Parallel Reduction in Resource Lambda-Calculus

Michele Pagani[1] Paolo Tranquilli[2]

[1]Dipartimento di Informatica
Università di Torino
paganI@di.unito.it

[2]Laboratoire de l'Informatique du Parallélisme
ENS de Lyon
paolo.tranquilli@ens-lyon.fr

Asian Symposium on Programming Languages and Systems
Seoul, 14-16/12/2009

Outline

Outline

Previously, on Resource Calculus – 1936

- Alonzo Church's λ-calculus.

$$M, N ::= x \mid \lambda x.M \mid MN \qquad \text{(terms)}$$
$$(\lambda x.M)N \to M\{N/x\} \qquad \text{(beta)}$$

(closed by context)

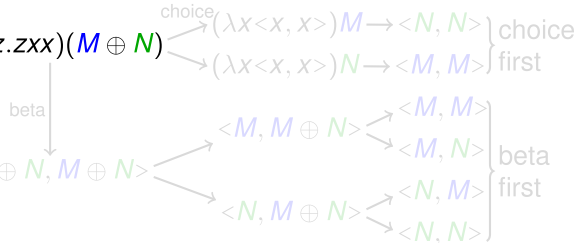
- A Turing-complete programming language stripped to the bone.
- The core of functional programming languages (e.g. ML⇝OCaml).
- The cradle of Curry-Howard isomorphism.

Previously, on Resource Calculus – '70s and on

- Nondeterministic λ-calculus (many authors/forms). E.g.

$$M, N ::= x \mid \lambda x.M \mid MN \mid M \oplus N \qquad \text{(terms)}$$
$$(\lambda x.M)N \to M\{N/x\} \qquad \text{(beta)}$$
$$M \oplus N \to M \qquad M \oplus N \to N \qquad \text{(choice)}$$

- But: $(\lambda x, z.zxx)(M \oplus N)$

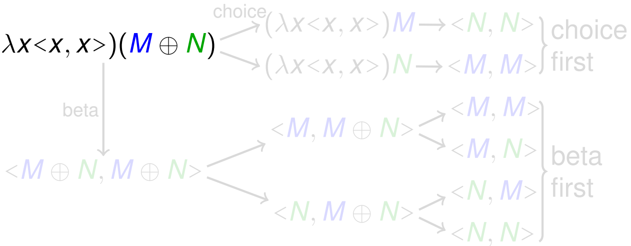


Previously, on Resource Calculus – ’70s and on

- Nondeterministic λ-calculus (many authors/forms). E.g.

$$M, N ::= x \mid \lambda x.M \mid MN \mid M \oplus N \qquad \text{(terms)}$$
$$(\lambda x.M)N \to M\{N/x\} \qquad \text{(beta)}$$
$$M \oplus N \to M \qquad M \oplus N \to N \qquad \text{(choice)}$$

- But: $(\lambda x<x, x>)(M \oplus N)$

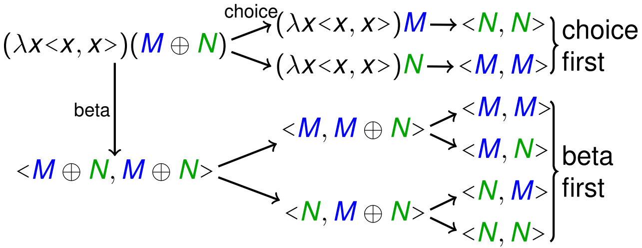


Previously, on Resource Calculus – ’70s and on

- Nondeterministic λ-calculus (many authors/forms). E.g.

$$M, N ::= x \mid \lambda x.M \mid MN \mid M \oplus N \qquad \text{(terms)}$$
$$(\lambda x.M)N \to M\{N/x\} \qquad \text{(beta)}$$
$$M \oplus N \to M \qquad M \oplus N \to N \qquad \text{(choice)}$$

- But:



Argument position is not linear

Let us model nondeterminism syntactically by formal sums:

$$M \oplus N \to M + N$$

We can restate the problem: argument position is not linear:

$$(M_1 + M_2)N = M_1 N + M_2 N \rightsquigarrow ✔$$

$$\lambda x.(M_1 + M_2) = \lambda x.M_1 + \lambda x.M_2 \rightsquigarrow ✔$$

$$M(N_1 + N_2) = MN_1 + MN_2 \rightsquigarrow ✖$$

With last equality $(\lambda x.\langle x, x\rangle)(M \oplus N)$ has different results

$\rightsquigarrow \quad \rightsquigarrow \quad \rightsquigarrow \quad \rightsquigarrow$

Nondeterminism not internal as intended.

Nondeterministic calculi were designed as lazy

$\Longrightarrow$ reducing only in linear positions.

Argument position is not linear

Let us model nondeterminism syntactically by formal sums:

$$M \oplus N \to M + N$$

We can restate the problem: argument position is not linear:

$$(M_1 + M_2)N = M_1 N + M_2 N \rightsquigarrow ✔$$

$$\lambda x.(M_1 + M_2) = \lambda x.M_1 + \lambda x.M_2 \rightsquigarrow ✔$$

$$M(N_1 + N_2) = MN_1 + MN_2 \rightsquigarrow ✘$$

With last equality system is not confluent

⇓ ⇓ ⇓ ⇓

Nondeterminism not internal as intended.

Nondeterministic calculi were designed as lazy

$\Longrightarrow$ reducing only in linear positions.

Previously, on Resource Calculus – 1993

- Gerard Boudol's λ-calculus with resources.

$$M, N ::= x \mid \lambda x.M \mid MP \qquad \text{(terms)}$$
$$P, Q ::= [M_1, \ldots, M_h, N_1^!, \ldots, N_k^!] \qquad \text{(bags)}$$

- Ideas:
  - there are ephemeral, one-use arguments. . .
  - and perpetual, reusable ones. . .
  - and they are mixed together in multisets (notation: $1 = [\,]$, $P \cdot Q$ union)

Gérard Boudol.
The lambda-calculus with multiplicities.
INRIA Research Report 2025, 1993.

Resource depletion in Boudol's calculus

Weak lazy head reduction, roughly:

$$
\begin{array}{llr}
(\lambda x.M)PQ_1 \dots Q_k;\, env & \to MQ_1 \dots Q_k;\, \langle\!\langle P/x \rangle\!\rangle env & \text{(beta)} \\
hQ_1 \cdots Q_k;\, \langle\!\langle [M^!] \cdot P/h \rangle\!\rangle env & \to MQ_1 \dots Q_k;\, \langle\!\langle [M^!] \cdot P/h \rangle\!\rangle env & \text{(fetch-!)} \\
hQ_1 \cdots Q_k;\, \langle\!\langle [M] \cdot P/h \rangle\!\rangle env & \to MQ_1 \dots Q_k;\, \langle\!\langle P/h \rangle\!\rangle env & \text{(fetch-}\ell\text{)}
\end{array}
$$

(skipping *much* bureaucracy here)

- Usual λ-calculus (lazy): $MN = M[N^!]$.
- Choice: $M \oplus N := (\lambda x.x)[M^!, N^!]$.
- Deadlocks (or rather starvation): $x; \langle\!\langle 1/x \rangle\!\rangle$.
- Surplus resources: ignored in Boudol's calculus, e.g.

$$
(\lambda x.x)[M, N] \to x; \langle\!\langle [M, N]/x \rangle\!\rangle \nearrow M; \langle\!\langle [N]/x \rangle\!\rangle \cong M; \emptyset \quad \searrow N; \langle\!\langle [M]/x \rangle\!\rangle \cong N; \emptyset
$$

Resource depletion in Boudol's calculus

Weak lazy head reduction, roughly:

$$
\begin{array}{llll}
(\lambda x.M)PQ_1\dots Q_k; env & \to MQ_1\dots Q_k; \langle\langle P/x\rangle\rangle env & & \text{(beta)}\\
hQ_1\cdots Q_k; \langle\langle [M^!]\cdot P/h\rangle\rangle env & \to MQ_1\dots Q_k; \langle\langle [M^!]\cdot P/h\rangle\rangle env & & \text{(fetch-!)}\\
hQ_1\cdots Q_k; \langle\langle [M]\cdot P/h\rangle\rangle env & \to MQ_1\dots Q_k; \langle\langle P/h\rangle\rangle env & & \text{(fetch-}\ell\text{)}
\end{array}
$$

(skipping *much* bureaucracy here)

- Usual λ-calculus (lazy): $MN = M[N^!]$.
- Choice: $M \oplus N := (\lambda x.x)[M^!, N^!]$.
- Deadlocks (or rather starvation): $x; \langle\langle 1/x\rangle\rangle$.
- Surplus resources: ignored in Boudol's calculus, e.g.

$$
(\lambda x.x)[M,N] \to x; \langle\langle [M,N]/x\rangle\rangle \begin{cases} \to M; \langle\langle [N]/x\rangle\rangle \cong M; \emptyset \\ \to N; \langle\langle [M]/x\rangle\rangle \cong N; \emptyset \end{cases}
$$

Resource depletion in Boudol's calculus

Weak lazy head reduction, roughly:

$$
\begin{array}{llr}
(\lambda x.M)PQ_1 \ldots Q_k; env & \to MQ_1 \ldots Q_k; \langle\!\langle P/x \rangle\!\rangle env & \text{(beta)} \\
hQ_1 \cdots Q_k; \langle\!\langle [M^!] \cdot P/h \rangle\!\rangle env & \to MQ_1 \ldots Q_k; \langle\!\langle [M^!] \cdot P/h \rangle\!\rangle env & \text{(fetch-!)} \\
hQ_1 \cdots Q_k; \langle\!\langle [M] \cdot P/h \rangle\!\rangle env & \to MQ_1 \ldots Q_k; \langle\!\langle P/h \rangle\!\rangle env & \text{(fetch-}\ell\text{)}
\end{array}
$$

(skipping *much* bureaucracy here)

- Usual λ-calculus (lazy): $MN = M[N^!]$.
- Choice: $M \oplus N := (\lambda x.x)[M^!, N^!]$.
- Deadlocks (or rather starvation): $x; \langle\!\langle 1/x \rangle\!\rangle$.
- Surplus resources: ignored in Boudol's calculus, e.g.

$$
(\lambda x.x)[M, N] \to x; \langle\!\langle [M, N]/x \rangle\!\rangle
\begin{array}{l}
\nearrow\ M; \langle\!\langle [N]/x \rangle\!\rangle \cong M; \emptyset \\
\searrow\ N; \langle\!\langle [M]/x \rangle\!\rangle \cong N; \emptyset
\end{array}
$$

Resource depletion in Boudol's calculus

Weak lazy head reduction, roughly:

$$(\lambda x.M)PQ_1 \dots Q_k; env \to MQ_1 \dots Q_k; \langle\!\langle P/x \rangle\!\rangle env \qquad \text{(beta)}$$
$$hQ_1 \cdots Q_k; \langle\!\langle [M^!] \cdot P/h \rangle\!\rangle env \to MQ_1 \dots Q_k; \langle\!\langle [M^!] \cdot P/h \rangle\!\rangle env \qquad \text{(fetch-!)}$$
$$hQ_1 \cdots Q_k; \langle\!\langle [M] \cdot P/h \rangle\!\rangle env \to MQ_1 \dots Q_k; \langle\!\langle P/h \rangle\!\rangle env \qquad \text{(fetch-}\ell\text{)}$$

(skipping *much* bureaucracy here)

- Usual λ-calculus (lazy): $MN = M[N^!]$.
- Choice: $M \oplus N := (\lambda x.x)[M^!, N^!]$.
- Deadlocks (or rather starvation): $x; \langle\!\langle 1/x \rangle\!\rangle$.
- Surplus resources: ignored in Boudol's calculus, e.g.

$$(\lambda x.x)[M,N] \to x; \langle\!\langle [M,N]/x \rangle\!\rangle \nearrow M; \langle\!\langle [N]/x \rangle\!\rangle \cong M; \emptyset$$
$$\searrow N; \langle\!\langle [M]/x \rangle\!\rangle \cong N; \emptyset$$

Resource depletion in Boudol's calculus

Weak lazy head reduction, roughly:

$$(\lambda x.M)PQ_1 \ldots Q_k; env \to MQ_1 \ldots Q_k; \langle\langle P/x \rangle\rangle env \quad \text{(beta)}$$
$$hQ_1 \cdots Q_k; \langle\langle [M^!] \cdot P/h \rangle\rangle env \to MQ_1 \ldots Q_k; \langle\langle [M^!] \cdot P/h \rangle\rangle env \quad \text{(fetch-!)}$$
$$hQ_1 \cdots Q_k; \langle\langle [M] \cdot P/h \rangle\rangle env \to MQ_1 \ldots Q_k; \langle\langle P/h \rangle\rangle env \quad \text{(fetch-}\ell\text{)}$$

(skipping *much* bureaucracy here)

- Usual λ-calculus (lazy): $MN = M[N^!]$.
- Choice: $M \oplus N := (\lambda x.x)[M^!, N^!]$.
- Deadlocks (or rather starvation): $x; \langle\langle 1/x \rangle\rangle$.
- Surplus resources: ignored in Boudol's calculus, e.g.

$$(\lambda x.x)[M, N] \to x; \langle\langle [M, N]/x \rangle\rangle \begin{cases} \to M; \langle\langle [N]/x \rangle\rangle \cong M; \emptyset \\ \to N; \langle\langle [M]/x \rangle\rangle \cong N; \emptyset \end{cases}$$

Initial motivation

Equivalence on ordinary terms M, N:

$$\forall \text{ context with resources } C[\,] : \; C[M]\!\downarrow \text{ iff } \; C[N]\!\downarrow$$
$$\Updownarrow$$
$$[\![M]\!] \approx [\![N]\!]$$

where $[\![\;]\!]$ is Milner's translation of λ-terms to π-calculus, $\approx$ is weak bisimulation.

Previously, on Resource Calculus – 2003

- Ehrhard and Regnier's differential λ-calculus.

$$M, N ::= x \mid \lambda x.M \mid M\mu \mid \mathsf{D}\, M \cdot N \qquad \text{(terms)}$$
$$\mu, \nu ::= M_1 + \cdots + M_k \qquad \text{(sums)}$$

Derivative of M along N

Q: what is the derivative of a function?
A: the best linear aproximation.

- Linear $\equiv$ using input exactly once $\rightsquigarrow$ ephemeral inputs!
- Calculus is non lazy and confluent.

Thomas Ehrhard and Laurent Regnier.
The differential lambda-calculus.
Theor. Comput. Sci., 309(1):1–41, 2003.

Previously, on Resource Calculus – 2003

- Ehrhard and Regnier's differential λ-calculus.

$$M, N ::= x \mid \lambda x.M \mid M\mu \mid \mathrm{D}\, M \cdot N \qquad \text{(terms)}$$
$$\mu, \nu ::= M_1 + \cdots + M_k \qquad \text{(sums)}$$

Q: what is the derivative of a function?
A: the best linear aproximation.

- Linear $\equiv$ using input exactly once $\rightsquigarrow$ ephemeral inputs!
- Calculus is non lazy and confluent.

Thomas Ehrhard and Laurent Regnier.
The differential lambda-calculus.
Theor. Comput. Sci., 309(1):1–41, 2003.

Previously, on Resource Calculus – 2003

- Ehrhard and Regnier's differential λ-calculus.

$$M, N ::= x \mid \lambda x.M \mid M\mu \mid \mathrm{D}\, M \cdot N \qquad \text{(terms)}$$
$$\mu, \nu ::= M_1 + \cdots + M_k \qquad \text{(sums)}$$

Q: what is the derivative of a function?
A: the best linear aproximation.

- Linear $\equiv$ using input exactly once $\rightsquigarrow$ ephemeral inputs!
- Calculus is non lazy and confluent.

Thomas Ehrhard and Laurent Regnier.
The differential lambda-calculus.
Theor. Comput. Sci., 309(1):1–41, 2003.

Previously, on Resource Calculus – 2003

- Ehrhard and Regnier's differential λ-calculus.

$$M, N ::= x \mid \lambda x.M \mid M\mu \mid \mathrm{D}\, M \cdot N \qquad \text{(terms)}$$
$$\mu, \nu ::= M_1 + \cdots + M_k \qquad \text{(sums)}$$

Q: what is the derivative of a program?
A: the best linear aproximation.

- Linear $\equiv$ using input exactly once $\rightsquigarrow$ ephemeral inputs!
- Calculus is non lazy and confluent.

Thomas Ehrhard and Laurent Regnier.
The differential lambda-calculus.
Theor. Comput. Sci., 309(1):1–41, 2003.

Previously, on Resource Calculus – 2003

- Ehrhard and Regnier's differential λ-calculus.

$$
\begin{aligned}
M, N &::= x \mid \lambda x.M \mid M\mu \mid \mathrm{D}\, M \cdot N & \text{(terms)}\\
\mu, \nu &::= M_1 + \cdots + M_k & \text{(sums)}
\end{aligned}
$$

Q: what is the derivative of a program?
A: the best linear aproximation.

- Linear $\equiv$ using input exactly once $\rightsquigarrow$ ephemeral inputs!
- Calculus is non lazy and confluent.

Thomas Ehrhard and Laurent Regnier.
The differential lambda-calculus.
Theor. Comput. Sci., 309(1):1–41, 2003.

Previously, on Resource Calculus – 2003

- Ehrhard and Regnier's differential λ-calculus.

$$M, N ::= x \mid \lambda x.M \mid M\mu \mid \mathrm{D}\, M \cdot N \qquad \text{(terms)}$$
$$\mu, \nu ::= M_1 + \cdots + M_k \qquad \text{(sums)}$$

Q: what is the derivative of a program?
A: the best linear aproximation.

- Linear $\equiv$ using input exactly once $\rightsquigarrow$ ephemeral inputs!
- Calculus is non lazy and confluent.

Thomas Ehrhard and Laurent Regnier.
The differential lambda-calculus.
Theor. Comput. Sci., 309(1):1–41, 2003.

Previously, on Resource Calculus – 2006

- Finite resource calculus (no perpetual arguments).

$$\begin{array}{rll} M, N ::= & x \mid \lambda x.M \mid MP & \text{(terms)} \\ \mu, \nu ::= & M_1 + \cdots + M_k & \text{(sums)} \\ P, Q ::= & [M_1, \ldots, M_k] & \text{(bags)} \end{array}$$

$$(\lambda x.M)[N_1, \ldots, N_k] \to M\langle\langle N_1/x\rangle\rangle \cdots \langle\langle N_k/x\rangle\rangle\{0/x\}$$

- $M\langle\langle N/x\rangle\rangle$: N in one occurrence of x, nondeterministically.
- Order of substitutions doesn't matter (Schwarz lemma!).
- Starvation: $x \in M \implies M\{0/x\} = 0$.
- Surplus not allowed: $x \notin M : M\langle\langle N/x\rangle\rangle = 0$.
- Always terminating, trivially confluent.

Thomas Ehrhard and Laurent Regnier.
Differential interaction nets.
Theor. Comput. Sci., 364(2):166–195, 2006.

Previously, on Resource Calculus – 2006

- Finite resource calculus (no perpetual arguments).

$$\begin{array}{rll} M, N ::= & x \mid \lambda x.M \mid MP & \text{(terms)} \\ \mu, \nu ::= & M_1 + \cdots + M_k & \text{(sums)} \\ P, Q ::= & [M_1, \ldots, M_k] & \text{(bags)} \end{array}$$

$$(\lambda x.M)[N_1, \ldots, N_k] \to M\langle\langle N_1/x\rangle\rangle \cdots \langle\langle N_k/x\rangle\rangle\{0/x\}$$

- $M\langle\langle N/x\rangle\rangle$: N in one occurrence of x, nondeterministically.
- Order of substitutions doesn't matter (Schwarz lemma!).
- Starvation: $x \in M \implies M\{0/x\} = 0$.
- Surplus not allowed: $x \notin M : M\langle\langle N/x\rangle\rangle = 0$.
- Always terminating, trivially confluent.

summing all possibilities

Thomas Ehrhard and Laurent Regnier.
Differential interaction nets.
Theor. Comput. Sci., 364(2):166–195, 2006.

Previously, on Resource Calculus – 2006

- Finite resource calculus (no perpetual arguments).

$$
\begin{array}{rll}
M, N ::= & x \mid \lambda x.M \mid MP & \text{(terms)} \\
\mu, \nu ::= & M_1 + \cdots + M_k & \text{(sums)} \\
P, Q ::= & [M_1, \ldots, M_k] & \text{(bags)}
\end{array}
$$

$$(\lambda x.M)[N_1, \ldots, N_k] \to M\langle\langle N_1/x\rangle\rangle \cdots \langle\langle N_k/x\rangle\rangle\{0/x\}$$

- $M\langle\langle N/x\rangle\rangle$: N in one occurrence of x, nondeterministically.
- Order of substitutions doesn't matter (Schwarz lemma!).
- Starvation: $x \in M \implies M\{0/x\} = 0$.
- Surplus not allowed: $x \notin M$, $M\langle\langle N/x\rangle\rangle = 0$.
- Always terminating, trivially confluent.

nondeterministic dead end

Thomas Ehrhard and Laurent Regnier.
Differential interaction nets.
Theor. Comput. Sci., 364(2):166–195, 2006.

Just for teasing: Taylor expanding ordinary terms

- Finite resource calculus is the target of the Taylor expansion of ordinary λ-terms.

$$x^* := x \qquad (\lambda x.M)^*$$

$$(MN)^* = \sum_{n=0}^{\infty} \frac{1}{n!} M^* [N^*]^k$$

- Potentially infinite $\rightsquigarrow$ infinite sum of finite elements.
- Term terminating $\rightsquigarrow$ non-zero finite terms give exact argument usage.

Thomas Ehrhard and Laurent Regnier.
Böhm trees, Krivine's machine and the Taylor expansion of lambda-terms.
In *CiE*, volume 3988 of *LNCS*, pages 186–197. Springer, 2006.

Outline

Resource Calculus, now

- Non lazy and non finite:

$$\begin{array}{rlr} M, N ::= & x \mid \lambda x.M \mid M P & \text{(terms)} \\ \mu, \nu ::= & M_1 + \cdots + M_k & \text{(sums)} \\ P, Q ::= & [N_1, \ldots, N_k] & \text{(bags)} \end{array}$$

$$(\lambda x.M)[M_1, \ldots, M_h, N_1^!, \ldots, N_k^!]$$
$$\downarrow$$
$$M\langle\langle M_1/x\rangle\rangle \cdots \langle\langle M_h/x\rangle\rangle\langle\langle N_1^!/x\rangle\rangle \cdots \langle\langle N_k^!/x\rangle\rangle\{0/x\}$$

- $M\langle\langle N^!/x\rangle\rangle = M\{N + x/x\}$
- However $M\langle\langle M/x\rangle\rangle$ as before does not work: not all occurrences of x are linear!

Depleting resources

- Linear substitution by inductive definition:

$$y\langle\langle N/x\rangle\rangle := \begin{cases} N & \text{if } y = x, \\ 0 & \text{otherwise,} \end{cases}$$

$$(MP)\langle\langle N/x\rangle\rangle := M\langle\langle N/x\rangle\rangle P + M(P\langle\langle N/x\rangle\rangle),$$

$$(P \cdot R)\langle\langle N/x\rangle\rangle := P\langle\langle N/x\rangle\rangle \cdot R + P \cdot R\langle\langle N/x\rangle\rangle.$$

$$[M]\langle\langle N/x\rangle\rangle := [M\langle\langle N/x\rangle\rangle],$$

$$[M^!]\langle\langle N/x\rangle\rangle := [M\langle\langle N/x\rangle\rangle, M^!],$$

Something familiar…

$$(P \cdot R)\langle\langle N/x\rangle\rangle = P\langle\langle N/x\rangle\rangle \cdot R + P \cdot R\langle\langle N/x\rangle\rangle,$$

$$[M^!]\langle\langle N/x\rangle\rangle = [M\langle\langle N/x\rangle\rangle] \cdot [M^!].$$

Depleting resources

- Linear substitution by inductive definition:

$$
\begin{aligned}
y\langle\langle N/x\rangle\rangle &:= \begin{cases} N & \text{if } y = x, \\ 0 & \text{otherwise,} \end{cases} \\
(MP)\langle\langle N/x\rangle\rangle &:= M\langle\langle N/x\rangle\rangle P + M(P\langle\langle N/x\rangle\rangle), \\
(P \cdot R)\langle\langle N/x\rangle\rangle &:= P\langle\langle N/x\rangle\rangle \cdot R + P \cdot R\langle\langle N/x\rangle\rangle. \\
[M]\langle\langle N/x\rangle\rangle &:= [M\langle\langle N/x\rangle\rangle], \\
[M^!]\langle\langle N/x\rangle\rangle &:= [M\langle\langle N/x\rangle\rangle, M^!],
\end{aligned}
$$

Something familiar…

$$(P \cdot R)\langle\langle N/x\rangle\rangle = P\langle\langle N/x\rangle\rangle \cdot R + P \cdot R\langle\langle N/x\rangle\rangle,$$

$$[M^!]\langle\langle N/x\rangle\rangle = [M\langle\langle N/x\rangle\rangle] \cdot [M^!].$$

Depleting resources

- Linear substitution by inductive definition:

$$
\begin{aligned}
y\langle\langle N/x\rangle\rangle &:= \begin{cases} N & \text{if } y = x, \\ 0 & \text{otherwise,} \end{cases} \\
(MP)\langle\langle N/x\rangle\rangle &:= M\langle\langle N/x\rangle\rangle P + M(P\langle\langle N/x\rangle\rangle), \\
(P \cdot R)\langle\langle N/x\rangle\rangle &:= P\langle\langle N/x\rangle\rangle \cdot R + P \cdot R\langle\langle N/x\rangle\rangle. \\
[M]\langle\langle N/x\rangle\rangle &:= [M\langle\langle N/x\rangle\rangle], \\
[M^!]\langle\langle N/x\rangle\rangle &:= [M\langle\langle N/x\rangle\rangle, M^!],
\end{aligned}
$$

Something familiar…

$$
\begin{aligned}
\frac{\partial u \cdot v}{\partial x} &= \frac{\partial u}{\partial x} \cdot v + u \cdot \frac{\partial v}{\partial x} \qquad , \\
\frac{\partial e^u}{\partial x} &= \frac{\partial u}{\partial x} \cdot e^u \qquad .
\end{aligned}
$$

Oh my, sums everywhere?

Fortunately, not: sums are pushed to surface only.

$$\lambda x.(M+N) = \lambda x.M + \lambda x.N$$
$$(M+N)P = MP + NP$$
$$M([N+L]\cdot P) = M([N]\cdot P) + M([L]\cdot P)$$
$$M([(N+L)^!]\cdot P) = M([N^!, M^!]\cdot P)$$

...and the zeroary versions

$$\lambda x.0 = 0 \qquad M([0]\cdot P) = 0$$
$$0P = 0 \qquad M([0^!]\cdot P) = MP$$

Again, something familiar...

$$[(M+N)^!] = [M^!]\cdot[N^!], \qquad [0^!] = 1$$

Oh my, sums everywhere?

Fortunately, not: sums are pushed to surface only.

$$\lambda x.(M+N) = \lambda x.M + \lambda x.N$$
$$(M+N)P = MP + NP$$
$$M([N+L]\cdot P) = M([N]\cdot P) + M([L]\cdot P)$$
$$M([(N+L)^!]\cdot P) = M([N^!, M^!]\cdot P)$$

...and the zeroary versions

$$\lambda x.0 = 0 \qquad M([0]\cdot P) = 0$$
$$0P = 0 \qquad M([0^!]\cdot P) = MP$$

Again, something familiar...

$$[(M+N)^!] = [M^!]\cdot[N^!], \qquad [0^!] = 1$$

Oh my, sums everywhere?

Fortunately, not: sums are pushed to surface only.

$$\lambda x.(M+N) = \lambda x.M + \lambda x.N$$
$$(M+N)P = MP + NP$$
$$M([N+L]\cdot P) = M([N]\cdot P) + M([L]\cdot P)$$
$$M([(N+L)^!]\cdot P) = M([N^!, M^!]\cdot P)$$

...and the zeroary versions

$$\lambda x.0 = 0 \qquad M([0]\cdot P) = 0$$
$$0P = 0 \qquad M([0^!]\cdot P) = MP$$

Again, something familiar...

$$\mathrm{e}^{a+b} = \mathrm{e}^a \cdot \mathrm{e}^b \quad , \quad \mathrm{e}^0 = 1$$

Summing up

- So reduction is:

$$(\lambda x.M)[M_1, \dots, M_h, N_1^!, \dots, N_k^!]$$
$$\downarrow$$
$$M\langle\langle M_1/x\rangle\rangle \cdots \langle\langle M_h/x\rangle\rangle \langle\langle N_1^!/x\rangle\rangle \cdots \langle\langle N_k^!/x\rangle\rangle \{0/x\}$$
$$\|$$
$$M\langle\langle M_1/x\rangle\rangle \cdots \langle\langle M_h/x\rangle\rangle \{\textstyle\sum_i N_i/x\}$$

- A generalized Schwarz lemma assures order is irrelevant.
- Formally, we consider sums to reduce one addend at a time...

Summing up

- So reduction is:

$$(\lambda x.M)[M_1,\dots,M_h,N_1^!,\dots,N_k^!]$$
$$\downarrow$$
$$M\langle\!\langle M_1/x\rangle\!\rangle\cdots\langle\!\langle M_h/x\rangle\!\rangle\langle\!\langle N_1^!/x\rangle\!\rangle\cdots\langle\!\langle N_k^!/x\rangle\!\rangle\{0/x\}$$
$$||$$
$$M\langle\!\langle M_1/x\rangle\!\rangle\cdots\langle\!\langle M_h/x\rangle\!\rangle\{\textstyle\sum_i N_i/x\}$$

- A generalized Schwarz lemma assures order is irrelevant.
- Formally, we consider sums to reduce one addend at a time...

Some examples!

- $(\lambda x.x[x^!])[M, N^!] \to x[x^!]\langle\langle M/x\rangle\rangle\{N/x\} = M[N^!] + N[M, N^!]$
- Fixed point operator $Y \to \lambda f.f(Yf)$ (using $MN = M[N^!]$)
- $Y[F^!, G^!] \to F[(Y[F^!, G^!])^!] + G[(Y[F^!, G^!])^!]$
- $Y[(\lambda d.\underline{0})^!, \mathtt{succ}^!] \to \underline{0} + \underline{1} + \underline{2} + \cdots$
- $$\cong \lambda k. \begin{cases} k! & \text{if } k \leq h \\ 0 & \text{(nondeterministic dead end) otherwise} \end{cases}$$

Some examples!

- $(\lambda x.x[x^!])[M, N^!] \to x[x^!]\langle\!\langle M/x \rangle\!\rangle\{N/x\} = M[N^!] + N[M, N^!]$
- Fixed point operator $Y \to \lambda f.f(Yf)$ (using $MN = M[N^!]$)
  - $Y[F^!, G^!] \to F[(Y[F^!, G^!])^!] + G[(Y[F^!, G^!])^!]$
  - $Y[(\lambda d.\underline{0})^!, \mathtt{succ}^!] \to \underline{0} + \underline{1} + \underline{2} + \cdots$
  - $\cong \lambda k. \begin{cases} k! & \text{if } k \leq h \\ 0 & \text{(nondeterministic dead end) otherwise} \end{cases}$

Some examples!

- $(\lambda x.x[x^!])[M, N^!] \to x[x^!]\langle\langle M/x\rangle\rangle\{N/x\} = M[N^!] + N[M, N^!]$
- Fixed point operator $Y \to \lambda f.f(Yf)$ (using $MN = M[N^!]$)
  - $Y[F^!, G^!] \to F[(Y[F^!, G^!])^!] + G[(Y[F^!, G^!])^!]$
  - $Y[(\lambda d.\underline{0})^!, \mathtt{succ}^!] \to \underline{0} + \underline{1} + \underline{2} + \cdots$
  -

$$\cong \lambda k. \begin{cases} k! & \text{if } k \leq h \\ 0 & \text{(nondeterministic dead end) otherwise} \end{cases}$$

Some examples!

- $(\lambda x.x[x^!])[M, N^!] \to x[x^!]\langle\!\langle M/x\rangle\!\rangle\{N/x\} = M[N^!] + N[M, N^!]$
- Fixed point operator $Y \to \lambda f.f(Yf)$ (using $MN = M[N^!]$)
  - $Y[F^!, G^!] \to F[(Y[F^!, G^!])^!] + G[(Y[F^!, G^!])^!]$
  - $Y[(\lambda d.\underline{0})^!, \mathtt{succ}^!] \to \underline{0} + \underline{1} + \underline{2} + \cdots$

$$\cong \lambda k. \begin{cases} k! & \text{if } k \leq h \\ 0 & \text{(nondeterministic dead end) otherwise} \end{cases}$$

Some examples!

- $(\lambda x.x[x^!])[M, N^!] \to x[x^!]\langle\langle M/x\rangle\rangle\{N/x\} = M[N^!] + N[M, N^!]$
- Fixed point operator $Y \to \lambda f.f(Yf)$ (using $MN = M[N^!]$)
  - $Y[F^!, G^!] \to F[(Y[F^!, G^!])^!] + G[(Y[F^!, G^!])^!]$
  - $Y[(\lambda d.\underline{0})^!, \mathtt{succ}^!] \to \underline{0} + \underline{1} + \underline{2} + \cdots$
  -

$$\cong \lambda k. \begin{cases} k! & \text{if } k \leq h \\ 0 & \text{(nondeterministic dead end) otherwise} \end{cases}$$

Some examples!

- $(\lambda x.x[x^!])[M, N^!] \to x[x^!]\langle\langle M/x\rangle\rangle\{N/x\} = M[N^!] + N[M, N^!]$
- Fixed point operator $Y \to \lambda f.f(Yf)$ (using $MN = M[N^!]$)
  - $Y[F^!, G^!] \to F[(Y[F^!, G^!])^!] + G[(Y[F^!, G^!])^!]$
  - $Y[(\lambda d.\underline{0})^!, \mathtt{succ}^!] \to \underline{0} + \underline{1} + \underline{2} + \cdots$
  - $Y[\lambda g \lambda k.\mathtt{iszero}\, k\, \underline{1}\, (\mathtt{mult}\, (g\, (\mathtt{pred}\, k))\, k)]^h$

$$\cong \lambda k. \begin{cases} k! & \text{if } k \leq h \\ 0 & \text{(nondeterministic dead end) otherwise} \end{cases}$$

Some examples!

- $(\lambda x.x[x^!])[M, N^!] \to x[x^!]\langle\langle M/x\rangle\rangle\{N/x\} = M[N^!] + N[M, N^!]$
- Fixed point operator $Y \to \lambda f.f(Yf)$ (using $MN = M[N^!]$)
  - $Y[F^!, G^!] \to F[(Y[F^!, G^!])^!] + G[(Y[F^!, G^!])^!]$
  - $Y[(\lambda d.\underline{0})^!, \mathtt{succ}^!] \to \underline{0} + \underline{1} + \underline{2} + \cdots$
  - $Y[\lambda g\lambda k.\mathtt{if}\ k = \underline{0}\ \mathtt{then}\ \underline{1}\ \mathtt{else}\ (g\,(\underline{k-1}))\times k]^h$

$$\cong \lambda k.\begin{cases} k! & \text{if } k \leq h \\ 0 & \text{(nondeterministic dead end) otherwise} \end{cases}$$

Some examples!

- $(\lambda x.x[x^!])[M, N^!] \to x[x^!]\langle\langle M/x\rangle\rangle\{N/x\} = M[N^!] + N[M, N^!]$
- Fixed point operator $Y \to \lambda f.f(Yf)$ (using $MN = M[N^!]$)
  - $Y[F^!, G^!] \to F[(Y[F^!, G^!])^!] + G[(Y[F^!, G^!])^!]$
  - $Y[(\lambda d.\underline{0})^!, \texttt{succ}^!] \to \underline{0} + \underline{1} + \underline{2} + \cdots$
  - $Y[\lambda g \lambda k.\texttt{if } k = \underline{0} \texttt{ then } \underline{1} \texttt{ else } (g\,(\underline{k-1})) \times k]^h$

$$\cong \lambda k. \begin{cases} k! & \text{if } k \leq h \\ 0 & \text{(nondeterministic dead end) otherwise} \end{cases}$$

Some examples!

- $(\lambda x.x[x^!])[M, N^!] \to x[x^!]\langle\langle M/x\rangle\rangle\{N/x\} = M[N^!] + N[M, N^!]$
- Fixed point operator $Y \to \lambda f.f(Yf)$ (using $MN = M[N^!]$)
  - $Y[F^!, G^!] \to F[(Y[F^!, G^!])^!] + G[(Y[F^!, G^!])^!]$
  - $Y[(\lambda d.\underline{0})^!, \mathtt{succ}^!] \to \underline{0} + \underline{1} + \underline{2} + \cdots$
  - $Y[\lambda g \lambda k.\mathtt{if}\ k = \underline{0}\ \mathtt{then}\ \underline{1}\ \mathtt{else}\ (g\,(\underline{k-1}))\times k]^h$

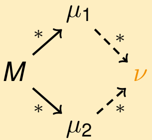
$$\cong \lambda k. \begin{cases} k! & \text{if } k \leq h \\ 0 & \text{(nondeterministic dead end) otherwise} \end{cases}$$

Outline

Confluence

Theorem

Reduction is confluent.



i.e. nondeterminism modeled is really internal (in non lazy setting).

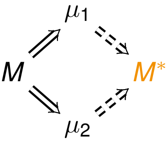
Sketch of the proof

- Standard Tait-Martin Löf technique.
- Parallel reductions

$$\frac{}{x \Rightarrow x}\ \texttt{var} \qquad \frac{M \Rightarrow \mu \qquad P \Rightarrow \pi}{MP \Rightarrow \mu\pi}\ \overline{@} \qquad \frac{M \Rightarrow \mu}{\lambda x.M \Rightarrow \lambda x.\mu}\ \overline{\lambda}$$

$$\frac{}{1 \Rightarrow 1}\ \texttt{bag1} \qquad \frac{M \Rightarrow \mu \qquad P \Rightarrow \pi}{[M] \cdot P \Rightarrow [\mu] \cdot \pi}\ \overline{\texttt{bag}\ell} \qquad \frac{M \Rightarrow \mu \qquad P \Rightarrow \pi}{[M^!] \cdot P \Rightarrow [\mu^!] \cdot \pi}\ \overline{\texttt{bag!}}$$

$$\frac{A_i \Rightarrow \alpha_i, \qquad \text{for } 1 \leq i \leq k}{\sum_{i=1}^k A_i \Rightarrow \sum_{i=1}^k \alpha_i}\ \overline{\texttt{sum}} \qquad \frac{M \Rightarrow \mu \qquad P \Rightarrow \pi}{(\lambda x.M)P \Rightarrow \mu\langle\langle \pi/x \rangle\rangle\{0/x\}}\ \overline{\texttt{g}}$$

- Development M^*: $(MP)^* = \begin{cases} M^* P^* & \text{if } M \neq \lambda x.N \\ N^*\langle\langle P^*/x \rangle\rangle\{0/x\} & \text{if } M = \lambda x.N \end{cases}$
- $\Rightarrow$ strongly confluent: 

, so $\rightarrow$ confluent.

Standardization

- In λ-calculus lazy and non-lazy is linked by standardization.
- Roughly, if $M \xrightarrow{*} N$ then you can start by a head ($\cong$ lazy) reduction.
- In resource calculus, place taken by $\xrightarrow{\circ}$, outer reduction, which does not reduce inside $M^!$ (and $\xrightarrow{\mathtt{i}}$ is inner reduction).

Theorem (Standardization)

If $M \xrightarrow{*} \mu$, then there is ν with $M \xrightarrow{\circ *} \nu \xrightarrow{\mathtt{i}*} \mu$.

- Proof with outer and inner variants of parallel reduction, adapting

Takahashi, M.:
Parallel reductions in lambda-calculus.
Information and Computation **118**(1) (April 1995) 120–127

So what?

- Resource Calculus gives an abstract and mathematic account of nondeterminism and (arguably) resource boundedness.
  - Obstacle: exact usage (no surplus). However we can set affine (i.e. usable at most once) resources as $[M] + 1$.
  - In any case, no clear account of must correctness yet.
  - Can we use this framework effectively?
- Non-lazy over lazy premits reasoning on optimal reductions. Has it sense here?
  - danger: reductions can duplicate the context, $[M^!] \to [M^!_1, M^!_2]\ldots$
- Typing: simple types and second order give indeed termination.
- Curry-Howard: link with differential linear logic.

THANKS

So what?

- Resource Calculus gives an abstract and mathematic account of nondeterminism and (arguably) resource boundedness.
  - Obstacle: exact usage (no surplus). However we can set affine (i.e. usable at most once) resources as $[M] + 1$.
  - In any case, no clear account of must correctness yet.
  - Can we use this framework effectively?
- Non-lazy over lazy premits reasoning on optimal reductions. Has it sense here?
  - danger: reductions can duplicate the context, $[M^!] \to [M_1^!, M_2^!] \ldots$
- Typing: simple types and second order give indeed termination.
- Curry-Howard: link with differential linear logic.

THANKS