

Interprétation (environnements)

Luc.Maranget@inria.fr

<http://www.enseignement.polytechnique.fr/profs/informatique/Luc.Maranget/TLP/>

- A Un petit tour autour du point fixe.
- B De l'évaluation à l'interprétation : environnements.
- C Interprétation par valeur.

Point fixe et réduction

On s'intéresse aux « solutions » d'équations du genre :

$x =_s t$, avec x variable, et t terme de PCF où x apparaît.

« $=_s$ » est l'égalité dans la sémantique.

Soit $u = \mathbf{Fix} \ x \rightarrow t$ avec

$$u \longrightarrow [u \backslash x]t$$

C'est-à-dire, inventons le point fixe explicite.

On pose l'exigence sémantique $t \longrightarrow t' \Rightarrow t =_s t'$.

Il est alors évident que u est solution de $x =_s t$.

Point fixe construit

Une première étape : « abstraire x dans t ».

$$x = (\mathbf{Fun} \ y \rightarrow t') \ x, \text{ avec } t' = [y \backslash x]t.$$

- On a trivialement $(\mathbf{Fun} \ y \rightarrow t') \ x \longrightarrow t$.
- Il apparaît clairement que l'on recherche le point-fixe $(x = \Phi \ x)$ d'une fonction $\Phi = \mathbf{Fun} \ y \rightarrow t'$.

Soit un terme u tel que $u \longrightarrow^* \Phi \ u \longrightarrow [u \backslash y]t'$,

Soit en fait (car $t' = [y \backslash x]t$) :

$$u \longrightarrow^* [u \backslash y][y \backslash x]t = [u \backslash x]t$$

C'est à dire que u est solution.

Construction explicite

De u tel que $u \longleftrightarrow^* \Phi u$.

Soit la fonction suivante `curry` :

Fun $f \rightarrow (\mathbf{Fun} \ x \rightarrow f \ (x \ x)) \ (\mathbf{Fun} \ x \rightarrow f \ (x \ x))$

On va montrer que pour tout terme Φ :

$$\text{curry } \Phi \longleftrightarrow^* \Phi (\text{curry } \Phi)$$

(\longleftrightarrow^* fermeture réflexive-symétrique-transitive de \longrightarrow , entraîne $=_s$).

Démonstration de $\text{curry } \Phi \longleftrightarrow^* \Phi (\text{curry } \Phi)$

Posons $v = \mathbf{Fun} \ x \rightarrow f \ (x \ x)$ Par une β -réduction il vient :

$$v \ v \longrightarrow f \ (v \ v)$$

Lemme : $t \longrightarrow t' \Rightarrow [u \setminus x] \longrightarrow [u \setminus x]t'$

$$\text{Donc : } [\Phi \setminus f](v \ v) \longrightarrow [\Phi \setminus f]f \ (v \ v). \tag{1}$$

$$\text{Or (une } \beta) : \text{curry } \Phi \longrightarrow [\Phi \setminus f](v \ v) \tag{2}$$

Et c'est fini :

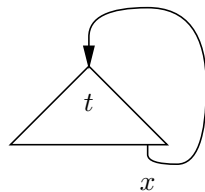
$$\text{curry } \Phi \longrightarrow [\Phi \setminus f](v \ v) \longrightarrow [\Phi \setminus f](f \ (v \ v)) = \Phi ([\Phi \setminus f](v \ v)) \longleftarrow \Phi (\text{curry } \Phi)$$

Astuce ultime

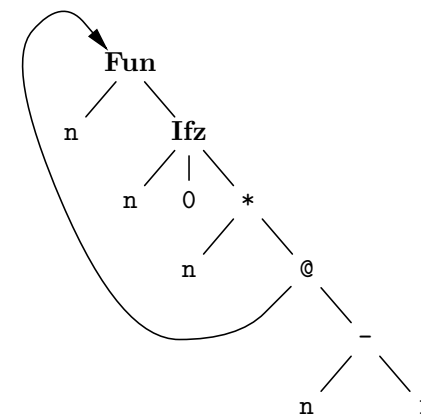
Notons $F = \mathbf{Fix} \ x \rightarrow t$. On effectue les substitution de x dans t , dès que prêt :

$$F \rightarrow [F \setminus x]t \rightarrow [[F \setminus x]t \setminus x]t \rightarrow [[[F \setminus x]t \setminus x]t \setminus x]t \rightarrow \dots$$

On obtient un terme « infini », représentable par un graphe :



Plus clair sur un exemple



Fabriquer un tel graphe

- En Caml, une définition récursive !

```
let rec pcf_fact =
  Fun ("n",
    Ifz (Var "n", Num 0,
      Op (Mul, Var "n",
        App (pcf_fact, Op (Sub, Var "n", Num 1))))))
```

- En Java/langage machine : modification d'un terme.

```
Pcf node =
  new App (null, new Op(Sub, new Var ("n"), new Num (1))) ;
Pcf top = new Fun("n",new Ifz (...)) ;
node.left = top
```

Environnements

Appel par nom

$$\frac{t_1 \hookrightarrow_n \mathbf{Fun} x \rightarrow t_3 \quad [t_2 \backslash x]t_3 \hookrightarrow_n v}{t_1 t_2 \hookrightarrow_n v}$$

$$\mathbf{Fun} x \rightarrow t \hookrightarrow_n \mathbf{Fun} x \rightarrow t$$

$$\frac{[t_1 \backslash x]t_2 \hookrightarrow_n v}{\mathbf{Let} x = t_1 \mathbf{In} t_2 \hookrightarrow_n v} \quad \frac{[(\mathbf{Fix} x \rightarrow t) \backslash x]t \hookrightarrow_n v}{\mathbf{Fix} x \rightarrow t \hookrightarrow_n v}$$

$$\frac{t_1 \hookrightarrow_n 0 \quad t_2 \hookrightarrow_n v}{\mathbf{Ifz} t_1 \mathbf{Then} t_2 \mathbf{Else} t_3 \hookrightarrow_n v} \quad \frac{t_1 \hookrightarrow_n n(n \neq 0) \quad t_3 \hookrightarrow_n v}{\mathbf{Ifz} t_1 \mathbf{Then} t_2 \mathbf{Else} t_3 \hookrightarrow_n v}$$

$$\frac{t_1 \hookrightarrow_n n_1 \quad t_2 \hookrightarrow_n n_2}{t_1 \mathbf{op} t_2 \hookrightarrow_n n_1 \mathbf{op} n_2} \quad n \hookrightarrow_n n$$

Appel par valeur

$$\frac{t_1 \hookrightarrow_v \mathbf{Fun} x \rightarrow t_3 \quad t_2 \hookrightarrow_v v_2 \quad [v_2 \backslash x]t_3 \hookrightarrow_v v}{t_1 t_2 \hookrightarrow_v v}$$

$$\mathbf{Fun} x \rightarrow t \hookrightarrow_v \mathbf{Fun} x \rightarrow t$$

$$\frac{t_1 \hookrightarrow_v v_1 \quad [v_1 \backslash x]t_2 \hookrightarrow_v v}{\mathbf{Let} x = t_1 \mathbf{In} t_2 \hookrightarrow_v v} \quad \frac{[(\mathbf{Fix} x \rightarrow t) \backslash x]t \hookrightarrow_v v}{\mathbf{Fix} x \rightarrow t \hookrightarrow_v v}$$

$$\frac{t_1 \hookrightarrow_v 0 \quad t_2 \hookrightarrow_v v}{\mathbf{Ifz} t_1 \mathbf{Then} t_2 \mathbf{Else} t_3 \hookrightarrow_v v} \quad \frac{t_1 \hookrightarrow_v n(n \neq 0) \quad t_3 \hookrightarrow_v v}{\mathbf{Ifz} t_1 \mathbf{Then} t_2 \mathbf{Else} t_3 \hookrightarrow_v v}$$

$$\frac{t_1 \hookrightarrow_v n_1 \quad t_2 \hookrightarrow_v n_2}{t_1 \mathbf{op} t_2 \hookrightarrow_v n_1 \mathbf{op} n_2} \quad n \hookrightarrow_v n$$

Substitution (dans un évaluateur)

Pour évaluer $(\mathbf{Fun} \ x \rightarrow t_1) \ t_2$, on calcule $[t_2 \setminus x]t_1$ (ou $[v_2 \setminus x]t_1$)

Si t_2 n'a pas de variable libre, alors la substitution est simplifiée :

$$[t_2 \setminus x](\mathbf{Fun} \ x \rightarrow t_1) = \mathbf{Fun} \ x \rightarrow t_1$$

$$[t_2 \setminus x](\mathbf{Fun} \ y \rightarrow t_1) = \mathbf{Fun} \ y \rightarrow [t_2 \setminus x]t_1$$

Et c'est tout, car y n'est certainement pas libre dans t_2 .

Toutefois,

- ▶ La substitution reste chère (copie de termes).
- ▶ Les corps de fonctions sont transformés (ce qui éloigne du modèle où les fonctions sont du code).

On veut mélanger évaluation et substitution, comme on avait mélangé recherche de radical et réduction.

Substitution

Pour évaluer $(\mathbf{Fun} \ x \rightarrow (x * x) + x) \ 4$ — noté $(\mathbf{Fun} \ x \rightarrow t) \ 4$.

$$\frac{\frac{4 \hookrightarrow_n 4 \quad 4 \hookrightarrow_n 4}{4 * 4 \hookrightarrow_n 16} \quad 4 \hookrightarrow_n 4}{(\mathbf{Fun} \ x \rightarrow t) \hookrightarrow_n (\mathbf{Fun} \ x \rightarrow t) \quad (4 * 4) + 4 \hookrightarrow_n 20} \quad (\mathbf{Fun} \ x \rightarrow t) \ 4 \hookrightarrow_n 20$$

- ▶ Évaluer $\mathbf{Fun} \ x \rightarrow t$.
- ▶ Effectuer la substitution $[4 \setminus x]t$, ce qui donne $(4 * 4) + 4$.
- ▶ Et un peu d'arithmétique (calculer $(4 * 4) + 4$).

Une alternative

- ▶ Garder la définition $x = 4$ dans un *environnement*.
- ▶ Évaluer le terme ouvert dans cet environnement, noté $E \vdash t \hookrightarrow v$.

$$\frac{\frac{[x = 4] \vdash x \hookrightarrow 4 \quad [x = 4] \vdash x \hookrightarrow 4}{[x = 4] \vdash (x * x) \hookrightarrow 16} \quad [x = 4] \vdash x \hookrightarrow 4}{[x = 4] \vdash (x * x) + x \hookrightarrow 20}$$

- ▶ La règle nouvelle étant du genre

$$[x = n] \vdash x \hookrightarrow n$$

- ▶ Cela revient à retarder la substitution, jusqu'à l'évaluation du corps de la fonction.

Priorité (à droite)

Valeur de $\mathbf{Let} \ x = 4 \ \mathbf{In} \ \mathbf{Let} \ x = 5 \ \mathbf{In} \ x$?

En toute rigueur

$$\frac{\frac{[5 \setminus x]x \hookrightarrow_n 5}{\mathbf{Let} \ x = 5 \ \mathbf{In} \ x \hookrightarrow_n 5}}{[4 \setminus x](\mathbf{Let} \ x = 5 \ \mathbf{In} \ x) \hookrightarrow_n 5} \quad \mathbf{Let} \ x = 4 \ \mathbf{In} \ \mathbf{Let} \ x = 5 \ \mathbf{In} \ x \hookrightarrow_n 5$$

- ▶ Effectuer $[4 \setminus x](\mathbf{Let} \ x = 5 \ \mathbf{In} \ x)$, soit $\mathbf{Let} \ x = 5 \ \mathbf{In} \ x$.
- ▶ Effectuer $[5 \setminus x]x$, soit 5.
- ▶ Évaluer le résultat (qui est en forme normale).
- ▶ En gros, on a calculé $[4 \setminus x]([5 \setminus x]x)$.

Priorité avec les environnements

Un environnement contient maintenant plusieurs définitions

$x = n$, et on applique la priorité.

$$\frac{\frac{[x = 4, x = 5] \vdash x \leftrightarrow 5}{[x = 4] \vdash \mathbf{Let} \ x = 5 \ \mathbf{In} \ x \leftrightarrow 5}}{\vdash \mathbf{Let} \ x = 4 \ \mathbf{In} \ \mathbf{Let} \ x = 5 \ \mathbf{In} \ x \leftrightarrow 5}}$$

- ▶ Évaluer **Let** $x = 5$ **In** x dans l'env. $[x = 4]$.
- ▶ Évaluer x dans l'env. $[x = 4, x = 5]$.
- ▶ Ce qui nous donne 5 qui est une valeur.
- ▶ Les définitions de l'env. sont à considérer dans l'ordre.

Formalisation de la liaison lexicale

On pourrait écrire :

$$[\dots, x = n] \vdash x \leftrightarrow n \qquad \frac{[\dots] \vdash x \leftrightarrow n}{[\dots, y = n_y] \vdash x \leftrightarrow n}$$

$$\frac{[\dots, x = n] \vdash t \leftrightarrow v}{[\dots] \vdash \mathbf{Let} \ x = n \ \mathbf{In} \ t \leftrightarrow v}$$

On préfère une notation plus abstraite.

$$\frac{E(x) = n}{E \vdash x \leftrightarrow n} \qquad \frac{E \oplus [x = n] \vdash t \leftrightarrow v}{E \vdash \mathbf{Let} \ x = n \ \mathbf{In} \ t \leftrightarrow v}$$

Les environnements sont des fonctions des variables. Avec un opérateur \oplus défini par

$$(E \oplus [x = n])(x) = n \qquad (E \oplus [x = n])(y) = E(y)$$

Implémentation de la liaison lexicale

Avec de belles listes d'associations :

```
type 'a env = (Var.t * 'a) list
```

Opération « \oplus ».

(* Ajouter l'association de x à t *)

```
let add x t env = (x,t)::env
```

Recherche :

```
let rec find x env = match env with
| [] -> raise (Error ("Variable libre : " ^ x))
| (y,t)::env -> if x=y then t else find x env
```

(C'est `List.assoc`).

La priorité à droite devient la priorité à gauche :

```
find x ((x,t1)::(x,t2)::...) -> t1
```

Mais c'est pareil.

Exemples

Généralisons les définitions des environnements : $[x = t]$ (et non plus `slt [x = n]`).

- ▶ On interprète **Let** $y = 4 + 5$ **In** $y + 3$ dans l'env. vide.
- ▶ On interprète $y + 3$ dans $[y = (4 + 5)]$.
- ▶ On interprète y dans $[y = (4 + 5)]$.
- ▶ On interprète $4 + 5$ dans... peu importe : terme clos.

Recommençons.

- ▶ On interprète **Let** $y = 4 + x$ **In** $y + 3$ dans $[x = 5]$.
- ▶ On interprète $y + 3$ dans $[x = 5, y = (4 + x)]$.
- ▶ On interprète y dans $[x = 5, y = (4 + x)]$.
- ▶ On interprète $4 + x$ dans... $[x = 5]$?

Le glaçon (*think*)

Dans l'exemple précédent, nous avons négligé la définition de la substitution.

- On interprète **Let** $y = 4 + x$ **In** $y + 3$ dans $[x = 5]$.

Donc on considère

$$[5 \setminus x](\mathbf{Let} \ y = 4 + x \ \mathbf{In} \ y + 3)$$

C'est-à-dire

$$\mathbf{Let} \ y = [5 \setminus x](4 + x) \ \mathbf{In} \ [5 \setminus x](y + 3)$$

Principe de l'interprétation = ne pas substituer par avance.

Donc, les environnements contiennent des applications de substitutions retardées : des *glaçons*, notés $\langle t \bullet E \rangle$.

Reprenons

Le terme clos 5 se représente facilement comme le glaçon $\langle 5 \bullet [] \rangle$

- On interprète **Let** $y = 4 + x$ **In** $y + 3$ dans $[x = \langle 5 \bullet [] \rangle]$.
- On interprète $y + 3$ dans $[x = \langle 5 \bullet [] \rangle, y = \langle 4 + x \bullet [x = \langle 5 \bullet [] \rangle] \rangle]$.
- On interprète y dans $[x = \langle 5 \bullet [] \rangle, y = \langle 4 + x \bullet [x = \langle 5 \bullet [] \rangle] \rangle]$.
 - ▷ Et donc on évalue $4+x$ dans l'environnement $[x = \langle 5 \bullet [] \rangle]$.
 - ▷ Ce qui finira par donner 9
- Ce qui finira par donner $9 + 3$ et donc 12.

Les règles de l'interprétation

Sachant que l'environnement lie des variables à des glaçons, qui sont eux-mêmes des paires « terme \times environnement ».

Deux règles « faciles »

$$\frac{E(x) = \langle t \bullet E' \rangle \quad E' \vdash t \hookrightarrow v}{E \vdash x \hookrightarrow v} \quad \frac{E \oplus [x = \langle t_1 \bullet E \rangle] \vdash t_2 \hookrightarrow v}{E \vdash \mathbf{Let} \ x = t_1 \ \mathbf{In} \ t_2 \hookrightarrow v}$$

Mais aussi, bien évidemment, l'arithmétique un peu aménagée.

$$E \vdash n \hookrightarrow n \quad \frac{E \vdash t_1 \hookrightarrow n_1 \quad E \vdash t_2 \hookrightarrow n_2}{E \vdash t_1 \ \mathbf{op} \ t_2 \hookrightarrow n_1 \ \mathbf{op} \ n_2}$$

La conditionnelle pareillement aménagée.

$$\frac{E \vdash t_1 \hookrightarrow 0 \quad E \vdash t_2 \hookrightarrow v}{E \vdash \mathbf{Ifz} \ t_1 \ \mathbf{Then} \ t_2 \ \mathbf{Else} \ t_3 \hookrightarrow v}$$

$$\frac{E \vdash t_1 \hookrightarrow n (n \neq 0) \quad E \vdash t_3 \hookrightarrow v}{E \vdash \mathbf{Ifz} \ t_1 \ \mathbf{Then} \ t_2 \ \mathbf{Else} \ t_3 \hookrightarrow v}$$

Et le point fixe pareillement aménagée.

$$\frac{E \oplus [x = \langle \mathbf{Fix} \ x \rightarrow t \bullet E \rangle] \vdash t \hookrightarrow v}{E \vdash \mathbf{Fix} \ x \rightarrow t \hookrightarrow v}$$

Fermeture

Nous avons la règle d'évaluation des fonctions

$$\mathbf{Fun} \ x \rightarrow t \hookrightarrow \mathbf{Fun} \ x \rightarrow t$$

Quelle peut-être la règle d'interprétation ?

$$E \vdash \mathbf{Fun} \ x \rightarrow t \hookrightarrow \langle x \bullet t \bullet E \rangle$$

On a très envie de dire $\langle \mathbf{Fun} \ x \rightarrow t \bullet E \rangle$, mais cela compliquerait exagérément le concept de valeur. On définit plutôt la *fermeture* $\langle x \bullet t \bullet E \rangle$ comme ce glaçon là.

Nouvelle règle de l'application.

$$\frac{E \vdash t_1 \hookrightarrow \langle x \bullet t'_1 \bullet E' \rangle \quad E' \oplus [x = \langle t_2 \bullet E \rangle] \vdash t'_1 \hookrightarrow v}{E \vdash t_1 t_2 \hookrightarrow v}$$

NB : Les valeurs ne sont plus des termes.

Impact de la fermeture

Quelle est la valeur du terme

```
Let x = 4 In
Let f = Fun y -> y + x In
Let x = 5 In
f x
```

Le point crucial est de savoir à quoi se rapporte x dans $y + x$.

À la définition de x active au moment de la définition de f (et non pas de son application).

Et donc le résultat est 9 (et non pas 10).

La fermeture est ici cruciale, f est liée au glaçon

$$\langle \mathbf{Fun} \ y \rightarrow y + x \bullet [x = \langle 4 \bullet [] \rangle] \rangle$$

Dont la valeur est la fermeture : $\langle y \bullet y + x \bullet [x = \langle 4 \bullet [] \rangle] \rangle$.

Et $y + x$ s'évalue dans $[x = \langle 4 \bullet [] \rangle, y = \langle x \bullet [\dots, x = \langle 5 \bullet \dots \rangle] \rangle]$.

$$\frac{\frac{\frac{[x = \dots, f = \dots, x = \langle 5 \bullet [\dots] \rangle] \vdash f \ 5 \hookrightarrow 10}{[x = \dots, f = \langle \mathbf{Fun} \ y \rightarrow y + x \bullet [x = \langle 4 \bullet [] \rangle] \rangle] \vdash \mathbf{Let} \ x = 5 \ \mathbf{In} \ \dots \hookrightarrow 10}}{[x = \langle 4 \bullet [] \rangle] \vdash \mathbf{Let} \ f = \mathbf{Fun} \ y \rightarrow y + x \ \mathbf{In} \ \dots \hookrightarrow 10}}{\vdash \mathbf{Let} \ x = 4 \ \mathbf{In} \ \dots \hookrightarrow 10}$$

Toutes les règles

Les valeurs v sont des entiers n ou des fermetures $\langle x \bullet t \bullet E \rangle$.

Les environnements E lient les variables x à des glaçons $\langle t \bullet E \rangle$.

$$E \vdash \mathbf{Fun} \ x \rightarrow t \hookrightarrow_n \langle x \bullet t \bullet E \rangle$$

$$\frac{E \vdash t_1 \hookrightarrow_n \langle x \bullet t'_1 \bullet E' \rangle \quad E' \oplus [x = \langle t_2 \bullet E \rangle] \vdash t'_1 \hookrightarrow_n v}{E \vdash t_1 t_2 \hookrightarrow_n v}$$

$$\frac{E \oplus [x = \langle \mathbf{Fix} \ x \rightarrow t \bullet E \rangle] \vdash t \hookrightarrow_n v}{E \vdash \mathbf{Fix} \ x \rightarrow t \hookrightarrow_n v}$$

$$\frac{E(x) = \langle t \bullet E' \rangle \quad E' \vdash t \hookrightarrow_n v}{E \vdash x \hookrightarrow_n v} \quad \frac{E \oplus [x = \langle t_1 \bullet E \rangle] \vdash t_2 \hookrightarrow_n v}{E \vdash \mathbf{Let} \ x = t_1 \ \mathbf{In} \ t_2 \hookrightarrow_n v}$$

$$E \vdash n \hookrightarrow_n n \quad \frac{E \vdash t_1 \hookrightarrow_n n_1 \quad E \vdash t_2 \hookrightarrow_n n_2}{E \vdash t_1 \ \mathbf{op} \ t_2 \hookrightarrow_n n_1 \ \mathbf{op} \ n_2}$$

$$\frac{E \vdash t_1 \hookrightarrow_n 0 \quad E \vdash t_2 \hookrightarrow_n v}{E \vdash \mathbf{Ifz} \ t_1 \ \mathbf{Then} \ t_2 \ \mathbf{Else} \ t_3 \hookrightarrow_n v} \quad \frac{E \vdash t_1 \hookrightarrow_n n (n \neq 0) \quad E \vdash t_3 \hookrightarrow_n v}{E \vdash \mathbf{Ifz} \ t_1 \ \mathbf{Then} \ t_2 \ \mathbf{Else} \ t_3 \hookrightarrow_n v}$$

Interpréter par valeur

Notable simplification

Les règles de l'évaluation par valeur.

$$\frac{t_1 \hookrightarrow_v \mathbf{Fun} \ x \rightarrow t_3 \quad t_2 \hookrightarrow_v v_2 \quad [v_2 \backslash x]t_3 \hookrightarrow_v v}{t_1 \ t_2 \hookrightarrow_v v}$$

$$\frac{t_1 \hookrightarrow_v v_1 \quad [v_1 \backslash x]t_2 \hookrightarrow_v v}{\mathbf{Let} \ x = t_1 \ \mathbf{In} \ t_2 \hookrightarrow_v v}$$

Les substitutions retardées sont de la forme $[v \backslash x] \dots$

Comme v est n (entier) ou $\langle x \bullet t \bullet E \rangle$ (fermeture), plus besoin de glaçon.

Malheureusement, il y a la récursion.

$$\frac{[(\mathbf{Fix} \ x \rightarrow t) \backslash x]t \hookrightarrow_v v}{\mathbf{Fix} \ x \rightarrow t \hookrightarrow_v v}$$

Interprétation par valeur, simplicité ?

Récursion avec environnement (et glaçon) :

$$\frac{E \oplus [x = \langle \mathbf{Fix} \ x \rightarrow t \bullet E \rangle] \vdash t \hookrightarrow_v v}{E \vdash \mathbf{Fix} \ x \rightarrow t \hookrightarrow_v v}$$

Donc : les environnements lient des variables à des valeurs v , ou à des glaçons de la forme $\langle \mathbf{Fix} \ x \rightarrow t \bullet E \rangle$ (exclusivement).

Deux règles pour les variables au lieu d'une.

$$\frac{E(x) = v}{E \vdash x \hookrightarrow_v v}$$

$$\frac{E(x) = \langle \mathbf{Fix} \ x \rightarrow t \bullet E' \rangle \quad E' \vdash \mathbf{Fix} \ x \rightarrow t \hookrightarrow_v v}{E \vdash x \hookrightarrow_v v}$$

Et changer les règles de l'application et du **Let**.

$$\frac{E \vdash t_1 \hookrightarrow_v \langle x \bullet t' \bullet E' \rangle \quad E \vdash t_2 \hookrightarrow_v v_2 \quad E' \oplus [x = v_2] \vdash t' \hookrightarrow_v v}{E \vdash t_1 \ t_2 \hookrightarrow_v v}$$

$$\frac{E \vdash t_1 \hookrightarrow_v v_1 \quad E \oplus [x = v_1] \vdash t_2 \hookrightarrow_v v}{E \vdash \mathbf{Let} \ x = t_1 \ \mathbf{In} \ t_2 \hookrightarrow_v v}$$

Et c'est tout, mais il est dommage d'avoir des glaçons, juste pour coder la récursion.

Toutes les règles

Les valeurs v sont des entiers n ou des fermetures $\langle x \bullet t \bullet E \rangle$. Les environnements E lient les variables x à des valeurs v ou à des glaçons de la forme $\langle \mathbf{Fix} \ x \rightarrow t \bullet E \rangle$.

$$\frac{E(x) = v}{E \vdash x \hookrightarrow_v v} \quad \frac{E(x) = \langle \mathbf{Fix} \ x \rightarrow t \bullet E' \rangle \quad E' \vdash \mathbf{Fix} \ x \rightarrow t \hookrightarrow_v v}{E \vdash x \hookrightarrow_v v}$$

$$\frac{E \vdash t_1 \hookrightarrow_v \langle x \bullet t' \bullet E' \rangle \quad E \vdash t_2 \hookrightarrow_v v_2 \quad E' \oplus [x = v_2] \vdash t' \hookrightarrow_v v}{E \vdash t_1 t_2 \hookrightarrow_v v}$$

$$\frac{E \vdash t_1 \hookrightarrow_v v_1 \quad E \oplus [x = v_1] \vdash t_2 \hookrightarrow_v v}{E \vdash \mathbf{Let} \ x = t_1 \ \mathbf{In} \ t_2 \hookrightarrow_v v}$$

$$\frac{E \oplus [x = \langle \mathbf{Fix} \ x \rightarrow t \bullet E \rangle] \vdash t \hookrightarrow_v v}{E \vdash \mathbf{Fix} \ x \rightarrow t \hookrightarrow_v v} \quad E \vdash n \hookrightarrow_v n$$

$$\frac{E \vdash t_1 \hookrightarrow_v n_1 \quad E \vdash t_2 \hookrightarrow_v n_2}{E \vdash t_1 \ \mathbf{op} \ t_2 \hookrightarrow_v n_1 \ \mathbf{op} \ n_2} \quad \frac{E \vdash t_1 \hookrightarrow_v 0 \quad E \vdash t_2 \hookrightarrow_v v}{E \vdash \mathbf{Ifz} \ t_1 \ \mathbf{Then} \ t_2 \ \mathbf{Else} \ t_3 \hookrightarrow_v v}$$

$$\frac{E \vdash t_1 \hookrightarrow_v n(n \neq 0) \quad E \vdash t_3 \hookrightarrow_v v}{E \vdash \mathbf{Ifz} \ t_1 \ \mathbf{Then} \ t_2 \ \mathbf{Else} \ t_3 \hookrightarrow_v v}$$

Une remarque

En appel par valeur, **Fix** sert surtout à définir des fonctions.

Let fact =

Fix f Fun x -> Ifz x Then 1 Else x * f (x - 1) In
...

Interprétation de **Fix** $f \rightarrow \mathbf{Fun} \ x \rightarrow t$, noté **FixFun** $f \ x \rightarrow t$

$$\frac{E \oplus [f = \langle \mathbf{FixFun} \ f \ x \rightarrow t \bullet E \rangle] \vdash \mathbf{Fun} \ x \rightarrow t \hookrightarrow \langle x \bullet t \bullet E' \rangle}{E \vdash \mathbf{Fix} \ f \rightarrow \mathbf{Fun} \ x \rightarrow t \hookrightarrow \langle x \bullet t \bullet E' \rangle}$$

Avec

$$E' = E \oplus [f = \langle \mathbf{FixFun} \ f \ x \rightarrow t \bullet E \rangle]$$

Première variante

Si **Fix** $f \rightarrow \dots$ est nécessairement de la forme **FixFun** $f \ x \rightarrow t$, on peut remplacer les glaçons par une forme particulière de fermeture.

La nouvelle fermeture $\langle f \bullet x \bullet t \bullet E \rangle$, qui représente la fermeture à trois composants de la forme

$$\langle x \bullet t \bullet [f = \langle \mathbf{FixFun} \ f \ x \rightarrow t \bullet E \rangle] \rangle$$

On pose alors directement.

$$E \vdash \mathbf{FixFun} \ f \ x \rightarrow t \hookrightarrow_v \langle f \bullet x \bullet t \bullet E \rangle$$

On a alors une règle supplémentaire pour appliquer les nouvelles fermetures...

$$\frac{E \vdash t_1 \hookrightarrow_v \langle f \bullet x \bullet t \bullet E' \rangle \quad E \vdash t_2 \hookrightarrow_v v_2 \quad E' \oplus [f = \langle f \bullet x \bullet t \bullet E' \rangle, x = v_2] \vdash t \hookrightarrow_v v}{E \vdash t_1 t_2 \hookrightarrow_v v}$$

Astuce

Si on représente une fermeture à trois composants $\langle x \bullet t \bullet E \rangle$ comme une fermeture à quatre composants $\langle _f \bullet x \bullet t \bullet E \rangle$, alors la nouvelle règle d'application des fermetures remplace l'ancienne !

On pose donc :

$$\frac{_f \notin \mathcal{F}(t)}{E \vdash \mathbf{Fun} \ x \rightarrow t \hookrightarrow_v \langle _f \bullet x \bullet t \bullet E \rangle}$$

Et on *supprime* l'ancienne règle d'application.

$$\frac{E \vdash t_1 \hookrightarrow_v \langle x \bullet t' \bullet E' \rangle \quad E \vdash t_2 \hookrightarrow_v v_2 \quad E' \oplus [x = v_2] \vdash t' \hookrightarrow_v v}{E \vdash t_1 t_2 \hookrightarrow_v v}$$

Mais c'est bien artificiel (liaison de $_f$ inutile).

Encore plus malin

Revenons à l'interprétation de $t = \mathbf{FixFun} f x \rightarrow b$ (page 33)

$$\frac{E \oplus [f = g] \vdash \mathbf{Fun} x \rightarrow b \hookrightarrow_v \langle x \bullet b \bullet E \oplus [f = g] \rangle}{E \vdash \mathbf{FixFun} f x \rightarrow b \hookrightarrow_v \langle x \bullet b \bullet E \oplus [f = g] \rangle}$$

Où g note le glaçon $\langle \mathbf{FixFun} f x \rightarrow b \bullet E \rangle$.

Anticipons maintenant l'interprétation de g . Une fois

$$\langle x \bullet b \bullet E \oplus [f = g] \rangle$$

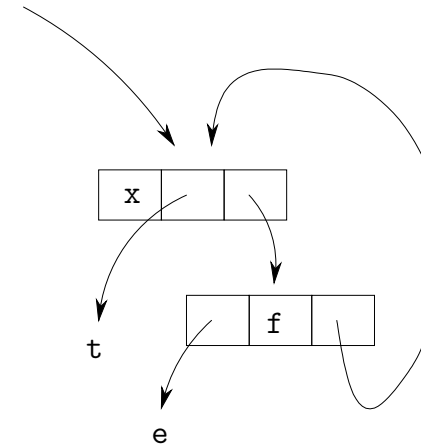
Et encore une fois

$$\langle x \bullet b \bullet E \oplus [f = \langle x \bullet b \bullet E \oplus [f = g] \rangle] \rangle$$

Etc. on obtient une fermeture *ordinaire*, mais *bouclée*, avec :

$$C = \langle x \bullet b \bullet E \oplus [f = C] \rangle$$

Un graphe



Encore plus fort

Soit un interpréteur (qui est au final un programme)

Reste à construire le graphe, pour interpréter $\mathbf{FixFun} f x \rightarrow b$ dans l'env. E , par ex. en Caml.

```
type value = (* Les valeurs *)
| Integer of int | Closure of Var.t * Ast.t * env
and env = (Var.t * value) list (* les environnements *)
```

```
let rec inter E t = match t with
...
| Fix (f, Fun (x, b)) ->
  let rec clo = Closure (x, b, (f, clo)::E) in
  clo
```

Avantage définitif : une seule sorte de fermeture.

► TP, divers interpréteurs.

► La prochaine fois, compilation.