Multiparty session types

Hernán Melgratti

ICC University of Buenos Aires-Conicet

26 February 2020 @ Pisa

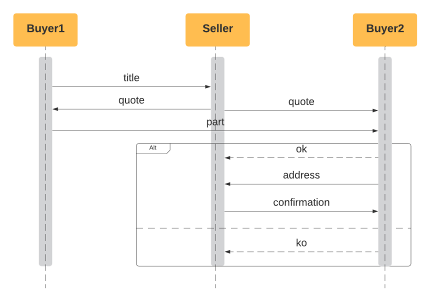
Multiparty session types[1]

- Extension of binary session types to multiparty sessions
- Asynchronous communications
- Interactions are abstracted as a global scenario, namely, Global types
  - specify dependencies and causal chains of multiparty asynchronous interactions

[1] Honda, K., Yoshida, N., & Carbone, M. Multiparty asynchronous session types. POPL 2008

Global Graph (Choreography)

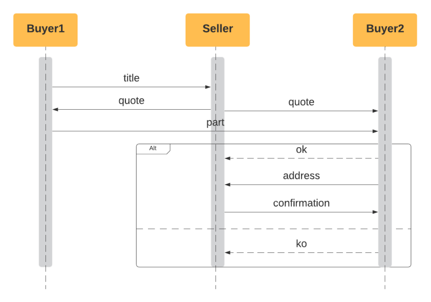
Two Buyers Protocol



$$\mathtt{TBProt} = \begin{array}{l} \mathtt{b1}{\rightarrow}\mathtt{s} : \langle \mathtt{string} \rangle. \\ \mathtt{s}{\rightarrow}\mathtt{b1} : \langle \mathtt{float} \rangle. \\ \mathtt{s}{\rightarrow}\mathtt{b2} : \langle \mathtt{float} \rangle. \\ \mathtt{b1}{\rightarrow}\mathtt{b2} : \langle \mathtt{float} \rangle. \\ \mathtt{b2}{\rightarrow}\mathtt{s} : \left\{ \begin{array}{l} ok : \mathtt{b2}{\rightarrow}\mathtt{s} : \langle \mathtt{string} \rangle.\mathtt{s}{\rightarrow}\mathtt{b2} : \langle \mathtt{string} \rangle.\mathtt{end}, \\ ko : \mathtt{end} \end{array} \right\} \end{array}$$

Local Types

Two Buyers Protocol



$$\texttt{Buyer1} \; = \; \texttt{!string.?float.!float.end}$$

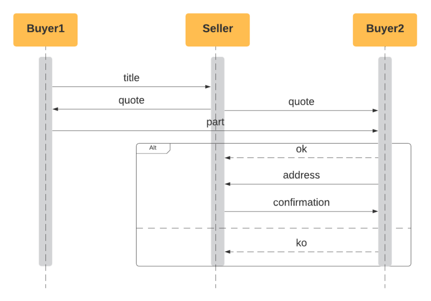
- The first message (string) is for the Seller and the second one (float) is for Buyer2? Information absent in the local type
- There are alternatives:
  - *a là* communicating machines: one channel for each pair in each direction

    $$\texttt{Buyer1} = b_1 s\texttt{!}\langle\texttt{string}\rangle\texttt{.}sb_1\texttt{?}\langle\texttt{float}\rangle\texttt{.}b_1 b_2\texttt{!}\langle\texttt{float}\rangle\texttt{.end}$$

  - Decorated global graphs

Global Graph (Choreography)

Two Buyers Protocol



$$
\begin{aligned}
\texttt{TBProt} &= \texttt{b1}{\to}\texttt{s} : x\,\langle\texttt{string}\rangle.\\
&\quad \texttt{s}{\to}\texttt{b1} : y\,\langle\texttt{float}\rangle.\\
&\quad \texttt{s}{\to}\texttt{b2} : z_1\,\langle\texttt{float}\rangle.\\
&\quad \texttt{b1}{\to}\texttt{b2} : z_2\,\langle\texttt{float}\rangle.\\
&\quad \texttt{b2}{\to}\texttt{s} : x \left\{ \begin{array}{l} ok : \texttt{b2}{\to}\texttt{s} : x\,\langle\texttt{string}\rangle.\texttt{s}{\to}\texttt{b2} : z_1\,\langle\texttt{string}\rangle.\texttt{end}, \\ ko : \texttt{end} \end{array} \right\}
\end{aligned}
$$

$$
\texttt{Buyer1} = x!\langle\texttt{string}\rangle.y?\langle\texttt{float}\rangle.z_2!\langle\texttt{float}\rangle.\texttt{end}
$$

First-order, finite MST

Syntax

$$
\begin{array}{rlll}
\eta ::= & & \mathsf{p}{\to}\mathsf{q} : x & \text{action} \\
\mathsf{G} ::= & & \eta\langle\tilde{\mathsf{S}}\rangle.\mathsf{G} & \text{interaction} \\
& \mid & \eta\{l_j : \mathsf{G}_j\}_{j \in J} & \text{branch} \\
& \mid & \mathsf{G} \mid \mathsf{G} & \text{parallel} \\
& \mid & \mathbf{end} & \text{termination} \\
\mathsf{S} ::= & & \mathtt{int} \mid \mathtt{unit} \mid \mathtt{bool} \mid \ldots & \text{basic sorts}
\end{array}
$$

- p, r, … : participants (also roles)
- x, y, …: communication channels
- l, … : labels
- $\tilde{\_}$: tuples

Realizations

Example

$$\begin{aligned}
\mathtt{G} &= \mathtt{p}{\rightarrow}\mathtt{q} : x\,\langle \mathtt{int}\rangle.\mathtt{p}{\rightarrow}\mathtt{r} : y\,\langle \mathtt{bool}\rangle.\mathtt{end} \\
P_p &= x\,!\,1.y\,!\,\mathtt{true}.\mathbf{0} \\
P_q &= x\,?(i).\mathbf{0} \\
P_r &= y\,?(j).\mathbf{0}
\end{aligned}$$

Realizations

Example

$$
\begin{aligned}
\mathtt{G} &= \mathtt{p}{\rightarrow}\mathtt{q} : x\,\langle \mathtt{int} \rangle.\mathtt{p}{\rightarrow}\mathtt{r} : x\,\langle \mathtt{bool} \rangle.\mathtt{end} \\
P_p &= x\mathtt{!}1.x\mathtt{!}\mathtt{true}.\mathbf{0} \\
P_q &= x\mathtt{?}(i).\mathbf{0} \\
P_r &= x\mathtt{?}(j).\mathbf{0}
\end{aligned}
$$

Realizations

Example

$$\begin{aligned}
\mathtt{G} &= \mathtt{p}{\rightarrow}\mathtt{q} : x\,\langle\mathtt{int}\rangle.\mathtt{p}{\rightarrow}\mathtt{r} : x\,\langle\mathtt{bool}\rangle.\mathtt{end} \\
P_p &= x\,!\,1.x\,!\,\mathtt{true}.\mathbf{0} \\
P_q &= x\,?(i).\mathbf{0} \\
P_r &= x\,?(j).\mathbf{0}
\end{aligned}$$

We cannot ensure that P_q gets 1 and P_r gets `true` (race on x)
Hence, `G` is bad (Output-to-Output bad)

Realizations

Example

$$
\begin{aligned}
\mathtt{G} &= \mathtt{p}{\to}\mathtt{q} : x\,\langle \mathtt{int} \rangle.\mathtt{r}{\to}\mathtt{q} : y\,\langle \mathtt{bool} \rangle.\mathtt{end} \\
P_p &= x\,!\,1.\mathbf{0} \\
P_q &= x\,?(i).y\,?(j).\mathbf{0} \\
P_r &= y\,!\,\mathtt{true}.\mathbf{0}
\end{aligned}
$$

Realizations

Example

$$\begin{aligned}
\mathtt{G} &= \mathtt{p}{\to}\mathtt{q} : x \, \langle \mathtt{int} \rangle \mathtt{.} \mathtt{r}{\to}\mathtt{q} : y \, \langle \mathtt{bool} \rangle \mathtt{.end} \\
P_p &= x\,\mathtt{!}\,1\mathtt{.}\mathbf{0} \\
P_q &= x\,\mathtt{?}(i)\mathtt{.}\,y\,\mathtt{?}(j)\mathtt{.}\mathbf{0} \\
P_r &= y\,\mathtt{!}\,\mathtt{true}\mathtt{.}\mathbf{0}
\end{aligned}$$

No races on channels x and y

Hence, $\mathtt{G}$ is good (Input-to-Input good)

Realizations

Example

$$
\begin{aligned}
\mathtt{G} &= \mathtt{p}{\to}\mathtt{q} : x\,\langle \mathtt{int}\rangle.\mathtt{r}{\to}\mathtt{q} : x\,\langle \mathtt{bool}\rangle.\mathtt{end} \\
P_p &= x!1.\mathbf{0} \\
P_q &= x?(i).x?(j).\mathbf{0} \\
P_r &= x!\mathtt{true}.\mathbf{0}
\end{aligned}
$$

Realizations

Example

$$\begin{aligned}
\mathtt{G} &= \mathtt{p}{\to}\mathtt{q} : x\,\langle \mathtt{int} \rangle.\mathtt{r}{\to}\mathtt{q} : x\,\langle \mathtt{bool} \rangle.\mathtt{end} \\
P_p &= x!1.\mathbf{0} \\
P_q &= x?(i).x?(j).\mathbf{0} \\
P_r &= x!\mathtt{true}.\mathbf{0}
\end{aligned}$$

Race on x

Hence, $\mathtt{G}$ is bad (Input-to-Input bad)

Realizations

Example

$$
\begin{aligned}
\mathtt{G} &= \mathtt{p}{\to}\mathtt{q} : x\,\langle\mathtt{int}\rangle.\mathtt{q}{\to}\mathtt{r} : x\,\langle\mathtt{int}\rangle.\mathtt{end}\\
P_p &= x\,!\,1.\mathbf{0}\\
P_q &= x?(i).x\,!\,i.\mathbf{0}\\
P_r &= x?(i).\mathbf{0}
\end{aligned}
$$

Realizations

Example

$$
\begin{aligned}
\mathtt{G} &= \mathtt{p}{\rightarrow}\mathtt{q} : x\,\langle\mathtt{int}\rangle.\mathtt{q}{\rightarrow}\mathtt{r} : x\,\langle\mathtt{int}\rangle.\mathtt{end} \\
P_p &= x!1.\mathbf{0} \\
P_q &= x?(i).x!i.\mathbf{0} \\
P_r &= x?(i).\mathbf{0}
\end{aligned}
$$

Race on x

Hence, G is bad (Input-to-Output bad)

Realizations

Example

$$
\begin{aligned}
\mathtt{G} &= \mathsf{p}{\to}\mathsf{q} : x \,\langle \mathtt{int} \rangle . \mathsf{q}{\to}\mathsf{r} : y \,\langle \mathtt{int} \rangle . \mathsf{end} \\
P_p &= x\,!\,1.\mathbf{0} \\
P_q &= x\,?(i).y\,!\,i.\mathbf{0} \\
P_r &= y\,?(i).\mathbf{0}
\end{aligned}
$$

Realizations

Example

$$\begin{aligned}
\mathtt{G} &= \mathtt{p}\to\mathtt{q} : x\,\langle\mathtt{int}\rangle.\mathtt{q}\to\mathtt{r} : y\,\langle\mathtt{int}\rangle.\mathtt{end}\\
P_p &= x!1.\mathbf{0}\\
P_q &= x?(i).y!i.\mathbf{0}\\
P_r &= y?(i).\mathbf{0}
\end{aligned}$$

No Races on x and y

Hence, $\mathtt{G}$ is good (Input-to-Output good)

Realizations

Example

$$
\begin{aligned}
\mathtt{G} &= \mathtt{p}{\to}\mathtt{q} : x\,\langle \mathtt{int} \rangle . \mathtt{p}{\to}\mathtt{q} : x\,\langle \mathtt{bool} \rangle . \mathtt{end} \\
P_p &= x\,!\,1 . x\,!\,\mathtt{true} . \mathbf{0} \\
P_q &= x\,?(i) . y\,?(j) . \mathbf{0}
\end{aligned}
$$

Realizations

Example

$$
\begin{aligned}
\mathtt{G} &= \mathtt{p}{\to}\mathtt{q} : x\,\langle \mathtt{int}\rangle.\mathtt{p}{\to}\mathtt{q} : x\,\langle \mathtt{bool}\rangle.\mathtt{end} \\
P_p &= x!1.x!\mathtt{true}.\mathbf{0} \\
P_q &= x?(i).y?(j).\mathbf{0}
\end{aligned}
$$

No Races on x
Hence, $\mathtt{G}$ is good (Input-to-Input, Output-to-Output good)

Realizations

Example

$$\begin{aligned}
\mathtt{G} &= \mathtt{p}{\to}\mathtt{q} : x\,\langle \mathtt{int} \rangle.\mathtt{s}{\to}\mathtt{r} : y\,\langle \mathtt{bool} \rangle.\mathtt{p}{\to}\mathtt{r} : x\,\langle \mathtt{bool} \rangle.\mathtt{end} \\
P_p &= x!1.x!\mathtt{true}.\mathbf{0} \\
P_q &= x?(i).\mathbf{0} \\
P_r &= y?(i).x?(j).\mathbf{0} \\
P_s &= y!\mathtt{true}.\mathbf{0}
\end{aligned}$$

Realizations

Example

$$
\begin{aligned}
\mathtt{G} &= \mathtt{p}{\to}\mathtt{q} : x\,\langle\mathtt{int}\rangle.\mathtt{s}{\to}\mathtt{r} : y\,\langle\mathtt{bool}\rangle.\mathtt{p}{\to}\mathtt{r} : x\,\langle\mathtt{bool}\rangle.\mathtt{end}\\
P_p &= x!1.x!\mathtt{true}.\mathbf{0}\\
P_q &= x?(i).\mathbf{0}\\
P_r &= y?(i).x?(j).\mathbf{0}\\
P_s &= y!\mathtt{true}.\mathbf{0}
\end{aligned}
$$

Races on x

Hence, G is bad

Realizations

Example

$$
\begin{aligned}
\mathtt{G} &= \mathtt{p}{\to}\mathtt{q} : x\,\langle \mathtt{int}\rangle.\mathtt{q}{\to}\mathtt{r} : y\,\langle \mathtt{bool}\rangle.\mathtt{p}{\to}\mathtt{r} : x\,\langle \mathtt{bool}\rangle.\mathtt{end}\\
P_p &= x!1.x!\mathtt{true}.\mathbf{0}\\
P_q &= x?(i).y!\mathtt{true}.\mathbf{0}\\
P_r &= y?(i).x?(j).\mathbf{0}
\end{aligned}
$$

Realizations

Example

$$\begin{aligned}
\mathtt{G} &= \mathtt{p}{\rightarrow}\mathtt{q} : x\,\langle \mathtt{int}\rangle.\mathtt{q}{\rightarrow}\mathtt{r} : y\,\langle \mathtt{bool}\rangle.\mathtt{p}{\rightarrow}\mathtt{r} : x\,\langle \mathtt{bool}\rangle.\mathtt{end} \\
P_p &= x!1.x!\mathtt{true}.\mathbf{0} \\
P_q &= x?(i).y!\mathtt{true}.\mathbf{0} \\
P_r &= y?(i).x?(j).\mathbf{0}
\end{aligned}$$

No races on x and y
Hence, $\mathtt{G}$ is good

This notion is formalised as LINEARITY (we postpone its definition)

Local types

Syntax

$\mathsf{T} ::=$		$x?\langle\tilde{\mathsf{S}}\rangle.\mathsf{T}$	receive
	$\mid$	$x!\langle\tilde{\mathsf{S}}\rangle.\mathsf{T}$	send
	$\mid$	$x \oplus \{l_i : \mathsf{T}_i\}_{i \in I}$	select
	$\mid$	$x \& \{l_i : \mathsf{T}_i\}_{i \in I}$	branch
	$\mid$	$\mathtt{end}$	termination
$\mathsf{S} ::=$		$\mathtt{int} \mid \mathtt{unit} \mid \mathtt{bool} \mid \ldots$	basic sorts

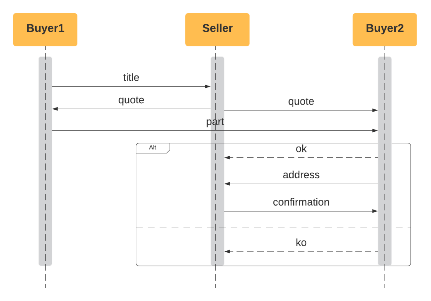
Projection

Definition

$$
\mathsf{G}\restriction \mathsf{p} = \begin{cases}
x\,!\langle\tilde{\mathsf{S}}\rangle\,.\,\mathsf{G}'\restriction \mathsf{p} & \text{if } \mathsf{G} = \mathsf{p}\to\mathsf{q} : x\,\langle\tilde{\mathsf{S}}\rangle\,.\,\mathsf{G}' \textit{ and } \mathsf{p} \neq \mathsf{q} \\
x\,?\langle\tilde{\mathsf{S}}\rangle\,.\,\mathsf{G}'\restriction \mathsf{p} & \text{if } \mathsf{G} = \mathsf{q}\to\mathsf{p} : x\,\langle\tilde{\mathsf{S}}\rangle\,.\,\mathsf{G}' \textit{ and } p \neq \mathsf{q} \\
\mathsf{G}'\restriction \mathsf{p} & \text{if } \mathsf{G} = \mathsf{q}\to\mathsf{r} : x\,\langle\tilde{\mathsf{S}}\rangle\,.\,\mathsf{G}' \textit{ and } \mathsf{p} \neq \mathsf{q} \neq \mathsf{r} \\
x \oplus \{l_i : \mathsf{G}_i\restriction \mathsf{p}\}_{i\in I} & \text{if } \mathsf{G} = \mathsf{p}\to\mathsf{q} : x\{l_i : \mathsf{G}_i\}_{i\in I} \textit{ and } \mathsf{p} \neq \mathsf{q} \\
x\&\{l_i : \mathsf{G}_i\restriction \mathsf{p}\}_{i\in I} & \text{if } \mathsf{G} = \mathsf{q}\to\mathsf{p} : x\{l_i : \mathsf{G}_i\}_{i\in I} \textit{ and } \mathsf{p} \neq \mathsf{q} \\
\mathsf{G}_1\restriction \mathsf{p} & \text{if } \mathsf{G} = \mathsf{q}\to\mathsf{r} : x\{l_i : \mathsf{G}_i\}_{i\in I} \textit{ and } \mathsf{p} \neq \mathsf{q} \neq \mathsf{r} \textit{ and} \\
& \hfill \forall i,j.\mathsf{G}_i\restriction \mathsf{p} = \mathsf{G}_j\restriction \mathsf{p} \\
\mathsf{G}_i\restriction \mathsf{p} & \text{if } \mathsf{G} = \mathsf{G}_1 \mid \mathsf{G}_2 \textit{ and } \mathsf{p} \in \mathsf{G}_i \textit{ and } \mathsf{p} \notin \mathsf{G}_j \textit{ and} \\
& \hfill i \neq j \in \{1,2\} \\
\mathsf{end} & \text{if } \mathsf{G} = \mathsf{G}_1 \mid \mathsf{G}_2 \textit{ and } \mathsf{p} \notin \mathsf{G}_1 \textit{ and } \mathsf{p} \notin \mathsf{G}_2
\end{cases}
$$

Global Graph (Choreography)

Two Buyers Protocol



$$\mathtt{TBProt} = \mathtt{b1}{\to}\mathtt{s} : x\,\langle\mathtt{string}\rangle.\\ \mathtt{s}{\to}\mathtt{b1} : y\,\langle\mathtt{float}\rangle.\\ \mathtt{s}{\to}\mathtt{b2} : z_1\,\langle\mathtt{float}\rangle.\\ \mathtt{b1}{\to}\mathtt{b2} : z_2\,\langle\mathtt{float}\rangle.\\ \mathtt{b2}{\to}\mathtt{s} : x \begin{Bmatrix} ok : \mathtt{b2}{\to}\mathtt{s} : x\,\langle\mathtt{string}\rangle.\mathtt{s}{\to}\mathtt{b2} : z_1\,\langle\mathtt{string}\rangle.\mathtt{end}, \\ ko : \mathtt{end} \end{Bmatrix}$$

Local Types

Two Buyers Protocol

$$\mathtt{TBProt} \;=\; \begin{array}{l} \mathtt{b1}{\to}\mathtt{s} : x\,\langle \mathtt{string}\rangle.\\ \mathtt{s}{\to}\mathtt{b1} : y\,\langle \mathsf{float}\rangle.\\ \mathtt{s}{\to}\mathtt{b2} : z_1\,\langle \mathsf{float}\rangle.\\ \mathtt{b1}{\to}\mathtt{b2} : z_2\,\langle \mathsf{float}\rangle.\\ \mathtt{b2}{\to}\mathtt{s} : x \left\{ \begin{array}{l} ok : \mathtt{b2}{\to}\mathtt{s} : x\,\langle \mathtt{string}\rangle.\mathtt{s}{\to}\mathtt{b2} : z_1\,\langle \mathtt{string}\rangle.\mathsf{end},\\ ko : \mathsf{end} \end{array} \right\} \end{array}$$

$$\begin{array}{lcl} \mathtt{TBProt}{\restriction}\mathtt{b1} &=& x\,!\langle \mathtt{string}\rangle.y\,?\langle \mathsf{float}\rangle.z_2\,!\langle \mathsf{float}\rangle.\mathsf{end}\\ \mathtt{TBProt}{\restriction}\mathtt{s} &=& y\,!\langle \mathsf{float}\rangle.z_1\,!\langle \mathsf{float}\rangle.\\ && \quad x\&\{ok : x\,?\langle \mathtt{string}\rangle.z_1\,!\langle \mathtt{string}\rangle.\mathsf{end}, ko : \mathsf{end}\}\\ \mathtt{TBProt}{\restriction}\mathtt{b2} &=& z_1\,?\langle \mathsf{float}\rangle.z_2\,?\langle \mathsf{float}\rangle.\\ && \quad x \oplus \{ok : x\,!\langle \mathtt{string}\rangle.z_1\,?\langle \mathtt{string}\rangle.\mathsf{end}, ko : \mathsf{end}\} \end{array}$$

Coherence (a.k.a well-formedness)

Coherence

$\mathtt{G}$ is coherent if it is linear and $\mathtt{G}\restriction\mathtt{p}$ is well-defined for each $\mathtt{p}$

Example

$$\mathtt{p}{\rightarrow}\mathtt{q} : x\{ok : \mathtt{r}{\rightarrow}\mathtt{s} : y\,\langle\mathtt{bool}\rangle.\mathtt{end}, quit : \mathtt{r}{\rightarrow}\mathtt{s} : y\,\langle\mathtt{int}\rangle.\mathtt{end}\}$$

Not coherent because $\mathtt{r}$ and $\mathtt{s}$ behave differently in the branches of the choice

Example

$$\mathtt{p}{\rightarrow}\mathtt{q} : x\,\langle\mathtt{int}\rangle.\mathtt{end} \mid \mathtt{p}{\rightarrow}\mathtt{s} : y\,\langle\mathtt{bool}\rangle.\mathtt{end}$$

Not coherent because $\mathtt{p}$ appears in two different parallel choreographies

Processes

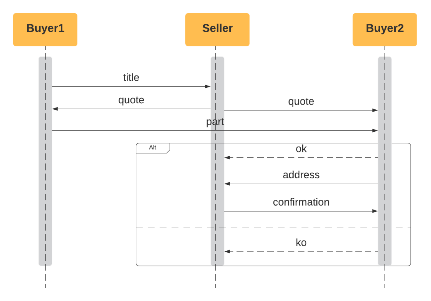
Syntax

P	$::=$	$s!\tilde{e}.P$	send
	$\mid$	$s?(\tilde{x}).P$	receive
	$\mid$	$s \triangleright l.P$	selection
	$\mid$	$s \triangleleft \{l_i : P_i\}_{i \in I}$	branch
	$\mid$	$\mathbf{0}$	ended
	$\mid$	$P \mid P$	parallel
	$\mid$	$\texttt{if}\ e\ \texttt{then}\ P\ \texttt{else}\ P$	conditional
	$\mid$	$a_{[i]}(\tilde{s}).P$	session acceptance
	$\mid$	$\overline{a}_{[2..n]}(\tilde{s}).P$	session request
	$\mid$	$(\nu w)P$	hiding
	$\mid$	$s :: \tilde{h}$	message queue
e	$::=$	$v \mid e\ or\ e \mid \ldots$	expressions
v	$::=$	$\texttt{true} \mid \texttt{false}$	values
w	$::=$	$a \mid \tilde{s}$	
h	$::=$	$\tilde{v} \mid l$	message in transit

$s, s_1, \ldots$ session channels; $x, y, \ldots$ variables; $a, b, \ldots$ protocol (shared) names

Global Graph (Choreography)

Two Buyers Protocol



$$
\begin{aligned}
\mathtt{TBProt} = {} & \mathtt{b1}{\rightarrow}\mathtt{s} : x\,\langle \mathtt{string}\rangle. \\
& \mathtt{s}{\rightarrow}\mathtt{b1} : y\,\langle \mathtt{float}\rangle. \\
& \mathtt{s}{\rightarrow}\mathtt{b2} : z_1\,\langle \mathtt{float}\rangle. \\
& \mathtt{b1}{\rightarrow}\mathtt{b2} : z_2\,\langle \mathtt{float}\rangle. \\
& \mathtt{b2}{\rightarrow}\mathtt{s} : x \left\{ \begin{array}{l} ok : \mathtt{b2}{\rightarrow}\mathtt{s} : x\,\langle \mathtt{string}\rangle.\mathtt{s}{\rightarrow}\mathtt{b2} : z_1\,\langle \mathtt{string}\rangle.\mathtt{end}, \\ ko : \mathtt{end} \end{array} \right\}
\end{aligned}
$$

Processes

Two Buyer Protocol

$$
\begin{aligned}
P_{\texttt{Buyer}_1} &= \overline{a}_{[2..3]}(b_1, b_2, b_2', s).P_1 \\
P_1 &= s\,\texttt{!}\,\text{“}\mathit{My\ Book}\text{”}\,\texttt{.}\,b_1\,\texttt{?}(\mathit{quote})\,\texttt{.}\,b_2\,\texttt{!}(\mathit{quote}\ /\ 2)\,\texttt{.}\,\mathbf{0} \\
P_{\texttt{Buyer}_2} &= a_{[2]}(b_1, b_2, b_2', s).P_2 \\
P_2 &= b_2\,\texttt{?}(\mathit{quote})\,\texttt{.}\,b_2'\,\texttt{?}(\mathit{contrib})\,\texttt{.} \\
&\qquad\quad \texttt{if}\ (\mathit{contrib} > \mathit{quote}/2) \\
&\qquad\qquad\quad \texttt{then}\ s \triangleright \mathit{ok}\,\texttt{.}\,s\,\texttt{!}\,\text{"}\mathit{via...}\text{"}\,\texttt{.}\,b_2\,\texttt{?}(x)\,\texttt{.}\,\mathbf{0} \\
&\qquad\qquad\quad \texttt{else}\ s \triangleright \mathit{ko}\,\texttt{.}\,\mathbf{0} \\
P_{\texttt{Seller}} &= a_{[3]}(b_1, b_2, b_2', s).Q \\
Q &= s\,\texttt{?}(\mathit{title})\,\texttt{.}\,b_1\,\texttt{!}\,\mathit{100}\,\texttt{.}\,b_2\,\texttt{!}\,\mathit{100}\,\texttt{.} \\
&\qquad\quad s \triangleleft \{\mathit{ok} : s\,\texttt{?}(x)\,\texttt{.}\,b_2\,\texttt{!}\ldots\texttt{.}\,\mathbf{0},\ \mathit{ko} : \mathbf{0}\}
\end{aligned}
$$

Processes

Semantics

$$\frac{}{\overline{a}_{[2..n]}(\tilde{s}).P_1 \mid a_{[2]}(\tilde{s}).P_2 \mid \ldots \mid a_{[n]}(\tilde{s}).P_n \xrightarrow{\tau} (\nu\tilde{s})(P_1 \mid \ldots \mid P_n \mid \tilde{s} : \emptyset)}\ \text{SInit}$$

$$\frac{\tilde{e} \downarrow \tilde{v}}{s\,!\,\tilde{e}\,\mathbf{.}\,P \mid s :: h \xrightarrow{s} P \mid s :: h \cdot \tilde{v}}\ \text{SSend}$$

$$\frac{}{s\,?(\tilde{x})\,\mathbf{.}\,P \mid s :: \tilde{v} \cdot h \xrightarrow{s} P\{\tilde{v}/\tilde{x}\} \mid s :: h}\ \text{SRec}$$

$$\frac{}{s \triangleright l\,\mathbf{.}\,P \mid s :: h \xrightarrow{s} P \mid s :: h \cdot l}\ \text{SSel}$$

$$\frac{j \in I}{s \triangleleft \{l_i : P_i\}_{i \in I} \mid s :: l_j \cdot h \xrightarrow{s} P_j \mid s :: h}\ \text{SBranch}$$

Processes

Semantics (cont)

$$\frac{e \downarrow \texttt{true}}{\texttt{if}\ e\ \texttt{then}\ P\ \texttt{else}\ Q \xrightarrow{\tau} P}\ \text{SThen}$$

$$\frac{e \downarrow \texttt{false}}{\texttt{if}\ e\ \texttt{then}\ P\ \texttt{else}\ Q \xrightarrow{\tau} Q}\ \text{SElse}$$

$$\frac{P \xrightarrow{\alpha} P'}{P \mid Q \xrightarrow{\alpha} P' \mid Q}\ \text{SPar}$$

$$\frac{P \xrightarrow{\alpha} P'}{(\nu w)P \xrightarrow{\alpha} (\nu w)P'}\ \text{SNew}$$

$$\frac{P \equiv Q \quad Q \xrightarrow{\alpha} Q' \quad Q' \equiv P'}{P \xrightarrow{\alpha} P'}\ \text{SStruct}$$

Processes

Structural equivalence

$$
\begin{aligned}
P \mid \mathbf{0} &\equiv P \\
P \mid Q &\equiv Q \mid P \\
(P \mid Q) \mid R &\equiv Q \mid (P \mid R) \\
(\nu w_1)(\nu w_2)P &\equiv (\nu w_2)(\nu w_1)P \\
(\nu w)P \mid Q &\equiv (\nu w)(P \mid Q) \qquad \text{if } w \notin \mathit{fn}(Q) \\
(\nu w)0 &\equiv 0 \\
(\nu \tilde{s})(\tilde{s} :: \emptyset) &\equiv 0
\end{aligned}
$$

The last rule stands for session termination

Typing

context for unlimited resources

$$\begin{array}{rll} \Gamma ::= & \emptyset & \textit{empty context} \\ & \mid x : \mathsf{S}, \Gamma & \textit{variables are of basic sorts} \\ & \mid a : \mathsf{G}, \Gamma & \textit{protocols are associated with global types} \end{array}$$

Expressions: $\Gamma \vdash e \triangleright \mathsf{S}$

$$\Gamma, x : \mathsf{S} \vdash x \triangleright \mathsf{S}$$

$$\Gamma, a : \mathsf{G} \vdash a \triangleright \mathsf{G}$$

$$\Gamma \vdash \mathtt{true} \triangleright \mathtt{bool}$$

$$\Gamma \vdash \mathtt{false} \triangleright \mathtt{bool}$$

...*(rules for remaining expressions)*

$$\frac{\Gamma \vdash \tilde{e_1} \triangleright \tilde{\mathsf{S}_1} \qquad \Gamma \vdash \tilde{e_2} \triangleright \tilde{\mathsf{S}_2}}{\Gamma \vdash \tilde{e_1} \cdot \tilde{e_2} \triangleright \tilde{\mathsf{S}_1} \cdot \tilde{\mathsf{S}_2}}$$

Typing

Context for linear resources

$$\begin{array}{rcll} \boldsymbol{\Delta} & ::= & \emptyset & \textit{empty context} \\ & \mid & \tilde{s} : \mathsf{T} \,@\, p & \textit{session channels are of local session types} \end{array}$$

Typing

Processes $\Gamma \vdash P \triangleright \Delta$

$$\frac{\Gamma \vdash a \triangleright \mathtt{G} \qquad part(\mathtt{G}) = \{\mathtt{p}_1, \ldots, \mathtt{p}_n\} \qquad |\tilde{s}| = ch(\mathtt{G}) \qquad \Gamma \vdash P \triangleright \Delta, \tilde{s} : \mathtt{G}\restriction \mathtt{p}_1 \,@\, p_1}{\Gamma \vdash \overline{a}_{[2..n]}(\tilde{s}).P \triangleright \Delta}\ \text{Req}$$

$$\frac{\Gamma \vdash a \triangleright \mathtt{G} \qquad i \in part(\mathtt{G}) \qquad |\tilde{s}| = ch(\mathtt{G}) \qquad \Gamma \vdash P \triangleright \Delta, \tilde{s} : \mathtt{G}\restriction i \,@\, i}{\Gamma \vdash a_{[i]}(\tilde{s}).P \triangleright \Delta}\ \text{Acc}$$

Typing

Processes $\Gamma \vdash P \triangleright \Delta$

$$\frac{\Gamma \vdash \tilde{e} \triangleright \tilde{\mathsf{S}} \qquad \Gamma \vdash P \triangleright \Delta, \tilde{y} : \mathsf{T} @ p}{\Gamma \vdash s_k ! \tilde{e} . P \triangleright \Delta, \tilde{s} : s_k ! \langle \tilde{\mathsf{S}} \rangle . \mathsf{T} @ p}\ \text{Send}$$

$$\frac{\Gamma, \tilde{x} : \tilde{\mathsf{S}} \vdash P \triangleright \Delta, \tilde{s} : \mathsf{T} @ p}{\Gamma \vdash s_k ? (\tilde{x}) . P \triangleright \Delta, \tilde{s} : s_k ? \langle \tilde{\mathsf{S}} \rangle . \mathsf{T} @ p}\ \text{Rec}$$

Typing

Processes $\Gamma \vdash P \triangleright \Delta$

$$\frac{\Gamma \vdash P \triangleright \Delta, \tilde{s} : \mathsf{T}_j \,@\, p \qquad\qquad j \in I}{\Gamma \vdash s_k \triangleright l_j \,.\, P \triangleright \Delta, \tilde{s} : s_k \oplus \{l_i : \mathsf{T}_i\}_{i \in I} \,@\, p}\ \text{Sel}$$

$$\frac{\Gamma \vdash P_i \triangleright \Delta, \tilde{s} : \mathsf{T}_i \,@\, p \quad \forall i \in I \qquad\qquad i \in I}{\Gamma \vdash s_k \triangleleft \{l_i : P_i\}_{i \in I} \triangleright \Delta, \tilde{s} : s_k \& \{l_i : \mathsf{T}_i\}_{i \in I} \,@\, p}\ \text{Branch}$$

Typing

Processes $\Gamma \vdash P \triangleright \Delta$

$$\frac{\Gamma \vdash P \triangleright \Delta_1 \qquad \Gamma \vdash Q \triangleright \Delta_2}{\Gamma \vdash P \mid Q \triangleright \Delta_1, \Delta_2}\ \text{Par}$$

$$\frac{\Gamma \vdash e \triangleright \texttt{bool} \qquad \Gamma \vdash P \triangleright \Delta \qquad \Gamma \vdash Q \triangleright \Delta}{\Gamma \vdash \texttt{if}\ e\ \texttt{then}\ P\ \texttt{else}\ Q \triangleright \Delta}\ \text{if}$$

$$\frac{\Gamma\ \textit{is}\ \mathsf{end}\ \textit{only}}{\Gamma \vdash \mathbf{0} \triangleright \Delta}\ \text{end}$$

$$\frac{\Gamma, a : \mathsf{G} \vdash P \triangleright \Delta}{\Gamma \vdash (\nu a)P \triangleright \Delta}\ \text{rest}$$

Typing

Two Buyer Protocol

$$
\begin{aligned}
P_{\texttt{Buyer}_1} &= \overline{a}_{[2..3]}(b_1, b_2, b_2', s).P_1 \\
P_1 &= s\,!\,\text{``}\mathit{My\ Book}\text{''}\,.\,b_1\,?(\mathit{quote})\,.\,b_2\,!(\mathit{quote}\ /\ 2)\,.\mathbf{0} \\
P_{\texttt{Buyer}_2} &= a_{[2]}(b_1, b_2, b_2', s).P_2 \\
P_2 &= b_2\,?(\mathit{quote})\,.\,b_2'\,?(\mathit{contrib})\,. \\
&\qquad \texttt{if}\ (\mathit{contrib} > \mathit{quote}/2) \\
&\qquad\quad \texttt{then}\ s \triangleright \mathit{ok}\,.\,s\,!\,''\mathit{via...}''\,.\,b_2\,?(x)\,.\mathbf{0} \\
&\qquad\quad \texttt{else}\ s \triangleright \mathit{ko}\,.\mathbf{0} \\
P_{\texttt{Seller}} &= a_{[3]}(b_1, b_2, b_2', s).Q \\
Q &= s\,?(\mathit{title})\,.\,b_1\,!\,\mathit{100}\,.\,b_2\,!\,\mathit{100}\,. \\
&\qquad s \triangleleft \{\mathit{ok} : s\,?(x)\,.\,b_2\,!\ldots.\mathbf{0},\ \mathit{ko} : \mathbf{0}\}
\end{aligned}
$$

Show that $\emptyset \vdash (\nu a)(P_{\texttt{Buyer}_1} \mid P_{\texttt{Buyer}_2} \mid P_{\texttt{Seller}}) \triangleright \emptyset$

Two Buyers Protocol

$$
\begin{aligned}
\mathtt{TBProt}\restriction\mathtt{b1} &= x\,!\langle\mathtt{string}\rangle.y\,?\langle\mathtt{float}\rangle.z_2\,!\langle\mathtt{float}\rangle.\mathtt{end} \\
\mathtt{TBProt}\restriction\mathtt{s} &= y\,!\langle\mathtt{float}\rangle.z_1\,!\langle\mathtt{float}\rangle. \\
&\quad x\&\{\mathit{ok} : x\,?\langle\mathtt{string}\rangle.z_1\,!\langle\mathtt{string}\rangle.\mathtt{end},\ \mathit{ko} : \mathtt{end}\} \\
\mathtt{TBProt}\restriction\mathtt{b2} &= z_1\,?\langle\mathtt{float}\rangle.z_2\,?\langle\mathtt{float}\rangle. \\
&\quad x \oplus \{\mathit{ok} : x\,!\langle\mathtt{string}\rangle.z_1\,?\langle\mathtt{string}\rangle.\mathtt{end},\ \mathit{ko} : \mathtt{end}\}
\end{aligned}
$$

Properties

Subject reduction (approx)

- $\Gamma \vdash P \triangleright \Delta$ such that Δ is coherent[a] and $P \xrightarrow{\alpha} P'$ imply $\Gamma \vdash P' \triangleright \Delta'$ where $\Delta = \Delta'$ or $\Delta \to \Delta'$

Reduction of specifications

$$\tilde{s} : s_k\,!\langle \mathsf{S}\rangle \mathsf{T}_1 \,@\, p,\ \ \tilde{s} : s_k\,?\langle \mathsf{S}\rangle \mathsf{T}_2 \,@\, p \to \tilde{s} : \mathsf{T}_1 \,@\, p,\ \ \tilde{s} : \mathsf{T}_2 \,@\, p$$

$$\tilde{s} : s_k \oplus l_j \mathsf{T} \,@\, p,\ \ \tilde{s} : s_k \& \{l_i : \mathsf{T}_j\}_{i \in I} \,@\, p \to \tilde{s} : \mathsf{T} \,@\, p,\ \ \tilde{s} : \mathsf{T}_j \,@\, p$$

$$\frac{\Delta_1 \to \Delta_1'}{\Delta_1, \Delta_2 \to \Delta_1', \Delta_2}$$

[a] linear, and all projections well-defined

Properties

Session fidelity (approx)

- $\Gamma \ \vdash \ P \ \triangleright \ \Delta$ such that Δ is coherent and $\Delta(\tilde{s}) = \mathtt{G}\!\upharpoonright\!\mathsf{p}_1 \,@\, p_n, \ldots, \mathtt{G}\!\upharpoonright\!\mathsf{p}_n \,@\, p_n$. If $P \xrightarrow{s_k} P'$ then $\mathtt{G} \rightarrow \mathtt{G}'$ and $\Gamma \ \vdash \ P' \ \triangleright \ \Delta'$ and $\Delta'(\tilde{s}) = \mathtt{G}'\!\upharpoonright\!\mathsf{p}_1 \,@\, p_1, \ldots, \mathtt{G}'\!\upharpoonright\!\mathsf{p}_n \,@\, p_n$.

Semantics of Global types

$$\mathsf{G} ::= \ldots \mid \mathsf{p}\rightsquigarrow\mathsf{q} : x\,\langle\tilde{\mathsf{S}}\rangle.\mathsf{G} \mid \mathsf{p}\rightsquigarrow\mathsf{q} : x\{l_j : \mathsf{G}_j\}$$

Semantics

$$\mathsf{p}\to\mathsf{q} : x\,\langle\tilde{\mathsf{S}}\rangle.\mathsf{G} \xrightarrow{\mathsf{p}\to\mathsf{q}:x\,\langle\tilde{\mathsf{S}}\rangle} \mathsf{p}\rightsquigarrow\mathsf{q} : x\,\langle\tilde{\mathsf{S}}\rangle.\mathsf{G} \qquad \mathsf{p}\rightsquigarrow\mathsf{q} : x\,\langle\tilde{\mathsf{S}}\rangle.\mathsf{G} \xrightarrow{\mathsf{p}\rightsquigarrow\mathsf{q}:x\,\langle\tilde{\mathsf{S}}\rangle} \mathsf{G}$$

$$\mathsf{p}\to\mathsf{q} : x\{l_j : \mathsf{G}_j\}_{j\in J} \xrightarrow{\mathsf{p}\to\mathsf{q}:x\,\langle l_j\rangle} \mathsf{p}\rightsquigarrow\mathsf{q} : x\{l_j : \mathsf{G}_j\}$$

$$\mathsf{p}\rightsquigarrow\mathsf{q} : x\{l_j : \mathsf{G}_j\} \xrightarrow{\mathsf{p}\rightsquigarrow\mathsf{q}:x\,\langle l_j\rangle} \mathsf{G}_j$$

$$\frac{\mathsf{G} \xrightarrow{\ell} \mathsf{G}' \quad \mathsf{p},\mathsf{q} \notin subj(\ell)}{\mathsf{p}\to\mathsf{q} : x\,\langle\tilde{\mathsf{S}}\rangle.\mathsf{G} \xrightarrow{\ell} \mathsf{p}\to\mathsf{q} : x\,\langle\tilde{\mathsf{S}}\rangle.\mathsf{G}'} \qquad \frac{\forall j \in J.\mathsf{G}_j \xrightarrow{\ell} \mathsf{G}'_j \quad \mathsf{p},\mathsf{q} \notin subj(\ell)}{\mathsf{p}\to\mathsf{q} : x\{l_j : \mathsf{G}'_j\}_{j\in J} \xrightarrow{\ell} \mathsf{p}\to\mathsf{q} : x\{l_j : \mathsf{G}'_j\}_{j\in J}}$$

$$\frac{\mathsf{G} \xrightarrow{\ell} \mathsf{G}' \quad \mathsf{p},\mathsf{q} \notin subj(\ell)}{\mathsf{p}\rightsquigarrow\mathsf{q} : x\,\langle\tilde{\mathsf{S}}\rangle.\mathsf{G} \xrightarrow{\ell} \mathsf{p}\rightsquigarrow\mathsf{q} : x\,\langle\tilde{\mathsf{S}}\rangle.\mathsf{G}'} \qquad \frac{\forall j \in J.\mathsf{G}_j \xrightarrow{\ell} \mathsf{G}'_j \quad \mathsf{p},\mathsf{q} \notin subj(\ell)}{\mathsf{p}\rightsquigarrow\mathsf{q} : x\{l_j : \mathsf{G}'_j\} \xrightarrow{\ell} \mathsf{p}\rightsquigarrow\mathsf{q} : x\{l_j : \mathsf{G}'_j\}}$$

$$\frac{\mathsf{G}_1 \xrightarrow{\ell} \mathsf{G}'_1}{\mathsf{G}_1 \mid \mathsf{G}_2 \xrightarrow{\mathsf{G}}{}'_1 \mid \mathsf{G}_2} \qquad \frac{\mathsf{G}_2 \xrightarrow{\ell} \mathsf{G}'_2}{\mathsf{G}_1 \mid \mathsf{G}_2 \xrightarrow{\mathsf{G}}{}_1 \mid \mathsf{G}'_2}$$

Properties

Progress (approx)

$\Gamma \ \vdash \ P \ \triangleright \ \Delta$ such that Δ is coherent, P simple, well-linked and queue-full. Then,

- If $P \not\equiv \mathbf{0}$ then $P \xrightarrow{\alpha} P'$ for some P',
- If $\Delta(\tilde{s}) = \mathtt{G}\!\upharpoonright \mathsf{p}_1 \,@\, p_n, \ldots, \mathtt{G}\!\upharpoonright \mathsf{p}_n \,@\, p_n$, and $\mathtt{G} \xrightarrow{\ell} \mathtt{G}'$ then $P \xrightarrow{s} P'$ and $ch(\ell) = s$.