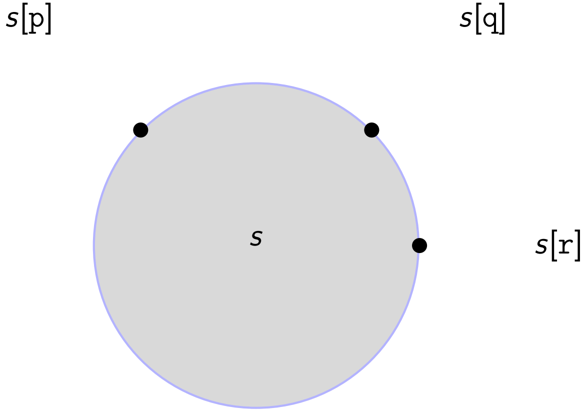
Fair Subtyping for Multi-Party Session Types

Luca Padovani

Dipartimento di Informatica, Università di Torino

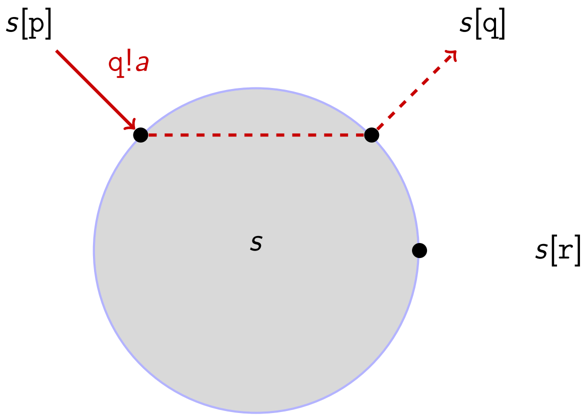
COORDINATION'11

Sessions and session types



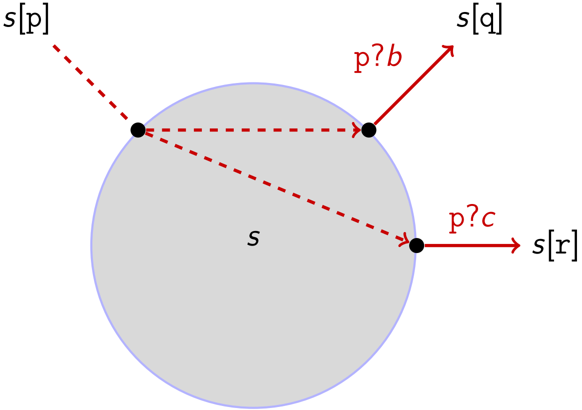
- $s[\mathtt{p}] : T = \mathtt{q}!a.T \oplus \mathtt{q}!b.\mathtt{r}!c.\mathsf{end}$
- $s[\mathtt{q}] : S = \mathtt{p}?a.S + \mathtt{p}?b.\mathsf{end}$
- $s[\mathtt{r}] : \mathtt{p}?c.\mathsf{end}$

Sessions and session types



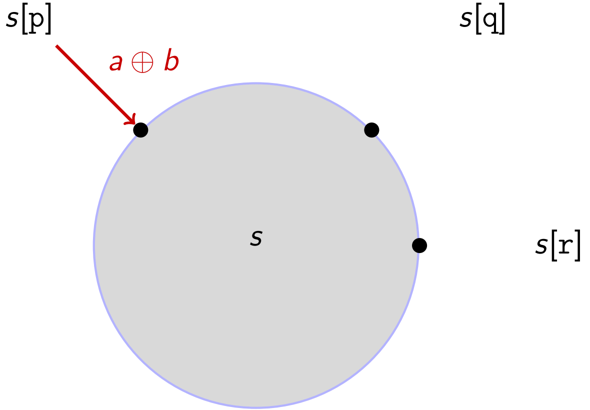
- $s[\mathtt{p}] : T = \mathtt{q}!a.T \oplus \mathtt{q}!b.\mathtt{r}!c.\mathsf{end}$
- $s[\mathtt{q}] : S = \mathtt{p}?a.S + \mathtt{p}?b.\mathsf{end}$
- $s[\mathtt{r}] : \mathtt{p}?c.\mathsf{end}$

Sessions and session types



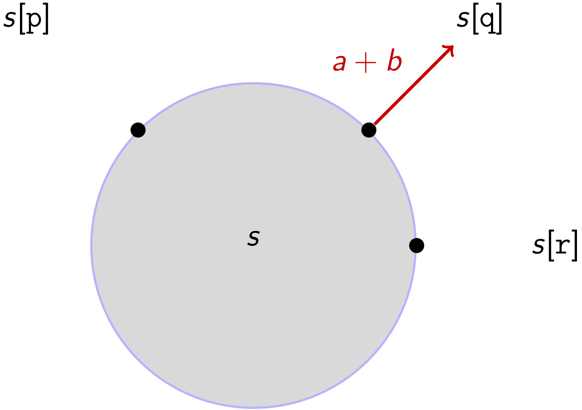
- $s[\mathtt{p}] : T = \mathtt{q}!a.T \oplus \mathtt{q}!b.\mathtt{r}!c.\mathsf{end}$
- $s[\mathtt{q}] : S = \mathtt{p}?a.S + \mathtt{p}?b.\mathsf{end}$
- $s[\mathtt{r}] : \mathtt{p}?c.\mathsf{end}$

Sessions and session types



- $s[\mathtt{p}] : T = \mathtt{q}!a.T \oplus \mathtt{q}!b.\mathtt{r}!c.\mathsf{end}$
- $s[\mathtt{q}] : S = \mathtt{p}?a.S + \mathtt{p}?b.\mathsf{end}$
- $s[\mathtt{r}] : \mathtt{p}?c.\mathsf{end}$

Sessions and session types



- $s[\mathtt{p}] : T = \mathtt{q}!a.T \oplus \mathtt{q}!b.\mathtt{r}!c.\mathsf{end}$
- $s[\mathtt{q}] : S = \mathtt{p}?a.S + \mathtt{p}?b.\mathsf{end}$
- $s[\mathtt{r}] : \mathtt{p}?c.\mathsf{end}$

Session correctness = safety + liveness

Safety

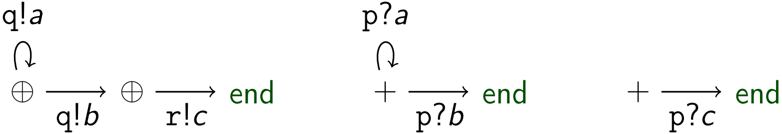
- no message of unexpected type is ever sent

Liveness

- every non-terminated participant eventually makes progress

Example: multi-party session

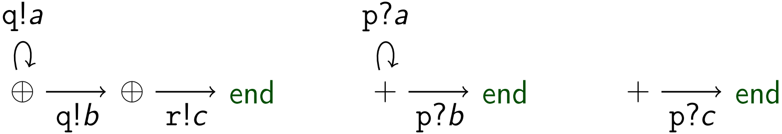
- $s[\mathtt{p}] : T = \mathtt{q}!a.T \oplus \mathtt{q}!b.\mathtt{r}!c.\mathsf{end}$
- $s[\mathtt{q}] : S = \mathtt{p}?a.S + \mathtt{p}?b.\mathsf{end}$
- $s[\mathtt{r}] : \mathtt{p}?c.\mathsf{end}$



Is this session correct?

Example: multi-party session

- $s[\mathtt{p}] : T = \mathtt{q}!a.T \oplus \mathtt{q}!b.\mathtt{r}!c.\mathsf{end}$
- $s[\mathtt{q}] : S = \mathtt{p}?a.S + \mathtt{p}?b.\mathsf{end}$
- $s[\mathtt{r}] : \mathtt{p}?c.\mathsf{end}$



Is this session correct? **Yes, under a fairness assumption**

Subtyping for session types

- Gay, Hole, **Subtyping for session types in the pi calculus**, 2005

$$\text{end} \leqslant_{\text{GH}} \text{end}$$

$$\frac{T_i \leqslant_{\text{GH}} S_i \;{}^{(i \in I)}}{\sum_{i \in I} ?a_i.T_i \leqslant_{\text{GH}} \sum_{i \in I \cup J} ?a_i.S_i} \qquad \frac{T_i \leqslant_{\text{GH}} S_i \;{}^{(i \in I)}}{\bigoplus_{i \in I \cup J} !a_i.T_i \leqslant_{\text{GH}} \bigoplus_{i \in I} !a_i.S_i}$$

$T \leqslant_{\text{GH}} S$ means. . .

- it is safe to use a channel of type T where a channel of type S is expected, or. . .
- it is safe to use a process that behaves as S where a process that behaves as T is expected

Subtyping for session types

- Gay, Hole, **Subtyping for session types in the pi calculus**, 2005

$$\text{end} \leqslant_{\text{GH}} \text{end}$$

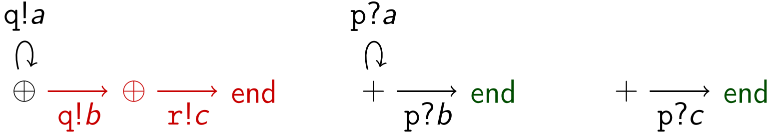
$$\frac{T_i \leqslant_{\text{GH}} S_i \ ^{(i \in I)}}{\sum_{i \in I} \mathtt{p}?a_i.T_i \leqslant_{\text{GH}} \sum_{i \in I \cup J} \mathtt{p}?a_i.S_i} \qquad \frac{T_i \leqslant_{\text{GH}} S_i \ ^{(i \in I)}}{\bigoplus_{i \in I \cup J} \mathtt{p}!a_i.T_i \leqslant_{\text{GH}} \bigoplus_{i \in I} \mathtt{p}!a_i.S_i}$$

$T \leqslant_{\text{GH}} S$ means...

- it is safe to use a channel of type T where a channel of type S is expected, or...
- it is safe to use a process that behaves as S where a process that behaves as T is expected

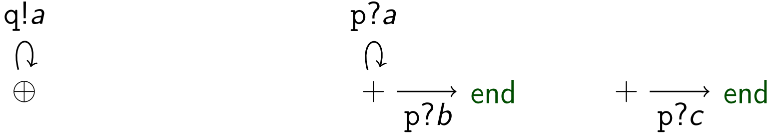
Example: multi-party session (and subtyping)

- $\mathtt{p} : T = \mathtt{q}!a.T \oplus \mathtt{q}!b.\mathtt{r}!c.\mathsf{end}$
- $\mathtt{q} : S = \mathtt{p}?a.S + \mathtt{p}?b.\mathsf{end}$
- $\mathtt{r} : \mathtt{p}?c.\mathsf{end}$



Example: multi-party session (and subtyping)

- $\mathtt{p} : T = \mathtt{q}!a.T$
- $\mathtt{q} : S = \mathtt{p}?a.S + \mathtt{p}?b.\mathsf{end}$
- $\mathtt{r} : \mathtt{p}?c.\mathsf{end}$



Is this session correct?

Dyadic vs multi-party sessions

In the dyadic setting...

- $\leqslant_{\mathsf{GH}}$ preserves both safety and liveness

$$\mathrm{p}!a.T \not\leqslant_{\mathsf{GH}} \mathsf{end}$$

(a process owning an endpoint is required to use it)

In the multi-party setting...

- $\leqslant_{\mathsf{GH}}$ preserves safety
- $\leqslant_{\mathsf{GH}}$ does not (necessarily) preserve liveness

How to fix subtyping

Definition (**correct** session)

- $T_1 \mid \cdots \mid T_n$ **correct** if
  $T_1 \mid \cdots \mid T_n \Longrightarrow S_1 \mid \cdots \mid S_n$ implies
  $S_1 \mid \cdots \mid S_n \Longrightarrow \mathsf{end} \mid \cdots \mid \mathsf{end}$

Definition (fair subtyping)

- $[\![T]\!] = \{M \mid (T \mid M) \text{ is } \textbf{correct}\}$
- $T \leqslant S \quad \text{iff} \quad [\![T]\!] \subseteq [\![S]\!]$

How to fix subtyping

Definition (**correct** session)

- $T_1 \mid \cdots \mid T_n$ **correct** if
  $T_1 \mid \cdots \mid T_n \Longrightarrow S_1 \mid \cdots \mid S_n$ implies
  $S_1 \mid \cdots \mid S_n \Longrightarrow \mathsf{end} \mid \cdots \mid \mathsf{end}$

Definition (fair subtyping)

- $[\![T]\!] = \{M \mid (T \mid M) \text{ is correct}\}$
- $T \leqslant S \quad \text{iff} \quad [\![T]\!] \subseteq [\![S]\!]$

How to fix subtyping

Definition (**correct** session)

- $T_1 \mid \cdots \mid T_n$ **correct** if
  $T_1 \mid \cdots \mid T_n \Longrightarrow S_1 \mid \cdots \mid S_n$ implies
  $S_1 \mid \cdots \mid S_n \Longrightarrow \mathsf{end} \mid \cdots \mid \mathsf{end}$

Definition (fair subtyping)

- $[\![T]\!] = \{M \mid (T \mid M) \text{ is correct}\}$
- $T \leqslant S \quad \text{iff} \quad [\![T]\!] \subseteq [\![S]\!]$

How to fix subtyping

Definition (**correct** session)

- $T_1 \mid \cdots \mid T_n$ **correct** if
  $T_1 \mid \cdots \mid T_n \Longrightarrow S_1 \mid \cdots \mid S_n$ implies
  $S_1 \mid \cdots \mid S_n \Longrightarrow \mathsf{end} \mid \cdots \mid \mathsf{end}$

Definition (fair subtyping)

- $[\![T]\!] = \{M \quad \mid \quad (T \mid M) \text{ is } \textbf{correct}\}$
- $T \leqslant S \quad \text{iff} \quad [\![T]\!] \subseteq [\![S]\!]$

Dilemma

$$\leqslant_{\mathsf{GH}} \qquad \text{versus} \qquad \leqslant$$

- $\leqslant_{\mathsf{GH}}$ is intuitive but unsound
- $\leqslant$ is sound but obscure

$\leqslant_{\mathsf{GH}}$ and $\leqslant$ are incomparable

$$
\begin{array}{rcl}
T & = & \mathtt{p}!a.T \\
S & = & \mathtt{q}?b.S
\end{array}
\qquad
\begin{array}{rcl}
T & \leqslant & S \\
S & \leqslant & T
\end{array}
\qquad
\begin{array}{rcl}
[\![T]\!] & = & \emptyset \\
[\![S]\!] & = & \emptyset
\end{array}
\qquad
\begin{array}{rcl}
T & \not\leqslant_{\mathsf{GH}} & S \\
S & \not\leqslant_{\mathsf{GH}} & T
\end{array}
$$

$\leqslant_{\mathsf{GH}}$ and $\leqslant$ are incomparable

$$T = \mathsf{p}!a.T \qquad S = \mathsf{q}?b.S$$

$$T \leqslant S \qquad S \leqslant T$$

$$[\![T]\!] = \emptyset \qquad [\![S]\!] = \emptyset$$

$$T \not\leqslant_{\mathsf{GH}} S \qquad S \not\leqslant_{\mathsf{GH}} T$$

not viable $[\![\mathsf{fail}]\!] = [\![T]\!] = [\![S]\!] = \cdots = \emptyset$

$[\![\ldots]\!] \neq \emptyset$ viable $\leqslant \subseteq \leqslant_{\mathsf{GH}}$

A normal form for session types

T is in normal form if either

- $T = \mathsf{fail}$, or
- $\mathsf{end} \in \mathsf{trees}(S)$ for every $S \in \mathsf{trees}(T)$

Proposition

For every T there exists $S \leqslant T$ in nf

Theorem

Let $T, S \neq \mathsf{fail}$ be in nf. Then $T \leqslant S$ implies $T \leqslant_{\mathsf{GH}} S$

A normal form for session types

T is in normal form if either

- $T = \text{fail}$, or
- $\text{end} \in \text{trees}(S)$ for every $S \in \text{trees}(T)$

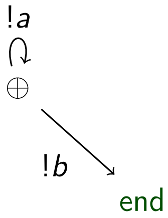
Proposition

For every T there exists $S \leqslant T$ in nf

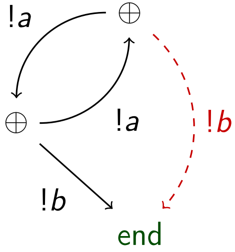
Theorem

Let $T, S \neq \text{fail}$ be in nf. Then $T \leqslant S$ implies $T \leqslant_{\text{GH}} S$

Experiment 1



$\overset{?}{\leqslant}$

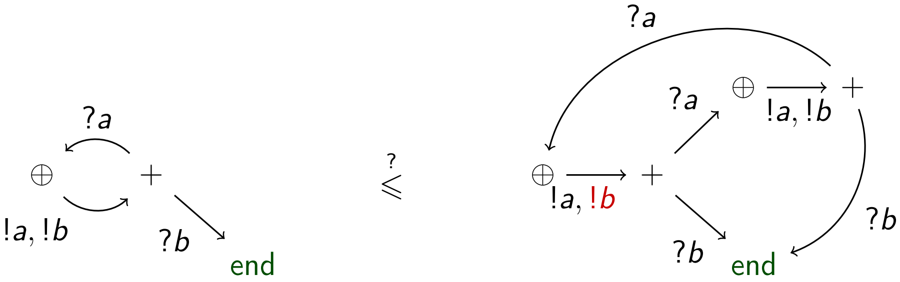


$T = !a.T \oplus !b.\text{end}$

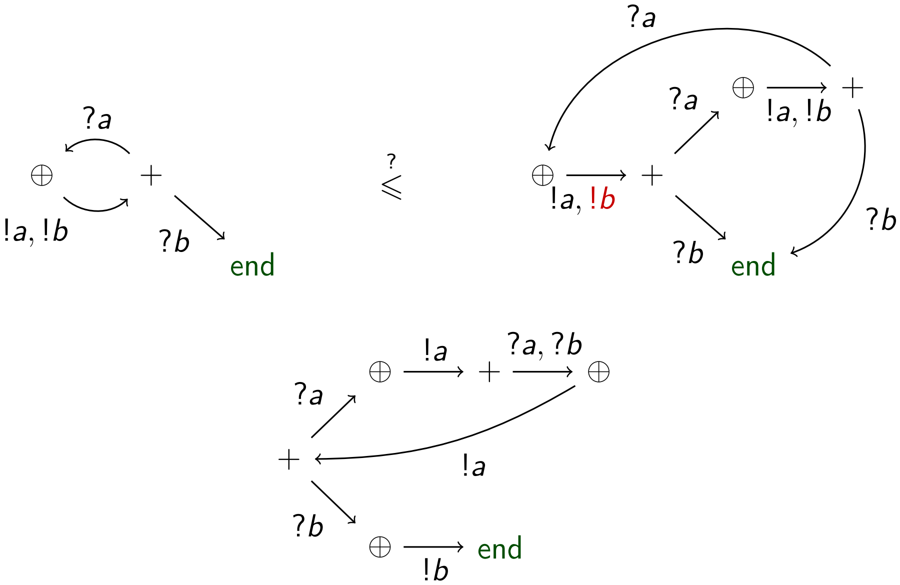
$S = !a.!a.S \oplus !b.\text{end}$

Is there a context M that discriminates between T and S?

Experiment 2



Experiment 2



Semantic subtyping comes to rescue

$$T \leqslant S \quad \stackrel{\text{def}}{\iff} \quad [\![T]\!] \subseteq [\![S]\!]$$

$$T \leqslant S \quad \iff \quad [\![T]\!] \setminus [\![S]\!] = \emptyset$$

$$T \text{ not viable} \quad \stackrel{\text{def}}{\iff} \quad [\![T]\!] = \emptyset$$

Idea

1. *Compute* $T - S$ *such that* $[\![T - S]\!] = [\![T]\!] \setminus [\![S]\!]$
2. *Reduce* $T \leqslant S$ *to checking* $T - S$ *not viable*

Semantic subtyping comes to rescue

$$T \leqslant S \quad \stackrel{\text{def}}{\iff} \quad [\![T]\!] \subseteq [\![S]\!]$$

$$T \leqslant S \quad \iff \quad [\![T]\!] \setminus [\![S]\!] = \emptyset$$

$$T \text{ not viable} \quad \stackrel{\text{def}}{\iff} \quad [\![T]\!] = \emptyset$$

Idea

1. *Compute* $T - S$ *such that* $[\![T - S]\!] = [\![T]\!] \setminus [\![S]\!]$
2. *Reduce* $T \leqslant S$ *to checking* $T - S$ *not viable*

Semantic subtyping comes to rescue

$$T \leqslant S \quad \stackrel{\text{def}}{\iff} \quad [\![T]\!] \subseteq [\![S]\!]$$

$$T \leqslant S \quad \iff \quad [\![T]\!] \setminus [\![S]\!] = \emptyset$$

$$T \text{ not viable} \quad \stackrel{\text{def}}{\iff} \quad [\![T]\!] = \emptyset$$

Idea

1. *Compute $T - S$ such that $[\![T - S]\!] = [\![T]\!] \setminus [\![S]\!]$*
2. *Reduce $T \leqslant S$ to checking $T - S$ not viable*

Semantic subtyping comes to rescue

$$T \leqslant S \quad \stackrel{\text{def}}{\iff} \quad [\![T]\!] \subseteq [\![S]\!]$$

$$T \leqslant S \quad \iff \quad [\![T]\!] \setminus [\![S]\!] = \emptyset$$

$$T \text{ not viable} \quad \stackrel{\text{def}}{\iff} \quad [\![T]\!] = \emptyset$$

Idea

1. *Compute $T - S$ such that $[\![T - S]\!] = [\![T]\!] \setminus [\![S]\!]$*
2. *Reduce $T \leqslant S$ to checking $T - S$ not viable*

Behavioral difference $[\![T - S]\!] = [\![T]\!] \setminus [\![S]\!]$

Intuitively

- Along every path shared by both T and S...
- ...turn end to fail

Formally

$$\mathsf{end} - \mathsf{end} = \mathsf{fail}$$

$$\sum_{i \in I} \mathtt{p}?a_i.T_i - \sum_{i \in I \cup J} \mathtt{p}?a_i.S_i = \sum_{i \in I} \mathtt{p}?a_i.(T_i - S_i)$$

$$\bigoplus_{i \in I \cup J} \mathtt{p}!a_i.T_i - \bigoplus_{i \in I} \mathtt{p}!a_i.S_i = \bigoplus_{i \in I} \mathtt{p}!a_i.(T_i - S_i) \oplus \bigoplus_{j \in J} \mathtt{p}!a_j.T_j$$

Proposition

$[\![T - S]\!] \neq \emptyset \iff [\![T]\!] \setminus [\![S]\!] \neq \emptyset$

Fair subtyping, at last

$$\mathsf{fail} \leqslant_{\mathsf{A}} T \qquad \mathsf{end} \leqslant_{\mathsf{A}} \mathsf{end}$$

$$\frac{T_i \leqslant_{\mathsf{A}} S_i \ ^{(i \in I)}}{\sum_{i \in I} \mathtt{p}?a_i.T_i \leqslant_{\mathsf{A}} \sum_{i \in I \cup J} \mathtt{p}?a_i.S_i} \qquad \frac{T_i \leqslant_{\mathsf{A}} S_i \ ^{(i \in I)} \qquad \mathsf{nf}(T - S) = \mathsf{fail}}{T = \bigoplus_{i \in I \cup J} \mathtt{p}!a_i.T_i \leqslant_{\mathsf{A}} \bigoplus_{i \in I} \mathtt{p}!a_i.S_i = S}$$

Theorem

$T \leqslant S$ *iff* $\mathsf{nf}(T) \leqslant_{\mathsf{A}} \mathsf{nf}(S)$

Fair subtyping, at last

$$\mathsf{fail} \leqslant_{\mathsf{A}} T \qquad \mathsf{end} \leqslant_{\mathsf{A}} \mathsf{end}$$

$$\frac{T_i \leqslant_{\mathsf{A}} S_i \ ^{(i \in I)}}{\sum_{i \in I} \mathtt{p}?a_i.T_i \leqslant_{\mathsf{A}} \sum_{i \in I \cup J} \mathtt{p}?a_i.S_i} \qquad \frac{T_i \leqslant_{\mathsf{A}} S_i \ ^{(i \in I)} \qquad \mathsf{nf}(T - S) = \mathsf{fail}}{T = \bigoplus_{i \in I \cup J} \mathtt{p}!a_i.T_i \leqslant_{\mathsf{A}} \bigoplus_{i \in I} \mathtt{p}!a_i.S_i = S}$$

Theorem

$T \leqslant S$ *iff* $\mathsf{nf}(T) \leqslant_{\mathsf{A}} \mathsf{nf}(S)$

Fair subtyping, at last

$$\mathsf{fail} \leqslant_{\mathsf{A}} T \qquad \mathsf{end} \leqslant_{\mathsf{A}} \mathsf{end}$$

$$\frac{T_i \leqslant_{\mathsf{A}} S_i \ ^{(i \in I)}}{\sum_{i \in I} \mathtt{p}?a_i.T_i \leqslant_{\mathsf{A}} \sum_{i \in I \cup J} \mathtt{p}?a_i.S_i} \qquad \frac{T_i \leqslant_{\mathsf{A}} S_i \ ^{(i \in I)} \qquad \mathsf{nf}(T - S) = \mathsf{fail}}{T = \bigoplus_{i \in I \cup J} \mathtt{p}!a_i.T_i \leqslant_{\mathsf{A}} \bigoplus_{i \in I} \mathtt{p}!a_i.S_i = S}$$

Theorem

$T \leqslant S$ *iff* $\mathsf{nf}(T) \leqslant_{\mathsf{A}} \mathsf{nf}(S)$

Fair subtyping, at last

$$\mathsf{fail} \leqslant_{\mathsf{A}} T \qquad \mathsf{end} \leqslant_{\mathsf{A}} \mathsf{end}$$

$$\frac{T_i \leqslant_{\mathsf{A}} S_i \ ^{(i \in I)}}{\sum_{i \in I} \mathtt{p}?a_i.T_i \leqslant_{\mathsf{A}} \sum_{i \in I \cup J} \mathtt{p}?a_i.S_i} \qquad \frac{T_i \leqslant_{\mathsf{A}} S_i \ ^{(i \in I)} \qquad \mathsf{nf}(T - S) = \mathsf{fail}}{T = \bigoplus_{i \in I \cup J} \mathtt{p}!a_i.T_i \leqslant_{\mathsf{A}} \bigoplus_{i \in I} \mathtt{p}!a_i.S_i = S}$$

Theorem

$T \leqslant S$ *iff* $\mathsf{nf}(T) \leqslant_{\mathsf{A}} \mathsf{nf}(S)$

(Fair) subtyping = (fair) testing preorder

- P passes test T
- $P \sqsubseteq Q$ iff P passes test T implies Q passes test T

"Unfair" testing

- De Nicola, Hennessy, **Testing equivalences for processes**, 1983
- ...

Fair testing

- Cleaveland, Natarajan, **Divergence and fair testing**, 1995
- Rensink, Vogler, **Fair testing**, 2007

Fair testing vs fair subtyping

Fair testing

- Cleaveland, Natarajan, **Divergence and fair testing**, 1995
- Rensink, Vogler, **Fair testing**, 2007

– denotational (= obscure) characterization

– no complete deduction system

– exponential

Fair subtyping

+ operational (= hopefully less obscure) characterization

+ complete deduction system

+ polynomial

More on fair subtyping

- Padovani, **Fair Subtyping for Multi-Party Session Types**, COORDINATION 2011

+ formal definitions and proofs
+ algorithms (viability, normal form, subtyping)

Work in progress: fair type checking

$$T = !a.T \oplus !b.\mathsf{end} \qquad\qquad P = u!a.P$$

$$\dfrac{\dfrac{u : T \vdash P}{u : !a.T \vdash u!a.P}\,(\text{T-Output}) \qquad T \leqslant !a.T}{u : T \vdash P}\,(\text{T-Narrow})$$

thank you