λ-calcul et logique informatique

bury@lsv.ens-cachan.fr

Exercice 1 — Stratégie interne faible
La stratégie interne faible, aussi appelée *appel par valeur*, est aussi proche que possible des langages de programmation traditionnels. Elle ne réduit jamais sous les abstractions, et ne réduit un β-redex que lorsque son argument est déja complètement réduit — on dit alors que c'est une *valeur*.

On définit l'ensemble des valeurs V par la grammaire suivante, où $\mathcal{V}$ est l'ensemble des variables et Λ l'ensemble de tous les termes :

$$V := \mathcal{V}V \dots V \mid \lambda x.\Lambda$$

On définit formellement notre stratégie comme suit :

$$\frac{v \in V}{(\lambda x.u)v \rhd u[x := v]} \qquad \frac{u \rhd u'}{uv \rhd u'v} \qquad \frac{u \in V \quad v \rhd v'}{uv \rhd uv'}$$

1. Utiliser le combinateur Y pour définir la fonction factorielle, de sorte que **fact** $\overline{n} \to_\beta^* \overline{n!}$. On rappelle la définition :

$$Y = \lambda f.\ (\lambda x.\ f\ (x\ x))\ (\lambda x.\ f\ (x\ x))$$

2. Quelles sont les réductions de **fact** (sans argument) par $\rhd^*$? Cela pose-t-il problème ? Si oui, proposer une solution.
3. Quelles sont les réductions de **fact** $\overline{0}$ par $\rhd^*$? Cela pose-t-il problème ? Si oui, proposer une solution.

Exercice 2 — Combinateurs
On définit $\mathcal{C}$ le calcul de combinateurs **SK** : les termes sont construits suivant la grammaire

$$M := \mathcal{V} \mid (M\ M) \mid \mathbf{S} \mid \mathbf{K}$$

et la réduction est plus petite congruence contenant

$$\mathbf{K}\ M\ N \to M \qquad \mathbf{S}\ M\ N\ P \to (M\ P)\ (N\ P)$$

1. On pose $\mathbf{I} := \mathbf{S}\ \mathbf{K}\ \mathbf{K}$. Réduire $\mathbf{I}\ M$ pour un terme M quelconque.
2. Construire une traduction de $\mathcal{C}$ dans Λ tel que $M \to N$ implique $[M]_\mathcal{C} \to_\beta [N]_\mathcal{C}$ pour tous $M, N \in \mathcal{C}$.
3. Définir une construction λ^*, prenant une variable et un terme de $\mathcal{C}$ et renvoyant un nouveau terme de $\mathcal{C}$ de sorte que $\lambda^*(x, M)N \to^* M[x := N]$ pour tout N. On procèdera par induction sur M, en commençant par les variables, et en utilisant le "distributeur" **S** pour l'application.

4. A-t-on $\mathbf{K} \leftrightarrow^* \lambda^*(x, \lambda^*(y, x))$?
5. Définir une traduction de Λ dans $\mathcal{C}$ telle que
   — $u \rhd v$ implique $[u]_\Lambda \to^* [v]_\Lambda$.
   — pour tout lambda-terme u, u est β-équivalent à $[[u]_\Lambda]_\mathcal{C}$

Exercice 3 — Autour de l'équivalence comportementale
Un contexte est un λ-terme avec un trou, qu'on représentera par un terme spécial $[]$. L'opération de mise en contexte $C[u]$ est définie comme le remplacement *textuel* de ce trou par le terme u : contrairement à la substitution $C[[] := u]$, il est important que ce remplacement provoque la capture des variables libres de u. Par exemple, si $C = (\lambda x.\ [])$ et $C' = (\lambda y.\ [])$ alors $C[x] = (\lambda x.\ x)$ est un terme différent de $C'[x] = (\lambda y.\ x)$.

Un contexte ne contient pas forcément un unique trou : on peut avoir $C[u] = u\ u$ pour tout u (deux trous), et aussi $C[u] = v$ pour tout u (aucun trou). On peut enfin naturellement généraliser la notion de contexte à plusieurs arguments, pour parler de contextes comme $C[u][v] = u\ (u\ v)$ pour tout u et v.

Pour faciliter le raisonnement sur les réductions de termes en contexte, nous admettrons le résultat suivant : Soient $u_1, \ldots, u_n$ des termes clos et C un contexte n-aire tel que $C[\vec{u}] \to v$. Alors on a :

(1) La réduction est dans C : v peut s'écrire $C'[\vec{u}]$ tel que pour tous termes $\vec{t}$ clos on a $C[\vec{t}] \to C'[\vec{t}]$.

ou (2) La réduction est dans un certain u_i : $C[\vec{t}] = C'[\vec{t}, t_i]$ pour tout $\vec{t}$, et $v = C'[\vec{u}, u']$ avec $u_i \to u'$.

ou (3) La réduction implique C et un u_i : $C[\vec{t}] = C'[\vec{t}, t_i\ w]$ pour tout $\vec{t}$, $u_i = (\lambda x.\ u')$ et $v = C'[\vec{u}, u'[x := w]]$.

On dit que deux termes u et v sont *séparables* s'il existe un contexte C tel que $C[u] \to^* T$ et $C[v] \to^* F$.

1. Montrer que la séparabilité est une relation irréflexive et symétrique.
2. Montrer que $(\lambda f \lambda x.\ f\ x)$ et $(\lambda f \lambda x.\ f\ (f\ x))$ sont séparables.
3. Montrer que si $C[\Omega] \to^* t$ avec t en forme normale, alors pour tout u clos on a aussi $C[u] \to^* t$.
   En déduire que I et Ω sont inséparables.

On écrit $u \downarrow_\beta$ quand u normalise faiblement pour la β-réduction, *i.e.*, $u \to^* t$ avec t sans β-redex. On dit que u et v sont en *équivalence comportementale*, noté $u \approx v$, si pour tout contexte C on a $C[u] \downarrow_\beta$ ssi $C[v] \downarrow_\beta$. De façon évidente, $\approx$ est une relation d'équivalence et une congruence, et deux termes α-équivalents sont en équivalence comportementale.

4. Montrer que $\to$ est contenue dans $\approx$.
5. Soient u, v tels que $u \to^* \lambda x.\ v$. Montrer $u \approx (\lambda y.\ u\ y)$ pour $y \notin \mathrm{FV}(u)$.
6. Montrer que si v n'est pas résoluble alors, pour toute substitution θ, $v\theta$ n'est pas résoluble.
7. Démontrer $u \not\approx v$ pour u résoluble et v non résoluble.
8. On montre finalement que l'équivalence comportementale est strictement incluse dans l'inséparabilité :
   (a) Donner deux termes t et t' inséparables mais tels que $t \not\approx t'$.
   (b) Montrer que si u et v sont séparables alors ils ne sont pas en équivalence comportementale.