λ-Calcul et Logique Informatique

Jean Goubault David Baelde
{goubault,baelde}@lsv.ens-cachan.fr

Exercice 1 — Chemins sur Λ^*
On définit l'ensemble des chemins d'un terme $u \in \Lambda$, noté $\mathcal{C}(u)$, comme l'ensemble des objets p tels que $p \prec u$ est dérivable dans le système suivant :

$$\frac{}{x \prec x} \quad \frac{P \prec M}{\mathbf{L}\ P \prec M\ N} \quad \frac{P \prec N}{\mathbf{R}\ P \prec M\ N} \quad \frac{P \prec M}{\lambda x.P \prec \lambda x.M}$$

On travaille modulo α, en renommant implicitement les variables liées pour pouvoir appliquer la règle de passage sous l'abstraction. Par conséquent, deux termes α-équivalents ont les mêmes chemins, et l'ensemble des chemins d'un terme est clos par α-équivalence.

1. Donner les chemins des termes $\lambda x.\ x\ y$, $\lambda x.\ x\ x$, et $\lambda x.\ (\lambda y.\ y)\ x$.
2. Que peut-on dire de deux termes qui ont les mêmes chemins ?
3. On étend notre définition de chemins à Λ^* en rajoutant la règle suivante :

$$\frac{P \prec M \quad Q \prec N}{P[x := Q] \prec (\lambda^* x.\ M)\ N}$$

   L'intuition graphique est qu'un chemin du redex est soit un chemin de M ne contenant pas x, soit un chemin de M se terminant sur x au bout duquel on a recollé un chemin de N.

   Montrer que $M \to N$ implique $\mathcal{C}(M) = \mathcal{C}(N)$, où $\to$ est la réduction de Λ^* qui ne touche qu'aux rédexes annotés d'une étoile.
4. Montrer la réciproque : $\mathcal{C}(M) = \mathcal{C}(N)$ implique $M \leftrightarrow^* N$, c'est à dire la convertibilité dans Λ^*.

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

Exercice 3 — Combinateurs

On définit 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 $[M] \to_\beta [N]$ 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 $u \to_\beta v$ implique $[u] \to^* [v]$.