An output-based semantics of $\Lambda\mu$ with explicit substitution in the π-calculus

Steffen van Bakel and Maria Grazia Vigliotti

Department of Computing,
Imperial College London

Summary

- We focus on output-based interpretation of λ, $\lambda\mathbf{x}$, $\lambda\mu$, $\Lambda\mu$, and $\Lambda\mu\mathbf{x}$ into the asynchronous π-calculus – enriched with pairing – that have their origin in mathematical logic.

Summary

- We focus on output-based interpretation of λ, $\lambda\mathbf{x}$, $\lambda\mu$, $\Lambda\mu$, and $\Lambda\mu\mathbf{x}$ into the asynchronous π-calculus – enriched with pairing – that have their origin in mathematical logic.
- We will show that it is possible to improve on all results that can be shown for Milner's encoding for lazy λ.

Summary

- We focus on output-based interpretation of λ, $\lambda\mathbf{x}$, $\lambda\mu$, $\Lambda\mu$, and $\Lambda\mu\mathbf{x}$ into the asynchronous π-calculus – enriched with pairing – that have their origin in mathematical logic.
- We will show that it is possible to improve on all results that can be shown for Milner's encoding for lazy λ.
- We show that our interpretation respects one-step reduction.

Summary

- We focus on output-based interpretation of λ, $\lambda\mathbf{x}$, $\lambda\mu$, $\Lambda\mu$, and $\Lambda\mu\mathbf{x}$ into the asynchronous π-calculus – enriched with pairing – that have their origin in mathematical logic.
- We will show that it is possible to improve on all results that can be shown for Milner's encoding for lazy λ.
- We show that our interpretation respects one-step reduction.
- Using a notion of type assignment for the π-calculus that uses the type constructor $\rightarrow$, we show that all Curry-assignable types to λ and $\Lambda\mu$-terms are preserved by the interpretations.

Summary

- We focus on output-based interpretation of λ, $\lambda\mathbf{x}$, $\lambda\mu$, $\Lambda\mu$, and $\Lambda\mu\mathbf{x}$ into the asynchronous π-calculus – enriched with pairing – that have their origin in mathematical logic.
- We will show that it is possible to improve on all results that can be shown for Milner's encoding for lazy λ.
- We show that our interpretation respects one-step reduction.
- Using a notion of type assignment for the π-calculus that uses the type constructor $\rightarrow$, we show that all Curry-assignable types to λ and $\Lambda\mu$-terms are preserved by the interpretations.
- We will also show termination results.

Bloo and Rose's λx

We need the explicit λ-calculus; the syntax is an extension of that of the λ-calculus:

$$M, N \ ::= \ x \mid \lambda x.M \mid MN \mid M \langle x := N \rangle$$

($\langle x := N \rangle$ is the *explicit* substitution.)

Bloo and Rose's $\lambda\mathbf{x}$

We need the explicit λ-calculus; the syntax is an extension of that of the λ-calculus:

$$M, N \ ::= \ x \mid \lambda x.M \mid MN \mid M\langle x := N\rangle$$

($\langle x := N\rangle$ is the *explicit* substitution.) Reduction now becomes *term rewriting*: the explicit variant $\rightarrow_{\mathbf{x}}$ of β-reduction is defined by:

$$\begin{array}{rcl} (\lambda x.M)N & \rightarrow & M\langle x := N\rangle \\ (MN)\langle x := L\rangle & \rightarrow & M\langle x := L\rangle N\langle x := L\rangle \\ (\lambda y.M)\langle x := L\rangle & \rightarrow & \lambda y.(M\langle x := L\rangle) \\ x\langle x := L\rangle & \rightarrow & L \\ M\langle x := L\rangle & \rightarrow & M, \quad (x \notin fv(M)) \end{array}$$

$$M \rightarrow N \ \Rightarrow \left\{ \begin{array}{lcl} ML & \rightarrow & NL \\ LM & \rightarrow & LN \\ \lambda x.M & \rightarrow & \lambda x.N \\ M\langle x := L\rangle & \rightarrow & N\langle x := L\rangle \\ L\langle x := M\rangle & \rightarrow & L\langle x := N\rangle \end{array} \right.$$

Notice that we can reduce *inside* the substitution.

Explicit lazy and head reduction

We define *explicit head reduction* $\rightarrow_{\mathbf{x}\mathrm{H}}$ on $\lambda\mathbf{x}$ as $\rightarrow_{\mathbf{x}}$, but for:

1. We change one case: $(PQ)\langle x := N\rangle \;\rightarrow\; (P\langle x := N\rangle\, Q)\langle x := N\rangle$

Explicit lazy and head reduction

We define *explicit head reduction* $\rightarrow_{\mathbf{x}\mathsf{H}}$ on $\lambda\mathbf{x}$ as $\rightarrow_{\mathbf{x}}$, but for:

1. We change one case: $(PQ)\langle x := N\rangle \;\rightarrow\; (P\langle x := N\rangle\, Q)\langle x := N\rangle$
2. We add the substitution rule:

   $M\langle x := N\rangle\langle y := L\rangle \;\rightarrow\; M\langle y := L\rangle\langle x := N\rangle\langle y := L\rangle$

Explicit lazy and head reduction

We define *explicit head reduction* $\rightarrow_{\mathbf{x}\mathsf{H}}$ on $\lambda\mathbf{x}$ as $\rightarrow_{\mathbf{x}}$, but for:

1. We change one case: $(PQ)\langle x := N\rangle \ \rightarrow\ (P\langle x := N\rangle\, Q)\langle x := N\rangle$
2. We add the substitution rule:
   $$M\langle x := N\rangle\langle y := L\rangle \ \rightarrow\ M\langle y := L\rangle\langle x := N\rangle\langle y := L\rangle$$
3. These rules are only applied if *hv*$(M) = x$; the exception is garbage collection.

Explicit lazy and head reduction

We define *explicit head reduction* $\rightarrow_{\mathbf{x}\mathsf{H}}$ on $\lambda\mathbf{x}$ as $\rightarrow_{\mathbf{x}}$, but for:

1. We change one case: $(PQ)\langle x := N\rangle \;\rightarrow\; (P\langle x := N\rangle\, Q)\langle x := N\rangle$
2. We add the substitution rule:
   $$M\langle x := N\rangle\langle y := L\rangle \;\rightarrow\; M\langle y := L\rangle\langle x := N\rangle\langle y := L\rangle$$
3. These rules are only applied if *hv*$(M) = x$; the exception is garbage collection.
4. We remove
   $$M \rightarrow N \;\Rightarrow\; \begin{cases} LM & \rightarrow\; LN \\ L\langle x := M\rangle & \rightarrow\; L\langle x := N\rangle \end{cases}$$

Explicit lazy and head reduction

We define *explicit head reduction* $\rightarrow_{\mathbf{x}\mathsf{H}}$ on $\lambda\mathbf{x}$ as $\rightarrow_{\mathbf{x}}$, but for:

1. We change one case: $(PQ)\langle x := N\rangle \rightarrow (P\langle x := N\rangle\, Q)\langle x := N\rangle$
2. We add the substitution rule:
$$M\langle x := N\rangle\langle y := L\rangle \rightarrow M\langle y := L\rangle\langle x := N\rangle\langle y := L\rangle$$
3. These rules are only applied if *hv*$(M) = x$; the exception is garbage collection.
4. We remove
$$M \rightarrow N \Rightarrow \begin{cases} LM & \rightarrow LN \\ L\langle x := M\rangle & \rightarrow L\langle x := N\rangle \end{cases}$$
5. Note that *explicit lazy reduction* $\rightarrow_{\mathbf{x}\mathsf{L}}$ is defined by removing
$$(\lambda y.M)\langle x := L\rangle \rightarrow \lambda y.(M\langle x := L\rangle)$$
$$M \rightarrow N \Rightarrow \lambda x.M \rightarrow \lambda x.N$$

Explicit lazy and head reduction

We define *explicit head reduction* $\rightarrow_{\mathbf{x}\mathsf{H}}$ on $\lambda\mathbf{x}$ as $\rightarrow_{\mathbf{x}}$, but for:

1. We change one case: $(PQ)\langle x := N\rangle \;\rightarrow\; (P\langle x := N\rangle\, Q)\langle x := N\rangle$
2. We add the substitution rule:
$$M\langle x := N\rangle\langle y := L\rangle \;\rightarrow\; M\langle y := L\rangle\langle x := N\rangle\langle y := L\rangle$$
3. These rules are only applied if *hv*$(M) = x$; the exception is garbage collection.
4. We remove
$$M \rightarrow N \;\Rightarrow\; \begin{cases} LM & \rightarrow\; LN \\ L\langle x := M\rangle & \rightarrow\; L\langle x := N\rangle \end{cases}$$
5. Note that *explicit lazy reduction* $\rightarrow_{\mathbf{x}\mathsf{L}}$ is defined by removing
$$(\lambda y.M)\langle x := L\rangle \;\rightarrow\; \lambda y.(M\langle x := L\rangle)$$
$$M \rightarrow N \;\Rightarrow\; \lambda x.M \rightarrow \lambda x.N$$

In both, only the *head*-variable will ever get replaced. Explicit lazy reduction corresponds to Krivine's machine.

Parigot's $\lambda\mu$

Terms: $M,N \ ::= \ x \mid \lambda x.M \mid MN \mid \mu\alpha.[\beta]M$

Type assignment: $A,B \ ::= \ \varphi \mid A{\rightarrow}B$

$$(Ax): \frac{}{\Gamma, x{:}A \vdash x : A \mid \Delta} \qquad (\mu): \frac{\Gamma \vdash M : B \mid \alpha{:}A, \beta{:}B, \Delta}{\Gamma \vdash \mu\alpha.[\beta]M : A \mid \Delta}$$

$$(\rightarrow I): \frac{\Gamma, x{:}A \vdash M : B \mid \Delta}{\Gamma \vdash \lambda x.M : A{\rightarrow}B \mid \Delta} \qquad (\rightarrow E): \frac{\Gamma \vdash M : A{\rightarrow}B \mid \Delta \quad \Gamma \vdash N : A \mid \Delta}{\Gamma \vdash MN : B \mid \Delta}$$

Reduction:

logical : $(\lambda x.M)N \ \rightarrow \ M[N/x]$

structural : $(\mu\alpha.[\beta]M)N \ \rightarrow \ \mu\gamma.([\beta]M[N{\cdot}\gamma/\alpha])$

where $([\alpha]M)[N{\cdot}\gamma/\alpha] \triangleq [\gamma](M\,[N{\cdot}\gamma/\alpha])N$.

De Groote defines $\Lambda\mu$ by splitting rule (μ) into

$$(\mu): \frac{\Gamma \vdash M : \bot \mid \alpha{:}A, \Delta}{\Gamma \vdash \mu\alpha.M : A \mid \Delta} \qquad (\bot): \frac{\Gamma \vdash M : A \mid \beta{:}A, \Delta}{\Gamma \vdash [\beta]M : \bot \mid \beta{:}A, \Delta}$$

$\Lambda\mu\mathbf{x}$

We make both substitutions explicit:

$$M, N \; ::= \; x \mid \lambda x.M \mid MN \mid M \langle x := N \rangle \mid \\ \mu\alpha.M \mid [\beta]M \mid M \langle \alpha := N{\cdot}\gamma \rangle$$

Reduction rules change accordingly. We add the rules

$$(\textit{T-cut}) : \frac{\Gamma, x{:}A \vdash M : B \mid \Delta \quad \Gamma \vdash N : A \mid \Delta}{\Gamma \vdash M \langle x := N \rangle : B \mid \Delta}$$

$$(\textit{C-cut}) : \frac{\Gamma \vdash M : C \mid \alpha{:}A{\rightarrow}B, \Delta \quad \Gamma \vdash N : A \mid \gamma{:}B, \Delta}{\Gamma \vdash M \langle \alpha := N{\cdot}\gamma \rangle : C \mid \Delta}$$

Explicit head reduction for $\Lambda\mu\mathbf{x}$

We define *explicit head reduction* $\rightarrow_{\mathbf{x}\mathrm{H}}$ on $\Lambda\mu\mathbf{x}$ as $\rightarrow_{\mathbf{x}}$, but for:

1. $(PQ)\langle x := N\rangle \rightarrow (P\langle x := N\rangle\, Q)\langle x := N\rangle$
   $(PQ)\langle \alpha := N{\cdot}\gamma\rangle \rightarrow (P\langle \alpha := N{\cdot}\gamma\rangle\, Q)\langle \alpha := N{\cdot}\gamma\rangle$
2. We add two substitution rules:
$$M\langle x := N\rangle\langle y := L\rangle \rightarrow M\langle y := L\rangle\langle x := N\rangle\langle y := L\rangle$$
$$M\langle \alpha := N{\cdot}\gamma\rangle\langle \beta := L{\cdot}\delta\rangle \rightarrow M\langle \beta := L{\cdot}\delta\rangle\langle \alpha := N{\cdot}\gamma\rangle\langle \beta := L{\cdot}\delta\rangle$$
3. Application of substitution only if *hv*$(M) = x$ (resp. *hn*$(M) = \alpha$).
4. We remove the contextual rules:
$$M \rightarrow N \;\Rightarrow\; \begin{cases} LM & \rightarrow\; LN \\ L\langle x := M\rangle & \rightarrow\; L\langle x := N\rangle \\ L\langle \alpha := M{\cdot}\gamma\rangle & \rightarrow\; L\langle \alpha := N{\cdot}\gamma\rangle \end{cases}$$

The asynch π-calculus with pairing

Definition *Channel names* and *data* are defined by:

a, b, c, d, x, y, z *names* $\qquad p ::= a \mid \langle a,b \rangle$ *data*

$P, Q ::= 0$	*Nil*	$\mid a(x).P$	*Input*
$\mid P \mid Q$	*Composition*	$\mid \overline{a}\langle p \rangle$	*Output*
$\mid \,!P$	*Replication*	$\mid \mathit{let}\ \langle x,y \rangle = z\ \mathit{in}\ P$	*Let construct*
$\mid (\nu a)\,P$	*Restriction*		

We write $a(x).\,\mathit{let}\ \langle y,z \rangle = x\ \mathit{in}\ P$ for $a(y,z).\,P$, $\overline{a}\langle\langle b,c \rangle\rangle$ for $\overline{a}\langle b,c \rangle$, and $(\nu m)\,(\nu n)\,P$ for $(\nu mn)\,P$.

We extend congruence by: $\mathit{let}\ \langle x,y \rangle = \langle a,b \rangle\ \mathit{in}\ R \equiv R[a/x, b/y]$.

To express that P is a function, we fix *simultaneously* an input and an output (cf. $(\rightarrow R)$ in the sequent calculus); so interfaces for functions are modelled by sending and receiving *pairs* of names.

Reduction, Observables and Equivalence

- The *reduction relation* $\to_\pi$ is defined by:

$$(\textit{synchronisation}):\quad \overline{a}\langle p\rangle \,|\, a(x).Q \;\to_\pi\; Q[p/x]$$

$$(\textit{hiding}):\quad P \to_\pi P' \;\Rightarrow\; (\nu n)P \to_\pi (\nu n)P'$$

$$(\textit{composition}):\quad P \to_\pi P' \;\Rightarrow\; P \,|\, Q \to_\pi P' \,|\, Q$$

$$(\textit{congruence}):\quad P \equiv Q \;\&\; Q \to_\pi Q' \;\&\; Q' \equiv P' \;\Rightarrow\; P \to_\pi P'$$

- $P{\downarrow}n$ (P *outputs on* n): $P \equiv (\nu b_1)\ldots b_m(\overline{n}\langle p\rangle \,|\, Q)$ for some Q, where $n \neq b_1 \ldots b_m$.

- $P{\Downarrow}n$ (P *will output on* n): there exists Q st $P \to_\pi^+ Q$ and $Q{\downarrow}n$.

- $P \sim_C Q$ (and call P and Q *contextually equivalent*): for all contexts $C[\cdot]$, and for all n, $C[P]{\Downarrow}n \iff C[Q]{\Downarrow}n$.

We show our results for $\sim_C$ (but could have used $\approx_\pi$).

Notice that $\overline{a}\langle b,c\rangle \,|\, a(x,y).Q \to_\pi Q[b/x, c/y]$.

(Classical) Type Assignment for π

$$(0) : \frac{}{0 : \ \vdash} \qquad (!) : \frac{P : \Gamma \vdash \Delta}{!P : \Gamma \vdash \Delta} \qquad (\mathit{out}) : \frac{}{\overline{a}\langle b\rangle : b{:}A \vdash a{:}A, b{:}A} \ (a \neq b)$$

$$(\nu) : \frac{P : \Gamma, a{:}A \vdash a{:}A, \Delta}{(\nu a)\, P : \Gamma \vdash \Delta} \qquad (\mathit{pair\text{-}out}) : \frac{}{\overline{a}\langle b,c\rangle : b{:}A \vdash a{:}A{\to}B, c{:}B}$$

$$(|) : \frac{P_1 : \Gamma \vdash \Delta \quad \cdots \quad P_n : \Gamma \vdash \Delta}{P_1 \mid \cdots \mid P_n : \Gamma \vdash \Delta} \qquad (\mathit{let}) : \frac{P : \Gamma, y{:}B \vdash x{:}A, \Delta}{\mathit{let}\ \langle x,y\rangle = z\ \mathit{in}\ P : \Gamma, z{:}A{\to}B \vdash \Delta} \ (1)$$

$$(W) : \frac{P : \Gamma \vdash \Delta}{P : \Gamma' \vdash \Delta'} \ (2) \qquad (\mathit{in}) : \frac{P : \Gamma, x{:}A \vdash x{:}A, \Delta}{a(x).\,P : \Gamma, a{:}A \vdash \Delta}$$

(1) $y,z \notin \Delta; x \notin \Gamma$; (2) $\Gamma' \supseteq \Gamma, \Delta' \supseteq \Delta$.

This system is not trivial:

$(\nu cb)\,(x(w).\overline{c}\langle w\rangle \mid c(v,d).(!b{\to}v \mid d{\to}a) \mid x(w).\overline{b}\langle w\rangle)$ is not typeable.

Logical translation of $\Lambda\mu\mathbf{x}$ terms

$$
\begin{aligned}
[\![x]\!]\, a &\triangleq x \rightarrow a \\
[\![\lambda x.M]\!]\, a &\triangleq (\nu x b)\,([\![M]\!]\, b \,|\, \overline{a}\langle x,b\rangle) \\
[\![MN]\!]\, a &\triangleq (\nu c)\,([\![M]\!]\, c \,|\; c(v,d).\,([\![v := N]\!] \;|\; d \rightarrow a)) \\
[\![M\,\langle x := N\rangle]\!]\, a &\triangleq (\nu x)\,([\![M]\!]\, a \,|\, !\,[\![N]\!]\, x)
\end{aligned}
$$

This is the encoding of [vBV-CONCUR'09].

Logical translation of $\Lambda\mu\mathbf{x}$ terms

$$
\begin{aligned}
\ulcorner x \lrcorner\, a &\triangleq x \rightarrow a \\
\ulcorner \lambda x.M \lrcorner\, a &\triangleq (\nu xb)\,(\ulcorner M \lrcorner\, b \mid \overline{a}\langle x,b\rangle) \\
\ulcorner MN \lrcorner\, a &\triangleq (\nu c)\,(\ulcorner M \lrcorner\, c \mid\ c(v,d).\,(\ulcorner v := N \lrcorner \mid\ d \rightarrow a)) \\
\ulcorner M\,\langle x := N\rangle \lrcorner\, a &\triangleq (\nu x)\,(\ulcorner M \lrcorner\, a \mid !\ulcorner N \lrcorner\, x)
\end{aligned}
$$

$$
\begin{aligned}
\ulcorner [\beta] M \lrcorner\, a &\triangleq \\
\ulcorner \mu\gamma.M \lrcorner\, a &\triangleq \\
\ulcorner M\,\langle \beta := N \cdot \gamma\rangle \lrcorner\, a &\triangleq
\end{aligned}
$$

Logical translation of $\Lambda\mu\mathbf{x}$ terms

$$
\begin{array}{rcl}
\ulcorner x \lrcorner\, a & \triangleq & x \rightarrow a \\
\ulcorner \lambda x.M \lrcorner\, a & \triangleq & (\nu x b)\,(\ulcorner M \lrcorner\, b \,|\, \overline{a}\langle x,b\rangle) \\
\ulcorner MN \lrcorner\, a & \triangleq & (\nu c)\,(\ulcorner M \lrcorner\, c \,|\; c(v,d).\,(\ulcorner v := N \lrcorner \;|\; d \rightarrow a)) \\
\ulcorner M\,\langle x := N\rangle \lrcorner\, a & \triangleq & (\nu x)\,(\ulcorner M \lrcorner\, a \,|\, \ulcorner x := N \lrcorner\,) \\
\ulcorner x := N \lrcorner & \triangleq & !\,\ulcorner N \lrcorner\, x \\
\ulcorner [\beta] M \lrcorner\, a & \triangleq & \\
\ulcorner \mu\gamma.M \lrcorner\, a & \triangleq & \\
\ulcorner M\,\langle \beta := N\cdot\gamma\rangle \lrcorner\, a & \triangleq &
\end{array}
$$

Logical translation of $\Lambda\mu\mathbf{x}$ terms

$$
\begin{aligned}
\ulcorner x \lrcorner\, a &\triangleq x \rightarrow a \\
\ulcorner \lambda x.M \lrcorner\, a &\triangleq (\nu x b)\,(\ulcorner M \lrcorner\, b \mid \overline{a}\langle x,b\rangle) \\
\ulcorner MN \lrcorner\, a &\triangleq (\nu c)\,(\ulcorner M \lrcorner\, c \mid\ c(v,d).\,(\ulcorner v := N \lrcorner \mid\ d \rightarrow a)) \\
\ulcorner M \langle x := N \rangle \lrcorner\, a &\triangleq (\nu x)\,(\ulcorner M \lrcorner\, a \mid \ulcorner x := N \lrcorner\,) \\
\ulcorner x := N \lrcorner &\triangleq\ !\,\ulcorner N \lrcorner\, x \\
\ulcorner [\beta] M \lrcorner\, a &\triangleq \ulcorner M \lrcorner\, \beta \\
\ulcorner \mu\gamma.M \lrcorner\, a &\triangleq \\
\ulcorner M \langle \beta := N \cdot \gamma \rangle \lrcorner\, a &\triangleq
\end{aligned}
$$

Logical translation of $\Lambda\mu\mathbf{x}$ terms

$$
\begin{aligned}
\ulcorner x \lrcorner\, a &\triangleq x \rightarrow a \\
\ulcorner \lambda x.M \lrcorner\, a &\triangleq (\nu xb)\,(\ulcorner M \lrcorner\, b \,|\, \overline{a}\langle x,b\rangle) \\
\ulcorner MN \lrcorner\, a &\triangleq (\nu c)\,(\ulcorner M \lrcorner\, c \,|\; c(v,d).\,(\ulcorner v := N \lrcorner \;|\; d \rightarrow a)) \\
\ulcorner M\,\langle x := N\rangle \lrcorner\, a &\triangleq (\nu x)\,(\ulcorner M \lrcorner\, a \,|\, \ulcorner x := N \lrcorner\,) \\
\ulcorner x := N \lrcorner &\triangleq\; !\,\ulcorner N \lrcorner\, x \\
\ulcorner [\beta]M \lrcorner\, a &\triangleq \ulcorner M \lrcorner\, \beta \\
\ulcorner \mu\gamma.M \lrcorner\, a &\triangleq (\nu \bullet)\,((\ulcorner M \lrcorner\, \bullet)[a/\gamma]) \\
\ulcorner M\,\langle \beta := N\cdot\gamma\rangle \lrcorner\, a &\triangleq
\end{aligned}
$$

So $\ulcorner \mu\gamma.[\beta]M \lrcorner\, a \triangleq (\nu\bullet)\,((\ulcorner M \lrcorner\, \beta)[a/\gamma]) = \ulcorner M[a/\gamma] \lrcorner\, \beta$, so $\lambda\mu$'s binding-and-naming has no representation in π.

Logical translation of $\Lambda\mu\mathbf{x}$ terms

$$
\begin{aligned}
\ulcorner x \lrcorner\, a &\triangleq x \rightarrow a \\
\ulcorner \lambda x.M \lrcorner\, a &\triangleq (\nu\, xb)\, (\ulcorner M \lrcorner\, b \,|\, \overline{a}\langle x,b\rangle\,) \\
\ulcorner MN \lrcorner\, a &\triangleq (\nu c)\, (\ulcorner M \lrcorner\, c \,|\; c(v,d).\, (\ulcorner v := N \lrcorner \;|\; d \rightarrow a)) \\
\ulcorner M \,\langle x := N\rangle \lrcorner\, a &\triangleq (\nu\, x)\, (\ulcorner M \lrcorner\, a \,|\, \ulcorner x := N \lrcorner\,) \\
\ulcorner x := N \lrcorner &\triangleq\, !\, \ulcorner N \lrcorner\, x \\
\ulcorner [\beta] M \lrcorner\, a &\triangleq \ulcorner M \lrcorner\, \beta \\
\ulcorner \mu\gamma.M \lrcorner\, a &\triangleq (\nu \bullet)\, ((\ulcorner M \lrcorner \bullet)\, [a/\gamma]) \\
\ulcorner M \,\langle \beta := N{\cdot}\gamma\rangle \lrcorner\, a &\triangleq (\nu\, \beta)\, (\ulcorner M \lrcorner\, a \,|\, \ulcorner \beta := N{\cdot}\gamma \lrcorner\,)
\end{aligned}
$$

Logical translation of $\Lambda\mu\mathbf{x}$ terms

$$
\begin{aligned}
\ulcorner x \lrcorner\, a &\triangleq x \rightarrow a \\
\ulcorner \lambda x.M \lrcorner\, a &\triangleq (\nu xb)\,(\ulcorner M \lrcorner\, b \,|\, \overline{a}\langle x,b\rangle) \\
\ulcorner MN \lrcorner\, a &\triangleq (\nu c)\,(\ulcorner M \lrcorner\, c \,|\; c(v,d).\,(\ulcorner v := N \lrcorner \;|\; d \rightarrow a)) \\
\ulcorner M \langle x := N \rangle \lrcorner\, a &\triangleq (\nu x)\,(\ulcorner M \lrcorner\, a \,|\, \ulcorner x := N \lrcorner\,) \\
\ulcorner x := N \lrcorner &\triangleq\; !\ulcorner N \lrcorner\, x \\
\ulcorner [\beta] M \lrcorner\, a &\triangleq \ulcorner M \lrcorner\, \beta \\
\ulcorner \mu\gamma.M \lrcorner\, a &\triangleq (\nu \bullet)\,((\ulcorner M \lrcorner\, \bullet)[a/\gamma]) \\
\ulcorner M \langle \beta := N \cdot \gamma \rangle \lrcorner\, a &\triangleq (\nu \beta)\,(\ulcorner M \lrcorner\, a \,|\, \ulcorner \beta := N \cdot \gamma \lrcorner\,) \\
\ulcorner \alpha := N \cdot \gamma \lrcorner &\triangleq\; !\alpha(v,d).\,(\ulcorner v := N \lrcorner \,|\, !d \rightarrow \gamma)
\end{aligned}
$$

Logical translation of $\Lambda\mu\mathbf{x}$ terms

$$
\begin{array}{rcl}
\llbracket x \rrbracket\, a & \triangleq & x \rightarrow a \\
\llbracket \lambda x.M \rrbracket\, a & \triangleq & (\nu\, x b)\, (\llbracket M \rrbracket\, b \,|\, \overline{a}\langle x,b \rangle) \\
\llbracket MN \rrbracket\, a & \triangleq & (\nu c)\, (\llbracket M \rrbracket\, c \,|\, !c(v,d).\, (\llbracket v := N \rrbracket \,|\, !d \rightarrow a)) \\
\llbracket M \langle x := N \rangle \rrbracket\, a & \triangleq & (\nu\, x)\, (\llbracket M \rrbracket\, a \,|\, \llbracket x := N \rrbracket\,) \\
\llbracket x := N \rrbracket & \triangleq & !\, \llbracket N \rrbracket\, x \\
\llbracket [\beta] M \rrbracket\, a & \triangleq & \llbracket M \rrbracket\, \beta \\
\llbracket \mu\gamma.M \rrbracket\, a & \triangleq & (\nu \bullet)\, ((\llbracket M \rrbracket \bullet)[a/\gamma]) \\
\llbracket M \langle \beta := N \cdot \gamma \rangle \rrbracket\, a & \triangleq & (\nu\, \beta)\, (\llbracket M \rrbracket\, a \,|\, \llbracket \beta := N \cdot \gamma \rrbracket\,) \\
\llbracket \alpha := N \cdot \gamma \rrbracket & \triangleq & !\alpha(v,d).\, (\llbracket v := N \rrbracket \,|\, !d \rightarrow \gamma)
\end{array}
$$

Logical translation of $\Lambda\mu\mathbf{x}$ terms

$$
\begin{aligned}
\ulcorner x \lrcorner\, a &\triangleq x(u).\,!u \rightarrow a \\
\ulcorner \lambda x.M \lrcorner\, a &\triangleq (\nu xb)\,(\ulcorner M \lrcorner\, b \mid \overline{a}\langle x,b\rangle) \\
\ulcorner MN \lrcorner\, a &\triangleq (\nu c)\,(\ulcorner M \lrcorner\, c \mid !c(v,d).\,(\ulcorner v := N \lrcorner \mid !d \rightarrow a)) \\
\ulcorner M\,\langle x := N\rangle \lrcorner\, a &\triangleq (\nu x)\,(\ulcorner M \lrcorner\, a \mid \ulcorner x := N \lrcorner\,) \\
\ulcorner x := N \lrcorner &\triangleq\, !\,(\nu w)\,(\overline{x}\langle w\rangle.\,\ulcorner N \lrcorner\, w) \\
\ulcorner [\beta]M \lrcorner\, a &\triangleq \ulcorner M \lrcorner\, \beta \\
\ulcorner \mu\gamma.M \lrcorner\, a &\triangleq (\nu \bullet)\,((\ulcorner M \lrcorner\, \bullet)[a/\gamma]) \\
\ulcorner M\,\langle \beta := N\cdot\gamma\rangle \lrcorner\, a &\triangleq (\nu \beta)\,(\ulcorner M \lrcorner\, a \mid \ulcorner \beta := N\cdot\gamma \lrcorner\,) \\
\ulcorner \alpha := N\cdot\gamma \lrcorner &\triangleq\, !\alpha(v,d).\,(\ulcorner v := N \lrcorner \mid !d \rightarrow \gamma)
\end{aligned}
$$

As is the case for Milner's translation and in contrast to the interpretation for the λ-calculus, a guard is placed on the replicated terms. This is to make sure that $(\nu x)\,(\ulcorner x := N \lrcorner\,) \sim_{\mathrm{C}} \mathit{0}$.

Logical translation of $\Lambda\mu\mathbf{x}$ terms

$$
\begin{aligned}
\ulcorner x \lrcorner\, a &\triangleq x(u).\,!u \rightarrow a \\
\ulcorner \lambda x.M \lrcorner\, a &\triangleq (\nu xb)\,(\ulcorner M \lrcorner\, b \mid \overline{a}\langle x,b\rangle) \\
\ulcorner MN \lrcorner\, a &\triangleq (\nu c)\,(\ulcorner M \lrcorner\, c \mid !c(v,d).\,(\ulcorner v := N \lrcorner \mid !d \rightarrow a)) \\
\ulcorner M\,\langle x := N\rangle \lrcorner\, a &\triangleq (\nu x)\,(\ulcorner M \lrcorner\, a \mid \ulcorner x := N \lrcorner) \\
\ulcorner x := N \lrcorner &\triangleq !(\nu w)\,(\overline{x}\langle w\rangle.\,\ulcorner N \lrcorner\, w) \\
\ulcorner [\beta] M \lrcorner\, a &\triangleq \ulcorner M \lrcorner\, \beta \\
\ulcorner \mu\gamma.M \lrcorner\, a &\triangleq (\nu \bullet)\,((\ulcorner M \lrcorner \bullet)[a/\gamma]) \\
\ulcorner M\,\langle \beta := N{\cdot}\gamma\rangle \lrcorner\, a &\triangleq (\nu \beta)\,(\ulcorner M \lrcorner\, a \mid \ulcorner \beta := N{\cdot}\gamma \lrcorner) \\
\ulcorner \alpha := N{\cdot}\gamma \lrcorner &\triangleq !\alpha(v,d).\,(\ulcorner v := N \lrcorner \mid !d \rightarrow \gamma)
\end{aligned}
$$

Notice the similarity between

$$
\begin{aligned}
\ulcorner MN \lrcorner\, a &= (\nu c)\,(\ulcorner M \lrcorner\, c \mid !c(v,d).\,(\ulcorner v := N \lrcorner \mid !d \rightarrow a)) \\
\ulcorner M\,\langle c := N{\cdot}\gamma\rangle \lrcorner\, a &= (\nu c)\,(\ulcorner M \lrcorner\, a \mid !c(v,d).\,(\ulcorner v := N \lrcorner \mid !d \rightarrow \gamma))
\end{aligned}
$$

Example

The translation of a β-redex reduces as:

$$
\begin{array}{ll}
\llbracket (\lambda x.P)Q \rrbracket\, a & \triangleq \\
(\nu c)\,((\nu xb)\,(\llbracket P \rrbracket\, b \mid \overline{c}\langle x,b\rangle) \mid !\, c(v,d).\,(\llbracket v := Q \rrbracket \mid !\, d \rightarrow a)) & \rightarrow_\pi \ (c) \\
(\nu bx)\,(\llbracket P \rrbracket\, b \mid !\, b \rightarrow a \mid \llbracket x := Q \rrbracket\,) \mid (\nu c)\,(!\, c(v,d).\,(\llbracket v := Q \rrbracket \mid !\, d \rightarrow a)) & \\
 & \sim_{\mathsf{G}} \ (\textit{garbage}) \\
(\nu bx)\,(\llbracket P \rrbracket\, b \mid !\, b \rightarrow a \mid \llbracket x := Q \rrbracket\,) & \triangleq \\
(\nu bx)\,(\llbracket P \rrbracket\, b \mid !\, b \rightarrow a \mid !\,(\nu w)\,(\overline{x}\langle w\rangle.\,\llbracket Q \rrbracket\, w)) & \sim_{\mathsf{R}} \ (\textit{renaming}) \\
(\nu x)\,(\llbracket P \rrbracket\, a \mid !\,(\nu w)\,(\overline{x}\langle w\rangle.\,\llbracket Q \rrbracket\, w)) & \triangleq \\
\llbracket P\,\langle x := Q\rangle \rrbracket\, a &
\end{array}
$$

This implies that β-reduction is implemented in π by at least one π-reduction.

Results

(*Type preservation*) : If $\Gamma \vdash_{\mu\mathbf{x}} M : A \mid \Delta$, then $\llbracket M \rrbracket\, a : \Gamma \vdash_{\pi} a{:}A, \Delta$.

Results

(*Type preservation*) : If $\Gamma \vdash_{\mu\mathbf{x}} M : A \mid \Delta$, then $[\![M]\!]\, a : \Gamma \vdash_{\pi} a{:}A, \Delta$.

(*Soundness*) : $M \rightarrow_{\mathbf{x}\mathrm{H}} N \Rightarrow [\![M]\!]\, a \rightarrow^{*}_{\pi}, \sim_{\mathrm{G}}, \sim_{\mathrm{R}} [\![N]\!]\, a$.

Results

(*Type preservation*) : If $\Gamma \vdash_{\mu\mathbf{x}} M : A \mid \Delta$, then $[\![M]\!]\, a : \Gamma \vdash_{\pi} a{:}A, \Delta$.

(*Soundness*) : $M \rightarrow_{\mathbf{x}\mathsf{H}} N \Rightarrow [\![M]\!]\, a \rightarrow^{*}_{\pi}, \sim_{\mathsf{G}}, \sim_{\mathsf{R}} [\![N]\!]\, a$.

(*Operational Soundness*) : 1. $M \rightarrow^{*}_{\mathbf{x}\mathsf{H}} N \Rightarrow [\![M]\!]\, a \sim_{\mathsf{C}} [\![N]\!]\, a$.

2. $M \uparrow_{\mathbf{x}\mathsf{H}} \Rightarrow [\![M]\!]\, a \uparrow_{\pi}$.

Results

(*Type preservation*) : If $\Gamma \vdash_{\mu\mathbf{x}} M : A \mid \Delta$, then $⟦M⟧\, a : \Gamma \vdash_{\pi} a{:}A, \Delta$.

(*Soundness*) : $M \rightarrow_{\mathbf{x}\mathrm{H}} N \Rightarrow ⟦M⟧\, a \rightarrow^{*}_{\pi}, \sim_{\mathrm{G}}, \sim_{\mathrm{R}} ⟦N⟧\, a$.

(*Operational Soundness*) : 1. $M \rightarrow^{*}_{\mathbf{x}\mathrm{H}} N \Rightarrow ⟦M⟧\, a \sim_{\mathrm{C}} ⟦N⟧\, a$.

2. $M \uparrow_{\mathbf{x}\mathrm{H}} \Rightarrow ⟦M⟧\, a \uparrow_{\pi}$.

(*Adequacy*) : $M =_{\mathbf{x}\mathrm{H}} N \Rightarrow ⟦M⟧\, a \sim_{\mathrm{C}} ⟦N⟧\, a$.

Results

($\textit{Type preservation}$) : If $\Gamma \vdash_{\mu\mathbf{x}} M : A \mid \Delta$, then $[\![M]\!]\, a : \Gamma \vdash_{\pi} a{:}A, \Delta$.

($\textit{Soundness}$) : $M \rightarrow_{\mathbf{x}\mathsf{H}} N \Rightarrow [\![M]\!]\, a \rightarrow^{*}_{\pi}, \sim_{\mathsf{G}}, \sim_{\mathsf{R}} [\![N]\!]\, a$.

($\textit{Operational Soundness}$) : 1. $M \rightarrow^{*}_{\mathbf{x}\mathsf{H}} N \Rightarrow [\![M]\!]\, a \sim_{\mathsf{C}} [\![N]\!]\, a$.

2. $M \uparrow_{\mathbf{x}\mathsf{H}} \Rightarrow [\![M]\!]\, a \uparrow_{\pi}$.

($\textit{Adequacy}$) : $M =_{\mathbf{x}\mathsf{H}} N \Rightarrow [\![M]\!]\, a \sim_{\mathsf{C}} [\![N]\!]\, a$.

($\textit{Operational completeness for}$ $\rightarrow_{\mathbf{x}\mathsf{H}}$) : If $[\![M]\!]\, a \rightarrow_{\pi} P$ then there exists N such that $P \rightarrow^{+}_{\pi}, \sim_{\mathsf{R}}, \sim_{\mathsf{G}} [\![N]\!]\, a$, and $M \rightarrow_{\mathbf{x}\mathsf{H}} N$.

Results

(*Type preservation*) : If $\Gamma \vdash_{\mu\mathbf{x}} M : A \mid \Delta$, then $\llbracket M \rrbracket a : \Gamma \vdash_{\pi} a{:}A, \Delta$.

(*Soundness*) : $M \to_{\mathbf{x}\mathrm{H}} N \Rightarrow \llbracket M \rrbracket a \to^{*}_{\pi}, \sim_{\mathrm{G}}, \sim_{\mathrm{R}} \llbracket N \rrbracket a$.

(*Operational Soundness*) : 1. $M \to^{*}_{\mathbf{x}\mathrm{H}} N \Rightarrow \llbracket M \rrbracket a \sim_{\mathrm{C}} \llbracket N \rrbracket a$.

2. $M \uparrow_{\mathbf{x}\mathrm{H}} \Rightarrow \llbracket M \rrbracket a \uparrow_{\pi}$.

(*Adequacy*) : $M =_{\mathbf{x}\mathrm{H}} N \Rightarrow \llbracket M \rrbracket a \sim_{\mathrm{C}} \llbracket N \rrbracket a$.

(*Operational completeness for $\to_{\mathbf{x}\mathrm{H}}$*) : If $\llbracket M \rrbracket a \to_{\pi} P$ then there exists N such that $P \to^{+}_{\pi}, \sim_{\mathrm{R}}, \sim_{\mathrm{G}} \llbracket N \rrbracket a$, and $M \to_{\mathbf{x}\mathrm{H}} N$.

(•) : We think these results also hold for $\approx_{\pi}$.

Results

(*Type preservation*) : If $\Gamma \vdash_{\mu\mathbf{x}} M : A \mid \Delta$, then $\llbracket M \rrbracket a : \Gamma \vdash_\pi a{:}A, \Delta$.

(*Soundness*) : $M \rightarrow_{\mathbf{x}\mathrm{H}} N \Rightarrow \llbracket M \rrbracket a \rightarrow^*_{\pi}, \sim_{\mathrm{G}}, \sim_{\mathrm{R}} \llbracket N \rrbracket a$.

(*Operational Soundness*) : 1. $M \rightarrow^*_{\mathbf{x}\mathrm{H}} N \Rightarrow \llbracket M \rrbracket a \sim_{\mathrm{C}} \llbracket N \rrbracket a$.

2. $M \uparrow_{\mathbf{x}\mathrm{H}} \Rightarrow \llbracket M \rrbracket a \uparrow_\pi$.

(*Adequacy*) : $M =_{\mathbf{x}\mathrm{H}} N \Rightarrow \llbracket M \rrbracket a \sim_{\mathrm{C}} \llbracket N \rrbracket a$.

(*Operational completeness for $\rightarrow_{\mathbf{x}\mathrm{H}}$*) : If $\llbracket M \rrbracket a \rightarrow_\pi P$ then there exists N such that $P \rightarrow^+_{\pi}, \sim_{\mathrm{R}}, \sim_{\mathrm{G}} \llbracket N \rrbracket a$, and $M \rightarrow_{\mathbf{x}\mathrm{H}} N$.

(*Termination*) : 1. If $M \rightarrow^*_{\mathbf{x}\mathrm{H}} N$, with N in explicit head-normal from, then $\llbracket M \rrbracket a \downarrow_\pi$.

2. If $M \rightarrow^*_{\beta\mu} N$, with N in head-normal from, then $\llbracket M \rrbracket a \downarrow_\pi$.

3. Let $M \in \Lambda\mu$. If $\llbracket M \rrbracket a \downarrow_\pi$ then there exists $N \in \Lambda\mu\mathbf{x}$ and L in $\rightarrow_{\lambda\mu}$-head normal form such that $\llbracket M \rrbracket a \sim_{\mathrm{C}} \llbracket N \rrbracket a$, and $M \rightarrow^*_{\mathbf{x}\mathrm{H}} N$ and $N \rightarrow^*_{:=} L$.

Conclusion

- We have found a new, simple and intuitive interpretation of $\lambda\mu$-terms in π, interpreting terms under output;

Conclusion

- We have found a new, simple and intuitive interpretation of $\lambda\mu$-terms in π, interpreting terms under output;
- It respects step-by-step explicit head reduction.

Conclusion

- We have found a new, simple and intuitive interpretation of $\lambda\mu$-terms in π, interpreting terms under output;
- It respects step-by-step explicit head reduction.
- We have shown that, for our context assignment system that uses the type constructor $\rightarrow$ for π and is based on classical logic, assignable types for $\lambda\mu$-terms are preserved by our interpretation as typeable π-processes.

Conclusion

- We have found a new, simple and intuitive interpretation of $\lambda\mu$-terms in π, interpreting terms under output;
- It respects step-by-step explicit head reduction.
- We have shown that, for our context assignment system that uses the type constructor $\rightarrow$ for π and is based on classical logic, assignable types for $\lambda\mu$-terms are preserved by our interpretation as typeable π-processes.
- We managed this without having to linearise the calculus and without adding new reduction rules.

Conclusion

- We have found a new, simple and intuitive interpretation of $\lambda\mu$-terms in π, interpreting terms under output;
- It respects step-by-step explicit head reduction.
- We have shown that, for our context assignment system that uses the type constructor $\rightarrow$ for π and is based on classical logic, assignable types for $\lambda\mu$-terms are preserved by our interpretation as typeable π-processes.
- We managed this without having to linearise the calculus and without adding new reduction rules.
- We have shown termination for $\beta\mu$-reduction (not just explicit head reduction).

Conclusion

- We have found a new, simple and intuitive interpretation of $\lambda\mu$-terms in π, interpreting terms under output;
- It respects step-by-step explicit head reduction.
- We have shown that, for our context assignment system that uses the type constructor $\rightarrow$ for π and is based on classical logic, assignable types for $\lambda\mu$-terms are preserved by our interpretation as typeable π-processes.
- We managed this without having to linearise the calculus and without adding new reduction rules.
- We have shown termination for $\beta\mu$-reduction (not just explicit head reduction).
- Full abstraction wrt *approximation semantics* and *bisimulation* seems feasible.

Questions

?