λ-calculs

Xavier Urbain

UCBL-1 – M1if09

Histoire

Modèle de calcul (Church, $\simeq 1930$)

But : fondation des mathématiques $\rightsquigarrow$ échec
Toutefois, partie fonctionnelle $\rightsquigarrow$ succès...

- Calculabilité (fonctions récursives, Turing-calculables)
- Programmation fonctionnelle
  - λ-calcul pur $\rightsquigarrow$ **`Lisp`**, etc.
  - λ-calcul typés $\rightsquigarrow$ famille **`ML`**, etc.

Syntaxe

λ-termes :

- Ensemble de variables, infini : X avec procédé pour var. fraîche
- Ensemble des λ-termes, ensemble inductif : Λ $(\Lambda(X))$
  - $x \in X$ alors $x \in \Lambda$
  - $E_1 \in \Lambda$ et $E_2 \in \Lambda$ alors $(E_1\ E_2) \in \Lambda$ application
  - $E \in \Lambda$ et $x \in X$ alors $\lambda x.E \in \Lambda$ abstraction
    « Fonction qui à x associe E »

Exemples :

$\lambda x.x$ « fonction identité »

$\lambda x.c$ « fonction constante c »

$(f\ x)$ « f appliquée à x »

Structure

Naturellement, raisonnement par induction sur les λ-termes :

Soit P prop. sur Λ,

- $P(x)$ pour tout $x \in X$
- $P(E_1)$ et $P(E_2)$ alors $P((E_1\ E_2))$
- $P(E)$ alors $P(\lambda x.E)$

Alors $P(E)$ pour tout $E \in \Lambda$ (car $\Lambda \subseteq \{E$ t.q. $P(E)\}$)

Variables libres

But : exprimer $\lambda x.E$, x « muet » x lié
Ex. : $\lambda x.x$ et $\lambda y.y$ équivalentes pour identité

Variables libres d'un λ-terme t ($FV(t)$) :

- $FV(x) = \{x\}$
- $FV((E_1\ E_2)) = FV(E_1) \cup FV(E_2)$
- $FV(\lambda x.E) = FV(E) - \{x\}$

Variables, substitution

Remplacement de variable par λ-terme dans λ-terme.

λ-terme obtenu en substituant x par F dans E ($E[x \mapsto F]$) :

- $E = y \in X$
  - Si $x = y$ alors F
  - Si $x \neq y$ alors y
- $E = (E_1\ E_2)$ alors $(E_1[x \mapsto F]\ E_2[x \mapsto F])$

Variables, substitution

Remplacement de variable par λ-terme dans λ-terme.

λ-terme obtenu en substituant x par F dans E ($E[x \mapsto F]$) :

- $E = y \in X \ldots$
- $E = (E_1\ E_2) \ldots$
- $E = \lambda y.E_1$
  - Si $x = y$ alors E
  - Sinon
    - Si $y \notin FV(F)$ alors $\lambda y.E_1[x \mapsto F]$
    - Si $y \in FV(F)$ alors changement de nom de y capture
      $z \notin FV(F) \cup FV(E_1) \rightsquigarrow \lambda z.E_1[y \mapsto z][x \mapsto F]$

Variables, α-conversion

Relation α-réduction ($\rightarrow_\alpha$) :

$$\lambda x.E \rightarrow_\alpha \lambda y.E[x \mapsto y] \text{ pour tout } y \notin FV(E)$$

Relation α-conversion ($=_\alpha$) : clôture de $\rightarrow_\alpha$

- Réflexive
- Symétrique
- Transitive
- Par contexte

α-conversion : congruence

Variables, α-conversion

Proposition.
E_1 et E_2 α-convertibles $\Leftrightarrow$

- $E_1 = E_2 = x \in X$
- $E_1 = (F_1\ G_1)$ et $E_2 = (F_2\ G_2)$ où
  $F_1 =_\alpha F_2$ et $G_1 =_\alpha G_2$
- $E_1 = \lambda x.F_1$ et $E_2 = \lambda y.F_2$ où
  $F_1[x \mapsto z] =_\alpha F_2[y \mapsto z]$ avec $z \notin FV(F_1) \cup FV(F_2)$

Dém. par recurrence

Corollaire.
$=_\alpha$: décidable

Désormais on travaille modulo $=_\alpha$-conversion

β-réduction

Expression de l'évaluation d'une fonction appliquée à un λ-terme

Relation β-réduction en un pas ($\rightarrow_\beta$)

$$(\lambda x.E_1\ E_2) \rightarrow_\beta E_1[x \mapsto E_2] \text{ et clôture par contexte}$$

Exemple :

$$\begin{array}{rcl}
(\lambda x.x\ c) & \rightarrow_\beta & x[x \mapsto c] = c \\
((\lambda x.\lambda y.(x\ y)\ b)\ c) & \rightarrow_\beta & (\lambda y.(x\ y)[x \mapsto b]\ c) \\
 & = & (\lambda y.(b\ y)\ c) \\
 & \rightarrow_\beta & (b\ y)[y \mapsto c] \\
 & = & (b\ c)
\end{array}$$

β-réduction

Expression de l'évaluation d'une fonction appliquée à un λ-terme

Relation β-réduction en un pas ($\rightarrow_\beta$)

$$(\lambda x.E_1\ E_2) \rightarrow_\beta E_1[x \mapsto E_2] \text{ et clôture par contexte}$$

Relation β-réduction, (resp. β-réduction stricte) :
Clôture transitive et réflexive (resp. transitive) de $\rightarrow_\beta$
Notée $\rightarrow_\beta^\star$ (resp. $\rightarrow_\beta^+$)

β-réduction

λ-terme E en forme normale s'il n'existe aucun F tel que $E \to_\beta F$

λ-terme F forme normale d'un λ-terme E si $E \to_\beta^\star F$ et F forme normale

$$((\lambda x.\lambda y.(x\ y)\ b)\ c) \to_\beta^\star (c\ b) \qquad \text{forme normale}$$

β-réduction

λ-terme E en forme normale s'il n'existe aucun F tel que $E \to_\beta F$

λ-terme F forme normale d'un λ-terme E si $E \to_\beta^\star F$ et F forme normale

$$\Delta = \lambda x.(x\ x) \qquad \text{forme normale}$$

$$\begin{array}{rclcl} \Omega & = & (\Delta\ \Delta) & = & (\lambda x.(x\ x)\ \Delta) \\ & & & \to_\beta & (x\ x)[x \mapsto \Delta] \\ & & & = & (\Delta\ \Delta) \quad = \quad \Omega \end{array}$$

Corollaire.
Un λ-terme n'a pas toujours de forme normale

Questions

Importance de l'ordre des calculs ?

Unicité du résultat ?

Expressivité ?

β-réduction

Théorème
Si un λ-terme a une forme normale alors celle-ci est unique

Même résultat quel que soit l'ordre des calculs

Théorème Church-Rosser
Si $E \to_\beta^\star E_1$ et $E \to_\beta^\star E_2$
Alors il existe $F \in \Lambda$ t. q. $E_1 \to_\beta^\star F$ et $E_2 \to_\beta^\star F$

C.-R. $\Rightarrow$ th.

β-réduction

$$\lambda y.y \leftarrow_\beta (\lambda x.\lambda y.y\ \Omega) \rightarrow_\beta \cdots$$

Suivant stratégie, forme normale atteinte. . . ou pas !

Réduction normale : choix du rédexe le plus à gauche

Théorème Curry
Si E a une forme normale F alors la réduction normale de E aboutit à F

Expressivité

Codage des entiers : entiers de Church
$n \in \mathbb{N} \quad \mapsto \quad \overline{n} = \lambda f.\lambda x.(\underbrace{f \cdots (f}_{n}\ x) \cdots)$ (fonction associant $f^n(x)$ à f et x)

Fonction successeur $succ$ t. q. pour tout $n \in \mathbb{N}, (succ\ \overline{n}) \rightarrow_\beta^\star \overline{n+1}$:
Ainsi $(((succ\ \overline{n})\ f)\ x) \rightarrow_\beta^\star (\underbrace{f \cdots (f}_{n+1}\ x)) =_\beta (f\ ((\overline{n}\ f)\ x))$
D'où $succ = \lambda n.\lambda f.\lambda x.(f\ ((n\ f)\ x))$

Exo. : add, mul, etc.

XU - UCBL-1 - M1if09 2019-2020 UN JEU DE SLIDES N'EST PAS UN POLY DE RÉFÉRENCE 18