Lambda-calcul simplement typé

Pierre Lescanne

25 avril 2007 – 14 : 22

Le paradoxe du barbier

$\Omega \equiv (\lambda x.xx)(\lambda x.xx)$ et $\mathbf{Y} \equiv \lambda f.(\lambda x f(xx))\ (\lambda x f(xx))$ contiennent des termes qui s'appliquent à eux-mêmes.
Le paradoxe du barbier est :

Le barbier rase tous ceux qui ne se rasent pas eux-mêmes.

Qui rase le barbier ?

Pour éviter les paradoxes, on cherche à éviter de tels termes.
On va donc typer les termes.
Typer est aussi bien pour la programmation.

Les objectifs du typage

Le typage a donc deux objectifs :

- préserver la correction, rien de mauvais ne peut arriver,
- préserver la terminaison, toutes les réductions se terminent.

En λ-calcul la terminaison s'appelle la normalisation forte.

Types, annotations des variables

Les types sont

- soit des types de base o,
- soit des types applications $\sigma \to \tau$.

Dans l'approche dite à la Church, les variables associées à un λ sont annotées par un type.

$$M, N \quad ::= \quad x \mid \lambda x^{\sigma}.M \mid M\ N$$

Soit Γ un ensemble de variables annotées par leurs types.

$$\Gamma \quad = \quad \{x_1^{\sigma_1}, ..., x_n^{\sigma_n}\}.$$

Jugements et règles

Un jugement de typage à la Church est une déclaration de la forme $\Gamma \vdash M : \sigma$.
On a alors les règles de typage suivante.

$$\frac{}{\Gamma \vdash x : \sigma \quad \text{si } x^\sigma \in \Gamma} \qquad \frac{\Gamma \cup \{x^\sigma\} \vdash M : \tau}{\Gamma \vdash \lambda x^\sigma . M : \sigma \to \tau.}$$

$$\frac{\Gamma \vdash M : \sigma \to \tau \quad \Gamma \vdash N : \sigma}{\Gamma \vdash M\ N : \tau}$$

Un exemple

A la Church on écrirait :

$$(\lambda f^{(o\to o)\to(o\to o)} x^{o\to o}.f(fx))\ (\lambda f^{o\to o} x^{o}.f(fx))$$

Les environnements

Dans l'approche à la Curry, on traite des termes usuels du lambda-calcul.
Il faut typer les variables libres, il faut donc faire des hypothèses sur les types de ces variables.
D'où la notion d'environnement.
Un environnement est un ensemble d'association de types à des variables.

$$\Gamma \equiv x_1 : \sigma_1, ..., x_n : \sigma_n$$

Les types

Un jugement est l'affirmation du type σ d'un terme M sous un certain environnement Γ :

$$\Gamma \vdash M : \sigma$$

Les règles

(Var) $\Gamma, x : \sigma \vdash x : \sigma$

(Abs) $$\frac{\Gamma, x : \sigma \vdash M : \tau}{\Gamma \vdash \lambda x.M : \sigma \to \tau}$$

(App) $$\frac{\Gamma \vdash M : \sigma \to \tau \qquad \Gamma \vdash N : \sigma}{\Gamma \vdash MN : \tau}$$

Exercices

Typez les termes

$B \equiv \lambda xyz.x(yz)$,

$I \equiv \lambda x.x$,

$C \equiv \lambda xyz.xzy$,

$K \equiv \lambda xy.x$,

$S \equiv \lambda xyz.xz(yz)$,

$\mathbf{p}\ \mathbf{2}\ \mathbf{2} \equiv (\lambda mn.m\ n)\ (\lambda fx.f(fx))\ (\lambda fx.f(fx))$

Exercices (suite)

1. Typez $II \equiv (\lambda x.x)(\lambda x.x)$.
2. Typez $\mathbf{2}\ \mathbf{2} \equiv (\lambda fx.f(fx))(\lambda fx.f(fx))$.
3. A-t-on le même type pour I (resp. $\mathbf{2}$) dans chaque cas ?

Conclusion : Le système de types simples n'est pas assez général. Il ne permet pas d'affecter un type unique à chaque terme.
Le type d'un même terme peut dépendre de son contexte.

$\boxed{\text{Si } \Gamma, x : \sigma \vdash M : \tau \text{ et } \Gamma \vdash N : \sigma \text{ alors } \Gamma \vdash M[x := N] : \tau}$

Démonstration : Par induction sur M.

$\boxed{\text{Si } \Gamma, x : \sigma \vdash M : \tau \text{ et } \Gamma \vdash N : \sigma \text{ alors } \Gamma \vdash M[x := N] : \tau}$

Démonstration : Par induction sur M.

• Si $M \equiv y$ alors Γ contient $y : \tau$ et $M[x := N] \equiv M \equiv y$ donc le résultat qui est $\Gamma \vdash y : \tau$ est clair.

• Si $M \equiv x$ alors $\tau = \sigma$ et $M[x := N] \equiv N$ et le résultat est l'une des hypothèses.

$\boxed{\text{Si } \Gamma, x : \sigma \vdash M : \tau \text{ et } \Gamma \vdash N : \sigma \text{ alors } \Gamma \vdash M[x := N] : \tau}$

Démonstration : Par induction sur M.

- Si $M \equiv P\ Q$ alors par (App) pour un certain τ', on a
  - $\Gamma, x : \sigma \vdash P : \tau' \to \tau$ duquel on tire par induction $\Gamma \vdash P[x := N] : \tau' \to \tau$
  - et $\Gamma, x : \sigma \vdash Q : \tau'$ duquel on tire par induction $\Gamma \vdash Q[x := N] : \tau'$.

Donc par (App) $\Gamma \vdash (P\ Q)[x := N] : \tau$ puisque $(P\ Q)[x := N] \equiv P[x := N]\ Q[x := N]$.

- Le cas $M \equiv \lambda x.P$ est similaire.

Réduction du sujet

Lemme Réduction du sujet

La β-réduction préserve le type.

Si $\Gamma \vdash M : \sigma$ et si $M \xrightarrow[\beta]{} N$ alors $\Gamma \vdash N : \sigma$.

$\boxed{\text{Si } \Gamma \vdash M : \sigma \text{ et si } M \xrightarrow[\beta]{} N \text{ alors } \Gamma \vdash N : \sigma.}$

Démonstration : Par induction sur la définition de $M \xrightarrow[\beta]{} N$.

- $M \equiv (\lambda x.M_1)M_2$
  $\Gamma \vdash (\lambda x.M_1)M_2 : \sigma$ vient de $\Gamma, x : \tau \vdash M_1 : \sigma$ et de $\Gamma \vdash M_2 : \tau$.
  D'autre part $N \equiv M_1[x := M_2]$
  or d'après le lemme $\Gamma \vdash M_1[x := M_2] : \sigma$.

$\boxed{\text{Si } \Gamma \vdash M : \sigma \text{ et si } M \xrightarrow[\beta]{} N \text{ alors } \Gamma \vdash N : \sigma.}$

Démonstration : Par induction sur la définition de $M \xrightarrow[\beta]{} N$.

- $M \equiv (\lambda x.M_1)M_2$
  $\Gamma \vdash (\lambda x.M_1)M_2 : \sigma$ vient de $\Gamma, x : \tau \vdash M_1 : \sigma$ et de $\Gamma \vdash M_2 : \tau$.
  D'autre part $N \equiv M_1[x := M_2]$
  or d'après le lemme $\Gamma \vdash M_1[x := M_2] : \sigma$.
- $M \equiv M_1 M_2$ avec $N \equiv N_1 M_2$ et $M_1 \xrightarrow[\beta]{} N_1$.
  $\Gamma \vdash M : \sigma$ vient de $\Gamma \vdash M_1 : \tau \to \sigma$ et $\Gamma \vdash M_2 : \tau$,
  par induction $\Gamma \vdash N_1 : \tau \to \sigma$ et donc $\Gamma \vdash N_1 M_2 : \sigma$.

$\boxed{\text{Si } \Gamma \vdash M : \sigma \text{ et si } M \xrightarrow[\beta]{} N \text{ alors } \Gamma \vdash N : \sigma.}$

Démonstration : Par induction sur la définition de $M \xrightarrow[\beta]{} N$.

- $M \equiv (\lambda x.M_1)M_2$
  $\Gamma \vdash (\lambda x.M_1)M_2 : \sigma$ vient de $\Gamma, x : \tau \vdash M_1 : \sigma$ et de $\Gamma \vdash M_2 : \tau$.
  D'autre part $N \equiv M_1[x := M_2]$
  or d'après le lemme $\Gamma \vdash M_1[x := M_2] : \sigma$.
- $M \equiv M_1 M_2$ avec $N \equiv N_1 M_2$ et $M_1 \xrightarrow[\beta]{} N_1$.
  $\Gamma \vdash M : \sigma$ vient de $\Gamma \vdash M_1 : \tau \rightarrow \sigma$ et $\Gamma \vdash M_2 : \tau$,
  par induction $\Gamma \vdash N_1 : \tau \rightarrow \sigma$ et donc $\Gamma \vdash N_1 M_2 : \sigma$.
- Les autres cas sont similaires.

Commentaires

- Si un terme est d'un certain type, il n'en sortira pas par β-réduction.
- Si un terme a une forme normale et s'il a un type, alors sa forme normale a ce type.

Les types à la Church

Les types à la Curry

La correspondance de Curry-Howard

Normalisation forte

Une autre démonstration de la normalisation forte

Les autres connecteurs

La mathématique
c'est l'art de donner le même nom à des choses différentes.

Henri Poincaré

La mathématique
c'est l'art de donner le même nom à des choses différentes.

Henri Poincaré

La logique
c'est l'art de donner des noms différents à la même chose.

Haskell Curry et William Alvin Howard

La correspondance de Curry-Howard

$$
\begin{array}{ll}
(\mathrm{Var}) & \Gamma, x : \sigma \vdash x : \sigma \\[1em]
(\mathrm{Abs}) & \dfrac{\Gamma, x : \sigma \vdash M : \tau}{\Gamma \vdash \lambda x.M : \sigma \to \tau} \\[1em]
(\mathrm{App}) & \dfrac{\Gamma \vdash M : \sigma \to \tau \qquad \Gamma \vdash N : \sigma}{\Gamma \vdash MN : \tau}
\end{array}
\qquad\Bigg|\qquad
\begin{array}{ll}
 & \Gamma, p \vdash p \\[1em]
\Rightarrow I & \dfrac{\Gamma, p \vdash q}{\Gamma \vdash p \Rightarrow q} \\[1em]
\Rightarrow E & \dfrac{\Gamma \vdash p \Rightarrow q \qquad \Gamma \vdash p}{\Gamma \vdash q}
\end{array}
$$

La correspondance de Curry-Howard

Dans $\Gamma \vdash M : \sigma$,

- M : est une annotation qui est le «terme de preuve»,
- σ peut-être vu comme un type ou comme une proposition.

La preuve de B

$$
\dfrac{\Rightarrow E \quad \dfrac{(p \Rightarrow q),\ (r \Rightarrow p), r \vdash p \Rightarrow q \qquad \mathcal{D}}{(p \Rightarrow q),\ (r \Rightarrow p), r \vdash q}}{\dfrac{(p \Rightarrow q),\ (r \Rightarrow p) \vdash r \Rightarrow q}{\dfrac{(p \Rightarrow q) \vdash (r \Rightarrow p) \Rightarrow r \Rightarrow q}{\vdash (p \Rightarrow q) \Rightarrow (r \Rightarrow p) \Rightarrow r \Rightarrow q}}}
$$

où $\mathcal{D}$ est

$$
\Rightarrow E \quad \dfrac{(p \Rightarrow q),\ (r \Rightarrow p), r \vdash r \Rightarrow p \qquad (p \Rightarrow q),\ (r \Rightarrow p), r \vdash r}{(p \Rightarrow q),\ (r \Rightarrow p), r \vdash p}
$$

La preuve de B annotée

$$\dfrac{\Rightarrow E \quad \dfrac{x : (p \Rightarrow q),\ y : (r \Rightarrow p), z : r \vdash x : p \Rightarrow q \qquad \mathcal{D}}{x : (p \Rightarrow q),\ y : (r \Rightarrow p), z : r \vdash x\ (y\ z) : q}}{\dfrac{x : (p \Rightarrow q),\ y : (r \Rightarrow p) \vdash \lambda z.x\ (y\ z) : r \Rightarrow q}{\dfrac{x : (p \Rightarrow q) \vdash \lambda yz.x\ (y\ z) : (r \Rightarrow p) \Rightarrow r \Rightarrow q}{\vdash \lambda xyz.x\ (y\ z) : (p \Rightarrow q) \Rightarrow (r \Rightarrow p) \Rightarrow r \Rightarrow q}}}$$

où $\mathcal{D}$ est

$$\Rightarrow E \quad \dfrac{x : (p \Rightarrow q),\ y : (r \Rightarrow p), z : r \vdash y : r \Rightarrow p \qquad x : (p \Rightarrow q),\ y : (r \Rightarrow p), z : r \vdash z : r}{x : (p \Rightarrow q),\ y : (r \Rightarrow p), z : r \vdash y\ z : p}$$

La preuve de B annotée

$$
\dfrac{\dfrac{x : (p \Rightarrow q),\ y : (r \Rightarrow p), z : r \vdash x : p \Rightarrow q \qquad \mathcal{D}}{x : (p \Rightarrow q),\ y : (r \Rightarrow p), z : r \vdash x\ (y\ z) : q}\ \Rightarrow E}{\dfrac{x : (p \Rightarrow q),\ y : (r \Rightarrow p) \vdash \lambda z.x\ (y\ z) : r \Rightarrow q}{\dfrac{x : (p \Rightarrow q) \vdash \lambda yz.x\ (y\ z) : (r \Rightarrow p) \Rightarrow r \Rightarrow q}{\vdash \lambda xyz.x\ (y\ z) : (p \Rightarrow q) \Rightarrow (r \Rightarrow p) \Rightarrow r \Rightarrow q}}}
$$

où $\mathcal{D}$ est

$$
\Rightarrow E\ \dfrac{x : (p \Rightarrow q),\ y : (r \Rightarrow p), z : r \vdash y : r \Rightarrow p \qquad x : (p \Rightarrow q),\ y : (r \Rightarrow p), z : r \vdash z : r}{x : (p \Rightarrow q),\ y : (r \Rightarrow p), z : r \vdash y\ z : p}
$$

La preuve du lemme B est le terme B !

Simplification de preuves

La preuve

$$\dfrac{\dfrac{\dfrac{(p \Rightarrow p), q \vdash p \Rightarrow p}{(p \Rightarrow p) \vdash q \Rightarrow p \Rightarrow p}\,(\Rightarrow I)}{\vdash (p \Rightarrow p) \Rightarrow q \Rightarrow p \Rightarrow p}\,(\Rightarrow I) \qquad \dfrac{p \vdash p}{\vdash p \Rightarrow p}\,(\Rightarrow I)}{\vdash q \Rightarrow p \Rightarrow p}\,(\Rightarrow E)$$

peut être réduite

Simplification de preuves

En effet, on fait une introduction immédiatement suivie d'une élimination :

$$\dfrac{\dfrac{\dfrac{(p \Rightarrow p), q \vdash p \Rightarrow p}{(p \Rightarrow p) \vdash q \Rightarrow p \Rightarrow p}\,(\Rightarrow I)}{\vdash (p \Rightarrow p) \Rightarrow q \Rightarrow p \Rightarrow p}\,(\Rightarrow I) \qquad \dfrac{p \vdash p}{\vdash p \Rightarrow p}\,(\Rightarrow I)}{\vdash q \Rightarrow p \Rightarrow p}\,(\Rightarrow E)$$

On peut obtenir une preuve de $\vdash q \Rightarrow p \Rightarrow p$ à partir de la preuve de $(p \Rightarrow p) \vdash q \Rightarrow p \Rightarrow p$.
Pour cela, il suffit de remplacer chaque occurrence de l'utilisation de l'hypothèse $(p \Rightarrow p)$ par sa preuve.

Simplification de preuves

En utilisant cette remarque

$$\dfrac{\dfrac{\dfrac{(p \Rightarrow p), q \vdash p \Rightarrow p}{(p \Rightarrow p) \vdash q \Rightarrow p \Rightarrow p}\,(\Rightarrow I)}{\vdash (p \Rightarrow p) \Rightarrow q \Rightarrow p \Rightarrow p}\,(\Rightarrow I) \qquad \dfrac{p \vdash p}{\vdash p \Rightarrow p}\,(\Rightarrow I)}{\vdash q \Rightarrow p \Rightarrow p}\,(\Rightarrow E)$$

donne

$$\dfrac{\dfrac{q, p \vdash p}{q \vdash p \Rightarrow p}\,(\Rightarrow I)}{\vdash q \Rightarrow p \Rightarrow p}\,(\Rightarrow I)$$

Simplification de preuves

Donc avec les annotations par les termes de preuve

$$\dfrac{\dfrac{\dfrac{x : (p \Rightarrow p), y : q \vdash x : p \Rightarrow p}{x : (p \Rightarrow p) \vdash \lambda y.x : q \Rightarrow p \Rightarrow p}\,(Abs)}{\vdash \lambda xy.x : (p \Rightarrow p) \Rightarrow q \Rightarrow p \Rightarrow p}\,(Abs) \qquad \dfrac{z : p \vdash z : p}{\vdash \lambda z.z : p \Rightarrow p}\,(Abs)}{\vdash (\lambda xy.x)\ (\lambda z.z) : q \Rightarrow p \Rightarrow p}\,(App)$$

donne

$$\dfrac{\dfrac{y : q, z : p \vdash z : p}{y : q \vdash \lambda z.z : p \Rightarrow p}\,(Abs)}{\vdash \lambda yz.z : q \Rightarrow p \Rightarrow p}\,(Abs)$$

Simplification de preuves

Donc avec les annotations par les termes de preuve

$$\dfrac{\dfrac{\dfrac{x : (p \Rightarrow p), y : q \vdash x : p \Rightarrow p}{x : (p \Rightarrow p) \vdash \lambda y.x : q \Rightarrow p \Rightarrow p}\,(Abs)}{\vdash \lambda xy.x : (p \Rightarrow p) \Rightarrow q \Rightarrow p \Rightarrow p}\,(Abs) \qquad \dfrac{z : p \vdash z : p}{\vdash \lambda z.z : p \Rightarrow p}\,(Abs)}{\vdash (\lambda xy.x)\ (\lambda z.z) : q \Rightarrow p \Rightarrow p}\,(App)$$

donne

$$\dfrac{\dfrac{y : q, z : p \vdash z : p}{y : q \vdash \lambda z.z : p \Rightarrow p}\,(Abs)}{\vdash (\lambda y.x)[x := \lambda z.z] \equiv \lambda yz.z : q \Rightarrow p \Rightarrow p}\,(Abs)$$

La β-réduction correspond à la simplification des preuves.

Simplification de preuves

$$
\dfrac{\dfrac{\dfrac{\mathcal{D}}{\phi, \Gamma \vdash \psi}}{\Gamma \vdash \phi \Rightarrow \psi}\,(\Rightarrow I) \qquad \dfrac{\mathcal{D}'}{\Gamma \vdash \phi}}{\Gamma \vdash \psi}\,(\Rightarrow E)
$$

se transforme en

$$
\dfrac{\mathcal{D}''}{\Gamma \vdash \psi}
$$

$\mathcal{D}''$ est la preuve $\mathcal{D}$ dans laquelle toutes les utilisations de l'hypothèse ϕ sont remplacées par la preuve $\mathcal{D}'$ de ϕ.

Simplification de preuves

$$\dfrac{\dfrac{\dfrac{\mathcal{D}}{x : \phi, \Gamma \vdash M : \psi}}{\Gamma \vdash (\lambda x.M) : \phi \Rightarrow \psi}\,(Abs) \qquad \dfrac{\mathcal{D}'}{\Gamma \vdash N : \phi}}{\Gamma \vdash (\lambda x.M)\ N : \psi}\,(App)$$

se transforme en

$$\dfrac{\mathcal{D}''}{\Gamma \vdash M[x := N] \in \psi}$$

$\mathcal{D}''$ (qui se note $M[x := N]$)
est la preuve $\mathcal{D}$ (qui se note M)
dans laquelle toutes les utilisations de l'hypothèse $x : \phi$ sont remplacées par la preuve $\mathcal{D}'$ de ϕ (qui se note N).

La correspondance complète de Curry-Howard

On obtient le tableau de correspondance :

types	propositions
termes	preuves
réduction	simplification des preuves

Terminaison de la simplification ?

Est-ce que le processus de simplification de preuves se termine ? Autrement dit est-ce que ce processus est vraiment un processus de simplification.

Normalisation forte

Définition : Un terme M est fortement normalisable si toute suite de réductions $M \xrightarrow[\beta]{} M_1 \xrightarrow[\beta]{} \ldots \xrightarrow[\beta]{} M_n \ldots$ est finie.

Théorème : Si $\Gamma \vdash M : \tau$ alors M est fortement normalisable.

Normalisation forte (démonstration), 1$^{\text{er}}$ lemme

Lemme : Si A, B et $\vec{C}$ sont fortement normalisables
et $AB\vec{C}$ n'est pas fortement normalisable
alors il y a un D tel que

1. $A \xrightarrow[\beta]{} \!\!\!\!\rightarrow \lambda x.D$
2. $D[x := B]\vec{C}$ n'est pas fortement normalisable.

Normalisation forte (démonstration), 1$^{\text{er}}$ lemme

Démonstration :

Puisque A, B et $\vec{C}$ sont fortement normalisables, la réduction infinie qui part de $AB\vec{C}$ est de la forme

$$AB\vec{C} \underset{\beta}{\twoheadrightarrow} (\lambda x.D)B_1\vec{C_1} \underset{\beta}{\longrightarrow} D[x := B_1]\vec{C_1} \underset{\beta}{\twoheadrightarrow} \dots$$

Mézalors le résultat vient immédiatement du fait que

$$D[x := B]\vec{C} \underset{\beta}{\twoheadrightarrow} D[x := B_1]\vec{C_1}.$$

Normalisation forte (démonstration), 1^{er} lemme

Remettons le lemme à l'endroit.

Lemme : Si A, B et $\vec{C}$ sont fortement normalisables
et si D est le premier tel que

1. $A \xrightarrow[\beta]{} \!\!\!\!\!\rightarrow \lambda x.D$
2. $D[x := B]\vec{C}$ est fortement normalisable.

alors $AB\vec{C}$ est fortement normalisable

Normalisation forte (démonstration), 2ème lemme

Lemme : Si Met P sont typés
et si M et P sont fortement normalisables,
alors $M[x := P]$ est fortement normalisable.

Normalisation forte (démonstration), $2^{\text{ème}}$ lemme

Démonstration :

Par induction sur le triplet $(\mathit{type}(P), h(M), \mathit{taille}(M))$, où

- où $\mathit{type}(P)$ est la complexité du type de P,
- $h(M)$ (ou hauteur) est la longueur de la plus longue réduction qui commence en M,
- $\mathit{taille}(M)$ est la taille de M (le nombre de symboles).

Normalisation forte (démonstration), $2^{\text{ème}}$ lemme

- $M = \lambda y.N$, clairement $h(M) = h(N)$, mais $taille(M) > taille(N)$. On applique l'induction.
  Donc $N[x := P]$ est fortement normalisable, donc $M[x := P] \equiv \lambda y N[x := P]$ est fortement normalisable.

Normalisation forte (démonstration), 2ème lemme

- $y\vec{R}$, les réductions ont lieu dans les R_i, donc clairement $h(M) \geq h(R_i)$, et $taille(M) > taille(R_i)$.
  On applique l'induction à chacun des R_i, on conclut que chaque $R_i[x := P]$ est fortement normalisable, donc $y...R_i[x := P]...$ est fortement normalisable, mais comme

  $$y...R_i[x := P]... \equiv (y\vec{R})[x := P].$$

  Par conséquent, $(y\vec{R})[x := P]$ est aussi fortement normalisable

Normalisation forte (démonstration), $2^{\text{ème}}$ lemme

- $M = (\lambda y.L)Q\vec{R}$
  Par induction, $L[x := P]$, $Q[x := P]$ et $\vec{R}[x := P]$ sont fortement normalisables.
  Posons $D \equiv L[y := Q]\ \vec{R}$.
  Si on veut utiliser le lemme précédent, il suffit de montrer que
  $L[x := P][y := Q[x := P]]\ \vec{R}[x := P] \equiv D[x := P]$
  est fortement normalisable .
  Or $h(M) > h(D)$ donc par induction $D[x := P]$ est fortement normalisable. On peut donc appliquer le 1^{er} lemme.
  Et donc par le lemme, $M[x := P]$ est fortement normalisable.

Normalisation forte (démonstration), 2$^{\text{ème}}$ lemme

- $M = xL\vec{Q}$
  On veut démontrer que $P\ L[x := P]\ \vec{Q}[x := P]$ est fortement normalisable.
  Notons que
  - $h(M) \geq h(L)$ et $taille(M) > taille(L)$
  - et $h(M) \geq h(Q_i)$ et $taille(M) > taille(Q_i)$.

  Donc, par induction, $L' \equiv L[x := P]$ et $\vec{Q}' \equiv \vec{Q}[x := P]$ sont fortement normalisables.

Normalisation forte (démonstration), $2^{\text{ème}}$ lemme

- $M = xL\vec{Q}$ (suite)

  Si on veut utiliser le lemme précédent, il suffit de montrer que si P_1 est le premier tel que $P \xrightarrow[\beta]{} \!\!\!\!\rightarrow \lambda y.P_1$

  alors $M' \equiv P_1[y := L']\vec{Q}'$ est fortement normalisable.

  Remarquons

  - que $type(x) = type(P) = type(\lambda y.P_1) = \sigma \to \tau$,
  - que $type(L) = type(L') = \sigma < type(P)$.
  - et que $type(P_1[y := L']) = \tau < type(P)$

Normalisation forte (démonstration), 2$^{\text{ème}}$ lemme

- $M = xL\vec{Q}$ (suite)
  On peut appliquer l'hypothèse d'induction et donc $P_1[y := L']$ est fortement normalisable.
  Parce que $type(P_1[y := L']) < type(P)$, on conclut par induction, que $M' \equiv (z\vec{Q}')[z := P_1[y := L']]$ est fortement normalisable
  Ainsi on peut appliquer le deuxième lemme et conclure que $M[x := P]$ est fortement normalisable.

Normalisation forte (démonstration)

Théorème : Si $\Gamma \vdash M : \tau$ alors M est fortement normalisable.

Le théorème est prouvé par induction structurelle sur le terme dont on cherche à prouver la normalisation forte.

- Le terme est une variable ou une abstraction c'est clair.
- Le terme est une application $P\ Q$
  alors on utilise la même astuce que dans le lemme en disant que
  $$P\ Q \quad \equiv \quad (zQ)[z := P]$$
  ça devient évident par le $2^{\text{ème}}$ lemme dont les hypothèses proviennent de l'induction structurelle.
  En effet si Q est fortement normalisable alors $z\ Q$ est fortement normalisable.

Réductibles

Soit $\mathcal{R}^\sigma$ des ensembles de termes

$$\begin{aligned}
\mathcal{R}^o &= \mathcal{SN} = \{\text{l'ensemble de fortement normalisables}\} \\
\mathcal{R}^{\sigma\to\tau} &= \{M \mid (\forall N \in \mathcal{R}^\sigma)\ M\ N \in \mathcal{R}^\tau\}
\end{aligned}$$

Les réductibles sont fortement normalisables

Lemme : Pour tout σ

- pour tous $M_1 \in \mathcal{SN}$, ..., $M_p \in \mathcal{SN}$ on a $x\ M_1 \ldots M_p \in \mathcal{R}^\sigma$,
- $\mathcal{R}^\sigma \subseteq \mathcal{SN}$.

Démonstration

Par induction sur les types.

- Si le type est o, alors clairement $x\ M_1 \ldots M_p \in \mathcal{R}^o = \mathcal{SN}$ et $\mathcal{R}^o \subseteq \mathcal{SN}$.

- Si le type est $\sigma \to \tau$, alors
  - Pour montrer que $x\ M_1 ... M_p \in \mathcal{R}^{\sigma\to\tau}$, il faut montrer que pour tout $N \in \mathcal{R}^{\sigma}$, on a $x\ M_1 ... M_p\ N \in \mathcal{R}^{\tau}$;

    or par induction $\mathcal{R}^{\sigma} \subseteq \mathcal{SN}$

    et donc aussi par induction $x\ M_1 ... M_p\ N \in \mathcal{R}^{\tau}$;
  - Soit $M \in \mathcal{R}^{\sigma\to\tau}$, on sait que pour tout $N \in \mathcal{R}^{\sigma}$

    on a $M\ N \in \mathcal{R}^{\tau} \subseteq \mathcal{SN}$

    on sait que $x \in \mathcal{R}^{\sigma}$,

    donc en particulier $M\ x \in \mathcal{R}^{\tau} \subseteq \mathcal{SN}$,

    et donc $M \in \mathcal{SN}$.

Lemme de saturation

$\boxed{\text{Si } M[x := P] \in \mathcal{R}^\sigma \text{ et si } P \in \mathcal{R}^\tau \text{ alors } (\lambda x.M)\ P \in \mathcal{R}^\sigma.}$

Démonstration

On voit que si $\sigma \equiv \sigma_n \to \ldots \to o$, alors

$$M \in \mathcal{R}^\sigma \quad \Leftrightarrow \quad \forall N_n \in \mathcal{R}^{\sigma_n} \ldots \forall N_1 \in \mathcal{R}^{\sigma_1}\ M\ N_n \ldots N_1 \in \mathcal{SN}.$$

Il faut donc montrer que $(\lambda x.M)\ P\ N_n \ldots N_1 \in \mathcal{SN}$ sachant que $M[x := P]\ N_n \ldots N_1 \in \mathcal{SN}$ et que $N_n \in \mathcal{R}^{\sigma_n} \ldots N_1 \in \mathcal{R}^{\sigma_1}, P : \mathcal{R}^\tau$.

On raisonne par induction sur $red(P) + red(M[x := P]\ N_n ... N_1)$.

où $red(Q)$ est la somme des longueurs de toutes les réductions qui commencent en Q.

Il faut montrer que toutes les réductions de $(\lambda x.M)\ P\ N_n ... N_1$ sont fortement normalisables.

- Si on contracte le redex $(\lambda x.M)\ P$ alors $M[x := P]\ N_n ... N_1 \in \mathcal{SN}$ par hypothèse.
- Si on contracte en M' un redex dans M, on sait que $M[x := P]\ N_n ... N_1 \xrightarrow[\beta]{} M'[x := P]\ N_n ... N_1$ donc $M'[x := P]\ N_n ... N_1 \in \mathcal{SN}$ et de plus $red(M'[x := P]\ N_n ... N_1) < red(M[x := P]\ N_n ... N_1)$ donc par induction $(\lambda x.M')\ P\ N_n ... N_1 \in \mathcal{SN}$
- Même démonstration si on contracte un redex dans N_i.

• Si on contracte le redex P en P', alors $red(P') < red(P)$.

$M[x := P]\ N_n ... N_1 \xrightarrow[\beta]{} \!\!\!\!\!\!\rightarrow M[x := P']\ N_n ... N_1$

donc $M[x := P']\ N_n ... N_1 \in \mathcal{SN}$,

et $red(M[x := P']\ N_n ... N_1) \leq red(M[x := P]\ N_n ... N_1)$

par induction $(\lambda x.M)\ P'\ N_n ... N_1 \in \mathcal{SN}$.

Lemme d'adéquation

Si $x_1 : \sigma_1, \ldots, x_n : \sigma_n \vdash M : \sigma$
et si $N_i \in \mathcal{R}^{\sigma_i}$
alors $M[x_1 := N_1, \ldots, x_n := N_n] \in \mathcal{R}^{\sigma}$.

Démonstration

Par induction sur M.

• Si $M = y$ avec $y \neq x_i$, alors $M[x_1 := N_1, \ldots, x_n := N_n] = y \in \mathcal{R}^{\sigma}$ par le lemme.

• Si $M = x_i$ alors $M[x_1 := N_1, \ldots, x_n := N_n] = N_i \in \mathcal{R}^{\sigma}$ par hypothèse.

- Si $M = \lambda x.M'$ et $x_1 : \sigma_1, ..., x_n : \sigma_n \vdash \lambda x.M' : \sigma \rightarrow \tau$
  alors $x_1 : \sigma_1, ..., x_n : \sigma_n, x : \sigma \vdash M' : \tau$.

Il faut montrer que pour tout $N \in \mathcal{R}^\sigma$, on a
$(\lambda x.M')[x_1 := N_1, ..., x_n := N_n]\ N \in \mathcal{R}^\tau$.

D'après le lemme de saturation il faut montrer que
$M'[x_1 := N_1, ..., x_n := N_n, x := N] \in \mathcal{R}^\tau$.
Ce qui vient par hypothèse d'induction.

- Si $M = P\ Q$ alors
  - $x_1 : \sigma_1, ..., x_n : \sigma_n \vdash P : \tau \to \sigma$
  - et $x_1 : \sigma_1, ..., x_n : \sigma_n \vdash Q : \tau$,

par induction

- $P[x_1 := N_1, ..., x_n := N_n] \in \mathcal{R}^{\tau \to \sigma}$
- et $Q[x_1 := N_1, ..., x_n := N_n] \in \mathcal{R}^{\tau}$.

- Si $M = P\ Q$ alors
  - $x_1 : \sigma_1, ..., x_n : \sigma_n \vdash P : \tau \to \sigma$
  - et $x_1 : \sigma_1, ..., x_n : \sigma_n \vdash Q : \tau$,

par induction
  - $P[x_1 := N_1, ..., x_n := N_n] \in \mathcal{R}^{\tau \to \sigma}$
  - et $Q[x_1 := N_1, ..., x_n := N_n] \in \mathcal{R}^{\tau}$.

Donc

$$\begin{aligned} &M\ N[x_1 := N_1, ..., x_n := N_n] \\ &\quad = M[x_1 := N_1, ..., x_n := N_n]\ N[x_1 := N_1, ..., x_n := N_n] \\ &\in \mathcal{R}^{\sigma}. \end{aligned}$$

Théorème de normalisation forte

Les typables sont réductibles.

$$\boxed{\text{Si } x_1 : \sigma_1, \ldots, x_n : \sigma_n \vdash M : \sigma \text{ alors } M \in \mathcal{R}^\sigma.}$$

On applique le lemme d'adéquation avec $N_i = x_i$ et on obtient $M \in \mathcal{R}^\sigma$.

Théorème de normalisation forte

Les typables sont réductibles.

$\boxed{\text{Si } x_1 : \sigma_1, \ldots, x_n : \sigma_n \vdash M : \sigma \text{ alors } M \in \mathcal{R}^\sigma.}$

On applique le lemme d'adéquation avec $N_i = x_i$ et on obtient $M \in \mathcal{R}^\sigma$.

Les typables sont fortement noramlisables.

$\boxed{\text{Si } x_1 : \sigma_1, \ldots, x_n : \sigma_n \vdash M : \sigma \text{ alors } M \in \mathcal{SN}.}$

On a $M \in \mathcal{SN}$ puisque $\mathcal{R}^\sigma \subseteq \mathcal{SN}$.

Le $\wedge$ et le produit cartésien

Le $\wedge$ s'interprète bien calculatoirement en ajoutant des opérateurs de produit cartésien.

Le constructeur de paires : $\langle \_, \_ \rangle$,

Les destructeurs ou projections : π^1 et π^2.

Le $\wedge$ et le produit cartésien

Les règles de typage

$$\frac{\Gamma \vdash M : \sigma \times \tau}{\Gamma \vdash \pi^1\ M : \sigma} \qquad \frac{\Gamma \vdash M : \sigma \times \tau}{\Gamma \vdash \pi^2\ M : \tau}$$

$$\frac{\Gamma \vdash M : \sigma \qquad \Gamma \vdash N : \tau}{\Gamma \vdash \langle M, N \rangle : \sigma \times \tau}$$

correspondent clairement

- aux deux règles d'éliminations du $\wedge$
- et à la règle d'introduction du $\wedge$.

Le $\wedge$ et le produit cartésien

On a deux règles de réduction

$$\begin{array}{rcl} \pi^1 \ \langle M, N\rangle & \rightarrow & M \\ \pi^2 \ \langle M, N\rangle & \rightarrow & N \end{array}$$

qui correspondent à la simplification de preuve :

$$\dfrac{\dfrac{\dfrac{\mathcal{D}}{\Gamma \vdash M : \sigma} \qquad \Gamma \vdash N : \tau}{\Gamma \vdash \langle M, N\rangle : \sigma \times \tau}}{\Gamma \vdash \pi^1\langle M, N\rangle : \sigma} \qquad \rightarrow \qquad \dfrac{\mathcal{D}}{\Gamma \vdash M : \sigma}$$

et la simplification symétrique.

Le $\vee$ et la somme

Le $\vee$ s'interprète comme la structure de données somme.

Les deux constructeurs : in_1 et in_2,

Le destructeur ou discriminateur : *case* *of* *in* | .

un peu le **match** de CAML.

Le $\vee$ et la somme

Les règles de typage sont

$$\frac{\Gamma \vdash M : \sigma}{\Gamma \vdash \mathrm{in}_1\ M : \sigma \vee \tau} \qquad \frac{\Gamma \vdash M : \tau}{\Gamma \vdash \mathrm{in}_2\ M : \sigma \vee \tau}$$

$$\frac{\Gamma \vdash M : \sigma \vee \tau \qquad \Gamma, x : \sigma \vdash P : \rho \qquad \Gamma, x : \tau \vdash Q : \rho}{\Gamma \vdash \mathit{case}\ M\ \mathit{of}\ x\ \mathit{in}\ P \mid Q : \rho}$$

Elles correspondent

- aux règles d'introduction du $\vee$
- et à la règle d'élimination du $\vee$.

Le $\vee$ et la somme

On a les règles de réduction

$$\begin{array}{lcl} \textit{case}\ (\mathrm{in}_1\ M)\ \textit{of}\ x\ \textit{in}\ P \mid Q & \rightarrow & P[x := M] \\ \textit{case}\ (\mathrm{in}_2\ M)\ \textit{of}\ x\ \textit{in}\ P \mid Q & \rightarrow & Q[x := M] \end{array}$$

qui correspond à la simplification de preuve.

$$\dfrac{\dfrac{\dfrac{\mathcal{D}'}{\Gamma \vdash M : \sigma}}{\Gamma \vdash \mathrm{in}_1\ M : \sigma \vee \tau} \qquad \dfrac{\mathcal{D}}{\Gamma, x : \sigma \vdash P : \rho} \qquad \dfrac{\mathcal{D}''}{\Gamma, x : \tau \vdash Q : \rho}}{\textit{case}\ M\ \textit{of}\ x\ \textit{in}\ P \mid Q : \rho} \quad \rightarrow \quad \dfrac{\mathcal{D}\{\mathcal{D}'/\Gamma, x : \sigma \vdash x : \sigma\}}{\Gamma \vdash P[x := M] : \rho}$$