

Factorizando espacios de derivación a través de tipos intersección

Gonzalo Ciruelos
Director: Pablo Barenbaum

Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

28 de junio de 2018

Espacios de derivación

Aritmética elemental y pares ordenados

$$\begin{aligned}\underline{n} + \underline{m} &\rightarrow \underline{n + m} \\ \underline{n} \cdot \underline{t} &\rightarrow \underbrace{\underline{t + t + \dots + t}}_{n \text{ veces}} \quad \text{si } n > 0 \\ \underline{0} \cdot \underline{t} &\rightarrow \underline{0}\end{aligned}$$

$$\begin{aligned}(t, s) &\rightarrow (t', s) && \text{si } t \rightarrow t' \\ (t, s) &\rightarrow (t, s') && \text{si } s \rightarrow s'\end{aligned}$$

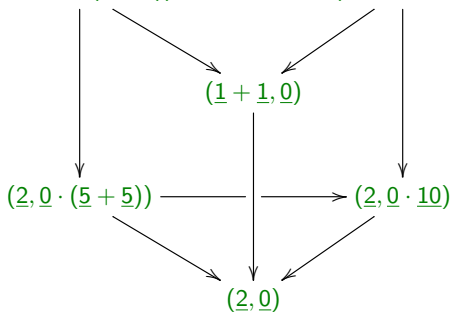
Espacios de derivación

$$\mathbb{D}[\underline{1} + \underline{1}] = \left(\underline{1} + \underline{1} \rightarrow \underline{2} \right)$$

$$\mathbb{D}[\underline{0} \cdot (\underline{5} + \underline{5})] = \underline{0} \cdot (\underline{5} + \underline{5}) \longrightarrow \underline{0} \cdot \underline{10}$$

$\underline{0}$

$$\mathbb{D}[(\underline{1} + \underline{1}, \underline{0} \cdot (\underline{5} + \underline{5}))] = (\underline{1} + \underline{1}, \underline{0} \cdot (\underline{5} + \underline{5})) \longrightarrow (\underline{1} + \underline{1}, \underline{0} \cdot \underline{10})$$



$$\mathbb{D}[(A, B)] \simeq \mathbb{D}[A] \times \mathbb{D}[B]$$

El cálculo- λ

Términos del cálculo- λ

$$t ::= x \mid t t \mid \lambda x. t$$

Por ejemplo,

$\lambda x. x$

$(\lambda x. z) y$

$(\lambda x. \lambda y. x) z w = ((\lambda x. (\lambda y. x)) z) w$

Variables libres

$$\text{fv}(x) \stackrel{\text{def}}{=} x$$

$$\text{fv}(u v) \stackrel{\text{def}}{=} \text{fv}(u) \cup \text{fv}(v)$$

$$\text{fv}(\lambda x. u) \stackrel{\text{def}}{=} \text{fv}(u) - \{x\}$$

El cálculo- λ

β -reducción

$$(\lambda x.t) s \rightarrow_{\beta} t\{x := s\}$$

...donde sustituir significa

$$x\{x := s\} \stackrel{\text{def}}{=} s$$

$$y\{x := s\} \stackrel{\text{def}}{=} y$$

$$(uv)\{x := s\} \stackrel{\text{def}}{=} u\{x := s\} v\{x := s\}$$

$$(\lambda y.u)\{x := s\} \stackrel{\text{def}}{=} \lambda y.u\{x := s\} \quad \text{si } x \neq y \text{ e } y \notin \text{fv}(s)$$

Por ejemplo

$$(\lambda x.x)((\lambda y.y)z) \rightarrow (\lambda y.y)z \rightarrow z$$

Espacios de derivación en el cálculo- λ

$$\mathbb{D}[(\lambda x.x)((\lambda y.y)z)] = \begin{array}{ccc} (\lambda x.x)((\lambda y.y)z) & \longrightarrow & (\lambda y.y)z \\ \downarrow & & \downarrow \\ (\lambda x.x)z & \longrightarrow & z \end{array}$$

Espacios de derivación en el cálculo- λ – Problemas

Creación

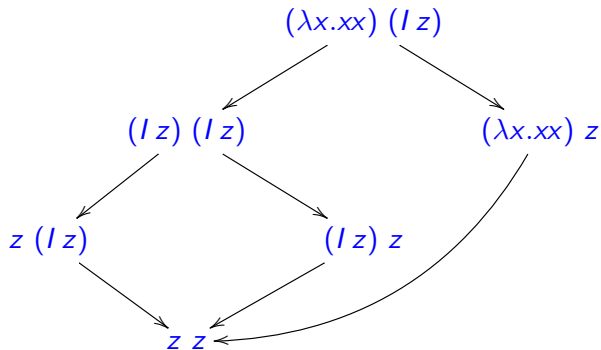
$$\begin{aligned} \mathbb{D}[(\lambda x.xx)(\lambda x.xx)] &= (\lambda x.xx)(\lambda x.xx) \\ &\downarrow \\ &(\lambda x.xx)(\lambda x.xx) \\ &\downarrow \\ &(\lambda x.xx)(\lambda x.xx) \\ &\downarrow \\ &\vdots \end{aligned}$$

$\mathbb{D}[(\lambda x.xx)(\lambda x.xx)]$ es infinito, mientras que $\mathbb{D}[(\lambda x.xx)]$ es finito.

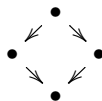
Espacios de derivación en el cálculo- λ – Problemas

Duplicación

$$\mathbb{D}[(\lambda x.xx) (I z)] =$$



$$\mathbb{D}[(\lambda x.xx) (I z)] \not\cong \mathbb{D}[(\lambda x.xx) \square] \times \mathbb{D}[I z] = \underbrace{\left((\lambda x.xx) \square \rightarrow \square \square \right)}_{\text{diamond}} \times \left(I z \rightarrow z \right)$$



Espacios de derivación en el cálculo- λ – Problemas

Borrado

$$\mathbb{D}[(\lambda x.y) (I z)] = (\lambda x.y) (I z) \longrightarrow (\lambda x.y) z$$

A triangular diagram with two arrows pointing from the middle terms to the variable y .

$$\mathbb{D}[(\lambda x.y) (I z)] \neq \mathbb{D}[(\lambda x.y) \square] \times \mathbb{D}[I z] = \underbrace{\left((\lambda x.y) \square \rightarrow y \right) \times \left(I z \rightarrow z \right)}_{\text{Diagram}}$$

A diagram showing four black dots arranged in a diamond shape with arrows pointing inward to a central point.

Residuos en el cálculo- λ

Definición

Sean R , S dos pasos coiniciales. Definimos el residuo de R después de S como lo que queda de el paso R después de hacer S : es un conjunto de pasos que salen del target de S . Lo escribimos como R/S .

Formalmente, se puede definir con posiciones o etiquetas.

La definición se puede extender para derivaciones: ρ/σ es lo que queda de ρ después hacer σ .

Para ordenar las derivaciones que salen de un mismo término, usamos el orden del prefijo.

Definición (Orden del prefijo)

$$[\rho] \sqsubseteq [\sigma] \stackrel{\text{def}}{\iff} \rho/\sigma = \epsilon$$

Espacios de derivación en el cálculo- λ

Definición (Equivalencia por permutaciones)

Decimos que dos secuencias de reducción ρ, σ son **equivalentes por permutación** si $\rho/\sigma = \epsilon$ y $\sigma/\rho = \epsilon$. Lo escribimos como $\rho \equiv \sigma$.

Definición (Espacio de derivación)

Si t es un término, $\mathbb{D}[t]$ es el conjunto de **secuencias de reducción** desde t :

$$\{\rho \mid \rho : t \rightarrow^* s \text{ es una secuencia de pasos de reescritura}\} / \equiv$$

Espacios de derivación en el cálculo- λ

Definición (Reticulado)

Un **reticulado** es un conjunto parcialmente ordenado (A, \leq) en el cual todo par de elementos tiene un supremo (mínima cota superior) y un ínfimo (máxima cota inferior).

Teorema (J.-J. Lévy)

En el cálculo- λ , $\mathbb{D}[t]$ forma un **semi-reticulado con supremos**, donde:

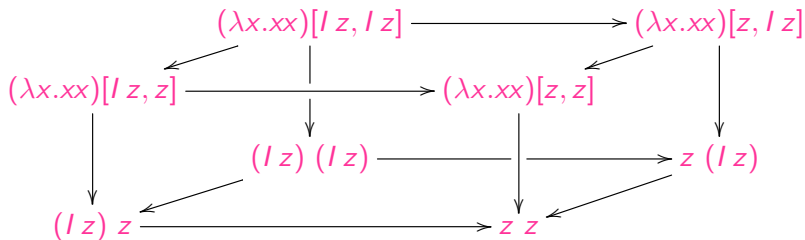
$$[\rho] \sqcup [\sigma] \stackrel{\text{def}}{=} [\rho(\sigma/\rho)]$$

$\mathbb{D}[t]$ no necesariamente es un reticulado.

Objetivo

Queremos **entender los espacios de derivación del cálculo- λ** .
Creemos que **explicitar el manejo de recursos puede ser útil**.

$$\mathbb{D}[(\lambda x.xx)[I z, I z]] =$$



El cálculo- λ distributivo ($\lambda^\#$)

- ▶ En sistemas de tipos intersección, distintas ocurrencias ligadas de una misma variable pueden tener distintos tipos.
- ▶ La no-idempotencia de la intersección nos da la capacidad de manejar cómo se usan los recursos.
- ▶ Nos basamos en el sistema \mathcal{W} , un sistema de tipos intersección no-idempotente.

Idea:

```
def f(x):  
    return x * x + x(100)
```

El parámetro se usa con dos tipos distintos. En consecuencia, el tipo de f es $[\text{Int}, \text{Int}, \text{Int} \rightarrow \text{Int}] \rightarrow \text{Int}$.

El cálculo- λ distributivo ($\lambda^\#$)

Definición ($\lambda^\#$ “naïf”)

Sintaxis

Términos	$t ::= x^A \mid t \vec{t} \mid \lambda x.t$	Listas de términos	$\vec{t} ::= [t_1, \dots, t_n]$
Tipos	$A ::= \alpha \mid \mathcal{M} \rightarrow A$	Multiconjuntos de tipos	$\mathcal{M} ::= [A_1, \dots, A_n]$
Contextos	$\Gamma ::= (\cdot) \mid \Gamma, x : \mathcal{M}$		

Tipado

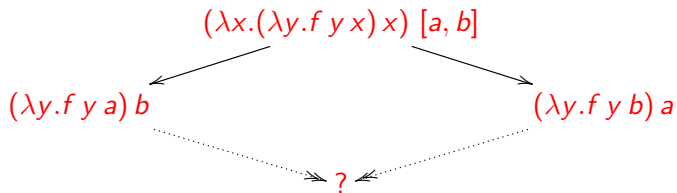
$$\frac{}{x : [A] \vdash x^A : A} \quad \frac{\Gamma \oplus x : \mathcal{M} \vdash t : A}{\Gamma \vdash \lambda x.t : \mathcal{M} \rightarrow A} \quad \frac{\Gamma \vdash t : [A_1, \dots, A_n] \rightarrow A \quad (\Delta_i \vdash s_i : A_i)_{i=1}^n}{\Gamma +_{i=1}^n \Delta_i \vdash t[s_1, \dots, s_n] : A}$$

Reducción

$$(\lambda x.t)[s_1, \dots, s_n] \rightarrow_\# t\{x := [s_1, \dots, s_n]\}$$

El cálculo- λ distributivo ($\lambda^\#$)

Problema! (no es confluente)



El cálculo- λ distributivo ($\lambda^\#$)

Definición ($\lambda^\#$)

Sintaxis

Términos	$t ::= x^A \mid t \vec{t} \mid \lambda^\ell x.t$	Listas de términos	$\vec{t} ::= [t_1, \dots, t_n]$
Tipos	$A ::= \alpha^\ell \mid \mathcal{M} \xrightarrow{\ell} A$	Multiconjuntos de tipos	$\mathcal{M} ::= [A_1, \dots, A_n]$
Contextos	$\Gamma ::= (\cdot) \mid \Gamma, x : \mathcal{M}$		

Tipado

$$\frac{}{x : [A] \vdash x^A : A} \quad \frac{\Gamma \oplus x : \mathcal{M} \vdash t : A}{\Gamma \vdash \lambda^\ell x.t : \mathcal{M} \xrightarrow{\ell} A} \quad \frac{\Gamma \vdash t : [A_1, \dots, A_n] \xrightarrow{\ell} B \quad (\Delta_i \vdash s_i : A_i)_{i=1}^n}{\Gamma +_{i=1}^n \Delta_i \vdash t[s_1, \dots, s_n] : B}$$

Reducción

$$(\lambda x.t)[s_1, \dots, s_n] \xrightarrow{\ell}_{\#} t\{x := [s_1, \dots, s_n]\}$$

La reducción es orientada por tipos.

Ejemplo de término. $(\lambda^1 x.x^{\alpha^2}) [y^{\alpha^2}]$

El cálculo- λ distributivo ($\lambda^\#$)

Definición (Términos correctos)

Decimos que un término tipable t es **correcto** si:

- ▶ Distintos lambdas tienen distintas etiquetas.
- ▶ Para todo multiconjunto de tipos $[A_1, \dots, A_n]$ que ocurra como una subfórmula en cualquier lugar de la derivación de tipos de t , si $i \neq j$ entonces A_i y A_j están decorados con distintas etiquetas en la raíz.

Comentario (Tipado único)

Si $\Gamma \vdash t : A$ es derivable y t correcto, entonces es la única derivación de tipo para t .

Lema (*Subject reduction*)

Si $\Gamma \vdash t : A$, el término t es correcto y $t \rightarrow_\# s$, entonces $\Gamma \vdash s : A$ y s es correcto.

El cálculo- λ distributivo ($\lambda^\#$)

Proposición (Confluencia)

El cálculo- $\lambda^\#$ cumple la propiedad de Church–Rosser.

En el ejemplo que antes no funcionaba:

$$\begin{array}{ccc} (\lambda^1 x. (\lambda^2 y. f^3 [x^4, y^5])[x^5]) [a^5, b^4] & \xrightarrow{1} & (\lambda^2 y. f^3 [b^4, y^5])[a^5] \\ \downarrow 2 & & \downarrow 2 \\ (\lambda^1 x. f^3 [x^4, x^5]) [a^5, b^4] & \xrightarrow{1} & f^3 [b^4, a^5] \end{array}$$

El cálculo- λ distributivo ($\lambda^\#$)

Proposición (Fuertemente normalizante)

No hay secuencias de reducción infinitas: $t_1 \rightarrow_\# t_2 \rightarrow_\# \dots$

Definición (Residuo)

Los **residuos** pueden definirse en $\lambda^\#$ utilizando las etiquetas de los lambdas.

Proposición (Ortogonalidad [cf. P.-A. Melliès])

El cálculo- $\lambda^\#$ es un sistema de reescritura abstracto ortogonal.

Lema

No hay duplicación ni borrado en $\lambda^\#$.

Proposición

En el cálculo- $\lambda^\#$, $\mathbb{D}[t]$ es un reticulado distributivo, a saber:

- ▶ existen supremos e ínfimos para cada par de reducciones, y
- ▶ las operaciones de *join* (\sqcup) y *meet* (\sqcap) distribuyen una sobre la otra.

Simulación

Definición (Refinamiento)

Damos una forma de relacionar términos correctos del cálculo- $\lambda^\#$ y el cálculo- λ .

$$\frac{}{x^\tau \times x} \quad \frac{t' \times t}{\lambda^\ell x.t' \times \lambda x.t} \quad \frac{t' \times t \quad s_i \times s \text{ para cada } i \in \{1, \dots, n\}}{t'[s_1, \dots, s_n] \times t s}$$

Un término λ puede tener muchos refinamientos:

$$\lambda^1 x.x^2 [] \times \lambda x.x x$$

$$\lambda^1 x.x^2 [x^3] \times \lambda x.x x$$

$$\lambda^1 x.x^2 [x^3, x^4] \times \lambda x.x x$$

...

También puede no tener ninguno, como $\Omega = (\lambda x.x x) (\lambda x.x x)$.

Simulación

Proposición (Simulación)

Del λ por el $\lambda^\#$. Si $t' \times t \rightarrow_\beta s$, entonces existe s' tal que:

$$\begin{array}{ccc} t & \xrightarrow{\beta} & s \\ \times & & \times \\ t' & \xrightarrow{\#} & s' \end{array}$$

Del $\lambda^\#$ por el λ . Si $t \times t' \rightarrow_\# s'$, entonces existen s y s'' tal que:

$$\begin{array}{ccc} t & \xrightarrow{\beta} & s \\ \times & & \times \\ t' & \xrightarrow{\#} s' & \xrightarrow{\#} s'' \end{array}$$

Simulación – *Head normal forms*

Definición (*Head normal form*)

Un término del cálculo- λ está en **head normal form** si no tiene redexes debajo de un contexto head:

$$H ::= \square \mid \lambda x.H \mid H t$$

Se define análogamente para el cálculo- $\lambda^\#$.

Proposición (La refinabilidad caracteriza la tenencia de *head normal forms*)

Dado t un término del cálculo- λ , son equivalentes:

1. El término t tiene una *head normal form*.
2. Existe un término t' del cálculo- $\lambda^\#$ tal que $t' \times t$.

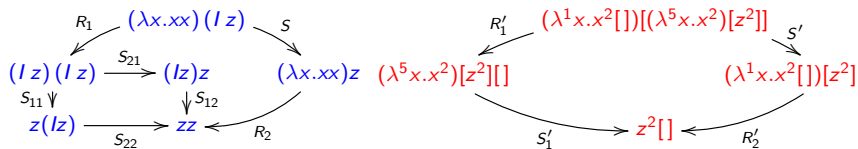
Simulación

Proposición (Simulación algebraica)

Para cada refinamiento $t' \times t$, la construcción dada por el resultado anterior es un morfismo de semirreticulados:

$$\begin{aligned} \mathbb{D}[t] &\rightarrow \mathbb{D}[t'] \\ \rho &\mapsto \rho/t' \end{aligned}$$

Ejemplo. Sean $l = \lambda x.x$ y $t = (\lambda x.xx)(l z)$. Es refinado por $t' = (\lambda^1 x.x^2[]) [(\lambda^5 x.x^2)[z^2]]$.



Basura

Definición (Basura)

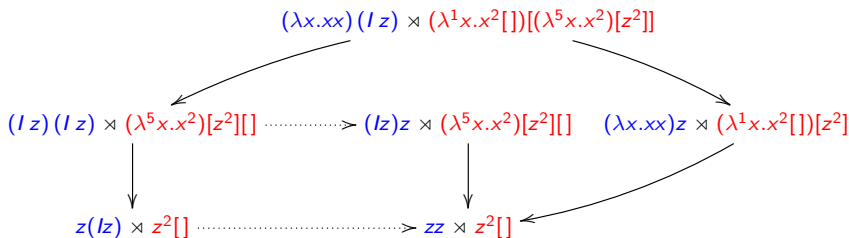
Sea $t' \times t$. Una derivación $\rho : t \rightarrow_{\beta}^* s$ es **t' -basura** si $\rho/t' = \epsilon$.

Definición (Libre de basura)

Sea $t' \times t$. Una derivación $\rho : t \rightarrow_{\beta}^* s$ es **t' -libre de basura** si para cada $\sigma \sqsubseteq \rho$, si ρ/σ es (t'/σ) -basura, entonces $\rho/\sigma = \epsilon$.

Notemos que las definiciones dependen de la elección de t' .

Ejemplo. Los pasos punteados son basura.



Basura y factorización

Teorema (Factorización)

Si $t' \times t$, existe un isomorfismo de semirreticulados:

$$\mathbb{D}[t] \simeq \int_{\mathcal{F}} \mathcal{G}$$

donde:

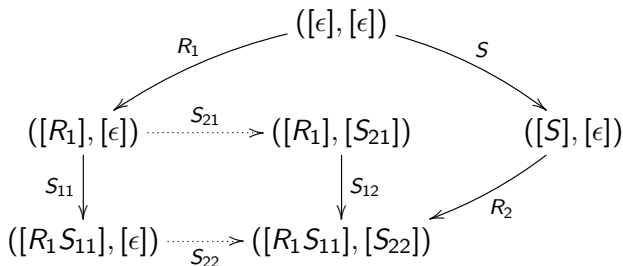
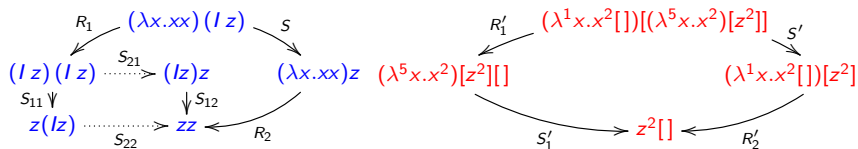
- ▶ \mathcal{F} es el reticulado de derivaciones t' -libres de basura.
- ▶ $\mathcal{G} : \mathcal{F} \rightarrow \text{Semilattice}$ es un funtor que a cada derivación libre de basura $\rho : t \rightarrow_{\beta}^* s$ en \mathcal{F} le asigna el semirreticulado de derivaciones basura que salen de s (que son las (t'/ρ) -basura).
- ▶ $\int_{\mathcal{F}} \mathcal{G}$ es la construcción de Grothendieck.

Corolario

Toda derivación ρ de t se puede factorizar de manera única en $\rho \equiv \rho_1 \rho_2$ donde ρ_1 es libre de basura y ρ_2 es basura.

Factorización – Ejemplo

Sea $t = (\lambda x.xx)(Iz)$. Lo refinaba $t' = (\lambda^1 x.x^2[]) [(\lambda^5 x.x^2)[z^2]]$.



Trabajo futuro

- ▶ Estudiar la relación de los términos que refinan a un t con la noción de aproximantes de *head normal forms*.
- ▶ Intentar hacer lo mismo con otros cálculos de recursos.
- ▶ Relacionar que la factorización dada con la factorización interna–externa de Melliés.
- ▶ Obtener resultados cuantitativos con la teoría desarrollada.