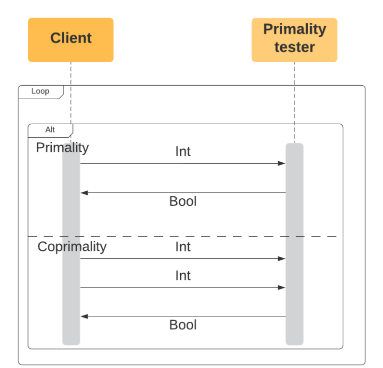
Binary Session Types (infinite behaviour)

Hernán Melgratti

ICC – Universidad de Buenos Aires – Conicet

Example

Infinite interactions (with equations)



- $\mathtt{Tester} = \&[\mathtt{Pr} : ?\mathtt{int}.!\mathtt{bool}.\mathtt{Tester},$
  $\mathtt{Co} : ?\mathtt{int}.?\mathtt{int}.!\mathtt{bool}.\mathtt{Tester}]$
- $\mathtt{Client} = \oplus[\mathtt{Pr} : !\mathtt{int}.?\mathtt{bool}.\mathtt{Client},$
  $\mathtt{Co} : !\mathtt{int}.!\mathtt{int}.?\mathtt{bool}.\mathtt{Client}]$

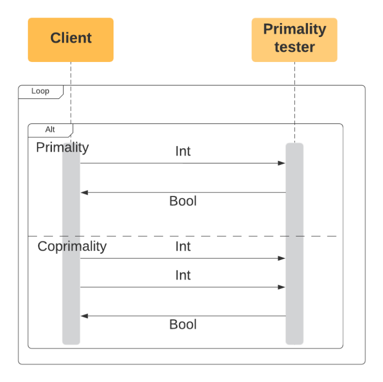
Syntax of Types

Session Types

$$
\begin{array}{rcll}
S, T & ::= & \mathtt{end} & \text{terminated session} \\
& | & ?t.S & \text{receive (input)} \\
& | & !t.S & \text{send (output)} \\
& | & \&[\mathtt{l}_i : T_i]_{i \in I} & \text{branch} \\
& | & \oplus[\mathtt{l}_i : T_i]_{i \in I} & \text{select} \\
& | & \mu X.S & \text{recursive session type} \\
& | & X & \text{session type variable} \\
s, t & ::= & \mathrm{S} & \text{A session type} \\
& | & \mathtt{int}, \mathtt{bool} & \text{basic types} \\
& | & \ldots & \text{other types} \\
& & & \\
\mathcal{L} & = & \{\mathtt{l}, \mathtt{l}_1, \ldots\} & \text{Set of labels}
\end{array}
$$

Example

Infinite interactions (without equations)



- $\texttt{Tester} = \mu X.\&[\texttt{Pr} : ?\texttt{int}.!\texttt{bool}.X, \quad \texttt{Co} : ?\texttt{int}.?\texttt{int}.!\texttt{bool}.X]$
- $\texttt{Client} = \mu X.\oplus[\texttt{Pr} : !\texttt{int}.?\texttt{bool}.X, \quad \texttt{Co} : !\texttt{int}.!\texttt{int}.?\texttt{bool}.X]$

Important

Contractive types

Recursive types are required to be *contractive*, i.e., containing no subexpressions of the form $\mu X\,.\,\mu X_1\,.\,\ldots\,\mu X_n\,.\,X$

Duality (naive, classical definition)

$\overline{S}$ is the dual of S

$$
\begin{aligned}
\overline{\mathsf{end}} &= \mathsf{end}\\
\overline{?t.S} &= !t.\overline{S}\\
\overline{!t.S} &= ?t.\overline{S}\\
\overline{\&[\mathtt{l}_i : T_i]_{i\in I}} &= \oplus[\mathtt{l}_i : \overline{T_i}]_{i\in I}\\
\overline{\oplus[\mathtt{l}_i : T_i]_{i\in I}} &= \&[\mathtt{l}_i : \overline{T_i}]_{i\in I}\\
\overline{\mu X.T} &= \mu X.\overline{T}\\
\overline{X} &= X
\end{aligned}
$$

Problem with the naive definition[1,2]

Communication of a recursive type

$S = \mu X.\,!X.X$
$\overline{S} = \mu X.\,?X.X$

Mismatch: S sends S, but $\overline{S}$ receives $\overline{S}$

Quick fix

- Recursion variables only occur in tail position in a session type
- Results in several papers do not hold for recursive types occurring in non-tail position.
- Alternatively, infinite terms for recursive types (coinductive definition)

[1] Giovanni Bernardi and Matthew Hennessy: Using Higher-Order Contracts to Model Session Types (Extended Abstract). CONCUR 2014

[2] Simon J. Gay, Peter Thiemann and Vasco T. Vasconcelos. Duality of Session Types: The Final Cut. Places 2020.

Syntax of Processes

Polarities

$p ::= + \mid - \mid \epsilon$ Optional polarities

Values (more in general expressions)

$$
\begin{array}{rll}
v, w ::= & x^p, y^q, \ldots & \text{(polarised) variables } \mathcal{X} = \{x, y, \ldots\} \\
\mid & () & \text{unit value} \\
\mid & \mathtt{true}, \mathtt{false} & \text{boolean values} \\
\mid & \ldots & \text{expressions}
\end{array}
$$

Processes

$$
\begin{array}{rll}
P, Q ::= & 0 & \text{terminated process} \\
\mid & x^p ?(y{:}\mathtt{t}) . P & \text{input} \\
\mid & x^p ! v . P & \text{output} \\
\mid & x^p \triangleright [\mathtt{l}_i : P_i]_{i \in I} & \text{branch} \\
\mid & x^p \triangleleft \mathtt{l} . P & \text{select} \\
\mid & P|Q & \text{parallel composition} \\
\mid & (\nu x{:}S)P & \text{channel creation} \\
\mid & !P & \text{replication}
\end{array}
$$

Example

Recursive service

$$P = !(loop?(x{:}\mathsf{S}).x \triangleright [next : loop!x.0, end : 0]) \mid loop!x.Q$$
$$S = \mu X.\&[next : X, end : \mathsf{end}]$$

Syntax of Types

Session Types

$S, T ::=$		$\mathtt{end}$	terminated session
	$\mid$	$?t.S$	receive (input)
	$\mid$	$!t.S$	send (output)
	$\mid$	$\&[\mathtt{l}_i : T_i]_{i \in I}$	branch
	$\mid$	$\oplus[\mathtt{l}_i : T_i]_{i \in I}$	select
	$\mid$	$\mu X.S$	recursive session type
	$\mid$	X	session type variable
$s, t ::=$		S	A session type
	$\mid$	$\mathtt{int}, \mathtt{bool}$	basic types
	$\mid$	$\ldots$	other types

Operational semantics

$$x^p\,!\,v\,.\,P \mid x^{\overline{p}}?(y{:}\mathtt{t})\,.\,Q \xrightarrow{x,-} P \mid Q\{v/y\} \text{ [R-Comm]}$$

Substitution

$$\begin{aligned}
x\{v/x\} &= v \\
x^p\{v/y\} &= x^p && \text{if } x \neq y \\
\\
0\{v/y\} &= 0 \\
(P|Q)\{v/y\} &= P\{v/y\}|Q\{v/y\} \\
(x^p?(z{:}\mathtt{t}).P)\{v/y\} &= x^p\{v/y\}?(z{:}\mathtt{t}).P\{v/y\} && \text{if } z \notin \mathrm{fn}(v) \cup \{y\}
\end{aligned}$$

Free names

fn

$$
\begin{aligned}
\mathrm{fn}(\mathtt{true}) = \mathrm{fn}(\mathtt{false}) \;=\; \mathrm{fn}(()) \;=\; & \emptyset \\
\mathrm{fn}(x^p) =\; & \{x^p\} \\
\\
\mathrm{fn}(0) =\; & \emptyset \\
\mathrm{fn}(P|Q) =\; & \mathrm{fn}(Q) \cup \mathrm{fn}(P) \\
\mathrm{fn}(x^p\,?(y{:}\mathtt{t})\,.\,P) =\; & \{x^p\} \cup (\mathrm{fn}(P) \setminus \{y\}) \\
\mathrm{fn}(x^p\,!\,v\,.\,P) =\; & \{x^p\} \cup \mathrm{fn}(v) \cup \mathrm{fn}(P) \\
\mathrm{fn}(x^p \triangleright [\mathtt{l}_i : P_i]_{i\in I}) =\; & \{x^p\} \cup (\bigcup_i \mathrm{fn}(P_i)) \\
\mathrm{fn}(x^p \triangleleft \mathtt{l}\,.\,P) =\; & \{x^p\} \cup \mathrm{fn}(P) \\
\mathrm{fn}((\nu x{:}S)P) =\; & \mathrm{fn}(P) \setminus \{x, x^+, x^-\} \\
\mathrm{fn}(!\,P) =\; & \mathrm{fn}(P)
\end{aligned}
$$

Operational semantics

$$x^{p}\,!\,v\,.\,P \mid x^{\overline{p}}\,?(y{:}\mathtt{t})\,.\,Q \xrightarrow{x,-} P \mid Q\{v/y\} \quad \text{[R-Comm]}$$

Substitution

$$
\begin{array}{rcll}
x\{v/x\} & = & v & \\
x^{p}\{v/y\} & = & x^{p} & \text{if } x \neq y \\
 & & & \\
0\{v/y\} & = & 0 & \\
(P|Q)\{v/y\} & = & P\{v/y\}|Q\{v/y\} & \\
(x^{p}\,?(z{:}\mathtt{t})\,.\,P)\{v/y\} & = & x^{p}\{v/y\}\,?(z{:}\mathtt{t})\,.\,P\{v/y\} & \text{if } z \notin \mathrm{fn}(v) \cup \{y\} \\
(x^{p}\,!\,w\,.\,P)\{v/y\} & = & x^{p}\{v/y\}\,!\,w\{v/y\}\,.\,P\{v/y\} & \\
(x^{p} \triangleright [\mathtt{l}_i : P_i]_{i\in I})\{v/y\} & = & x^{p}\{v/y\} \triangleright [\mathtt{l}_i : P_i\{v/y\}]_{i\in I} & \\
(x^{p} \triangleleft \mathtt{l}\,.\,P)\{v/y\} & = & x^{p}\{v/y\} \triangleleft \mathtt{l}\,.\,P\{v/y\} & \\
((\nu x{:}S)P)\{v/y\} & = & (\nu x{:}S)P\{v/y\} & \text{if } x \notin \mathrm{fn}(v) \cup \{y\} \\
(!\,P)\{v/y\} & = & !\,(P\{v/y\}) &
\end{array}
$$

Operational semantics

$$x^p\,!\,v\,.\,P \mid x^{\overline{p}}?(y{:}\mathtt{t})\,.\,Q \xrightarrow{x,-} P \mid Q\{v/y\} \text{ [R-Comm]}$$

$$x^p \triangleleft \mathtt{l}_i\,.\,P \mid x^{\overline{p}} \triangleright [\mathtt{l}_j : Q_j]_{j\in I} \xrightarrow{x,\mathtt{l}_i} P \mid Q_i$$

Operational semantics

$$x^p\,!\,v\,.\,P \mid x^{\overline{p}}\,?(y{:}\mathtt{t})\,.\,Q \xrightarrow{x,-} P \mid Q\{v/y\} \;\text{[R-Comm]}$$

$$\frac{i \in I}{x^p \triangleleft \mathtt{l}_i\,.\,P \mid x^{\overline{p}} \triangleright [\mathtt{l}_j : Q_j]_{j \in I} \xrightarrow{x,\mathtt{l}_i} P \mid Q_i}\;\text{[R-Select]}$$

Operational semantics

$$x^p\,!\,v\,.\,P \mid x^{\overline{p}}\,?(y{:}\mathtt{t})\,.\,Q \xrightarrow{x,-} P \mid Q\{v/y\} \quad \text{[R-Comm]}$$

$$\frac{p \in \{+,-\} \qquad\qquad i \in I}{x^p \triangleleft \mathtt{l}_i\,.\,P \mid x^{\overline{p}} \triangleright [\mathtt{l}_j : Q_j]_{j\in I} \xrightarrow{x,\mathtt{l}_i} P \mid Q_i}\ \text{[R-Select]}$$

$$\frac{P \xrightarrow{x,\mathtt{l}} P' \qquad\qquad S \xrightarrow{\mathtt{l}} T}{(\nu x{:}S)P \xrightarrow{\tau,-} (\nu x{:}T)P'}\ \text{[R-NewS]}$$

Semantics of Types

$$?t\,.\,S \xrightarrow{-} S \qquad\qquad !t\,.\,S \xrightarrow{-} S$$

$$\&[\mathtt{l}_i : T_i]_{i\in I} \xrightarrow{\mathtt{l}_i} T_i \qquad\qquad \oplus[\mathtt{l}_i : T_i]_{i\in I} \xrightarrow{\mathtt{l}_i} T_i$$

$$\frac{S\{\mu X\,.\,S/X\} \xrightarrow{\beta} T}{\mu X\,.\,S \xrightarrow{\beta} T}$$

Operational semantics

$$x^p\,!\,v\,.\,P \mid x^{\overline{p}}?(y{:}\mathtt{t})\,.\,Q \xrightarrow{x,-} P \mid Q\{v/y\}\ \text{[R-Comm]}$$

$$\frac{i \in I}{x^p \triangleleft \mathtt{l}_i\,.\,P \mid x^{\overline{p}} \triangleright [\mathtt{l}_j : Q_j]_{j\in I} \xrightarrow{x,\mathtt{l}_i} P \mid Q_i}\ \text{[R-Select]}$$

$$\frac{P \xrightarrow{x,\mathtt{l}} P' \qquad S \xrightarrow{x,\mathtt{l}} T}{(\nu x{:}S)P \xrightarrow{\tau,-} (\nu x{:}T)P'}\ \text{[R-NewS]}$$

$$\frac{P \xrightarrow{\alpha,\mathtt{l}} P' \qquad \alpha \neq x}{(\nu x{:}S)P \xrightarrow{\alpha,\mathtt{l}} (\nu x{:}S)P'}\ \text{[R-New]}$$

$$\frac{P \xrightarrow{\alpha,\mathtt{l}} P'}{P|Q \xrightarrow{\alpha,\mathtt{l}} P'|Q}\ \text{[R-Par]}$$

Structural equivalence

$$
\begin{aligned}
P|0 &\equiv P \\
P|Q &\equiv Q|P \\
(P|Q)|R &\equiv Q|(P|R) \\
(\nu x{:}S)(\nu y{:}T)P &\equiv (\nu y{:}T)(\nu x{:}S)P \\
(\nu x{:}S)P|Q &\equiv (\nu x{:}S)(P|Q) && \text{if } x^p \notin \mathrm{fn}(Q) \\
(\nu x{:}S)0 &\equiv 0 && \text{if } S = \mathbf{end} \\
!P &\equiv P|\ !P
\end{aligned}
$$

Operational semantics

$$x^{p}\,\mathtt{!}\,v\,\mathtt{.}\,P \mid x^{\overline{p}}\mathtt{?}(y{:}\mathtt{t})\,\mathtt{.}\,Q \xrightarrow{x,-} P \mid Q\{v/y\} \;\text{[R-Comm]}$$

$$\frac{i \in I}{x^{p} \triangleleft \mathtt{l}_i\,\mathtt{.}\,P \mid x^{\overline{p}} \triangleright [\mathtt{l}_j : Q_j]_{j\in I} \xrightarrow{x,\mathtt{l}_i} P \mid Q_i}\;\text{[R-Select]}$$

$$\frac{P \xrightarrow{x,\mathtt{l}} P' \qquad S \xrightarrow{x,\mathtt{l}} T}{(\nu x{:}S)P \xrightarrow{\tau,-} (\nu x{:}T)P'}\;\text{[R-NewS]}$$

$$\frac{P \xrightarrow{\alpha,\mathtt{l}} P' \qquad \alpha \neq x}{(\nu x{:}S)P \xrightarrow{\alpha,\mathtt{l}} (\nu x{:}S)P'}\;\text{[R-New]}$$

$$\frac{P \xrightarrow{\alpha,\mathtt{l}} P'}{P|Q \xrightarrow{\alpha,\mathtt{l}} P'|Q}\;\text{[R-Par]}$$

$$\frac{P \equiv Q \qquad Q \xrightarrow{\alpha,\mathtt{l}} Q' \qquad Q' \equiv P'}{P \xrightarrow{\alpha,\mathtt{l}} P'}\;\text{[R-Cong]}$$

Typing

Type Judgement

$$\Gamma \vdash P$$

P uses channels as specified by Γ

Structural equivalence

$$
\begin{aligned}
P|0 &\equiv P \\
P|Q &\equiv Q|P \\
(P|Q)|R &\equiv Q|(P|R) \\
(\nu x{:}S)(\nu y{:}T)P &\equiv (\nu y{:}T)(\nu x{:}S)P \\
(\nu x{:}S)P|Q &\equiv (\nu x{:}S)(P|Q) && \text{if } x^p \notin \mathrm{fn}(Q) \\
(\nu x{:}S)0 &\equiv 0 && \text{if } S = \mathbf{end} \\
!P &\equiv P|\ !P
\end{aligned}
$$

$$
\begin{aligned}
P &= \,!(\mathit{loop}?(x{:}S).x \triangleright [\mathit{next} : \mathit{loop}!x.0, \mathit{end} : 0]) \mid \mathit{loop}!x.Q \\
S &= \mu X.\&[\mathit{next} : X, \mathit{end} : \mathbf{end}]
\end{aligned}
$$

Would it be possible to assign a session type to loop?

Syntax of Types

Session Types

$S, T ::=$		$\mathtt{end}$	terminated session
	$\mid$	$?t.S$	receive (input)
	$\mid$	$!t.S$	send (output)
	$\mid$	$\&[\mathtt{l}_i : T_i]_{i \in I}$	branch
	$\mid$	$\oplus[\mathtt{l}_i : T_i]_{i \in I}$	select
	$\mid$	$\mu X.S$	recursive session type
	$\mid$	X	session type variable
$s, t ::=$		S	A session type
	$\mid$	$\mathtt{int}$, $\mathtt{bool}$	basic types
	$\mid$	[T]	shared channels
	$\mid$	…	other types

Typing

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

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

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

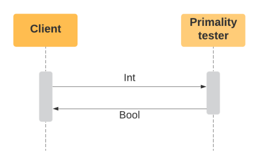
$$\frac{\Gamma, x^p : S_j \vdash P \qquad j \in I}{\Gamma, x^p : \oplus[\mathtt{l}_i : S_i]_{i\in I} \vdash x^p \triangleleft \mathtt{l}_j.P}\ \text{[T-Choice]} \qquad \frac{\Gamma, x^p : S_i \vdash P_i \quad \forall i \in I}{\Gamma, x^p : \&[\mathtt{l}_i : S_i]_{i\in I} \vdash x^p \triangleright [\mathtt{l}_i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

$$\frac{\Gamma \text{ completed}}{\Gamma \vdash 0}\ \text{[T-Nil]} \qquad \frac{\Gamma \vdash P \quad \Gamma \text{ Unlimited}}{\Gamma \vdash !P}\ \text{[T-Rep]}$$

Γ Unlimited iff $\forall x \in \mathrm{dom}(\Gamma).\Gamma(x) \notin T$

Unlimited channels for services

Tester



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

- We give an implementation over the session endpoints x^+ (for the server) and x^- (for the client)
  $P_{\mathtt{server}} = x^+\mathtt{?}(y{:}\mathtt{int}).x^+\mathtt{!true}.0$
  $P_{\mathtt{client}} = x^-\mathtt{!1}.x^-\mathtt{?}(z{:}\mathtt{bool}).\mathtt{Q}$
- The system is the parallel composition

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

- Alternatively,

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

$$P_{\mathtt{server}} \mid (\nu x{:}\mathtt{Tester})(server\mathtt{!}x^+.P_{\mathtt{client}}) \mid (\nu x{:}\mathtt{Tester})(server\mathtt{!}x^+.\ldots)$$