LES FONCTIONS RÉCURSIVES PRIMITIVES SONT λ-DÉFINISSABLES

Leçons 912, 929.

Références Lassaigne, Rougemont – Logique et fdt de l'info p.193

Odifreddi – Classical Recursion Theory p.84.

Théorème

Les fonctions primitives récursives sont λ-définissables.

Résumé

0 – Entiers de Church

Définition de λ-définissable.

I – Fonctions élémentaires

II – Composition

III – Récursion : combinateurs de points fixes.

0 • Les entiers de Church sont les $\underline{n} = \lambda fx.\underbrace{f \cdots f}_{n \text{ fois}} x$

• On dit qu'un λ-terme F représente la fonction $f : \mathbb{N}^n \to \mathbb{N}$ si $\forall k_1, \dots, k_n \in \mathbb{N}$ $f(k_1, \dots, k_n) = k \Rightarrow F\,\underline{k_1} \dots \underline{k_n} =_\beta \underline{k}$.

I • Écrivons les fonctions élémentaires :

- Zéro : $\lambda x.\underline{0}$.
- Projection : $\lambda x_1 \dots x_n . x_i$
- Successeur : $\lambda zfx.\, f z f x$

car alors $(\lambda zfx.\, fzfx)(\lambda gy.\, g^n y)$

$= \lambda fx.\, f(\lambda gy.\, g^n y) f x$

$= \lambda fx.\, f f^n y$

$= \underline{n+1}$

III | • Schéma de composition :

Soit $h : \mathbb{N}^n \to \mathbb{N}$

$g_1, \ldots, g_n : \mathbb{N}^p \to \mathbb{N}$ récursives primitives.

On les représente par $H, G_1, \ldots, G_n$, et on pose

$$F = \lambda x_1 \cdots x_p . H(G_1 \vec{x}) \cdots (G_n \vec{x})$$
$$= H \, \underline{g_1(\vec{x})} \cdots \underline{g_n(\vec{x})}$$
$$= \underline{h(g_1(\vec{x}), \ldots, g_n(\vec{x}))}$$

IV | • La récurrence, c'est un petit peu compliqué, on va utiliser un combinateur de point fixe Y :

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

On a alors pour tout terme F :

$$YF = (\lambda x. F x x)(\lambda x. F x x)$$
$$= F(\lambda x. F x x)(\lambda x. F x x)$$
$$= F(YF)$$

• On va aussi admettre qu'il existe une fonction λ prédécesseur P telle que $\begin{cases} P \, \underline{k+1} = \underline{k} . \\ P \, \underline{0} = \underline{0} . \end{cases}$

• Prenons maintenant des fonctions f, g, h récursives primitives telles que

$$f(\vec{x}, n) = \begin{cases} g(\vec{x}) & \text{si } n = 0 \\ h(\vec{x}, n-1, f(\vec{x}, n-1)) & \text{sinon} \end{cases}$$

où g et h sont récursives primitives représentées par les termes G, H.

Cela peut se traduire par la quête au λ-terme vérifiant :

$$\begin{cases} R\, u v\, \underline{0} = u \\ R\, u v\, \underline{k+1} = v\, \underline{k}\, (R\, u v\, \underline{k}) \end{cases}$$

C'est l'opérateur de récursion !

Avec un tel terme, on peut représenter f par :

$$F = \lambda \vec{x} y .\, R (G \vec{x}) (\lambda ab .\, H \vec{x} ab)\, y$$

Car on a alors

- $F \vec{x}\, \underline{0} = R (G\vec{x})(\lambda ab.\, H\vec{x}ab)\, \underline{0}$
$\qquad = G\vec{x}$

- $F\vec{x}\, \underline{k+1} = R(G\vec{x})(\lambda ab. H\vec{x}ab)\, \underline{k+1}$
$\qquad = (\lambda ab.\, H\vec{x}ab)\, \underline{k}\, (R(G\vec{x})(\lambda ab.\, H\vec{x}ab)\, \underline{k})$
$\qquad = (\lambda ab.\, H\vec{x}ab)\, \underline{k}\, (F\vec{x}\, \underline{k})$
$\qquad = H\vec{x}\, \underline{k}\, (F x\, \underline{k})$

- Il ne reste plus qu'à construire R !

Regardons le terme

$$\mathcal{U} = \lambda xyz .\, z (\lambda d. y)\, x .$$

Il correspond à $\mathcal{U}\, u v\, \underline{0} = \underline{0}\, (\lambda d. v)\, u = u$

$\mathcal{U}\, u v\, \underline{k} = \underline{k}\, (\lambda d. v) u = (\lambda d. v)^k u = v .$

Alors R est le point fixe de

$$\lambda cxyz .\, \mathcal{U} (x) (y (Pz) (c x y (Pz)))\, z .$$

d'où $R = Y (\lambda cxyz .\, \mathcal{U} x (y (Pz) (cxy (Pz))) z)$