

Un interpréteur du lambda-calcul

Brouillon

Jean-Louis Krivine

Equipe Preuves, programmes et systèmes

Université Paris VII, C.N.R.S.

2 Place Jussieu 75251 Paris cedex 05

e-mail krivine@pps.jussieu.fr

Les termes du λ -calcul sont écrits avec la notation $(u)t$ pour l'application de u à t .

On a trois zones dans la mémoire: la zone des termes (où sont écrits les λ -termes à exécuter), la zone des environnements et la pile.

Dans la zone des environnements, on a des objets des types suivants :

- environnement : c'est une suite finie (e, x_1, \dots, x_k) , où e est l'adresse d'un environnement, et x_1, \dots, x_k des λ -variables (valeurs de variables du λ -calcul).
- λ -variable : c'est un couple (t, e) formé de l'adresse d'un terme (dans la zone des termes) et de l'adresse d'un environnement.

La pile contient des λ -variables.

Une λ -variable est donc une «fermeture», c'est-à-dire le couple d'un terme t et de son environnement e .

Exécution d'un terme

Le terme t_0 à exécuter est écrit, sous forme « compilée » dans la zone des termes. La forme « compilée » d'un terme consiste à remplacer chaque occurrence de λx par λ , et chaque occurrence de variable par un couple d'entiers (ν, k) (voir plus bas). Ceci suppose que le terme t_0 est clos. La zone des termes ne contient donc que des termes clos.

Toutefois, il y a des symboles de constante, dont l'exécution consiste en un programme prédéfini. Par exemple:

- si le symbole de constante est le nom d'un autre terme (clos), le programme consiste à exécuter ce terme.
- les entrées-sorties.

L'exécution consiste à mettre constamment à jour une variable (T, E) . T est l'adresse du sous-terme courant (pointeur d'instruction), et E l'environnement courant.

Au départ, T est l'adresse du premier terme t_0 à exécuter. Comme ce terme est clos, E est le pointeur nul (pointe sur l'environnement vide).

A un instant quelconque, il y a trois cas possibles, suivant que le terme pointé par T est une abstraction, une application ou une variable.

- Exécution de $\lambda x_1 \dots \lambda x_n t$ où t ne commence pas par λ ; T pointe donc à λx_1 .
On crée un environnement (e, x_1, \dots, x_n) : e est l'adresse de E , x_1, \dots, x_n sont dépilées. On met dans E l'adresse de ce nouvel environnement et on va exécuter t (on fait pointer T sur t).
- Exécution de $(t)u$.
On empile la variable (adresse de u, E) et on va exécuter t (T pointe à t , E ne change pas).
- Exécution de x (variable du λ -calcul).
On cherche la valeur de la variable x dans l'environnement E , de la façon suivante : il s'agit d'une occurrence liée de x dans le terme initial t_0 . On lui a donc associé deux entiers ν, k . Si $\nu = 0$, la valeur cherchée est celle de la k -ième variable de l'environnement E . Si $\nu \geq 1$, soient E_1 l'environnement dont l'adresse est dans E , E_2 celui dont l'adresse est dans E_1 , etc. Alors la valeur de x est celle de la k -ième variable de E_ν .
Cette valeur est un couple (T', E') , que l'on met dans (T, E) .

Remarque.

Intuitivement, on considère les symboles $\lambda x, (, x$ du λ -calcul comme des instructions machine :

- « λx » est : dépiler dans x et avancer le pointeur d'instruction.
- « $($ » est : empiler l'adresse de la « $)$ » correspondante et avancer le pointeur d'instruction.
- « x » est : aller à l'adresse contenue dans x .

Il reste à indiquer comment on détermine les entiers ν, k pour chaque occurrence d'une variable x , c'est-à-dire comment on « compile » un terme clos t . Plus généralement, on va calculer ν pour une occurrence de x dans un terme t quelconque, et k lorsque cette occurrence est liée dans t ; ceci par récurrence sur la longueur de t .

Si $t = x$, on pose $\nu = 0$. Si $t = uv$, l'occurrence considérée de x est dans u (resp. v). On calcule ν , et éventuellement k , dans u (resp. v).

Soit maintenant $t = \lambda x_1 \dots \lambda x_n u$ avec $n > 0$, u étant un terme qui ne commence pas par λ . Si l'occurrence considérée de x est libre dans t , on calcule ν dans t en calculant ν dans u et en ajoutant 1. Si l'occurrence considérée de x est liée dans u , on calcule ν et k dans u . Enfin, si cette occurrence est libre dans u et liée dans t , on a alors $x = x_i$. On calcule ν dans u , et on pose $k = i$.

Comme on le voit, il s'agit d'une variante de la notation de de Bruijn pour les λ -termes.
