Una prueba constructiva del axioma de elección dependiente en lógica clásica

Hugo Herbelin[1], Étienne Miquey[1,2]

[1]Team πr^2 (INRIA), PPS, Université Paris-Diderot
[2]Fac. de Ingeniería, Universidad de la República, Uruguay

15 de Marzo 2016

En el capítulo anterior...

La meta

1. **Objetivo :** definir un sistema de tipos para tener una **prueba** del axioma de elección dependiente

2. **Bonus :** corrección del sistema sin meta-uso del axioma de elección

3. **Super-bonus :** equiconsistencia con otro sistema lógico

La meta

1. **Objetivo :** definir un sistema de tipos para tener una **prueba** del axioma de elección dependiente
2. **Bonus :** corrección del sistema sin meta-uso del axioma de elección
3. **Super-bonus :** equiconsistencia con otro sistema lógico

La meta

1. **Objetivo :** definir un sistema de tipos para tener una **prueba** del axioma de elección dependiente

2. **Bonus :** corrección del sistema sin meta-uso del axioma de elección

3. **Super-bonus :** equiconsistencia con otro sistema lógico

Teoría de tipos de Martin-Löf (1973)

Ingredientes

$$\frac{\Gamma, a : A \vdash p : B}{\Gamma \vdash \lambda a.p : A \to B} \to_I \qquad \frac{\Gamma, x : T \vdash p : A}{\Gamma \vdash \lambda x.p : \forall x^T A} \forall_I$$

$$\frac{\Gamma \vdash p : A[t/x] \quad \Gamma \vdash t : T}{\Gamma \vdash (t, p) : \exists x^T A} \exists_I \qquad \frac{\Gamma \vdash p : \exists x^T A(x)}{\Gamma \vdash \mathtt{prf}\ p : A(\mathtt{wit}\ p)} \exists_E$$

Axioma de elección

$$AC_A : \forall x^A \exists y^B P(x,y) \to \exists f^{A\to B} \forall x^A P(x, f(x))$$

$$\begin{array}{lcl} AC_A & := & \lambda H.(\lambda x.\mathtt{wit}(Hx), \lambda x.\mathtt{prf}(Hx)) \\ & : & \forall x^A \exists y^B P(x,y) \to \exists f^{A\to B} \forall x^A P(x, f(x)) \end{array}$$

Lógica clásica vs tipos dependientes

- En lógica clásica :

$$H_0 := \texttt{callcc}_{\alpha}(1, \phi(\texttt{throw}_{\alpha}(2, p)) : \exists x P(x)$$

- Restricción a la elección numerable :

$$AC_{\mathbb{N}} : \forall x^{\mathbb{N}} \exists y^{B} P(x, y) \to \exists f^{\mathbb{N} \to B} \forall x^{\mathbb{N}} P(x, f(x))$$

- Prueba :

Lógica clásica vs tipos dependientes

- En lógica clásica :

$$H_0 := \texttt{callcc}_{\alpha}(1, \phi(\texttt{throw}_{\alpha}(2, p))) : \exists x P(x)$$

- Restricción a la elección numerable :

$$AC_{\mathbb{N}} : \forall x^{\mathbb{N}} \exists y^{B} P(x, y) \to \exists f^{\mathbb{N} \to B} \forall x^{\mathbb{N}} P(x, f(x))$$

- Prueba :

Lógica clásica vs tipos dependientes

- En lógica clásica :

$$H_0 := \mathtt{callcc}_{\alpha}(1, \phi(\mathtt{throw}_{\alpha}(2, p)) : \exists x P(x)$$

- Restricción a la elección numerable :

$$AC_{\mathbb{N}} : \forall x^{\mathbb{N}} \exists y^{B} P(x, y) \to \exists f^{\mathbb{N} \to B} \forall x^{\mathbb{N}} P(x, f(x))$$

- Prueba :

Lógica clásica vs tipos dependientes

- En lógica clásica :

$$H_0 := \texttt{callcc}_{\alpha}(1, \phi(\texttt{throw}_{\alpha}(2, p))) : \exists x P(x)$$

- Restricción a la elección numerable :

$$AC_{\mathbb{N}} : \forall x^{\mathbb{N}} \exists y^{B} P(x,y) \to \exists f^{\mathbb{N}\to B} \forall x^{\mathbb{N}} P(x, f(x))$$

- Prueba :

$$AC := \lambda H.(\lambda n.\,\texttt{wit}(Hn), \lambda n.\,\texttt{prf}(Hn))$$

$$\begin{aligned} AC := \lambda H.(\lambda n.&\texttt{if } n = 0 \texttt{ then } \texttt{wit}\, H\ 0 \texttt{ else} \\ &\texttt{if } n = 1 \texttt{ then } \texttt{wit}\, H\ 1 \texttt{ else } \dots\ , \\ \lambda n.&\texttt{if } n = 0 \texttt{ then } \texttt{prf}\, H\ 0 \texttt{ else} \\ &\texttt{if } n = 1 \texttt{ then } \texttt{prf}\, H\ 1 \texttt{ else } \dots\ ) \end{aligned}$$

Lógica clásica vs tipos dependientes

- En lógica clásica :

$$H_0 := \mathtt{callcc}_{\alpha}(1, \phi(\mathtt{throw}_{\alpha}(2, p)) : \exists x P(x)$$

- Restricción a la elección numerable :

$$AC_{\mathbb{N}} : \forall x^{\mathbb{N}} \exists y^{B} P(x, y) \to \exists f^{\mathbb{N} \to B} \forall x^{\mathbb{N}} P(x, f(x))$$

- Prueba :

$$\begin{aligned} AC := \lambda H.\, & \mathtt{let}\, H_0 = H\ 0\, \mathtt{in} \\ & \mathtt{let}\, H_1 = H\ 1\, \mathtt{in} \\ & \ldots \\ & (\lambda n.\mathtt{if}\ n = 0\ \mathtt{then}\ \mathtt{wit}\ H_0\ \mathtt{else} \\ & \quad \mathtt{if}\ n = 1\ \mathtt{then}\ \mathtt{wit}\ H_1\ \mathtt{else}\ \ldots\ , \\ & \lambda n.\mathtt{if}\ n = 0\ \mathtt{then}\ \mathtt{prf}\ H_0\ \mathtt{else} \\ & \quad \mathtt{if}\ n = 1\ \mathtt{then}\ \mathtt{prf}\ H_1\ \mathtt{else}\ \ldots\ ) \end{aligned}$$

Lógica clásica vs tipos dependientes

- En lógica clásica :

$$H_0 := \texttt{callcc}_{\alpha}(1, \phi(\texttt{throw}_{\alpha}(2, p)) : \exists x P(x)$$

- Restricción a la elección numerable :

$$AC_{\mathbb{N}} : \forall x^{\mathbb{N}} \exists y^{B} P(x,y) \to \exists f^{\mathbb{N}\to B} \forall x^{\mathbb{N}} P(x, f(x))$$

- Prueba :

$$AC := \lambda H.\,\texttt{let}\, H_\infty = (H\ 0, H\ 1, \ldots, H\ n, \ldots)\,\texttt{in}$$
$$(\lambda n.\texttt{nth}\ n\ H_\infty, \lambda n.\texttt{nth}\ n\ H_\infty)$$

Lógica clásica vs tipos dependientes

- En lógica clásica :

$$H_0 := \texttt{callcc}_{\alpha}(1, \phi(\texttt{throw}_{\alpha}(2, p)) : \exists x P(x)$$

- Restricción a la elección numerable :

$$AC_{\mathbb{N}} : \forall x^{\mathbb{N}} \exists y^{B} P(x,y) \to \exists f^{\mathbb{N}\to B} \forall x^{\mathbb{N}} P(x, f(x))$$

- Prueba :

$$AC := \lambda H.\, \texttt{let}\, H_\infty = \texttt{cofix}^0_{fn}(H\ n, f(S(n)))\,\texttt{in}$$
$$(\lambda n.\texttt{nth}\ n\ H_\infty, \lambda n.\texttt{nth}\ n\ H_\infty)$$

dPA$^\omega$

Ref : Hugo Herbelin (LICS'12)

A Constructive Proof of Dependent Choice, Compatible with Classical Logic

Un sistema formal :

- **clásico** : $p, q ::= \dots \mid \texttt{catch}_\alpha\, p \mid \texttt{throw}_\alpha\, p$
- con **tipos dependientes** :
  - fórmulas : $A, B ::= \dots \mid [a : A] \to B \mid t = u,$
  - términos : $t, u ::= \dots \mid \texttt{wit}\, p,$
  - pruebas : $p, q ::= \dots \mid \texttt{prf}\, p$
- una **compartimentación** :

$$\frac{\Gamma \vdash p : \exists x^T A(x) \quad p \text{ Nef}}{\Gamma \vdash \texttt{prf}\, p : A(\texttt{wit}\, p)} \qquad \frac{\Gamma \vdash p : \exists x^T A(x) \quad \Gamma, x : T, a : A \vdash q : B}{\Gamma \vdash \texttt{dest}\, p\, \texttt{as}\, ((x, a))\, \texttt{in}\, q : B}$$

- call-by-value y **sharing** : $p, q ::= \dots \mid \texttt{let}\, a = q\, \texttt{in}\, p$
- con constructores inductivos y **coinductivos** :
  $p, q ::= \dots \mid \texttt{ind}\, t\, \texttt{of}\, [p \mid (x, a).q] \mid \texttt{cofix}^t_{bn}\, p$
- **pereza** para el $\texttt{cofix}$

Lenguaje

Términos

$t, u ::= x \mid 0 \mid S(t) \mid \texttt{rec}\ t\ \texttt{of}\ [t \mid (x,y).t] \mid \lambda x.t \mid t\ u \mid \texttt{wit}\ p$

Pruebas

$$
\begin{array}{rcl}
p, q & ::= & a \mid () \mid \iota_i(p) \mid (p_1, p_2) \mid (t, p) \mid \lambda a.p \mid \lambda x.p \\
 & \mid & \texttt{case}\ p\ \texttt{of}\ [a_1.p_1 \mid a_2.p_2] \\
 & \mid & \texttt{split}\ p\ \texttt{as}\ (a_1, a_2)\ \texttt{in}\ q \\
 & \mid & \texttt{dest}\ p\ \texttt{as}\ (x, a)\ \texttt{in}\ q \\
 & \mid & \texttt{let}\ a = q\ \texttt{in}\ p \\
 & \mid & p\ q \mid p\ t \mid \texttt{exfalso}\ p \\
 & \mid & \texttt{cofix}^t_{bx}\ p \mid \texttt{ind}\ t\ \texttt{of}\ [p_0 \mid (x,a).p_S] \\
 & \mid & \texttt{refl} \mid \texttt{subst}\ p\ q \mid \texttt{prf}\ p \\
 & \mid & \texttt{catch}_\alpha\ p \mid \texttt{throw}_\alpha\ p
\end{array}
$$

Reglas de reducción

Términos :

$$
\begin{array}{lcl}
\mathtt{wit}(t,p) & \triangleright & t \\
(\lambda x.t)u & \triangleright & t[u/x] \\
\mathtt{rec}\,0\,\mathtt{of}\ [t_0 \mid (x,y).t_S] & \triangleright & t_0 \\
\mathtt{rec}\,S(t)\,\mathtt{of}\ [t_0 \mid (x,y).t_S] & \triangleright & t_S[t/x][\mathtt{rec}\,t\,\mathtt{of}\ [t_0 \mid (x,y).t_S]/y]
\end{array}
$$

Reglas de reducción

Call-by-value :

$$
\begin{array}{lcl}
\texttt{let}\ a = \iota_i(p)\ \texttt{in}\ q & \triangleright & \texttt{let}\ b = p\ \texttt{in}\ q[\iota_i(b)/a] \\
\texttt{let}\ a = (p_1, p_2)\ \texttt{in}\ q & \triangleright & \texttt{let}\ a_1 = p_1\ \texttt{in}\ \texttt{let}\ a_2 = p_2\ \texttt{in}\ q[(a_1, a_2)/a] \\
\texttt{let}\ a = (t, p)\ \texttt{in}\ q & \triangleright & \texttt{let}\ b = p\ \texttt{in}\ q[(t, b)/a] \\
\texttt{let}\ a = \lambda x.p\ \texttt{in}\ q & \triangleright & q[\lambda x.p/a] \\
\texttt{let}\ a = \lambda b.p\ \texttt{in}\ q & \triangleright & q[\lambda b.p/a] \\
\texttt{let}\ a = ()\ \texttt{in}\ q & \triangleright & q[()/a] \\
\texttt{let}\ a = b\ \texttt{in}\ q & \triangleright & q[b/a]
\end{array}
$$

$$
\begin{array}{lcl}
\texttt{case}\ \iota_i(p)\ \texttt{of}\ [a_1.p_1 \mid a_2.p_2] & \triangleright & \texttt{let}\ a_i = p\ \texttt{in}\ p_i \\
\texttt{split}(p_1, p_2)\ \texttt{as}\ (a_1, a_2)\ \texttt{in}\ q & \triangleright & \texttt{let}\ a_1 = p_1\ \texttt{in}\ \texttt{let}\ a_2 = p_2\ \texttt{in}\ q \\
\texttt{dest}(t, p)\ \texttt{as}\ (x, a)\ \texttt{in}\ q & \triangleright & \texttt{let}\ a = p\ \texttt{in}\ q[t/x] \\
\texttt{prf}(t, p) & \triangleright & p \\
(\lambda a.p)q & \triangleright & \texttt{let}\ a = q\ \texttt{in}\ p \\
(\lambda x.p)t & \triangleright & p[t/x] \\
\texttt{subst refl}\ p & \triangleright & p \\
\texttt{ind}\ 0\ \texttt{of}\ [p_0 \mid (x, a).p_S] & \triangleright & p_0 \\
\texttt{ind}\ S(t)\ \texttt{of}\ [p_0 \mid (x, a).p_S] & \triangleright & p_S[t/x][\texttt{ind}\ t\ \texttt{of}\ [p_0 \mid (x, a).p_S]/a]
\end{array}
$$

Reglas de reducción

Lógica clásica :

$$
\begin{array}{lcl}
F[\mathtt{exfalso}\,p] & \rhd & \mathtt{exfalso}\,p \\
F[\mathtt{throw}_\alpha\,p] & \rhd & \mathtt{throw}_\alpha\,p \\
F[\mathtt{catch}_\alpha\,p] & \rhd & \mathtt{catch}_\alpha\,F[p[F/\alpha]] \\
\\
\mathtt{exfalso}\,\mathtt{exfalso}\,p & \rhd & \mathtt{exfalso}\,p \\
\mathtt{exfalso}\,\mathtt{throw}_\beta\,p & \rhd & \mathtt{throw}_\beta\,p \\
\mathtt{exfalso}\,\mathtt{catch}_\alpha\,p & \rhd & \mathtt{exfalso}\,p[\mathtt{exfalso}[]/\alpha] \\
\\
\mathtt{throw}_\beta\,\mathtt{exfalso}\,p & \rhd & \mathtt{exfalso}\,p \\
\mathtt{throw}_\beta\,\mathtt{throw}_\alpha\,p & \rhd & \mathtt{throw}_\alpha\,p \\
\mathtt{throw}_\beta\,\mathtt{catch}_\alpha\,p & \rhd & \mathtt{throw}_\beta\,p[\beta/\alpha] \\
\mathtt{catch}_\alpha\,\mathtt{throw}_\alpha\,p & \rhd & \mathtt{catch}_\alpha\,p \\
\mathtt{catch}_\beta\,\mathtt{catch}_\alpha\,p & \rhd & \mathtt{catch}_\beta\,p[\beta/\alpha]
\end{array}
$$

Reglas de reducción

Cofix :

- pereza :

$$F_c[] \ ::= \ \texttt{case}[]\,\texttt{of}\ [a_1.p_1 \mid a_2.p_2] \mid \texttt{split}[]\,\texttt{as}\ (a_1,a_2)\,\texttt{in}\ q \mid \texttt{dest}[]\,\texttt{as}\ (x,a)\,\texttt{in}\ q$$

$$F_c[\texttt{cofix}^t_{ax}p] \quad \triangleright \quad \texttt{let}\ c = \texttt{cofix}^t_{ax}p\ \texttt{in}\ F_c[c]$$

- desplegamiento :

$$\texttt{let}\ a = \texttt{cofix}^t_{x,b}\ p\ \texttt{in}\ D[F_c[a]] \quad \triangleright \texttt{let}\ a = p[\lambda y.\texttt{cofix}^y_{bx}p/b][t/x]\ \texttt{in}\ D[F_c[a]]$$

- control :

$$C[] ::= \texttt{exfalso}[] \mid \texttt{throw}_\beta[] \mid \texttt{catch}_\alpha[]$$

$$\texttt{let}\ a = \texttt{cofix}^t_{bx}p\ \texttt{in}\ C[q] \quad \triangleright \quad C[\texttt{let}\ a = \texttt{cofix}^t_{bx}p\ \texttt{in}\ q]$$

Reglas de reducción

Fórmulas :

$$
\begin{array}{lcl}
0 = 0 & \triangleright & \top \\
0 = S(u) & \triangleright & \bot \\
S(t) = 0 & \triangleright & \bot \\
S(t) = S(u) & \triangleright & t = u \\
\nu_{fx}^{t} A & \triangleright & A[t/x][\nu_{fx}^{y} A / f(x) = 0]
\end{array}
$$

Tipaje

$$\frac{\Gamma \vdash p : \exists x^T A(x) \quad p\mathbf{Nef}}{\Gamma \vdash \mathtt{prf}\, p : A(\mathtt{wit}\, p)}$$

$$\frac{\Gamma \vdash p : [a : A] \Rightarrow B \quad \Gamma \vdash q : A \quad a \notin FV(B) \text{ si } q \text{ no es } \mathbf{Nef}}{\Gamma \vdash p\, q : B[q/a]}$$

Negative-elimination-free

$$N, M ::= \ldots \mid \lambda a.p \mid (N, N) \mid \iota_i(N) \mid \mathtt{prf}\, N \mid \mathtt{let}\, a = M\, \mathtt{in}\, N \mid \ldots$$

$\looparrowright$ $\mathtt{catch}_\alpha, \mathtt{throw}_\alpha, \mathtt{cofix}^x_{bn}\, p, p\, q, p\, t$ sólo dentro de $\lambda a.p$ o $\lambda x.p$

Tipaje

$$\frac{(a:A)\in\Gamma}{\Gamma\vdash a:A}\ \textsc{axiom}\qquad \frac{\Gamma\vdash p:A\qquad A\equiv B}{\Gamma\vdash p:B}\ \textsc{conv}\qquad \frac{}{\Gamma\vdash ():\top}\ \top_I\qquad \frac{\Gamma\vdash p:\bot}{\Gamma\vdash \texttt{exfalso}\ p:C}\ \bot_E$$

$$\frac{\Gamma\vdash p_1:A_1\qquad \Gamma\vdash p_2:A_2}{\Gamma\vdash (p_1,p_2):A_1\wedge A_2}\ \wedge_I\qquad \frac{\Gamma\vdash p:A_i}{\Gamma\vdash \iota_i(p):A_1\vee A_2}\ \vee_I^i$$

$$\frac{\Gamma\vdash p:A_1\wedge A_2\qquad \Gamma,a_1:A_1,a_2:A_2\vdash q:B[(a_1,a_2)/a]\qquad a\notin FV(B)\text{ if } p \text{ not N-elimination-free}\qquad a_1,a_2\notin FV(B)}{\Gamma\vdash \texttt{split}\ p\ \texttt{as}\ (a_1,a_2)\ \texttt{in}\ q:B[p/a]}\ \wedge_E$$

$$\frac{p:A_1\vee A_2\quad \Gamma,a_1:A_1\vdash p_1:B[\iota_1(a_1)/a]\quad \Gamma,a_2:A_2\vdash p_2:B[\iota_2(a_2)/a]\quad a\notin FV(B)\text{ if } p \text{ not N-elimination-free}\quad a_1,a_2\notin FV(B)}{\Gamma\vdash \texttt{case}\ p\ \texttt{of}\ [a_1.p_1\,|\,a_2.p_2]:B[p/a]}\ \vee_E$$

$$\frac{\Gamma,a:A\vdash p:B}{\Gamma\vdash \lambda a.p:[a:A]\to B}\ \to_I\qquad \frac{\Gamma\vdash p:[a:A]\to B\quad \Gamma\vdash q:A\quad a\notin FV(B)\text{ if } q \text{ not N-elimination-free}}{\Gamma\vdash p\,q:B[q/a]}\ \to_E$$

$$\frac{\Gamma,\alpha:A^{\perp\!\!\!\perp}\vdash p:A}{\Gamma\vdash \texttt{catch}_\alpha\ p:A}\ \textsc{catch}\qquad \frac{\Gamma\vdash p:A\qquad (\alpha:A^{\perp\!\!\!\perp})\in\Gamma}{\Gamma\vdash \texttt{throw}_\alpha\ p:C}\ \textsc{throw}$$

Tipaje

$$\frac{\Gamma, x : T \vdash p : A}{\Gamma \vdash \lambda x.p : \forall x^T A}\ \forall_I \qquad \frac{\Gamma \vdash p : \forall x^T A \qquad \Gamma \vdash t : T}{\Gamma \vdash pt : A[t/x]}\ \forall_E$$

$$\frac{\Gamma \vdash p : A[t/x] \qquad \Gamma \vdash t : T}{\Gamma \vdash (t,p) : \exists x^T A}\ \exists_I \qquad \frac{\Gamma \vdash p : \exists x^T A \qquad \Gamma, x : T, a : A \vdash q : B}{\Gamma \vdash \mathtt{dest}\ p\ \mathtt{as}\ (x,a)\ \mathtt{in}\ q : B}\ \exists_E$$

$$\frac{\Gamma \vdash p : \exists x^T A \qquad p \text{ is N-elimination-free}}{\Gamma \vdash \mathtt{prf}\, p : A[\mathtt{wit}\, p/x]}\ \exists_E^{\text{PRF}}$$

$$\frac{\Gamma \vdash t : \mathbb{N}}{\Gamma \vdash \mathtt{refl} : t = t}\ \text{REFL} \qquad \frac{\Gamma \vdash p : t = u \qquad \Gamma \vdash q : A[t/x] \qquad x \notin Dom(\Gamma)}{\Gamma \vdash \mathtt{subst}\, p\, q : A[u/x]}\ \text{SUBST}$$

$$\frac{\Gamma \vdash t : \mathbb{N} \qquad \Gamma \vdash p : A[0/x] \qquad \Gamma, x : T, a : A \vdash q : A[S(x)/x]}{\Gamma \vdash \mathtt{ind}\ t\ \mathtt{of}\ [p \mid (x,a).q] : A[t/x]}\ \text{IND}$$

$$\frac{\Gamma \vdash p : A \qquad \Gamma, a : A \vdash q : B \qquad a \notin FV(B) \text{ if } p \text{ not N-elimination-free}}{\Gamma \vdash \mathtt{let}\ a = p\ \mathtt{in}\ q : B[p/a]}\ \text{CUT}$$

$$\frac{\Gamma \vdash t : T \qquad \Gamma, f : T \to \mathbb{N}, x : T, b : \forall y\, f(y) = 0 \vdash p : A \qquad f \text{ positive in } A}{\Gamma \vdash \mathtt{cofix}^t_{bx} p : \nu^t_{fx} A}\ \nu_I$$

Tipaje

$$\frac{(x:T)\in\Gamma}{\Gamma\vdash x:T}\qquad\frac{\Gamma,x:U\vdash t:T}{\Gamma\vdash\lambda x.t:U\to T}\qquad\frac{\Gamma\vdash t:U\to T\qquad\Gamma\vdash u:U}{\Gamma\vdash t\,u:T}\qquad\frac{}{\Gamma\vdash 0:\mathbb{N}}\qquad\frac{\Gamma\vdash t:\mathbb{N}}{\Gamma\vdash S(t):\mathbb{N}}$$

$$\frac{\Gamma\vdash t:\mathbb{N}\qquad\Gamma\vdash t_0:U\qquad\Gamma,x:\mathbb{N},y:U\vdash t_S:U}{\Gamma\vdash \mathtt{rec}\ t\ \mathtt{of}\ [t_0\,|\,(x,y).t_S]:U}\qquad\frac{\Gamma\vdash p:\exists x^T A\quad p\text{ is N-elimination-free}}{\Gamma\vdash \mathtt{wit}\,p:T}\,\exists_E^{\mathrm{WIT}}$$

Elección numerable

Elección numerable

$$
\begin{aligned}
AC_{\mathbb{N}} \quad &:= \quad ?? \\
&: \quad \forall n \exists y P(n,y) \to \exists f \forall n P(n, f(n))
\end{aligned}
$$

Elección numerable

Elección numerable

$$\begin{array}{rcl} AC_{\mathbb{N}} & := & \lambda H.\mathtt{let}\ a = \mathtt{cofix}^0_{bn}\ (Hn, b(S(n))) \\ & & \quad \mathtt{in}\ (\lambda n.\,\mathtt{wit}(\mathtt{nth}(n,a)), \lambda n.\,\mathtt{prf}(\mathtt{nth}(n,a))) \\ & : & \forall n \exists y P(n,y) \to \exists f \forall n P(n, f(n)) \end{array}$$

dónde

$$\mathtt{nth}(n,a) := \pi_1(\mathtt{ind}\ n\ \mathtt{of}\ [a \mid (x,c).\pi_2(c)])$$

Proof :

↬ pizarrón

Elección numerable

Elección numerable

$$\begin{array}{rcl} AC_{\mathbb{N}} & := & \lambda H.\mathtt{let}\ a = \mathtt{cofix}^0_{bn}\ (Hn, b(S(n))) \\ & & \qquad \mathtt{in}\ (\lambda n.\,\mathtt{wit}(\mathtt{nth}(n,a)), \lambda n.\,\mathtt{prf}(\mathtt{nth}(n,a))) \\ & : & \forall n \exists y P(n,y) \to \exists f \forall n P(n, f(n)) \end{array}$$

dónde

$$\mathtt{nth}(n,a) \ := \ \pi_1(\mathtt{ind}\ n\ \mathtt{of}\ [a \mid (x,c).\pi_2(c)])$$

Proof :

$\looparrowright$ pizarrón

Elección dependiente

Elección dependiente

$$\begin{aligned} DC \quad := \quad & ?? \\ : \quad & \forall x \exists y P(x,y) \\ & \to \forall x_0 \exists f (f(0) = x_0 \land \forall n P(f(n), f(S(n)))) \end{aligned}$$

Elección dependiente

Elección dependiente

$$
\begin{array}{rcl}
DC & := & \lambda H \lambda x_0.\mathtt{let}\, b = \mathtt{s}\, H\, x_0\, \mathtt{in} \\
 & & \qquad\quad (\lambda n.\, \mathtt{wit}(\mathtt{nth}(n, x_0, b)), \\
 & & \qquad\quad \mathtt{refl}, \lambda n.\pi_1(\mathtt{prf}(\mathtt{prf}(\mathtt{nth}(n, x_0, b))))) \\
 & : & \forall x \exists y P(x,y) \\
 & & \to \forall x_0 \exists f (f(0) = x_0 \wedge \forall n P(f(n), f(S(n))))
\end{array}
$$

dónde

$$
\begin{array}{rcl}
\mathtt{s}\, a\, x & := & \mathtt{cofix}^{x}_{bn}\ \mathtt{dest}\, b\, n\, \mathtt{as}\, (y,c)\, \mathtt{in}\, (y, (c, b\, y)) \\
\mathtt{nth}(n, x_0, b) & := & \mathtt{ind}\, n\, \mathtt{of} \\
 & & [b \mid (m,c).(\mathtt{wit}(\mathtt{prf}(c)), \pi_2(\mathtt{prf}(\mathtt{prf}(c))))]
\end{array}
$$

El drama

(Dónde se ve que en la vida nada es fácil)

Propiedades

Subject reduction

Si $\Gamma \vdash p : A$ y $p \triangleright q$, entonces $\Gamma \vdash q : A$.

Normalización

Si $\Gamma \vdash p : A$ entonces p se normaliza.

Correción

$$\nvdash_{dPA^\omega} \bot$$

Propiedades

Subject reduction

Si $\Gamma \vdash p : A$ y $p \triangleright q$, entonces $\Gamma \vdash q : A$.

Normalización

Si $\Gamma \vdash p : A$ entonces p se normaliza.

Correción

$$\nvdash_{dPA^\omega} \bot$$

Propiedades

Subject reduction

Si $\Gamma \vdash p : A$ y $p \triangleright q$, entonces $\Gamma \vdash q : A$.

Normalización

Si $\Gamma \vdash p : A$ entonces p se normaliza.

Correción

$$\nvdash_{dPA^\omega} \bot$$

Corrección

Conservatividad

Si A es $\to$-ν-$\mathtt{wit}$-$\forall$-free, y $\vdash_{dPA^\omega} p : A$, entonces existe V tq $\vdash_{HA^\omega} V : A$

Conservatividad (bis)

Si A es Σ^0_1, $\vdash_{dPA^\omega} A$ implica $\vdash_{HA^\omega} A$

Correción

$$\nvdash_{dPA^\omega} \bot$$

Corrección

Conservatividad

Si A es $\to$-ν-wit-$\forall$-free, y $\vdash_{dPA^\omega} p : A$, entonces existe V tq $\vdash_{HA^\omega} V : A$

Proof :

Sea $p : A$ un término de prueba cerrado. Por normalización, p se reduce en :

- V : $*$-freeness de A implica que V no contiene $\mathtt{catch}_\alpha, \mathtt{throw}_\alpha, \mathtt{prf}$
- $D[V]$ con $D[] := \mathtt{let}\, a = \mathtt{cofix}^t_{b,x}\ p\, \mathtt{in}\, D[]$ y $a \notin FV(V)$
- $\mathtt{catch}_\alpha\ D[V]$: necesariamente $\alpha \notin FV(V)$

Conservatividad (bis)

Si A es Σ^0_1, $\vdash_{dPA^\omega} A$ implica $\vdash_{HA^\omega} A$

Corrección

Conservatividad

Si A es $\to$-ν-wit-$\forall$-free, y $\vdash_{dPA^\omega} p : A$, entonces existe V tq $\vdash_{HA^\omega} V : A$

Conservatividad (bis)

Si A es Σ^0_1, $\vdash_{dPA^\omega} A$ implica $\vdash_{HA^\omega} A$

Proof :

Toda formula Σ^0_1 es equivalente a una formula $\to$-ν-wit-$\forall$-free.

Correción

$\nvdash_{dPA^\omega} \bot$

Corrección

Conservatividad

Si A es $\to$-ν-$\mathtt{wit}$-$\forall$-free, y $\vdash_{dPA^\omega} p : A$, entonces existe V tq $\vdash_{HA^\omega} V : A$

Conservatividad (bis)

Si A es Σ^0_1, $\vdash_{dPA^\omega} A$ implica $\vdash_{HA^\omega} A$

Correción

$$\nvdash_{dPA^\omega} \bot$$

Normalización

Normalización

Si $\Gamma \vdash_{dPA^\omega} p : A$ entonces p se normaliza.

Proof ?

- Si p no tiene $\texttt{cofix}_{x,b}^{t}\ q$, p termina
- Si p no tiene $\texttt{catch}_\alpha\ /\ \texttt{throw}_\alpha$, p termina
- Pero si tiene los dos...
  ↬ exploración infinita de una rama co-inductiva
  ↬ sub-secuencia sín backtrack
  ↬ y eso no puede ser !

Normalización

Normalización

Si $\Gamma \vdash_{dPA^\omega} p : A$ entonces p se normaliza.

Proof ?

- Si p no tiene $\texttt{cofix}^t_{x,b}\ q$, p termina
- Si p no tiene $\texttt{catch}_\alpha$ / $\texttt{throw}_\alpha$, p termina
- Pero si tiene los dos...
  ↬ exploración infinita de una rama co-inductiva
  ↬ sub-secuencia sín backtrack
  ↬ y eso no puede ser !

Normalización

Normalización

Si $\Gamma \vdash_{dPA^{\omega}} p : A$ entonces p se normaliza.

Proof ?

- Si p no tiene $\mathtt{cofix}^{t}_{x,b}\ q$, p termina
- Si p no tiene $\mathtt{catch}_{\alpha}$ / $\mathtt{throw}_{\alpha}$, p termina
- Pero si tiene los dos...
  ↬ exploración infinita de una rama co-inductiva
  ↬ sub-secuencia sín backtrack
  ↬ y eso no puede ser !

Normalización

Normalización

Si $\Gamma \vdash_{dPA^\omega} p : A$ entonces p se normaliza.

Proof ?

- Si p no tiene $\texttt{cofix}^t_{x,b}\ q$, p termina
- Si p no tiene $\texttt{catch}_\alpha$ / $\texttt{throw}_\alpha$, p termina
- Pero si tiene los dos...
  $\looparrowright$ exploración infinita de una rama co-inductiva
  $\looparrowright$ sub-secuencia sín backtrack
  $\looparrowright$ y eso no puede ser !

Normalización

Normalización

Si $\Gamma \vdash_{dPA^\omega} p : A$ entonces p se normaliza.

Proof ?

- Si p no tiene $\texttt{cofix}^t_{x,b}\ q$, p termina
- Si p no tiene $\texttt{catch}_\alpha$ / $\texttt{throw}_\alpha$, p termina
- Pero si tiene los dos...
  $\looparrowright$ exploración infinita de una rama co-inductiva
  $\looparrowright$ sub-secuencia sín backtrack
  $\looparrowright$ ~~y eso no puede ser~~ !

Propiedades

Subject reduction

Si $\Gamma \vdash p : A$ y $p \triangleright q$, entonces $\Gamma \vdash q : A$.

Normalización

Si $\Gamma \vdash p : A$ entonces p se normaliza.

Conservatividad

If A is $\rightarrow$-ν-$\mathtt{wit}$-$\forall$-free, and $\vdash_{dPA^\omega} p : A$, then $\vdash_{HA^\omega} p : A$

Corrección

$$\nvdash_{dPA^\omega} \bot$$

Propiedades

Subject reduction

Si $\Gamma \vdash p : A$ y $p \triangleright q$, entonces $\Gamma \vdash q : A$.

Normalización

Si $\Gamma \vdash p : A$ entonces p se normaliza.

Conservatividad

If A is $\rightarrow$-ν-$\texttt{wit}$-$\forall$-free, and $\vdash_{dPA^\omega} p : A$, then $\vdash_{HA^\omega} p : A$

Corrección

$$\nvdash_{dPA^\omega} \bot$$

Propiedades

Subject reduction

Si $\Gamma \vdash p : A$ y $p \triangleright q$, entonces $\Gamma \vdash q : A$.

Normalización

Si $\Gamma \vdash p : A$ entonces p se normaliza.

Conservatividad

If A is $\rightarrow$-ν-$\mathtt{wit}$-$\forall$-free, and $\vdash_{dPA^\omega} p : A$, then $\vdash_{HA^\omega} p : A$

Corrección

$$\nvdash_{dPA^\omega} \bot$$

Propiedades

Subject reduction

Si $\Gamma \vdash p : A$ y $p \triangleright q$, entonces $\Gamma \vdash q : A$.

Normalización

Si $\Gamma \vdash p : A$ entonces p se normaliza.

Conservatividad

If A is $\rightarrow$-ν-$\mathtt{wit}$-$\forall$-free, and $\vdash_{dPA^\omega} p : A$, then $\vdash_{HA^\omega} p : A$

Corrección

$$\nvdash_{dPA^\omega} \bot$$

Propiedades

Subject reduction

Si $\Gamma \vdash p : A$ y $p \triangleright q$, entonces $\Gamma \vdash q : A$.

Normalización

Si $\Gamma \vdash p : A$ entonces p se normaliza.

Conservatividad

If A is $\rightarrow$-ν-$\texttt{wit}$-$\forall$-free, and $\vdash_{dPA^\omega} p : A$, then $\vdash_{HA^\omega} p : A$

Corrección

$$\nvdash_{dPA^\omega} \bot$$

« Le plan d'attaque »

(Rumbo a las peripecias)

Continuación : aspecto computaciónal

Intuición

Continuación = reificación del futuro de la computación

Continuation-passing style (CPS)

Pasaje de una continuación a cada étapa del cálculo.

- Técnica muy utilizada en compilación
- Explicitación del control
  $\looparrowright$ ejemplo : $(a + b) + c$
- Muy útil para definir los operadores de control como cc !

Continuación : aspecto computaciónal

Intuición

Continuación = reificación del futuro de la computación

Continuation-passing style (CPS)

Pasaje de una continuación a cada étapa del cálculo.

- Técnica muy utilizada en compilación
- Explicitación del control
  $\looparrowright$ ejemplo : (a + b) + c
- Muy útil para definir los operadores de control como $\mathfrak{cc}$!

Continuación : aspecto computaciónal

Intuición

Continuación = reificación del futuro de la computación

Continuation-passing style (CPS)

Pasaje de una continuación a cada étapa del cálculo.

- Técnica muy utilizada en compilación
- Explicitación del **control**
  $\looparrowright$ ejemplo : $(a + b) + c$
- Muy útil para definir los operadores de control como $\mathbf{cc}$!

Continuación : aspecto lógico

Lógica clásica :

- lógica intuicionista + axioma clásica
- Encaje en lógica intuicionista :
  ↬ traducción negativa :

$$A \mapsto (A \to \bot) \to \bot$$

Cálculo clásico :

- cálculo intuicionista + manipulación explicita de continuaciones
- "compilación" en un cálculo intuicionista :
  ↬ traducción :

$$t : A \mapsto [\![t]\!] : (A \to \bot) \to \bot$$

Continuación : aspecto lógico

Lógica clásica :

- lógica intuicionista + axioma clásica
- Encaje en lógica intuicionista :
  $\looparrowright$ traducción negativa :

$$A \mapsto (A \to \bot) \to \bot$$

Cálculo clásico :

- cálculo intuicionista + manipulación explicita de continuaciones
- "compilación" en un cálculo intuicionista :
  $\looparrowright$ traducción :

$$t : A \mapsto [\![t]\!] : (A \to \bot) \to \bot$$

Continuación : aspecto lógico

Lógica clásica :

- lógica intuicionista + axioma clásica
- Encaje en lógica intuicionista :
  $\looparrowright$ traducción negativa :

$$A \mapsto (A \to \bot) \to \bot$$

Cálculo clásico :

- cálculo intuicionista + manipulación explicita de continuaciones
- "compilación" en un cálculo intuicionista :
  $\looparrowright$ traducción **CPS** :

$$t : A \mapsto [\![t]\!] : \color{red}{(A \to \bot)} \to \bot$$

Normalización por CPS

- **Lenguaje 1 :** λ-coso, reducción, tipaje
- **Lenguaje 2 :** λ-cosa, reducción, tipaje, normalización

Normalización

$$\Gamma \vdash_1 p : A \qquad \Rightarrow \qquad p \text{ normaliza.}$$

Proof :

1. Si $\Gamma \vdash_1 p : A$ entonces $[\![\Gamma]\!] \vdash_2 [\![p]\!] : [\![A]\!]$
2. Si $p \rightarrow_1 q$ entonces $[\![p]\!] \xrightarrow{+}_2 [\![q]\!]$

$\hookrightarrow$ si p no normalizaba, tampoco normalizaría $[\![p]\!]$, absurdo !

Normalización por CPS

- **Lenguaje 1 :** λ-coso, reducción, tipaje
- **Lenguaje 2 :** λ-cosa, reducción, tipaje, normalización

Normalización

$$\Gamma \vdash_1 p : A \qquad \Rightarrow \qquad p \text{ normaliza.}$$

Proof :

1. Si $\quad \Gamma \vdash_1 p : A \quad$ entonces $\quad [\![\Gamma]\!] \vdash_2 [\![p]\!] : [\![A]\!]$
2. Si $\quad p \rightarrow_1 q \quad$ entonces $\quad [\![p]\!] \xrightarrow{+}_2 [\![q]\!]$

$\hookrightarrow$ si p no normalizaba, tampoco normalizaría $[\![p]\!]$, absurdo !

Normalización por CPS

- **Lenguaje 1 :** λ-coso, reducción, tipaje
- **Lenguaje 2 :** λ-cosa, reducción, tipaje, normalización

Normalización

$$\Gamma \vdash_1 p : A \qquad \Rightarrow \qquad p \text{ normaliza.}$$

Proof :

1. Si $\Gamma \vdash_1 p : A$ entonces $[\![\Gamma]\!] \vdash_2 [\![p]\!] : [\![A]\!]$
2. Si $p \rightarrow_1 q$ entonces $[\![p]\!] \xrightarrow{+}_2 [\![q]\!]$

$\looparrowright$ si p no normalizaba, tampoco normalizaría $[\![p]\!]$, absurdo !

Normalización por CPS

- **Lenguaje 1 :** λ-coso, reducción, tipaje
- **Lenguaje 2 :** λ-cosa, reducción, tipaje, normalización

Normalización

$$\Gamma \vdash_1 p : A \qquad \Rightarrow \qquad p \text{ normaliza.}$$

Proof :

1. Si $\Gamma \vdash_1 p : A$ entonces $[\![\Gamma]\!] \vdash_2 [\![p]\!] : [\![A]\!]$
2. Si $p \rightarrow_1 q$ entonces $[\![p]\!] \xrightarrow{+}_2 [\![q]\!]$

$\looparrowright$ si p no normalizaba, tampoco normalizaría $[\![p]\!]$, absurdo !

Normalización por CPS

Interes :

- método clásico, entre otras cosas para cálculo clásico
- permite un entendimiento profundo
- traducción de tipos !
- sin meta-uso del axioma de elección ?

Inconvenientes :

- muy técnico
- nuestro lenguaje es poco estándar

Dificultad :

$$\texttt{let}\ a = \texttt{cofix}^t_{bx}\, p\ \texttt{in}\ C[q] \quad \triangleright \quad C[\texttt{let}\ a = \texttt{cofix}^t_{bx}\, p\ \texttt{in}\ q]$$

Normalización por CPS

Interes :

- método clásico, entre otras cosas para cálculo clásico
- permite un entendimiento profundo
- traducción de tipos !
- sin meta-uso del axioma de elección ?

Inconvenientes :

- muy técnico
- nuestro lenguaje es poco estándar

Dificultad :

$$\mathtt{let}\ a = \mathtt{cofix}^t_{bx}\, p\ \mathtt{in}\ C[q] \quad \triangleright \quad C[\mathtt{let}\ a = \mathtt{cofix}^t_{bx}\, p\ \mathtt{in}\ q]$$

Normalización por CPS

Interes :

- método clásico, entre otras cosas para cálculo clásico
- permite un entendimiento profundo
- traducción de tipos !
- sin meta-uso del axioma de elección ?

Inconvenientes :

- muy técnico
- nuestro lenguaje es poco estándar

Dificultad :

$$\mathtt{let}\ a = \mathtt{cofix}^t_{bx} p\ \mathtt{in}\ C[q] \quad \triangleright \quad C[\mathtt{let}\ a = \mathtt{cofix}^t_{bx} p\ \mathtt{in}\ q]$$

Semantic artefacts

CPS : Ariola et al (FLOPS'12)

Classical Call-by-Need Sequent Calculi : The Unity of Semantic Artifacts

Semántica operacional :

- Cálculo big-step : *variante del $\bar{\lambda}\mu\tilde{\mu}$-cálculo ?*
- Definición small-step
  $\looparrowright$ reducción *context-free*
- Traducción CPS : *variante del λ-cálculo ?*
- Corrección computacional

Normalización :

- Tipaje de la traducción : *sistema F ?*
- Happy end ?

Semantic artefacts

CPS : Ariola et al (FLOPS'12)

Classical Call-by-Need Sequent Calculi : The Unity of Semantic Artifacts

Semántica operacional :

- Cálculo big-step : *variante del $\bar{\lambda}\mu\tilde{\mu}$-cálculo ?*
- Definición small-step
  ↬ reducción *context-free*
- Traducción CPS : *variante del λ-cálculo ?*
- Corrección computacional

Normalización :

- Tipaje de la traducción : *sistema F ?*
- Happy end ?

Semantic artefacts

CPS : Ariola et al (FLOPS'12)

Classical Call-by-Need Sequent Calculi : The Unity of Semantic Artifacts

Semántica operacional :

- Cálculo big-step : *variante del* $\bar{\lambda}\mu\tilde{\mu}$*-cálculo ?*
- Definición small-step
  $\looparrowright$ reducción *context-free*
- Traducción CPS : *variante del* λ*-cálculo ?*
- Corrección computacional

Normalización :

- Tipaje de la traducción : *sistema F ?*
- Happy end ?

Semantic artefacts

CPS : Ariola et al (FLOPS'12)

Classical Call-by-Need Sequent Calculi : The Unity of Semantic Artifacts

Semántica operacional :

- Cálculo big-step : *variante del $\bar{\lambda}\mu\tilde{\mu}$-cálculo ?*
- Definición small-step
  $\looparrowright$ reducción *context-free*
- Traducción CPS : *variante del λ-cálculo ?*
- Corrección computacional

Normalización :

- Tipaje de la traducción : *sistema F ?*
- Happy end ?

Semantic artefacts

CPS : Ariola et al (FLOPS'12)

Classical Call-by-Need Sequent Calculi : The Unity of Semantic Artifacts

Semántica operacional :

- Cálculo big-step : *variante del $\bar{\lambda}\mu\tilde{\mu}$-cálculo ?*
- Definición small-step
  $\looparrowright$ reducción *context-free*
- Traducción CPS : *variante del λ-cálculo ?*
- Corrección computacional

Normalización :

- Tipaje de la traducción : *sistema F ?*
- Happy end ?

Un *avant-goût*

Pizarrón : normalización por CPS del λ_c-cálculo

Parentesis

Call-by-name :

$$(\lambda a.p) \star (\text{cc}\, q) \cdot \pi \quad \to \quad p[\text{cc}\, q/a] \star \pi$$

- CPS :

$$\begin{array}{rcl} [\![p \star \pi]\!] & := & [\![p]\!]\,[\![\pi]\!] \\ [\![\lambda a.p]\!]\,k & := & k\,(\lambda a.[\![p]\!]) \\ [\![q \cdot \pi]\!]\,p & := & p\,[\![q]\!]\,[\![\pi]\!] \end{array}$$

$$[\![A \to B]\!] \;:=\; \neg\neg(\neg\neg[\![A]\!] \to \neg\neg[\![B]\!])$$

Call-by-value :

$$\lambda a.p \star \text{cc}\, q \cdot e \quad \to \quad \text{cc}\, q \star [\lambda a.p] \cdot e \quad \stackrel{?}{\to} \quad V \star \lambda a.p \cdot \pi \quad \to \quad p[V/a] \star \pi$$

- CPS :

$$\begin{array}{rcl} [\![p \star \pi]\!] & := & [\![p]\!]\,[\![\pi]\!] \\ [\![\lambda a.p]\!]\,k & := & k\,(\lambda a.[\![p]\!]) \\ [\![q \cdot \pi]\!]\,p & := & [\![q]\!](\lambda u.p\,u\,[\![\pi]\!]) \end{array}$$

$$[\![A \to B]\!] \;:=\; \neg\neg([\![A]\!] \to \neg\neg[\![B]\!])$$

Parentesis

Call-by-name :

$$(\lambda a.p) \star (\mathbf{œ} q) \cdot \pi \quad \to \quad p[\mathbf{œ} q/a] \star \pi$$

- CPS :

$$\begin{array}{rcl} [\![p \star \pi]\!] & := & [\![p]\!]\, [\![\pi]\!] \\ [\![\lambda a.p]\!]\, k & := & k\,(\lambda a.[\![p]\!]) \\ [\![q \cdot \pi]\!]\, p & := & p\, [\![q]\!]\, [\![\pi]\!] \end{array}$$

$$[\![A \to B]\!] \;:=\; \neg\neg(\neg\neg[\![A]\!] \to \neg\neg[\![B]\!])$$

Call-by-value :

$$\lambda a.p \star \mathbf{œ} q \cdot e \quad \to \quad \mathbf{œ} q \star [\lambda a.p] \cdot e \quad \stackrel{?}{\to} \quad V \star \lambda a.p \cdot \pi \quad \to \quad p[V/a] \star \pi$$

- CPS :

$$\begin{array}{rcl} [\![p \star \pi]\!] & := & [\![p]\!]\, [\![\pi]\!] \\ [\![\lambda a.p]\!]\, k & := & k\,(\lambda a.[\![p]\!]) \\ [\![q \cdot \pi]\!]\, p & := & [\![q]\!](\lambda u.p\, u\, [\![\pi]\!]) \end{array}$$

$$[\![A \to B]\!] \;:=\; \neg\neg([\![A]\!] \to \neg\neg[\![B]\!])$$

En el próximo (y último capítulo)...

$\bar{\lambda}\mu\tilde{\mu}$: un cálculo de secuentes

(Nuestra arma principal y su uso)