Patrick Thévenon

DEA de Mathématiques

Université Claude Bernard - Lyon 1
Ecole normale Supérieure de Lyon
Université de Savoie - Chambéry
Université Jean Monnet - Saint Etienne

Année universitaire 2002-2003

Le lambda calcul symétrique

Mémoire de DEA
Présenté à
l'Université de Savoie - Chambéry
Sous la direction de
René David

Table des matières

Introduction

Dans le domaine de la logique mathématique, et plus particulièrement dans celui du calcul typé, il est possible d'introduire divers systèmes de typage en fonction des besoins. Celui que nous allons aborder ici est le λ-calcul symétrique introduit par Barbanera et Berardi. Celui-ci est explicitement basé sur une négation involutive, et pour cela comporte des règles symétriques.
Ce calcul sera introduit en deux fois. En effet, dans une première partie, nous allons introduire un calcul propositionnel dont nous allons prouver la forte normalisation, bien qu'il ne vérifie pas la propriété de Church-Rosser. Ce calcul étant trop faible d'un point de vue contenu algorithmique, nous allons alors dans une seconde partie l'élargir. Le nouveau calcul sera noté $\lambda_{\mathbf{PA}}^{Sym}$-calcul pour la raison qu'il contient l'arithmétique de Peano.
Le $\lambda_{\mathbf{PA}}^{Sym}$-calcul est assez riche pour nous permettre de prouver un résultat d'extraction à partir d'un théorème dit de forme normale. Il s'agit par exemple, à partir de certaines preuves d'existence d'un d'entier vérifiant une propriété, d'extraire par réduction du terme associé un entier vérifiant la propriété. Pour que ce résultat soit interessant, il est alors nécessaire que nous prouvions que la propriété de forte normalisation est conservée. C'est ce qui occupera la troisième partie.
Ce mémoire comporte donc essentiellement deux aspects. Le premier est l'introduction d'une méthode de candidats dits symétriques pour prouver la forte normalisation de chaque calcul. Cette méthode due à Tait et Girard utilisera fortement l'existence d'un point fixe permettant de définir les candidats. Dans l'annexe de ce mémoire, nous verrons d'ailleurs rapidement des mises en oeuvre différentes de cette même méthode, pour d'autres calculs du second ordre. Le second aspect est plus algorithmique, puisqu'il s'agit de la preuve du théorème de forme normal qui a des applications pour l'extraction à partir de preuves dans l'arithmétique de Peano.

Notations utilisées

λ^{Sym}_{Prop} : désigne le lambda calcul propositionnel symétrique.

$P[Q/x] = P[x := Q]$: le terme P dans lequel on a remplacé les occurences libres de x par Q.

[t : A] ou [A] dans un arbre de preuve : hypothèse qui a été utilisée.

$\overline{F}$ dans un arbre de preuve : une formule F qui est démontrable sans hypothèses.

$FV(P)$: l'ensemble des variables libres du terme P.

Var_C : l'ensemble des variables de type C.

$Term_C$: l'ensemble des termes de type C.

SN_C : l'ensemble des termes de type C fortement normalisables.

Les notations dépendant plus particulièrement de définitions seront données au cours du texte.

Chapitre 1

Un lambda calcul symétrique

1.1 Définitions

La base sur laquelle nous construisons les types est formée de deux ensembles de base de types :

- l'ensemble des types atomiques $\mathcal{A} = \{a, b, \dots\}$
- l'ensemble des types atomiques de négation $\mathcal{A}^\perp = \{a^\perp, b^\perp, \dots\}$

Définition : *(i) L'ensemble des types minimaux (nous écrirons aussi m-types) est défini par la grammaire suivante :*

$$A ::= \alpha \,|\, \alpha^\perp \,|\, A \wedge A \,|\, A \vee A$$

où α parcourt $\mathcal{A}$ et $\alpha^\perp$ parcourt $\mathcal{A}^\perp$.

(ii) l'ensemble des types est défini par la grammaire :

$$C ::= A \,|\, \perp$$

Il est nécessaire de définir les m-types car on souhaite avoir un calcul dans lequel les formules ne contiennent pas l'absurde, qui dans un système de typage peut être pensé comme le type vide. Un tel choix a été motivé pour des raisons techniques, qui apparaîtrons dans les prochaines pages. On remarque que ce n'est pas une restriction, car la formule $A \wedge \perp$ peut être identifiée à $\perp$ et $A \vee \perp$ avec A.

Par la suite, des lettres comme A, B, C, A_1, C_2, ... dénoterons des types.

Nous devons définir, à cause de l'existence de $\mathcal{A}$ et de $\mathcal{A}^\perp$, ce qu'est la négation d'un type, puisqu'elle n'est pas définie à partir de $\perp$.

Définition : *La négation $A^\perp$ d'un type A est définie par induction de la manière suivante :*

$$(\alpha)^\perp = \alpha^\perp \,;\, (\alpha^\perp)^\perp = \alpha \,;\, (A \wedge B)^\perp = A^\perp \vee B^\perp \,;\, (A \vee B)^\perp = A^\perp \wedge B^\perp$$

On remarque alors que la négation de ce calcul est involutive :

Lemme : *Soit A un type. Alors $A^{\perp\perp} = A$*
Preuve : Elle se fait par induction sur l'écriture de A.
Si $A = \alpha$, alors $(A^\perp)^\perp = (\alpha^\perp)^\perp = \alpha = A$.
Si $A = \alpha^\perp$, alors $(A^\perp)^\perp = ((\alpha^\perp)^\perp)^\perp = (\alpha)^\perp = \alpha^\perp = A$.
Si $A = B \wedge C$, alors $(A^\perp)^\perp = (B^\perp \vee C^\perp)^\perp = (C^\perp)^\perp \wedge (C^\perp)^\perp$, et par hypothèse d'induction $(A^\perp)^\perp = B \wedge C = A$.
Si $A = B \vee C$, alors $(A^\perp)^\perp = (B^\perp \wedge C^\perp)^\perp = (C^\perp)^\perp \vee (C^\perp)^\perp$, et par hypothèse d'induction $(A^\perp)^\perp = B \vee C = A$. □

Définition : *Les termes du calcul λ^{Sym}_{Prop} (calcul propositionnel symétrique) sont définis par les règles suivantes :*

$$\frac{}{x^A : A}\, var)$$

$$\frac{A_1 \,:\, A_1 \quad P_2 \,:\, A_2}{\langle P_1, P_2\rangle \,:\, A_1 \wedge A_2}\,\langle,\rangle) \qquad \frac{P_i \,:\, A_i}{\sigma_i(P_i) \,:\, A_1 \vee A_2}\,\sigma_i)(i=1,2)$$

$$\frac{\begin{array}{c}[x : A]\\ \vdots \\ P \,: \perp\end{array}}{\lambda x.P \,:\, A^\perp}\,\lambda) \qquad \frac{P_1 \,:\, A^\perp \quad P_2 \,:\, A}{P_1 \star P_2 \,: \perp}\,\star)$$

Par la suite, les lettres P, Q, Q_1, ... désignerons les termes, et x, y, ... les variables. Nous omettrons parfois d'expliciter les règles utilisées.

On appelle l'opérateur '$\star$' l'application symétrique, du fait que si l'on se donne deux termes $P^{A^\perp}$ et Q^A (de type $A^\perp$ et $A = A^{\perp\perp}$ respectivement), $P^{A^\perp} \star Q^A$ et $Q^A \star P^{A^\perp}$ sont deux λ^{Sym}_{Prop}-termes valides.

Définition : *On introduit maintenant les règles de λ^{Sym}_{Prop}-réduction :*

$$\beta)\ (\lambda x.P) \star Q \to_\beta P[Q/x] \quad et \quad \beta^\perp)\ Q \star (\lambda x.P) \to_{\beta^\perp} P[Q/x]$$

Si $x \notin FV(P)$:

$$\eta)\ \lambda x.(P \star x) \to_\eta P \quad et \quad \eta^\perp)\ \lambda x.(x \star P) \to_{\eta^\perp} P$$

Pour $i = 1, 2$:

$$\pi)\ \langle P_1, P_2\rangle \star \sigma_i(Q_i) \to_\pi P_i \star Q_i \quad et \quad \pi^\perp)\ \sigma_i(Q_i) \star \langle P_1, P_2\rangle \to_{\pi^\perp} Q_i \star P_i$$

Pour P de type $\perp$, et $E[-]$ un contexte de type $\perp$, tels que $E[-]$ ne capture aucune variable libre de P :

$$Triv)\ E[P] \to_{Triv} P$$

Enfin, $\to_1$ est l'union des réductions définies ci-dessus, et $\to$ est la clôture réflexive et transitive de $\to_1$.

En termes de dérivation, on peut voir les règles comme suit :

(β) :

$$\frac{\dfrac{\begin{array}{c}[x \,:\, A]\\ \vdots \\ P \,: \perp\end{array}}{\lambda x.P \,:\, A^\perp} \quad Q \,:\, A}{(\lambda x.P) \star Q \,: \perp} \quad \rightsquigarrow \quad \begin{array}{c}Q \,:\, A\\ \vdots \\ P[x := Q] \,: \perp\end{array}$$

(η) :

$$\frac{\dfrac{P \,:\, A \quad [x \,:\, A^\perp]}{P \star x \,: \perp}}{\lambda x(P \star x) \,:\, A} \quad \rightsquigarrow \quad P \,:\, A$$

(π) :

$$\frac{\dfrac{P_1 \,:\, A_1 \quad P_2 \,:\, A_2}{\langle P_1, P_2\rangle \,:\, A_1 \wedge A_2} \qquad \dfrac{Q_i \,:\, A_i^\perp}{\sigma_i(Q_i) \,:\, A_1^\perp \vee A_2^\perp}}{\langle P_1, P_2\rangle \star \sigma_i(Q_i) \,: \perp} \quad \rightsquigarrow \quad \frac{P_i \,:\, A_i \quad Q_i \,:\, A_i^\perp}{P_i \star Q_i \,: \perp}$$

Les règles ($\beta^\perp$), ($\eta^\perp$) et ($\pi^\perp$) se voient de manière semblable, grâce au fait que '$\star$' soit symétrique. On remarque alors que l'on a la

Propriété : *Les règles de λ^{Sym}_{Prop}-réduction conservent le type.*

Définition : *Soit k un entier naturel, et P un terme.*
(i) k est une borne pour P si l'arbre de réduction de P a une profondeur $\leq k$.
(ii) P est dit fortement normalisable si il a une borne. Autrement dit, P est fortement normalisable si et seulement si toute réduction partant de P est finie, ce qui est équivalent, par le lemme de König, à : P a un arbre de réduction fini.

1.2 Remarques sur le calcul classique

1.2.1 Les règles de calcul

En regard du calcul propositionnel classique, pour lequel il est besoin de règles supplémentaires, ici on peut les dériver. Nous avons défini l'élimination et l'introduction de l'absurde, l'introduction de la conjonction et de la disjonction. Dérivons les autres règles.
l'élimination de la conjonction :

$$\dfrac{\dfrac{A_1 \wedge A_2 \equiv (A_1^\perp \vee A_2^\perp)^\perp \qquad \dfrac{[A_i^\perp]}{A_1^\perp \vee A_2^\perp}\,\sigma_i}{\perp}\,\star)}{A_i}\,\lambda)$$

En définissant (il y a équivalence en logique classique) $A \to B =_{DEF} A^\perp \vee B$, la règle d'élimination de l'implication se dérive :

$$\dfrac{\dfrac{A \to B \equiv A^\perp \vee B \qquad \dfrac{A \quad [B^\perp]}{A \wedge B^\perp \equiv (A^\perp \vee B)^\perp}\,\langle,\rangle)}{\perp}\,\star)}{B}\,\lambda)$$

Voici la dérivation de la règle d'élimination de la disjonction :

$$\dfrac{\dfrac{A \vee B \qquad \dfrac{\dfrac{\dfrac{\begin{matrix}[A]\\ \vdots \\ C\end{matrix} \quad [C^\perp]}{\perp}\,\star)}{A^\perp}\,\lambda) \qquad \dfrac{\dfrac{\begin{matrix}[B]\\ \vdots \\ C\end{matrix} \quad [C^\perp]}{\perp}\,\star)}{B^\perp}\,\lambda)}{A^\perp \wedge B^\perp = (A \vee B)^\perp}\,\langle,\rangle)}{\perp}\,\star)}{C}\,\lambda)$$

Avant de finir, et pour clarifier la lecture, prouvons que $\vdash A \vee A^{\perp}$:

$$
\dfrac{[(A \vee A^{\perp})^{\perp}] \qquad \dfrac{\dfrac{\dfrac{[(A \vee A^{\perp})^{\perp}] \qquad \dfrac{[A]}{A \vee A^{\perp}}\,\sigma_1)}{\perp}\,\star)}{A^{\perp}}\,\lambda)}{A \vee A^{\perp}}\,\sigma_2)}{\dfrac{\perp}{A \vee A^{\perp}}\,\lambda)}\,\star)
$$

Enfin, voilà la dérivation de la règle d'introduction de l'implication :

$$
\dfrac{\dfrac{}{A \vee A^{\perp}} \qquad \dfrac{[A^{\perp}]}{A^{\perp} \vee B}\,\sigma_1) \qquad \dfrac{\begin{matrix}[A]\\ \vdots \\ B\end{matrix}}{A^{\perp} \vee B}\,\sigma_2)}{A^{\perp} \vee B \equiv A \rightarrow B}\,\vee_e
$$

1.2.2 Les règles de réduction

La β-réduction et la η-réduction définies dans le λ^{Sym}_{Prop}-calcul peuvent être vues comme des cas particuliers de la β-réduction et la η-réduction du calcul classique. Revoyons ces dernières ; Pour β :

$$
\dfrac{\dfrac{\begin{matrix}[x\,:\,A]\\ \vdots \\ u\,:\,B\end{matrix}}{\lambda xu\,:\,A \rightarrow B} \qquad v\,:\,A}{(\lambda xu)v\,:\,B} \quad \rightarrow_\beta \quad \begin{matrix}v\,:\,A\\ \vdots \\ u[x:=v]\,:\,B\end{matrix}
$$

Pour η :

$$
\dfrac{\dfrac{t\,:\,A \rightarrow B \quad [x\,:\,A]}{(t)x\,:\,B}}{\lambda x(t)x\,:\,A \rightarrow B} \quad \rightarrow_\eta \quad t\,:\,A \rightarrow B
$$

Si l'on considère que $A^{\perp}$ est vu dans le calcul classique comme $A \rightarrow \perp$, on voit que les réductions du λ^{Sym}_{Prop}-calcul sont celles du calcul classique dans le cas $B = \perp$.

La $\wedge$-coupure peut être vue comme une π-réduction suivie d'une η-réduction. En effet :

$$
\dfrac{\dfrac{\dfrac{P_1\,:\,A_1 \quad P_2\,:\,A_2}{\langle P_1, P_2\rangle\,:\,A_1 \wedge A_2} \qquad \dfrac{[x\,:\,A_i^{\perp}]}{\sigma_i(x)\,:\,A_1^{\perp} \vee A_2^{\perp}}}{\langle P_1, P_2\rangle \star \sigma_i(x)\,:\perp}}{\lambda x(\langle P_1, P_2\rangle \star \sigma_i(x))\,:\,A_i} \quad \rightsquigarrow \quad P_i\,:\,A_i
$$

Avec la suite de réductions :

$$
\lambda x(\langle P_1, P_2\rangle \star \sigma_i(x)) \rightarrow_\pi \lambda x(P_i \star x) \rightarrow_\eta P_i
$$

1.3 Forte normalisation de λ^{Sym}_{Prop}

Le λ^{Sym}_{Prop}-calcul n'a pas la propriété de Church-Rosser. On peut vérifier que le terme

$$T = \lambda x^{\alpha^\perp \wedge \alpha}.(\lambda y^\alpha.x \star \sigma_1^{\alpha,\alpha^\perp}(y)) \star (\lambda z^{\alpha^\perp}.x \star \sigma_2^{\alpha,\alpha^\perp}(z))$$

a deux formes normales distinctes. En effet, on peut voir que T se réduit par une succession de règles (β) et $(\beta^\perp)$ à :

$$\lambda x^{\alpha^\perp \wedge \alpha}.(x \star \sigma_1^{\alpha,\alpha^\perp}(\lambda z^{\alpha^\perp}.x \star \sigma_2^{\alpha,\alpha^\perp}(z))) \qquad \lambda x^{\alpha^\perp \wedge \alpha}.(x \star \sigma_2^{\alpha,\alpha^\perp}(\lambda y^\alpha.x \star \sigma_1^{\alpha,\alpha^\perp}(y)))$$

qui sont deux termes distincts.
Autre exemple peut-être plus frappant, le terme

$$T = \lambda z.\{(\lambda y.((\lambda u.(\langle y, z\rangle \star \sigma_2(u))) \star t_1)) \star (\lambda x.((\lambda u.(\langle x, z\rangle \star \sigma_2(u))) \star t_2))\}$$

de type quelconque A, avec t_1 et t_2 libres dans T quelconques et clos de type A, peut se réduire soit à t_1, soit à t_2. En effet, on a

$$\begin{array}{rcl} T & \rightarrow & \lambda z.\{(\lambda y.((\lambda u.z \star u) \star t_1)) \star (\lambda x.((\lambda u.z \star u) \star t_2))\} \\ & \rightarrow & \lambda z.\{(\lambda y.(z \star t_1)) \star (\lambda x.(z \star t_2))\} \end{array}$$

Et à partir de là, en utilisant la règle (β) ou $(\beta^\perp)$, nous obtenons une différence. Dans le premier cas,

$$T \rightarrow \lambda z.(z \star t_1)[y := \lambda x.(z \star t_2)] \rightarrow \lambda z.(z \star t_1) \rightarrow t_1$$

Dans le second cas,

$$T \rightarrow \lambda z.(z \star t_2)[x := \lambda y.(z \star t_1)] \rightarrow \lambda z.(z \star t_2) \rightarrow t_2$$

Par contre, on a le

Théorème : *Soit C un type. Alors tout terme de type C est fortement normalisable. On peut aussi écrire*

$$Term_C = SN_C$$

Dans cette section, nous allons nous attacher à la preuve de ce théorème. Mais tout d'abord, introduisons encore quelques définitions.

1.3.1 Définition des candidats symétriques

Définition : *Soient A, A_1, A_2 des m-types. Nous définissons les opérateurs :*

$Pair_{A_1,A_2} : \mathcal{P}(Term_{A_1}) \times \mathcal{P}(Term_{A_2}) \longrightarrow \mathcal{P}(Term_{A_1 \wedge A_2})$ *Par*
$Pair_{A_1,A_2}(X_1, X_2) := \{\langle P_1, P_2\rangle \ : \ A_1 \wedge A_2 \mid P_1 \in X_1 \text{ et } P_2 \in X_2\}$

$Sigma^i_{A_1,A_2} : \mathcal{P}(Term_{A_i}) \longrightarrow \mathcal{P}(Term_{A_1 \vee A_2})$ *Par*
$Sigma^i_{A_1,A_2}(X_i) := \{\sigma_i(P_i) \ : \ A_1 \vee A_2 \mid P_i \in X_i\} \quad (i = 1, 2)$

$Lambda_A : \mathcal{P}(Term_{A^\perp}) \longrightarrow \mathcal{P}(Term_A)$ *Par*
$Lambda_A(X) := \{\lambda x.P \ : \ A \mid \forall Q \in X \quad P[Q/x] \in SN_\perp\}$

On peut remarquer immédiatement que pour A fixé, l'opérateur $Lambda_A$ est décroissant (l'ordre est l'inclusion).

Définition : *Soit A un m-type. Par induction sur la structure de A, nous définissons simultanément :*

(a) Les opérateurs :
$$\begin{cases} Neg_A : \mathcal{P}(Term_{A^\perp}) & \longrightarrow & \mathcal{P}(Term_A) \\ Neg_{A^\perp} : \mathcal{P}(Term_A) & \longrightarrow & \mathcal{P}(Term_{A^\perp}) \end{cases}$$

(b) les ensembles $[\![A]\!]$ et $[\![A^\perp]\!]$

Comme suit :

(a) soient $X \subseteq Term_A$ et $Y \subseteq Term_{A^\perp}$. Alors :

- Si $A = \alpha$,
$$\begin{cases} Neg_\alpha(Y) & := & Var_\alpha \cup Lambda_\alpha(Y) \\ Neg_{\alpha^\perp}(X) & := & Var_{\alpha^\perp} \cup Lambda_{\alpha^\perp}(X) \end{cases}$$

- Si $A = A_1 \wedge A_2$ (et donc $A^\perp = A_1^\perp \vee A_2^\perp$),
$$\begin{cases} Neg_A(Y) & := & Var_A \cup Pair([\![A_1]\!], [\![A_2]\!]) \cup Lambda_A(Y) \\ Neg_{A^\perp}(X) & := & Var_{A^\perp} \cup (\bigcup_{i=1}^2 Sigma^i([\![A_i^\perp]\!]) \cup Lambda_{A^\perp}(X) \end{cases}$$

- Si $A = A_1 \vee A_2$ (et donc $A^\perp = A_1^\perp \wedge A_2^\perp$), comme la négation est involutive, on doit définir en accord avec le point précédent :
$$\begin{cases} Neg_A(Y) & := & Var_A \cup (\bigcup_{i=1}^2 Sigma^i([\![A_i]\!]) \cup Lambda_A(Y) \\ Neg_{A^\perp}(X) & := & Var_{A^\perp} \cup Pair([\![A_1^\perp]\!], [\![A_2^\perp]\!]) \cup Lambda_{A^\perp}(X) \end{cases}$$

(b) Du fait que $Lambda_A$ est décroissant, on observe de même que Neg_A et $Neg_{A^\perp}$ sont aussi décroissants. Alors la composition des deux opérateurs $Neg_A \circ Neg_{A^\perp}$ est croissante. Ainsi, une fois qu'ont été définis Neg_A et $Neg_{A^\perp}$ pour un A fixé, il est possible, grâce au théorème du point fixe de Tarski, trouver un point fixe de cette composition. Notons X_0 ce point fixe. Alors nous définissons :

$$\begin{cases} [\![A]\!] & := & X_0 \\ [\![A^\perp]\!] & := & Neg_{A^\perp}(X_0) \end{cases}$$

Nous étendons la définition des ensembles $[\![A]\!]$ à tous les types en définissant $[\![\perp]\!] := SN_\perp$.

Remarquons que comme $[\![A]\!]$ est un point fixe de $Neg_A \circ Neg_A^\perp$, l'opérateur Neg_A vérifie l'égalité $[\![A]\!] = Neg_A([\![A^\perp]\!])$. Nous avons alors une application

$$[\![-]\!] : A \longmapsto [\![A]\!] \subseteq Term_A$$

Qui vérifie $[\![\perp]\!] = SN_\perp$ et :

(i)	$Var_A \subseteq [\![A]\!]$		
(ii)	$\langle P_1, P_2 \rangle \in [\![A_1 \wedge A_2]\!]$	$\Leftrightarrow$	$P_1 \in [\![A_1]\!]$ *et* $P_2 \in [\![A_2]\!]$
(iii)	$\sigma_i(P_i) \in [\![A_1 \vee A_2]\!]$	$\Leftrightarrow$	$P_i \in [\![A_i]\!]$ *(i=1,2)*
(iv)	$\lambda x.P \in [\![A]\!]$	$\Leftrightarrow$	$\forall Q \in [\![A^\perp]\!] \quad P[Q/x] \in [\![\perp]\!]$

Vérifions-le brièvement :

(i) : Par définition même de $[\![A]\!] = Neg_A([\![A^\perp]\!])$, l'inclusion est donnée.
(ii) : Les seuls types s'écrivant $\langle P_1, P_2 \rangle$ dans $Neg_{A_1 \wedge A_2}([\![(A_1 \wedge A_2)^\perp]\!])$ sont les éléments de $Pair([\![A_1]\!], [\![a_2]\!])$. Ainsi par définition de $Pair$, nous avons bien l'équivalence.
(iii) : Les seuls types s'écrivant $\sigma_i(P_i)$ dans $Neg_{A_1 \vee A_2}([\![(A_1 \vee A_2)^\perp]\!])$ sont les éléments de $Sigma^i([\![A_i^\perp]\!])$. Ainsi par définition de $Sigma^i$, nous avons bien l'équivalence.
(iv) : Les seuls types s'écrivant $\lambda x.P$ dans $Neg_A([\![A^\perp]\!])$ sont les éléments de l'ensemble $Lambda_A([\![A^\perp]\!])$. Ainsi, par définition de $Lambda$, comme $[\![\perp]\!] = SN_\perp$, nous avons aussi l'équivalence.

Ces ensembles $[\![A]\!]$ sont appelés les candidats symétriques. Il nous faut maintenant prouver des propriétés qu'ils vérifient.

1.3.2 Propriétés des candidats symétriques

Lemme : *Soit C un type. Alors*

$$[\![C]\!] \subseteq SN_C$$

Preuve : Par induction sur la structure de C, considérons les différentes formes d'un élément $P \in [\![C]\!]$.

- Si $P = x^C$, il est clair que $P \in SN_C$

- Si $P = \langle P_1, P_2\rangle$, alors $C = C_1 \wedge C_2$, et par *(ii)*, $P_i \in [\![C_i]\!]$ $(i = 1,2)$. Par hypothèse d'induction appliquée à C_i, on obtient que $P_i \in SN_{C_i}$. Comme les seules réductions de $\langle P_1, P_2\rangle$ possibles sont les réductions de P_1 et P_2, $P \in SN_C$.

- Si $P = \sigma_i(P_i)$, alors $C = C_1 \vee C_2$, et par *(iii)*, $P_i \in [\![C_i]\!]$. Par hypothèse d'induction appliquée à C_i, on obtient que $P_i \in SN_{C_i}$. Comme les seules réductions de $\sigma_i(P_i)$ possibles sont les réductions de P_i, $P \in SN_C$.

- Si $P = \lambda x.P'$, alors x est une variable de type $C^\perp$. Donc par *(i)*, $x \in [\![C^\perp]\!]$ puis par *(iv)*, $P' \in [\![\perp]\!] = SN_\perp$. Toute réduction de $\lambda x.P'$ est soit une réduction sur P', soit une η ou une $\eta^\perp$-réduction (losrque $P' = P'_1 \star x$ ou $P' = x \star P'_1$ respectivement). Si k est une borne pour P', on voit alors que $k+1$ est une borne pour P. Ainsi, $P \in SN_C$.

- Si $P = P_1 \star P_2$, alors $C = \perp$, donc $P \in [\![\perp]\!] = SN_\perp = SN_C$. □

Lemme : Soient C un m-type et $P \in Term_C$. Alors

$$\textit{(a)}\ (\forall Q \in [\![C^\perp]\!] \quad P \star Q \in SN_\perp) \Rightarrow P \in [\![C]\!]$$

$$\textit{(b)}\ (\forall Q \in [\![C^\perp]\!] \quad Q \star P \in SN_\perp) \Rightarrow P \in [\![C]\!]$$

Preuve : Comme '$\star$' est symétrique, il nous suffit de prouver *(a)*. Cette preuve se fera par induction sur la structure de P. Notons que comme $C \neq \perp$, on ne peut pas avoir $P = P_1 \star P_2$. Supposons donc que $\forall Q \in [\![C^\perp]\!] \quad P \star Q \in SN_\perp$.

- Si $P = \alpha^C$, par *(i)* nous avons $P \in [\![C]\!]$.

- Si $P = \langle P_1, P_2\rangle$, alors $C = C_1 \wedge C_2$, et grâce à *(ii)* il suffit de prouver que $P_i \in [\![C_i]\!]$ pour $i = 1,2$. Grâce à l'hypothèse d'induction, il suffit de prouver que pour tout $Q_i \in [\![C_i^\perp]\!]$, $P_i \star Q_i \in SN_\perp$. Or, pour tout $Q_i \in [\![C_i^\perp]\!]$ et par *(iii)*, $\sigma_i(Q_i) \in [\![C_1^\perp \vee C_2^\perp]\!] = [\![C^\perp]\!]$. Par hypothèse, nous avons alors $P \star \sigma_i(Q_i) \in SN_\perp$. Or ce dernier terme se réduit à $P_i \star Q_i$. Donc $P_i \star Q_i \in SN_\perp$ et $P \in [\![C]\!]$.

- Si $P = \sigma_i(P_i)$, alors $C = C_1 \vee C_2$, et grâce à *(iii)* il suffit de prouver que $P_i \in [\![C_i]\!]$. Grâce à l'hypothèse d'induction, il suffit de prouver que pour tout terme $Q_i \in [\![C_i^\perp]\!]$, $P_i \star Q_i \in SN_\perp$. Or, pour tous $Q_j \in [\![C_j^\perp]\!]$ $(j = 1,2)$ et par *(ii)*, $\langle Q_1, Q_2\rangle \in [\![C_1^\perp \wedge C_2^\perp]\!] = [\![C^\perp]\!]$. Par hypothèse, nous avons alors $P \star \langle Q_1, Q_2\rangle \in SN_\perp$. Or ce terme se réduit à $P_i \star Q_i$. Donc $P_i \star Q_i \in SN_\perp$ et $P \in [\![C]\!]$.

- Si $P = \lambda x.P'$, alors par hypothèse, $\forall Q \in [\![C^\perp]\!] \quad \lambda x.P' \star Q \in SN_\perp$. Or ce dernier terme se réduit à $P'[Q/x]$. Ainsi, $\forall Q \in [\![C^\perp]\!] \quad P'[Q/x] \in SN_\perp$. Grâce à *(iv)* nous obtenons alors que $P \in [\![C]\!]$. □

Lemme : *Soit C un type. Alors :*

$$P \in [\![C]\!],\ P \to_1 P' \Rightarrow P' \in [\![C]\!]$$

Preuve : Par induction sur la composition de C, considérons $P \in [\![C]\!]$.

- Si $P = \alpha^C$, P est normal, et donc il ne peut y avoir de réduction. Ce cas n'existe donc pas.

- Si $P = \langle P_1, P_2\rangle$, alors $C = C_1 \wedge C_2$ et par *(ii)*, $P_i \in [\![C_i]\!]$ pour $i = 1, 2$. Alors soit $P_1 \rightarrow_1 P_1'$, soit $P_2 \rightarrow_1 P_2'$. Disons que nous sommes dans le premier cas (le second cas est symétrique). Alors $P' = \langle P_1', P_2\rangle$. Par hypothèse d'induction, $P_1' \in [\![C_1]\!]$, et comme $P_2 \in [\![C_2]\!]$, par *(ii)* nous obtenons $P' \in [\![C]\!]$.

- Si $P = \sigma_i(P_i)$, alors $C = C_1 \vee C_2$ et par *(iii)*, $P_i \in [\![C_i]\!]$. Ainsi, $P_i \rightarrow_1 P_i'$, et $P' = \sigma_i(P_i')$. Par hypothèse d'induction, $P_i' \in [\![C_i]\!]$, et par *(iii)* nous obtenons $P' \in [\![C]\!]$.

- Si $P = \lambda x.R$, alors il y a deux cas :

+ $R \rightarrow_1 R'$ et $P' = \lambda x.R'$. Par *(iv)*, pour tout $Q \in [\![C^\perp]\!]$, $R[Q/x] \in SN_\perp$. Donc $R'[Q/x] \in SN_\perp$ et le point *(iv)* permet d'affirmer que $P' \in [\![C]\!]$.

+ $R = S \star x$ (ou par symétrie $R = x \star S$) et $P' = S$ (c'est à dire que l'on a effectué une réduction η ou par symétrie $\eta^\perp$). Pour prouver que $P' \in [\![C]\!]$, il suffit par le lemme précédent de prouver que pour tout $Q \in [\![C^\perp]\!] \quad S \star Q \in SN_\perp$. Or $x \notin FV(S)$ par définition de la réduction η, donc $R[Q/x] = S \star Q$ et $R[Q/x]$ appartient à $SN_\perp$ grâce à *(iv)*. Ainsi, $P' \in [\![C]\!]$.

- Si $P = P_1 \star P_2$, alors $C = \perp$ et $P \in [\![\perp]\!] = SN_\perp$. Comme $P \rightarrow_1 P'$, nous obtenons $P' \in SN_\perp = [\![C]\!]$. □

Lemme : *Soit C un m-type et soient $P \in [\![C]\!]$ et $Q \in [\![C^\perp]\!]$. Alors*

$$P \star Q \in SN_\perp$$

ce qui équivaut à

$$\forall S (P \star Q \longrightarrow_1 S \Rightarrow S \in SN_\perp)$$

Preuve : Prouvons d'abord l'équivalence des deux affirmations. Si $P \star Q$ appartient à $SN_\perp = [\![\perp]\!]$, alors par le lemme précédent, nous obtenons immédiatement la deuxième affirmation. Si maintenant pour tout S, $(P \star Q \longrightarrow_1 S \Rightarrow S \in SN_\perp)$, alors il est clair que $P \star Q \in SN_\perp$, puisque quelque soit la première réduction appliquée à $P \star Q$, on aboutit à un terme fortement normalisable de type $\perp$.

Pour prouver le lemme, il nous suffit alors de prouver la seconde affirmation. Comme $[\![C]\!] \subseteq SN_C$, il existe une borne n pour P. De même il existe une borne m pour Q. La preuve se fera par induction sur le couple (structure de C, $n+m$). Nous avons huit cas à considérer, groupés par symétrie :

1. $\begin{cases} P = \lambda x.P_1 & , \quad S = P_1[Q/x] \\ Q = \lambda x.Q_1 & , \quad S = Q_1[P/x] \end{cases}$

Pour ce cas, *(iv)* conclut immédiatement.

2. $\begin{cases} P = \langle P_1, P_2\rangle & , \quad Q = \sigma_i(Q_i) & , \quad S = P_i \star Q_i \\ P = \sigma_i(P_i) & , \quad Q = \langle Q_1, Q_2\rangle & , \quad S = P_i \star Q_i \end{cases}$

On utilise l'induction sur la structure de C, et la première affirmation avec $P_i \in [\![C_i]\!]$ et $Q_i \in [\![C_i^\perp]\!]$ grâce à *(ii)* et *(iii)*.

3. $\begin{cases} P \longrightarrow_1 P_1 & , \quad S = P_1 \star Q \\ Q \longrightarrow_1 Q_1 & , \quad S = P \star Q_1 \end{cases}$

Par le lemme précédent, $P_1 \in [\![C]\!]$ (resp. $Q_1 \in [\![C^\perp]\!]$). La borne de P_1 (resp. Q_1) étant strictement inférieur à n (resp. m), on obtient par induction sur la somme $n + m$, et en utilisant la première affirmation, que $S \in SN_\perp$.

4. $\begin{cases} P \star Q \longrightarrow_{Triv} S & , \quad S \text{ est un sous-terme de } P \\ P \star Q \longrightarrow_{Triv} S & , \quad S \text{ est un sous-terme de } Q \end{cases}$

Ici, S étant un sous-terme de P ou Q, qui sont fortement normalisables, est fortement normalisable. Or S se réduit de $P \star Q$ qui est de type $\perp$. Par conséquent S est de type $\perp$, fortement normalisable : $S \in SN_\perp$. □

Lemme : *Soit C un type et $P \in Term_C$ tel que $FV(P) \subseteq \{x_1^{A_1}, \ldots, x_n^{A_n}\}$. Alors*

$$\forall Q_1 \in [\![A_1]\!] \ldots Q_n \in [\![A_n]\!] \quad P[Q_1/x_1, \ldots, Q_n/x_n] \in [\![C]\!]$$

Preuve : Par induction sur la structure de P.

- Si P est une variable, il existe i tel que $P = x_i$ et $C = A_i$. Il suit alors que $P[Q_1/x_1, \dots, Q_n/x_n] = Q_i \in [\![A_i]\!] = [\![C]\!]$.

- Si $P = \langle P_1, P_2 \rangle$, alors $C = C_1 \wedge C_2$ et par hypothèse d'induction, pour $i = 1, 2$, $P_i[Q_1/x_1, \dots, Q_n/x_n] \in [\![C_i]\!]$. Alors par *(ii)*, $P[Q_1/x_1, \dots, Q_n/x_n] \in [\![C]\!]$.

- Si $P = \sigma_i(P_i)$, alors $C = C_1 \vee C_2$ et en utilisant l'hypothèse d'induction, on a $P_i[Q_1/x_1, \dots, Q_n/x_n] \in [\![C_i]\!]$. Alors par *(iii)*, $P[Q_1/x_1, \dots, Q_n/x_n] \in [\![C]\!]$.

- Si $P = P_1 \star P_2$, alors $C = \perp$. Si $P_1 \in Term_{C_1}$, on a $P_2 \in Term_{C_1^\perp}$. Ainsi par hypothèse d'induction, nous avons $P_1[Q_1/x_1, \dots, Q_n/x_n] \in [\![C_1]\!]$ et également $P_2[Q_1/x_1, \dots, Q_n/x_n] \in [\![C_1^\perp]\!]$. Alors $P[Q_1/x_1, \dots, Q_n/x_n] \in [\![\perp]\!] = [\![C]\!]$ grâce au lemme précédent.

- Si $P = \lambda y.P_1$, alors $y \notin \{x_1^{A_1}, \dots, x_n^{A_n}\}$ (y n'est pas une variable libre de P). Ainsi, $P[Q_1/x_1, \dots, Q_n/x_n] = \lambda y.P_1[Q_1/x_1, \dots, Q_n/x_n]$. Par *(iv)*, il suffit de prouver que $\forall Q \in [\![C^\perp]\!] \quad P_1[Q_1/x_1, \dots, Q_n/x_n, Q/y] \in SN_\perp$. Or cela provient de l'hypothèse d'induction. □

1.3.3 Preuve de la forte normalisation

Rappelons le théorème :

Théorème : *Soit C un type. Alors*

$$Term_C = SN_C$$

Preuve : Il est clair que $SN_C \subseteq Term_C$. Prouvons donc l'autre inclusion. Soit donc $P \in Term_C$. Notons $FV(P) = \{x_1^{A_1}, \dots, x_n^{A_n}\}$. Alors pour tout i, $x_i \in [\![A_i]\!]$ par *(i)*. Le lemme précédant permet d'affirmer que $P = P[x_1/x_1, \dots, x_n/x_n] \in [\![C]\!]$. Alors nous savons que $P \in SN_C$. □

Chapitre 2

Arithmétique de Peano

Le λ^{Sym}_{Prop}-calcul défini et étudié dans le chapitre précédant est trop pauvre pour permettre la recherche du contenu algorithmique des raisonnements classiques. En fait, il ne s'agissait que d'un point de départ. Nous voulons une logique dans laquelle il est possible d'exprimer et de prouver des programmes ayant plus de contenu que le calcul propositionnel. Le choix a donc été l'arithmétique de Peano. Dans la suite nous allons définir un calcul, correspondant à une version de déduction naturelle de l'arithmétique de Peano, basé sur le système λ^{Sym}_{Prop}. Ce nouveau calcul sera noté $\lambda^{Sym}_{\mathbf{PA}}$.

2.1 Définitions

Les types atomiques pour ce calcul ne vont plus être les même que pour le λ^{Sym}_{Prop}-calcul. Ici, les types atomiques seront formés à partir de **PA**-termes de type Int. Il nous faut donc dans un premier temps définir les **PA**-termes. Comme dans le chapitre précédant, nous allons commencer par définir les méthodes de contruction des termes, pour ensuite donner les règles de réduction pour les **PA**-termes, puis les $\lambda^{Sym}_{\mathbf{PA}}$-termes.

Définition (PA-termes) : *(i) Les types sont construit par la grammaire suivante :*

$$G ::= Int \,|\, G \to G$$

où Int est le type de constante.
*Les **PA**-termes sont tous les termes construits à partir des règles de construction suivantes. Soient g une variable numérique ou fonctionnelle, G, G_1 et G_2 des types.*

$$g^G : G \qquad\qquad 0 : Int$$

$$\frac{\begin{matrix}[g^{G_1} : G_1] \\ \vdots \\ p : G_2\end{matrix}}{\lambda g^{G_1}.p : G_1 \to G_2} \qquad\qquad \frac{p_1 : G_1 \to G_2 \quad p_2 : G_1}{(p_1)p_2 : G_2}$$

$$\frac{u : Int}{\mathbf{s}u : Int} \qquad\qquad \frac{u : Int \quad p : G \quad f : Int \to G \to G}{\mathbf{Rec}(u,p,f) : G}$$

*(ii) Sur les **PA**-termes, on définit les règles de réduction :*

$$\begin{array}{llll}
\beta_{\mathbf{PA}}) & (\lambda g.p)q & \longrightarrow_{\beta_{\mathbf{PA}}} & p[q/g] \\
\mathbf{Rec}_0) & \mathbf{Rec}(0,p,f) & \longrightarrow_{Rec_0} & p \\
\mathbf{Rec_s}) & \mathbf{Rec}(\mathbf{s}u,p,f) & \longrightarrow_{Rec_s} & f(u)\mathbf{Rec}(u,p,f)
\end{array}$$

(iii) Nous noterons $\rightarrow_{1\mathbf{PA}}$ la réunion des réductions définies ci-dessus, et $\rightarrow_{\mathbf{PA}}$ la clôture réflexive, transitive et contextuelle de $\rightarrow_{1\mathbf{PA}}$. Nous noterons enfin $\simeq$ la plus petite relation symétrique obtenue par $\rightarrow_{\mathbf{PA}}$

On peut remarquer que l'on a la

Propriété : *Les règles de réduction des* **PA***-termes conservent le type.*

Par la suite, les variables numériques (du type Int) seront notées n, m, ..., et les **PA**-termes seront notés u, v, t, ...

Nous verrons plus loin, au chapitre suivant, que les **PA**-termes sont fortement nomalisables. Alors à un **PA**-terme t sera associé $nf(t)$, sa forme normale. Le terme t sera dit nombre si il vaut 0 ou $\mathbf{s}^{k+1}0$ pour un entier naturel k, où $\mathbf{s}^0 = 0$ et $\mathbf{s}^{k+1}0 = \mathbf{s}(\mathbf{s}^k 0)$.

Définition ($\lambda^{Sym}_{\mathbf{PA}}$-termes) : *(i) Les types du système $\lambda^{Sym}_{\mathbf{PA}}$ sont définis de la manière suivante. Soient*

$$\mathcal{A} = \{u = v \mid u, v \ \mathbf{PA}\textit{-termes de type } Int\}$$

et

$$\mathcal{A}^\perp = \{u \neq v \mid u, v \ \mathbf{PA}\textit{-termes de type } Int\}$$

Alors les types minimaux sont définis par la grammaire

$$A ::= \alpha \,|\, \alpha^\perp \,|\, A \wedge A \,|\, A \vee A \,|\, \exists n.A \,|\, \forall n.A$$

où $\alpha \in \mathcal{A}$ et $\alpha^\perp \in \mathcal{A}^\perp$
La négation des types minimaux est définie par induction de la manière suivante

$$\begin{array}{rl}
(\alpha)^\perp = \alpha^\perp & (\alpha^\perp)^\perp = \alpha \\
(A \wedge B)^\perp = A^\perp \vee B^\perp & (A \vee B)^\perp = A^\perp \wedge B^\perp \\
(\exists n.A)^\perp = \forall n.A^\perp & (\forall n.A)^\perp = \exists n.A^\perp
\end{array}$$

Les types sont alors définis par la grammaire

$$C ::= A \,|\, \perp$$

Ce que nous définissons ici est bien une extension du λ^{Sym}_{Prop}-calcul. Par la suite, nous ne donnerons alors dans les définitions que les extensions au $\lambda^{Sym}_{\mathbf{PA}}$-calcul.

(ii) Les règles de construction des termes du $\lambda^{Sym}_{\mathbf{PA}}$-calcul sont données par :

- Règles atomiques

$$\frac{}{\mathbf{PA}_1 \,:\, (u = u)}\,\mathbf{PA}_1) \qquad \frac{P \,:\, (u = t)}{\mathbf{PA}_2(P) \,:\, (t = u)}\,\mathbf{PA}_2)$$

$$\frac{P \,:\, (u = v) \quad Q \,:\, (v = t)}{\mathbf{PA}_3(P,Q) \,:\, (u = t)}\,\mathbf{PA}_3) \qquad \frac{P \,:\, (\mathbf{s}0 = 0)}{\mathbf{PA}_4(P) \,:\perp}\,\mathbf{PA}_4)$$

$$\frac{P \,:\, (u = v)}{\mathbf{PA}_5(P) \,:\, (\mathbf{s}u = \mathbf{s}v)}\,\mathbf{PA}_5) \qquad \frac{P \,:\, (\mathbf{s}u = \mathbf{s}v)}{\mathbf{PA}_6(P) \,:\, (u = v)}\,\mathbf{PA}_6)$$

- Règles logiques. Nous ajoutons aux règles du λ^{Sym}_{Prop}-calcul les règles :

$$\frac{P\,:\,A}{\lambda_\forall n.P\,:\,\forall n.A}\,\lambda_\forall)\ \textit{(*)} \qquad \frac{P\,:\,A(t)}{\sigma_t(P)\,:\,\exists n.A(n)}\,\sigma_t)$$

$$\frac{u\,:\,Int \quad P\,:\,A(0) \quad \begin{array}{c}[n\,:\,Int]\,[x\,:\,A(n)]\\ \vdots \\ F\,:\,A(\mathbf{s}n)\end{array}}{\mathbf{Ind}_{n,x}(u,P,F)\,:\,A(u)}\,\mathbf{Ind})\ \textit{(**)}$$

$$\frac{P\,:\,A(u)}{P\,:\,A(u')}\,Conv)\ si\ u \simeq u'$$

(*) Pour tout $x^B \in FV(P)$, $n \notin FV(B)$.
(**) où n n'est pas libre en dehors de $A(n)$.

Les règles $(\langle,\rangle)$, (σ_i), (σ_t) et $(\lambda_\forall)$ seront appelées règles d'introduction. Nous dirons que le terme P représente la preuve A s'il est possible de dériver $P\,:\,A$. Nous nomerons terme atomique tout terme formé uniquement à partir de règles atomiques. Un terme ne contenant aucune variable libre de type **PA** sera dit **PA**-clos.

Définition (règles de $\lambda^{Sym}_{\mathbf{PA}}$-réduction) : *Ajoutons aux règles du λ^{Sym}_{Prop}-calcul les règles suivantes :*

$\beta_\forall)$ $(\lambda_\forall n.P) \star \sigma_t(Q) \longrightarrow_{\beta_\forall} P[t/n] \star Q$
$\beta_\forall^\perp)$ $\sigma_t(Q) \star (\lambda_\forall n.P) \longrightarrow_{\beta_\forall^\perp} Q \star P[t/n]$

$\mathbf{Ind}_0)$ $\mathbf{Ind}(0,P,F) \longrightarrow_{\mathbf{Ind}_0} P$
$\mathbf{Ind}_\mathbf{s})$ $\mathbf{Ind}_{n,x}(\mathbf{s}^{k+1}0,P,F) \longrightarrow_{\mathbf{Ind}_\mathbf{s}} F[\mathbf{s}^k0, \mathbf{Ind}_{n,x}(\mathbf{s}^k0,P,F)]$ *(*)*

*$Comp)u \rightarrow_{comp} u'$ si u et u' sont des **PA**-termes et $u \rightarrow_{\mathbf{PA}} u'$ (**)*

(*) Lorsque l'on note $F = F[n,x]$, la réduction exprime la substitution.
*(**) Cette règle permet de réduire les **PA**-termes dans les $\lambda^{Sym}_{\mathbf{PA}}$-termes.*

Comme précédement, on peut observer que l'on a la

Propriété : *Les règles de $\lambda^{Sym}_{\mathbf{PA}}$-réduction conservent le type.*

Le $\lambda^{Sym}_{\mathbf{PA}}$-calcul conserve également la propriété du λ^{Sym}_{Prop}-calcul :

Théorème : *Les termes de $\lambda^{Sym}_{\mathbf{PA}}$ sont fortement normalisables.*
Preuve : Celle-ci se fera plus tard. Il s'agira d'une extention de celle donnée au chapitre précédent. □

2.2 Forme normale

Nous introduisons maintenant deux ensembles de types, Σ^0_1 et Π^0_1. Des termes du premier, il sera possible d'extraire le contenu constructif exprimé par les types.

Définition : *(i) L'ensemble Σ^0_1 de types (formules) est composé à partir de termes **PA**-clos ne contenant ni des éléments de $\mathcal{A}^\perp$, ni des quantificateurs universels. Plus précisément, Σ^0_1 est la restriction aux éléments **PA**-clos de l'ensemble défini par la grammaire :*

$$S ::= \mathcal{A} \mid S \wedge S \mid S \vee S \mid \exists n.S$$

où n parcours la catégorie des variables numériques.

*(ii) L'ensemble Π^0_1 de types (formules) est composé à partir des types D tels que $D^\perp \in \Sigma^0_1$. Plus précisément, Π^0_1 est la restriction aux éléments **PA**-clos de l'ensemble défini par la grammaire :*

$$P ::= \mathcal{A}^\perp \,|\, P \wedge P \,|\, P \vee P \,|\, \forall n.P$$

où n parcours la catégorie des variables numériques.

Définition : *Un ensemble Δ de règles atomiques est dit consistant s'il n'existe pas de preuve atomique et close de $\perp$. Un système λ^{Sym}_Δ est dit consistant s'il n'existe pas de termes clos de type $\perp$.*

Lemme : *L'ensemble des règles atomiques de* **PA** *est consistant.*
Preuve : Soit une P preuve close et atomique. Considérons son arbre, dont la taille est n. Nous allons prouver que P n'est pas construit avec la règle $\mathbf{PA}_4$. Il n'y aura donc pas de preuve atomique close de type $\perp$ (qui n'existe que par la règle $\mathbf{PA}_4$).

Par récurrence décroissante, prouvons que pour tout $k \in \{1, \ldots, n\}$, au niveau k, les conclusions sont du type $(u = u)$ (égalité syntaxique), et que la règle $\mathbf{PA}_4$ n'est pas utilisée.

Au niveau n, comme P est close et atomique, il ne peut y avoir de prémisses, sinon P ne pourrait être close (aucune règle atomique ne lie les variables). Par conséquent la seule règle utilisée au niveau n est $\mathbf{PA}_1$, et les conclusions sont donc du type $(u = u)$.

Soit $2 \leq k+1 \leq n$ tel que au niveau $k+1$ les conclusions sont du type $(u = u)$, et que la règle $\mathbf{PA}_4$ n'est pas utilisée. Regardons le niveau k. Les prémisses viennent soit des conclusions du niveau $k+1$ et donc sont du type $(u = u)$ par hypothèse de récurrence, soit sont vides (toujours grâce au fait que P est atomique close). En regardant les règles atomiques, on voit alors que la règle $\mathbf{PA}_4$ ne peut être utilisée, et du fait que les prémisses non vides sont du type $(u = u)$, il est immédiat que les conclusions sont du même type. $\square$

Voici maintenant l'énoncé du théorème important de ce chapitre.

Théorème (de forme normale) : *Soit C un type qui est ou bien $\perp$, ou bien dans Σ^0_1. Soit P un terme clos et* **PA***-clos, normal de type C. Alors*

(i) Si C est atomique, alors P est atomique.

(ii) Si C est non-atomique, alors P termine par une des règles $(\langle,\rangle)$, (σ_i) ou (σ_t).

Remarquons que la restriction à Σ^0_1 est importante. En effet, un terme clos du type $\alpha \vee \alpha^\perp$ n'est pas de la forme $\sigma_i(P)$ car $(\alpha \vee \alpha^\perp)$ se prouve sans prouver ni α ni $\alpha^\perp$ (on a $\vdash \alpha \vee \alpha^\perp$).

Une conséquence de ce théorème est le

Corollaire : *Si Δ est consistant, alors λ^{Sym}_Δ est consistant. En particulier, par le lemme précédent, $\lambda^{Sym}_{\mathbf{PA}}$ est consistant.*
Preuve : Soit Δ un ensemble consistant de règles atomiques. Supposons qu'il existe un terme clos P' de type $\perp$. Sa forme normale P est encore close et du type $\perp$. D'après le théorème, nous savons alors que P est atomique. Ainsi P est une preuve atomique et close de $\perp$, ce qui contredit le fait que Δ soit consistant. Par conséquent, λ^{Sym}_Δ est consistant.

Le but jusqu'à la fin de la section est maintenant de prouver le théorème de forme normale. Pour cela, il va nous falloir donner quelques lemmes.

Lemme : *Soit u un* **PA***-terme normal. Alors il est dans un des deux cas suivants :*
(a) u termine par une introduction (0, **s** *ou λ)*

*(b) u est formé uniquement par des éliminations (**Rec** ou application) suivies par une variable. Plus précisément, si l'on représente u sous la forme d'un arbre, la branche la plus à gauche n'est formée que d'éliminations et termine par une variable.*

Preuve : - Si u est une variable, u est dans le cas (b).

- Si u termine par une règle d'introduction, u est dans le cas (a).

- Si u termine par une règle d'élimination, considérons le sous-terme strict u' le plus à gauche dans u. Comme u est normal, u' l'est également. Par hypothèse d'induction, on peut alors affirmer que u' est dans le cas (b), car sinon u' se terminerait par une introduction, et on pourrait alors appliquer une des trois règles de réduction sur u qui ne serait pas normal.

Ainsi, u est dans le cas (b). □

Lemme : *Soit u un* **PA***-terme clos de type Int. Alors sa forme normale est un nombre. C'est à dire que $nf(u) = \mathbf{s}^k 0$ pour un $k \geq 0$.*

Preuve : Comme les types sont conservés par réduction, nous obtenons que $nf(u)$ est de type Int, qui est également clos. Il suffit alors de prouver le lemme pour u normal, par induction sur u. Soit donc u un **PA**-terme clos normal de type Int. Alors u vérifie les hypothèses du lemme précédent. Or u ne peut être dans le cas (b), puisque u est clos et que la variable qui terminerait la succession d'éliminations serait libre. Par conséquent u est dans le cas (a). Mais u est de type Int, donc ne peut pas s'écrire $\lambda x.u'$. Ainsi, soit $u = 0$, et la preuve est terminée, soit $u = \mathbf{s}u'$, et l'induction permet de conclure. □

Lemme : *(i) Soit P un terme* **PA***-clos normal. Alors P n'est pas de la forme* **Ind** *(bien que* **Ind** *puisse apparaitre dans P).*

(ii) Soit $P_1 \star P_2$ un terme **PA***-clos normal. Alors l'un des deux termes P_1 et P_2 est une variable.*

Preuve : *(i)* Par contradiction, supposons que P est de la forme $\mathbf{Ind}(u, Q, F)$. Par le lemme précédent, il suit que u, qui est normal et clos de type Int, est un nombre $\mathbf{s}^k 0$ pour $k \geq 0$. Alors il est possible de réduire P avec une des règles ($\mathbf{Ind}_0$) ou ($\mathbf{Ind}_\mathbf{s}$). Ceci est contradictoire, car P est supposé normal.

(ii) Par contradiction, supposons que ni P_1 ni P_2 ne sont des variables. Ce sont deux termes normaux **PA**-clos. En notant $A^\perp$ le type de P_1, on sait que P_2 est de type A. Voyons les cas :

- P_1 ou P_2 est de la forme $\lambda x.Q$. Alors $P_1 \star P_2$ n'est pas normal. Contradiction.
- P_1 ou P_2 est de la forme $\mathbf{Ind}(u, P, F)$. Alors *(i)* est contredit.
- P_1 ou P_2 est une application ($\star$). Mais dans le cas où, par exemple, P_2 est une application, nous avons $A = \perp$, et la construction $P_1 \star P_2$ est impossible.

Alors P_1 et P_2 ne peuvent provenir que de règles d'introduction ou de règles atomiques. Mais si P_1 ou P_2 provient de règle atomique, alors il est de type atomique, et l'autre terme est de type atomique de négation. Comme les règles d'introduction et atomiques ne permettent pas d'obtenir un type atomique de négation, nous obtenons une contradiction.

Ainsi, P_1 et P_2 ne peuvent provenir que de règles d'introduction.

- Si un des deux est de type ($\langle , \rangle$) alors l'autre est nécessairement de type (σ_i). Alors $P_1 \star P_2$ n'est pas normal. Contradiction.
- Si un des deux est de type ($\lambda_\forall$) alors l'autre est nécessairement de type (σ_t). Alors $P_1 \star P_2$ n'est pas normal. Contradiction.

Nous concluons donc que nécessairement P_1 ou P_2 est une variable. □

Définition : *Soit P un terme de type C. P est un Σ_1^0-terme si*
(1) P est **PA***-clos ;*
(2) $C \in \Sigma_1^0$ ou $C = \perp$;
(3) Pour tout $x \in FV(P)$, si D est le type de x, $D \in \Pi_1^0$.

Lemme : *Soit P un Σ_1^0-terme normal, et Q un sous-terme de P. Alors :*
1. Q est une variable ssi il est de type Π_1^0. Dans ce cas, $Q = x$ apparaît dans P sous la forme $x \star Q'$ ou $Q' \star x$.
2. Q n'est pas une variable ssi c'est un Σ_1^0-terme.
Preuve : Par induction sur la structure de P.

- On voit tout de suite que les cas $P = x$ et $P = \lambda_\forall n.P'$ ne peuvent apparaitre, car ce ne sont pas des Σ_1^0-termes normaux.

- Le cas $P = \mathbf{Ind}(u, Q, F)$ ne peut apparaître non plus car P est un terme **PA**-clos, et ainsi le lemme précédant nous donne l'impossibilité de ce cas.

- Si $P = \langle P_1, P_2\rangle$, alors P est de type $A_1 \wedge A_2$, avec A_1 et A_2 dans Σ_1^0. Donc P_1 et P_2 sont des Σ_1^0-termes normaux. On utilise l'hypothèse d'induction, en remarquant que les sous-termes stricts de P sont les sous-termes de P_1 et de P_2.

- Si $P = \sigma_i(P_i)$, alors P est de type $A_1 \vee A_2$, avec A_1 et A_2 dans Σ_1^0. Donc P_i est un Σ_1^0-terme normal. L'hypothèse d'induction nous permet alors de conclure, puisque les sous-termes stricts de P sont les sous-termes de P_i.

- Si $P = \lambda x.P'$, comme P est un Σ_1^0-terme, on en conclut par définition que x a son type dans Π_1^0. Comme P' est de type $\perp$, et que $FV(P') = FV(P)\bigcup\{x\}$, on en conclut que P' est un Σ_1^0-terme. Cela nous permet d'utiliser l'hypothèse d'induction pour conclure.

- Si $P = P_1 \star P_2$, alors par le lemme précédent, on sait que P_1 ou P_2, disons P_1 est une variable x. Alors $x \in FV(P)$ et par définition, x est de type Π_1^0. Il suit que P_2 est de type Σ_1^0, et comme $FV(P_2) \subset FV(P)$, que P_2 est un Σ_1^0-terme. On utilise alors l'ypothèse d'induction pour conclure, puisque les sous-termes stricts de P sont x et les sous-termes de P_2.

- Si $P = \sigma_t(P')$, alors P est de type $\exists n.A(n)$, et P' est de type $A(t)$. P' est alors un Σ_1^0-terme, et on peut appliquer l'hypothèse d'induction pour conclure, en remarquant que les sous-termes stricts de P sont les sous-termes de P'.

- Si $P = \mathbf{PA}_i(P_1, \ldots, P_2)$ $(1 \leq i \leq 6)$, alors les P_i ont nécessairement un type atomique, et vérifient $FV(P_i) \subseteq FV(P)$. On conclut par hypothèse d'induction. □

Corollaire : *Soit P un Σ_1^0-terme normal. Alors*
(i) P ne contient pas de symbole $\lambda_\forall$.
(ii) P ne contient pas de symbole **Ind**.
Preuve : *(i)* Cette assertion provient directement du lemme précédent, puisque si P avait un sous-terme de la forme $\lambda_\forall n.P'$, celui-ci ne serait pas un Σ_1^0-terme, ce qui est une contradiction.

(ii) Par contradiction, supposons que P contienne le symbole **Ind**. Considérons alors $\mathbf{Ind}(u, R, F)$ le sous-terme maximal de P qui commence par **Ind**. Alors u est **PA**-normal par hypothèse, et comme P est normal, u contient nécessairement au moins une **PA**-variable libre n. Or P est **PA**-clos, et donc contient le symbole $\lambda_\forall$, ce qui contredit *(i)*. □

Nous pouvons maintenant prouver, dans le cas restreint des termes minimaux, le théorème de forme normale. Introduisons tout d'abord la notion de terme minimal.

Définition : *Un terme (preuve) dans $\lambda_{\mathbf{PA}}^{Sym}$ est minimal s'il est construit uniquement à partir des règles $(\langle,\rangle)$, (σ_i) et (σ_t).*

Lemme : *Soit P un terme minimal* **PA***-clos. Soit α un élément de $\mathcal{A}\bigcup\{\perp\}$. Alors :*

(i)	$P : \alpha$	$\Rightarrow$	*P est atomique*
(ii)	$P : A_1 \wedge A_2$	$\Rightarrow$	*P est de la forme $\langle P_1, P_2\rangle$ avec P_1 et P_2 minimaux*
(iii)	$P : A_1 \vee A_2$	$\Rightarrow$	*P est de la forme $\sigma_i(P_i)$ avec P_i minimal*
(iv)	$P : \exists n.A$	$\Rightarrow$	*P est de la forme $\sigma_t(P')$ avec P' minimal*

Preuve : *(i)* Par induction sur la structure de P. Comme P ne contient que des règles atomiques et d'introduction et est de type atomique, il a nécessairement la

forme $\mathbf{PA}_i(P_1, \ldots, P_2)$ avec $1 \leq i \leq 6$. Comme les P_i sont minimaux, l'hypothèse d'induction appliquée aux P_i permet de conclure.
(ii), (iii), (iv) Comme P ne contient que des règles atomiques et d'introduction, il est nécessairment de la forme $\langle P_1, P_2 \rangle$, $\sigma_i(P_i)$ et $\sigma_t(P')$ respectivement. P_1, P_2, P_i et P' sont minimaux car P l'est. $\Box$

Voici un lemme important qui va nous permettre de conclure par la preuve du théorème de forme normale.

Lemme : *Soit P un Σ_1^0-terme normal clos. Alors*
(i) Si P ne contient aucun symbole λ, alors il est minimal.
(ii) P ne contient pas de symbole λ.
On conclut alors que P est minimal. Par le lemme précédent, on obtient enfin la preuve du théorème de forme normale.
Preuve : *(i)* Soit P un Σ_1^0-terme clos sans symbole λ. Par définition, il nous faut prouver que P ne contient pas de variables libres ou liées, d'application symétrique ou de **Ind** ou de symbole $\lambda_\forall$. Comme P est clos, et que les variables ne sont liées que par des λ, P ne peut contenir de variables. Si P contient une application, alors par un lemme précédent, il contiendrait une variable, ce qui contredirait ce ce nous venons de prouver. Le corollaire précédent nous permet de conclure pour les cas **Ind** et $\lambda_\forall$.

(ii) Commençons d'abord par prouver que P ne contient pas de sous-terme de la forme $\lambda x.Q$ tel que $x \in FV(Q)$. Par contradiction, supposons que de tels sous-termes existent. Considérons alors $\lambda x'.Q'$ le plus court de ceux-ci. Puisque $x' \in FV(Q')$ et que P est un Σ_1^0-terme normal, par un lemme précédent, nous savons que x' apparaît dans Q' comme $x' \star Q''$ ou $Q'' \star x'$, disons le premier cas. Considérons le plus petit de ces termes, ce qui entraine le fait que $x' \notin FV(Q'')$. Ainsi, $Q' = C[x' \star Q'']$ où $C[]$ est un contexte.

Par minimalité de la longueur de $\lambda x'.Q'$, on sait que $FV(x' \star Q'') \subseteq FV(Q')$. Il suit que soit $Q' = x' \star Q''$ (i.e. $C[] = []$) soit que l'on peut alors réduire : $Q' \rightarrow_{Triv} x' \star Q''$. Cette dernière possibilité contredit la normalité de P. La première possibilité est également impossible, car on pourrait réduire également $Q = \lambda x' x' \star Q'' \rightarrow_\eta Q''$.

Nous pouvons alors en conclure qu'il n'existe pas de sous-terme de P de la forme $\lambda x.Q$ avec $x \in FV(Q)$. Il suit de ce fait que dans P, aucune variable n'a été déchargée (liée), puis comme P est clos, tous ses sous-termes le sont également.

Finissons en prouvons qu'il n'existe pas de sous-terme de la forme $\lambda x.Q$ quelconque. Par contradiction, prenons $\lambda x'.Q'$ un plus court sous-terme de cette forme, soit de telle sorte que λ n'apparaisse pas dans Q'. Comme Q' est de type $\perp$, ce n'est pas une variable, et ainsi par un lemme précédent, c'est un Σ_1^0-terme normal clos. Par le point *(i)* il suit que Q' est une preuve close minimale de $\perp$, et par le lemme précédent qu'il est atomique. Cela contredit la consistance des règles atomiques. $\Box$

2.3 Plus loin

Le théorème de forme normale prouvé dans la section précédente permet l'extraction du contenu calculatoire de preuves classiques. D'une preuve normalisée d'une disjonction, nous pouvons obtenir la preuve de l'une des disjonctions, et d'une preuve de formule existentielle, nous pouvons obtenir un témoin de l'existence. De façon plus générale, lorsque nous est donnée une preuve close d'une Σ_1^0-formule dans $\lambda_{\mathbf{PA}}^{Sym}$, il est possible, par l'inspection de la preuve normalisée, d'obtenir les témoins de toutes les sous-formules de la forme $\exists n.A(n)$. Cela signifie aussi que si l'on a une formule de la forme $\forall m.\exists n.A(m, n)$, on peut trouver, depuis une preuve P de celle-ci, une fonction $f : \mathbf{N} \rightarrow \mathbf{N}$ telle que pour tout $k \in \mathbf{N}$, $A(k, f(k))$ est vraie, c'est à dire que l'on peut expliciter l'entier n pour m donné. Pour k fixé dans $\mathbf{N}$, pour obtenir $f(k)$, il suffit de normaliser la preuve $\lambda x^{\forall n.A^{\perp}(k,n)}.(\sigma_k(x) \star P)$.

Donnons ici un exemple très simple : la preuve de $\forall m \exists n(n = sm)$ qui s'obtient de la manière suivante :

$$\dfrac{\dfrac{\dfrac{\overline{\mathbf{PA}_1 \ : \ m = m}}{\mathbf{PA}_5(\mathbf{PA}_1) \ : \ \mathbf{s}m = \mathbf{s}m}}{\sigma_{\mathbf{s}m}\mathbf{PA}_5(\mathbf{PA}_1) \ : \ \exists n(n = \mathbf{s}m)}}{\lambda_{\forall} m.\sigma_{\mathbf{s}m}\mathbf{PA}_5(\mathbf{PA}_1) \ : \ \forall m \exists n(n = \mathbf{s}m)}$$

Ainsi, $P = \lambda_{\forall} m.\sigma_{\mathbf{s}m}\mathbf{PA}_5(\mathbf{PA}_1)$. Réduisons alors :

$$\begin{array}{rcl} \lambda x.(\sigma_k(x) \star P) & \to & \lambda x.(x \star \sigma_{\mathbf{s}k}\mathbf{PA}_5(\mathbf{PA}_1)) \\ & \to & \sigma_{\mathbf{s}k}\mathbf{PA}_5(\mathbf{PA}_1) \end{array}$$

On voit donc que la forme normale termine bien par un σ_t, et que $f(k) = \mathbf{s}k$.

Grâce aux lemmes prouvés dans la section précédente, il est possible de donner des résultats interessants d'un point de vue théorique et applicatif.

Lemme : *Soit $P(x_1^{N_1}, \ldots, x_k^{N_k})$ un Σ_1^0-terme normal. Alors soit il est minimal (et par conséquent clos), soit il contient un sous-terme $x_i \star Q_i$ (ou $Q_i \star x_i$) avec $Q_i^{N_i^{\perp}}$ minimal.*

Preuve : Si P est clos (i.e. si $k = 0$), on utilise le dernier lemme de la section précédente. Sinon, nous savons par la section précédente que chaque x_i apparaît nécessairement sous la forme $x_i \star Q_i$ (ou $Q_i \star x_i$). Considérons un tel terme minimal, disons $x_i \star Q_i$. Alors Q_i est clos, et donc est un Σ_1^0-terme par un lemme précédent. Le dernier lemme de la section précédente nous donne le résultat. □

D'un point de vue théorique, ce lemme nous explique que pour toute

$$x_1^{N_1}, \ldots, x_k^{N_k} \vdash_{\lambda_{\mathbf{PA}}^{Sym}} P \ : \ A \text{ avec } A \in \Sigma_1^0 \text{ et } N_i \in \Pi_1^0$$

P contient soit un exemple pour A, soit un contre-exemple à un des N_i.

D'un point de vue applicatif, ce lemme peut être utilisé pour accélerer le processus d'extraction de contenu constructif. Si la Σ_1^0-preuve $P[R_1/x_1, \ldots, R_k/x_k]$ contient des termes clos R_i de type Π_1^0, il n'est pas nécessaire de les considérer dans le processus de normalisation. Il suffit de les remplacer par de nouvelles variables, car les formes normales de P sont les formes normales de $P[R_1/x_1, \ldots, R_k/x_k]$. Prouvons cette affirmation :

Il suffit d'observer que si la forme normale de P est minimale, alors elle est close, et par conséquent c'est aussi une preuve normale de $P[R_1/x_1, \ldots, R_k/x_k]$. De plus, par le lemme, la forme normale de P pourrait avoir des termes $x_i \star Q_i$ (ou $Q_i \star x_i$) avec Q_i minimaux. Or cela serait contradictoire, puisque en remplaçant les x_i par les R_i, nous aboutirions à une preuve close de $\perp$, ce qui est impossible.

Chapitre 3

Forte normalisation de $\lambda_{\mathbf{PA}}^{Sym}$

Nous allons maintenant nous attacher à la preuve de la propriété de forte normalisation du $\lambda_{\mathbf{PA}}^{Sym}$-calcul.

Pour ce faire, nous allons modifier le système $\lambda_{\mathbf{PA}}^{Sym}$ par le suivant : Au lieu d'utiliser les ensembles

$$\mathcal{A} = \{u = v \mid u, vPA\text{-termes de type } Int\}$$

et

$$\mathcal{A}^{\perp} = \{u \neq v \mid u, vPA\text{-termes de type } Int\}$$

nous allons prendre les singletons

$$\underline{\mathcal{A}} = \{\bigcup\} \text{ et } \underline{\mathcal{A}^{\perp}} = \{\bigcup^{\perp}\}$$

Les règles de construction et de réduction restent les mêmes, si ce n'est que $\bigcup$ remplace les formules atomiques et que $\bigcup^{\perp}$ remplace les négations de formules atomiques. On obtient alors le $\lambda_{\mathbf{PA}-\bigcup}^{Sym}$. Ainsi, pour tout terme du $\lambda_{\mathbf{PA}}^{Sym}$-calcul il existe un terme pour le $\lambda_{\mathbf{PA}-\bigcup}^{Sym}$-calcul et les arbres de réduction des deux termes sont isomorphes. Pour prouver la forte normalisation de $\lambda_{\mathbf{PA}}^{Sym}$ il suffit alors de prouver celle de $\lambda_{\mathbf{PA}-\bigcup}^{Sym}$, ce que nous allons faire.

Notre preuve va consister en une extension de celle du théorème de forte normalisation de λ_{Prop}^{Sym}, avec l'exposé des mêmes lemmes. Nous ne prouverons alors que les parties étendues, les parties communes se prouvant de manière identique.

Nous allons comme précédemment introduire les candidats symétriques.

3.1 Candidats symétriques pour $\lambda_{\mathbf{PA}-\bigcup}^{Sym}$

Définition : *Aux fonctions déjà définies, nous ajoutons :*
$Ind_{0,A}$, $Ind_{\mathbf{s},A}$, $Ind_{\bullet,A}$, Ind_A, $IND_A : \mathcal{P}(Term_A) \longrightarrow \mathcal{P}(Term_A)$ *par*

$Ind_{0,A}(X) := \{\mathbf{Ind}(v, P, F) : A \mid nf(v) = 0,\ P \in X \cap SN_A,\ F \in SN_A\}$
$Ind_{\mathbf{s},A}(X) := \{\mathbf{Ind}(v, P, F) : A \mid P \in SN_A,\ F[\mathbf{s}^k 0, \mathbf{Ind}(\mathbf{s}^k 0, P, F)] \in X \cap SN_A\}$
$Ind_{\bullet,A}(X) := \{\mathbf{Ind}(v, P, F) : A \mid nf(v) \text{ n'est pas un nombre},\ P \in SN_A,\ F \in SN_A\}$
$Ind_A(X) := Ind_{0,A}(X) \cup Ind_{\mathbf{s},A}(X) \cup Ind_{\bullet,A}(X)$
$IND_A(X) :=$*le plus petit point fixe de l'application croissante* $Y \mapsto X \cup Ind_A(Y)$

Puis nous définissons

$PourTout_{n,A} : \mathcal{P}(Term_A) \longrightarrow \mathcal{P}(Term_{\forall n.A})$ *par*
$PourTout_{n,A}(X) := \{\lambda_{\forall} n.P : \forall n.A \mid \text{pour tout } t : Int,\ P[t/n] \in X\}$

$Existe_{n,A} : \mathcal{P}(Term_A) \longrightarrow \mathcal{P}(Term_{\exists n.A})$ *par*
$Existe_{n,A}(X) := \{\sigma_t(P) : \exists n.A \mid t : Int,\ P \in X\}$

$Ax_{\mathbf{PA}_i} : \mathcal{P}(Term_{\bigcup}) \times \cdots \times \mathcal{P}(Term_U) \longrightarrow \mathcal{P}(Term_U)$ *par*
$Ax_{\mathbf{PA}_i}(X_1, \ldots, X_{n_i}) := \{\mathbf{PA}_i(P_1, \ldots, P_{n_i}) \mid P_1 \in X_1, \ldots, P_{n_i} \in X_{n_i}\}$ *(*$1 \leq i \leq 6$*)*

Définition : *Soit A un m-type. Par induction sur la structure de A, nous définissons simultanément :*
(a) Les opérateurs

$$\begin{cases} Neg_A : \mathcal{P}(Term_{A^\perp}) & \longrightarrow & \mathcal{P}(Term_A) \\ Neg_{A^\perp} : \mathcal{P}(Term_A) & \longrightarrow & \mathcal{P}(Term_{A^\perp}) \end{cases}$$

(b) L'opérateur
$NEG_A : \mathcal{P}(Term_{A^\perp}) \longrightarrow \mathcal{P}(Term_A)$

(c) Les ensembles $[\![A]\!]$ *et* $[\![A^\perp]\!]$

Comme suit :

(a) Soit $X \subseteq Term_A$ *et* $Y \subseteq Term_{A^\perp}$*. Par la propriété d'involution de la négation, il nous suffit de donner les trois cas ci dessous :*

- $A = \bigcup$

$$\begin{cases} Neg_{\bigcup}(Y) & := & Var_{\bigcup} \bigcup (\cup_{i=1}^{6} Ax_{\mathbf{PA}_i}(SN_{\bigcup}, \ldots, SN_{\bigcup})) \bigcup Lambda_{\bigcup}(Y) \\ Neg_{\bigcup^\perp}(X) & := & Var_{\bigcup^\perp} \bigcup Lambda_{\bigcup^\perp}(X) \end{cases}$$

- $A = A_1 \wedge A_2$ *(*$A^\perp = A_1^\perp \vee A_2^\perp$*)*

$$\begin{cases} Neg_A(Y) & := & Var_A \bigcup Pair_A([\![A_1]\!], [\![A_2]\!]) \bigcup Lambda_A(Y) \\ Neg_{A^\perp}(X) & := & Var_{A^\perp} \bigcup (\cup_{i=1}^{2} Sigma^i([\![A_i^\perp]\!])) \bigcup Lambda_{A^\perp}(X) \end{cases}$$

- $A = \forall n.A_1$ *(*$A^\perp = \exists n.A_1^\perp$*)*

$$\begin{cases} Neg_A(Y) & := & Var_A \bigcup PourTout_{n,A_1}([\![A_1]\!]) \bigcup Lambda_A(Y) \\ Neg_{A^\perp}(X) & := & Var_{A^\perp} \bigcup Existe_{n,A_1^\perp}([\![A_1^\perp]\!]) \bigcup Lambda_{A^\perp}(X) \end{cases}$$

(b) Pour tout A, une fois Neg_A *définit, nous définissons* NEG_A *de la manière suivante :*
Soit $X \subseteq Term_{A^\perp}$

$$NEG_A(X) := IND_A(Neg_A(X))$$

(c) Pour tout A, Neg_A *est un opérateur décroissant (on rappelle que* $Lambda_A$ *l'est). De plus, comme* IND_A *est croissant, nous obtenons que* NEG_A *est décroissant. Il suit que la composition* $NEG_A \circ NEG_{A^\perp}$ *est croissante. Alors par le théorème de Tarski, une fois qu'est défini* NEG_A*,* $NEG_A \circ NEG_{A^\perp}$ *a un plus petit point fixe, que nous notons* X_0*. Nous définissons alors :*

$$\begin{cases} [\![A]\!] & := & X_0 \\ [\![A^\perp]\!] & := & NEG_{A^\perp}(X_0) \end{cases}$$

Nous étendons enfin la définition de l'application $[\![-]\!]$ *en posant* $[\![\perp]\!] := SN_\perp$*.*

Grâce à ce que nous venons de définir, nous voyons que nous avons l'existance d'une application

$$[\![-]\!] : A \longmapsto [\![A]\!] \subseteq Term_A$$

qui vérifie les mêmes propriétés que celle rencontrée précédemment, plus

(v) *Si* $v : Int$ *avec* $nf(v) = 0$, *alors*
$\mathbf{Ind}(v, P, F) \in [\![A]\!] \Leftrightarrow P \in [\![A]\!] \cap SN_A$ *et* $F \in SN_A$

(vi) *Si* $v : Int$ *avec* $nf(v) = \mathbf{s}^{k+1}0$, *alors*
$\mathbf{Ind}(v, P, F) \in [\![A]\!] \Leftrightarrow P \in SN_A,\ F[\mathbf{s}^k 0, \mathbf{Ind}(\mathbf{s}^k 0, P, F)] \in [\![A]\!] \cap SN_A$

(vii) *Si* $v : Int$ *et* $nf(v)$ *n'est pas un nombre, alors*
$\mathbf{Ind}(v, P, F) \in [\![A]\!] \Leftrightarrow P,\ F \in SN_A$

(viii) $\lambda_\forall n.P \in [\![\forall n.A]\!] \Leftrightarrow \forall t : Int,\ P[t/n] \in [\![A]\!]$

(ix) $\sigma_t(P) \in [\![\exists n.A]\!] \Leftrightarrow P \in [\![A]\!]$

(x) $\mathbf{PA}_i(P_1, \ldots, P_{n_i}) \in [\![\bigcup]\!] \Leftrightarrow \forall j \in \{0, \ldots, n_i\},\ P_j \in SN_{\bigcup}$ *(pour* $1 \leq i \leq 6$*)*

Ces points se voient directement par la construction, après avoir remarqué que

$$[\![A]\!] = NEG_A([\![A^\perp]\!]) = IND_A(Neg_A([\![A^\perp]\!])) = Neg_A([\![A^\perp]\!]) \cup Ind_A([\![A]\!])$$

3.2 Preuve de la forte normalisation

3.2.1 forte normalisation des PA-termes

Par la suite, nous allons avoir besoin de la

Propriété : *Les* **PA***-termes sont fortement normalisables. Plus précisément, soit* t *un* **PA***-terme. Alors* t *est fortement normalisable.*

Avant d'en donner la preuve, il va nous falloir donner quelques lemmes, après avoir donné une définition pour des candidats. Nous nous plaçons dans cette sous-section dans le cadre des **PA**-termes.

Définition : *Soit* $\| - \|$ *une application qui à un type* C *associe un ensemble de termes de type* C *définie par induction sur la construction de* C *par :*

$$\|Int\| := SN_{Int}$$

$$\|A \to B\| := Var_{A \to B} \cup \{t : A \to B | \forall u \in \|A\|, (t)u \in \|B\|\}$$

Lemme : *Soit* C *un type. Alors* $Var_C \subseteq \|C\| \subseteq SN_C$.
Preuve : Par induction sur la construction de C. Si $C = Int$ c'est clair. Si $C = A \to B$, alors il est clair par définition que $Var_{A \to B} \subseteq \|A \to B\|$. Soit $t \in \|C\|$. Si t est une variable, il est clair que $t \in SN_C$. Sinon, soit $x \in Var_A$. Par induction, $x \in \|A\|$ et par définition de $\|A \to B\|$, nous obtenons que $(t)x \in \|B\|$. Or par hypothèse d'induction, $\|B\| \subseteq SN_B$, donc $(t)x \in SN_B$ et par conséquent, en tant que sous-terme, $t \in SN_{A \to B} = SN_C$. □

Lemme : *Soient* C *un type, et* a, b *et* $\vec{c}$ *des termes fortement normalisables. Alors*

$$a[b/x]\vec{c} \in \|C\| \Rightarrow (\lambda x.a)b\vec{c} \in \|C\|$$

Preuve : Par induction sur la construction de C. Si $C = Int$, nous prouvons la contraposée : Si $(\lambda x.a)b\vec{c} \notin \|Int\| = SN_{Int}$ et comme a, b et $\vec{c}$ sont fortement normalisables, une réduction infinie passe nécessairement par la réduction du rédex de tête :

$$(\lambda x.a)b\vec{c} \to (\lambda x.a')b'\vec{c}' \to a'[b'/x]\vec{c} \to \ldots$$

Or $a \to a'$, $b \to b'$ et $\vec{c} \to \vec{c}'$, donc

$$(\lambda x.a)b\vec{c} \to a[b/x]\vec{c} \to a'[b/x]\vec{c}' \to \ldots$$

Donc $a[b/x]\vec{c} \notin SN_{Int}$.

Si $C = A \to B$, et si $a[b/x]\vec{c} \in \|C\|$, on veut que $(\lambda x.a)b\vec{c} \in \|A \to B\|$. Pour cela, soit $u \in \|A\|$, il suffit de prouver que $(\lambda x.a)b\vec{c}u \in \|B\|$. Par induction, il suffit d'avoir $a[b/x]\vec{c}u \in \|B\|$, ce qui est vrai par définition de $\|A \to B\|$ comme $a[b/x]\vec{c} \in \|A \to B\|$ et $u \in \|A\|$. □

Lemme : *Soient C un type et u, p, f et $\vec{v}$ des termes fortement normalisables. Alors*

$$(a) u \to 0, (p)\vec{v} \in \|C\| \Rightarrow \mathbf{Rec}(u, p, f)\vec{v} \in \|C\|$$

$$(b) u \to \mathbf{s}u', f(u')\mathbf{Rec}(u', p, f)\vec{v} \in \|C\| \Rightarrow \mathbf{Rec}(u, p, f)\vec{v} \in \|C\|$$

$$(c) u \text{ n'est ni dans le cas } (a) \text{ ni le cas } (b) \Rightarrow \mathbf{Rec}(u, p, f)\vec{v} \in \|C\|$$

Preuve : Par induction sur la construction de C.
- Si $C = Int$, par contraposée, supposons que $\mathbf{Rec}(u, p, f)\vec{v} \notin \|Int\| = SN_{Int}$. Du fait que u, p, f et $\vec{v}$ soient fortement normalisables, une réduction infinie passe nécessairement par une réduction de **Rec**, et pour cela on est nécessairement pour u dans le cas (a) ou (b). Donc (c) est prouvé.
Si u est dans le cas (a), alors la réduction infinie est du type

$$\mathbf{Rec}(u, p, f)\vec{v} \to \mathbf{Rec}(0, p', f')\vec{v}' \to (p')\vec{v}' \to \dots$$

Or $p \to p'$, $\vec{v} \to \vec{v}'$ et

$$\mathbf{Rec}(u, p, f)\vec{v} \to \mathbf{Rec}(0, p, f)\vec{v} \to (p)\vec{v} \to (p')\vec{v}' \to \dots$$

et $(p)\vec{v} \notin \|Int\|$. Donc (a) est prouvé.
Si u est dans le cas (b), alors la réduction infinie est du type

$$\mathbf{Rec}(u, p, f)\vec{v} \to \mathbf{Rec}(\mathbf{s}u', p', f')\vec{v}' \to f'(u')\mathbf{Rec}(u', p', f')\vec{v}' \to \dots$$

Or $p \to p'$, $f \to f'$, $\vec{v} \to \vec{v}'$ et

$$\mathbf{Rec}(u, p, f)\vec{v} \to \mathbf{Rec}(\mathbf{s}u', p, f)\vec{v} \to f(u')\mathbf{Rec}(u', p, f)\vec{v} \to f'(u')\mathbf{Rec}(u', p', f')\vec{v}' \dots$$

et $f(u')\mathbf{Rec}(u', p, f)\vec{v} \notin \|Int\|$. Donc (b) est prouvé.
- Si $C = A \to B$, voyons les cas pour u. Si l'on est dans le cas (a), on veut prouver que $\mathbf{Rec}(u, p, f)\vec{v} \in \|A \to B\|$. Pour cela, soit $a \in \|A\|$, il suffit de prouver que $\mathbf{Rec}(u, p, f)\vec{v}a \in \|B\|$. Or par hypothèse d'induction, il suffit de prouver, comme $u \to 0$, que $(p)\vec{v}a \in \|B\|$. Cela est vrai car $(p)\vec{v} \in [\![A \to B]\!]$ et $a \in \|A\|$.
Si l'on est dans le cas (b), comme précédemment, il suffit de prouver que pour $a \in \|A\|$, $\mathbf{Rec}(u, p, f)\vec{v}a \in \|B\|$. Or par hypothèse d'induction, il suffit de prouver, comme $u \to \mathbf{s}u'$, que $f(u')\mathbf{Rec}(u', p, f)\vec{v}a \in \|B\|$. Cela est vrai car

$$f(u')\mathbf{Rec}(u', p, f)\vec{v} \in \|B\|$$

et $a \in \|A\|$.
Si l'on est dans le cas (c), il suffit encore de prouver que pour $a \in \|A\|$,

$$\mathbf{Rec}(u, p, f)\vec{v}a \in \|B\|$$

C'est ce que nous obtenons par hypothèse d'induction, car u est dans le cas (c), et $a \in \|A\| \subseteq SN_A$. □

Lemme : *Soit C un type, et soient $f \in \|Int \to C \to C\|$, $u \in \|Int\|$ et $p \in \|C\|$. Alors* $\mathbf{Rec}(u, p, f) \in \|C\|$.
Preuve : Nous allons utiliser le lemme précédent, en remarquant que les hypothèses de forte normalisation sont acquises par le premier lemme. Comme nous avons

$u \in \|Int\| = SN_{Int}$, il existe un entier maximum k tel que $u \to \mathbf{s}^k u'$ avec u' ne se réduisant pas à $\mathbf{s}u''$. La preuve se fait alors par récurrence sur l'entier k.
Si $k = 0$, alors il y a deux cas. Soit $u \to 0$ et comme $p \in \|C\|$ le point (a) nous permet de conclure, soit nous sommes dans le cas (c), et la conclusion est immédiate. Supposons le résultat vrai pour $k \geq 0$, prouvons le pour $k+1$. Soit donc un terme u avec $u \to \mathbf{s}^{k+1}u'$, u' ne se réduisant pas à $\mathbf{s}u''$ et $k+1$ maximum vérifiant cette propriété. Alors $u \to \mathbf{s}u_1$ avec u_1 vérifiant l'hypothèse de récurrence. Cela signifie que $\mathbf{Rec}(u_1, p, f) \in \|C\|$. Or $u_1 \in SN_{Int} = \|Int\|$ et $f \in \|Int \to C \to C\|$ donc $f(u_1) = (f)u_1 \in \|C \to C\|$ puis $f(u_1)\mathbf{Rec}(u_1, p, f) = ((f)u_1)\mathbf{Rec}(u_1, p, f) \in \|C\|$. Nous avons donc toutes les hypothèses de (b) qui nous permettent de conclure. □

Lemme : *Soient A et $(A_i)_{1\leq i\leq n}$ des types. Supposons que t soit un terme de type A, dont les variables libres sont dans $(t_i)_{1\leq i\leq n}$ avec t_i de type A_i pour tout i. Soient $u_i \in \|A_i\|$. Alors $t[u_i/x_i] \in \|A\|$.*
Preuve : Elle se fait par récurrence sur le nombre de règles utilisées pour typer t. Nous regardons la dernière.
- Si $t = x$, alors x est une variable libre de t donc est un des t_i. Par conséquent le résultat est immédiat.
- Si $t = 0$, alors t est de type Int, est comme $\|Int\| = SN_{Int}$ et que $0 \in SN_{Int}$, nous avons le résultat.
- Si $t = \mathbf{s}t'$, alors t, comme t', est de type Int. Par hypothèse de récurrence, $t'[u_i/x_i] \in \|Int\| = SN_{Int}$. Il est clair qu'il en est de même pour $t[u_i/x_i] = \mathbf{s}t'[u_i/x_i]$.
- Si $t = \lambda x.t'$, alors t est du type $A = C \to D$ et t' est du type D. Nous voulons prouver que $t[u_i/x_i] = \lambda x.t'[u_i/x_i] \in \|C \to D\|$. Pour cela, soit $u \in \|C\| \subseteq SN_C$, il nous suffit de prouver que $(\lambda x.t'[u_i/x_i])u \in \|D\|$. Or par récurrence, pour tout $v \in \|C\|$, $t'[u_i/x_i, v/x] \in \|D\|$. En particulier pour $v = x \in Var_C \in \|C\|$, nous avons $t'[u_i/x_i] \in \|D\| \subseteq SN_D$. Nous avons les hypothèses du lemme vu plus haut, il nous suffit alors de prouver que $t'[u_i/x_i][u/x] \in \|D\|$. Or par la récurrence nous obtenons que $t'[u_i/x_i][u/x] = t'[u_i/x_i, u/x] \in \|D\|$, d'où le résultat.
- Si $t = (w)v$, alors w est de type $B \to A$ et v de type B. par récurrence, $w[u_i/x_i] \in \|B \to A\|$ et $v[u_i/x_i] \in \|B\|$. Par définition de $\|B \to A\|$, nous obtenons bien que $t[u_i/x_i] = (w[u_i/x_i])v[u_i/x_i] \in \|A\|$.
- Si $t = \mathbf{Rec}(u, p, f)$, alors $t[u_i/x_i] = \mathbf{Rec}(u[u_i/x_i], p[u_i/x_i], f[u_i/x_i])$. Par récurence $u[u_i/x_i] \in \|Int\|$, $p[u_i/x_i] \in \|A\|$ et $f[u_i/x_i] \in \|Int \to A \to A\|$. Donc le résultat est immédiat par le lemme précédent. □

Nous pouvons enfin conclure sur la preuve de la propriété, car on peut prendre dans le lemme ci-dessus $u_i = t_i \in Var_{A_i} \subseteq \|A_i\|$. Alors $t[t_i/t_i] = t \in \|A\| \subseteq SN_A$.

3.2.2 propriétés des candidats

Redonnons maintenant les énoncés des lemmes, et étendons leur preuve à notre cas.

Lemme : *Soit C un type. Alors*

$$[\![C]\!] \subseteq SN_C$$

Preuve :
- Si $P = \mathbf{Ind}(v, Q, F)$, alors soient a et b des bornes pour v et Q respectivement. Soit c une borne pour F si $nf(v) = 0$ ou $nf(v)$ n'est pas un nombre, et une borne pour $F[\mathbf{s}^k 0, \mathbf{Ind}(\mathbf{s}^k 0, Q, F)]$ si $nf(v) = \mathbf{s}^{k+1}0$. les nombres a, b et c existent par les points *(v)* ou *(vi)* ou *(vii)* et le fait que les **PA**-termes sont fortement normalisables. Alors $a + b + 2c$ est une borne pour P, car dans le pire des cas, $nf(v) = s^{k+1}0$, il serait possible de réduire v, Q et F dans P en $s^{k+1}0$, Q' et F' respectivement sous forme normale, puis de réduire à $F'[\mathbf{s}^k 0, \mathbf{Ind}(\mathbf{s}^k 0, Q', F')]$. Donc $P \in SN_C$.

- Si $P = \sigma_t(Q)$, alors $C = \exists n.C_1$. Par le point *(ix)* et par hypothèse d'induction, nous obtenons que $Q \in SN_{C_1}$ puis que $P \in SN_C$.

- Si $P = \lambda_\forall n.Q$, alors $C = \forall n.C_1$. Par le point *(viii)* et par hypothèse d'induction, nous obtenons que $\forall t : Ind,\ Q[t/n] \in SN_{C_1}$, donc que $Q \in SN_{C_1}$. Il suit alors que $P \in SN_C$.

- Si $P = \mathbf{PA}_i(P_1, \ldots, P_{n_i})$ $(1 \leq i \leq 6)$, alors par le point *(x)* et par hypothèse d'induction, nous obtenons directement que $P \in SN_C$.

Grâce à ce lemme, nous pouvons remarquer que les points *(v)* et *(vi)* peuvent être simplifiés comme suit :

(v) *Si* $v : Int$ *avec* $nf(v) = 0$*, alors*
$\mathbf{Ind}(v, P, F) \in [\![A]\!] \quad \Leftrightarrow P \in [\![A]\!]$ *et* $F \in SN_A$

(vi) *Si* $v : Int$ *avec* $nf(v) = \mathbf{s}^{k+1}0$*, alors*
$\mathbf{Ind}(v, P, F) \in [\![A]\!] \quad \Leftrightarrow P \in SN_A,\ F[\mathbf{s}^k 0, \mathbf{Ind}(\mathbf{s}^k 0, P, F)] \in [\![A]\!]$

Lemme : Soient C un m-type et $P \in Term_C$. Alors

$$\textit{(a)}\ (\forall Q \in [\![C^\perp]\!] \quad P \star Q \in SN_\perp) \Rightarrow P \in [\![C]\!]$$

$$\textit{(b)}\ (\forall Q \in [\![C^\perp]\!] \quad Q \star P \in SN_\perp) \Rightarrow P \in [\![C]\!]$$

Preuve : Par symétrie des deux points, nous ne prouvons que le *(a)*. A cause de la présence des termes de la forme $\mathbf{Ind}(v, R, F)$, il nous faut faire une double induction, la seconde étant sur l'arbre de réduction de P.

- Si $P = \mathbf{Ind}(v, R, F)$ avec $nf(v) = 0$, alors par *(v)* nous avons à prouver que $R \in [\![C]\!]$ et $F \in SN_C$. Comme F est un sous-terme de $\mathbf{Ind}(v, R, F) \star Q$ qui est fortement normalisable par hypothèse, $F \in SN_C$. Ensuite, comme

$$\mathbf{Ind}(v, R, F) \star Q \rightarrow \mathbf{Ind}(0, R, F) \star Q \rightarrow_1 R \star Q$$

On peut affirmer grâce à l'hypothèse que pour tout $Q \in [\![C^\perp]\!]$, $R \star Q \in SN_\perp$. Alors par la seconde induction, nous obtenons $R \in [\![C]\!]$.

- Si $P = \mathbf{Ind}(v, R, F)$ avec $nf(v) = \mathbf{s}^{k+1}0$, alors par *(vi)* nous avons à prouver que $R \in SN_C$ et $F[\mathbf{s}^k 0, \mathbf{Ind}(\mathbf{s}^k 0, R, F)] \in [\![C]\!]$. Le premier fait est clair par l'hypothèse faite. De plus, toujours par hypothèse, et comme pour tout $Q \in [\![C^\perp]\!]$

$$\mathbf{Ind}(v, R, F) \star Q \rightarrow \mathbf{Ind}(\mathbf{s}^{k+1}0, R, F) \star Q \rightarrow_1 F[\mathbf{s}^k 0, \mathbf{Ind}(\mathbf{s}^k 0, R, F)] \star Q$$

nous obtenons que $F[\mathbf{s}^k 0, \mathbf{Ind}(\mathbf{s}^k 0, R, F)] \star Q \in SN_\perp$. Alors par la seconde induction, $F[\mathbf{s}^k 0, \mathbf{Ind}(\mathbf{s}^k 0, R, F)] \in [\![C]\!]$.

- Si $P = \mathbf{Ind}(v, R, F)$ et $nf(v)$ n'est pas un nombre, alors il est clair grâce à l'hypothèse que $R,\ F \in SN_C$. La conclusion vient grâce au point *(vii)*.

- Si $P = \lambda_\forall n.P_1$, alors $C = \forall n.C_1$. Par le point *(viii)*, il nous suffit de prouver que $P_1[t/n] \in [\![C_1]\!]$ pour tout $t : Int$. Grâce à l'hypothèse et au point *(ix)*, nous savons que pour tous $t : Int$ et $Q' \in [\![C_1^\perp]\!]$, $(\lambda_\forall n.P_1) \star \sigma_t(Q') \in SN_\perp$, puis par réduction que $P_1[t/n] \star Q' \in SN_\perp$. On applique alors l'hypothèse d'induction pour obtenir que $P_1[t/n] \in [\![C_1]\!]$ pour tout $t : Int$.

- Si $P = \sigma_t(P_1)$, alors $C = \exists n.C_1$. Par le point *(ix)*, il nous suffit de prouver que $P_1 \in [\![C_1]\!]$. Or, par le point *(viii)* et par hypothèse, nous avons $\forall Q' \in [\![C_1^\perp]\!]$, $\sigma_t(P_1) \star (\lambda_\forall m.Q') \in SN_\perp$ pour $m \notin FV(Q')$, car $Q'[t/m] = Q'$ pour tout $t : Int$. Ainsi nous avons aussi $\forall Q' \in [\![C_1^\perp]\!]$, $P_1 \star Q' \in SN_\perp$. Par hypothèse d'induction, nous obtenons alors $P_1 \in [\![C_1]\!]$.

- Si $P = \mathbf{PA}_i(P_1, \ldots, P_{n_i})$ $(1 \leq i \leq 6)$, alors le résultat est immédiat par le point *(x)* et par hypothèse d'induction. □

Lemme : *Soit C un type. Alors :*

$$P \in [\![C]\!],\ P \rightarrow_1 P' \Rightarrow P' \in [\![C]\!]$$

Preuve : De même que pour le lemme précédent, il nous faut faire une double induction, la seconde étant sur l'arbre de réduction de P (celui-ci étant fini d'après le premier lemme de la section).

- Si $P = \mathbf{Ind}(v, R, F)$ avec $nf(v) = 0$, alors il y a quatre cas différents :

+ Si $\mathbf{Ind}(v, R, F) \to \mathbf{Ind}(v, R, F')$, soit $F \to_1 F'$. Alors par hypothèse, et grâce au point *(v)*, $R \in [\![C]\!]$ et $F \in SN_C$. Alors $F' \in SN_C$ également, et le point *(v)* permet de conclure que $\mathbf{Ind}(v, R, F') \in [\![C]\!]$.

+ Si $\mathbf{Ind}(v, R, F) \to \mathbf{Ind}(v, R', F)$, soit $R \to_1 R'$. Alors $R \in [\![C]\!]$ par le point *(v)*. Comme

$$\mathbf{Ind}(v, R, F) \to \mathbf{Ind}(0, R, F) \to_1 R$$

nous obtenons par la seconde induction que $R' \in [\![C]\!]$. Alors grâce à l'hypothèse et au point *(v)*, $F \in SN_C$.

+ Si $\mathbf{Ind}(v, R, F) \to \mathbf{Ind}(v', R, F)$, soit $v \to_1 v'$. La conclusion vient immédiatement de l'hypothèse et du point *(v)*.

+ Si $\mathbf{Ind}(v, R, F) \to R$ (lorsque $v = 0$), la conclusion vient directement de l'hypothèse et du point *(v)*.

- Si $P = \mathbf{Ind}(v, R, F)$ avec $nf(v) = \mathbf{s}^{k+1}0$, alors il y a quatre cas différents :

+ Si $\mathbf{Ind}(v, R, F) \to \mathbf{Ind}(v, R, F')$, soit $F \to_1 F'$. Par hypothèse, et par le point *(vi)*, nous obtenons que $F[\mathbf{s}^k0, \mathbf{Ind}(\mathbf{s}^k0, R, F)] \in [\![C]\!]$ et $R \in SN_C$. Puisque

$$\begin{array}{rcl} \mathbf{Ind}(v, R, F) & \to & \mathbf{Ind}(\mathbf{s}^{k+1}0, R, F) \\ & \to_1 & F[\mathbf{s}^k0, \mathbf{Ind}(\mathbf{s}^k0, R, F)] \\ & \to_1 & F[\mathbf{s}^k0, \mathbf{Ind}(\mathbf{s}^k0, R, F')] \end{array}$$

Nous avons alors, par la seconde induction, que $F[\mathbf{s}^k0, \mathbf{Ind}(\mathbf{s}^k0, R, F')] \in [\![C]\!]$. De plus, puisque

$$\begin{array}{rcl} \mathbf{Ind}(v, R, F) & \to & \mathbf{Ind}(v, R, F') \\ & \to & \mathbf{Ind}(\mathbf{s}^{k+1}0, R, F') \\ & \to_1 & F'[\mathbf{s}^k0, \mathbf{Ind}(\mathbf{s}^k0, R, F')] \end{array}$$

on obtient également $F'[\mathbf{s}^k0, \mathbf{Ind}(\mathbf{s}^k0, R, F')] \in [\![C]\!]$. La conclusion vient alors grâce au point *(vi)*.

+ Si $\mathbf{Ind}(v, R, F) \to \mathbf{Ind}(v, R', F)$, soit $R \to_1 R'$. Grâce à l'hypothèse et au point *(vi)*, nous obtenons que $F[\mathbf{s}^k0, \mathbf{Ind}(\mathbf{s}^k0, R, F)] \in [\![C]\!]$ et $R \in SN_C$. Puisque

$$\begin{array}{rcl} \mathbf{Ind}(v, R, F) & \to & \mathbf{Ind}(\mathbf{s}^{k+1}0, R, F) \\ & \to_1 & F[\mathbf{s}^k0, \mathbf{Ind}(\mathbf{s}^k0, R, F)] \\ & \to_1 & F[\mathbf{s}^k0, \mathbf{Ind}(\mathbf{s}^k0, R', F)] \end{array}$$

Nous avons alors, par la seconde induction, que $F[\mathbf{s}^k0, \mathbf{Ind}(\mathbf{s}^k0, R', F)] \in [\![C]\!]$. Or $R \in SN_C$, donc $R' \in SN_C$, et le point *(vi)* nous permet de conclure.

+ Si $\mathbf{Ind}(v, R, F) \to \mathbf{Ind}(v', R, F)$, soit $v \to_1 v'$. Le résultat est immédiat grâce à l'hypothèse et au point *(vi)*.

+ Si $\mathbf{Ind}(v, R, F) \to F[\mathbf{s}^k0, \mathbf{Ind}(\mathbf{s}^k0, R, F)]$ (lorsque $v = \mathbf{s}^{k+1}$). Le résultat est immédiat grâce à l'hypothèse et au point *(vi)*.

- Si $P = \mathbf{Ind}(v, R, F)$ et $nf(v)$ n'est pas un nombre, alors il y a trois cas :

+ Si v est réduit, alors le point *(vii)* nous permet de conclure, car toute réduction dans P est une réduction dans R ou dans F

+ Si R est réduit en R', alors par le point *(vii)* $R \in SN_C$. Donc $R' \in SN_C$, et le point *(vii)* permet de conclure.

+ Si F est réduit an F', alors par le point *(vii)* $F \in SN_C$. Donc $F' \in SN_C$, et le point *(vii)* permet de conclure.

- Si $P = \lambda_\forall n.P_1$, la conclusion vient immédiatement de l'hypothèse et du point *(viii)*.

- Si $P = \sigma_t(Q)$, la conclusion vient immédiatement de l'hypothèse et du point *(ix)*.

- Si $P = \mathbf{PA}_i(P_1, \ldots, P_{n_i})$ (pour $1 \leq i \leq 6$), alors la conclusion vient du point *(x)*. $\square$

Lemme : *Soit C un m-type et soient $P \in [\![C]\!]$ et $Q \in [\![C^\perp]\!]$. Alors*

$$P \star Q \in SN_\perp$$

ce que nous savons être équivalent à

$$\forall S(P \star Q \longrightarrow_1 S \Rightarrow S \in SN_\perp)$$

Preuve : Il nous faut ajouter deux cas symétriques :

$$5. \begin{cases} P = \lambda_\forall n.P_1,\ Q = \sigma_t(Q_1) \text{ et } S = P_1[t/n] \star Q_1 \\ P = \sigma_t(P_1),\ Q = \lambda_\forall n.Q_1 \text{ et } S = P_1 \star Q_1[t/n] \end{cases}$$

Prouvons le premier cas, le second étant symétrique. Grâce au point *(viii)*, nous savons que $P_1[t/n] \in [\![A_1]\!]$ pour tout $t : Int$, et $Q_1 \in [\![A_1^\perp]\!]$. Par induction, nous obtenons alors $P_1[t/n] \star Q_1 \in SN_\perp$ pour tout $t : Int$, en particulier pour le t considéré. $\square$

Ici, il va nous falloir modifier l'énoncer du lemme afin de nous permettre de prouver le théorème pour $\lambda^{Sym}_{\mathbf{PA}-\bigcup}$.

Lemme : *Soit C un type et $P \in Term_C$ tel que*

$$FV(P) \subseteq \{x_1^{A_1}, \ldots, x_n^{A_n}\} \cup \{g_1, \ldots, g_m\}$$

*Où les x_i sont des variables de type A_i $(1 \leq i \leq n)$ et les g_j sont des variables **PA**-termes de **PA**-type G_j $(1 \leq j \leq m)$. Alors*
$\forall t_1 : G_1 \ldots t_m : G_m, \forall Q_1 \in [\![A_1]\!] \ldots Q_n \in [\![A_n]\!]$

$$P[t_1/g_1, \ldots, t_m/g_m][Q_1/x_1, \ldots, Q_n/x_n] \in [\![C]\!]$$

Preuve : Par double induction : la première induction sur la structure de P, et la seconde sur la somme des valeurs des nombres dans P. Nous simplifierons les notations en écrivant $[R/y]$ au lieu de $[R_1/y_1, \ldots, R_n/y_n]$. Nous rappelons que dans $\lambda^{Sym}_{\mathbf{PA}-\bigcup}$, $C = C[t/g]$ pour tout type C, variable **PA**-terme g et **PA**-terme t.

- Si P est une variable, il existe i tel que $P = x_i$ et $C = A_i$ (ou j tel que $P = g_j$ et $C = G_j$). Il suit alors que
$P[t/g][Q/x] = Q_i \in [\![A_i]\!] = [\![C]\!]$ (ou $P[t/g][Q/x] = t_j \in [\![G_j]\!] = [\![C]\!]$).

- Si $P = \langle P_1, P_2 \rangle$, alors $C = C_1 \wedge C_2$ et par hypothèse d'induction, pour $i = 1, 2$, $P_i[t/g][Q/x] \in [\![C_i]\!]$. Alors par *(ii)*, $P[t/g][Q/x] \in [\![C]\!]$.

- Si $P = \sigma_i(P_i)$, alors $C = C_1 \vee C_2$ et en utilisant l'hypothèse d'induction, on a $P_i[t/x][Q/x] \in [\![C_i]\!]$. Alors par *(iii)*, $P[t/x][Q/x] \in [\![C]\!]$.

- Si $P = P_1 \star P_2$, alors $C = \perp$. Si $P_1 \in Term_{C_1}$, on a $P_2 \in Term_{C_1^\perp}$. Ainsi par hypothèse d'induction, $P_1[t/g][Q/x] \in [\![C_1]\!]$ et $P_2[t/g][Q/x] \in [\![C_1^\perp]\!]$. Et grâce au lemme précédent, $P[t/g][Q/x] \in [\![\perp]\!] = [\![C]\!]$.

- Si $P = \lambda y.P_1$, alors $(\lambda y.P_1)[t/g][Q/x] = \lambda y.P_1[t/g][Q/x]$, puisque l'on sait, quitte à renommer les variables liées, que $y \notin \{x_1, \ldots, x_n\}$. Par le point *(iv)*, il nous suffit de prouver que pour tout $R \in [\![C^\perp]\!]$, $P_1[t/g][Q/x, R/y] \in [\![\perp]\!]$, ce qui est immédiat par hypothèse d'induction.

- Si $P = \mathbf{Ind}(v, R, F)$ avec $nf(v) = 0$, alors par hypothèse d'induction nous obtenons que $R[t/g][Q/x]$, $F[t/g][Q/x] \in [\![C]\!]$. De plus, par un lemme précédent, $F[t/g][Q/x] \in SN_C$. Comme

$$\mathbf{Ind}(v, R, F)[t/g][Q/x] = \mathbf{Ind}(v[t/g][Q/x], R[t/g][Q/x], F[t/g][Q/x]$$

et $nf(v) = nf(v[t/g][Q/x]) = 0$, le point *(v)* permet de conclure que $P[t/g][Q/x] \in [\![C]\!]$.

- Si $P = \mathbf{Ind}_{n,y}(v, R, F)$ avec $nf(v) = \mathbf{s}^{k+1}0$, alors par hypothèse d'induction nous obtenons que $R[t/g][Q/x]$, $F[t/g][Q/x] \in [\![C]\!]$. Par le point *(vi)*, il nous suffit de prouver que

$$F[t/g][Q/x][\mathbf{s}^k 0, \mathbf{Ind}_{n,y}(\mathbf{s}^k 0, R, F)[t/g][Q/x]] \in [\![C]\!]$$

Or avec nos conventions d'écriture, ce dernier terme est

$$F[t/g, \mathbf{s}^k 0][Q/x, \mathbf{Ind}_{n,y}(\mathbf{s}^k 0, R, F)[t/g][Q/x]/y]$$

et n et y sont liés dans $\mathbf{Ind}_{n,y}(\mathbf{s}^k 0, R, F)$. De plus, par la seconde hypothèse d'induction, $\mathbf{Ind}_{n,y}(\mathbf{s}^k 0, R, F)[t/g][Q/x] \in [\![C]\!]$, ce qui nous permet d'appliquer l'hypothèse d'induction.

- Si $P = \mathbf{Ind}_{n,y}(v, R, F)$ et $nf(v)$ n'est pas un nombre, alors l'hypothèse d'induction nous donne $R[t/g][Q/x]$, $F[t/g][Q/x] \in [\![C]\!]$. D'où, grâce à un lemme précédent, $R[t/g][Q/x]$, $F[t/g][Q/x] \in SN_C$. Alors par le point *(vii)*, nous concluons que $P[t/g][Q/x] \in [\![C]\!]$.

- Si $P = \mathbf{PA}_i(P_1, \ldots, P_{n_i})$ (pour $1 \leq i \leq 6$), la conclusion provient directement du point *(x)*. $\square$

3.2.3 Preuve du théorème

Nous pouvons maintenant prouver le théorème de forte normalisation pour le $\lambda^{Sym}_{\mathbf{PA}-\bigcup}$-calcul et donc pour le $\lambda^{Sym}_{\mathbf{PA}}$-calcul. Rappelons le théorème :

Théorème : *Soit C un type. Alors*

$$Term_C = SN_C$$

Preuve : Soit $P : C$ un terme avec $FV(P) = \{x_1^{A_1}, \ldots, x_n^{A_n}\} \cup \{g_1, \ldots, g_m\}$. Par le point *(i)*, nous obtenons $x_i \in [\![A_i]\!]$ pour $i = 1, \ldots, n$. Par suite, le dernier lemme de la section précédente nous permet d'obtenir que $P = P[x/x][g/g] \in [\![C]\!]$. Un lemme précédent nous permet de conclure que $P \in SN_C$. $\square$

Annexe A

Extensions du λ-calcul symétrique

Dans l'article [Babe], et comme nous venons de le voir, le $\lambda_{\mathbf{PA}}^{Sym}$-calcul a la propriété de forte normalisation.

Nous allons donner brièvement dans cette annexe deux autres preuves de forte normalisation, pour des extensions de calculs au second ordre. Nous essaierons de montrer quels sont les points communs dans la structure et la méthode, ainsi que les différences.

Il ne s'agira pas de refaire les preuves en entier, mais bien d'en dégager les idées principales. La première preuve provient de [Pa], et se rapproche de celle de [Babe], le calcul étant le même, auquel a été oté l'arithmétique. La seconde, qui provient de [Ya], est légèrement différente, étant faite sur un $\lambda\mu$-calcul, notre calcul étant apparenté à du μ-calcul.

A.1 Extension du λ_{Prop}^{Sym}-calcul au second ordre

Le texte [Pa] étend le λ_{Prop}^{Sym}-calcul au second ordre. Le calcul est le même que le $\lambda_{\mathbf{PA}}^{Sym}$ que nous connaissons déjà, toutes les règles et constructions contenant l'arithmétique ayant été oté. Une différence de structure apparaît dans la preuve. En effet, alors que dans ce que nous avons vu précédemment les candidats de réductibilité étaient définis en même temps que l'interprétation des types, ici il y a deux étapes.

On définit tout d'abord les candidats de réductibilité, en utilisant une notion de couple, notion déjà sous-jacente dans la preuve que nous connaissons.

Définitions : *Soit $\mathcal{T}$ l'ensemble des termes. Pour C et D dans $\mathcal{P}(\mathcal{T})$, $\mathcal{S} \subseteq \mathcal{P}(\mathcal{T})$, on définit :*

$$C \times D := \{\langle u, v\rangle; u \in C, v \in D\}$$

$$C + D := \{\sigma_1(u); u \in C\} \cup \{\sigma_2(v); v \in D\}$$

$$\neg(C) := \{\lambda x.u; \text{ pour tout } v \in C, u[v/x] \in SN\}$$

$$\cap\mathcal{S} := \{\lambda_\forall u; \text{pour tout } C \in \mathcal{S}, u \in C\}$$

$$\cup\mathcal{S} := \{\sigma_t(u); \text{il exite } C \in \mathcal{S}, u \in C\}$$

Pour une fonction $F : \mathcal{P}(\mathcal{T}) \to \mathcal{P}(\mathcal{T})$ une application croissante, nous noterons $\mu X.F(X)$ son plus petit point-fixe.

Soient C et D dans $\mathcal{P}(\mathcal{T})$, alors on définit

$$Neg_D(C) = Var \cup D \cup \neg(C)$$

Pour D donné, l'application Neg_D est décroissante, donc $Neg_D \circ Neg_{D'}$ est croissante et $\mu X.Neg_D \circ Neg_{D'}(X)$ est bien défini.
On définit alors l'ensemble $\mathcal{R}$ de paires de réductibilité comme étant le plus petit sous-ensemble de $\mathcal{P}(\mathcal{T}) \times \mathcal{P}(\mathcal{T})$ tel que :
1) $(\mu X.Neg_\emptyset \circ Neg_\emptyset(X), Neg_\emptyset(\mu X.Neg_\emptyset \circ Neg_\emptyset(X))) \in \mathcal{R}$
2) Si (C, C') et (D, D') sont dans $\mathcal{R}$, alors
$(\mu X.Neg_{C\times D} \circ Neg_{C'+D'}(X), Neg_{C'+D'}(\mu X.Neg_{C\times D} \circ Neg_{C'+D'}(X))) \in \mathcal{R}$
3) Si $\emptyset \neq \mathcal{F} \subseteq \mathcal{R}$, $\mathcal{S} = p_1\mathcal{F}$ et $\mathcal{S}' = p_2\mathcal{F}$, où p_1 et p_2 sont les projections, alors
$(\mu X.Neg_{\cap\mathcal{S}} \circ Neg_{\cup\mathcal{S}'}(X), Neg_{\cup\mathcal{S}'}(\mu X.Neg_{\cap\mathcal{S}} \circ Neg_{\cup\mathcal{S}'}(X))) \in \mathcal{R}$
4) Si $(C, C') \in \mathcal{R}$ alors $(C', C) \in \mathcal{R}$.
L'ensemble $\mathcal{R}_0$ des candidats de réductibilité est $\mathcal{R}_0 = p_1\mathcal{R} = p_2\mathcal{R}$.

On peut remarquer que les ensembles $C \times D$, $\neg(C)$,... sont une sorte de généralisation des application $Pair$, *Lambda*,... que nous connaissons déjà. Elles permettent alors une définition plus simple de l'opérateur Neg. L'idée qui se trouve dans les couples est d'avoir comme nous le connaissons un travail sur les couples $([\![A]\!], [\![A^\perp]\!])$, qui seront notés dans le cas présent $([\![A]\!]^\alpha, [\![\neg A]\!]^\alpha)$.

A partir de la définition qui a été donnée ci-dessus, on peut prouver le lemme suivant.

Lemme : *Soit $(C, C') \in \mathcal{R}$, alors on est dans l'un des cas suivants.*
1) $C = Neg_\emptyset(C')$ et $C' = Neg_\emptyset(C)$
2) $C = Neg_{D_1\times D_2}(C')$ et $C' = Neg_{D'_1+D'_2}(C)$ où $(D_i, D'_i) \in \mathcal{R}$, $i = 1, 2$
3) $C = Neg_{D_1+D_2}(C')$ et $C' = Neg_{D'_1\times D'_2}(C)$ où $(D_i, D'_i) \in \mathcal{R}$, $i = 1, 2$
4) $C = Neg_{\cap\mathcal{S}}(C')$ et $C' = Neg_{\cup\mathcal{S}'}(C)$ où $\mathcal{S} = p_1\mathcal{F}$, $\mathcal{S}' = p_2\mathcal{F}$ et $\mathcal{F} \subseteq \mathcal{R}$
5) $C = Neg_{\cup\mathcal{S}}(C')$ et $C' = Neg_{\cap\mathcal{S}'}(C)$ où $\mathcal{S} = p_1\mathcal{F}$, $\mathcal{S}' = p_2\mathcal{F}$ et $\mathcal{F} \subseteq \mathcal{R}$.

Lemme : *Soient $(C, C') \in \mathcal{R}$, et $u \in \mathcal{T}$ Alors*

$$\lambda x.u \in C \Leftrightarrow \lambda x.u \in \neg(C')$$

Le lemme ci-dessus est l'équivalent du point *(iv)* des propriétés que nous avons vues. Avec ceci, il est maintenant possible de faire la preuve des lemmes que nous connaissons déjà, ceci de la même manière que précédemment. Enonçons les lemmes.

Lemme : *Si $C \in \mathcal{R}_0$ alors $Var \subseteq C \subseteq SN$.*

Lemme : *Soient $(C, C') \in \mathcal{R}$, et u, $u' \in \mathcal{T}$. Si $u \in C$ et $u \to u'$, alors $u' \in C$.*

Lemme : *Soient $(C, C') \in \mathcal{R}$, et u, $u' \in \mathcal{T}$. Si $u \in C$ et $u' \in C'$, alors $u \star u' \in SN$.*

A partir de maintenant, il faut introduire l'interprétation des formules.

Définition : *Soit Δ l'ensemble des types atomiques et des types atomiques de négation. Une valuation α est une application de Δ dans $\mathcal{R}_0$ telle que pour tout type de variable X, $(\alpha(X), \alpha(\neg X)) \in \mathcal{R}$. Pour $U \in \Delta$ et $C \in \mathcal{R}_0$, on note $\alpha[C/U]$ la valuation α' définie par $\alpha'(U) = C$ et $\alpha'(V) = \alpha(V)$ pour $V \neq U$.*
On généralise une valuation α à l'interprétation $[\![A]\!]^\alpha$ d'un m-type A définie de la manière suivante :

$[\![X]\!]^\alpha := \alpha(X)$ *pour X un type de variable.*
$[\![\neg X]\!]^\alpha := \alpha(\neg X)$ *pour X un type de variable.*
$[\![A \wedge B]\!]^\alpha := \mu X.Neg_{[\![A]\!]^\alpha\times[\![B]\!]^\alpha} \circ Neg_{[\![\neg A]\!]^\alpha+[\![\neg B]\!]^\alpha}(X)$.
$[\![A \vee B]\!]^\alpha := Neg_{[\![A]\!]^\alpha+[\![B]\!]^\alpha}([\![\neg A \wedge \neg B]\!]^\alpha)$
$[\![\forall X A]\!]^\alpha := \mu X.Neg_{\cap\mathcal{S}} \circ Neg_{\cup\mathcal{S}'}(X)$

$[\![\exists X A]\!]^\alpha := Neg_{\cup\mathcal{S}}([\![\forall X \neg X]\!]^\alpha)$.
où $\mathcal{S} = \{[\![A]\!]^{\alpha[C/X, C'/\neg X]}; (C, C') \in \mathcal{R}\}$ *et* $\mathcal{S}' = \{[\![\neg A]\!]^{\alpha[C/X, C'/\neg X]}; (C, C') \in \mathcal{R}\}$
On étend encore la définition aux types en définissant $[\![\bot]\!]^\alpha := SN$

Voici le lemme que nous annoncions plus haut.

Lemme : *Pour toute valuation* α *et m-type* A, $([\![A]\!]^\alpha, [\![\neg A]\!]^\alpha) \in \mathcal{R}$.

Une propriété nous permettant de vérifier que la notion d'interprétation est bien définie :

Lemme : *Soient* A *et* B *des m-types, et* α *et* α' *des valuations.*
(1) Si $\alpha(X) = \alpha'(X)$ *et* $\alpha(\neg X) = \alpha'(\neg X)$ *pour tout type de variable* X *libre dans* A*, alors* $[\![A]\!]^\alpha = [\![A]\!]^{\alpha'}$.
(2)$[\![A[B/X]]\!]^\alpha = [\![A]\!]^{\alpha[[\![B]\!]^\alpha/X, [\![\neg B]\!]^\alpha/\neg X]}$.

On peut enfin énoncer le dernier lemme et le théorème de forte normalisation pour le calcul du second ordre, qui se prouvent de la même manière que nous avons vue.

Lemme : *Soient* $A_1, A_2, \ldots, A_n$ *des m-types, et* C *un type. Soient* x_i *de type* A_i $(i = 1, \ldots, n)$*, et* t *de type* C *et de variables libres les* x_i*. Si* $u_i \in [\![A_i]\!]^\alpha$ $(i = 1, \ldots, n)$*, alors*

$$t[u_1/x_1, \ldots, u_n/x_n] \in [\![C]\!]^\alpha$$

Théorème : *Soient* $A_1, A_2, \ldots, A_n$ *des m-types, et* C *un type. Soient* x_i *de type* A_i $(i = 1, \ldots, n)$*, et* t *de type* C *et de variables libres les* x_i*. Alors* t *est fortement normalisable.*

A.2 Extension du $\lambda\mu$-calcul au second ordre

Ici, le calcul étant différent, il nous faut tout d'abord l'introduire. Il s'agit ici d'un $\lambda\mu$-calcul qui utilise des variables du premier ordre (notées $t, u, \ldots$) et les variables du second ordre, ou prédicats n-aires (notées $X^n, Y^n, \ldots$), et des variables (notées $\alpha, \beta, \ldots$). une difficulté rencontrée ici est que la négation n'est pas un opérateur involutif à priori, et qu'il va falloir travailler en conséquence.

Définition : *L'ensemble des termes du premier ordre constiste en la constante* 0*, la fonction unaire* S*, et le symbole de fonction* f *de toute fonction récursive primitive sur les entiers naturels. Une proposition du second ordre est construite sur la grammaire :*

$$A ::= t_1 = t_2 \,|\, X_i^n t_1 \ldots t_n \,|\, A \to A \,|\, \forall x_i A \,|\, \forall X_i^n A$$

Une formule est une proposition A *ou sa négation* $\bullet A$*, ou* $\bot$*. Les autres connecteurs sont définis en utilisant la construction du second ordre ; Par exemple,* $A \wedge B$ *est défini comme* $\forall X^0.(A \to B \to X) \to X$.

Définition : *L'ensemble des axiomes consiste en les axiomes d'égalité, les axiomes définissant les fonctions récursives primitives et la proposition* $\mathrm{s}0 = 0 \to \forall X.X$*. Ces axiomes peuvent être formulé par des règles atomiques, comme celles données par la construction des* **PA***-termes. Les* $\lambda\mu$*-termes sont construits par les règles suivantes*

$$\frac{}{axiom_i \,:\, A} \qquad \frac{\alpha^C}{\alpha \,:\, C}$$

$$\frac{M \,:\, \bullet A \quad N \,:\, A}{[M]N \,:\bot}\,[] \qquad \frac{M \,:\, A \to B \quad N \,:\, A}{MN \,:\, B}\,app.$$

$$\dfrac{\begin{matrix}[\alpha : A]\\ \vdots \\ M : \perp\end{matrix}}{\mu\alpha.M : \bullet A}\,\mu \qquad \dfrac{\begin{matrix}[\alpha : \bullet A]\\ \vdots \\ M : \perp\end{matrix}}{\mu\alpha.M : A}\,\mu \qquad \dfrac{\begin{matrix}[\alpha : A]\\ \vdots \\ M : B\end{matrix}}{\lambda\alpha.M : A \to B}\,\lambda$$

$$\dfrac{M : \forall x A}{Mt : A[t/x]}\,app^1. \qquad \dfrac{M : \forall X A}{MT : A[T/X]}\,app^2.$$

$$\dfrac{M : A}{\lambda x.M : \forall x A}\,\lambda^1 \qquad \dfrac{M : A}{\lambda X.M : \forall X A}\,\lambda^2$$

Où dans les règles λ^1 et λ^2, x ou X n'est pas libre dans M.

Définition : *Voici les règles de réductions associées. Soient β, γ et δ des variables nouvelles. Alors*
$(\lambda)(\lambda\alpha.M)N \to M[N/\alpha]$
$(\lambda_1)(\lambda x.M)t \to M[t/x]$
$(\lambda_2)(\lambda X^n.M)T \to M[T/X^n]$
$(\mu)[M]\mu\alpha.N \to N[M/\alpha]$ *et sa règle symétrique* $([\mu\alpha.M]N \to M[N/\alpha])$
$(\zeta)(\mu\alpha.M)N \to \mu\beta.M[\mu\gamma.[\beta](\gamma N)/\alpha]$ *et sa règle symétrique*
$(\zeta_1)(\mu\alpha.M)t \to \mu\beta.M[\mu\gamma.[\beta](\gamma t)/\alpha]$
$(\zeta_2)(\mu\alpha.M)T \to \mu\beta.M[\mu\gamma.[\beta](\gamma T)/\alpha]$
Nous noterons $\to_1$ l'union de ces réductions, et $\to$ la clôture réflexive et transitive de $\to_1$. On note $\omega(M)$ la longueur de la plus longue réduction si elle existe, sinon $\omega(M)$ n'est pas défini. Alors M est fortement normalisable si et seulement si $\omega(M)$ existe.

A partir de maintenant commence la preuve du théorème de forte normalisation, en commençant par quelques notions.

Définition :
1. Un terme qui commence par μ est appelé une μ-forme.
2. Pour un ensemble S de termes de type C, on définit $Cl(S)$ comme étant le plus petit ensemble qui satisfait :
(a) $S \cup Var_C \subseteq Cl(S)$.
(b) si pour tout L tel que $MN \to_1 L$ on a $L \in Cl(S)$, alors $MN \in Cl(S)$.
(c) Soit t une variable du premier ordre. Si pour tout L tel que $Mt \to_1 L$ on a $L \in Cl(S)$, alors $Mt \in Cl(S)$.
(d) Soit T une variable du second ordre. Si pour tout L tel que $MT \to_1 L$ on a $L \in Cl(S)$, alors $MT \in Cl(S)$.
3. L'ensemble des termes fortement normalisables de type $\perp$ est noté encore $\perp$.
4. Pour un ensemble S de termes de type $C \neq \perp$ on note

$$\bullet S := \{\mu\alpha.M | \forall N \in S, M[N/\alpha] \in \perp\}$$

où α est une variable de type C et M est de type $\perp$.
5. On définit l'opérateur D par $D(\chi) = Cl(\chi \cup \bullet\bullet\chi)$. Pour tout ordinal κ,

$$D^\kappa(\chi) := D(\bigcup_{\tau<\kappa} D^\tau(\chi))$$

On remarque que l'opérateur $\bullet$ est décroissant (c'est l'analogue de l'opérateur *Lambda* de [Babe] et de l'opérateur $\neg$ de [Pa]), et par suite que D est croissant.

Définition : *Soit ω_1 le plus petit ordinal non dénombrable. Soit S un ensemble de termes de type A fortement normalisables. On suppose de plus que S ne contient pas de μ-formes et est clos pour la relation de réduction. Alors $D^{\omega_1}(S)$ est appelé* *candidat de réductibilité de A. Nous admettrons que par la monotonie de D, un candidat de réductibilité est un point fixe de D. L'ensemble des candidats de A est noté R_A, et R est la réunion de tous les R_A.*

Lemme : *Soit $\mathcal{R}$ un candidat de réductibilité, $\mathcal{R} = D^{\omega_1}(S)$. Alors :*
1. Tous les termes dans $\mathcal{R}$ sont fortement normalisables.
2. $\mathcal{R} = Cl(S \cup \bullet \bullet \mathcal{R})$.
3. Pour $M \in \bullet\mathcal{R}$ et $N \in \mathcal{R}$, $[M]N \in\perp$.

Le point 3. du lemme ci-dessus correspond au lemme que nous connaissons, affirmant que si $M \in [\![A]\!]$ et $N \in [\![A^\perp]\!]$, alors $M \star N \in SN_\perp$. Pour le voir, il suffit de prouver que si $[M]N \to_1 L$, alors L est fortement normalisable. Cela se prouve de la même manière que celle que nous connaissons : par induction sur la somme $\omega(M) + \omega(N)$.

Le lemme précédent suffisait dans les autres cas pour prouver le lemme de substitution. Ici, le calcul étant plus riche, il faut définir d'autres candidats, et prouver un lemme semblable pour ceux-ci.

Définition : *Soient $\mathcal{A} \in R_A$ et $\mathcal{B} \in R_B$. Soient $(t_i)_{i\in I}$ et $(T_j)_{j\in J}$ des familles non vides du premier et second ordre respectivement. Soient de plus $\mathcal{A}_i$ et $\mathcal{A}_j$ des candidats pour $A[t_i/x]$ ($i \in I$) et $A[T_j/X]$ ($j \in J$) respectivement. On définit alors les candidats $\mathcal{A} \to \mathcal{B}$, $\wedge^1_{i\in I}\mathcal{A}_i$ et $\wedge^2_{j\in J}\mathcal{A}_j$ par :*

$$\mathcal{A} \to \mathcal{B} := D^{\omega_1}(L(\mathcal{A}, \mathcal{B}))$$

$$\bigwedge_{i\in I}^{1} \mathcal{A}_i := D^{\omega_1}(\Pi^1_{i\in I}\mathcal{A}_i)$$

$$\bigwedge_{j\in J}^{2} \mathcal{A}_j := D^{\omega_1}(\Pi^2_{j\in J}\mathcal{A}_j)$$

où

$$L(\mathcal{A}, \mathcal{B}) := \{\lambda\alpha^A.M | \forall N \in A, M[N/\alpha^A] \in \mathcal{B}\}$$

$$\Pi^1_{i\in I}\mathcal{A}_i := \{\lambda x.M | \forall i \in I, M[t_i/x] \in \mathcal{A}_i\}$$

$$\Pi^2_{j\in J}\mathcal{A}_j := \{\lambda X.M | \forall j \in J, M[T_j/X] \in \mathcal{A}_j\}$$

Lemme :
1. Soit $\mathcal{A} \in R_A$ et $\mathcal{B} \in R_B$. Si $M \in \mathcal{A} \to \mathcal{B}$ et $N \in \mathcal{A}$, alors $MN \in \mathcal{B}$.
2. Soient $(t_i)_{i\in I}$ et $(\mathcal{A}_i)_{i\in I}$ comme ci-dessus. Si $M \in \wedge^1_{i\in I}\mathcal{A}_i$ alors $Mt_i \in \mathcal{A}_i$.
3. Soient $(T_j)_{j\in J}$ et $(\mathcal{A}_j)_{j\in J}$ comme ci-dessus. Si $M \in \wedge^2_{j\in J}\mathcal{A}_j$ alors $MT_j \in \mathcal{A}_j$.

Preuve : 1. Posons $\mathcal{A} = D^{\omega_1}(S)$. Soit κ le plus petit ordinal tel que $M \in D^\kappa(L(\mathcal{A}, \mathcal{B}))$ et τ le plus petit ordinal tel que $N \in D^\tau(S)$. Par induction sur $(\kappa + \tau, \omega(M) + \omega(N))$ nous prouvons que si $MN \to_1 L$, alors $L \in \mathcal{B}$, ce qui suffit à la preuve (voir la définition de Cl).

- Si $L = M'N'$, avec $M \to_1 M'$ et $N = N'$ ou $N \to_2 N'$ et $M = M'$, le résultat est immédiat par induction sur $\omega(M) + \omega(N)$).

- Si $M = \lambda\alpha.M_1$ et $L = M_1[N/\alpha]$, comme $M \in L(\mathcal{A}, \mathcal{B})$, nous avons le résultat par définition.

- Si $M = \mu\alpha.M_1$ et L est obtenu par une des réductions (ζ), alors L est de la forme $\mu\beta.M_1[\mu\gamma.[\beta](\gamma N)/\alpha]$. Soient $J \in \bullet\mathcal{B}$ et $K \in D^{\kappa_1}(L(\mathcal{A}, \mathcal{B}))$ avec $\kappa_1 < \kappa$. Nous supposons que κ_1 est le plus petit ordinal tel que $D^{\kappa_1}(L(\mathcal{A}, \mathcal{B}))$ contienne K. Par hypothèse d'induction on a $KN \in \mathcal{B}$. Il suit que $[J](KN) \in\perp$. K et κ_1 étant arbitraires,

$$\mu\gamma.[J](\gamma N) \in \bullet \cup_{\kappa_1<\kappa} D^{\kappa_1}(L(\mathcal{A}, \mathcal{B}))$$

Comme M est une μ-forme, $M \in \bullet\bullet \cup_{\kappa_1<\kappa} D^{\kappa_1}(L(\mathcal{A},\mathcal{B}))$. Et par par suite on a $M_1[\mu\gamma.[J](\gamma N)/\alpha] \in \perp$, et comme $J \in \bullet\mathcal{B}$, nous obtenons que $L \in \bullet\bullet\mathcal{B}$. Comme $\bullet\bullet\mathcal{B} \subset \mathcal{B}$, la conclusion suit.

- Si $N = \mu\alpha.N_1$ et L est obtenu par une des réductions (ζ), alors on opère de manière analogue au cas précédent pour conclure.

2. et 3. se prouvent manière analogue. □

Il ne nous reste plus qu'à introduire les interprétations avant de donner le lemme de substitution qui permettra de conclure.

Définition : *Notons $\mathcal{T}$ l'ensemble des termes du premier ordre, $\mathcal{F}^n$ celui des fonctions de $\mathcal{T}^n$ dans R. Soit $[\![-]\!]$ une application qui associe à variable du premier ordre (resp. une variable d'ordre n) un élément de $\mathcal{T}$ (resp. de $\mathcal{F}^n$). On étend cette fonction à tous les types en posant $[\![\perp]\!] = \perp$ et par induction sur la construction :*

$$[\![\bullet A]\!] = \bullet[\![A]\!]$$

$$[\![A \to B]\!] = [\![A]\!] \to [\![B]\!]$$

$$[\![\forall x A]\!] = \bigwedge_{t\in\mathcal{T}}^{1} [\![A]\!]_{[t/x]}$$

$$[\![\forall X^n A]\!] = \bigwedge_{f\in\mathcal{F}^n}^{2} [\![A]\!]_{[f/X^n]}$$

où $[\![-]\!]_{[a/b]}$ est défini par $[\![b]\!]_{[a/b]} = a$ et pour $c \neq b$, $[\![c]\!]_{[a/b]} = [\![c]\!]$.

Lemme : *Soit M un terme de type A, et de variables libres du premier ordre $(x_i)_{1\leq i\leq m}$ et du second ordre $(X_j)_{1\leq j\leq n}$, et de variables $(\alpha_k^{A_k})_{1\leq k\leq p}$. Soit $[\![-]\!]$ une application comme ci-dessus. Pour tout $1 \leq j \leq n$, puis $t_1, \ldots t_r$ (où r est l'arité de X_j), $[\![X_j]\!]t_1 \ldots t_r \in R_{B_j[t/x]}$. Prenons enfin $N_k \in [\![A_k]\!]$ pour $1 \leq k \leq p$. Soit $\tilde{M}$ défini à partir de M par la substitution des x_i par les $[\![x_i]\!]$, des X_j par les B_j puis des α_k par les N_k. Alors $\tilde{M} \in [\![A]\!]$.*

En conséquence de ce lemme et grâce au premier de cette section, on a alors le

Théorème : *Tous les termes sont fortement normalisables.*

A.3 Conclusion

Si la mise en place de la preuve peut varier dans chaque cas, il reste une idée générale. Chaque preuve utilise de manière importante la notion de point fixe pour définir les candidats de réductibilité. Ces candidats permettent alors par un argument de substitution d'obtenir le résultat souhaité, à savoir la forte normalisation de chaque calcul du second ordre introduit ici.

Bibliographie

[BaBe] F. Barbanera et S. Berardi, *A Symmetric Lambda Calculus for "Classical" Program Extraction*, Université de Turin (1996).

[CoLa] R. Cori et D. Lascar, *Logique mathématique*, tomes 1 & 2, Dunod (2003).

[DaNoRa] R. David, K. Nour et C. Raffalli, *Introduction à la logique*, Dunod (2001).

[Pa] M. Parigot, *Strong normalisation of second order symmetric λ-calculus*, Université Paris 7 (2000).

[Ya] Y. Yamagata, *Strong normalisation of second order symmetric lambda-mu calculus*, University of Tokyo (2001).