

---

# 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

$t, s, \dots ::= x^A \mid \lambda x^A. t \mid ts$

$A, B, \dots ::= \tau \mid A \rightarrow B$

# Cálculo lambda

## ► Términos y tipos

$t, s, \dots ::= x^A \mid \lambda x^A. t \mid ts$

$A, B, \dots ::= \tau \mid A \rightarrow B$

## ► Sistema de tipos

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

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

# Cálculo lambda

## ► Términos y tipos

$t, s, \dots ::= x^A \mid \lambda x^A. t \mid ts$

$A, B, \dots ::= \tau \mid A \rightarrow B$

## ► Sistema de tipos

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

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

## ► Ejemplos de tipado

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

# Cálculo lambda

## ► Términos y tipos

$$t, s, \dots ::= x^A \mid \lambda x^A. t \mid ts$$

$$A, B, \dots ::= \tau \mid A \rightarrow B$$

## ► Sistema de tipos

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

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

## ► Ejemplos de tipado

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

## ► Computación

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

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

# Correspondencia de Curry-Howard

# Correspondencia de Curry-Howard

**tipos = proposiciones**

$$A, B, \dots ::= \tau \mid A \rightarrow B$$

$$A, B, \dots ::= p \mid A \Rightarrow B$$



# Correspondencia de Curry-Howard

**tipos = proposiciones**

$$A, B, \dots ::= \tau \mid A \rightarrow B$$

$$A, B, \dots ::= p \mid A \Rightarrow B$$

Reglas de inferencia

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

$$\frac{}{\Gamma, A \vdash A} (ax)$$

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B} (\Rightarrow_i)$$

$$\frac{\Gamma \vdash A \rightarrow B \quad \Gamma \vdash A}{\Gamma \vdash B} (\Rightarrow_e)$$

# Correspondencia de Curry-Howard

tipos = proposiciones

$$A, B, \dots ::= \tau \mid A \rightarrow B$$

$$A, B, \dots ::= p \mid A \Rightarrow B$$

Reglas de inferencia

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

$$\frac{}{\Gamma, A \vdash A} (ax)$$

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B} (\Rightarrow_i)$$

$$\frac{\Gamma \vdash A \rightarrow B \quad \Gamma \vdash A}{\Gamma \vdash B} (\Rightarrow_e)$$

programas = pruebas

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

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

$$\frac{x : A \vdash \lambda x^A. fx : A \Rightarrow B}{\vdash \lambda f^{A \Rightarrow B}. \lambda x^A. fx : A \Rightarrow B} (\Rightarrow_i)$$

# Correspondencia de Curry-Howard

tipos = proposiciones

$$A, B, \dots ::= \tau \mid A \rightarrow B$$

$$A, B, \dots ::= p \mid A \Rightarrow B$$

Reglas de inferencia

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

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

programas = pruebas

$$\frac{}{x : A \vdash x : A} (ax) \quad \frac{}{f : A \Rightarrow B, x : A \vdash f : A \Rightarrow B} (ax) \quad \frac{}{f : A \Rightarrow B, x : A \vdash x : A} (ax)$$

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

computación = simplificación

$$\frac{\Gamma, x : A \vdash r : B}{\Gamma \vdash \lambda x^A. r : A \rightarrow B} (abs) \quad \frac{\Gamma \vdash \lambda x^A. r : A \rightarrow B \quad \Gamma \vdash s : A}{\Gamma \vdash (\lambda x^A. r)s : B} (app) \quad \Rightarrow \quad \Gamma \vdash [x := s]r : B$$

# Correspondencia de Curry-Howard

tipos = proposiciones

$$A, B, \dots ::= \tau \mid A \rightarrow B$$

$$A, B, \dots ::= p \mid A \Rightarrow B$$

Reglas de inferencia

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

$$\frac{}{\Gamma, A \vdash A} (ax)$$

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B} (\Rightarrow_i)$$

$$\frac{\Gamma \vdash A \rightarrow B \quad \Gamma \vdash A}{\Gamma \vdash B} (\Rightarrow_e)$$

programas = pruebas

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

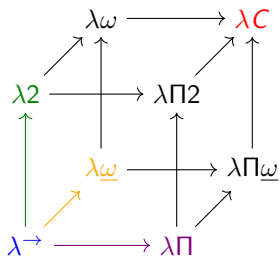
$$\frac{\frac{\frac{}{f : A \Rightarrow B, x : A \vdash f : A \Rightarrow B} (ax) \quad \frac{}{f : A \Rightarrow B, x : A \vdash x : A} (ax)}{f : A \Rightarrow B, x : A \vdash fx : B} (\Rightarrow_e)}{\frac{x : A \vdash \lambda x^A. fx : A \Rightarrow B}{\vdash \lambda f^{A \Rightarrow B}. \lambda x^A.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

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



$(\lambda \rightarrow)$  términos dependen de términos

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

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

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

$(\lambda C)$  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

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

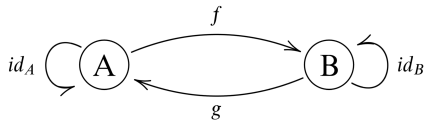
# Teoría de proposiciones/tipos isomorfos

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

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

tales que

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



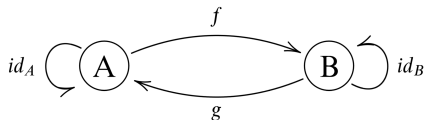
# Teoría de proposiciones/tipos isomorfos

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

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

tales que

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



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



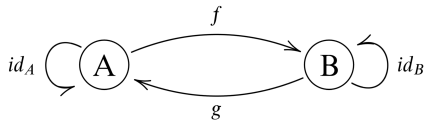
# Teoría de proposiciones/tipos isomorfos

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

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

tales que

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



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

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

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

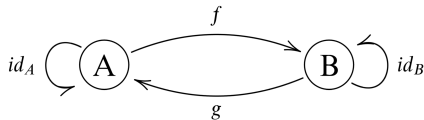
# Teoría de proposiciones/tipos isomorfos

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

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

tales que

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



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

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

# 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  $\text{suma} : (int, int) \rightarrow int$  no funciona si le paso dos argumentos  $\text{suma } 1\ 2$
- ▶ 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

$$f \stackrel{\text{def}}{=} \lambda p. \text{fst } p \star \text{snd } p$$

$$f \langle r, s \rangle$$

$$f \ r \ s$$

$$f' \stackrel{\text{def}}{=} \lambda x. \lambda y. x \star y$$

$$f' r \ s$$

$$f' \langle r, s \rangle$$

- ▶ Dinámicamente: adaptación implícita de programas y contextos

$$f \langle r, s \rangle \longrightarrow r \star s$$

$$f \ r \ s \longrightarrow f' r \ s \longrightarrow r \star s$$

$$f \ r \ s \longrightarrow f \langle r, s \rangle \longrightarrow r \star s$$

$$f' \langle r, s \rangle \longrightarrow f' r \ s \longrightarrow r \star s$$

$$f' \langle r, s \rangle \longrightarrow f' r \ s \longrightarrow r \star s$$

$$f' r \ s \longrightarrow r \star s$$

# Sistema I Polimórfico

## Definiciones

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

$$f \stackrel{\text{def}}{=} \lambda p. \text{fst } p \star \text{snd } p$$

$$f' \stackrel{\text{def}}{=} \lambda x. \lambda y. x \star y$$

$$f \langle r, s \rangle$$

$$f' r s$$

$$f r s$$

$$f' \langle r, s \rangle$$

- Dinámicamente: adaptación implícita de programas y contextos

$$C_1[p_1] \longrightarrow r$$

$$C_2[p_1] \longrightarrow C_2[p_2] \longrightarrow r$$

$$C_2[p_1] \longrightarrow C_1[p_1] \longrightarrow r$$

$$C_1[p_2] \longrightarrow C_1[p_1] \longrightarrow r$$

$$C_1[p_2] \longrightarrow C_2[p_2] \longrightarrow r$$

$$C_2[p_2] \longrightarrow r$$

# Sistema I Polimórfico

## Definiciones

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

$$\begin{aligned}
 A \times B &\equiv B \times A \\
 (A \times B) \rightarrow C &\equiv A \rightarrow B \rightarrow C \\
 A \rightarrow (B \times C) &\equiv (A \rightarrow B) \times (A \rightarrow C) \\
 \forall X.(A \rightarrow B) &\equiv A \rightarrow \forall X.B \quad X \notin FV(A)
 \end{aligned}$$

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

- Dinámicamente: adaptación implícita de programas y contextos

$$\begin{aligned}
 \langle r, s \rangle &\rightleftharpoons \langle s, r \rangle \\
 r \langle s, t \rangle &\rightleftharpoons rst \\
 \lambda x^A. \langle r, s \rangle &\rightleftharpoons \langle \lambda x^A. r, \lambda x^A. s \rangle \\
 \Lambda X. \lambda x^A. r &\rightleftharpoons \lambda x^A. \Lambda X. r \quad \text{si } X \notin FV(A)
 \end{aligned}$$

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

$\rightarrow \quad := \quad \rightleftharpoons^* \quad . \hookrightarrow \quad . \rightleftharpoons^*$

# Sistema I Polimórfico

## Ejemplo

$$\begin{array}{c}
 \vdash \lambda f^{A \rightarrow B}. \lambda x^A. f \ x : (A \rightarrow B) \rightarrow A \rightarrow B \\
 \hline
 \vdash \lambda f^{A \rightarrow B}. \lambda x^A. f \ x : ((A \rightarrow B) \times A) \rightarrow B \quad (\equiv) \\
 \hline
 \vdash \lambda f^{A \rightarrow B}. \lambda x^A. f \ x : (A \times (A \rightarrow B)) \rightarrow B \quad (\equiv) \\
 \hline
 \vdash \lambda f^{A \rightarrow B}. \lambda x^A. f \ x : A \rightarrow (A \rightarrow B) \rightarrow B \quad (\equiv) \\
 \hline
 \vdash (\lambda f^{A \rightarrow B}. \lambda x^A. f \ x) y : (A \rightarrow B) \rightarrow B \quad \vdash y : A \quad (\rightarrow_e) \\
 \hline
 \vdash (\lambda f^{A \rightarrow B}. \lambda x^A. f \ x) y \ g : B \quad \vdash g : A \rightarrow B \quad (\rightarrow_e)
 \end{array}$$

$$(\lambda f^{A \rightarrow B}. \lambda x^A. f \ x) y^A g^{A \rightarrow B} \quad \xleftrightarrow{\text{curry}}$$

$$(\lambda f^{A \rightarrow B}. \lambda x^A. f \ x) \langle y^A, g^{A \rightarrow B} \rangle \quad \xleftrightarrow{\text{comm}}$$

$$(\lambda f^{A \rightarrow B}. \lambda x^A. f \ x) \langle g^{A \rightarrow B}, y^A \rangle \quad \xleftrightarrow{\text{curry}}$$

$$(\lambda f^{A \rightarrow B}. \lambda x^A. f \ x) g^{A \rightarrow B} y^A$$

# Sistema I Polimórfico

## Resultados

### Propiedades

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

### Consecuencias

- ▶ Aplicación parcial mejorada

$$p : A \rightarrow B \rightarrow C \qquad p\ a \qquad p\ b$$

- ▶ Proyección de funciones



# Sistema I Polimórfico

## Resultados

### Propiedades

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

### Consecuencias

- ▶ Aplicación parcial mejorada

$p : A \rightarrow B \rightarrow C$

$p\ a\ \textcolor{red}{p\ b}$

$elem\ x\ xs$

$\textcolor{red}{elem\ x}\ \textcolor{red}{elem\ xs}\ \textcolor{red}{elem\ xs\ x}$

- ▶ Proyección de funciones

# Sistema I Polimórfico

## Resultados

### Propiedades

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

### Consecuencias

- ▶ Aplicación parcial mejorada
- ▶ Proyección de funciones

$$\begin{array}{llll} p : A \rightarrow (B \times C) & \text{fst } (p x^A) & \rightarrow^* & \text{fst } \langle r_1, r_2 \rangle \rightarrow r_1 \\ p' : A \rightarrow B & (\text{fst } p) x^A & \rightarrow^* & p' x^A \rightarrow^* r_1 \end{array}$$

# Sistema I Polimórfico

## Resultados

### Propiedades

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

### Consecuencias

- Aplicación parcial mejorada
- Proyección de funciones

$$p : A \rightarrow (B \times C) \qquad \text{fst } (p \ x^A) \rightarrow^* \text{fst } \langle r_1, r_2 \rangle \rightarrow r_1$$

$$p' : A \rightarrow B \qquad (\text{fst } p) \ x^A \rightarrow^* p' \ x^A \rightarrow^* r_1$$

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

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

# Sistema I Polimórfico

## Resultados

### Propiedades

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

### Consecuencias

- ▶ Aplicación parcial mejorada
- ▶ Proyección de funciones

```
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
```

```
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
- ▶ Propositiones/tipos isomorfos
- ▶ Internalización de isomorfismos en lógicas/lenguajes del cubo lambda
- ▶ Internalización de isomorfismos en lenguajes prácticos