Binary Session Types

Hernán Melgratti

ICC University of Buenos Aires-Conicet

Typing

Type Judgement

$$\Gamma \vdash P$$

P uses channels as specified by Γ

Environments Γ

- Partial function from polarized names to types
- Written $x_1^{p_1} : t_1, x_2^{p_2} : t_2, \ldots, x_n^{p_n} : t_n$
- Its satisfies one of the following conditions
  - $x^+, x^-, x \notin dom(\Gamma)$
  - $x \in dom(\Gamma)$ and $x^+, x^- \notin dom(\Gamma)$
  - $x^p \in dom(\Gamma)$ and $p \in \{+, -\}$ and $x^{\overline{p}}, x \notin dom(\Gamma)$
  - $x^+, x^- \in dom(\Gamma)$ and $x \notin dom(\Gamma)$

Typing

$x^{+} : ?\mathtt{int}.!\mathtt{bool}.\mathtt{end} \vdash x^{+}?(y{:}\mathtt{int}).x^{+}!\mathtt{true}.0$

$x^{+} : ?\mathtt{int}.!\mathtt{bool}.\mathtt{end} \nvdash x^{+}?(y{:}\mathtt{int}).x^{+}!y.0$

$x^{+} : ?\mathtt{int}.\mathtt{end}, y^{-} : !\mathtt{int}.\mathtt{end} \vdash x^{+}?(z{:}\mathtt{int}).y^{-}!z.0$

$x^{+} : ?\mathtt{int}.\mathtt{end}, y^{-} : !\mathtt{bool}.\mathtt{end} \nvdash 0$

$\vdash (\nu x{:}?\mathtt{int}.\mathtt{end})(x^{+}?(z{:}\mathtt{int}).0 \mid x^{-}!1.0)$

$\nvdash (\nu x{:}?\mathtt{int}.\mathtt{end})(x^{+}?(z{:}\mathtt{int}).0)$

$\nvdash (\nu x{:}?\mathtt{int}.\mathtt{end})(x^{+}?(z{:}\mathtt{int}).0 \mid x^{-}!1.0 \mid x^{-}!2.0)$

Typing

$$\not\vdash (\nu x{:}?\mathtt{int}.?\mathtt{int}.\mathtt{end})(x^{+}?(z{:}\mathtt{int}).x^{+}?(z{:}\mathtt{int}).0 \mid x^{-}!1.0 \mid x^{-}!2.0)$$

Think about

$$
\begin{aligned}
(\nu x{:}?\mathtt{int}.!\mathtt{int}.?\mathtt{int}.!\mathtt{int}.\mathtt{end})(\\
& x^{+}?(z{:}\mathtt{int}).x^{+}!(z+1).0 \mid \\
& x^{+}?(z{:}\mathtt{int}).x^{+}!(z+1).0 \mid \\
& x^{-}!1.x^{-}?(z{:}\mathtt{int}).Q_1 \mid \\
& x^{-}!2.x^{-}?(z{:}\mathtt{int}).Q_2 \qquad)
\end{aligned}
$$

Typing

$$\frac{\Gamma_1 \vdash P_1 \qquad \Gamma_2 \vdash P_2}{\Gamma_1 + \Gamma_2 \vdash P_1 | P_2}\text{[T-Par]}$$

$$x^+ : \mathtt{Tester}, x^- : \overline{\mathtt{Tester}} \vdash P_{\mathtt{server}} \mid P_{\mathtt{client}}$$

where

$\mathtt{Tester} = \mathtt{?int.!bool.end}$

$P_{\mathtt{server}} = x^+\mathtt{?}(y\mathtt{:int})\mathtt{.}x^+\mathtt{!true.0}$

$P_{\mathtt{client}} = x^-\mathtt{!1.}x^-\mathtt{?}(z\mathtt{:bool})\mathtt{.Q}$

Typing

$$\frac{\Gamma_1 \vdash P_1 \qquad \Gamma_2 \vdash P_2}{\Gamma_1 + \Gamma_2 \vdash P_1 | P_2}\,[\text{T-Par}]$$

$$x^+ : \mathtt{Tester}, x^- : \overline{\mathtt{Tester}} \nvdash P_{\mathtt{server}} \mid P_{\mathtt{client}} \mid P_{\mathtt{client}}$$

where

$\mathtt{Tester} = \mathtt{?int.!bool.end}$

$P_{\mathtt{server}} = x^+\mathtt{?}(y{:}\mathtt{int}).x^+\mathtt{!true}.0$

$P_{\mathtt{client}} = x^-\mathtt{!1}.x^-\mathtt{?}(z{:}\mathtt{bool}).\mathtt{Q}$

Typing

$$\frac{\Gamma_1 \vdash P_1 \qquad \Gamma_2 \vdash P_2}{\Gamma_1 + \Gamma_2 \vdash P_1 | P_2}\,[\text{T-Par}]$$

Context split

$$\begin{array}{rcll}
\Gamma + x^{+} : t & = & \Gamma, x^{+} : t & \text{if } x, x^{+} \not\in \mathrm{dom}(\Gamma) \\
\Gamma + x^{-} : t & = & \Gamma, x^{-} : t & \text{if } x, x^{-} \not\in \mathrm{dom}(\Gamma) \\
\Gamma + \ x : t & = & \Gamma, x : t & \text{if } x, x^{+}, x^{-} \not\in \mathrm{dom}(\Gamma) \\
(\Gamma, x : t) + \ x : t & = & \Gamma, x : t & \text{if } t \text{ is not a session type}
\end{array}$$

Extended on context as

$$\begin{array}{rcl}
\Gamma + \emptyset & = & \Gamma \\
\Gamma + (x^{p} : t, \Delta) & = & (\Gamma + x^{p} : t) + \Delta
\end{array}$$

Linear usage of session endpoints

Typing

$$\frac{\Gamma_1 \vdash P_1 \qquad \Gamma_2 \vdash P_2}{\Gamma_1 + \Gamma_2 \vdash P_1 | P_2}\text{[T-Par]} \qquad \frac{\Gamma, x^+ : S, x^- : \overline{S} \vdash P}{\Gamma \vdash (\nu x{:}S)P}\text{[T-Res]}$$

$$(\nu x{:}\mathtt{Tester})(P_{\mathtt{server}} \mid P_{\mathtt{client}})$$

where

$\mathtt{Tester} = \mathtt{?int.!bool.end}$

$P_{\mathtt{server}} = x^+\mathtt{?}(y{:}\mathtt{int}).x^+\mathtt{!true.0}$ *(faulty)*

$P_{\mathtt{client}} = x^-\mathtt{!1.}x^-\mathtt{?}(z{:}\mathtt{bool}).\mathtt{Q}$

Typing

$$\frac{\Gamma_1 \vdash P_1 \qquad \Gamma_2 \vdash P_2}{\Gamma_1 + \Gamma_2 \vdash P_1 | P_2}\,[\text{T-Par}] \qquad \frac{\Gamma, x^+ : S, x^- : \overline{S} \vdash P}{\Gamma \vdash (\nu x{:}S)P}\,[\text{T-Res}]$$

$$\frac{\Gamma, x^p : S, y : t \vdash P}{\Gamma, x^p : ?t.S \vdash x^p?(y{:}t).P}\,[\text{T-In}] \qquad \frac{\Gamma_1 \vdash v : t \qquad \Gamma_2, x^p : S \vdash P}{\Gamma_1 + (\Gamma_2, x^p : !t.S) \vdash x^p!v.P}\,[\text{T-Out}]$$

Auxiliary Typing on expressions $\Gamma \vdash v : t$

$\emptyset \vdash \mathtt{true} : \mathtt{bool} \qquad \emptyset \vdash \mathtt{false} : \mathtt{bool}$

$\emptyset \vdash () : \mathtt{unit} \qquad x^p : t \vdash x^p : t$

Typing

$$\frac{\Gamma_1 \vdash P_1 \qquad \Gamma_2 \vdash P_2}{\Gamma_1 + \Gamma_2 \vdash P_1 | P_2}\text{[T-Par]} \qquad \frac{\Gamma, x^+ : S, x^- : \overline{S} \vdash P}{\Gamma \vdash (\nu x{:}S)P}\text{[T-Res]}$$

$$\frac{\Gamma, x^p : S, y : t \vdash P}{\Gamma, x^p : \mathtt{?}t\,\mathtt{.}\,S \vdash x^p\mathtt{?}(y{:}\mathtt{t})\,\mathtt{.}\,P}\text{[T-In]} \qquad \frac{\Gamma_1 \vdash v : t \qquad \Gamma_2, x^p : S \vdash P}{\Gamma_1 + (\Gamma_2, x^p : \mathtt{!}t\,\mathtt{.}\,S) \vdash x^p\mathtt{!}v\,\mathtt{.}\,P}\text{[T-Out]}$$

$$\frac{\Gamma, x^p : S_j \vdash P \qquad j \in I}{\Gamma, x^p : \oplus[\mathtt{l}_i : S_i]_{i \in I} \vdash x^p \triangleleft \mathtt{l}_j\,\mathtt{.}\,P}\text{[T-Choice]} \qquad \frac{\Gamma, x^p : S_i \vdash P_i \quad \forall i \in I}{\Gamma, x^p : \&[\mathtt{l}_i : S_i]_{i \in I} \vdash x^p \triangleright [\mathtt{l}_i : P_i]_{i \in I}}\text{[T-Branch]}$$

$$\frac{\Gamma \text{ completed}}{\Gamma \vdash 0}\text{[T-Nil]}$$

Γ completed if $\Gamma(x^p) = S$ implies $S = \mathtt{end}$

Terminology

We say P is *well-typed* if there exists Γ s.t $\Gamma \vdash P$

Example

Is P well-typed?

- $P = x^{+}?(y{:}\mathtt{int}).x^{+}!\mathtt{true}.0$
- Yes! take $\Gamma = x^{+} : ?\mathtt{int}.!\mathtt{bool}.\mathsf{end}$

$$\dfrac{\dfrac{\emptyset \vdash \mathtt{true} : \mathtt{bool} \qquad \dfrac{x^{+} : \mathsf{end}, y : \mathtt{int} \text{ completed}}{x^{+} : \mathsf{end}, y : \mathtt{int} \vdash 0}\,\text{[T-Nil]}}{x^{+} : !\mathtt{bool}.\mathsf{end}, y : \mathtt{int} \vdash x^{+}!\mathtt{true}.0}\,\text{[T-Out]}}{x^{+} : ?\mathtt{int}.!\mathtt{bool}.\mathsf{end} \vdash x^{+}?(y{:}\mathtt{int}).x^{+}!\mathtt{true}.0}\,\text{[T-In]}$$

Example

Is P well-typed?

- $P = x^+?(y{:}!\mathtt{bool}.\mathtt{end}).y!\mathtt{true}.0$
- Yes! take $\Gamma = x^+ : ?(!\mathtt{bool}.\mathtt{end}).\mathtt{end}$

$$\dfrac{\dfrac{\emptyset \vdash \mathtt{true} : \mathtt{bool} \qquad \dfrac{x^+ : \mathtt{end}, y : \mathtt{end}\ \text{completed}}{x^+ : \mathtt{end}, y : \mathtt{end} \vdash 0}\,[\text{T-Nil}]}{x^+ : \mathtt{end}, y : !\mathtt{bool}.\mathtt{end} \vdash y!\mathtt{true}.0}\,[\text{T-Out}]}{x^+ : ?(!\mathtt{bool}.\mathtt{end}).\mathtt{end} \vdash x^+?(y{:}!\mathtt{bool}.\mathtt{end}).y!\mathtt{true}.0}\,[\text{T-In}]$$

Example

Is P well-typed

- $P = x^{+}?(y{:}!\mathtt{int}.\mathtt{end}).y!\mathtt{true}.0$

  No!

  Try with $\Gamma = x^{+} : ?(!\mathtt{int}.\mathtt{end}).\mathtt{end}$

$$
\dfrac{\emptyset \nvdash \mathtt{true} : \mathtt{int} \qquad \dfrac{x^{+} : \mathtt{end}, y : \mathtt{end}\ \text{completed}}{x^{+} : \mathtt{end}, y : \mathtt{end} \vdash 0}\,[\text{T-Nil}]}{\dfrac{x^{+} : \mathtt{end}, y : !\mathtt{int}.\mathtt{end} \vdash y!\mathtt{true}.0}{x^{+} : ?(!\mathtt{int}.\mathtt{end}).\mathtt{end} \vdash x^{+}?(y{:}!\mathtt{int}.\mathtt{end}).y!\mathtt{true}.0}\,[\text{T-In}]}\,[\text{T-Out}]
$$

Example

Is P well-typed?

- $P = x^+?(y{:}?\mathtt{int}.\mathtt{end}).y!1.0$
  No! Try with $\Gamma = x^+ : ?(?\mathtt{int}.\mathtt{end}).\mathtt{end}$

$$\dfrac{\dfrac{\text{There is a mismatch: } y : ?\mathtt{int}.\mathtt{end} \text{ and } y!1.0}{x^+ : \mathtt{end}, y : ?\mathtt{int}.\mathtt{end} \vdash y!1.0}\,[\text{T-Out}]}{x^+ : ?(?\mathtt{int}.\mathtt{end}).\mathtt{end} \vdash x^+?(y{:}?\mathtt{int}.\mathtt{end}).y!1.0}\,[\text{T-In}]$$

Rethinking $(\texttt{int} \to \texttt{bool}) \to \texttt{bool}$

$?(?\texttt{int}.!\texttt{bool}.\texttt{end}).!\texttt{bool}.\texttt{end}$ is not $(\texttt{int} \to \texttt{bool}) \to \texttt{bool}$

$g : (\texttt{int} \to \texttt{bool}) \to \texttt{bool}$
$\mathsf{g}\ \mathsf{f} = \mathsf{f}\ 1$

We may implement g as

$$P_g = x^+?(y{:}\mathtt{t_f}).y!1.y?(z{:}\texttt{bool}).x^+!z.0$$

where $\mathtt{t_f} = ?\texttt{int}.!\texttt{bool}.\texttt{end}$

But

$$x^+ : ?\mathtt{t_f}.!\texttt{bool}.\texttt{end} \nvdash P_g$$

However

$$x^+ : ?\overline{\mathtt{t_f}}.!\texttt{bool}.\texttt{end} \vdash P_g$$

What about $?(!\mathtt{int}.?\mathtt{bool}.\mathtt{end}).!\mathtt{bool}.\mathtt{end}$?

$g : (\mathtt{int} \to \mathtt{bool}) \to \mathtt{bool}$
g y = (y 1) & (y 2)

We may implement g as

$$P_g = x^+?(y{:}\overline{\mathtt{t_f}}).y!1.y?(z_1{:}\mathtt{bool}).y!2.y?(z_2{:}\mathtt{bool}).x^+!z_1\&z_2.0$$

where $\mathtt{t_f} = ?\mathtt{int}.!\mathtt{bool}.\mathtt{end}$

However

$$x^+ : ?\overline{\mathtt{t_f}}.!\mathtt{bool}.\mathtt{end} \nvdash P_g$$

The parameter y **must** be used just for **one** application

On linearity

- Consider $P = x^{+}\,!\,y^{+}\,.\,y^{+}\,!\,1\,.\,0$.
- Does the following hold?

$$\Gamma, x^{+} : !(!\mathtt{int}.\mathtt{end}).\mathtt{end}, y^{+} : !\mathtt{int}.\mathtt{end} \vdash P$$

$$\frac{\Gamma_1 \vdash v : t \qquad \Gamma_2, x^{p} : S \vdash P}{\Gamma_1 + (\Gamma_2, x^{p} : !t.S) \vdash x^{p}\,!\,v\,.\,P}\,\text{[T-Out]}$$

- No. Why does [T-Out] ban P?
- Take

$$Q = (\nu y{:}!\mathtt{int}.\mathtt{end})(\nu x{:}!(!\mathtt{int}.\mathtt{end}).\mathtt{end})(\\ P \mid x^{-}?(z{:}!\mathtt{int}.\mathtt{end}).z\,!\,2\,.\,0 \mid y^{-}?(z{:}\mathtt{int}).0)$$

- $Q \xrightarrow{\tau,-} Q'$ where

$$Q' = (\nu y{:}!\mathtt{int}.\mathtt{end})(\nu x{:}\mathtt{end})(y^{+}\,!\,1\,.\,0 \mid y^{+}\,!\,2\,.\,0 \mid y^{-}?(z{:}\mathtt{int}).0)$$

  where two processes concurrently send on y^{+}
- A process does not use a session endpoint after *delegating* it (i.e., sending it over a different session endpoint)

Results

Theorem (Type Preservation)

- *If* $\Gamma \vdash P$ *and* $P \xrightarrow{\tau,-} Q$ *then* $\Gamma \vdash Q$.
- *If* $\Gamma, x^p : S, x^{\overline{p}} : \overline{S} \vdash P$ *and* $P \xrightarrow{x,\mathtt{l}} Q$ *then* $S \xrightarrow{\mathtt{l}} T$ *and* $\Gamma, x^p : T, x^{\overline{p}} : \overline{T} \vdash Q$.

Theorem (Type Safety)

Let $\Gamma \vdash P$ *where* Γ *balanced* [2]

- *If* $P \equiv (\nu \tilde{z}{:}\tilde{S})(x^p ! v \,.\, P_1 \mid x^{\overline{p}} ?(y{:}t) \,.\, P_2 \mid Q)$ *with* $p \in \{+, -\}$ *then* $x, x^+, x^- \notin \mathrm{fn}(Q)$ *and* $\Gamma, \tilde{z} : \tilde{S} \vdash v : t$
- *If* $P \equiv (\nu \tilde{z}{:}\tilde{S})(x^p \triangleleft \mathtt{l}_j \,.\, R \mid x^{\overline{p}} \triangleright [\mathtt{l}_i : P_i]_{i \in I} \mid Q)$ *with* $p \in \{+, -\}$ *then* $x, x^+, x^- \notin \mathrm{fn}(Q)$ *and* $j \in I$.

[2] Γ is balanced if $x^p : S$ and $x^{\overline{p}} : T$ implies $S = \overline{T}$

Properties

Does the following hold?

$$\vdash (\nu x{:}?\mathtt{int}.\mathtt{end})(x^{+}?(z{:}\mathtt{int}).x^{-}!z.0)$$

Yes!

- The process is well-typed and deadlocked

The type system ensures

- *Type Safety* in communication (e.g., received values are of the expected type)
- *Session Fidelity* (e.g., communication follows the flow described by the session type)
- The type system does not ensure deadlock-freedom

Deadlock

Deadlocked Process

$$P = x^{+}?(z{:}\mathtt{int}).y^{-}!1.0 \quad | \quad y^{+}?(z{:}\mathtt{int}).x^{-}!1.0$$

Is P well-typed?

$$\vdash (\nu x{:}?\mathtt{int}.\mathtt{end})(\nu y{:}?\mathtt{int}.\mathtt{end})P$$

Yes!

- The process is well-typed and deadlocked
- The type system does not check the dependencies between different sessions

Deadlock-freedom by design (linear logic approaches)

- Connection drawn between linear logic and session-typed pi-calculus gave rise to type systems that guarantee deadlock-freedom
  - Luís Caires, Frank Pfenning: Session Types as Intuitionistic Linear Propositions. CONCUR 2010.
  - Philip Wadler: Propositions as sessions. ICFP 2012.
- The type system imposes some structural constraint on programs
  - two processes share at most one channel
  - Hence, there are no circular dependencies