Le $\lambda\upsilon$-calcul (suite du sujet de 2001).

Documents autorisés (en particulier le poly).

Les questions sont annotées par un niveau de difficulté, variant de (0) (facile) à (3) (difficile). On notera bien que les questions les plus difficiles sont aussi celles qui rapportent le plus de points. Un candidat qui ne répondrait qu'au questions évaluées (0) ou (1), même à la perfection, n'aura ainsi (probablement) aucune chance d'atteindre la moyenne.

Le but de ce problème est d'étudier le $\lambda\upsilon$-calcul, un λ-calcul à substitutions explicites proposé par J. Rouyer-Degli et P. Lescanne en 1994. Note : la version exposée ici a une infinité de constantes $\underline{1}$, $\underline{2}$, $\underline{3}$, etc., pour dénoter les variables liées, alors que l'examen de 2001 utilisait une seule constante 1, et codait 2 comme $1[\uparrow]$, 3 comme $1[\uparrow][\uparrow]$, etc.

Comme dans le $\lambda\sigma$-calcul, les *expressions* du $\lambda\upsilon$-calcul sont ceux définis par la grammaire :

$$
\begin{array}{rcll}
 & & & \text{Termes} \\
M, N, P, \ldots & ::= & x & \text{Variables} \\
 & | & \underline{n} & (n \in \mathbb{N}, n \geq 1) \text{ Indices de de Bruijn 1, 2, 3, ...} \\
 & | & MM & \text{Applications} \\
 & | & \lambda M & \text{Abstractions} \\
 & | & M[S] & \text{Application de substitution} \\
 & & & \text{Piles (substitutions explicites)} \\
S & ::= & \uparrow & \text{Shift} \\
 & | & \Uparrow S & \text{Lift} \\
 & | & M/ & \text{Slash}
\end{array}
$$

Les changements par rapport au $\lambda\sigma$-calcul sont : 1. on a un opérateur explicite $\Uparrow$, et $\Uparrow S$ correspond à la pile $1 \cdot (S\circ \uparrow)$ du $\lambda\sigma$-calcul ; 2. on a un nouvel opérateur de pile $/$, et $M/$ correspond à la pile $M \cdot id$ du $\lambda\sigma$-calcul ; 3. on n'a pas les constructions $M \cdot S$ et $S \circ S'$ du $\lambda\sigma$-calcul ; 4. on a une infinité d'indices de de Bruijn $\underline{1}$, $\underline{2}$, $\underline{3}$, etc. De plus, on n'a pas de variable de pile.

La traduction d'un λ-terme u en un $\lambda\upsilon$-terme $u^\bullet(\ell)$ (dans le contexte de la liste de variables ℓ, comme pour le $\lambda\sigma$-calcul étudié dans le cours) est définie comme suit. (Contrairement au $\lambda\sigma$-calcul, le terme $\underline{n}$ n'est *pas* une abréviation.)

$$
\begin{array}{rcll}
x^\bullet(\ell) & \hat{=} & \underline{n} & \text{si } n \text{ est le premier entier tel que } x = \ell(n) \\
 & & x\underbrace{[\uparrow][\uparrow]\ldots[\uparrow]}_{k \text{ fois}} & \text{si } x \text{ n'est pas dans } \ell, k \text{ est la longueur de } \ell \\
(uv)^\bullet(\ell) & \hat{=} & u^\bullet(\ell)v^\bullet(\ell) & \\
(\lambda x \cdot u)^\bullet(\ell) & \hat{=} & \lambda(u^\bullet(x :: \ell)) &
\end{array}
$$

(On rappelle que la liste ℓ de longueur k est par convention la suite $\ell(1), \ell(2), \ldots, \ell(k)$, où $\ell(i)$ dénote donc le $i$ième élément de la liste.)

Les règles de réduction du $\lambda\upsilon$-calcul sont données en figure 1. Le système formé de toutes les règles sauf (β) s'appelle υ.

(β)	$(\lambda M)N \to M[N/]$		
$(Var/)$	$\underline{1}[M/] \to M$	$(Var \Uparrow)$	$\underline{1}[\Uparrow S] \to \underline{1}$
$(\uparrow /)$	$\underline{n+1}[N/] \to \underline{n}$	$(\uparrow\Uparrow)$	$\underline{n+1}[\Uparrow S] \to \underline{n}[S][\uparrow]$
(App)	$(MN)[S] \to M[S](N[S])$	(Lam)	$(\lambda M)[S] \to \lambda(M[\Uparrow S])$
$(Var \uparrow)$	$\underline{n}[\uparrow] \to \underline{n+1}$		

FIG. 1 – Les règles de réduction de $\lambda\upsilon$ ($n \geq 1$)

I. Réduction.

1. (2) Montrer que, pour tout λ-terme v, pour toute liste ℓ de variables, pour tout entier k, pour toutes variables $y_1, \ldots, y_p$, et pour toute variable z non libre dans v, si toutes les variables libres de v sont dans ℓ ou parmi $y_1, \ldots, y_p$, alors

$$v^\bullet(y_1 :: \ldots :: y_p :: \ell)[\Uparrow^p \uparrow] \to_\upsilon^* v^\bullet(y_1 :: \ldots :: y_p :: z :: \ell)$$

où $\Uparrow^p S$ dénote $\underbrace{\Uparrow \ldots \Uparrow}_{p \text{ fois}} S$.

Important : dans cette question et toutes les autres, on donnera systématiquement les noms des règles employées à chaque étape de chaque réduction.

2. (0) En déduire que, pour tout λ-terme v, pour toute liste ℓ de variables contenant toutes les variables libres de v, pour tout entier k, pour toutes variables $x_1, \ldots, x_k$ non libres dans v :

$$v^\bullet(\ell) \underbrace{[\uparrow] \ldots [\uparrow]}_{k \text{ fois}} \to_\upsilon^* v^\bullet(x_1 :: \ldots :: x_k :: \ell)$$

3. (2) Montrer que, pour tous λ-termes u et v, pour toute liste de variables ℓ contenant toutes les variables libres de u et de v, pour toute variable x, pour tout entier k, pour toutes variables $x_1, \ldots, x_k$ distinctes de x et non libres dans v :

$$u^\bullet(x_1 :: \ldots :: x_k :: x :: \ell)[\Uparrow^k v^\bullet(\ell)/] \to_\upsilon^* (u[x := v])^\bullet(x_1 :: \ldots :: x_k :: \ell)$$

où $\Uparrow^k S$ dénote $\underbrace{\Uparrow \ldots \Uparrow}_{k \text{ fois}} S$.

4. (1) En déduire que si $u \to v$ en λ-calcul, alors pour toute liste ℓ de variables contenant les variables libres de u, $u^\bullet(\ell) \to^+ v^\bullet(\ell)$.

II. Confluence et terminaison de υ.

1. (1) On définit l'interprétation suivante μ_1 des $\lambda\upsilon$-termes (et piles) vers les entiers naturels supérieurs ou égaux à 2 :

$$\mu_1(\underline{n}) = 2^n \quad \mu_1(MN) = \mu_1(M) + \mu_1(N) \quad \mu_1(x) = 2$$
$$\mu_1(\lambda M) = \mu_1(M) + 1 \quad \mu_1(M[S]) = \mu_1(M)\mu_1(S)$$
$$\mu_1(\uparrow) = 2 \quad \mu_1(\Uparrow S) = \mu_1(S) \quad \mu_1(M/) = \mu_1(M)$$

Montrer que si M se réécrit en une étape vers N par υ, alors $\mu_1(M) \geq \mu_1(N)$. Montrer de plus que l'inégalité est stricte, c'est-à-dire que $\mu_1(M) > \mu_1(N)$, dès que la règle utilisée n'est ni (App) ni $(\uparrow\Uparrow)$.

2. (2) De façon similaire, trouver une interprétation μ_2 des $\lambda\upsilon$-termes (et piles) vers les entiers naturels supérieurs ou égaux à 1 telle que, pour tous termes ou piles M et N, si M se réécrit en une étape vers N par l'une des règles (App) ou $(\uparrow\Uparrow)$, alors $\mu_2(M) > \mu_2(N)$.
3. (1) En déduire que υ termine.
4. (0) On admettra que υ est localement confluent. (Comme dit en cours, ceci est vérifiable par machine, bien que le cas présent ayant un nombre infini de règles de réduction pose quelques problèmes pratiques.) Pourquoi υ est-il confluent sur les termes clos ?

III. Confluence du $\lambda\upsilon$-calcul.

1. (0) Montrer, en considérant le $\lambda\upsilon$-terme $(\lambda x)N[\uparrow]$, où x est une variable et N est un terme normal que l'on choisira, que le $\lambda\upsilon$-calcul n'est *pas* confluent.

   Dans la suite, on va montrer que le problème est dû aux variables (x en l'occurrence ici). On ne considérera dans la suite de la partie III et même en partie IV que des $\lambda\upsilon$-termes *clos*.
2. (0) Démontrer que toute pile du $\lambda\upsilon$-calcul est de la forme $\Uparrow^p (M/)$ pour un certain terme M, ou bien $\Uparrow^p (\uparrow)$.
3. (3) Définir une traduction des $\lambda\upsilon$-termes clos M en λ-termes $M^\circ(P)$, paramétrée par une liste P de λ-termes, comme dans le cours (partie sur le $\lambda\sigma$-calcul) ; comme dans le cours, $M^\circ(P)$ ne sera défini que pour P suffisamment longue. Cette traduction devra être telle que :
   (a) $\underline{n}^\circ(P)$ est défini dès que la longueur de P est d'au moins n, et vaut le $n$ième élément de la liste P ;
   (b) pour tous $\lambda\upsilon$-termes M et N, si $M \to N$ en $\lambda\upsilon$-calcul et $M^\circ(P)$ est défini, alors $N^\circ(P)$ aussi et $M^\circ(P) \to^* N^\circ(P)$ en λ-calcul ;
   (c) si $M^\circ(P)$ est défini et $P = [u_1; \ldots; u_m]$, alors $(M^\circ(P))^\bullet(\ell)$ et
   $$M\, \underbrace{[\Uparrow^m\uparrow]...[\Uparrow^m\uparrow]}_{n \text{ fois}}[\Uparrow^{m-1} (u_m^\bullet(\ell)/)]...[\Uparrow^2 u_3^\bullet(\ell)/][\Uparrow u_2^\bullet(\ell)/][u_1^\bullet(\ell)/]$$
   sont σ-convertibles, où n est la longueur de ℓ.

   Justifier.
4. (2) Déduire du dernier point ci-dessus que si $P = \ell = [x_1; \ldots; x_n]$, et si $M^\circ(P)$ est défini, alors $M \to_\upsilon^* (M^\circ(P))^\bullet(\ell)$.
5. (1) Utiliser la technique d'interprétation de Hardin et en déduire que $\lambda\upsilon$ est confluent sur les $\lambda\upsilon$-termes clos. Pour être précis, montrer que si M, N_1, N_2 sont des $\lambda\upsilon$-termes clos et que $M \to^* N_1$ et $M \to^* N_2$, alors il existe un $\lambda\upsilon$-terme clos N tel que $N_1 \to^* N$ et $N_2 \to^* N$.

IV. Machines à réduction pour $\lambda\upsilon$.

On cherche à définir une machine à réduction, dans le style de la machine de Krivine pour $\lambda\sigma$ du cours. Une *configuration* de la machine est un triplet $M, R, args$, où M est un $\lambda\upsilon$-terme clos, R est une liste $S_1; S_2; \ldots; S_n$ de $\lambda\upsilon$-piles closes S_i, $1 \leq i \leq n$, et $args$ est une liste d'arguments $N_1; N_2; \ldots; N_k$, tous clos. La sémantique intuitive de $M, R, args$ est alors le $\lambda\upsilon$-terme $S(M, R, args) = M[S_1][S_2]\ldots[S_n]N_1N_2\ldots N_k$.

Si $R = S_1; S_2; \ldots; S_n$, on notera $M[[R]]$ le terme $M[S_1][S_2]\ldots[S_n]$. On notera $\Uparrow R$ la liste $\Uparrow S_1; \Uparrow S_2; \ldots; \Uparrow S_n$. On notera aussi $R; S$ la liste $S_1; S_2; \ldots; S_n; S$ et $S; R$ la liste $S; S_1; S_2; \ldots; S_n$.

1. (3) Produire la machine, sous forme d'un nombre fini de règles $M, R, args \to M', R', args'$. Cette machine devra vérifier que :
   (a) si $M, R, args \to M', R', args'$ alors $S(M, R, args)$ se réduit en zéro, une ou plusieurs étapes du $\lambda\upsilon$-calcul en $S(M', R', args')$;
   (b) lorsque la machine termine, c'est sur une configuration de la forme $\lambda M, R, []$ ou bien $\underline{n}, [], args$.

   Justifier. Pourquoi est-il légitime de dire qu'il s'agit d'une machine de réduction de tête faible ?