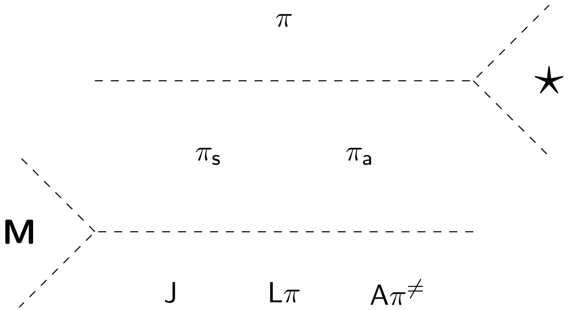
Mixed Choice in Session Types

Kirstin Peters   joint work with Nobuko Yoshida

Universität Augsburg University

March 25, 2024

- Palamidessi proved that the π-calculus with mixed choice (π) is strictly more expressive than the asynchronous π-calculus (π_{a}) via leader election in symmetric networks as distinguishing feature
- but there are more levels of synchrony relevant for the π-calculus



What about the typed fragments of session typed languages that enjoy safety and deadlock-freedom?

$$\mathcal{P}_\pi\text{:}\ P ::= \sum_{i \in \mathrm{I}} \alpha_i.P_i \mid (\nu x)P \mid P \mid P \mid !P \qquad \alpha ::= y(x) \mid \overline{y}z \mid \tau$$

$$\begin{aligned}\mathcal{P}_{\mathsf{CMV}}\text{:}\ P ::=\ & y!v.P \mid y?xP \mid x \triangleleft \mathrm{l}.P \mid x \triangleright \{\mathrm{l}_i : P_i\}_{i \in \mathrm{I}} \\ & \mid P \mid P \mid (\nu yz)P \mid \text{if } v \text{ then } P \text{ else } P \mid \mathbf{0}\end{aligned}$$

$$\mathcal{P}_{\mathsf{CMV}^+}\text{:}\ P ::= y \sum_{i \in \mathrm{I}} M_i \mid P \mid P \mid (\nu yz)P \mid \text{if } v \text{ then } P \text{ else } P \mid \mathbf{0}$$

$$M ::= \mathrm{l}{*}v.P \qquad * ::= ! \mid ?$$

in *Mixed Sessions* by F. Casal, A. Mordido, and V.T. Vasconcelos

$$\begin{aligned}S = (\nu xy)(\ & y\ (\mathrm{l}!\mathsf{false}.S_1 + \mathrm{l}?z.S_2) \mid x\ (\mathrm{l}!\mathsf{true}.\mathbf{0} + \mathrm{l}?z.\mathbf{0}) \mid \\ & y\ (\mathrm{l}!\mathsf{false}.S_3 + \mathrm{l}?z.S_4)\ )\end{aligned}$$

more flexibility: e.g. in produce-consumer examples

- CMV^+ increases the flexibility in comparison to CMV
- Does CMV^+ increase the expressive power ($\text{CMV}^+ > \text{CMV}$)?
- We do not expect that for linear choices, but what about unrestricted?

> Mixed Sessions do not increase the expressive power of choice,
> neither in linear nor unrestricted choices.

- Why is the expressive power of unrestricted choices not increased?

$$\begin{tikzcd} \pi \arrow[d, dashed, bend right, "\text{LE}\ \times"'] \arrow[d, dashed, bend left, "\times\ \star"] \\ \text{CMV}^+ \arrow[r] & \text{CMV} \end{tikzcd}$$

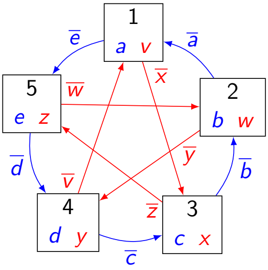
- $\pi \overset{\times}{\dashrightarrow} \text{CMV}^+$ via Leader Election
- $\pi \overset{\times}{\dashrightarrow} \text{CMV}^+$ via the Pattern $\star$
- $\text{CMV}^+ \longrightarrow \text{CMV}$

Definition (Leader Election)

$P = (\nu\tilde{x})(P_1 \mid \ldots \mid P_k)$ *elects a leader* $1 \leq n \leq k$ if for all $P \Longmapsto P'$ there exists $P \Longmapsto P' \Longmapsto P''$ such that $P'''{\downarrow_n}$ for all P''' with $P'' \Longmapsto P'''$, but $P''{\not\Downarrow_m}$ for any $m \in \{1, \ldots, k\}$ with $m \neq n$.

Leader Election in the π-Calculus:

$$\mathsf{S}_{\pi}^{\mathsf{LE}} = (\nu\tilde{n})\,(S_1 \mid S_2 \mid S_3 \mid S_4 \mid S_5)$$
$$S_1 = \overline{e} + a.\big(\overline{x} + v.\overline{1}\big)$$
$$S_2 = \overline{a} + b.\big(\overline{y} + w.\overline{2}\big)$$
$$S_3 = \overline{b} + c.\big(\overline{z} + x.\overline{3}\big)$$
$$S_4 = \overline{c} + d.\big(\overline{v} + y.\overline{4}\big)$$
$$S_5 = \overline{d} + e.\big(\overline{w} + z.\overline{5}\big)$$



$$\mathsf{S}_{\pi}^{\mathsf{LE}} \longmapsto (\nu\tilde{n})\big(\overline{x} + v.\overline{1} \mid S_3 \mid S_4 \mid S_5\big) \longmapsto (\nu\tilde{n})\big(\overline{x} + v.\overline{1} \mid \overline{z} + x.\overline{3} \mid S_5\big)$$
$$\longmapsto \overline{3} \mid (\nu\tilde{n})S_5 \not\longmapsto$$

Theorem ($\pi \not\dashrightarrow \mathrm{CMV}^+$ via Leader Election)

There is no good encoding from the π-calculus into CMV^+.

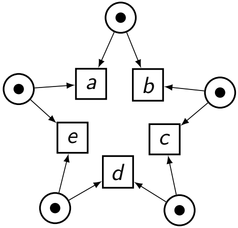
- we cannot solve leader election in symmetric networks of odd degree in CMV^+
- construct a potentially infinite sequence of steps that always eventually restores the symmetry of the original network
- main ingredient: a **confluence lemma**

$$
\begin{tikzcd}
& (\nu \tilde{n})\,(P_1 \mid Q_1) \arrow[dr, mapsto] \arrow[dr, dashed, bend left, "b"] & \\
(\nu \tilde{n})\,(P \mid Q) \arrow[ur, mapsto] \arrow[dr, mapsto] \arrow[ur, dashed, bend left, "a"] \arrow[dr, dashed, bend right, "b"'] & & (\nu \tilde{n})\,(P_3 \mid Q_3) \\
& (\nu \tilde{n})\,(P_2 \mid Q_2) \arrow[ur, mapsto] \arrow[ur, dashed, bend right, "a"'] &
\end{tikzcd}
$$

by the syntax the choice construct is limited to a single channel endpoint

Definition (Synchronisation Pattern $\star$)

- $i : \mathsf{P}^\star \longmapsto P_i$ for $i \in \{a, b, c, d, e\}$ with $P_i \neq P_j$ if $i \neq j$
- a is in conflict with b, b is in conflict with c, …, e is in conflict with a
- every pair of steps in $\{a, b, c, d, e\}$ that is not in conflict is distributable



Synchronisation Pattern $\star$ in the π-Calculus:

$$\mathsf{S}^\star_\pi = \overline{a} + b.\overline{o_b} \mid \overline{b} + c.\overline{o_c} \mid \overline{c} + d.\overline{o_d} \mid \overline{d} + e.\overline{o_e} \mid \overline{e} + a.\overline{o_a}$$

Theorem ($\pi \dashrightarrow\!\!\!\!\!\times\;$ CMV^+ via the Pattern $\star$)

There is no good encoding from the π-calculus into CMV^+.

main ingredient: there are no $\star$ in CMV^+

- assume that there is a $\star$ with five steps a, b, c, d, e
- each step reduces two choices C_i and C_j on matching endpoints
- because of the conflicts, neighbours compete for a choice
- it is impossible to close such a cycle with odd degree

$$
\begin{array}{ccccc}
 & & a & & \\
e \to C_1 & & & & C_2 \leftarrow b \\
C_5 & & & & C_3 \\
d & & C_4 & & c
\end{array}
$$

by the semantics an endpoint can interact with exactly one other endpoint

- *Mixed Sessions* provides an encoding $[\![\cdot]\!]^{\mathsf{CMV}^+}_{\mathsf{CMV}}$ from CMV$^+$ into CMV

$$\begin{aligned} S = (\nu xy)(\ \ & y\,(\mathrm{l}!\mathsf{false}.S_1 + \mathrm{l}?z.S_2) \mid x\,(\mathrm{l}!\mathsf{true}.\mathbf{0} + \mathrm{l}?z.\mathbf{0}) \mid \\ & y\,(\mathrm{l}!\mathsf{false}.S_3 + \mathrm{l}?z.S_4)\ \ ) \end{aligned}$$

$$[\![\Gamma \vdash S]\!]^{\mathsf{CMV}^+}_{\mathsf{CMV}} \Longmapsto T_1$$

$$\begin{aligned} T_1 = (\nu xy)\big(\, & y?c.c \triangleright \{\ \mathrm{l}_? : \Big(c!\mathsf{false}.[\![S_1]\!]^{\mathsf{CMV}^+}_{\mathsf{CMV}} \mid J_1\Big),\\ & \qquad\qquad\quad \mathrm{l}_! : \Big(c?z.[\![S_2]\!]^{\mathsf{CMV}^+}_{\mathsf{CMV}} \mid J_2\Big)\ \} \\ & \mid (\nu st)\big(s \triangleright \{\ \mathrm{l}_1 : (\nu cd)\,(x!c.d \triangleleft \mathrm{l}_!.\,(d!\mathsf{true}.\mathbf{0} \mid J_3)),\\ & \qquad\qquad\qquad \mathrm{l}_2 : (\nu cd)\,(x!c.d \triangleleft \mathrm{l}_?.\,(d?z.\mathbf{0} \mid J_4))\ \} \\ & \qquad\qquad \mid t \triangleleft \mathrm{l}_1.\mathbf{0} \mid t \triangleleft \mathrm{l}_2.\mathbf{0}\big) \\ & \mid y?c.c \triangleright \{\ \mathrm{l}_? : \Big(c!\mathsf{false}.\,[\![S_3]\!]^{\mathsf{CMV}^+}_{\mathsf{CMV}} \mid J_5\Big),\\ & \qquad\qquad\quad \mathrm{l}_! : \Big(c?z.\,[\![S_4]\!]^{\mathsf{CMV}^+}_{\mathsf{CMV}} \mid J_6\Big)\ \}\big) \end{aligned}$$

- *Mixed Sessions* prove operational completeness for $[\![\cdot]\!]^{\mathsf{CMV}^+}_{\mathsf{CMV}}$
- we add the missing soundness proof

Theorem ($\mathsf{CMV}^+ \longrightarrow \mathsf{CMV}$)

The encoding $[\![\cdot]\!]^{\mathsf{CMV}^+}_{\mathsf{CMV}}$ from CMV^+ into CMV is good.
By this encoding source terms in CMV^+ and their literal translations in CMV are related by coupled similarity.

the difference between inputs and outputs in a CMV^+-choice
can be completely captured by labels in CMV-branching

choice in Mixed Sessions can:

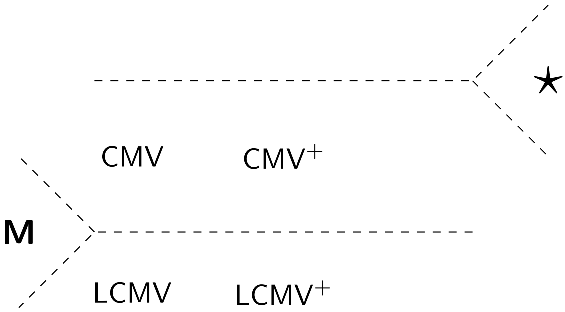
- **not** solve leader election
  (in symmetric networks of odd degree)
- **not** express the synchronisation pattern $\star$
  (the $\star$ captures the expressive power of mixed choice in π)
- express the synchronisation pattern **M**
  (the **M** captures the expressive power of separate choice in π)

$+$

the difference between inputs and outputs in a CMV$^+$-choice can be completely captured by labels in CMV-branching

Corollary (CMV$^+$-Choice is Separate and not Mixed)

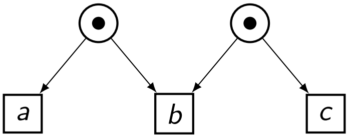
The extension of CMV *given by* CMV$^+$ *introduces a form of separate choice rather than mixed choice.*



- because of unrestricted names, CMV/CMV$^+$ do not ensure deadlock-freedom
- LCMV = linearly typed fragment of CMV
- LCMV$^+$ = linearly typed fragment of CMV$^+$

Synchronisation Pattern **M**

A fully reachable pure **M** in Petri nets [van Glabbeek, Goltz, Schicke '08/'12]:



Theorem

A Petri net is distributable iff it does not contain a fully reachable pure **M**.

[Peters, Nestmann, Goltz '13]:

A process calculus is distributable iff it cannot express a non-local **M**.

Definition (Synchronisation Pattern **M**)

Let $\langle \mathcal{P}, \longmapsto \rangle$ be a process calculus and $\mathrm{P}^{\mathbf{M}} \in \mathcal{P}$ such that:

- $\mathrm{P}^{\mathbf{M}}$ can perform at least three alternative steps $a\colon \mathrm{P}^{\mathbf{M}} \longmapsto P_a$, $b\colon \mathrm{P}^{\mathbf{M}} \longmapsto P_b$, and $c\colon \mathrm{P}^{\mathbf{M}} \longmapsto P_c$ such that P_a, P_b, and P_c are pairwise different.
- The steps a and c are parallel in $\mathrm{P}^{\mathbf{M}}$.
- But b is in conflict with both a and c.

In this case, we denote the process $\mathrm{P}^{\mathbf{M}}$ as **M**. If the steps a and c are distributable in $\mathrm{P}^{\mathbf{M}}$, then we call the **M** *non-local*. Otherwise, the **M** is called *local*.

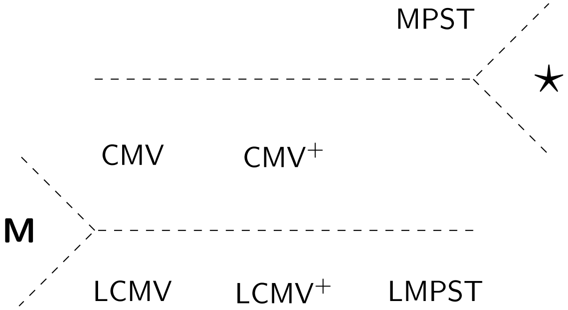
Non-Local **M** in π_a

$$
\begin{tikzcd}[column sep=small]
 & \bar{y}u \mid y(x).P_1 \arrow[dr] \arrow[ddl] & \mid & \bar{y}v \mid y(x).P_2 \arrow[dl] \arrow[ddr] & \\
 & & \boxed{\bar{y}u \mid P_1\{{}^{v}/_{x}\} \mid y(x).P_2} & & \\
\boxed{P_1\{{}^{u}/_{x}\} \mid \bar{y}v \mid y(x).P_2} & & & & \boxed{\bar{y}u \mid y(x).P_1 \mid P_2\{{}^{v}/_{x}\}}
\end{tikzcd}
$$

There are no **M** in LCMV or LCMV$^+$.

- the conflicts in **M** require two competing choices
- choice is limited to exactly two session endpoints
- the conflict between *a* and *b* leads to a conflict between *a* and *c*

- consider MPST as given e.g. in [Honda, Yoshida, Carbone '08]



- LMPST = the fragment of MPST that ensures safety and deadlock-freedom

Corollary (CMV$^+$-Choice is Separate and not Mixed)

The extension of CMV *given by* CMV$^+$ *introduces a form of separate choice rather than mixed choice.*

Reasons:

- Syntax: choice construct is limited to a single channel endpoint
- Semantics: an endpoint can interact with exactly one other endpoint

it is a limitation of the syntax and semantics of the language
but **not of the type system**

helps us to introduce mixed choice to the
unrestricted or non-linear parts of other session calculi

- a decidable and typed (safe and deadlock-free) version of MPST that can express $\star$ is under submission

Thank you for your attention!