Igualando proposiciones/tipos isomorfos en las lógicas/lenguajes del cubo lambda

Cristian Sottile

Doctorando UBA/UNQ

Jano

Director

Fidel

Codirector

1er Encuentro de Estudiantes de Doctorado de CPI

24 de junio de 2022

Cálculo lambda

Cálculo lambda

- Términos y tipos

$$t, s, \dots \quad ::= \quad x^A \mid \lambda x^A.\, t \mid ts \qquad\qquad A, B, \dots \quad ::= \quad \tau \mid A \to B$$

Cálculo lambda

- Términos y tipos

$$t, s, \ldots \quad ::= \quad x^A \mid \lambda x^A.\, t \mid ts \qquad\qquad A, B, \ldots \quad ::= \quad \tau \mid A \to B$$

- Sistema de tipos (consideramos contextos $\Gamma = \{x : A, y : B, \ldots\}$)

$$\frac{}{\Gamma, x : A \vdash x : A}(var) \qquad \frac{\Gamma, x : A \vdash r : B}{\Gamma \vdash \lambda x^A.\, r : A \to B}(abs) \qquad \frac{\Gamma \vdash r : A \to B \qquad \Gamma \vdash s : A}{\Gamma \vdash rs : B}(app)$$

Cálculo lambda

- Términos y tipos

$$t, s, \dots \quad ::= \quad x^A \mid \lambda x^A.\, t \mid ts \qquad\qquad A, B, \dots \quad ::= \quad \tau \mid A \to B$$

- Sistema de tipos (consideramos contextos $\Gamma = \{x : A, y : B, \dots\}$)

$$\frac{}{\Gamma, x : A \vdash x : A}(var) \qquad \frac{\Gamma, x : A \vdash r : B}{\Gamma \vdash \lambda x^A.\, r : A \to B}(abs) \qquad \frac{\Gamma \vdash r : A \to B \qquad \Gamma \vdash s : A}{\Gamma \vdash rs : B}(app)$$

- Ejemplos de tipado

$$\frac{\dfrac{}{x : A \vdash x : A}\ (var)}{\vdash \lambda x^A.\, x : A \to A}\ (abs)$$

Cálculo lambda

- Términos y tipos

$$t, s, \ldots \quad ::= \quad x^A \mid \lambda x^A.\, t \mid ts \qquad\qquad A, B, \ldots \quad ::= \quad \tau \mid A \to B$$

- Sistema de tipos (consideramos contextos $\Gamma = \{x : A, y : B, \ldots\}$)

$$\frac{}{\Gamma, x : A \vdash x : A}(var) \qquad \frac{\Gamma, x : A \vdash r : B}{\Gamma \vdash \lambda x^A.\, r : A \to B}(abs) \qquad \frac{\Gamma \vdash r : A \to B \qquad \Gamma \vdash s : A}{\Gamma \vdash rs : B}(app)$$

- Ejemplos de tipado

$$\dfrac{\dfrac{}{x : A \vdash x : A}(var)}{\vdash \lambda x^A.\, x : A \to A}(abs) \qquad \dfrac{\dfrac{\dfrac{\dfrac{}{f : A \to B, x : A \vdash f : A \to B}(var) \quad \dfrac{}{f : A \to B, x : A \vdash x : A}(var)}{f : A \to B, x : A \vdash fx : B}(app)}{x : A \vdash \lambda x^A.\, fx : A \to B}(abs)}{\vdash \lambda f^{A \to B}.\, \lambda x^A.\, fx : A \to B}(abs)$$

- Computación $\qquad (\lambda x.\, r)s \quad \to \quad [x := s]r \qquad (\beta\text{-conversión})$

$$\begin{aligned} \underline{(\lambda f^{A \to B}.\, \lambda x^A.\, fx)(\lambda x^A.\, x)}t \quad &\to \quad [f := \lambda x^A.\, x](\lambda x^A.\, fx)t \quad = \quad \underline{(\lambda x^A.\, (\lambda x^A.\, x)x)t} \\ &\to \quad [x := t]((\lambda x^A.\, x)x) \quad = \quad \underline{(\lambda x^A.\, x)t} \quad \to \quad t \end{aligned}$$

Correspondencia de Curry-Howard

Correspondencia de Curry-Howard

tipos = proposiciones

$$A, B, \ldots \quad ::= \quad \tau \mid A \to B \qquad\qquad A, B, \ldots \quad ::= \quad p \mid A \Rightarrow B$$

Correspondencia de Curry-Howard

tipos = proposiciones

$$A, B, \dots \quad ::= \quad \tau \mid A \rightarrow B \qquad\qquad A, B, \dots \quad ::= \quad p \mid A \Rightarrow B$$

Reglas de inferencia (consideramos contextos $\Gamma = \{A, B, \dots\}$)

$$\frac{}{\Gamma, A \vdash A}(ax) \qquad \frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B}(\Rightarrow_i) \qquad \frac{\Gamma \vdash A \rightarrow B \qquad \Gamma \vdash A}{\Gamma \vdash B}(\Rightarrow_e)$$

Correspondencia de Curry-Howard

tipos = proposiciones

$$A, B, \ldots \quad ::= \quad \tau \mid A \to B \qquad\qquad A, B, \ldots \quad ::= \quad p \mid A \Rightarrow B$$

Reglas de inferencia (consideramos contextos $\Gamma = \{A, B, \ldots\}$)

$$\frac{}{\Gamma, A \vdash A}(ax) \qquad \frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B}(\Rightarrow_i) \qquad \frac{\Gamma \vdash A \to B \qquad \Gamma \vdash A}{\Gamma \vdash B}(\Rightarrow_e)$$

programas = pruebas

$$\dfrac{\dfrac{}{x : A \vdash x : A}\ (ax)}{\vdash \lambda x^A.\ x : A \Rightarrow A}\ (\Rightarrow_i)$$

$$\dfrac{\dfrac{\dfrac{\dfrac{}{f : A \Rightarrow B, x : A \vdash f : A \Rightarrow B}\ (ax) \qquad \dfrac{}{f : A \Rightarrow B, x : A \vdash x : A}\ (ax)}{f : A \Rightarrow B, x : A \vdash fx : B}\ (\Rightarrow_e)}{x : A \vdash \lambda x^A.\ fx : A \Rightarrow B}\ (\Rightarrow_i)}{\vdash \lambda f^{A \Rightarrow B}.\ \lambda x^A.\ fx : A \Rightarrow B}\ (\Rightarrow_i)$$

Correspondencia de Curry-Howard

tipos = proposiciones

$$A, B, \ldots \quad ::= \quad \tau \mid A \to B \qquad\qquad A, B, \ldots \quad ::= \quad p \mid A \Rightarrow B$$

Reglas de inferencia (consideramos contextos $\Gamma = \{A, B, \ldots\}$)

$$\frac{}{\Gamma, A \vdash A}(ax) \qquad \frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B}(\Rightarrow_i) \qquad \frac{\Gamma \vdash A \to B \qquad \Gamma \vdash A}{\Gamma \vdash B}(\Rightarrow_e)$$

programas = pruebas

$$\dfrac{\dfrac{}{x : A \vdash x : A}\,(ax)}{\vdash \lambda x^A.\, x : A \Rightarrow A}\,(\Rightarrow_i)$$

$$\dfrac{\dfrac{\dfrac{\dfrac{}{f : A \Rightarrow B, x : A \vdash f : A \Rightarrow B}\,(ax) \qquad \dfrac{}{f : A \Rightarrow B, x : A \vdash x : A}\,(ax)}{f : A \Rightarrow B, x : A \vdash fx : B}\,(\Rightarrow_e)}{x : A \vdash \lambda x^A.\, fx : A \Rightarrow B}\,(\Rightarrow_i)}{\vdash \lambda f^{A\Rightarrow B}.\, \lambda x^A.\, fx : A \Rightarrow B}\,(\Rightarrow_i)$$

computación = simplificación

$$\dfrac{\dfrac{\Gamma, x : A \vdash r : B}{\Gamma \vdash \lambda x^A.\, r : A \to B}\,(abs) \qquad \Gamma \vdash s : A}{\Gamma \vdash (\lambda x^A.\, r)s : B}\,(app) \qquad \Longrightarrow \qquad \Gamma \vdash [x := s]r : B$$

Correspondencia de Curry-Howard

tipos = proposiciones

$$A, B, \ldots \quad ::= \quad \tau \mid A \to B \qquad\qquad A, B, \ldots \quad ::= \quad p \mid A \Rightarrow B$$

Reglas de inferencia (consideramos contextos $\Gamma = \{A, B, \ldots\}$)

$$\frac{}{\Gamma, A \vdash A}(ax) \qquad \frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B}(\Rightarrow_i) \qquad \frac{\Gamma \vdash A \to B \qquad \Gamma \vdash A}{\Gamma \vdash B}(\Rightarrow_e)$$

programas = pruebas

$$\dfrac{\dfrac{}{x : A \vdash x : A}\,(ax)}{\vdash \lambda x^A.\, x : A \Rightarrow A}\,(\Rightarrow_i)$$

$$\dfrac{\dfrac{\dfrac{\dfrac{}{f : A \Rightarrow B, x : A \vdash f : A \Rightarrow B}\,(ax) \qquad \dfrac{}{f : A \Rightarrow B, x : A \vdash x : A}\,(ax)}{f : A \Rightarrow B, x : A \vdash fx : B}\,(\Rightarrow_e)}{x : A \vdash \lambda x^A.\, fx : A \Rightarrow B}\,(\Rightarrow_i)}{\vdash \lambda f^{A \Rightarrow B}.\, \lambda x^A.\, fx : A \Rightarrow B}\,(\Rightarrow_i)$$

El cálculo lambda es un lenguaje de la lógica

Consecuencias

- Sistemas basados en cálculo lambda constituyen sistemas de pruebas
- Estudio de la lógica desde la computación
- Estudio de la computación desde la lógica

Cubo lambda

$$\begin{tikzcd}
 & \lambda\omega \arrow[rr] & & \lambda C \\
\lambda 2 \arrow[ur] \arrow[rr] & & \lambda\Pi 2 \arrow[ur] & \\
 & \lambda\underline{\omega} \arrow[uu] \arrow[rr] & & \lambda\Pi\underline{\omega} \arrow[uu] \\
\lambda^{\to} \arrow[uu] \arrow[ur] \arrow[rr] & & \lambda\Pi \arrow[uu] \arrow[ur] &
\end{tikzcd}$$

$(\lambda^{\to})$ términos dependen de términos

$(\uparrow \lambda 2)$ términos dependen de tipos

$(\nearrow \lambda\underline{\omega})$ tipos dependen de tipos

$(\to \lambda\Pi)$ tipos dependen de términos

(λC) todas las anteriores

Teoría de proposiciones/tipos isomorfos

Dos proposiciones/tipos A y B son isomorfos si y solo si existen

$$f : A \Rightarrow B \qquad g : B \Rightarrow A$$

tales que

$$g \circ f = id_A \qquad f \circ g = id_B$$

Teoría de proposiciones/tipos isomorfos

Dos proposiciones/tipos A y B son isomorfos si y solo si existen

$$f : A \Rightarrow B \qquad g : B \Rightarrow A$$

tales que

$$g \circ f = id_A \qquad f \circ g = id_B$$

$$\begin{tikzcd}
A \arrow[loop left, "id_A"] \arrow[r, bend left, "f"] & B \arrow[loop right, "id_B"] \arrow[l, bend left, "g"]
\end{tikzcd}$$

Teoría de proposiciones/tipos isomorfos

Dos proposiciones/tipos A y B son isomorfos si y solo si existen

$$f : A \Rightarrow B \qquad g : B \Rightarrow A$$

tales que

$$g \circ f = id_A \qquad f \circ g = id_B$$

$$\begin{tikzcd}
A \arrow[loop left, "id_A"] \arrow[r, bend left, "f"] & B \arrow[loop right, "id_B"] \arrow[l, bend left, "g"]
\end{tikzcd}$$

Ejemplo: $\qquad (A \wedge B) \Rightarrow C \equiv A \Rightarrow B \Rightarrow C \qquad$ (currificación)

Teoría de proposiciones/tipos isomorfos

Dos proposiciones/tipos A y B son isomorfos si y solo si existen

$$f : A \Rightarrow B \qquad g : B \Rightarrow A$$

tales que

$$g \circ f = id_A \qquad f \circ g = id_B$$

$$\begin{tikzcd} A \arrow[r, bend left, "f"] \arrow[loop left, "id_A"] & B \arrow[l, bend left, "g"] \arrow[loop right, "id_B"] \end{tikzcd}$$

Ejemplo: $\qquad (A \wedge B) \Rightarrow C \equiv A \Rightarrow B \Rightarrow C \qquad$ (currificación)

$$curry = \lambda x^{(A \wedge B) \Rightarrow C}.\lambda y^A.\lambda z^B.x\langle y, z\rangle : ((A \wedge B) \Rightarrow C) \Rightarrow (A \Rightarrow B \Rightarrow C)$$

$$uncurry = \lambda x^{A \Rightarrow B \Rightarrow C}.\lambda y^{A \wedge B}.x(\pi_1 y)(\pi_2 y) : (A \Rightarrow B \Rightarrow C) \Rightarrow ((A \wedge B) \Rightarrow C)$$

Teoría de proposiciones/tipos isomorfos

Dos proposiciones/tipos A y B son isomorfos si y solo si existen

$$f : A \Rightarrow B \qquad g : B \Rightarrow A$$

tales que

$$g \circ f = id_A \qquad f \circ g = id_B$$

$$\begin{tikzcd}
A \arrow[r, bend left, "f"] \arrow[loop left, "id_A"] & B \arrow[l, bend left, "g"] \arrow[loop right, "id_B"]
\end{tikzcd}$$

Ejemplo: $\qquad (A \wedge B) \Rightarrow C \equiv A \Rightarrow B \Rightarrow C \qquad$ (currificación)

$$curry = \lambda x^{(A \wedge B) \Rightarrow C}.\lambda y^{A}.\lambda z^{B}.x\langle y, z\rangle : ((A \wedge B) \Rightarrow C) \Rightarrow (A \Rightarrow B \Rightarrow C)$$

$$uncurry = \lambda x^{A \Rightarrow B \Rightarrow C}.\lambda y^{A \wedge B}.x(\pi_1 y)(\pi_2 y) : (A \Rightarrow B \Rightarrow C) \Rightarrow ((A \wedge B) \Rightarrow C)$$

Se puede probar

$$uncurry \circ curry = id_{(A \wedge B) \Rightarrow C} \qquad curry \circ uncurry = id_{A \Rightarrow B \Rightarrow C}$$

Internalización de isomorfismos

- Incompatibilidad en sistemas tradicionales
  - Una prueba p de $A \wedge B$ no prueba $B \wedge A$
  - Si expreso el lema de asociatividad de la suma como $n + (m + p) = (n + m) + p$, para probar que $(a + b) + c = a + (b + c)$ debo aplicar antes la simetría
  - La función $suma : (int, int) \rightarrow int$ no funciona si le paso dos argumentos $suma\ 1\ 2$
- Automatización
  - ¿Podemos?
  - ¿Tiene sentido?
- Con prueba de A se puede probar B
- Adaptación automática: prueba de A prueba B
- e.g. $A \wedge B$ prueba $B \wedge A$

Sistema I Polimórfico

Definiciones

- Extensión de Sistema F internalizando la teoría de tipos isomorfos
- Estáticamente: ampliación de combinaciones permitidas

$$
\begin{array}{rlll}
f & \stackrel{\text{def}}{=} \lambda p.\ \mathit{fst}\ p \star \mathit{snd}\ p & f\langle r, s\rangle & f\ r\ s \\
f' & \stackrel{\text{def}}{=} \lambda x.\ \lambda y.\ x \star y & f'\ r\ s & f'\langle r, s\rangle
\end{array}
$$

- Dinámicamente: adaptación implícita de programas y contextos

$$
\begin{array}{lllll}
f\langle r, s\rangle & & & \longrightarrow & r \star s \\
f\ r\ s & \longrightarrow & f'\ r\ s & \longrightarrow & r \star s \\
f\ r\ s & \longrightarrow & f\langle r, s\rangle & \longrightarrow & r \star s \\
f'\langle r, s\rangle & \longrightarrow & f\langle r, s\rangle & \longrightarrow & r \star s \\
f'\langle r, s\rangle & \longrightarrow & f'\ r\ s & \longrightarrow & r \star s \\
f'\ r\ s & & & \longrightarrow & r \star s
\end{array}
$$

Sistema I Polimórfico

Definiciones

- Extensión de Sistema F internalizando la teoría de tipos isomorfos
- Estáticamente: ampliación de combinaciones permitidas

$$
\begin{array}{rlll}
f & \stackrel{\text{def}}{=} \lambda p.\ \mathit{fst}\ p \star \mathit{snd}\ p & f\,\langle r, s\rangle & f\ r\ s \\
f' & \stackrel{\text{def}}{=} \lambda x.\ \lambda y.\ x \star y & f'\ r\ s & f'\langle r, s\rangle
\end{array}
$$

- Dinámicamente: adaptación implícita de programas y contextos

$$
\begin{array}{ccccc}
C_1[p_1] & & & \longrightarrow & r \\
C_2[p_1] & \longrightarrow & C_2[p_2] & \longrightarrow & r \\
C_2[p_1] & \longrightarrow & C_1[p_1] & \longrightarrow & r \\
C_1[p_2] & \longrightarrow & C_1[p_1] & \longrightarrow & r \\
C_1[p_2] & \longrightarrow & C_2[p_2] & \longrightarrow & r \\
C_2[p_2] & & & \longrightarrow & r
\end{array}
$$

Sistema I Polimórfico

Definiciones

- Extensión de Sistema F internalizando la teoría de tipos isomorfos
- Estáticamente: ampliación de combinaciones permitidas

$$\begin{array}{rcll} A \times B & \equiv & B \times A & \\ (A \times B) \to C & \equiv & A \to B \to C & \\ A \to (B \times C) & \equiv & (A \to B) \times (A \to C) & \\ \forall X.(A \to B) & \equiv & A \to \forall X.B & {\scriptstyle X \notin FV(A)} \end{array}$$

$$\frac{A \equiv B \qquad p : A}{p : B}(\equiv)$$

- Dinámicamente: adaptación implícita de programas y contextos

$$\begin{array}{rcll} \langle r, s \rangle & \rightleftarrows & \langle s, r \rangle & \\ r\langle s, t \rangle & \rightleftarrows & rst & \\ \lambda x^A.\ \langle r, s \rangle & \rightleftarrows & \langle \lambda x^A.\ r, \lambda x^A.\ s \rangle & \\ \Lambda X.\lambda x^A.r & \rightleftarrows & \lambda x^A.\Lambda X.r & {\scriptstyle \text{si } X \notin FV(A)} \end{array}$$

$$\frac{s : A}{(\lambda x^A.\ r)s \quad \hookrightarrow \quad [x := s]r}(\beta)$$

$$\to \quad := \quad \rightleftarrows^* \ \cdot \ \hookrightarrow \ \cdot \ \rightleftarrows^*$$

Sistema I Polimórfico

Ejemplo

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\vdash \lambda f^{A\to B}.\ \lambda x^{A}.\ f\ x : (A\to B)\to A\to B}{\vdash \lambda f^{A\to B}.\ \lambda x^{A}.\ f\ x : ((A\to B)\times A)\to B}\ (\equiv)}{\vdash \lambda f^{A\to B}.\ \lambda x^{A}.\ f\ x : (A\times (A\to B))\to B}\ (\equiv)}{\vdash \lambda f^{A\to B}.\ \lambda x^{A}.\ f\ x : A\to (A\to B)\to B}\ (\equiv) \quad \vdash y : A}{\vdash (\lambda f^{A\to B}.\ \lambda x^{A}.\ f\ x)\ y : (A\to B)\to B}\ (\to_e) \quad \vdash g : A\to B}{\vdash (\lambda f^{A\to B}.\ \lambda x^{A}.\ f\ x)\ y\ g : B}\ (\to_e)
$$

$$
\begin{aligned}
&(\lambda f^{A\to B}.\ \lambda x^{A}.\ f\ x)\ y^{A}\ g^{A\to B} && \rightleftarrows_{curry}\\
&(\lambda f^{A\to B}.\ \lambda x^{A}.\ f\ x)\ \langle y^{A}, g^{A\to B}\rangle && \rightleftarrows_{comm}\\
&(\lambda f^{A\to B}.\ \lambda x^{A}.\ f\ x)\ \langle g^{A\to B}, y^{A}\rangle && \rightleftarrows_{curry}\\
&(\lambda f^{A\to B}.\ \lambda x^{A}.\ f\ x)\ g^{A\to B}\ y^{A}
\end{aligned}
$$

Sistema I Polimórfico

Resultados

Propiedades

- Preservación de tipos (si $r : A$ y $r \to s$, entonces $s : A$)
- Normalización fuerte (todos los programas tipados terminan)

Consecuencias

- Aplicación parcial mejorada

$$p : A \to B \to C \qquad\qquad p\,a \qquad p\,b$$

- Proyección de funciones

Sistema I Polimórfico

Resultados

Propiedades

- Preservación de tipos (si $r : A$ y $r \to s$, entonces $s : A$)
- Normalización fuerte (todos los programas tipados terminan)

Consecuencias

- Aplicación parcial mejorada

$$p : A \to B \to C \qquad\qquad p\ a \qquad {\color{red} p\ b}$$

$$elem\ x\ xs \qquad\qquad {\color{red} elem\ x} \qquad {\color{red} elem\ xs} \qquad {\color{red} elem\ xs\ x}$$

- Proyección de funciones

Sistema I Polimórfico

Resultados

Propiedades

- Preservación de tipos (si $r : A$ y $r \to s$, entonces $s : A$)
- Normalización fuerte (todos los programas tipados terminan)

Consecuencias

- Aplicación parcial mejorada
- Proyección de funciones

$$
\begin{array}{lllllll}
p : A \to (B \times C) & \quad & \mathit{fst}\ (p\, x^A) & \to^* & \mathit{fst}\ \langle r_1, r_2 \rangle & \to & r_1 \\
p' : A \to B & \quad & (\mathit{fst}\ p)\ x^A & \to^* & p'\, x^A & \to^* & r_1
\end{array}
$$

Sistema I Polimórfico

Resultados

Propiedades

- Preservación de tipos (si $r : A$ y $r \to s$, entonces $s : A$)
- Normalización fuerte (todos los programas tipados terminan)

Consecuencias

- Aplicación parcial mejorada
- Proyección de funciones

$$
\begin{array}{lllllll}
p : A \to (B \times C) & \quad & \mathit{fst}\ (p\, x^A) & \to^* & \mathit{fst}\ \langle r_1, r_2 \rangle & \to & r_1 \\
p' : A \to B & \quad & (\mathit{fst}\ p)\ x^A & \to^* & p' x^A & \to^* & r_1
\end{array}
$$

$$
\mathit{divMod}\ n_1\ \ n_2 \qquad\qquad \mathit{divMod}\ 5\ 2 \longrightarrow \langle 2, 1 \rangle
$$

$$
(\mathit{fst}\ \mathit{divMod})\ (5, 2) \longrightarrow \mathit{div}\ (5, 2) \longrightarrow 2
$$

Sistema I Polimórfico

Resultados

Propiedades

- Preservación de tipos (si $r : A$ y $r \to s$, entonces $s : A$)
- Normalización fuerte (todos los programas tipados terminan)

Consecuencias

- Aplicación parcial mejorada
- Proyección de funciones

```haskell
splitMin :: [a] -> (a,[a])
splitMin = \xs -> case xs of
                    [x]    -> (x,[])
                    (x:xs) -> (min x (fst r), max x (fst r) : snd r)
  where r = splitMin xs
```

```haskell
  fst splitMin
-->
  \xs -> case xs of
           [x]    -> x
           (x:xs) -> max x (splitMin xs)
```

Conclusiones y trabajo futuro

- Relación entre la computación y la lógica
- Lógicas/lenguajes del cubo lambda
- Proposiciones/tipos isomorfos
- Internalización de isomorfismos en lógicas/lenguajes del cubo lambda
- Internalización de isomorfismos en lenguajes prácticos