On the Expressiveness of MPST

Kirstin Peters   joint work with Nobuko Yoshida

Universität Augsburg University

UNIVERSITY OF OXFORD

June 20, 2025

ICE 2025 in Lille, France

- we analyse the expressive power of variants of MPST focusing on (A)Synchrony
- there are synchronous[1]

$$(k![\tilde{e}]; P_1) \mid (k?(\tilde{x}) \text{ in } P_2) \to P_1 \mid P_2[^{\tilde{c}}/_{\tilde{x}}] \qquad (\tilde{e} \downarrow \tilde{c}) \qquad [Com]$$

and asynchronous[2] variants of MPST

$$s!\langle \tilde{e} \rangle; P \mid s : \tilde{h} \to P \mid s : \tilde{h} \cdot \tilde{v} \qquad \tilde{e} \downarrow \tilde{v} \qquad [Send]$$

$$s?(\tilde{x}); P \mid s : \tilde{v} \cdot \tilde{h} \to P[^{\tilde{v}}/_{\tilde{x}}] \mid s : \tilde{h} \qquad [Recv]$$

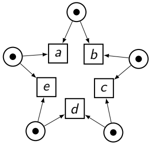
- the kind of semantics does not directly influence the *expressive power*
- we consider *abstract expressive power*, i.e., what systems can do and not how they do that

[1] *Language Primitives and Type Discipline for Structured Communication-Based Programming* by Kohei Honda, Vasco T. Vasconcelos, and Makoto Kubo in ESOP 1998.

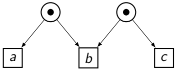
[2] *Multiparty Asynchronous Session Types* by Kohei Honda, Nobuko Yoshida, and Marco Carbone in POPL 2008.

How do we distinguish synchronous and asynchronous expressive power?

- Palamidessi[3] 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
- simpler proofs through *synchronisation pattern*[4]:



all synchronous languages can express a $\star$

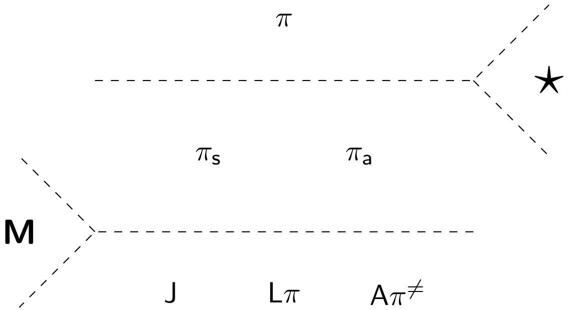


minimal amount of synchronisation
not enough for a synchronous language
but no asynchronous language can express **M**

[3] *Comparing the Expressive Power of the Synchronous and the Asynchronous π-Calculus* by Catuscia Palamidessi in POPL 1997.

[4] *On Distributability in Process Calculi* by Kirstin Peters, Uwe Nestmann, and Ursula Goltz in ESOP 2013.

- there are several levels of synchrony relevant for the π-calculus



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

1. we analyse mixed choice in binary sessions[5]
   - flexible mixed choice construct[6]
   - show that these mixed choices do not raise expressive power
   - see why this is the case
2. we use this information to raise expressiveness in MPST[7]
   - provide a hierarchy showing
     which features of choice influence expressiveness

[5] *Mixed Choice in Session Types* by Kirstin Peters and Nobuko Yoshida in Information and Computation 2024.

[6] *Mixed Sessions* by Filipe Casal, Andreia Mordido, and Vasco T. Vasconcelos in Theoretical Computer Science 2022.

[7] *Separation and Encodability in Mixed Choice Multiparty Sessions* by Kirstin Peters and Nobuko Yoshida in LICS 2024.

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

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

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

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

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

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

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

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

- Why is the expressive power of unrestricted choices <u>not</u> increased?

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

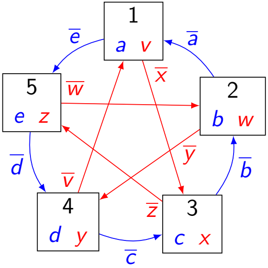
- $\pi \dashrightarrow\!\!\!\!\!\times\;\;$ CMV$^+$ via Leader Election
- $\pi \dashrightarrow\!\!\!\!\!\times\;\;$ CMV$^+$ via the Pattern $\star$
- CMV$^+$ $\longrightarrow$ CMV

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:

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



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

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

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

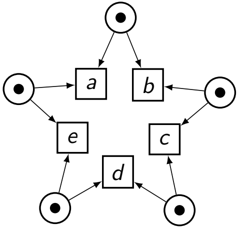
- we cannot solve leader election in symmetric networks of odd degree in CMV$^+$
- construct a potentially infinite sequence of steps that always eventually restores the symmetry of the original network
- main ingredient: a **confluence lemma**

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

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:

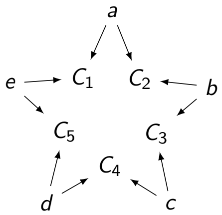
$$\mathsf{S}^\star_\pi = \overline{a} + b.\overline{o_b} \mid \overline{b} + c.\overline{o_c} \mid \overline{c} + d.\overline{o_d} \mid \overline{d} + e.\overline{o_e} \mid \overline{e} + a.\overline{o_a}$$

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



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

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

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

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

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

- *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:

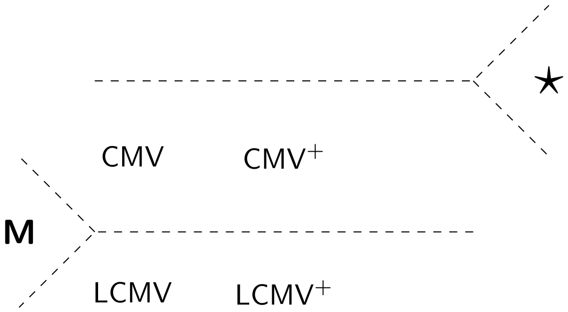
- **not** solve leader election
  (in symmetric networks of odd degree)
- **not** express the synchronisation pattern $\star$
  (the $\star$ captures the expressive power of mixed choice in π)
- express the synchronisation pattern **M**
  (the **M** captures the expressive power of separate choice in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

$+$

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

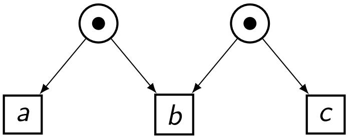
Corollary (CMV$^+$-Choice is Separate and not Mixed)

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

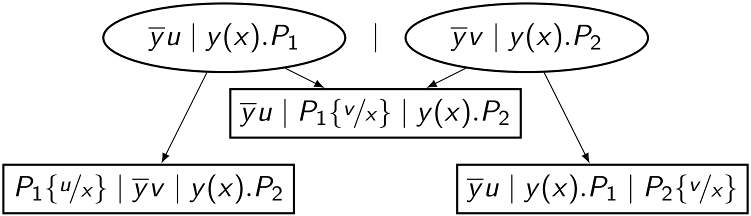
Definition (Synchronisation Pattern **M**)

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

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

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

Non-Local **M** in π_a



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

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:

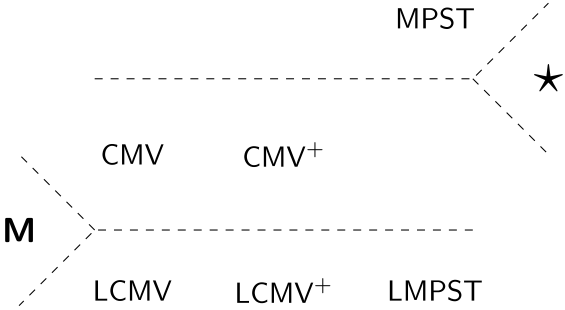
- Syntax: choice construct is limited to a single channel endpoint
- Semantics: an endpoint can interact with exactly one other endpoint

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

Our Impression was: and we were right ;-)

- consider *standard* MPST



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

- since we learned why mixed choice in *Mixed Sessions* did not raise expressiveness, we could in *Separation and Encodability in Mixed Choice Multiparty Sessions* introduce *real* mixed choice in MP

MCMP – nondeterministic mixed choice, single session, no initialisation

Definition (Syntax)

$$
\begin{array}{rclr}
v & ::= & x, y, z, \ldots \;\mid\; 1, 2, \ldots \;\mid\; \mathsf{true}, \mathsf{false} & \text{(variables, numbers, booleans)}\\
\pi & ::= & \mathsf{p}!\ell\langle v\rangle \;\mid\; \mathsf{p}?\ell(x) & \text{(output prefix, input prefix)}\\
P & ::= & \mathbf{0} \;\mid\; \mathsf{X} \;\mid\; \mu\mathsf{X}.P & \text{(nil, proc var, recursion)}\\
 & \mid & \sum_{i\in I} \pi_i.P_i & \text{(mixed choice)}\\
 & \mid & \texttt{if } v \texttt{ then } P \texttt{ else } P & \text{(conditional)}\\
M & ::= & \mathsf{p} \triangleleft P \;\mid\; M \mid M & \text{(multiparty session, parallel)}
\end{array}
$$

- flexible typing and subtyping rules

$$\frac{\forall i \in I, \forall j \in J_i, \; (\Gamma \vdash \pi_i.P_j \triangleright \mathsf{p}_i \dagger_i \ell_i\langle U_i\rangle; T_i)}{\Gamma \vdash \sum_{i \in I} \sum_{j \in J_i} \pi_i.P_j \triangleright \sum_{i \in I} \mathsf{p}_i \dagger_i \ell_i\langle U_i\rangle; T_i} \text{ [TSum]}$$

$$\frac{\forall i \in I, \; T_i \leqslant T_i'}{\sum_{i \in I} \mathsf{p}!\ell_i\langle U_i\rangle; T_i \leqslant \sum_{i \in I \cup J} \mathsf{p}!\ell_i\langle U_i\rangle; T_i'} \text{ [SSel]}$$

$$\frac{\forall i \in I, \; T_i \leqslant T_i'}{\sum_{i \in I \cup J} \mathsf{p}?\ell_i\langle U_i\rangle; T_i \leqslant \sum_{i \in I} \mathsf{p}?\ell_i\langle U_i\rangle; T_i'} \text{ [SBra]}$$

Theorem (Subject Reduction)

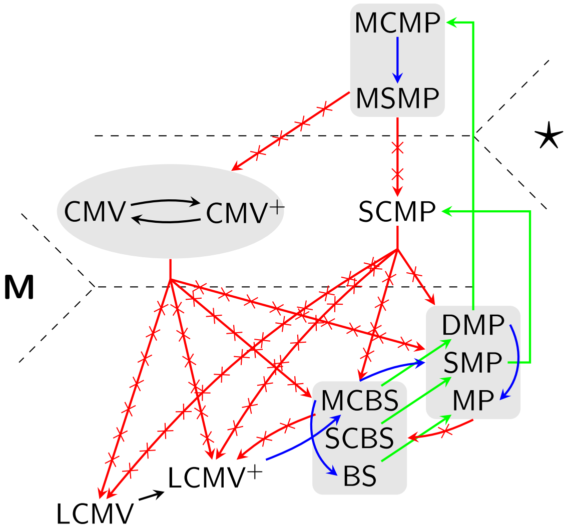
Assume $\Gamma \vdash M \triangleright \Delta$.
If $M \longrightarrow M'$, *then there exists* Δ' *such that* $\Gamma \vdash M' \triangleright \Delta'$ *and* $\Delta \longrightarrow^{*} \Delta'$.

Corollary (Communication Safety)

Assume $\vdash M \triangleright \Delta$.
For all M', *such that* $M \longrightarrow^{*} M'$, M' *is not a session error.*

Theorem (Deadlock-Freedom)

Assume $\vdash M \triangleright \Delta$ *and* $\mathrm{dfree}(\Delta)$. *Then* M *is deadlock-free.*



Thank you for your attention!