Sémantique compositionnelle

Cours de Licence de Sciences du Langage (L3)
Alain Lecomte – Professeur, Université Paris 8

Leçon 1 – Introduction à la logique des prédicats du premier ordre

1 Langages prédicatifs

1.1 La notion de langage (au sens formel)

1.1.1 Les origines : l'usage des notions de fonction et de variable

Traduire des phrases d'une langue ordinaire dans un langage formel sert principalement à exprimer avec rigueur une signification possible de telles phrases. Exprimer la signification d'une phrase dans la langue ordinaire nous conduirait en effet dans une régression infinie : qu'est-ce qui nous donnerait la signification de cette signification ? Au contraire, quand nous l'exprimerons dans un langage formel, nous pourrons définir exactement, grâce aux règles explicites de ce langage, ce que nous entendons par la signification à donner à une phrase, d'autant que nous aurons une procédure précise pour déterminer sa valeur de vérité dans une situation donnée (« situation » que nous nommerons par la suite « modèle »).

De plus, un langage formel sera défini comme non ambigu : étant donnée une expression de ce langage, elle ne peut avoir qu'une seule lecture possible, contrairement à certaines phrases de notre langue ordinaire comme « tout étudiant doit rencontrer un professeur ».

Comme dit en cours, les notions que nous utiliserons dans ce cours proviennent essentiellement d'une conception théorique remontant au logicien philosophe allemand Gottlob FREGE (1848 – 1925). Celui-ci a eu l'idée d'utiliser des concepts mathématiques, comme celui de *fonction*, afin de transformer la logique en une discipline rigoureuse.

On peut par exemple traduire une phrase comme *Pierre est malade* en l'application d'une fonction f à un objet (ou argument) p. La fonction, dans ce cas, serait une fonction f de l'ensemble des individus (*Pierre, Marie, Luc, Ali, Fatima, Laura…*) dans l'ensemble réduit à deux valeurs de vérité : 0 et 1. Cette fonction f serait telle que, si x est un individu de l'ensemble, $f(x)$ serait égal à 1 (= Vrai) si, dans la situation considérée, il est vrai que l'individu x est malade, et égal à 0 (= Faux) sinon.
On peut étendre ce raisonnement à des phrases plus complexes, par exemple possédant un sujet et un objet (direct ou indirect), comme *Pierre parle à Marie*. Dans ce cas, c'est le « verbe » *parle_à* qui se traduira par une fonction. Cette fonction sera dite *à deux arguments* (ou à deux variables) parce qu'il faut deux individus pour obtenir une phrase, à partir d'un tel verbe. Soit f cette fonction, elle est de l'ensemble des couples d'individus (*(Pierre, Pierre), (Pierre, Marie), (Pierre, Luc), …, (Marie, Pierre), (Marie, Marie), (Marie, Luc)* etc.) dans l'ensemble $\{0, 1\}$. Elle est telle que, étant donné un couple (x, y), $f(x , y)$ est égal à 1 si, dans la situation considérée il est vrai que x parle à y et égal à 0 sinon.

Dans de telles représentations, on notera la présence de deux types d'expressions : celles qui s'interprètent comme des entités bien identifiées, par exemple : *Pierre, Luc, Marie* etc. ou bien des phrases comme *Pierre est malade*, ou bien *Pierre parle à Marie*, et celles à qui il manque quelque chose pour qu'elles deviennent de telles entités (qui ont donc une sorte de « trou pour un argument »), par exemple les propriétés (*être malade*) ou bien les verbes

(*regarder, aimer, parler_à*). Si *Pierre* dénote un individu particulier, si *Pierre est malade* « dénote » une valeur de vérité particulière, en revanche *est malade* tout seul ne dénote rien de tel, car il lui manque quelque chose (un sujet) pour devenir une phrase.
Frege qualifiait de *saturée* une expression de la première espèce, et de *non saturée* une expression de la deuxième.

Les expressions non saturées sont des *fonctions*. Les expressions saturées sont les éléments d'un ensemble de base (le domaine des individus) ou bien les éléments de {0, 1} (c'est-à-dire les valeurs de vérité).

On voit alors tout de suite qu'il y a plusieurs manières de passer d'une expression non saturée à une expression saturée.

La première manière consiste simplement à remplir les trous par des noms d'individu, comme nous venons de le faire, obtenant *Pierre est malade* ou *Pierre regarde Marie*, qu'on représente par **malade(Pierre)** et par **regarde(Pierre, Marie)**.
Mais on pourrait aussi remplir ces trous par des *variables*, obtenant par exemple **malade**(x), **regarde(Pierre**, y), **regarde**(x, **Marie**) ou **regarde**(x, y). Quel serait le « sens » intuitif de telles expressions ?
Si **malade(Pierre**) possède une valeur de vérité bien définie dans une situation donnée (où l'on n'ignore rien de Pierre et de son état de santé !), en revanche **malade**(x) ne possède pas de telle valeur de vérité bien définie car tout dépend de la valeur qu'on donne à x ! Bien sûr, si x = **Pierre**, **malade**(x) est vrai si Pierre est malade, mais si x = **Marie** et que Marie n'est pas malade, **malade**(x) est faux. Des expressions dont les trous sont remplis par des variables se comportent donc encore exactement comme les expressions non saturées dont nous sommes partis. Ce n'est pas étonnant : *une variable sert seulement à indiquer un « trou » dans une expression*, on utilise plusieurs variables parce que souvent une expression peut posséder plusieurs trous et qu'on ne souhaite pas les identifier. Par exemple la notation **regarde**(x, y) indique qu'il y a deux individus à trouver pour obtenir une proposition complète, alors que **regarde**(x, x) signifierait qu'il n'y en a qu'un à trouver, qu'on mettrait aux deux places indiquées par un x : on ne pourrait obtenir que des phrases du genre *Pierre regarde Pierre*, autrement dit *Pierre se regarde* (réflexif). On peut remarquer que, concernant la langue ordinaire, nous avons une situation similaire avec *les pronoms*. Une phrase comme *il regarde Marie*, donnée hors contexte, possède une valeur de vérité qui varie en fonction du référent qu'on donne à *il*.

1.1.2 Quantificateurs et variables liées

Il y a une deuxième manière de passer à une expression saturée. C'est ce qui se produit lorsque nous analysons des phrases comme *Pierre parle à une fille* ou bien *tout garçon regarde Marie*. Nul doute que ces phrases sont tout aussi saturées que la phrase *Pierre parle à Marie* : elles ont, elles aussi, une valeur de vérité bien déterminée dans une situation donnée.
Il y a pourtant une différence : « une fille » est indéterminé, on ne peut pas représenter cette expression par un nom de fille particulier. De même, « tout garçon » n'est pas représenté par un garçon particulier, dont on mettrait le nom à la place de la variable y. Une phrase comme *Pierre parle à une fille* s'interprète plutôt comme « il y a une fille à laquelle Pierre parle » (avec une signification existentielle attachée à « il y a ») et une phrase comme *Tout garçon regarde Marie* s'interprète comme « quel que soit le garçon, il regarde Marie ».

Afin de représenter ces faits dans son langage formel, Frege introduit la notion de *quantificateur. Un quantificateur est attaché à une variable, qui lui est associée dans la* *formule* (par exemple on aura un quantificateur (∀x) ou un quantificateur (∃y)) et il exerce une action à distance sur toutes les apparitions de cette variable dans la formule, pourvu que celles-ci figurent bien dans *sa portée* (nous donnerons une définition précise de ces notions plus loin). On dit alors que toutes les occurrences de la variable qui figurent dans la portée du quantificateur sont *liées*, et que les autres (celles qui ne figurent dans la portée d'aucun quantificateur) sont *libres*. Ainsi une phrase comme *Pierre parle à une fille* sera représentée par la formule :

$$(\exists x)(parle\_à(pierre, x) \wedge fille(x))$$

Si nous voulons maintenant exprimer par une formule la signification de la phrase *Tout garçon regarde Marie*, nous devons bien faire attention au fait que nous ne pouvons pas simplement prendre une formule analogue à la précédente et remplacer (∃x) par (∀x) ! car la formule :

$$(\forall x)(regarde(x, marie) \wedge garçon(x))$$

ne possède pas la même valeur de vérité que la phrase que nous voulons représenter ! En effet, cette formule est vraie si et seulement si tous les individus regardent Marie **et** sont des garçons ! (ce qui entraînerait du coup que, Marie étant un individu… elle serait elle aussi un garçon !). En réalité, le fait de regarder Marie est concentré sur les garçons en tant que sous-ensemble de l'ensemble des individus, il est donc *subordonné au fait d'être un garçon*. Dit autrement : être un garçon est une *condition suffisante* pour regarder Marie ! D'où la formule :

$$(\forall x)(garçon(x) \Rightarrow regarde(x, marie))$$

Intuitivement, cette formule est vraie si et seulement si, étant donné n'importe quel individu x, s'il est un garçon, alors il regarde Marie (mais rien n'est dit concernant les filles).
Dans ces deux cas, les deux occurrences de la seule et unique variable apparaissant dans l'expression sont *liées*. On dit que la formule est *close*.

1.2 Syntaxe d'un langage formel : le langage de la logique des prédicats du premier ordre

Dans les exemples précédents, nous avons traduit des phrases de la langue en des expressions (formules) utilisant des signes qui n'appartiennent pas à la langue, comme $\forall, \exists, \wedge, \vee$. Ces formules appartiennent à un système formel, ou langage formel. Qu'est-ce qu'un langage formel, et en quoi se distingue-t-il d'une langue (naturelle) ? Un langage formel est défini par un ensemble de règles *a priori*, à partir d'un stock de symboles également donné *a priori*. Ainsi notre langage formel actuel contient-il pour l'instant :

1.2.1 Symboles et règles syntaxiques

Symboles (vocabulaire) :

- des symboles qui servent à désigner (*dénoter*) des objets (ou des *individus*)
- ces symboles se subdivisent en deux catégories : les **constantes individuelles** (Pierre, …) et les **variables individuelles** : x, y, z, …
- des symboles qu'on supposera être uniquement des **constantes**, qui servent à désigner des propriétés (« être norvégienne »…) ou des relations (« suivre » en parlant d'un étudiant et d'un cours, « épouser » en parlant d'un individu et d'un autre, mais on pourrait avoir aussi « être assis entre… et … » en parlant d'étudiants dans la salle de cours), et qu'on appelle des symboles de prédicats, ou plus brièvement **prédicats**,

- comme indiqué dans les exemples donnés entre parenthèses, ces prédicats peuvent représenter des *propriétés d'individus*, auquel cas ils sont dits *unaires* (ou d'arité 1), ou bien représenter des *relations* entre *deux* entités, auquel cas ils sont dits *binaires* (ou d'arité 2), ou bien représenter des relations entre *trois* entités, auquel cas ils sont dits *ternaires* (ou d'arité 3), etc.
- des symboles logiques qui se répartissent en deux catégories :
  - les connecteurs : $\neg$, $\wedge$, $\vee$, $\Leftrightarrow$, $\Rightarrow$
  - les quantificateurs : $\forall$, $\exists$
- des symboles de ponctuation : (,) et éventuellement d'autres formes de parenthèses

Règles de formation :

- on appellera termes les symboles de constante et de variable individuelles
- si $t_1, \ldots, t_n$ sont des termes et si A est un symbole de prédicat d'arité n, alors $A(t_1, \ldots, t_n)$ est une formule atomique,
- si A et B sont des formules, alors $(A \vee B)$, $(A \wedge B)$, $(A \Rightarrow B)$, $(A \Leftrightarrow B)$ sont des formules,
- si A est une formule, $(\neg A)$ est une formule,
- si ξ est une variable individuelle et si A est une formule, alors $(\forall\xi)A$ et $(\exists\xi)$ A sont des formules

Un tel langage formel, basé sur la logique des prédicats, est appelé **langage prédicatif**. Un langage prédicatif particulier sera associé à tout ensemble de phrases étudiées. Il comportera exactement les constantes individuelles et les symboles de prédicat dont on aura besoin pour traduire les phrases.

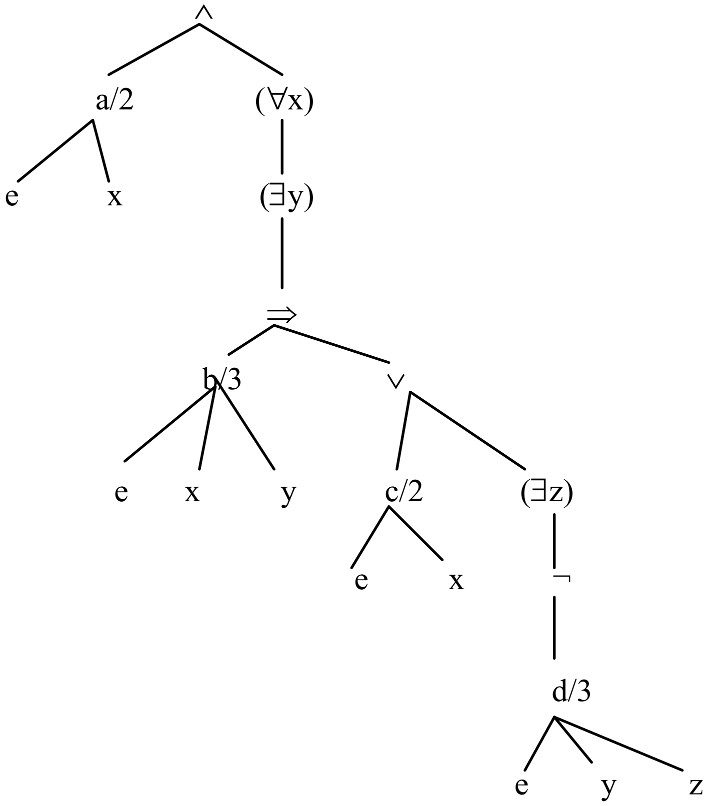
On voit bien en quoi un tel langage diffère d'une langue : un langage formel est artificiel, son vocabulaire et ses règles de syntaxe sont définis a priori, à la différence évidemment d'une langue naturelle dont on ne finit jamais de découvrir les règles…

1.2.2 Arbre des sous-formules

Grâce à cette définition récursive, on peut associer à toute formule *un arbre*, qu'on appelle *arbre des sous-formules*. Par exemple, soit la formule: (où e et f sont supposées être des constantes individuelles)

(15) $(\mathbf{A}/_2(e, x) \wedge (\forall x)(\exists y)(\mathbf{B}/_3(e, x, y) \Rightarrow (\mathbf{C}/_2(e, x) \vee (\exists z) \neg\mathbf{D}/_3(e, y, z))))$

on peut la représenter par l'arbre:



Les nœuds d'un tel arbre peuvent être numérotés de la façon suivante: le nœud racine est étiqueté par la suite vide < > et chaque noeud reçoit pour étiquette une suite de n chiffres de sorte que la suite des n-1 premiers indique le noeud dont il descend et le dernier son rang dans l'ordre gauche-droite parmi tous ceux qui ont le même noeud-père. Ainsi les nœuds de l'arbre de la figure ci-dessus reçoivent comme numéros successivement :
< >, 1, 2, 11, 12, 21, 211, 2111, 2112, 21111, 21112, 21113 etc. On constate que le premier e reçoit l'étiquette: 11, le deuxième : 21111 et le troisième : 211211. On dira que e possède trois *occurrences* dans cette formule, désignées respectivement par: 11, 21111 et 211211.

1.2.3 Portée d'un quantificateur

Définition [portée d'un quantificateur]: étant donnée une formule A contenant un quantificateur (Qξ), où ξ est une variable, on appelle **portée** de ce quantificateur toute la sous-formule dominée par (Qξ) dans l'arbre des sous-formules de A.

Occurrence d'une variable : toute formule peut donner lieu à une analyse arborescente. On peut alors repérer chaque noeud de l'arbre par un numéro d'ordre (cf. procédure ci-dessus) qu'on appelle aussi son adresse. Une occurrence d'une variable est l'adresse où elle apparaît.

Occurrence de variable liée: une occurrence de variable ζ sera dite *liée* si elle est dans la portée d'un quantificateur (∀ζ) ou (∃ζ).

Occurrence de variable libre: une occurrence de variable ζ est dite *libre* dans le cas contraire.

Exemple:
Dans la formule :

$$(16)\quad (\mathbf{a}/2(e, x) \wedge (\forall x)(\exists y)(\mathbf{b}/3(e, x, y) \Rightarrow (\mathbf{c}/2(e, x) \vee (\exists z)\ \neg\mathbf{d}/3(e, y, z))))$$

la première occurrence de x est libre, par contre les deux suivantes sont liées. Toutes les occurrences de z et de y sont liées.

Remarque importante: il est recommandé lorsque une même variable occurre dans une formule tantôt libre tantôt liée de changer le nom soit de la variable libre, soit de la variable liée. La vérité d'une formule ne dépend pas du nom qu'on donne aux variables. Ainsi la formule précédente pourra s'écrire aussi bien :

$$(17)\quad (\mathbf{a}/2(e, u) \wedge (\forall x)(\exists y)(\mathbf{b}/3(e, x, y) \Rightarrow (\mathbf{c}/2(e, x) \vee (\exists z)\ \neg\mathbf{d}/3(e, y, z))))$$

ou:

$$(18)\quad (\mathbf{a}/2(e, x) \wedge (\forall u)(\exists y)(\mathbf{b}/3(e, u, y) \Rightarrow (\mathbf{c}/2(e, u) \vee (\exists z)\ \neg\mathbf{d}/3(e, y, z))))$$

Un tel changement de nom de variable a l'avantage de nous faire éviter toute confusion et de définir pour toute formule F deux ensembles disjoints:

Libre (F) = ensemble des variables libres dans F,
Liée (F) = ensemble des variables liées dans F.

Définition : Une formule sera dite *close* si toutes les occurrences de variables qui y figurent sont liées. On parlera aussi dans ce cas de *proposition*.

Comme on le notera, ce n'est pas le cas de la formule (17) ci-dessus. En effet, dans (17), u est libre. (17) n'est donc pas une proposition (au sens où nous entendons par là une expression qui est soit vraie, soit fausse) car sa valeur de vérité dépendra de la valeur donnée à u.

1.2.4 Substitution d'un terme à une variable dans une expression prédicative

Etant donnée une formule prédicative ϕ contenant une variable libre ζ, et un terme τ, nous définirons le résultat de la *substitution* de τ à ζ dans ϕ comme l'expression obtenue quand toutes les occurrences de ζ ont été remplacées par des occurrences de τ. On notera $\phi[\tau/\zeta]$ le résultat de la substitution de τ à ζ dans ϕ. On peut écrire $\phi(\tau)$ à la place de $\phi[\tau/\zeta]$, à condition de se souvenir que les places occupées maintenant par τ étaient celles repérées par ζ. Cette précaution est évidemment fondamentale si ϕ est une expression contenant d'autres variables que ζ.
Exemples:
Considérons la formule:

$$(19)\quad (\mathbf{a}/2(e, u) \wedge (\forall x)(\mathbf{b}/3(e, x, y) \Rightarrow (\mathbf{c}/2(e, x) \vee (\exists z)\ \neg\mathbf{d}/3(e, y, z))))$$

qui possède les variables libres u et y. Le résultat de la substitution de la constante k à y dans cette formule (opération que nous noterons: [k/y]) sera:

$$(20)\quad (\mathbf{a}/2(e, u) \wedge (\forall x)(\mathbf{b}/3(e, x, k) \Rightarrow (\mathbf{c}/2(e, x) \vee (\exists z)\ \neg\mathbf{d}/3(e, k, z))))$$

Le résultat de la substitution de la variable v à y dans cette formule sera:

$$(21)\quad (\mathbf{a}/2(e, u) \wedge (\forall x)(\mathbf{b}/3(e, x, v) \Rightarrow (\mathbf{c}/2(e, x) \vee (\exists z)\ \neg\mathbf{d}/3(e, v, z))))$$

De même le résultat de la substitution de x à y dans cette formule serait:

(22) $(\mathbf{a}/_2(e, u) \wedge (\forall x)(\mathbf{b}/_3(e, x, x) \Rightarrow (\mathbf{c}/_2(e, x) \vee (\exists z) \neg \mathbf{d}/_3(e, x, z))))$

mais comme on peut le voir aisément, il y a une différence "structurelle" entre (21) et (22). Dans (21), la substitution de v à y *préserve les liens* de la formule: il n'y a pas de nouveau lien créé. Ce n'est pas le cas de (22) où par substitution de x à y, parce que les occurrences de y sont dans le champ d'un quantificateur portant sur x, immédiatement de nouveaux liens sont créés, qui ne figuraient pas dans la formule originale. Il y a là quelque chose d'illicite, qui se comprend bien si nous considérons seulement le deuxième conjoint de la formule: $(\forall x)(\mathbf{b}/_3(e, x, y) \Rightarrow (\mathbf{c}/_2(e, x) \vee (\exists z) \neg \mathbf{d}/_3(e, y, z)))$. En effet, la formule originale serait une fonction propositionnelle alors que la formule à laquelle on arriverait par simple substitution serait une formule close (une proposition). Nous éviterons ce type de situation en restreignant les substitutions d'un terme τ à une variable ζ aux cas où **τ est libre pour ζ**.

Définition : dans une expression $\phi(\zeta)$ (où x est une variable libre) on dira que le terme τ est *libre pour* ζ si la substitution de τ à ζ dans $\phi(\zeta)$ ne crée aucun lien supplémentaire par un quantificateur.

Exemple: dans '$(\exists y)$ $(x < y)$', z est libre pour x, mais y n'est pas libre pour x, car la substitution de y à x conduirait à: '$(\exists y)$ $(y < y)$'. (Or, une théorie qui admettrait la première formule pour vraie de tout x par exemple, n'admettrait pas nécessairement la seconde pour vraie: voir ce qui se passe quand on interprète le signe "<" comme la relation d'ordre strict usuelle sur les réels). Dans '$(\exists y)$ $(x < y)$', il y a un seul lien: entre le quantificateur $(\exists y)$ et l'unique occurrence de la variable y. Dans '$(\exists y)$ $(y < y)$', il y a deux liens: entre le quantificateur et les deux occurrences de y.

Exercice:
soit le langage prédicatif suivant:

- var. individuelles : x, y, z
- const. individuelles : p, j, m
- prédicats : **c**/2, **e**/2, **f**/1

Montrer qu'on peut former les formules suivantes :

- $\mathbf{f}(j)$
- $\mathbf{f}(x)$
- $(\exists z)\mathbf{f}(z)$
- $(\exists z)(\forall x)(\forall y)((\mathbf{c}(x,y) \wedge \mathbf{f}(y)) \Rightarrow \mathbf{e}(z,x))$,
- $\neg((\exists z)(\forall x)(\mathbf{f}(x) \Rightarrow \mathbf{c}(z, x)))$

Noter que dans cette liste, seule la deuxième formule a la particularité d'avoir une variable libre. Dans les autres formules, soit il n'y a pas de variable, soit elles sont toutes liées. Remarquer aussi que nos règles de formation n'interdisent pas une formule telle que :

- $(\forall x)(\mathbf{f}(y) \Rightarrow \mathbf{e}(j, y))$

où la variable y est libre et la variable x n'apparaît pas dans la portée du quantificateur $(\forall x)$. Cela s'interprétera plus tard comme une quantification vide. On pourrait aussi avoir par exemple :

- $(\forall x)$ $\mathbf{f}(j)$

Exercices

1. Traduire en logique des prédicats du premier ordre les phrases suivantes :
   - *Il existe un nombre plus petit que tous les autres*
   - *Il n'existe pas de nombre plus grand que tous les autres*

- *Tout nombre entier est tel qu'il en existe un strictement plus grand que lui*
- *Un étudiant a remis une copie à chaque professeur*
- *Tout étudiant a remis une copie à un professeur*
- *Tout volume cubique qui se trouve à gauche d'un cylindre est devant un tétraèdre*
- *Il y a quelqu'un qui est tel que toutes les personnes qui le connaissent l'apprécient*
- *Personne n'est tel que toute personne qui le connaît l'apprécie*

2. Faire l'analyse arborescente de la formule :

$$(1) \qquad (\forall x)(\forall z)(((\exists y)(A(x,y) \wedge A(y,z))) \Rightarrow B(x,z))$$

   Est-ce une formule close ?
3. Mêmes questions avec :

$$(2) \qquad (\forall x)(((\exists y)(A(x,y) \wedge A(y,z))) \Rightarrow B(x,z))$$

4. Dans la formule (1) ci-dessus, peut-on substituer la variable w à la variable z ? Peut-on substituer la variable y à la variable z ?
   Dans la formule ci-dessous :

$$(3)\ (\forall x)(\forall z)((A(x,y) \wedge A(y,z)) \Rightarrow (\exists y)B(y,x,z))$$

   Quelles sont les occurrences de variables libres, de variables liées ? Peut-on transformer cette formule de manière à ce que variables libres et variables liées forment deux ensembles bien séparés ?