# Factoring Derivation Spaces via Intersection Types

APLAS 2018

**Pablo Barenbaum** **Gonzalo Ciruelos**

# Derivation spaces

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

$$\mathbb{D}[3+4*5] = \left( 3+4*5 \longrightarrow 3+20 \longrightarrow 23 \right)$$

$$\mathbb{D}[(1+1,\ 3+4*5)] =$$

$$\begin{array}{ccccc}
(1+1,\ 3+4*5) & \longrightarrow & (1+1,\ 3+20) & \longrightarrow & (1+1,\ 23) \\
\downarrow & & \downarrow & & \downarrow \\
(2,\ 3+4*5) & \longrightarrow & (2,\ 3+20) & \longrightarrow & (2,\ 23)
\end{array}$$

$$\underbrace{\mathbb{D}[(A,B)] \simeq \mathbb{D}[A] \times \mathbb{D}[B]}_{\text{isomorphism of lattices}}$$

# Derivation spaces in the $\lambda$-calculus

**Creation.**

$$\begin{array}{rcl}
\mathbb{D}[(\lambda x.f\,(x\,x))\,\lambda x.f\,(x\,x)] & = & (\lambda x.f\,(x\,x))\,\lambda x.f\,(x\,x) \\
& & \downarrow \\
& & f\,((\lambda x.f\,(x\,x))\,\lambda x.f\,(x\,x)) \\
& & \downarrow \\
& & f\,(f\,((\lambda x.f\,(x\,x))\,\lambda x.f\,(x\,x))) \\
& & \downarrow \\
& & \vdots
\end{array}$$

$\mathbb{D}[\lambda x.f\,(x\,x)]$ is finite but $\mathbb{D}[(\lambda x.f\,(x\,x))\lambda x.f\,(x\,x)]$ is infinite.

# Derivation spaces in the $\lambda$-calculus

**Duplication.**

$$\mathbb{D}[(\lambda x.x\,x)\,(I\,y)] \quad = \quad \begin{tikzcd}
 & & (\lambda x.x\,x)\,(I\,y) \arrow[dl] \arrow[dr] & \\
 & (I\,y)\,(I\,y) \arrow[dl] \arrow[dr] & & (\lambda x.x\,x)\,y \arrow[ddll, bend left] \\
y\,(I\,y) \arrow[dr] & & (I\,y)\,y \arrow[dl] & \\
 & y\,y & &
\end{tikzcd}$$

$$\mathbb{D}[(\lambda x.x\,x)\,(I\,y)] \not\simeq \underbrace{\mathbb{D}[(\lambda x.x\,x)\Box] \times \mathbb{D}[I\,y]}$$

$$\begin{tikzcd}
 & \bullet \arrow[dl] \arrow[dr] & \\
\bullet \arrow[dr] & & \bullet \arrow[dl] \\
 & \bullet &
\end{tikzcd}$$

# Derivation spaces in the $\lambda$-calculus

**Erasure.**

$$\mathbb{D}[(\lambda x.z)(I\,y)] \;=\;
\begin{tikzcd}
(\lambda x.z)\,(I\,y) \arrow[dd] \arrow[dr] & \\
 & (\lambda x.z)\,y \arrow[dl] \\
z &
\end{tikzcd}$$

$$\mathbb{D}[(\lambda x.z)\,(I\,y)] \not\simeq \underbrace{\mathbb{D}[(\lambda x.z)\Box] \times \mathbb{D}[I\,y]}_{
\begin{tikzcd}[column sep=small, row sep=small]
 & \bullet \arrow[dl] \arrow[dr] & \\
\bullet \arrow[dr] & & \bullet \arrow[dl] \\
 & \bullet &
\end{tikzcd}}$$

# Derivation spaces in the $\lambda$-calculus

### Definition (Derivation space)

If $t$ is a term, $\mathbb{D}[t]$ is the set of **reduction sequences**[1] from $t$:

$$\left\{\rho \mid \rho : t \to^* s \text{ is a sequence of rewrite steps}\right\} \Big/ \equiv$$

Partially ordered by the **prefix order**:

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

[1] Modulo permutation equivalence.

# Derivation spaces in the $\lambda$-calculus

Theorem (J.-J. Lévy)

*In the $\lambda$-calculus, $\mathbb{D}[t]$ forms an* **upper semilattice** *with:*

$$[\rho] \sqcup [\sigma] \quad \stackrel{\mathrm{def}}{=} \quad [\rho\,(\sigma/\rho)]$$

*It is not necessarily a lattice.*

# (Soft) goals and hypotheses

Goal: **understanding derivation spaces**.

Hypothesis: **explicit resource management may be helpful**.

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

$$\begin{tikzcd}
 & (\lambda x.x\,x)\,[I\,y, I\,y] \arrow[rr] \arrow[dl] \arrow[dd] & & (\lambda x.x\,x)\,[y, I\,y] \arrow[dl] \arrow[dd] \\
(I\,y)\,(I\,y) \arrow[rr] \arrow[dd] & & y\,(I\,y) \arrow[dd] & \\
 & (\lambda x.x\,x)\,[I\,y, y] \arrow[rr] \arrow[dl] & & (\lambda x.x\,x)\,[y, y] \arrow[dl] \\
(I\,y)\,y \arrow[rr] & & y\,y &
\end{tikzcd}$$

# The distributive $\lambda$-calculus ($\lambda^{\#}$)

Proof-term notation for the non-idempotent intersection type system $\mathcal{W}$.

## Definition (Proto-$\lambda^{\#}$)

**Syntax.**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Terms | $t$ | $::=$ | $x^A \mid \lambda\ x.t \mid t\ \vec{t}$ | Lists of terms | $\vec{t}$ | $::=$ | $[t_1, \dots, t_n]$ |
| Types | $A$ | $::=$ | $\alpha \mid \mathcal{M} \to A$ | Multisets of types | $\mathcal{M}$ | $::=$ | $[A_1, \dots, A_n]$ |
| Contexts | $\Gamma$ | $::=$ | $(.) \mid \Gamma, x : \mathcal{M}$ | | | | |

**Typing.**

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

**Reduction.**

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

Each free occurrence of $x^A$ consumes exactly one argument $s_i$ of type $A$.

# The distributive $\lambda$-calculus ($\lambda^{\#}$)

Proof-term notation for the non-idempotent intersection type system $\mathcal{W}$.

## Definition (Proto-$\lambda^{\#}$)

**Syntax.**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Terms | $t$ | $::=$ | $x^A \mid \lambda\ x.t \mid t\,\vec{t}$ | Lists of terms | $\vec{t}$ | $::=$ | $[t_1, \ldots, t_n]$ |
| Types | $A$ | $::=$ | $\alpha\ \mid \mathcal{M} \to A$ | Multisets of types | $\mathcal{M}$ | $::=$ | $[A_1, \ldots, A_n]$ |
| Contexts | $\Gamma$ | $::=$ | $(.) \mid \Gamma, x : \mathcal{M}$ | | | | |

**Typing.**

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

**Reduction.**

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

Each free occurrence of $x^A$ consumes exactly one argument $s_i$ of type $A$.

**(Non-confluent).**

$$(\lambda x.(\lambda y.f\ y\ x)\ x)\,[a, b]$$

# The distributive $\lambda$-calculus ($\lambda^{\#}$)

## Definition ($\lambda^{\#}$)

**Syntax.**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Terms | $t$ | $::=$ | $x^A \mid \lambda^{\mathsf{L}} x.t \mid t\,\vec{t}$ | Lists of terms | $\vec{t}$ | $::=$ | $[t_1, \ldots, t_n]$ |
| Types | $A$ | $::=$ | $\alpha^{\mathsf{L}} \mid \mathcal{M} \xrightarrow{\mathsf{L}} A$ | Multisets of types | $\mathcal{M}$ | $::=$ | $[A_1, \ldots, A_n]$ |
| Contexts | $\Gamma$ | $::=$ | $(.) \mid \Gamma, x : \mathcal{M}$ | | | | |

**Typing.**

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

**Reduction.**

$$(\lambda^{\mathsf{L}} x.t)\,[s_1, \ldots, s_n] \xrightarrow{\mathsf{L}}_{\#} t\{x := [s_1, \ldots, s_n]\}$$

Each free occurrence of $x^A$ consumes exactly one argument $s_i$ of type $A$.

# The distributive $\lambda$-calculus ($\lambda^{\#}$)

### Remark (Unique typing)

If $\Gamma \vdash t : A$ is derivable, there is a unique typing derivation for $t$.

### Definition (Correct terms)

A typable term $t$ is **correct** if:

- Different lambdas are decorated with different labels.
- Given a multiset of types $[A_1, \dots, A_n]$ occurring as a subformula anywhere in the typing derivation of $t$, if $i \neq j$ then $A_i$ and $A_j$ are decorated with different labels at the root.

### Lemma (Subject reduction)

If $\Gamma \vdash t : A$, the term $t$ is correct and $t \to_{\#} s$ then $\Gamma \vdash s : A$ and $s$ is correct.

# The distributive $\lambda$-calculus ($\lambda^\#$)

**Proposition (Confluence)**

The $\lambda^\#$-calculus has the Church–Rosser property.

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

# The distributive $\lambda$-calculus ($\lambda^{\#}$)

### Proposition (Strong normalization)

There is no infinite reduction sequence $t_1 \to_{\#} t_2 \to_{\#} \ldots$.

*[cf. System $\mathcal{W}$]*

Residuals can be defined in $\lambda^{\#}$ using the labels over the lambdas.

### Lemma

There is no duplication nor erasure in $\lambda^{\#}$.

### Proposition

In the $\lambda^{\#}$-calculus, $\mathbb{D}[t]$ is a distributive lattice.
There are joins ($\sqcup$) and meets ($\sqcap$) that distribute over each other.

# Simulation

### Definition (Refinement)

Refinement ($\ltimes$) relates correct $\lambda^{\#}$-terms and $\lambda$-terms:

$$\frac{}{x^A \ltimes x} \qquad \frac{t' \ltimes t}{\lambda^{\mathsf{L}} x.t' \ltimes \lambda x.t} \qquad \frac{t' \ltimes t \quad (s'_i \ltimes s)_{i=1}^{n}}{t'\,[s'_1, \ldots, s'_n] \ltimes t\,s}$$

A $\lambda$-term may have many refinements:

$$\begin{array}{rcl}
(\lambda^1 x.y^2\,[]\,[])\,[] & \ltimes & (\lambda x.y\,x\,x)\,z \\
(\lambda^1 x.y^2\,[]\,[x^3])\,[z^3] & \ltimes & (\lambda x.y\,x\,x)\,z \\
(\lambda^1 x.y^2\,[x^3]\,[x^4, x^5])\,[z^5, z^3, z^4] & \ltimes & (\lambda x.y\,x\,x)\,z \\
& \ldots &
\end{array}$$

# Simulation

Proposition (Simulation)

**Forward.** If $t' \ltimes t \to_\beta s$ there is a term $s'$ such that:

$$
\begin{tikzcd}
t \arrow[r, "\beta"'] \arrow[d, phantom, "\rotatebox{90}{$\ltimes$}"] & s \arrow[d, phantom, "\rotatebox{90}{$\ltimes$}"] \\
t' \arrow[r, dotted, twoheadrightarrow, "\#"'] & s'
\end{tikzcd}
$$

**Reverse.** If $t \rtimes t' \to_\# s'$ there are terms $s$, $s''$ such that:

$$
\begin{tikzcd}
t \arrow[rr, dotted, "\beta"'] \arrow[d, phantom, "\rotatebox{90}{$\rtimes$}"] & & s \arrow[d, phantom, "\rotatebox{90}{$\rtimes$}"] \\
t' \arrow[r, "\#"'] & s' \arrow[r, dotted, twoheadrightarrow, "\#"'] & s''
\end{tikzcd}
$$

# Simulation

Proposition (Refinement characterizes head normalization)

The following are equivalent:

1. The term $t$ has a refinement $t' \ltimes t$.
2. The term $t$ has a head normal form.

*[cf. System $\mathcal{W}$]*

# Simulation

### Proposition (Algebraic simulation)

For each refinement $t' \ltimes t$ the construction given by the Simulation result is a morphism of upper semilattices:

$$\begin{array}{ccc} \mathbb{D}[t] & \to & \mathbb{D}[t'] \\ \rho & \mapsto & \rho/t' \end{array}$$

Its definition and properties resemble residual theory.
*E.g.* there is a "cube lemma":

$$(\rho/t')/(\sigma/t') \equiv (\rho/\sigma)/(t'/\sigma)$$

# Garbage and factorization

### Definition (Garbage)

Let $t' \ltimes t$. A derivation $\rho : t \to_\beta^* s$ is **$t'$-garbage** if $\rho / t' = \epsilon$.

The notion of garbage depends on the choice of $t'$.

The dotted steps are garbage:

$$\begin{tikzcd}
 & & (\lambda^1 x.x^2[])[(\lambda^3 z.z^2)[y^2]] \ltimes (\lambda x.x\,x)\,(I\,y) \arrow[dl] \arrow[drr] & & \\
 & (\lambda^3 z.z^2)[y^2][] \ltimes (I\,y)\,(I\,y) \arrow[dl] \arrow[drr, dotted] & & & (\lambda^1 x.x^2[])[y^2] \ltimes (\lambda x.x\,x)\,y \arrow[ddlll, bend left] \\
y^2[] \ltimes y\,(I\,y) \arrow[dr, dotted] & & & (\lambda^3 z.z^2)[y^2][] \ltimes (I\,y)\,y \arrow[dll] & \\
 & y^2[] \ltimes y\,y & & &
\end{tikzcd}$$

# Garbage and factorization

### Theorem (Factorization)

*If $t' \ltimes t$ there is an isomorphism of upper semilattices:*

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

*where:*

- $\int_{\mathcal{F}} \mathcal{G}$ *is the Grothendieck construction.*
- $\mathcal{F}$ *is the lattice of* **garbage-free** *derivations.*
- $\mathcal{G} : \mathcal{F} \to \mathsf{Semilattice}$ *is a functor.*
  *For each $\rho : t \to_{\beta}^{*} s$ in $\mathcal{F}$, we write $\mathcal{G}(\rho)$ for the semilattice of* **garbage** *derivations starting at $s$.*

In particular, for any derivation $\rho : t \to_{\beta}^{*} s$ there is a **unique** way to factor $\rho \equiv \rho_1 \, \rho_2$ such that $\rho_1$ is garbage-free and $\rho_2$ is garbage.

# Future work

- Generalize to other rewriting systems.
- Show that the notion of garbage is not *ad hoc*.
- Relate with Melliès external–internal factorization.
- Use other resource calculi instead of $\lambda^{\#}$.