donc plus seulement appel aux axiomes de la théorie des ordres et aux règles de raisonnement purement logiques. On peut toujours se ramener à une théorie du premier ordre (la théories des ensembles de Zermelo ou de Zermelo-Fraenkel est elle-même une théorie du premier ordre) mais, d'un point de vue logique, il faut alors étudier la théorie avec les axiomes ensemblistes supplémentaires utiles.

Parmi les théories qui ne sont pas du premier ordre, nous avons vu les axiomes de Peano, qui utilisent la notion d'ensemble, la théorie étudiée est donc une théorie « modulo » la théorie des ensembles. Il existe une version premier ordre (plus faible) que l'on appelle l'arithmétique de Peano et qui permet de développer l'arithmétique élémentaire, mais pas l'analyse réelle.

L'axiomatisation des réels n'est pas non plus une théorie du premier ordre. L'axiome d'Archimède fait appel aux entiers :

$$\forall x, y \in \mathbb{R} (x > 0 \Rightarrow \exists n \in \mathbb{N} \ n \cdot x > y)$$

L'axiome de la borne supérieure demande de quantifier sur les sous-ensembles de A (si on utilise la complétude, il faut quantifier sur les suites).

Par contre la théorie axiomatique des groupes, celle des anneaux, celle des corps sont des théories du premier ordre (mais on utilise, par exemple en théorie des groupes, la notion de groupe simple, qui n'est pas du premier ordre).

2 Langages du premier ordre.

On commence par définir la *syntaxe* d'un langage du premier ordre, c'est à dire les constructions correctes pour les *termes* qui désignent des objets mathématiques, et les *formules* qui désignent des propriétés de ces objets ou des assertions à propos de ceux-ci.

Parmi les signes utilisés on distingue entre ceux qui apparaissent dans tous les langages, variable, quantificateurs (Pour tout x ...), connecteur (implication), et ceux qui sont spécifiques au domaine étudié, que l'on appelle signature du langage.

2.1 Signature.

Une *signature* est la donnée :
- d'une suite de symboles de constantes ;
- d'une suite de symboles de fonctions chacun muni d'un entier appelé "arité" du symbole, qui indique le nombre de places de la fonction ;
- d'une suite de symboles de prédicats, chacun également muni d'une arité.

Chacune de ces suites peut-être vide.

La théorie des ordres a pour signature $(\leqslant)$ (un seul symbole de prédicat d'arité 2), l'arithmétique de Peano a pour signature $(0, s, +, \cdot, \leqslant)$ où 0 est un symbole de constante, s un symbole de fonction d'arité 1, $+$ et $\cdot$ des symboles de fonction d'arité 2, et $\leqslant$ un symbole de prédicat d'arité 2.

Sauf précision l'égalité fait toujours partie du langage (on précise parfois langage égalitaire du premier ordre), et le signe de l'égalité n'apparaît donc pas dans la signature.

On va définir dans la suite le langage égalitaire du premier ordre de signature $\mathcal{S}$, on dira plus rapidement le langage de signature $\mathcal{S}$, voire le langage $\mathcal{S}$.

2.2 Les termes.

On définit tout d'abord les *termes* d'un langage donné, qui désignent des objets. Ils sont écrits avec des lettres, x, (, s, $+$, etc. Comme tous les éléments syntaxiques que nous allons introduire, les termes peuvent être vus comme des suites finies de lettres. Voyons tout d'abord un exemple.

2.2.1 Termes de l'arithmétique.

Le langage est celui de l'arithmétique de Peano, de signature $\mathcal{P} = (0, s, +, \cdot, \leqslant)$ avec les arités usuelles. On suppose donnée un ensemble dénombrable de variables $\mathcal{V}$. L'ensemble $\mathcal{T}_\mathcal{P}$ des termes de signature $\mathcal{P}$ est par définition le plus petit ensemble (de mots) vérifiant :

variable toute variable x de $\mathcal{V}$ appartient à $\mathcal{T}_\mathcal{P}$.

constante 0 appartient à $\mathcal{T}_\mathcal{P}$.

successeur si t appartient à $\mathcal{T}_\mathcal{P}$, $s\,t$ appartient à $\mathcal{T}_\mathcal{P}$.

addition et multiplication si t et t' appartiennent à $\mathcal{T}_\mathcal{P}$, $(t + t')$ et $(t \cdot t')$ appartiennent à $\mathcal{T}_\mathcal{P}$.

Une telle définition est dite inductive. Elle est analogue à celle des entiers en théorie des ensembles, et on a de la même façon une propriété analogue à la récurrence, que dans ce contexte on appelle souvent *propriété d'induction*, qui énonce que tout ensemble qui vérifie les quatre clauses énoncées ci-dessus, contient $\mathcal{T}_\mathcal{P}$. On peut associer à chaque terme de $\mathcal{T}_\mathcal{P}$ un *arbre de dérivation* ; voici quelques termes et les arbres de dérivations associés :

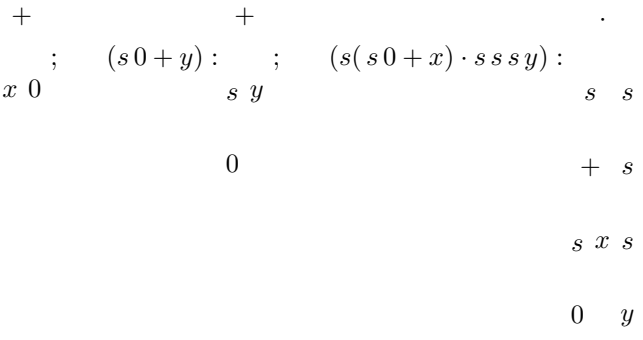
$(x + 0)$:

;

$(s\,0 + y)$:

;

$(s(s\,0 + x) \cdot s\,s\,s\,y)$:



On aura besoin également d'un analogue de la définition par récurrence, on dit plutôt *définition par induction*, qui suppose que les clauses ci-dessus ont été énoncées de façon à satisfaire la propriété dite de *lecture unique*, qui énonce essentiellement qu'un terme de $\mathcal{T}_\mathcal{P}$ n'a qu'une seul arbre de dérivation, il n'y a qu'une façon d'obtenir ce terme.

Il suffit d'avoir mis suffisamment de parenthèses et autres séparateurs (virgule, espacements etc.) pour que cette propriété soit vérifiée, et nous admettrons que c'est bien le cas de la définition ci-dessus. La propriété de lecture unique énonce que le terme écrit comme mot (linéairement) est bien une représentation fidèle de son arbre de dérivation, qui serait la « vraie » notion de terme, mais évidemment les arbres sont peu commodes à écrire ! On se permettra d'ailleurs les abréviations habituelles. Dans la syntaxe telle qu'elle a été définie, les évéentuelles parenthèses les plus à l'extérieur d'un terme ne sont pas nécessaires : on écrit $s\,0 + x$ plutôt que $(s\ 0 + x)$. Les règles de priorité entre opérations permettent d'éviter certaines parenthèses : $x + y \cdot z$, plutôt que $x + (y \cdot z)$.

Bien entendu le signe $\leqslant$, qui est un symbole de prédicat, n'apparaît pas dans les termes.

2.2.2 Cas général.

On va définir les termes d'un langage de signature $\mathcal{S}$ en considérant que tous les symboles de fonction sont en notation préfixe (ce que nous n'avons pas fait dans le cas de $+$ et $\cdot$ ci-dessus).

L'ensemble $\mathcal{T}_\mathcal{S}$ des termes de signature $\mathcal{S}$ est défini inductivement par :

variable toute variable x de $\mathcal{V}$ appartient à $\mathcal{T}_\mathcal{S}$;

constante toute constante appartient à $\mathcal{T}_\mathcal{S}$;

fonction pour chaque symbole de fonction n-aire f de $\mathcal{S}$, si $t_1, \ldots, t_n$ appartiennent à $\mathcal{T}_\mathcal{S}$, alors $f\,t_1\,\ldots\,t_n$ appartient à $\mathcal{T}_\mathcal{S}$.

Le choix des notations (parenthèses, espaces ou "," pour séparer les arguments) n'a pas grande importance. Le principal est que les notations ci-dessus assurent la propriété de lecture unique : un terme de $\mathcal{T}_\mathcal{S}$ a un seul arbre de dérivation. On pourra donc utiliser des définitions par induction sur les termes.

Certains langages peuvent ne pas avoir de symboles de fonction ni de constante, comme le langage des ordre ($<$) ou celui de la théorie des ensembles ($\in$), qui ont donc pour seuls termes les variables.

2.2.3 termes clos.

On peut maintenant définir par induction, par exemple, la notion de *terme clos* de $\mathcal{T}_\mathcal{S}$, un terme sans variables :

variable Une variable x de $\mathcal{V}$ n'est jamais un terme clos.

constante Toute constante est un terme clos.

fonction Pour chaque symbole de fonction n-aire f de $\mathcal{S}$, $f\,t_1\,\ldots\,t_n$ est un terme clos si $t_1, \ldots, t_n$ sont des termes clos.

Remarquons qu'un langage, comme celui des ordres, sans constantes, n'a pas de termes clos.

2.3 Formules atomiques.

2.3.1 L'arithmétique.

On reprend le langage de l'arithmétique, de signature $\mathcal{P}$. Les *formules atomiques* sont formées à partir de l'égalité et des symboles de prédicat du langage, ici $\leqslant$.

égalité Si t_1 et t_2 sont des termes de $\mathcal{P}$, $t_1 = t_2$ est une formule atomique de $\mathcal{P}$.

ordre Si t_1 et t_2 sont des termes de $\mathcal{P}$, $t_1 \leqslant t_2$ est une formule atomique de $\mathcal{P}$.

Par exemple $s\,0 + x = y$, $(s\,0 + x) \cdot (s\,0 + x) = 0$, $0 \leqslant 0$ sont des formules atomiques de l'arithmétique. Remarquez que les formules atomiques de l'arithmétique de Peano sont essentiellement les égalités polynomiales et les inégalités polynomiales sur les entiers.

Une formule atomique close est définie comme une formule qui n'est construite qu'avec des termes clos, ainsi parmi les formules ci-dessus la seule formule close est $0 \leqslant 0$. Qaund on interprètera les formules, une formule close est vraie ou fausse. Cependant la formule atomique $x = x$ n'est pas une formule close, même si intuitivement elle ne dépend pas de x.

2.3.2 Cas général.

On considère que le langage est égalitaire. Exceptionnellement il nous arrivera de considérer des langages non égalitaires, auquel cas on supprime bien sûr la première des clauses de la définition.

égalité Si t_1 et t_2 sont des termes de $\mathcal{S}$, $t_1 = t_2$ est une formule atomique de $\mathcal{S}$.

prédicat Pour chaque symbole de prédicat n-aire P de $\mathcal{S}$, si $t_1, \ldots, t_n$ sont des termes de $\mathcal{S}$, alors $P\,t_1 \ldots t_n$ est une formule atomique de $\mathcal{S}$.

On définit les formules atomiques closes (aucune variable n'apparaît dans la formule).

Pour la plupart des langages étudiés, les symboles de prédicats du langage seront des symboles de prédicat binaire (l'ordre, l'appartenance) que l'on notera comme d'habitude de façon infixe (comme $\leqslant$ pour l'arithmétique ci-dessus).

2.4 Formules.

2.4.1 Définition.

On construit de nouvelles formules à l'aide de *connecteurs* et de *quantificateurs*. Prenons comme connecteurs $(\bot, \neg, \to, \vee, \wedge)$. La constante propositionnelle $\bot$ (pour l'absurde) est considérée comme un connecteur à 0 arguments. Le choix est assez arbitraire, on pourrait ajouter $\top$ (pour le vrai) et $\leftrightarrow$ pour l'équivalence, on verra que ça n'a pas grande importance. Les quantificateurs sont $\forall$ et $\exists$.

L'ensemble des *formules* du langage $\mathcal{S}$ est défini inductivement par les clauses suivantes

formules atomiques les formules atomiques de $\mathcal{S}$ sont des formules ;

absurde $\bot$ est une formule ;

négation Si F est une formule $\neg F$ est une formule ;

implication Si F et G sont des formules $(F \to G)$ est une formule ;

conjonction Si F et G sont des formules $(F \wedge G)$ est une formule ;

disjonction Si F et G sont des formules $(F \vee G)$ est une formule ;

quantification universelle Si F est une formule et x une variable, alors $\forall x\ F$ est une formule ;

quantification existentielle Si F est une formule et x une variable, alors $\exists x\ F$ est une formule.

Quand une formule n'utilise pas de quantificateurs on dit que c'est une *formule propositionnelle*.

Là encore on admettra le théorème de lecture unique, et donc on utilisera les définitions par induction.

Voici deux formules du langage de l'arithmétique $\mathcal{P}$:

$$\forall x\ (x \leqslant s\,0 \to (x = 0 \vee x = s\,0)), \quad \exists y(x = 2 \cdot y \vee x = 2 \cdot y + s\,0)$$

Dans une structure donnée, la vérité de la première de ces deux formules est déterminée (elle est vraie dans $\mathbb{N}$ et fausse dans $\mathbb{Z}$ par exemple), bien que la variable x apparaisse dans la formule, on dit que la variable x est liée dans cette formule, et que la formule est close. La vérité de la seconde de ces formules dans une structure donnée ne peut se déterminer sans dire ce que vaut x : la variable x est dite libre dans cette formule (la variable y est elle liée). En fait ces notions se définissent de façon purement syntaxique, sans faire appel à la notion de validité dans une structure. C'est l'objet du paragraphe suivant.

2.4.2 formule close, variables libres et liées.

La notion de formule close se définit sans difficulté pour les formules propositionnelles. En présence de quantificateurs, c'est un peu plus compliqué, on peut maintenant le faire proprement grâce aux définitions par induction.

On ne va pas définir formellement la notion intuitive d'*occurrence* d'un symbole dans une formule : la place ou l'endroit où ce symbole apparaît, ce serait inutilement technique. On pourrait alors définir par induction les notions déjà abordées d'occurrences de variables libres et liées. Ceci ne nous empêchera d'ailleurs pas d'utiliser ces notions. En fait on peut le plus souvent se contenter de définitions où la notion d'occurrence reste implicite comme dans ce qui suit. Tout d'abord, on peut définir par induction *x apparaît dans la formule F* qui se dit également *x a une occurrence dans F* (définition laissée au lecteur).

On définit ensuite *x apparaît libre dans une formule F* (ou x a une *occurrence libre* dans F) également par induction sur les formules :

formules atomiques si x apparaît dans une formule atomique A, alors x apparaît libre dans A.

absurde x n'apparaît pas libre dans $\bot$.

négation x apparaît libre dans $\neg F$ quand x apparaît libre dans F.

implication ... x apparaît libre dans $(F \to G)$ quand x apparaît libre dans F ou quand x apparaît libre dans G. De même pour la conjonction et la disjonction.

quantifications Si $y \neq x$, x apparaît libre dans $\forall y\ F$ quand x apparaît libre dans F, sinon x n'apparaît pas libre dans $\forall x\ F$. De même pour la quantification existentielle.

On peut maintenant définir une *formule close* ou énoncé : c'est une formule F telle qu'aucune variable de $\mathcal{V}$ n'apparaît libre dans F. On dira que x *apparaît liée dans* F, ou que x a une *occurrence liée* dans F quand x apparaît dans F et que x n'apparaît pas libre dans F. Dans une formule close toutes les occurrences de toutes les variables qui apparaissent dans la formule sont liées.

On considérera que deux formules sont identiques si elles sont en fait *identiques modulo un renommage cohérent des variables liées*. Il s'agit d'une notion purement syntaxique, que l'on peut définir formellement par induction (on ne le fera pas). Par exemple $\exists z\ y = x + z$ est une formule dans laquelle x et y apparaissent libres et z liée. Elle est identique à $\exists t\ y = x + t$: on a renommé z en t qui est bien une variable liée, mais elle n'est pas identique à $\exists z\ t = x + z$, qui est une formule qui « parle » de x et t, et plus de x et y.

3 Interprétation.

3.1 Introduction.

On va maintenant définir formellement *la sémantique* d'un langage du premier ordre. Il s'agit de définir l'interprétation dans une structure donnée des termes et des formules du langage définis à la section précédente.

On va tout simplement formaliser la définition intuitive d'interprétation pratiquée par exemple en algèbre. Il est essentiel dans ce qui suit qu'il n'y ait que deux valeurs de vérité. C'est à dire que dans une structure, un énoncé clos sera vrai ou faux. Peu importe qu'éventuellement nous ne sachions pas quelle alternative est la bonne.

Pour définir la sémantique, on se sert librement des notions logique usuelles, de l'égalité (des éléments d'un structure), des quantificateurs et connecteurs logiques usuels. On parle de langage objet et meta-langage : le langage objet est celui dont on a défini la syntaxe et dont on va définir la sémantique. Le meta-langage est celui que l'on utilise pour décrire ceci.

3.2 Signature et structure.

Étant donné une signature $\mathcal{S}$, une *$\mathcal{S}$-structure* $\mathcal{M}$ est la donnée :
- d'un ensemble *non vide*, soit M, appelé *ensemble de base* de la structure $\mathcal{M}$;
- d'un élément $\overline{c}^{\mathcal{M}}$ de M pour chaque symbole de constante c de $\mathcal{S}$;
- d'une fonction[3] $\overline{f}^{\mathcal{M}}$ de M^n dans M pour chaque symbole de fonction f d'arité n dans $\mathcal{S}$;
- d'un sous-ensemble $\overline{R}^{\mathcal{M}}$ de M^n pour chaque symbole de prédicat R d'arité n de $\mathcal{S}$.

Ainsi, si nous prenons le langage de l'arithmétique de signature $\mathcal{P} = (0, s, +, \cdot, \leqslant)$ défini au paragraphe 2.2.1, on peut définir $\mathcal{N} = (\mathbb{N}, \overline{0}^{\mathcal{N}}, \overline{s}^{\mathcal{N}}, \overline{+}^{\mathcal{N}}, \overline{\cdot}^{\mathcal{N}}, \overline{\leqslant}^{\mathcal{N}})$ muni des constantes, opérations et relations, usuelles pour interpréter chacun des symboles de $\mathcal{P}$. Ainsi s est interprété par la fonction de $\mathbb{N} \to \mathbb{N}$ qui ajoute 1 à un entier, $+$ par la fonction addition usuelle (à deux arguments) etc.

Des notations comme $\overline{s}^{\mathbb{N}}$, $\overline{+}^{\mathbb{N}}$ sont assez lourdes. S'il n'y a pas d'ambiguïté sur la structure concernée, on oubliera l'indication de celle-ci, par exemple on notera $\overline{s}$, $\overline{+}$. On peut même de noter de la même façon symbole interprétation, s'il est clair d'après le contexte que l'on parle de l'interprétation.

Une autre $\mathcal{P}$-structure est $\mathbb{Z}$ muni des opérations et prédicats usuels, $\mathbb{Z}/2\mathbb{Z}$ en interprétant 0 par le 0 de $\mathbb{Z}/2\mathbb{Z}$, les opérations avec leur interprétation usuelle, et l'ordre par exemple par $\{(0,0),(0,1),(1,1)\}$ (ce qui signifie que dans cette structure, on a $0 \leqslant 0$, $0 \leqslant 1$ et $1 \leqslant 1$, mais pas $1 \leqslant 0$)[4]. On pourrait tout aussi bien construire une structure $\mathcal{N}'$ d'ensemble de base $\mathbb{N}$, où l'ordre est usuel, mais 0 est interprété par 1, s par la fonction constante nulle etc. Ce sont les axiomes que doit satisfaire la structure qui imposeront des contraintes sur l'interprétation.

L'ensemble de base d'une structure est toujours non vide[5]. Par contre il peut contenir un seul élément, mais c'est un cas assez dégénéré puisque l'interprétation des symboles de fonctions et de constantes est imposée, et que les symboles de prédicats n'ont que deux interprétations possibles ($\emptyset$ ou M^n).

3.3 Les formules étendues aux éléments de la structure.

On veut définir l'interprétation des termes et des formules closes. Mais il n'y a pas de définition par induction sur les seules formules closes à cause des quantificateurs : si la formule $\forall x\ F$ est close, on s'attend par contre en général à ce que la formule F dépende elle de x.

Intuitivement, on interprète une formule avec par exemple une seule variable libre, par l'ensemble des éléments de la structure qui « rendent » cette formule vraie. Une formule avec une variable libre décrit en effet une propriété des éléments de l'ensemble de base de la structure. S'il y a deux variables libres ce seront des ensembles de couples, etc.

Pour définir formellement l'interprétation, on choisit d'étendre le langage de départ $\mathcal{S}$ au langage $\mathcal{S}_M$ obtenu en ajoutant tous les éléments de la structure $\mathcal{M}$ comme nouvelles constantes. Par exemple si le langage $\mathcal{S}$ a pour signature $(\leqslant)$, $2 \leqslant 3$ est une formule du langage $\mathcal{S}_{\mathbb{Z}}$ (mais pas du langage $\mathcal{S}$). C'est juste une convention formelle pour permettre l'écriture d'éléments de la structure dans les formules.

[3] Les fonctions sont toujours supposées *partout définies*.

[4] Cet ordre n'est pas compatible avec l'addition

[5] Outre que le cas où l'ensemble de base est vide a assez peu d'intérêt, cela compliquerait les systèmes de démonstrations si l'on voulait que celles-ci restent correctes dans de telles structures.

3.4 Interprétation.

Soit $\mathcal{M}$ une $\mathcal{S}$-structure d'ensemble de base M. On va définir successivement l'interprétation des termes (par induction), des formules atomiques, et des formules (par induction). Les défintions par induction peuvent être vues comme des définitions par récurrence sur la longueur de la formule.

3.4.1 Interprétation des termes du langage étendu.

On définit maintenant l'interprétation des termes clos du langage étendu : les symbols de constantes et fonctions de $\mathcal{S}$ ont l'interprétation attendue, les éléments de $\mathcal{M}$ sont interprétés par eux-mêmes.

Plus formellement, l'interprétation d'un terme t du langage de signature $\mathcal{S}$ dans $\mathcal{M}$ est notée $\overline{t}^{\mathcal{M}}$. C'est un élément de M qui est défini par induction sur t.

élément de $\mathcal{M}$ $\overline{m}^{\mathcal{M}} = m$, pour $m \in M$;

constante $\overline{c}^{\mathcal{M}} = \overline{c}^{\mathcal{M}}$ pour c une constante de $\mathcal{S}$;

fonction $\overline{f\,t_1 \dots t_n}^{\mathcal{M}} = \overline{f}^{\mathcal{M}}(\overline{t_1}^{\mathcal{M}}, \dots, \overline{t_n}^{\mathcal{M}})$ pour f un symbole de fonction de $\mathcal{S}$ d'arité n.

3.4.2 Interprétation des formules : notations

Une formule est interprétée par "vrai" ou "faux". Pour dire qu'une formule close du langage étendu $F(m_1, \dots, m_p)$ est *vraie dans la structure* $\mathcal{M}$ (m_i sont des éléments de $\mathcal{M}$) on note alors :

$$\mathcal{M} \models F(m_1, \dots, m_p)$$

et on dira également que la formule $F(m_1, \dots, m_p)$ est *satisfaite* ou encore *valide* dans la structure $\mathcal{M}$. Dans le cas contraire on notera $\mathcal{M} \not\models F(m_1, \dots, m_p)$. On dit encore que $\mathcal{M}$ est un *modèle de la formule* $F(m_1, \dots, m_p)$.

La satisfaction des formules closes du langage initial est juste un cas particulier de la définition qui précède et on note alors simplement $\mathcal{M} \models F$.

Dans la suite, on écrit parfois F, même pour une formule du langage étendu par des éléments de la structure.

3.4.3 Interprétation des formules atomiques closes du langage étendu.

prédicat $\mathcal{M} \models R\,t_1 \dots, t_n$ ssi $(\overline{t_1}^{\mathcal{M}}, \dots, \overline{t_n}^{\mathcal{M}}) \in \overline{R}^{\mathcal{M}}$, pour R un prédicat de S d'arité n ;

égalité $\mathcal{M} \models t = t'$ ssi $\overline{t}^{\mathcal{M}} = \overline{t'}^{\mathcal{M}}$.

On remarque que dans la dernière assertion, le signe "$=$" utilisé dans deux sens différents : comme symbole du langage (première occurrence), et dans son sens usuel, celui de l'identité des éléments de la structure pour la deuxième occurrence (qui est l'élégalité du meta-langage).

La différence essentielle entre l'égalité et les autres symboles de prédicat est que pour l'égalité, l'interprétation est imposée : c'est l'identité des éléments de la structure.

3.4.4 Interprétation des formules closes du langage étendu.

On peut maintenant définir l'interprétation des formules par induction. Notez bien que pour les connecteurs binaires, définis sur F et G, il n'y a que 4 cas possibles suivant que $\mathcal{M} \models F$ ou $\mathcal{M} \not\models F$ et $\mathcal{M} \models G$ ou $\mathcal{M} \not\models G$.

Formules atomiques Voir le paragraphe précédent.

absurde $\mathcal{M} \not\models \bot$.

négation $\mathcal{M} \models \neg F$ ssi $\mathcal{M} \not\models F$.

conjonction $\mathcal{M} \models (F \wedge G)$ ssi $\mathcal{M} \models F$ et $\mathcal{M} \models G$.

disjonction $\mathcal{M} \not\models (F \vee G)$ ssi $\mathcal{M} \not\models F$ et $\mathcal{M} \not\models G$.

Le "ou" est le ou inclusif du meta-langage, usuel en mathématique. Si ceci est clair, on peut aussi bien écrire $\mathcal{M} \models (F \vee G)$ ssi $\mathcal{M} \models F$ ou $\mathcal{M} \models G$.

implication $\mathcal{M} \not\models (F \to G)$ ssi $\mathcal{M} \models F$ et $\mathcal{M} \not\models G$.

Par conséquent $\mathcal{M} \models (F \to G)$ dans les trois autres cas possibles : $\mathcal{M} \models F$ et $\mathcal{M} \models G$, $\mathcal{M} \models \neg F$ et $\mathcal{M} \models G$, $\mathcal{M} \models \neg F$ et $\mathcal{M} \models \neg G$).

En fait on aurait pu écrire $\mathcal{M} \models (F \to G)$ ssi $\mathcal{M} \models F \Rightarrow \mathcal{M} \models G$,

où le signe $\Rightarrow$ désigne l'implication dans le meta-langage. Le sens intuitif de l'implication, du fait qu'il n'y ait que deux valeurs de vérité, donne l'interprétation ci-dessus.

quantification universelle $\mathcal{M} \models \forall x\ F(x)$ ssi $\mathcal{M} \models F(m)$ pour tout élément m de M (on ne substitue que les occurrences de x libres dans $F(x)$).

quantification existentielle $\mathcal{M} \models \exists x\ F(x)$ ssi $\mathcal{M} \models F(m)$ pour au moins un élément m de $\mathcal{M}$ (même remarque).

Remarquez que dès qu'une formule close contient des quantificateurs, on a effectivement besoin des formules étendues pour définir l'interprétation.

Une formule close de $\mathcal{S}$ est dite *universellement valide* quand elle est satisfaite dans toutes les $\mathcal{S}$-structures.

Un exemple de formule universellement valide pour n'importe quelle signature $\mathcal{S}$ est $\forall x\ x = x$. En effet pour toute structure $\mathcal{M}$, pour tout élément m de l'ensemble de base de $\mathcal{M}$, $\mathcal{M} \models m = m$, et donc pour toute structure $\mathcal{M}$, $\mathcal{M} \models \forall x\ x = x$.

Voyons un autre exemple, prenons une signature $\mathcal{S}$ quelconque, et soit $F(x)$ une formule de $\mathcal{S}$ ayant x comme seule variable libre. alors $\forall x\ F(x) \rightarrow \exists x\, F(x)$ est universellement valide. En effet soit $\mathcal{M}$ une $\mathcal{S}$-structure. Supposons que $\mathcal{M} \models \forall x\ F$. Cela signifie par définition que pour tout élément m de l'ensemble de base de $\mathcal{M}$ $\mathcal{M} \models F(m)$. Comme cet ensemble de base est non vide, il contient au moins un élément m_0 et comme $\mathcal{M} \models F(m_0)$, $\mathcal{M} \models \exists x F$. On a bien montré que $\mathcal{M} \models \forall x\ F(x) \rightarrow \exists x\, F(x)$.

Dans l'exemple ci-dessus, n'importe quelle formule $F(x)$ avec x pour seule variable libre convient. On parle alors de *schéma de formules*.

On a vérifié sur ces exemples que la définition de la validité dans une structure fonctionnait correctement. C'est bien de ceci qu'il s'agit, on apprend rien de neuf sur la validité des énoncés eux-mêmes.

On arrête là pour le moment au sujet des formules universellement valides. On verra dans les chapitres suivant des schémas de formules universellement valides purement propositionnels, et d'autres utilisant les quantificateurs comme le précédent.

3.4.5 Remarques.

Cette formalisation de la sémantique des formules du premier ordre ci-dessus est due à Tarski. Il faut bien comprendre que plus que de définir l'interprétation d'une formule, il s'agit de formaliser celle-ci. En effet vous avez peu de chance de comprendre cette définition sans avoir déjà pratiqué les mathématiques, en particulier la négation, la conjonction et les quantificateurs ! Ainsi les connecteurs et les quantificateurs sont tout simplement définis ... en utilisant ces mêmes connecteurs dans le meta-langage. Ceci peut paraître tautologique, mais ne l'est pas car cette définition formelle permet de faire proprement des démonstrations de résultats non triviaux utilisant la sémantique, comme le théorème de complétude déjà mentionné.

3.5 Satisfaisabilité, axiomatisation.

3.5.1 Formules satisfaisables.

Une formule close F du langage de signature $\mathcal{S}$ est dite *satisfaisable* si elle possède au moins un modèle, c'est à dire s'il existe une $\mathcal{S}$-structure $\mathcal{M}$ telle que $\mathcal{M} \models F$.

La notion de formule satisfaisable est en quelque sorte duale de la notion de formule universellement valide :

Proposition 3.1 *Une formule close F est satisfaisable ssi $\neg F$ n'est pas universellement valide.*

Démonstration. La formule F est satisfaisable ssi existe un modèle $\mathcal{M}$ de F. Par définition de l'interprétation $\mathcal{M} \models F$ ssi $\mathcal{M} \not\models \neg F$, donc F est satisfaisable ssi $\neg F$ n'est pas universellement valide. ■

Une formule satisfaisable définit donc une classe de structures : celles qui la satisfont. Par exemple la formule du langage de l'égalité (signature vide) :

$$\exists x \exists y\ \neg x = y$$

est satisfaite dans toutes les structures égalitaires – c'est-à-dire simplement les ensembles – ayant au moins deux éléments. On dit qu'elle *axiomatise* cette notion.

De même la formule

$$\exists x \exists y \forall z (z = x \vee z = y)$$

est satisfaite dans les ensembles ayant au plus deux éléments, et la conjonction des deux formules précédentes :

$$\exists x \exists y\ \neg x = y \wedge \exists x \exists y \forall z (z = x \vee z = y)$$

dans les ensembles ayant deux éléments.

3.5.2 Quelques notations.

Tout d'abord on peut s'autoriser des abréviations usuelles. Par exemple on écrira $x \neq y$ pour $\neg x = y$, $\neq$ n'est pas un symbole du langage, $x \neq y$ désigne la formule $\neg x = y$, et il est clair qu'à une formule de signature $\mathcal{S} \cup \{\neq\}$ apparaît, on fait correspondre par une substitution « simple » une formule du langage $\mathcal{S}$.

En généralisant les exemples précédents, on s'aperçoit facilement que l'on peut axiomatiser les ensembles à au moins (resp. au plus, resp. exactement) trois éléments, puis généraliser à un entier n quelconque. Pour axiomatiser la notion d'ensemble à au moins n éléments on écrira :

$$\exists x_1 \dots \exists x_n \bigwedge_{0 \leqslant i < j \leqslant n} x_i \neq x_j. \tag{1}$$

Pour pouvoir exprimer ceci on a introduit des notations *qui ne font pas partie du langage étudié* mais qui là encore renvoient *pour chaque entier* n à une formule du premier ordre. En ce qui concerne la suite de quantificateurs $\exists x_1 \dots \exists x_n\ F$, la signification pour chaque entier n est claire. La conjonction $\bigwedge_{0 \leqslant i < j \leqslant n} x_i \neq x_j$, même si elle pourrait se définir syntaxiquement, utilise implicitement les propriétés dite d'associativité et de commutativité de la conjonction qui sont sémantiques (l'interprétation d'une suite de conjonctions ne dépend ni de l'ordre ni du parenthésage). Tout ce qui nous intéresse est de savoir qu'il existe pour chaque entier n une formule du langage notée $\bigwedge_{0 \leqslant i < j \leqslant n} x_i \neq x_j$ vérifiant pour toute structure $\mathcal{M}$:

$$\mathcal{M} \models \bigwedge_{0 \leqslant i < j \leqslant n} x_i \neq x_j \text{ ssi pour tout couple } i, j \text{ tel que } 0 \leqslant i < j \leqslant n\ \mathcal{M} \models x_i \neq x_j\ .$$

Il faut bien prendre garde à ce que, pour prendre un exemple, on ne peut parler que de *conjonctions finies*, les conjonctions infinies, qui intuitivement ont un sens n'existent pas en langage du premier ordre. Dans le même ordre d'idée, l'entier n dans la formule (1) *ne fait pas partie du langage*. En particulier on ne peut quantifier sur n dans le langage. Par exemple on dirait volontiers qu'un ensemble est infini à la condition :

$$\text{Pour tout entier } n\ \ \exists x_1 \dots \exists x_n \bigwedge_{0 \leqslant i < j \leqslant n} x_i \neq x_j.$$

Mais ceci n'est pas une formule du langage du premier ordre égalitaire (pour éviter les confusions, on n'a pas utilisé le signe $\forall$ pour la quantification universelle sur n). On peut montrer qu'il n'est pas possible d'axiomatiser la notion d'ensemble infini par une seule formule du premier ordre.

3.5.3 Théories.

Un moyen de remédier partiellement à certains manques d'expressivité de la logique du premier ordre est d'utiliser des ensembles infinis de formules. Évidemment il faudra bien trouver un moyen « fini » de décrire cette infinité de formules.

Une *théorie du premier ordre* dans un langage de signature $\mathcal{S}$ est un ensemble de formules *closes* du langage (comme on se situe en logique du premier ordre on parlera simplement de théorie).

Une $\mathcal{S}$-structure est $\mathcal{M}$ *modèle d'une théorie* T quand elle est modèle de chacune des formules de la théorie. On note comme pour les formules $\mathcal{M} \models T$.

Une théorie est dite *satisfaisable* si elle a un modèle, c'est à dire s'il existe une structure $\mathcal{M}$ qui est modèle de la théorie. Une théorie est dite *incohérente*, inconsistante, ou contradictoire dans le cas contraire, c'est à dire si elle n'a pas de modèle.

Une propriété des $\mathcal{S}$-structures est dite *axiomatisée* par une théorie T quand les structures ont la propriété en question si et seulement si elles sont modèles de T.

Ainsi la notion d'ensemble infini est axiomatisée par la théorie (infinie) :

$$\{\exists x_1 \dots \exists x_n \bigwedge_{0 \leqslant i < j \leqslant n} x_i \neq x_j \ / \ n \in \mathbb{N}\}. \tag{2}$$

Une propriété *axiomatisable* est une propriété qui est axiomatisée par une certaine théorie. Quand de plus cette propriété est axiomatisée par une théorie finie on dit qu'elle est *finiment axiomatisable*. Il est facile de voir qu'alors la propriété est axiomatisée par une seule formule : la conjonction des axiomes de la théorie.

Tout ne peut pas se résoudre en considérant une infinité de formules. Ainsi on peut exprimer dans le langage de l'égalité pour chaque entier n qu'un ensemble possède au plus n éléments, comme devrait vous en convaincre l'écriture suivante :

$$\exists x_1 \dots \exists x_n\ \forall z (z = x_1 \vee \dots \vee z = x_n)$$

que l'on peut aussi écrire :

$$\exists x_1 \dots \exists x_n \ \forall z \bigvee_{i=1}^{n} z = x_i \ .$$

Un ensemble est fini s'il vérifie :

$$\text{il existe un entier } n \text{ tel que } \exists x_1 \dots \exists x_n \ \forall z \bigvee_{i=1}^{n} z = x_i \ .$$

mais il n'y a aucun moyen apparent d'exprimer ceci dans le langage égalitaire du premier ordre, même avec une infinité de formules. On peut montrer qu'en fait la notion d'ensemble fini ne s'axiomatise pas au premier ordre.

On a vu que l'on pouvait exprimer qu'un ensemble est fini ou infini par une formule de la théorie des ensembles, qui est elle même une théorie du premier ordre. Ceci n'a rien de contradictoire avec ce qui précède, puisque ces notions seront relatives à un modèle « attendu » de la théorie des ensembles.

3.5.4 Arithmétique de Peano.

Voici comment on axiomatise l'arithmétique au premier ordre.

successeur non nul $\forall x \in \mathbb{N} \ s(x) \neq 0$;

injectivité du successeur $\forall x, y \in \mathbb{N} \ (s(x) = s(y) \Rightarrow x = y)$;

récurrence Pour tout prédicat $P(x, x_1, \dots, x_p)$ du langage de l'arithmétique (on note $\overline{a} = a_1, \dots, a_p$) :

$$\forall \overline{a} \, (\ P[0, \overline{a}], \forall y \, (P[y, \overline{a}] \Rightarrow P[s\, y, \overline{a}]) \Rightarrow \forall x \ P[x, \overline{a}] \ ) \ ;$$

addition $\forall x \ x + 0 = x$; $\forall x, y \ x + s\, y = s(x + y)$;

multiplication $\forall x \ x \cdot 0 = 0$; $\forall x, y \ x \cdot s\, y = x \cdot y + x$;

ordre $\forall x \ (x \leqslant 0 \Leftrightarrow x = 0)$; $\forall x, y \ [x \leqslant s\, y \Leftrightarrow (x \leqslant y \vee x = s\, y)]$.

La différence essentielle avec les axiomes de Peano vus en théorie des ensembles, c'est que le langage a été restreint aux propriétés polynômiales et leurs combinaisons par connecteurs et quantifications. En particulier, la récurrence ne porte « que » sur ces propriétés. Il n'y a plus de théorème général de définition par récurrence (on a pris pour axiomes les définitions par récurrence de l'addition, de la multiplication, et de l'ordre), même si on peut en fait parler indirectement de certaines fonctions, comme l'exponentielle, en représentant leur graphe par une formule[6].

Cette théorie est suffisante pour développer l'arithmétique élémentaire mais pas l'analyse réelle.

3.6 Morphismes.

On généralise les notions de morphisme déjà abordées en algèbre à une signature quelconque.

Étant donné une signature $\mathcal{S}$ on appelle *$\mathcal{S}$-homomorphisme* de $\mathcal{S}$-structures, une application ϕ d'une $\mathcal{S}$-structure $\mathcal{M}$ dans une $\mathcal{S}$-structure $\mathcal{N}$ qui vérifie :

constante $\phi\left(\overline{c}^{\mathcal{M}}\right) = \overline{c}^{\mathcal{N}}$ pour tout symbole de constante c de $\mathcal{S}$.

fonction $\phi\left(\overline{f}^{\mathcal{M}}(m_1, \dots, m_n)\right) = \overline{f}^{\mathcal{N}}(\phi(m_1), \dots, \phi(m_n))$ pour tout symbole de fonction f d'arité n, pour tous $m_1, \dots, m_n \in \mathcal{M}$.

prédicat Si $(m_1, \dots, m_n) \in \overline{R}^{\mathcal{M}}$ alors $(\phi(m_1), \dots, \phi(m_n)) \in \overline{R}^{\mathcal{N}}$, pour tout symbole de prédicat R d'arité n, pour tous $m_1, \dots, m_n \in \mathcal{M}$.

Cette définition ne fait intervenir que la signature, et aucune notion d'axiomatique. Par exemple l'identité est un morphisme $(\mathbb{N}, 0, +)$ dans $(\mathbb{Z}, 0, +)$ (signature $(0, +)$), qui est un morphisme de monoïde mais pas de groupe. Si l'on veut parler de morphisme de groupe, on ajoutera au langage un symbole de fonction unaire "$-$" pour l'opposé.

Une conséquence immédiate de cette définition est la suivante :

Lemme 3.2 *Soit $\mathcal{S}$ une signature, deux $\mathcal{S}$-structures $\mathcal{M}$ et $\mathcal{N}$, ϕ un homomorphisme de $\mathcal{M}$ dans $\mathcal{N}$. Soit t un terme du langage de $\mathcal{S}$ dont les variables libres sont parmi $x_1, \dots, x_p$. Soit $m_1, \dots, m_p$ des éléments de $\mathcal{M}$. On a :*

$$\phi\left(\overline{t(m_1, \dots, m_p)}^{\mathcal{M}}\right) = \overline{t(\phi(m_1), \dots, \phi(m_p))}^{\mathcal{N}}$$

[6] Ce résultat n'est pas complètement évident a priori ; Gödel en a eu besoin pour la démonstration de son théorème d'incomplétude en 1931. Il utilise une astuce où intervient le théorème des restes chinois, pour coder la suite des calculs successifs de l'exponentielle, par itération de multiplications.

Démonstration. La preuve est immédiate par induction sur la définition des termes.

Un *monomorphisme* ou *plongement* de $\mathcal{S}$-structures de $\mathcal{M}$ dans $\mathcal{N}$ est un homomorphisme injectif ϕ de $\mathcal{M}$ dans $\mathcal{N}$ qui vérifie de plus que

prédicat $(m_1, \ldots, m_n) \in \overline{R}^{\mathcal{M}}$ **ssi** $(\phi(m_1), \ldots, \phi(m_n)) \in \overline{R}^{\mathcal{N}}$, pour tout symbole de prédicat R d'arité n, pour tous $m_1, \ldots, m_n \in \mathcal{M}$.

Un *isomorphisme* de $\mathcal{S}$-structures de $\mathcal{M}$ dans $\mathcal{N}$ est un homomorphisme bijectif ϕ de $\mathcal{M}$ dans $\mathcal{N}$ dont la réciproque est un homomorphisme de $\mathcal{S}$-structure, ou encore un plongement bijectif.

Intuitivement il devrait être clair que deux structures isomorphes vérifient les mêmes formules closes. Ce résultat se démontre par induction. Comme la définition de l'interprétation des formules closes passe par celle des formules en général, on est amené à utiliser des formules étendues soient pas des éléments de $\mathcal{M}$, soient pas des éléments de $\mathcal{N}$, pour démontrer ce lemme :

Lemme 3.3 *Soit $\mathcal{S}$ une signature, soit $\mathcal{M}$ et $\mathcal{N}$ et deux $\mathcal{S}$ structures isomorphes par ϕ. Pour toute formule $F(x_1, \ldots, x_p)$ du langage du premier ordre sur $\mathcal{S}$ dont les variables libres sont parmi $x_1, \ldots, x_p$, pour tout $m_1, \ldots, m_p \in \mathcal{M}$, on a :*

$$\mathcal{M} \models F(m_1, \ldots, m_p) \quad ssi \quad \mathcal{N} \models F(\phi(m_1), \ldots, \phi(m_p)) \ .$$

On en déduit, dans le cas des formules closes, le résultat attendu :

Proposition 3.4 *Soit $\mathcal{S}$ une signature, soit $\mathcal{M}$ et $\mathcal{N}$ et deux $\mathcal{S}$ structures isomorphes par ϕ. Pour toute formule close F du langage du premier ordre sur $\mathcal{S}$*

$$\mathcal{M} \models F \quad ssi \quad \mathcal{N} \models F \ .$$

Démonstration (lemme). On montre par induction sur la structure des formules le résultat *pour toute suite d'éléments de $\mathcal{M}$* de longueur le nombre de variables libres de la formule. Le résultat est intuitivement évident, et de fait aucun cas ne présente de difficulté. Passons en quelques uns en revue.

formules atomiques On utilise le lemme 3.2. Si la formule atomique est une égalité c'est une conséquence immédiate de l'injectivité de ϕ. Si la formule atomique est un prédicat, c'est une conséquence de la définition de morphisme pour ϕ et ϕ^{-1} (c'est à dire que ϕ est un plongement).

négation C'est une conséquence immédiate de l'hypothèse d'induction. On utilise la définition de la satisfaction, et le fait que l'on démontre bien une équivalence par induction.

conjonction On utilise directement la définition de la satisfaction, et l'hypothèse d'induction.

quantification universelle C'est une conséquence de l'hypothèse d'induction et de la surjectivité de ϕ. On a $F = \forall x \ F'(x_1, \ldots, x_p, x)$. Soient $m_1, \ldots, m_p \in \mathcal{M}$. Par définition de la satisfaction :

$\mathcal{M} \models \forall x \ F'(m_1, \ldots, m_p)$ si et seulement si

$$\text{pour tout } m \text{ de } \mathcal{M},\ \mathcal{M} \models F'(m_1, \ldots, m_p, m) \tag{1}$$

$\mathcal{N} \models \forall x \ F'(\phi(m_1), \ldots, \phi(m_p))$ si et seulement si

$$\text{pour tout } n \text{ de } \mathcal{N},\ \mathcal{N} \models F'(\phi(m_1), \ldots, \phi(m_p), n] \tag{2}$$

L'hypothèse d'induction donne pour toute suite $(m_1, \ldots, m_p, m)$:

$$\begin{array}{c} \mathcal{M} \models F'(m_1, \ldots, m_p, m) \\ \text{ssi} \\ \mathcal{N} \models F(\phi(m_1), \ldots, \phi(m_p), \phi(m)) \end{array}$$

On doit démontrer que (1) $\Leftrightarrow$ (2).

(1) $\Rightarrow$ (2) Soit n un élément quelconque de $\mathcal{N}$, comme ϕ est surjective, il existe m dans $\mathcal{M}$ tel que $\phi(m) = n$. On utilise l'hypothèse d'induction pour $(m_1, \ldots, m_p, m)$.

(2) $\Rightarrow$ (1) Soit m un élément quelconque de $\mathcal{M}$, on utilise l'hypothèse d'induction pour $(m_1, \ldots, m_p, m)$.

Les autre cas se démontrent essentiellement de la même façon, par exemple la clause pour l'existentielle utilise également la surjectivité de ϕ. ■

On verra en section 5 que deux structures peuvent vérifier les mêmes formules closes d'un langage donné sans être isomorphes.

3.7 Sous-structure.

Soient deux structures $\mathcal{M}$ d'ensemble de base M et $\mathcal{N}$ d'ensemble de base N pour la même signature $\mathcal{S}$. On dit que $\mathcal{N}$ *est un sous-structure de* $\mathcal{M}$ quand :

- $N \subset M$;
- L'identité sur N est un plongement de $\mathcal{N}$ dans $\mathcal{M}$.

Un sous-ensemble N de M définit une sous-structure de $\mathcal{M}$ pour les restrictions des interprétations des symboles de la signature à N si et seulement

- $N \neq \emptyset$;
- Si c est un symbole de constante de $\mathcal{S}$, $\overline{c}^{\mathcal{M}} \in \mathcal{N}$;
- L'ensemble N est clos pour les $\overline{f}^{\mathcal{M}}$, f un symbole de fonction de $\mathcal{S}$.

On retrouve des notions bien connues en algèbre (sous-groupe, sous-anneau, etc.).

On appelle *formule universelle* une formule qui ne possède que des quantificateurs universels et de plus tous en tête de la formule, *formule existentielle* une formule qui ne possède que des quantificateurs existentiels et de plus tous en tête de la formule.

Par exemple dans le langage de la théorie des groupes de signature $\left(e, \cdot, (.)^{-1}\right)$, tous les axiomes de groupes sont des formules universelles :

- $\forall x, y, z\ (x \cdot y) \cdot z = x \cdot (y \cdot z)$;
- $\forall x\ x \cdot e = x$, $\forall x\ e \cdot x = x$;
- $\forall x\ x \cdot x^{-1} = e$, $\forall x\ (x)^{-1} \cdot x = e$.

On peut remarquer que :

Proposition 3.5 *Si F est une formule close universelle, si $\mathcal{M} \models F$, et si $\mathcal{N}$ est une sous-structure de $\mathcal{M}$, alors $\mathcal{N} \models F$.*

Démonstration. Immédiat par induction sur l'ensemble des formules de $\mathcal{S}$ étendues aux éléments de $\mathcal{N}$. ■

Dans le cas des groupes, dont les axiomes sont universels, on retrouve le résultat bien connu que H est un sous-groupe de G si et seulement si H est non vide, contient l'élément neutre, et qu'il est stable par multiplication et passage à l'inverse.

Exercice 1 Énoncer et justifier le résultat symétrique pour les formules closes existentielles.

3.8 Relation de conséquence sémantique.

Soit $\mathcal{S}$ une signature, T une théorie (c'est à dire un ensemble de formules closes) du langage de signature $\mathcal{S}$, et F une formule close du même langage. Nous dirons que *la théorie T a pour conséquence F* quand pour tout modèle $\mathcal{M}$ de T, est modèle de F, et nous noterons $\vdash_T F$ (ou $T \vdash F$),

$$\vdash_T F \text{ signifie « pour tout } \mathcal{M} \text{ si } \mathcal{M} \models T \text{ , alors } \mathcal{M} \models F \text{ ».}$$

Une conséquence immédiate de la définition de la satisfaction pour $\bot$ est que :

$$\text{Une théorie } T \text{ est incohérente ssi } \vdash_T \bot .$$

Nous avons déjà vu la définition de formule universellement valide, remarquons qu'un formule close est universellement valide ssi $\vdash F$. On dira que deux formules F et G *sont équivalentes* dans la théorie T quand tout modèle de T qui est modèle de F est également modèle de G et réciproquement, et l'on notera $F \equiv_T G$.

Si on utilise le signe $\leftrightarrow$ pour l'équivalence en tant que connecteur, qui est définie de façon usuelle par :

$$F \leftrightarrow G = (F \rightarrow G) \wedge (G \rightarrow F),$$

on voit qu'une conséquence immédiate des définitions qui précèdent et de la définition de la satisfaction est que pour toute théorie T, pour toute formules closes F et G :

$$F \equiv_T G \quad \text{ssi} \quad \vdash_T F \leftrightarrow G .$$

On généralise la conséquence et l'équivalence aux formules quelconques. Étant donné deux formules $F(x_1, \ldots, x_p)$ et $G(x_1, \ldots, x_p)$ dont les variables libres sont parmi $x_1, \ldots, x_p$

- On dit que $F(x_1, \ldots, x_p)$ a pour conséquence $G(x_1, \ldots, x_p)$ dans la théorie T, et on note $F(x_1, \ldots, x_p) \vdash_T G(x_1, \ldots, x_p)$, quand pour tout modèle $\mathcal{M}$ de T, pour tout p-uplets d'éléments de $\mathcal{M}$:

  $$\text{si } \mathcal{M} \models F(m_1, \ldots, m_p), \text{ alors } \mathcal{M} \models G(m_1, \ldots, m_p)$$

  (on vérifie que ceci équivaut à : $\mathcal{M} \models \forall x_1, \ldots, x_p\ (F \rightarrow G)$)

– On dit que $F(x_1, \ldots, x_p)$ et $G(x_1, \ldots, x_p)$ sont équivalentes dans la théorie T, et on note $F(x_1, \ldots, x_p) \equiv_T G(x_1, \ldots, x_p)$, quand pour tout modèle $\mathcal{M}$ de T, pour tout p-uplets d'éléments de $\mathcal{M}$:

$$\mathcal{M} \models F(m_1, \ldots, m_p) \text{ si et seulement si } \mathcal{M} \models G(m_1, \ldots, m_p)$$

(on vérifie que ceci équivaut à : $\mathcal{M} \models \forall x_1, \ldots, x_p \ (F \leftrightarrow G)$)

On généralise également ces définitions aux théories : une théorie T' est conséquence d'une théorie T si toute formule de T' est conséquence de T, deux théories T et T' sont équivalentes si T' est conséquence de T et T conséquence de T'.

4 Quelques équivalences classiques.

4.1 Equivalences propositionnelles.

On s'intéresse dans cette section aux formules propositionnelles, c'est à dire sans quantificateurs. On peut définir inductivement l'interprétation d'une formule propositionnelle close dans le langage étendu aux éléments du modèle). Les formules propositionnelles closes n'ont pas d'occurences de variables. Pour étudier ces formules, on peut donc « oublier » la structure des formules atomiques : c'est le calcul propositionnel.

Dans la suite A, B et C désignent des formules quelconques. On a 2 possibilités de valeurs de vérité pour A, 4 pour A et B, 8 pour A, B et C etc. Dans le cas des formules propositionnelles, on peut résumer la définition de la validité dans les *tables de vérité* qui suivent (0 pour faux, 1 pour vrai) :

A	$\neg A$
0	1
1	0

A	B	$A \wedge B$	$A \vee B$	$A \rightarrow B$	$A \Leftrightarrow B$
0	0	0	0	1	1
0	1	0	1	1	0
1	0	0	1	0	0
1	1	1	1	1	1

Si A et B sont des formules closes, leurs interprétations et celles des formules composées, dans une structure donnée, correspond à une ligne de la table de vérité.

On va donner quelques équivalences usuelles. Une façon de les vérifier et d'utiliser la définition de la validité et les tables de vérité. Une table de vérité a 2^n lignes si n variables apparaissent dans les formules en jeu. Par exemple la table de vérité de 4 lignes ci-dessous permet de vérifier que $(A \wedge B) \vee (\neg A \wedge \neg B) \equiv A \Leftrightarrow B$.

A	B	$A \wedge B$	$\neg A$	$\neg B$	$\neg A \wedge \neg B$	$(A \wedge B) \vee (\neg A \wedge \neg B)$
0	0	0	1	1	1	1
0	1	0	1	0	0	0
1	0	0	0	1	0	0
1	1	1	0	0	0	1

Soit en utilisant des tables de vérité, soit en « raisonnant » directement, on montre les équivalences qui suivent.

4.1.1 Négation des connecteurs usuels.

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

Lois de de Morgan :

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

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

$$\neg(A \leftrightarrow B) \equiv (A \leftrightarrow \neg B)$$
$$\neg(A \leftrightarrow B) \equiv (\neg A \leftrightarrow B)$$

4.1.2 Expression des connecteurs avec $\neg$, $\wedge$ et $\vee$.

$$\bot \equiv (A \wedge \neg A)$$
$$\top \equiv (A \vee \neg A)$$

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

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

Tous les connecteurs que l'on a introduit s'expriment avec les trois seuls connecteurs $\neg$, $\wedge$ et $\vee$. De façon plus générale, tout connecteur « possible » a une table de vérité, qui se décrit avec ces trois connecteurs : chaque ligne se décrit avec $\wedge$ et $\neg$, la table est une disjonction des ces formules. On dit que $\{\neg, \wedge, \vee\}$ est un *système complet de connecteurs*.

Par les lois de de Morgan, $\{\neg, \wedge\}$, est donc aussi un système complet de connecteurs.

4.1.3 Propriétés de la disjonction et de la conjonction.

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 :

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

idempotence :

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

absorption :

$$(A \wedge \bot) \equiv \bot$$
$$(A \wedge (A \vee B)) \equiv A$$
$$(A \vee \top) \equiv \top$$
$$(A \vee (A \wedge B)) \equiv A$$

neutre :

$$(A \wedge \top) \equiv A$$
$$(A \vee \bot) \equiv A$$

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

4.1.4 Propriétés de l'implication et de l'équivalence.

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

contraposée :

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

Distributivité à droite :

$$(A \to (B \wedge C)) \equiv ((A \to B) \wedge (A \to C))$$
$$(A \to (B \vee C)) \equiv ((A \to B) \vee (A \to C))$$

Pseudo-distributivité à gauche :

$$((A \wedge B) \to C) \equiv ((A \to C) \vee (B \to C))$$
$$((A \vee B) \to C) \equiv ((A \to C) \wedge (B \to C))$$

Remarque : il n'y a pas de propriété analogue à ces 4 dernières pour l'équivalence.

$\bot$ et $\top$:

$$(A \to \bot) \equiv \neg A \quad (\bot \to A) \equiv \top \quad (A \to \top) \equiv \top \quad (\top \to A) \equiv A$$

$$(A \leftrightarrow \bot) \equiv \neg A \quad (A \leftrightarrow \top) \equiv A$$

4.1.5 Encore quelques équivalences.

$$(\neg A \to A) \equiv A \quad ((A \to B) \to A) \equiv A \quad ((A \to B) \to B) \equiv (A \vee B)$$

4.2 Equivalences utilisant les quantificateurs.

Les équivalences qui suivent ne sont plus propositionnelles : plus question de table de vérité pour les vérifier. On peut les montrer en reprenant la définition sémantique.

La signature du langage est quelconque. On désigne par F et G des formules quelconques.

Quantifications "inutiles" :

Si x n'apparait pas libre dans F :

$$\forall x\ F \equiv F \quad \exists x\ F \equiv F \tag{1}$$

Commutation des quantificateurs :

$$\forall x \forall y\ F \equiv \forall y \forall x\ F \quad \exists x \exists y\ F \equiv \exists y \exists x\ F$$

$$\exists x \forall y\ F \vdash \forall y \exists x\ F$$

mais en général (cela dépend de F et de la structure dans laquelle on interprète la formule) la réciproque est fausse :

$$\forall y \exists x\ F \nvdash \exists x \forall y\ F$$

Négation des quantificateurs :

$$\neg \forall x\ F \equiv \exists x \neg F \quad \neg \exists x\ F \equiv \forall x \neg F \tag{2}$$

Conjonction et disjonction :

$$\forall x(F \wedge G) \equiv \forall x\ F \wedge \forall x\ G \quad \exists x(F \vee G) \equiv \exists x\ F \vee \exists x\ G \tag{3}$$

$$\exists x(F \wedge G) \vdash \exists x\ F \wedge \exists x\ G \quad \forall x\ F \vee \forall x\ G \vdash \forall x(F \vee G)$$

Mais « en général » la réciproque est fausse :

$$\exists x\ F \wedge \exists x\ G \nvdash \exists x(F \wedge G) \quad \forall x(F \vee G) \nvdash \forall x\ F \vee \forall x\ G$$

par contre si x n'apparaît pas dans G :

$$\exists x(F \wedge G) \equiv \exists x\ F \wedge G \quad \forall x(F \vee G) \equiv \forall x\ F \vee G \tag{4}$$

Remarquez que l'on pouvait déjà déduire des équivalences précédentes que si x n'apparaît pas dans G :

$$\exists x(F \vee G) \equiv \exists x\ F \vee G \quad \forall x(F \wedge G) \equiv \forall x\ F \wedge G$$

Implication :

$$\exists x(F \to G) \equiv \forall x\ F \to \exists x\ G$$
$$\exists x\ F \to \forall x\ G \vdash \forall x(F \to G)$$

mais « en général » la réciproque est fausse :

$$\forall x(F \to G) \nvdash \exists x\ F \to \forall x\ G$$

par contre si x n'apparait pas libre dans F :

$$\forall x(F \to G) \equiv F \to \forall x\ G \tag{5}$$

et si x n'apparait pas libre dans G :

$$\forall x(F \to G) \equiv \exists x\ F \to G \tag{6}$$

À l'aide des équivalences (2) qui précèdent, et des équivalences propositionnelles, on montre facilement que toute formule de la logique du premier ordre est équivalente à une formule qui n'utilise que $\exists, \wedge, \neg$, ou encore $\forall, \to, \bot$. On peut donc se restreindre à l'un des ces ensembles de quantificateurs et connecteurs pour démontrer des propriétés sémantiques des formules.

4.2.1 D'autres quantificateurs.

On ne peut pas vraiment parler de système complet pour la logique du premier ordre. Par exemple des quantifications comme «Il existe une infinité de x tels que F » ne s'expriment pas dans le langage du premier ordre.

Par contre il est bien connu que l'on peut exprimer les quantificateurs « il existe au plus un x tel que F » et « il existe un unique x tel que F » en calcul des prédicats ave égalité :

$$!x\ F \equiv_d \forall y \forall y' (F[y/x] \rightarrow F[y'/x] \rightarrow y = y')\ .$$

$$\exists !x\ F \equiv_d \exists x\ F \wedge !\, x\ F\ .$$

On montre facilement que :

$$\exists !x\ F \equiv \exists x (F \wedge \forall y (F[y/x] \rightarrow y = x))\ .$$
$$\neg !x\ F \equiv \exists y \exists y' (F[y/x] \wedge F[y'/x] \wedge y \neq y')\ .$$
$$\neg \exists !x\ F \equiv \forall x \neg F \vee \exists y \exists y' (F[y/x] \wedge F[y'/x] \wedge y \neq y')\ .$$

Exercice 2 Exprimer en calcul des prédicats égalitaire « il existe au plus un couple (x, y) tel que F », « il existe un unique couple (x, y) tel que F ». Généraliser aux n-uplets.

5 Un exemple simple d'élimination des quantificateurs.

On donne ici un exemple simple d'utilisation de ce qui précède. Il est courant en mathématiques de chercher un équivalent sans quantificateurs d'une formule quantifiée. Par exemple, on apprend au lycée que l'équation du second degré a une solution réelle si et seulement si son déterminant est non nul, ce qui signifie que dans $\mathbb{R}$, la formule $a \neq 0 \wedge \exists x\ ax^2 + bx + c = 0$ équivaut à $a \neq 0 \wedge b^2 - 4ac \geqslant 0$. On a *éliminé* le quantificateur existentiel.

Dans certaines théories, il est possible de montrer que *toute* formule est équivalente à une formule sans quantificateurs, on dit alors qu'une telle théorie admet *l'élimination des quantificateurs.* Une telle théorie sera donc beaucoup plus simple à étudier.

Un des exemples les plus simples d'un telle théorie est la théorie des *ordres totaux denses sans extrémités.* On exprime cette théorie dans le langage de signature $(<)$ (ordre strict). Les axiomes sont :

anti-réflexivité $\forall x \neg x < x$;

transitivité $\forall x \forall y \forall z (x < y \to y < z \to x < z)$;

totalité $\forall x \forall y (x < y \vee x = y \vee y < x)$;

densité $\forall x \forall y (x < y \to \exists a (x < a \wedge a < y)$;

sans majorant $\forall x \exists y\ x < y$;

sans minorant $\forall x \exists y\ y < x$.

Appelons T cette théorie dans le reste du paragraphe.

On va utiliser que toute formule s'écrit avec seulement le quantificateur existentiel, la disjonction, la conjonction, et la négation, ce qui a déjà été remarqué à la section précédente, se démontre formellement par une induction sur les formules utilisant des équivalences du paragraphe précédent. On utilisera tout de même la constante pour le faux, $\bot$ (elle équivaut à $x < x$), et celle pour le vrai, $\top \equiv_{def} \neg\bot$. Comme $\bot$ équivaut à $x < x$ et $\top$ à $x = x$ (x variable quelconque), cela ne contredit pas ce qui précède.

5.1 Simplifcation des formules propositionnelles

Une première remarque est que, en présence de l'axiome de totalité de l'ordre, il est possible d'éliminer la négation sur les formules atomiques.

Lemme 5.1 *Si la théorie T contient l'axiome de totalité, pour toutes variables x et y :*

$$\neg x < y \equiv_T y < x \vee y = x$$
$$\neg x = y \equiv_T x < y \vee y < x$$

Démonstration. C'est une conséquence immédiate de la totalité. ■

Lemme 5.2 *Une formule propositionnelle (non nécessairement close) de la théorie des ordres totaux est équivalente à une formule n'utilisant d'autres connecteurs que $\wedge$ et $\vee$ (pas de négation), et pour seules formules atomiques des égalités et des inégalités strictes, $\top$ et $\bot$.*

Démonstration. On montre le résultat conjointement pour une formule propositionnelle quelconque et sa négation, par induction sur les formules (en plus des formules atomiques, il suffit de traiter $\neg$, $\wedge$, $\vee$). La démonstration par induction consiste à démontrer le résultat pour une formule, en supposant le résultat pour les formules la constituant. Cette hypothèse est appelée hypothèse d'induction. Dans le cas des formules atomiques,qui n'ont pas de composants, l'hypothèse d'induction est vide. Un tel raisonnement est similaire au raisonneent par récurrence.

Formules atomiques. Lemme précédent.

Négation On suppose le résultat pour A : A et $\neg A$ s'écrivent avec $\wedge$, $\vee$ des égalités, et des inégalités strictes. On montre le résultat pour $\neg A$: c'est directement l'hypothèse d'induction pour $\neg A$, pour $\neg\neg A$, on a $\neg\neg A \equiv A$.

Conjonction Ici on suppose le résultat pour A et B. On en déduit directement que $(A \wedge B)$ a un équivalent de la forme voulue. On le déduit également pour $\neg(A \wedge B)$ car $\neg(A \wedge B) \equiv \neg A \vee \neg B$.

Disjonction $\neg(A \vee B) \equiv \neg A \wedge \neg B$ et hypothèse d'induction pour A et B.

Il suffit de traiter ces connecteurs car tous les connecteurs car $\wedge$, $\vee$ et $\neg$ forment un système complet de connecteurs. ■

On aura besoin de mettre les formules propositionnelles sans négation sous une forme particulière, dite *forme normale disjonctive*, qui est une disjonction de conjonctions de formules atomiques, et ne possède pas non plus de négation (dans le cas général une forme normale disjonctive peut utiliser des négations devant les formules atomiques). Le lemme qui suit est vrai quel que soit le langage, et se démontre en logique pure (il ne dépend pas d'une théorie particulière).

Lemme 5.3 *Une formule propositionnelle F (non nécessairement close) n'utilisant d'autres connecteurs que $\wedge$ et $\vee$ (pas de négation), $\top$, $\bot$, est équivalente à une formule utilisant les mêmes formules atomiques et qui est une disjonction de conjonctions de ces formules atomiques. C'est-à-dire que si $\mathcal{P}$ est l'ensemble nécessairement fini des formules atomiques de F, alors :*

$$F \equiv \bigvee_{i=1}^{p} \bigwedge_{j=1}^{q_i} p_{ij} \quad \textit{avec pour tous } i \leqslant p,\ j \leqslant q_i\ p_{ij} \in \mathcal{P} \cup \{\bot, \top\}\ .$$

Démonstration. Le résultat se démontre par induction pour les formules propositionnelles écrites avec $\wedge$ et $\vee$. Il est évident pour les formules atomiques, $\bot$ et $\top$ (cas particuliers où la disjontion est réduite à une conjonction elle même réduite à un atome).Il se déduit directement de l'hypothèse d'induction pour le $\vee$. Dans le cas du $\wedge$ il se déduit de la distributivité du $\wedge$ sur le $\vee$. ■

5.2 Élimination d'un existentiel

Le lemme clef pour l'élimination des quantificateurs est le suivant :

Lemme 5.4 (lemme clef) *Soit F une formule propositionnelle du langage de signature $(<)$ utilisant pour seul connecteur $\wedge$, et dont les variables sont parmi $x_0, x_1, \ldots, x_n$. Alors il existe une formule propositionnelle F' du même langage, utilisant pour seul connecteur $\wedge$ dont les variables libres sont parmi $x_1, \ldots, x_n$ et telle que dans la théorie T des ordres denses sans extrémités :*

$$\exists x_0\ F \equiv_T F'\ .$$

Démonstration. L'idée de la preuve est simple, et peut se résumer ainsi : une fois les conditions ne concernant pas x_0 réalisées, et qui portent sur des variables $x_1, \ldots, x_n$, si la formule n'est pas contradictoire, il est toujours possible de trouver un tel x_0 qui doit satisfaire certaines relations d'ordre vis à vis des $x_1, \ldots, x_n$, en vertu des axiomes de densité, d'absence de majorant et de minorant.

Détaillons tout d'abord la construction de la formule F'.

- Il y a un nombre fini de formules $x_i < x_j$, $x_i = x_j$ possibles, on peut donc définir la formule F_0 obtenue, tout d'abord en ajoutant à F toutes les formules $x_i < x_j$ et $x_i = x_j$ qui sont conséquence de F dans la théorie T (on omet les égalité $x_i = x_i$, qui sont forcément réalisées) :

$$F_0 = \left(\bigwedge_{\substack{i,j=0 \\ F \vdash_T x_i < x_j}}^{n} x_i < x_j \right) \wedge \left(\bigwedge_{\substack{0 \leqslant i < j \leqslant n \\ F \vdash_T x_i = x_j}} x_i = x_j \right)\ .$$

  En particulier F_0 contient chaque formule atomique qui apparaît dans F et donc $F \equiv F_0$. Si F_0 contient une formule $x_i < x_i$, les relations induites par les égalités et inégalités de la formule F sur $x_1, \ldots, x_n$ ne sont pas réalisables parce qu'il y a un cycle (anti-réflexivité). La formule F est équivalente à $\bot$, la formule $\exists x_0\ F$ également.
- Si la variable x_0 apparait dans une égalité de F_0 (et donc de F) : $F \vdash x_0 = x_i$, alors la formule $\exists x_0\ F$ est équivalente à F' obtenue à partir de F_0 en supprimant toutes les formules où x_0 apparait (on pourrait également penser remplaçer dans F toutes les occurrences de x_0 par x_i, mais c'est « déjà fait » dans F_0).
- Dans tous les autres cas la formule F est équivalente à la formule F' obtenue en supprimant dans F_0 toutes les formules atomiques où x_0 apparait. On prend $F' = \top$ si ceci supprime toutes les formules. On utilise pour cela soit l'absence de minorant si x_0 apparait toujours à droite du signe $<$, soit l'absence de majorant si x_0 apparait toujours à gauche du signe $<$, soit la densité si x_0 apparait à la fois à droite et à gauche du signe $<$.

Voici quelques cas particuliers.

$\exists y(x_1 < y \wedge y < x_2 \wedge x_1 = x_2) \equiv_T \bot$	cycle
$\exists y(x_1 < y \wedge x_3 < x_2 \wedge y = x_3) \equiv_T x_1 < x_3 \wedge x_3 < x_2$	égalité
$\exists y(x_1 < y \wedge y < x_2 \wedge x_1 = x_3) \equiv_T x_1 < x_2 \wedge x_3 < x_2 \wedge x_1 = x_3$	densité
$\exists y(x_1 < y \wedge x_2 < y) \equiv_T \top$	pas de majorant
$\exists y(y < x_1 \wedge x_3 < x_2 \wedge x_1 = x_3) \equiv_T x_1 < x_2 \wedge x_3 < x_2 \wedge x_1 = x_3$	pas de minorant

Détaillons le cas où x_0 apparait à droite et à gauche du signe $<$ dans F_0, mais pas dans une égalité, et où F_0 ne contient pas de formule du type $x_i < x_i$. Le fait que $\exists x_0\ F$ a pour conséquence $\exists x_0\ F_0$ (on n'a fait qu'ajouter des conséquences de F à la conjonction), donc $\exists x_0\ F'$ (on supprime certaines des formules de la conjonction), donc F' (x_0 n'apparait pas dans F') est immédiat.

Pour la réciproque : soit $\mathcal{M}$ une structure modèle de $\mathcal{T}$, et $m_1, \ldots, m_p$ des éléments de $\mathcal{M}$ tels que $\mathcal{M} \models F'(m_1, \ldots, m_p)$. Il suffit de trouver $m_0 \in \mathcal{M}$, tel que $\mathcal{M} \models F_0(m_0, m_1, \ldots, m_p)$. Soit m_{i_0} le plus grand des m_i tel que $x_i < x_0$ apparait dans F, et m_{i_1} le plus petit des m_i tel que $x_0 < x_i$ apparait dans F_0. Comme les formules $x_{i_0} < x_0$ et $x_0 < x_{i_1}$, apparaissent dans F, par construction de F' (la transitivité est un axiome de T), la formule $x_{i_0} < x_{i_1}$ apparait dans la conjonction F', donc $\mathcal{M} \models m_{i_0} < m_{i_1}$. Par densité ($\mathcal{M} \models \mathcal{T}$), il existe m_0 tel que $\mathcal{M} \models m_{i_0} < m_0 < m_{i_1}$. Par construction $\mathcal{M} \models F_0(m_0, m_1, \ldots, m_p)$, donc $\mathcal{M} \models F(m_0, m_1, \ldots, m_p)$, donc $\mathcal{M} \models \exists x_0\ F(x_0, m_1, \ldots, m_p)$.

Les deux autres cas, x_0 n'apparait qu'à droite de $<$, et x_0 n'apparait qu'à gauche de $<$ sont analogues, en utilisant l'absence de minorant et l'absence de majorant. ■

5.3 Conclusion

Proposition 5.5 *Une formule du premier ordre du langage de $(<)$ est équivalente dans la théorie T des ordres denses sans extrémités à une formule sans quantificateurs dont les variables libres sont parmi les variables libres de F.*

Démonstration. Par induction sur les formules. On suppose que celles-ci ne contient pas de quantificateur universel. En effet, on peut les éliminer par l'équivalence :

$$\forall x\ F \equiv \neg\exists x \neg F\ .$$

Pour les formules atomiques c'est évident. Si la formule est une négation, une conjonction ou une disjonction, c'est une conséquence immédiate de l'hypothèse d'induction.

Dans le cas du quantificateur existentiel, on se ramène par hypothèse d'induction à une quantification existentielle sur une formule propositionnelle, avec pour seuls connecteurs $\wedge$ et $\vee$ (et pas de négation) d'après le lemme 5.2. Cette formule est équivalente à une disjonction de conjonctions de formules atomiques d'après le lemme 5.3. On sait que le quantificateur existentiel commute avec la disjonction :

$$\exists x \bigvee_{i=1}^{p} C_i \equiv \bigvee_{i=1}^{p} \exists x\ C_i.$$

Pour chacune des C_i conjonction d'égalités et d'inégalités strictes, le quantificateur existentiel s'élimine par le lemme clef 5.4. ■

Corollaire 5.6 *Pour toute formule close du premier ordre F dans le langage $(<)$,*

$$\vdash_T F \text{ ou } \vdash_T \neg F\ .$$

Démonstration. En effet d'après le lemme précédent F est équivalente à une formule sans quantificateurs ni variables libres, qui ne peut donc être que $\top$ ou $\bot$. ■

Une telle théorie, c'est à dire une théorie T vérifiant pour toute formule F que $\vdash_T F$ ou $\vdash_T \neg F$, est dite *complète*.

Les modèles d'une théorie complète T vérifient les mêmes formules closes du premier ordre : celles qui sont démontrables dans la théorie T. Par exemple $(\mathbb{Q}, <)$ et $(\mathbb{R}, <)$ satisfont les mêmes formules closes (remarquez que ces deux structures ne sont pas isomorphes, puisque les ensembles de base n'ont pas même cardinalité).

Une conséquence simple de l'élimination des quantificateurs est la suivante. Comme $(\mathbb{Q}, <)$ est une sous-structure de $(\mathbb{R}, <)$ et que toute formule même non close, est équivalente à une formule sans quantificateurs, une formule close du langage de la théorie des ordres étendue avec tous les éléments de $\mathbb{Q}$ comme symboles de constantes est vraie dans $\mathbb{R}$ si et seulement si elle est vraie dans $\mathbb{R}$.

Il existe des théories qui ne sont évidemment pas complètes, comme la théorie des ordres, la théorie des groupes etc.

Il existe des théories qui ont un modèle « attendu » et dont on pourrait souhaiter qu'elles soient complètes, mais qui ne le sont pas. C'est le cas par exemple de l'arithmétique de Peano. Cela signifie par exemple que l'arithmétique de Peano a des modèles qui ne vérifient pas les mêmes formules que $\mathbb{N}$. C'est une conséquence du premier théorème d'incomplétude de Gödel. En fait, d'après le premier théorème d'incomplétude de Gödel, il n'y a pas de théorie « que l'on puisse énoncer de façon raisonnable » (en un sens que l'on sait préciser), qui soit cohérente, qui permette de développer « suffisament » d'arithmétique (comme l'arithmétique de Peano ou la théorie des ensembles), et qui soit complète.

Index

Table des matières