On non-commutative logic and process calculi

Matteo Acclavio[1] Giovanni Bernardi[3]
Giulia Manara[2,3] Fabrizio Montesi[4]

[1]University of Southern Denmark
[2]Università degli Studi Roma Tre – Dipartimento di Matematica e Fisica
[3]IRIF, Université Paris Cité

1 November 2023 – Rome, Italy

SCALP 23/24

Non commutative logic and process calculi

Process calculi:

Representing *concurrent systems*

$\leftrightsquigarrow$

Pomset logic :

Extension of MLL with a non-commutative and self-dual connective

$\lhd$

In this presentation

We will talk about:

- A fragment of CCS

In this presentation

We will talk about:

- A fragment of CCS
- Logic system NMAL

In this presentation

We will talk about:

- A fragment of CCS
- Logic system NMAL
- Correspondence between NMAL proofs and CCS processes executions

In this presentation

We will talk about:

- A fragment of CCS
- Logic system NMAL
- Correspondence between NMAL proofs and CCS processes executions
- A known problem - catalyzers

In this presentation

We will talk about:

- A fragment of CCS
- Logic system NMAL
- Correspondence between NMAL proofs and CCS processes executions
- A known problem - catalyzers
- Future work

A fragment of CCS

Given the sets Names = $\{a, b, \ldots\}$ and Conames = $\{\overline{a}, \overline{b}, \ldots\}$

P, Q, G	$:=$	Nil	terminated process
	$\mid$	$P \mid Q$	parallel composition
	$\mid$	$P + Q$	sum - non determinism
	$\mid$	$a.P$	prefixing - execute a, then P

A structural equivalence :

$$P \mid Q \equiv Q \mid P \qquad P \mid (Q \mid R) \equiv (P \mid Q) \mid R \qquad P + Q \equiv Q + P \qquad P \mid \text{Nil} \equiv P$$

Operational Semantic

$$\text{com}\ \frac{}{a.P \mid \overline{a}.Q \rightarrow P \mid Q}$$

interaction is synchronisation

$$\text{sum}\ \frac{P \rightarrow P'}{P + Q \rightarrow P'} \qquad \text{sum}\ \frac{Q \rightarrow Q'}{P + Q \rightarrow Q'}$$

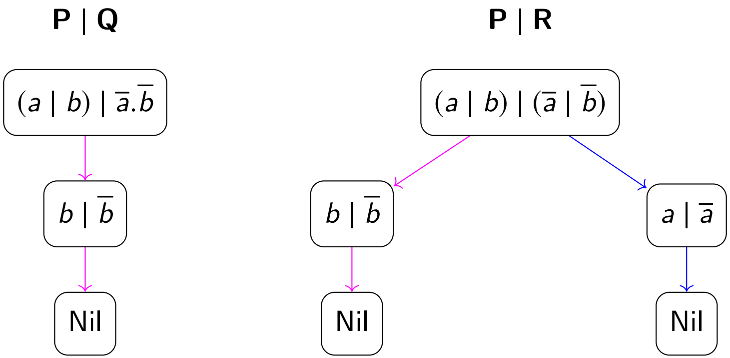
choice between two behaviours

$$\text{Par}\ \frac{P \rightarrow P'}{P \mid Q \rightarrow P' \mid Q} \qquad \text{Par}\ \frac{Q \rightarrow Q'}{P \mid Q \rightarrow P \mid Q'}$$

parallel computation

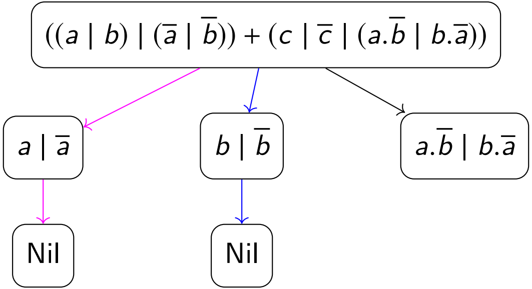
$$\text{struc}\ \frac{P \equiv P' \rightarrow Q' \equiv Q}{P \rightarrow Q}$$

Examples


P | Q
$(a \mid b) \mid \overline{a}.\overline{b}$
$b \mid \overline{b}$
Nil
P | R
$(a \mid b) \mid (\overline{a} \mid \overline{b})$
$b \mid \overline{b}$
$a \mid \overline{a}$
Nil
Nil

Examples

$$(\mathbf{P} \mid \mathbf{R}) + \mathbf{S}$$



NMAL: Non commutive , multiplicative additive logic

$$\text{ax}\ \frac{}{\vdash a, \overline{a}} \qquad \parr\ \frac{\vdash \Gamma, A, B}{\vdash \Gamma, A \parr B} \qquad \otimes\ \frac{\vdash \Gamma, A \quad \vdash B, \Delta}{\vdash \Gamma, A \otimes B, \Delta} \qquad \triangleleft\ \frac{\vdash A, C, \Gamma \quad \vdash B, D, \Delta}{\vdash A \triangleleft B, C \triangleleft D, \Gamma, \Delta}$$

$$\oplus\ \frac{\vdash \Gamma, A_i}{\vdash \Gamma, A_1 \oplus A_2}\ i \in \{1, 2\} \qquad \&\ \frac{\vdash \Gamma, A \quad \vdash \Gamma, B}{\vdash \Gamma, A \,\&\, B} \qquad \text{mix}\ \frac{\vdash \Gamma \quad \vdash \Delta}{\vdash \Gamma, \Delta}$$

NMAL: Non commutive , multiplicative additive logic

$$\text{ax}\ \frac{}{\vdash a, \overline{a}} \qquad \parr\ \frac{\vdash \Gamma, A, B}{\vdash \Gamma, A \parr B} \qquad \otimes\ \frac{\vdash \Gamma, A \quad \vdash B, \Delta}{\vdash \Gamma, A \otimes B, \Delta} \qquad \lhd\ \frac{\vdash A, C, \Gamma \quad \vdash B, D, \Delta}{\vdash A \lhd B, C \lhd D, \Gamma, \Delta}$$

$$\oplus\ \frac{\vdash \Gamma, A_i}{\vdash \Gamma, A_1 \oplus A_2}\ i \in \{1, 2\} \qquad \&\ \frac{\vdash \Gamma, A \quad \vdash \Gamma, B}{\vdash \Gamma, A \,\&\, B} \qquad \text{mix}\ \frac{\vdash \Gamma \quad \vdash \Delta}{\vdash \Gamma, \Delta}$$

Theorem (cut elimination)

Given a proof $\pi \in \text{NMAL} \cup \{\text{cut}\}$ *of conclusion* $\vdash \Gamma$ *there exists a proof* $\pi' \in \text{NMAL}$ *having the same conclusion* $\vdash \Gamma$.

Translation

We translate processes into formulas :

$$\llbracket P \mid Q \rrbracket = \llbracket P \rrbracket \parr \llbracket Q \rrbracket$$

$$a.P = a \triangleleft \llbracket P \rrbracket$$

$$P + Q = \llbracket P \rrbracket \oplus \llbracket Q \rrbracket$$

Some intuitions

Reduction steps and NMAL's rules look alike

$$\dfrac{}{a.P \mid \overline{a}.Q \longrightarrow P \mid Q}\, com \quad \rightsquigarrow \quad \dfrac{\dfrac{\dfrac{}{\vdash a, \overline{a}}\, ax \qquad \vdash [\![P]\!], [\![Q]\!]}{\vdash a \triangleleft [\![P]\!], \overline{a} \triangleleft [\![Q]\!]}\, \triangleleft}{\vdash a \triangleleft [\![P]\!] \parr \overline{a} \triangleleft [\![Q]\!]}\, \parr$$

$$\dfrac{P \longrightarrow P'}{P + Q \longrightarrow P'}\, sum \quad \rightsquigarrow \quad \dfrac{\begin{array}{c} \vdash [\![P]\!]' \\ \pi \, \| \\ \vdash [\![P]\!] \end{array}}{\vdash [\![P]\!] \oplus [\![Q]\!]}\, \oplus$$

One reduction step : one linear implication

Theorem

If $P \longrightarrow Q$ *then* $\vdash [\![Q]\!] \multimap [\![P]\!]$ *is provable in* NMAL

One reduction step : one linear implication

Theorem

If $P \longrightarrow Q$ *then* $\vdash [\![Q]\!] \multimap [\![P]\!]$ *is provable in* NMAL

Proof.

By induction. If $P \equiv a.P_1 \mid \overline{a}.P_2$ and $Q \equiv P_1 \mid P_2$ then

$$
\parr\;\dfrac{\lhd\;\dfrac{ax\;\dfrac{}{a, \overline{a}} \qquad \otimes\;\dfrac{\dfrac{\pi_1 \,\|}{\overline{[\![P]\!]_1}, [\![P_1]\!]} \qquad \dfrac{\pi_2 \,\|}{\overline{[\![P]\!]_2}, [\![P_2]\!]}}{\overline{[\![P]\!]_1} \otimes \overline{[\![P]\!]_2}, [\![P_1]\!], [\![P_2]\!]}}{\overline{[\![P]\!]_1} \otimes \overline{[\![P]\!]_2}, a. \lhd [\![P_1]\!], \overline{a}. \lhd [\![P_2]\!]}}{\overline{[\![P]\!]_1} \otimes \overline{[\![P]\!]_2}, a. \lhd [\![P_1]\!] \parr \overline{a}. \lhd [\![P_2]\!]}
$$

$\square$

Correspondence between executions and proofs

Theorem (Execution to proof)

If $P_n \to \cdots \to P_0 \to \mathsf{Nil}$, *then* $\vdash [\![P_n]\!]$ *is provable in* NMAL.

Correspondence between executions and proofs

Theorem (Execution to proof)

If $P_n \longrightarrow \cdots \longrightarrow P_0 \longrightarrow \mathsf{Nil}$, *then* $\vdash [\![P_n]\!]$ *is provable in* NMAL.

Proof.

We define a proof π_n of P_n by induction:

- if $n = 0$, then $P_0 \longrightarrow \mathsf{Nil}$ and $P_0 = a \mid \overline{a}$. Thus $\pi_0 = \dfrac{\dfrac{}{a, \overline{a}}\,\mathrm{ax}}{a \parr \overline{a}}\,\parr$
- if $n > 0$, then $\dfrac{\begin{array}{c}\pi_{n-1} \,\Vert\, \mathrm{IH} \\ [\![P_{n-1}]\!]\end{array} \quad \begin{array}{c}\Vert\, \mathrm{Thm} \\ [\![P_{n-1}]\!] \multimap [\![P_n]\!]\end{array}}{[\![P_n]\!]}\,\mathrm{cut} \quad \rightsquigarrow_{\text{cut-elim}} \quad \begin{array}{c}\Vert\, \mathrm{NMAL} \\ [\![P_n]\!]\end{array} = \pi_n$

□

Correspondence between executions and proofs

Theorem (Execution to proof)

If $P_n \longrightarrow \cdots \longrightarrow P_0 \longrightarrow \mathsf{Nil}$, *then* $\vdash [\![P_n]\!]$ *is provable in* NMAL.

Proof.

We define a proof π_n of P_n by induction:

- if $n = 0$, then $P_0 \longrightarrow \mathsf{Nil}$ and $P_0 = a \mid \overline{a}$. Thus $\pi_0 = \dfrac{\dfrac{}{a, \overline{a}}\,\mathrm{ax}}{a \parr \overline{a}}\,\parr$

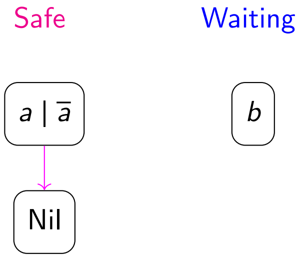
- if $n > 0$, then $$\dfrac{\dfrac{\pi_{n-1} \,\Vert\, \mathrm{IH}}{[\![P_{n-1}]\!]} \quad \dfrac{\Vert\, \mathrm{Thm}}{[\![P_{n-1}]\!] \multimap [\![P_n]\!]}}{[\![P_n]\!]}\,\mathrm{cut} \quad \rightsquigarrow_{\text{cut-elim}} \quad \dfrac{\Vert\, \mathrm{NMAL}}{[\![P_n]\!]} = \pi_n$$

□

Theorem (Proof to execution)

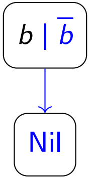
$[\![P]\!]$ *is provable in* NMAL *then* $P \longrightarrow^* \mathsf{Nil}$

A natural question



Is P safe? Waiting?

Catalyzers



$\overline{b}$ is a *catalyzer* for b

How to find a catalyzer for P?

Yes we can

$$\text{Com}^{\checkmark}\ \frac{C \vdash \Gamma}{C \vdash a, \overline{a}, \Gamma} \qquad \text{Par}\ \frac{C \vdash \Gamma, A, B}{C \vdash \Gamma, A \parr B} \qquad \text{Plus}\ \frac{C \vdash \Gamma, A_i}{C \vdash \Gamma, A_1 \oplus A_2}\ i \in \{1, 2\}$$

$$\text{Com}\ \frac{C \vdash \Gamma, A, B}{C \vdash \Gamma, a \triangleleft A, \overline{a} \triangleleft B}$$

$$\text{Release}\ \frac{C \triangleleft \overline{a} \vdash \Gamma, A}{C \vdash \Gamma, a \triangleleft A}\ \text{if no other rule is applicable}$$

Example

$$P = a.\overline{b} \mid b.\overline{a}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\color{blue}{\overline{a} \triangleleft a} \vdash}\ \text{Release}}{\color{blue}{\overline{a}} \vdash \overline{a}}\ \text{Com}}{\color{blue}{\overline{a}} \vdash \overline{b}, b \triangleleft \overline{a}}\ \text{Release}}{\vdash a \triangleleft \overline{b}, b \triangleleft \overline{a}}\ \text{Par}}{\vdash (a \triangleleft \overline{b}) \parr (b \triangleleft \overline{a})}$$

$\color{blue}{\overline{a}.a}$ is a catalyzer for P

And for full CCS ?

P, Q, G	$:=$	Nil	terminated process
	$\mid$	$P \mid Q$	parallel composition
	$\mid$	$P + Q$	sum - non determinism
	$\mid$	$a.P$	prefixing - execute a, then P
	$\mid$	$\nu a.P$	restriction - makes a private

Communication only allowed under ν

$$\textit{com}\ \frac{}{\nu a.(a.P \mid \overline{a}.Q) \rightarrow \nu a.(P \mid Q)}$$

Not really a smooth operator...

To deadlock or not to deadlock?

Safe	Waiting	Deadlocked
$\nu a.(a \mid \overline{a})$ $\rightarrow$ Nil	b	$\nu b.b$
		$\nu a.\nu b.(a.\overline{b} \mid b.\overline{a})$

Is P safe, waiting or deadlocked?

And the catalyzer ?

Work in progress

Deadlock war:

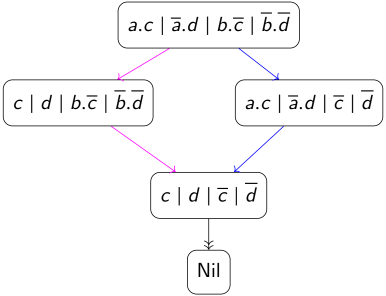
Enriching Session Types : Kobayashi [5]

Quest for catalyzers:

CCS : Bernardi et al. [1]
π-calculus : Montesi et al.[3]

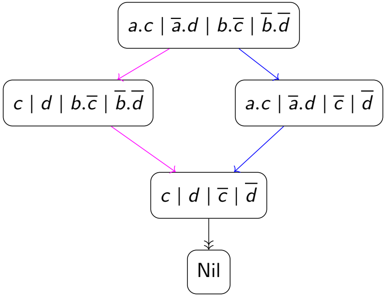
IDEA: use NMAL to give a unified approach

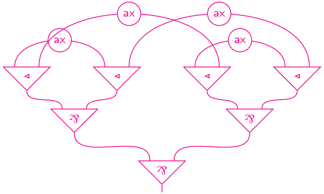
Future work



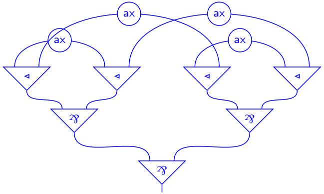
$$\parr \dfrac{\triangleleft \dfrac{\text{ax}\dfrac{}{a, \overline{a}} \qquad \triangleleft \dfrac{\text{ax}\dfrac{}{b, \overline{b}} \qquad \text{ax-mix}\dfrac{\text{ax}\dfrac{}{c, \overline{c}} \qquad \text{ax}\dfrac{}{d, \overline{d}}}{c, d, \overline{c}, \overline{d}}}{c, d, b \triangleleft \overline{c}, \overline{b} \triangleleft \overline{d}}}{a \triangleleft c, \overline{a} \triangleleft d, b \triangleleft \overline{c}, \overline{b} \triangleleft \overline{d}}}{((a \triangleleft c) \parr (\overline{a} \triangleleft d)) \parr ((b \triangleleft \overline{c}) \parr (\overline{b} \triangleleft \overline{d}))} \quad \sim \quad \parr \dfrac{\triangleleft \dfrac{\text{ax}\dfrac{}{b, \overline{b}} \qquad \triangleleft \dfrac{\text{ax}\dfrac{}{a, \overline{a}} \qquad \text{ax-mix}\dfrac{\text{ax}\dfrac{}{c, \overline{c}} \qquad \text{ax}\dfrac{}{d, \overline{d}}}{c, d, \overline{c}, \overline{d}}}{a \triangleleft c, \overline{a} \triangleleft d, \overline{c}, \overline{d}}}{a \triangleleft c, \overline{a} \triangleleft d, b \triangleleft \overline{c}, \overline{b} \triangleleft \overline{d}}}{((a \triangleleft c) \parr (\overline{a} \triangleleft d)) \parr ((b \triangleleft \overline{c}) \parr (\overline{b} \triangleleft \overline{d}))}$$

Future work





=



$((a \lhd c) \parr (\overline{a} \lhd d)) \parr ((b \lhd \overline{c}) \parr (\overline{b} \lhd \overline{d}))$

$((a \lhd c) \parr (\overline{a} \lhd d)) \parr ((b \lhd \overline{c}) \parr (\overline{b} \lhd \overline{d}))$

Bibliography

Giovanni Bernardi and Adrian Francalanza.
Full-abstraction for client testing preorders.
Science of Computer Programming, 168:94–117, 2018.

Paola Bruscoli.
A purely logical account of sequentiality in proof search.
In Peter J. Stuckey, editor, *Logic Programming*, pages 302–316, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

Marco Carbone, Ornela Dardha, and Fabrizio Montesi.
Progress as compositional lock-freedom.
In Eva Kühn and Rosario Pugliese, editors, *Coordination Models and Languages - 16th IFIP WG 6.1 International Conference, COORDINATION 2014, Held as Part of the 9th International Federated Conferences on Distributed Computing Techniques, DisCoTec 2014, Berlin, Germany, June 3-5, 2014, Proceedings*, volume 8459 of *Lecture Notes in Computer Science*, pages 49–64. Springer, 2014.

Ross Horne and Alwen Tiu.
Towards proofs as successful executions of processes.
2016.

Naoki Kobayashi.
A new type system for deadlock-free processes.
In Christel Baier and Holger Hermanns, editors, *CONCUR 2006 – Concurrency Theory*, pages 233–247, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

Christian Retoré.
Pomset Logic: The Other Approach to Noncommutativity in Logic, page 299–345.
Springer International Publishing, 2021.