Name-Passing in an Ambient-Like Calculus and Its Proof Using Spatial Logic

Xudong GUAN, DI.FCT.UNL.PT

Part I. The Calculus

Background - Safe Ambients

Grave interference in Mobile Ambients:

$$a[in\ c \mid b[out\ a]\ ] \mid c[\ ] \longrightarrow^* c[b[\ ] \mid a[\ ]\ ]$$

$$a[in\ c \mid b[out\ a]\ ] \mid c[\ ] \longrightarrow^* b[\ ] \mid c[a[\ ]\ ]$$

Safe Ambients:

$$a[in\ c.\overline{\texttt{out}}\ a \mid b[out\ a]\ ] \mid c[\overline{\texttt{in}}\ c] \longrightarrow^* c[b[\ ] \mid a[\ ]\ ]$$

$$a[\overline{\texttt{out}}\ a.in\ c \mid b[out\ a]\ ] \mid c[\overline{\texttt{in}}\ c] \longrightarrow^* b[\ ] \mid c[a[\ ]\ ]$$

Backgroud - ST/IM type system

Type system (ST/IM) is essential to eliminate GI in SA, since MA $\subset$ SA:

$$a[\overline{\texttt{out}}\, a \mid in\ c \mid b[out\ a]\ ] \mid c[\overline{\texttt{in}}\, c]$$

ST type - single-threaded and usually mobile

$$a[\texttt{out}\ s.\texttt{in}\ s'.\cdots.\overline{\texttt{open}}\ a \mid a'[P] \mid s''[Q] \mid ...]$$

IM type - immobile, persistent, and usually multi-threaded

$$s[\,!\,\overline{\texttt{in}}\ s \mid !\,\overline{\texttt{out}}\ s \mid !\,\texttt{open}\ a \mid a[P] \mid s'[Q] \mid ...]$$

From Typed SA (ST agent, IM server) to Wagon

Agent-server interactions:

$$\text{SA} \quad \left(\begin{array}{ccc} a[\texttt{in}\ s.M \mid P] \mid s[\,!\,\overline{\texttt{in}}\ s \mid Q] & \longrightarrow & s[a[M \mid P] \mid !\,\overline{\texttt{in}}\ s \mid Q] \\ s[\,!\,\overline{\texttt{out}}\ s \mid a[\texttt{out}\ s.M \mid P] \mid Q] & \longrightarrow & a[M \mid P] \mid s[\,!\,\overline{\texttt{out}}\ s \mid Q] \\ s[\,!\,\texttt{open}\ a \mid a[\overline{\texttt{open}}\ a \mid P] \mid Q] & \longrightarrow & s[\,!\,\texttt{open}\ a \mid P \mid Q] \end{array}\right)$$

$$\text{Wagon} \left(\begin{array}{ccc} s.M\langle P\rangle \mid s[Q] & \longrightarrow & s[M\langle P\rangle \mid Q] \\ s[\uparrow.M\langle P\rangle \mid Q] & \longrightarrow & M\langle P\rangle \mid s[Q] \\ s[\texttt{dis}\langle P\rangle \mid Q] & \longrightarrow & s[P \mid Q] \end{array}\right)$$

Agent-agent interaction:

$$\text{SA} \qquad a[\texttt{in}\ b.\overline{\texttt{open}}\ a \mid P] \mid b[\overline{\texttt{in}}\ b.\texttt{open}\ a.M \mid Q] \longrightarrow^{*} b[M \mid P \mid Q]$$

$$\text{Wagon} \quad \texttt{put}\langle P\rangle \mid \texttt{get}.M\langle Q\rangle \longrightarrow M\langle P \mid Q\rangle$$

The Wagon Calculus

Syntax:

$$
\begin{array}{rcl}
M, N & ::= & \texttt{dis} \mid \texttt{put} \mid \texttt{get}.M \mid \uparrow.M \mid a.M \\
P, Q & ::= & \mathbf{0} \mid (\nu\, a)P \mid (P \mid Q) \mid a[P] \mid M\langle P\rangle \mid !\, M\langle P\rangle
\end{array}
$$

Reduction rules:

$$
\begin{array}{rcl}
a.M\langle P\rangle \mid a[Q] & \longrightarrow & a[M\langle P\rangle \mid Q] \\
a[\uparrow.M\langle P\rangle \mid Q] & \longrightarrow & M\langle P\rangle \mid a[Q] \\
\texttt{put}\langle P\rangle \mid \texttt{get}.M\langle Q\rangle & \longrightarrow & M\langle P \mid Q\rangle
\end{array}
$$

Reduction context: $P \mid -$, $(\nu\, a)-$, $a[-]$, but not $M\langle -\rangle$

Dissolving (`dis` is often dropped): $\langle P\rangle \equiv P$

Example - Routing

$$
\begin{array}{ll}
& Marseille[\,toLisbon[\,!\,\texttt{get}.\uparrow.\uparrow.Lisbon\langle\rangle] \mid toLisbon.\texttt{put}\langle P\rangle] \mid Lisbon[Q] \\
\longrightarrow & Marseille[\,toLisbon[\,!\,\texttt{get}.\uparrow.\uparrow.Lisbon\langle\rangle \mid \texttt{put}\langle P\rangle]\,] \mid Lisbon[Q] \\
\longrightarrow & Marseille[\,toLisbon[\,!\,\texttt{get}.\uparrow.\uparrow.Lisbon\langle\rangle \mid \uparrow.\uparrow.Lisbon\langle P\rangle]\,] \mid Lisbon[Q] \\
\longrightarrow & Marseille[\,toLisbon[\,!\,\texttt{get}.\uparrow.\uparrow.Lisbon\langle\rangle] \mid \uparrow.Lisbon\langle P\rangle] \mid Lisbon[Q] \\
\longrightarrow & Marseille[\,toLisbon[\,!\,\texttt{get}.\uparrow.\uparrow.Lisbon\langle\rangle]\,] \mid Lisbon\langle P\rangle \mid Lisbon[Q] \\
\longrightarrow & Marseille[\,toLisbon[\,!\,\texttt{get}.\uparrow.\uparrow.Lisbon\langle\rangle]\,] \mid Lisbon[P \mid Q]
\end{array}
$$

Part II. The Encoding

State-of-the-Art of Name-Passing in Ambients

Name-passing in MA/SA *with* local communication [CG98, LS00].

Name-pasisng in SA *without* local communication [Zimmer00] – very tricky, and complex!

- channel servers (queues for I/O processes)
- I/O processes interacting inside channel servers
- creation of explicit substitution servers (after each communication)

Formal proof of [Zimmer00]:

- Encoding may follow source — yes!
- Encoding always follows source — conjecture.

Encoding Name-Passing – Overview

π_a: $a(b).p \mid a\langle c\rangle \longrightarrow p\{b := c\}$

$$\Downarrow$$

π_a with explicit substitution $[b := c]$ and channel $a[...]$

$$\Downarrow$$

Wagon implementation:

$$\begin{aligned} [b := c] &\implies b[\,!\,\texttt{get}.\uparrow.c.\texttt{put}\langle\rangle] \\ a[...] &\implies a[\,!\,\texttt{get}\langle\rangle \mid ...] \\ a\langle c\rangle,\ a(b).p &\implies a.\texttt{put}\langle...\rangle \end{aligned}$$

π_a with Explicit Channel and Subsitution [Zimmer00]

Obtained from π_a by providing every free name a with an empty explicit channel $a[\,]$, and breaking down $a(b).p \mid a\langle c\rangle \longrightarrow p\{b := c\}$ into the following 5 low-level rules:

$$\left(\begin{array}{rcll}
a(b).p \mid a[...] & \longrightarrow & a[(b).p \mid ...] & \texttt{routing-1} \\
a\langle c\rangle \mid a[...] & \longrightarrow & a[\langle c\rangle \mid ...] & \texttt{routing-2} \\
a[(b).p \mid \langle c\rangle \mid ...] & \longrightarrow & a[...] \mid (\nu\, b)([b := c] \mid p) & \texttt{communication} \\
[b := c] \mid b(d).p & \longrightarrow & [b := c] \mid c(d).p & \texttt{routing-3} \\
[b := c] \mid b\langle d\rangle & \longrightarrow & [b := c] \mid c\langle d\rangle & \texttt{routing-4}
\end{array}\right)$$

The Encoding $\langle\!\langle p \rangle\!\rangle$

$$
\begin{aligned}
[\![\mathbf{0}]\!] &\triangleq \mathbf{0} \\
[\![(\nu\, a)p]\!] &\triangleq (\nu\, a)(a[\mathbf{chn}] \mid [\![p]\!]) \\
[\![p \mid q]\!] &\triangleq [\![p]\!] \mid [\![q]\!] \\
[\![a\langle b\rangle]\!] &\triangleq a.\mathtt{put}\langle comm.\mathtt{put}\langle \mathbf{fwd}\ b\rangle\rangle \\
[\![a(b).p]\!] &\triangleq (\nu\, b)(b[\mathbf{0}] \\
&\qquad \mid a.\mathtt{put}\langle comm.\mathtt{get}. \uparrow . \uparrow .b\langle \uparrow \langle [\![p]\!]\rangle\rangle\rangle)
\end{aligned}
$$

$$
\begin{aligned}
\mathbf{chn} &\triangleq\ !\,\mathtt{get}\langle\rangle \mid comm[\mathbf{0}] \\
\mathbf{fwd}\ b &\triangleq\ !\,\mathtt{get}. \uparrow .b.\mathtt{put}\langle\rangle \\
\langle\!\langle p \rangle\!\rangle_X &\triangleq [\![p]\!] \mid \textstyle\prod_{a \in X} a[\mathbf{chn}] \\
\langle\!\langle p \rangle\!\rangle &\triangleq \langle\!\langle p \rangle\!\rangle_{fn(p)}
\end{aligned}
$$

Routing Into Explicit Channel

$$
\begin{array}{rcl}
[\![\mathbf{0}]\!] & \triangleq & \mathbf{0} \\
[\![(\nu\, a)p]\!] & \triangleq & (\nu\, a)(a[\mathbf{chn}] \mid [\![p]\!]) \\
[\![p \mid q]\!] & \triangleq & [\![p]\!] \mid [\![q]\!] \\
[\![a\langle b\rangle]\!] & \triangleq & a.\mathtt{put}\langle comm.\mathtt{put}\langle \mathbf{fwd}\ b\rangle\rangle \\
[\![a(b).p]\!] & \triangleq & (\nu\, b)(b[\mathbf{0}] \\
 & & \mid a.\mathtt{put}\langle comm.\mathtt{get}.\uparrow.\uparrow.b\langle \uparrow\langle [\![p]\!]\rangle\rangle\rangle)
\end{array}
$$

$$
\begin{array}{rcl}
\mathbf{chn} & \triangleq & !\,\mathtt{get}\langle\rangle \mid comm[\mathbf{0}] \\
\mathbf{fwd}\ b & \triangleq & !\,\mathtt{get}.\uparrow.b.\mathtt{put}\langle\rangle \\
\langle\!\langle p\rangle\!\rangle_X & \triangleq & [\![p]\!] \mid \prod_{a\in X} a[\mathbf{chn}] \\
\langle\!\langle p\rangle\!\rangle & \triangleq & \langle\!\langle p\rangle\!\rangle_{fn(p)}
\end{array}
$$

$a\langle c\rangle \mid a[...] \longrightarrow a[\langle c\rangle \mid ...]$ `routing-2`

$\Downarrow$

$a.\mathtt{put}\langle comm.\mathtt{put}\langle \mathbf{fwd}\ c\rangle\rangle \mid a[\,!\,\mathtt{get}\langle\rangle \mid ...] \longrightarrow^* a[\,!\,\mathtt{get}\langle\rangle \mid comm.\mathtt{put}\langle \mathbf{fwd}\ c\rangle \mid ...]$

Routing With Explicit Substitution

$$
\begin{array}{rcl}
[\![\mathbf{0}]\!] & \triangleq & \mathbf{0} \\
[\![(\nu\, a)p]\!] & \triangleq & (\nu\, a)(a[\mathbf{chn}] \mid [\![p]\!]) \\
[\![p \mid q]\!] & \triangleq & [\![p]\!] \mid [\![q]\!] \\
[\![a\langle b\rangle]\!] & \triangleq & a.\mathtt{put}\langle comm.\mathtt{put}\langle \mathbf{fwd}\ b\rangle\rangle \\
[\![a(b).p]\!] & \triangleq & (\nu\, b)(b[\mathbf{0}] \\
 & & \mid a.\mathtt{put}\langle comm.\mathtt{get}.\uparrow.\uparrow.b\langle\uparrow\langle[\![p]\!]\rangle\rangle\rangle)
\end{array}
$$

$$
\begin{array}{rcl}
\mathbf{chn} & \triangleq & !\,\mathtt{get}\langle\rangle \mid comm[\mathbf{0}] \\
\mathbf{fwd}\ b & \triangleq & !\,\mathtt{get}.\uparrow.b.\mathtt{put}\langle\rangle \\
\langle\!\langle p\rangle\!\rangle_X & \triangleq & [\![p]\!] \mid \prod_{a\in X} a[\mathbf{chn}] \\
\langle\!\langle p\rangle\!\rangle & \triangleq & \langle\!\langle p\rangle\!\rangle_{fn(p)}
\end{array}
$$

$b[\mathbf{fwd}\ c] \mid b\langle d\rangle \longrightarrow b[\mathbf{fwd}\ c] \mid c\langle d\rangle$ `routing-4`

$$\Downarrow$$

$b[\,!\,\mathtt{get}.\uparrow.c.\mathtt{put}\langle\rangle] \mid b.\mathtt{put}\langle\ldots\rangle$

$\longrightarrow b[\,!\,\mathtt{get}.\uparrow.c.\mathtt{put}\langle\rangle \mid \mathtt{put}\langle\ldots\rangle]$

$\longrightarrow b[\,!\,\mathtt{get}.\uparrow.c.\mathtt{put}\langle\rangle \mid\uparrow.c.\mathtt{put}\langle\ldots\rangle]$

$\longrightarrow b[\,!\,\mathtt{get}.\uparrow.c.\mathtt{put}\langle\rangle] \mid c.\mathtt{put}\langle\ldots\rangle$

Communication

$$\begin{array}{rcl} [\![\mathbf{0}]\!] & \triangleq & \mathbf{0} \\ [\![(\nu\, a)p]\!] & \triangleq & (\nu\, a)(a[\mathbf{chn}] \mid [\![p]\!]) \\ [\![p \mid q]\!] & \triangleq & [\![p]\!] \mid [\![q]\!] \\ [\![a\langle b\rangle]\!] & \triangleq & a.\mathtt{put}\langle \mathit{comm}.\mathtt{put}\langle \mathbf{fwd}\ b\rangle\rangle \\ [\![a(b).p]\!] & \triangleq & (\nu\, b)(b[\mathbf{0}] \\ & & \mid a.\mathtt{put}\langle \mathit{comm}.\mathtt{get}.\uparrow.\uparrow.b\langle\uparrow\langle[\![p]\!]\rangle\rangle\rangle) \end{array}$$

$$\begin{array}{rcl} \mathbf{chn} & \triangleq & !\,\mathtt{get}\langle\rangle \mid \mathit{comm}[\mathbf{0}] \\ \mathbf{fwd}\ b & \triangleq & !\,\mathtt{get}.\uparrow.b.\mathtt{put}\langle\rangle \\ \langle\!\langle p\rangle\!\rangle_X & \triangleq & [\![p]\!] \mid \prod_{a\in X} a[\mathbf{chn}] \\ \langle\!\langle p\rangle\!\rangle & \triangleq & \langle\!\langle p\rangle\!\rangle_{fn(p)} \end{array}$$

$a[(b).p \mid \langle c\rangle \mid ...] \longrightarrow a[...] \mid (\nu\, b)(b[\mathbf{fwd}\ c] \mid p)$ `communication`

$$\Downarrow$$

$(\nu\, b)(b[\,] \mid a[\mathit{comm}.\mathtt{get}.\uparrow.\uparrow.b\langle\uparrow\langle[\![p]\!]\rangle\rangle \mid \mathit{comm}.\mathtt{put}\langle\mathbf{fwd}\ c\rangle \mid \mathit{comm}[...] \mid ...])$

$\longrightarrow^2 (\nu\, b)(b[\,] \mid a[\mathit{comm}[\mathtt{get}.\uparrow.\uparrow.b\langle\uparrow\langle[\![p]\!]\rangle\rangle \mid \mathtt{put}\langle\mathbf{fwd}\ c\rangle \mid ...] \mid ...])$

$\longrightarrow (\nu\, b)(b[\,] \mid a[\mathit{comm}[\uparrow.\uparrow.b\langle\uparrow\langle[\![p]\!]\rangle \mid \mathbf{fwd}\ c\rangle \mid ...] \mid ...])$

$\longrightarrow^3 (\nu\, b)(b[\uparrow\langle[\![p]\!]\rangle \mid \mathbf{fwd}\ c] \mid a[\mathit{comm}[...] \mid ...])$

$\longrightarrow a[\mathit{comm}[...] \mid ...] \mid (\nu\, b)(b[\mathbf{fwd}\ c] \mid [\![p]\!])$

Part III. The Proof

Overview

Goal: Finding the right equivalence $\simeq$ s.t.

- **Follows:** $b[\textbf{fwd}\ c] \mid b.\texttt{put}\langle ... \rangle \simeq b[\textbf{fwd}\ c] \mid c.\texttt{put}\langle ... \rangle$
- **Always follows:** A stronger requirement: all auxiliary reduction implies $\simeq$.

Solution:

- Find the general shape of the encoding using logic formulas, then find its reduction-closed closure.
- Enumerating all possible interactions using the closure.

Chemical Representation

Motivation: Tighter name scope

Method: Breaking down $a[P \mid Q \mid ...]$ (vertical structure) to $P@a \mid Q@a \mid ...$ (horizontal structure) using the following heating/cooling rules:

$$a[A \mid B] \rightleftharpoons a[A] \mid B@a \qquad a[0] \rightleftharpoons a$$

Syntax: $A ::= \mathbf{0} \mid (\nu\, a)A \mid (A \mid B) \mid a[A] \mid M\langle A\rangle \mid !\, M\langle A\rangle \mid a \mid A@a$

After fully heated, no $a[...]$ will appear in a solution. It is fully horizontal. Each component is of the form $M\langle ...\rangle@s$ or $a@s$.

Reduction of (fully heated) solutions:

$$\begin{array}{rcl} a.M\langle A\rangle@s \mid a@s & \hookrightarrow & M\langle A\rangle@a@s \mid a@s \\ \uparrow.M\langle A\rangle@a@s & \hookrightarrow & M\langle A\rangle@s \\ \texttt{get}.M\langle A\rangle@s \mid \texttt{put}\langle B\rangle@s & \hookrightarrow & M\langle A \mid B\rangle@s \end{array}$$

Spatial Logic

A spatial logic formula $\mathbb{A}$ is a set of ($\rightleftharpoons$-closed) solutions.

$$
\begin{array}{rcl}
u,\, v & ::= & x \mid X \mid u \cup v \\
\mathbb{M},\, \mathbb{N} & ::= & \mathtt{dis} \mid \mathtt{put} \mid \mathtt{get}.\mathbb{M} \mid \uparrow.\mathbb{M} \mid u.\mathbb{M} \\
\mathbb{A},\, \mathbb{B} & ::= & \mathbf{0} \mid (\nu\, a)\mathbb{A} \mid (\mathbb{A} \mid \mathbb{B}) \mid a[\mathbb{A}] \mid \mathbb{M}\langle\mathbb{A}\rangle \mid !\,\mathbb{M}\langle\mathbb{A}\rangle \mid a \mid \mathbb{A}@u \\
& \mid & \mathbb{A}^* \mid \prod_{a \in u} \mathbb{A} \mid (\nu\, x)\mathbb{A} \mid \mathbb{A}(\vec{u})
\end{array}
$$

$$
\begin{array}{rcl}
0 & \triangleq & \{0\} \\
\mathtt{put}\langle\mathbb{A}\rangle & \triangleq & \{\mathtt{put}\langle A\rangle \mid A \in \mathbb{A}\} \\
u.M\langle\mathbb{A}\rangle & \triangleq & \{a.M\langle A\rangle \mid a \in u \wedge A \in \mathbb{A}\} \\
\mathbb{A}@u & \triangleq & \{A@a \mid a \in u \wedge A \in \mathbb{A}\} \\
\mathbb{A} \mid \mathbb{B} & \triangleq & \{C \mid \exists A \in \mathbb{A}.\exists B \in \mathbb{B}.C \rightleftharpoons A \mid B\} \\
(\nu\, a)\mathbb{A} & \triangleq & \{(\nu\, a)A \mid A \in \mathbb{A}\}
\end{array}
$$

$$
\begin{array}{rcl}
(\nu\, x)\mathbb{A} & \triangleq & \{(\nu\, X)A \mid A \in \mathbb{A}\{x := X\}\} \\
\mathbb{A}^* & \triangleq & \{C \mid \exists k \geq 0.\exists A_1 \in \mathbb{A}.\cdots.\exists A_k \in \mathbb{A}.C \rightleftharpoons A_1 \mid ... \mid A_k\} \\
 & \cdots &
\end{array}
$$

A closure is a reduction-closed formula.

Finding the Closure for the Encoding

Start point:

$$X.\texttt{put}\langle ...\rangle^* \mid ((\nu\, b)(b[\,] \mid X.\texttt{put}\langle ...\rangle))^* \mid \prod_{a \in X} a[comm[]]$$

$$\Downarrow$$

$$(\nu\, Y)(XY.\texttt{put}\langle ...\rangle^* \mid ((\nu\, b)(b[\,] \mid XY.\texttt{put}\langle ...\rangle))^* \mid \prod_{a \in XY} a[comm[]])$$

$$\Downarrow$$

For all p with $fn(p) = X$ $\mathbb{O}_1^*$, $\mathbb{I}_1^*$, $\mathbb{C}$

$$\langle\!\langle p \rangle\!\rangle \in (\nu\, y)(Xy.\texttt{put}\langle ...\rangle^* \mid ((\nu\, b)(b[\,] \mid Xy.\texttt{put}\langle ...\rangle))^* \mid \prod_{a \in Xy} a[comm[]])$$

$$\triangleq (\nu\, y)(\mathbb{O}_1^* \mid \mathbb{I}_1^* \mid \mathbb{C})$$

Finding the Reduction Closure

$\mathbb{O}_1/\mathbb{I}_1$-solutions move into channels: $\mathbb{O}_2^*$, $\mathbb{I}_2^*$

$$\ldots \mid (\mathtt{put}\langle\ldots\rangle@Xy)^* \mid ((\nu\, b)(b[\,] \mid \mathtt{put}\langle\ldots\rangle@Xy))^*$$

$\mathbb{O}_2/\mathbb{I}_2$-solutions communicate with $!\,\mathtt{get}\langle\rangle$, and dissolve: $\mathbb{O}_3^*$, $\mathbb{I}_3^*$

$$\ldots \mid (comm.\mathtt{put}\langle\ldots\rangle@Xy)^* \mid ((\nu\, b)(b[\,] \mid comm.\mathtt{get}.\uparrow.\uparrow.b\langle\ldots\rangle@Xy))^*$$

$\mathbb{O}_3/\mathbb{I}_3$-solutions move into *comm*: $\mathbb{O}_4^*$, $\mathbb{I}_4^*$

$$\ldots \mid (\mathtt{put}\langle\ldots\rangle@comm@Xy)^* \mid ((\nu\, b)(b[\,] \mid \mathtt{get}.\uparrow.\uparrow.b\langle\ldots\rangle@comm@Xy))^*$$

$\mathbb{O}_4/\mathbb{I}_4$-solutions interact inside *comm*: $\mathbb{I}_5^*$

$$\ldots \mid ((\nu\, b)(b[\,] \mid \uparrow.\uparrow.b\langle\ldots\rangle@comm@Xy))^*$$

$\mathbb{I}_5$-solutions leave *comm*: $\mathbb{I}_6^*$

$$\ldots \mid ((\nu\, b)(b[\,] \mid \uparrow.b\langle\ldots\rangle@Xy))^*$$

$\mathbb{I}_6$-solutions leave channel $\mathbb{I}_7^*$

$$\ldots \mid ((\nu\, b)(b[\,] \mid b\langle\ldots\rangle)^*$$

$\mathbb{I}_7$-solutions move into b: $\mathbb{I}_8^*$

$$\ldots \mid ((\nu\, b)(b[\,] \mid \uparrow \langle\ldots\rangle @b \mid \textbf{fwd}\ Xy@b)^*$$

The collection of explicit subsitutions generated by $\mathbb{I}_8$-solutions: $\mathbb{V}$

$$(\nu\, z)(\ldots \mid \prod_{a \in z} a[\,!\,\texttt{get}.\uparrow .Xyz.\texttt{put}\langle\rangle])$$

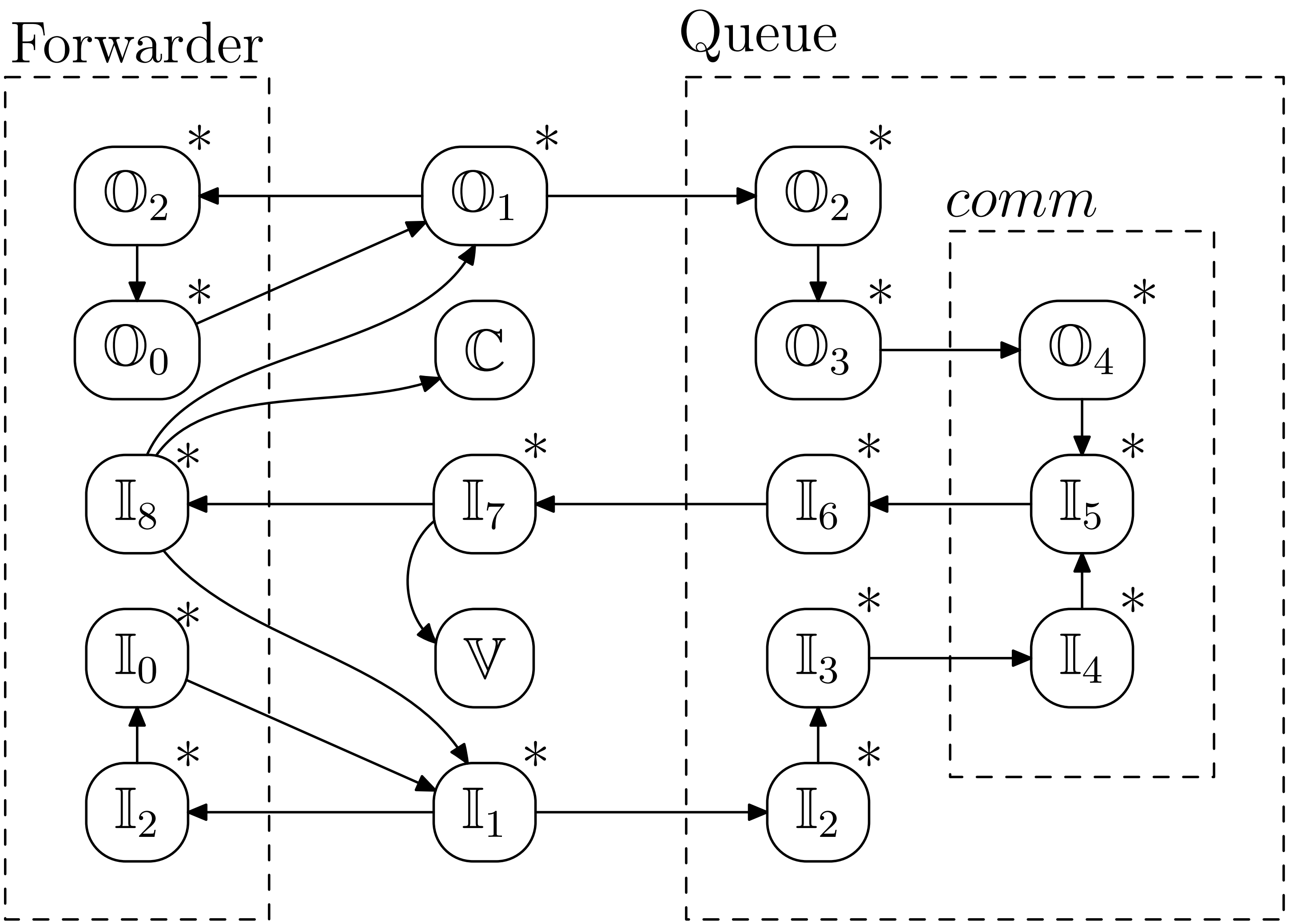
$\mathbb{O}_1/\mathbb{I}_1$-solutions route inside explicit substitutions: $\mathbb{O}_0^* \mid \mathbb{I}_0^*$

$$\ldots \mid (\uparrow .Xyz.\texttt{put}\langle\ldots\rangle @z)^* \mid ((\nu\, b)(b[\,] \mid \uparrow .Xyz.\texttt{put}\langle\ldots\rangle @z))^*$$

The final closure $\mathbb{CPI}_c$ are (roughly):

$$(\nu\, y)(\nu\, z)(\mathbb{C} \mid \mathbb{V} \mid \mathbb{O}_0^* \mid \cdots \mid \mathbb{O}_4^* \mid \mathbb{I}_0^* \mid \cdots \mid \mathbb{I}_8^*)$$

The Closure: $\mathbb{CPI}_c$



Results

Definition: May-testing congruence under $\mathbb{CPI}_c$: $\mathbb{CPI}_c \vdash A \simeq B$

For any observer $\mathcal{C}$, we require $\mathcal{C}(A) \in \mathbb{CPI}_c$ and $\mathcal{C}(B) \in \mathbb{CPI}_c$, i.e. the observer should behave like (the encoding of) a π-process.

Thus, possible interactions between $\mathcal{C}$ and A, B are limited by the shape of $\mathbb{CPI}_c$.

Lemma: All auxiliary reductions of the encoding are may-testing congruent under $\mathbb{CPI}_c$ (while the others are one-to-one correspond to reductions in the original process).

Theorem: The encoding follows, and always follows any reduction of the original process, up to the above equivalence.