# 2 Le calcul de prédicats

## 2.1 Syntaxe

**Les termes d'individus** Soit $\mathcal{V}_1$ un ensemble infini dénombrable de variables. Soit $\mathcal{C}_1$ un ensemble du *constantes fonctionnelles* dites aussi *symboles de fonction*. On note $f$, $f_1$, .., $f_n$, ..,$g$, .., $h$, etc.. les symboles de fonction. On définit de $\mathcal{C}_1$ dans $I\!N$ une fonction dite d'*arité* qui associe à chaque symbole de fonction le nombre de ses arguments. Un symbole d'arité nulle est appelé communément une *constante*. On définit alors l'ensemble $\mathcal{T}_1$ des *termes d'individus* ou, plus simplement, des *termes*, selon les clauses :

1. toute variable $x \in \mathcal{V}_1$ est un terme;
2. si $f \in \mathcal{C}_1$ est d'arité $n$ et si $t_1$, ..,$t_n$ sont des termes alors $f(t_1, .., t_n)$ est un terme.

On note, en général $t$, $t_1$, .., $t_n$, ..,$u$, $u_1$, ..,$v$, etc.. les termes.

On définit inductivement sur la forme de $t$ l'ensemble des ses variables que l'on note $\mathcal{V}(t)$ :

1. si $t = x \in \mathcal{V}_1$ alors $\mathcal{V}(t) = \{x\}$;
2. si $t = f(t-1, .., t_n)$ alors $\mathcal{V}(t) = \bigcup_{i=1}^{n} \mathcal{V}(t_i)$.

On définit sur les termes la *substitution* d'un terme aux occurrences d'une variable. On note $t[u/x]$ le terme obtenu en remplaçant dans $t$ toutes les occurences de la variable $x$ par le terme $u$. On définit par induction sur $t$ :

1. si $t = x$ alors $x[u/x] = u$;
2. si $t = y$, $y \in \mathcal{V}_1$ et $y \neq x$ alors $y[u/x] = y$;
3. si $t = f(t_1, .., t_n)$ alors $f(t_1, .., t_n)[u/x] = f(t_1[u/x], .., t_n[u/x])$.

**Les formules** Soit $\mathcal{P}_1$ un ensemble de *symboles de prédicats*. On note $P$, $P_1$, .., $Q$, ..,$R$ les symboles de prédicats. Comme les symboles de fonction, les symboles de prédicats sont munis d'une fonction d'arité. Aux connecteurs du calcul des propositions, on rajoute les deux symboles $\forall$ et $\exists$ que l'on appelle *quatificateurs*. Ils se prononcent respectivement ≪*pour tou(te)s*≫ et ≪*il existe*≫ . On définit alors $\mathcal{F}_1$ l'ensemble des formules du calcul des prédicats :

1. si $P \in \mathcal{P}_1$ est d'arité $n$ et si $t_1, .., t_n$ sont des termes alors $P(t_1, .., t_n)$ est une fromule – on parle dans ce cas de formule *atomique*;
2. si $A$ et $B$ sont des formules alors $\neg A$, $A \wedge B$, $A \vee B$ et $A \Rightarrow B$ sont des formules;
3. si $A$ est une formule et $x$ une variable alors $\forall x.A$ et $\exists x.A$ sont des formules.

On note les formules du calcul des prédicats comme celles du calcul des propositions.

Les quantificateurs font partie de cette catégorie de symboles qu'on appelle *lieurs*. Les lieurs s'appliquent aux variables apparaissant dans leur *portée* et permettent de définir les notions de variables *libres* et variables *liées* d'une formule; voire d'*occurence libre* et d'*occurence liée* d'une variable dans une formule. Définissons toutes ces notions. Soit $A$ une formule, on note $\mathcal{V}_f(A)$ l'ensemble de ses variables libres et $\mathcal{V}_b(A)$ l'ensemble de ses variables liées que l'on définit mutuellement par induction sur $A$ :

1. si $A = P(t_1, .., t_n)$ alors $\mathcal{V}_f(A) = \bigcup_{i=1}^{n} \mathcal{V}(t_i)$ et $\mathcal{V}_b(A) = \emptyset$;
2. si $A = \neg B$ alors $\mathcal{V}_f(A) = \mathcal{V}_f(B)$ et $\mathcal{V}_b(A) = \mathcal{V}_b(B)$
3. si $A = B \wedge C$ ou $A = B \vee C$ ou $A = B \Rightarrow C$ alors $\mathcal{V}_f(A) = \mathcal{V}_f(B) \cup \mathcal{V}_f(C)$ et $\mathcal{V}_b(A) = \mathcal{V}_b(B) \cup \mathcal{V}_b(C)$;
4. si $A = \forall x.B$ ou $A = \exists x.B$ alors $\mathcal{V}_f(A) = \mathcal{V}_f(B) - \{x\}$ et $\mathcal{V}_b(A) = \mathcal{V}_b(B) \cup \{x\}$.

Une variable $x$ a une occurence libre dans une formule $A$ si et seulement si $x \in \mathcal{V}_f(A)$. Dualement, une variable $x$ a une occurence liée dans une formule $A$ si et seulement si $x \in \mathcal{V}_b(A)$. Nous somme donc dans cette situation étrange où une variable peut avoir à la fois une occurence libre *et* une occurence liée dans une même formule. Par exemple $x$ est à la fois libre et liée dans $P(x) \Rightarrow \forall x.Q(x)$.

Dans la formule $\forall x.A$ ou $\exists x.A$, la portée du quantificateur sur $x$ est la formule $A$. On peut vérifier que toutes les occurrences de $x$ dans $A$ sont alors liées.

On veut, pour les formules, définir une notion de substitution d'un terme à une variable. La présence dans les formules de lieurs va nous compliquer un peu la tâche. En effet si $c$ est une constante, la substitution $(\forall x.A)[c/x]$ définie sans plus de précaution donnerait $\forall c.A[c/x]$ ce qui a autant[4] de sens qu'une formule commenąnt par ≪*pour tout 0, ...*≫ La substitution d'un terme à une variable $x$ dans une formule $A$ ne sera donc définie que pour les occurences libres de $x$ dans $A$.
Il faut, lorsque, par le mécanisme de la substitution, on insère un terme à l'intérieur d'une formule, prendre garde au phénomène dit de *capture* de variables. En effet, dans un terme, toutes les variables sont libres et il faut éviter que la substitution fasse *tomber* l'une d'elles dans la portée d'un quantificateur. Si l'on n'évite pas ce phénomène de capture, la substitution peut mofifier de façon non voulue le sens des formules. Par exemple : $(\exists y.x = y)[y + 1/x]$ deviendrait $\exists y.y + 1 = y$.
Pour résoudre ce problème, on utilise le fait que le nom des variables liée n'a pas importance : les formules $\forall x \exists y.x = y$ a le même sens que $\forall z \exists y.z = y$ et que $\forall x \exists z.x = z$. En revanche, et c'est un avatar du phénomène de capture, les formules $\forall x \exists y.x + 1 = y$ n'est pas assimilable à $\forall x \exists x.x + 1 = x$. On pourra donc, à loisir changer le nom de la variable $x$ dans $\forall x.A$ en prenant toutefois garde à ne pas choisir de nom qui soit celui d'une autre variable liée de $A$. Ce choix est toujours possible puisque le nombre de variable d'une formule est toujours fini alors que l'ensemble $\mathcal{V}_1$ des variables est supposé infini. On s'autorisera donc à parler de sl nouvelle variable (pour $A$ ou $t$) pour signifier : une variable n'ayant pas d'occurrence (ni libre, ni liée) dans une formule $A$ ou un terme $t$.

Nous pouvons, aprés ce long préambule, passer à la définitin de la substitution du terme $t$ à la variable $x$ dans la formule $A$ :

1. si $A = P(t_1, .., t_n)$, avec $P \in \mathcal{P}_1$, alors $A[t/x] = P(t_1[t/x], .., t_n[t/x])$;
2. si $A = \neg B$ alors $A[t/x] = \neg B[t/x]$;
3. si $A = B \wedge C$, ou $A = B \vee C$, ou $A = B \Rightarrow C$ alors, respectivement, $A[t/x] = B[t/x] \wedge C[t/x]$, ou $A[t/x] = B[t/x] \vee C[t/x]$, ou $A[t/x] = B[t/x] \Rightarrow C[t/x]$;
4. si $A = \forall x.B$ ou $A = \exists x.B$ alors $A[t/x] = A$;
5. si $A = \forall y.B$ ou $A = \exists y.B$, avec $y \neq x$, alors $A[t/x] = \forall z.A[z/y][t/x]$, respectivement, $A[t/x] = \exists z.A[z/y][t/x]$ où $z$ est une nouvelle variable (pour $A$ et pour $t$)

## 2.2 Sémantique

Les formules du calcul des prédicats, plus expressives que celles du calcul des propositions, permettent de parler des *individus*, des *points* ou des *éléments* d'un *domaine* : l'ensemble des entiers, l'ensemble des états d'une machine, etc... Ces individus sont *dénotés* par les variables et les termes. Les symboles de fonction servent à exprimer les opérations auquelles peuvent être soumis les éléments du domaine. Les symboles de prédicats servent à exprimer les *relations* qu'entretiennent entre eux les individus.
L'objet du discours du calcul des prédicats est ce que l'on appelle une *structure*. Par exemple : le corps ordonné des réels de domaine $I\!\!R$ dont les élément peuvent être dans la relation $\leq$ et sur lesquels s'appliquent les opérations $+$ ou $\times$ (on peut aussi donner les deux éléments *distingués* 0 et 1). Pour parler de cette structure, il nous faudra nous donner un *langage* comportant un symbole de relation binaire, deux symbole de fonction binaires ainsi que de symboles de constantes. Les formules du calcul des prédicats construites sur ce langage, sur les connecteurs propositionnels et sur les quantificateurs seront ≪vraie≫ ou ≪fausses≫ relativement à la structure visée.

**Les symboles** Pour interpréter les formules du calcul des prédicats, il faut donc commencer par interpréter les symboles de constantes, les symboles de fonctions et les symboles de prédicats. Soit donc un domaine $\mathcal{D}$ :

1. à chaque symbole de constante $c$, on assigne un élément de $\mathcal{D}$ que l'on note $\|c\|_{\mathcal{D}}$[5];
2. à chaque symbole de fonction $f$ n-aire, on assigne une fonction de $\mathcal{D}^n$ dans $\mathcal{D}$ que l'on note $\|f\|_{\mathcal{D}}$;
3. à chaque symbole de prédicat n-aire $P$, on associe un sous ensemble de $\mathcal{D}^n$ que l'on note $\|P\|_{\mathcal{D}}$

Si on appelle $L$ notre langage (*ie* un ensemble $\mathcal{F}_1$ et un ensemble $\mathcal{P}_1$) Toute interprétation des symboles de ce langage telle que nous venons de le décrire constitue ce que l'on appelle une $L$-structure.

[4] *ie* aussi peu de ...
[5] on s'autorisera à omettre de préciser le domaine d'interprétation pour noter plus simplement $\|c\|$

**Les termes** L'interprétation d'un terme tel que $f(c)$ où $c$ est une constante peut nous être donnée par la seule interprétation des symboles. En revanche, il n'en va pas de même pour $f(x)$ où $x$ est une variable dont l'interprétaion, justement peut varier. L'interprétation des termes sera donc relative à une interprétation des (symboles de) variables. Soit donc $\alpha$ une fonction de $\mathcal{V}_1$ dans $\mathcal{D}$ que l'on appelera une *assignation*. On étend $\alpha$ à l'ensemble des termes en définissant $\bar{\alpha}(t)$ comme :

1. si $t = x$ et $x \in \mathcal{V}_1$ alors $\bar{\alpha}(t) = \alpha(x)$;
2. si $t = c$ alors $\bar{\alpha}(t) = \|c\|$;
3. si $t = f(t_1, .., t_n)$ alors $\bar{\alpha}(t) = \|f\|(\bar{\alpha}(t_1), .., \bar{\alpha}(t_n))$.

Si $\alpha$ est une assignation, $x \in \mathcal{V}_1$ et $a \in \mathcal{D}$, on notera $\alpha_{[x \mapsto a]}$ la nouvelle assignation définie par

$\alpha_{[x \mapsto a]}(x) = a$ et
$\alpha_{[x \mapsto a]}(y) = \alpha(y)$ si $y \neq x$.

**Les formules** Soit donc $\mathcal{M}$ une $L$-structure, $\alpha$ une assignation et $A$ une formule, nous allons définir, inductivement sur $A$, la notion de *satisfaction* de $A$ pour $\mathcal{M}$ et $\alpha$ (ce que l'on note, un peu par abus, $< \mathcal{M}, \alpha > \models A$) On notera $< \mathcal{M}, \alpha > \not\models A$ la négation de la satisfaction :

1. si $A = P(t_1, .., t_n)$ alors $< \mathcal{M}, \alpha > \models A$ si et seulement si $(\bar{\alpha}(t_1), .., \bar{\alpha}(t_n)) \in \|P\|$;
2. si $A = \neg B$ alors $< \mathcal{M}, \alpha > \models A$ si et seulement si $< \mathcal{M}, \alpha > \not\models B$;
3. si $A = B \wedge C$ alors $< \mathcal{M}, \alpha > \models A$ si et seulement si $< \mathcal{M}, \alpha > \models B$ et $< \mathcal{M}, \alpha > \models C$;
4. si $A = B \vee C$ alors $< \mathcal{M}, \alpha > \models A$ si et seulement si $< \mathcal{M}, \alpha > \models B$ ou $< \mathcal{M}, \alpha > \models C$;
5. si $A = B \Rightarrow C$ alors $< \mathcal{M}, \alpha > \models A$ si et seulement si $< \mathcal{M}, \alpha > \not\models B$ ou $< \mathcal{M}, \alpha > \models C$;
6. si $A = \forall x.B$ alors $< \mathcal{M}, \alpha > \models A$ si et seulement si pour tout $a \in \mathcal{D} < \mathcal{M}, \alpha_{[x \mapsto a]} > \models A$;
7. si $A = \exists x.B$ alors $< \mathcal{M}, \alpha > \models A$ si et seulement si pour au moins un $a \in \mathcal{D} < \mathcal{M}, \alpha_{[x \mapsto a]} > \models A$.

Si $A$ est une formule close alors le fait que $A$ soit satisfaite (ou non) ne dépend pas de l'assignation $\alpha$[6]. On peut alors simplement écrire $< \mathcal{M} \models A$. On dit alors que $\mathcal{M}$ *satisfait* $A$, ou que $A$ est *vraie dans* $\mathcal{M}$ ou encore que $\mathcal{M}$ est un *modèle* de $A$.

## 2.3 Conséquence sémantique

Comme nous l'avons fait pour le calcul des propositions, on définit pour le calcul des prédicats une notion de conséquence sémantique. Donnons auparavent quelques définitions.

Une formule close $A$ (du langage $L$) est dite *universellement valide* si et seulement si, pour toute $L$-structure $\mathcal{M}$, on a $\mathcal{M} \models A$.

Une formule close $A$ est *incosistante* si et seulement si sa négation est universellement valide.

On appelle *théorie* (du langage $L$) tout ensemble de formules closes (de $L$).

Une $L$-structure $\mathcal{M}$ est un *modèle* d'une théorie T si et seulement si pour toute formule $A \in \mathrm{T}$, on a $\mathcal{M} \models A$. On écrira alors $\mathcal{M} \models \mathrm{T}$.

Une théorie T est *consistante* si et seulement si elle admet un modèle (*ie* : il existe $\mathcal{M}$ tel que $\mathcal{M} \models \mathrm{T}$)

On définit enfin la conséquence sémantique comme une relation entre théories et formules closes :

Une formule close $A$ est conséquence sémantique d'une théorie T si et seulement si pour toute $L$-structure $\mathcal{M}$, si $\mathcal{M} \models \mathrm{T}$ alors $\mathcal{M} \models A$. On note : $\mathrm{T} \models A$.

Les formules universellement valides, qui sont l'équivalent pour le calcul des prédicats des tautologies du calcul des propositions, sont les formules conséquences de la *théorie vide*, ce que l'on peut noter $\models A$, si $A$ est universellement valide.

[6] c'est un théorème qu'il faudrait démontrer

## 2.4 Preuves formelles

On étend les systèmes de déduction naturelles et du calcul des séquents donnés en (1.4) aux nouvelles constructions du calcul des prédicats : les quantificateurs.

**Déduction naturelle** Si $\Gamma$ est un ensemble de formules, nous dirons qu'une variable $x$ n'est pas libre dans $\Gamma$ si et seulement si elle n'est libre dans aucune formule de $\Gamma$.

| L'UNIVERSELLE | |
|---|---|
| Intro | Elim |
| $\dfrac{\Gamma \vdash A}{\Gamma \vdash \forall x.A}\ \forall_i$ <br> si $x$ n'est pas libre dans $\Gamma$ | $\dfrac{\Gamma \vdash \forall x.A}{\Gamma \vdash A[t/x]}\ \forall_e$ <br> pour tout terme $t$ |

La contrainte ≪$x$ n'est pas libre dans $\Gamma$≫ sur la règle d'introduction permet d'éviter les inférences fallacieuses du style : ≪si $x = 0$ alors $\forall x.x = 0$≫

| L'EXISTENTIELLE | |
|---|---|
| Intro | Elim |
| $\dfrac{\Gamma \vdash A[t/x]}{\Gamma \vdash \exists x.A}\ \exists_i$ <br> pour tout terme $t$ | $\dfrac{\Gamma \vdash \exists x.A}{\Gamma \vdash A[x_0/x]}\ \exists_e$ <br> si $x_0$ n'est libre ni dans $\Gamma$ ni dans A |

Dualement, la contrainte ≪$x_0$ n'est libre ni dans $\Gamma$ ni dans $A$≫permet d'éviter de conclure que $x \neq 0$ des hypothèse $x = 0$ et $\exists y.y \neq 0$.

**Le calcul des séquents**

| L'UNIVERSELLE | |
|---|---|
| Gauche | Droite |
| $\dfrac{A[t/x], \Gamma \vdash \Delta}{\forall x.A, \Gamma \vdash \Delta}\ \forall_g$ <br> pour tout terme $t$ | $\dfrac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta, \forall x.A}\ \forall_d$ <br> si $x$ n'est libre ni dans $\Gamma$ ni dans$\Delta$ |

| L'EXISTENTIELLE | |
|---|---|
| Gauche | Droite |
| $\dfrac{A[x_0/x], \Gamma \vdash \Delta}{\exists x.A, \Gamma \vdash \Delta}\ \exists_g$ <br> si $x_0$ n'est libre ni dans $\Gamma$ ni dans $\Delta$ | $\dfrac{\Gamma \vdash \Delta, A[t/x]}{\Gamma \vdash \Delta, \exists x.A}\ \exists_d$ <br> pour tout terme $t$ |

**Correction, complétude, décidabilité** Nous retrouvons pour le calcul des prédicats l'adéquation que nous avons montré pour le calcul des proposition entre vérités sémantique et dérivabilité formelle :

THÉORÈME (6) : Pour tout ensemble de formules $\Gamma$ et toute formule $A$, $\Gamma \models A$ si et seulement si $\Gamma \vdash A$.

Nous ne donnons (pour l'instant) pas les preuves de ce théorème. Le sens ≪correction≫ s'obtient de façon analogue au calcul des propositions en démontrant que chacune des nouvelles règle conserve la validité des séquents. Pour ce qui est du ses ≪complétude≫, les choses sont rendue un peu plus compliquées par le fait qu'une formule qui n'est pas prouvable peut avoir un arbre de dérivation de séquents infini (ce qui n'était pas le cas avec le calcul des propositions) Rappelons nous que les ensembles de formule $\Gamma$ et $\Delta$ sont vraiment des ensembles en ce sens que $\Gamma, A, a = \Gamma, A$. On peut expliciter cette règle comme règle de *contraction*:

| CONTRACTION | |
|---|---|
| Gauche | Droite |
| $\dfrac{A, A, \Gamma \vdash \Delta}{A, \Gamma \vdash \Delta}\, cont_g$ | $\dfrac{\Gamma \vdash \Delta, A, A}{\Gamma \vdash \Delta, A, A}\, cont_d$ |

On peut alors construire la dérivation suivante (stupide, soit) qui utilise un symbole de fonction $f$ et un symbole de constante $a$ :

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\vdots}{\Gamma, \forall x.A, A[f(f(a))/x], A[f(a)/x], A[a/x] \vdash \Delta}}{\Gamma, \forall x.A, \forall x.A, A[f(a)/x], A[a/x] \vdash \Delta}\,\forall_g}{\Gamma, \forall x.A, A[f(a)/x], A[a/x] \vdash \Delta}\,cont_g}{\Gamma, \forall x.A, \forall x.A, A[a/x] \vdash \Delta}\,\forall_g}{\Gamma, \forall x.A, A[a/x] \vdash \Delta}\,cont_g}{\Gamma, \forall x.A, \forall x.A \vdash \Delta}\,cont_g}{\Gamma, \forall x.A \vdash \Delta}\,\forall_g$$

On est cependant sauvé par un résultat important du calcul des prédicat : le *théorème de Herbrand* qui dit, en substance, que lorsqu'une formule est prouvable, on n'a besoin que d'un nombre fini de ses instances. Nous n'irons pas plus avant dans l'argumentation `:-)`

Bien qu'en théorie, il existe une dérivation de chaque formule universellement valide, on ne peut pas pour autant en tirer une procédure de décision. Ceci est toujours dû au fait qu'il est toujours possible de prolonger un arbre de dérivation dans le calcul des prédicats. On n'a donc plus de notion de dérivation maximale qui permettait, dans le calcul des propositions, de décider si une proposition est vraie ou fausse (*ie* : tautologique ou non). Il existe, bien entendu, une preuve plus formelle de l'*indécidabilité* du calcul des prédicats. Celle-ci utilise la technique de la réduction à un problème connu : on sait qu'il n'existe pas de procédure de calcul capable de dire si un autre programme s'arrête ou non (c'est ce que l'on appelle l'indécidabilité du *problème de l'arrêt*; on montre que l'on *encoder* le problème de l'arrêt comme une formule du calcul des prédicats; comme le problème de l'arrêt est indécidable, la vérité de la formule qui le représente l'est également; et donc, puisqu'il existe au moins une formule indécidable, l'ensemble du calcul des prédicat est indécidable.
Ceci, bien entendu, n'invalide en rien le fait de savoir concevoir des programmes de démonstration automatique pour le calcul des prédicats. Le théroème d'indécidabilité dit simplement qu'il est vain d'espérer trouver un programme pour décider de *toutes* les formules universellement valides.