Processes as Formulas, and Choreographies as Proofs

Matteo Acclavio

UNIVERSITY OF SUSSEX

Based on joint works with Giulia Manara and Fabrizio Montesi

Southern and Midlands Logic Seminar, Brighton 11/12/2024

What I want

An elegant logical framework
to reason about concurrent programs and
to design concurrent programming languages

Example: choreographic programming

Idea behind choreographic programming: specify the system globally...

$$C, C_\ell \coloneqq \underset{\text{end}}{\mathbf{0}} \quad | \quad \underset{\text{communication}}{[\mathsf{p}.x \to \mathsf{q}.y]; C} \quad | \quad \underset{\text{choice}}{\mathsf{p}.L \to \mathsf{q}.L' : k \left\{ \begin{array}{c|c} \ell : C_\ell & \ell \in L \\ \ell : S_\ell & \ell \in L' \setminus L \end{array} \right\}} \quad | \quad \underset{\text{restriction}}{(\nu x) C^x}$$

Example: choreographic programming

Idea behind choreographic programming: specify the system globally...

$$C, C_\ell \coloneqq \underset{\text{end}}{0} \quad | \quad \underset{\text{communication}}{[\mathsf{p}.x \rightarrow \mathsf{q}.y]; C} \quad | \quad \underset{\text{choice}}{\mathsf{p}.L \rightarrow \mathsf{q}.L' : k \left\{ \begin{array}{c|c} \ell : C_\ell & \ell \in L \\ \ell : S_\ell & \ell \in L' \setminus L \end{array} \right\}} \quad | \quad \underset{\text{restriction}}{(\nu x) C^x}$$

...and then compile local programs

$$\mathrm{EPP}(C) = \mathrm{EPP}_{\mathsf{p}_1}(C) \mid \cdots \mid \mathrm{EPP}_{\mathsf{p}_n}(C)$$

Main benefit: deadlock-free by design

Example: choreographic programming

Idea behind choreographic programming: specify the system globally. . .

$$C, C_\ell \coloneqq \underset{\text{end}}{0} \quad | \quad \underset{\text{communication}}{[\mathsf{p}.x \to \mathsf{q}.y]; C} \quad | \quad \underset{\text{choice}}{\mathsf{p}.L \to \mathsf{q}.L' : k \left\{ \begin{array}{l|l} \ell : C_\ell & \ell \in L \\ \ell : S_\ell & \ell \in L' \setminus L \end{array} \right\}} \quad | \quad \underset{\text{restriction}}{(\nu x) C^x}$$

. . . and then compile local programs

$$\mathrm{EPP}(C) = \mathrm{EPP}_{\mathsf{p}_1}(C) \mid \cdots \mid \mathrm{EPP}_{\mathsf{p}_n}(C)$$

Main benefit: deadlock-free by design

Question: are choreographies complete wrt deadlock-freedom?

Context

Off-the-shelf Solutions

Dynamic logics

Dynamic logics

Programs in PDL = regular languages (elements of a Kleene Algebra)

Dynamic logics

Programs in PDL = regular languages (elements of a Kleene Algebra)

however

Kleene Algebra + $\underbrace{\text{commutations}}_{\text{interleaving}} \overset{\text{Kozen '96}}{\Longrightarrow}$ undecidability whether $\alpha = \beta$

Dynamic logics

Programs in PDL = regular languages (elements of a Kleene Algebra)

however

$$\text{Kleene Algebra} + \underbrace{\text{commutations}}_{\color{red}{\text{interleaving}}} \overset{\text{Kozen '96}}{\Longrightarrow} \text{undecidability whether } \alpha = \beta$$

Thus

in any "concurrent-PDL" $\vdash [\alpha]\top \Leftrightarrow [\beta]\top$ is undecidable[1]

[1] A possible solution is to separate the trace reasoning from the operational semantics (A. et al '24 "On Propositional Dynamic Logic and Concurrency")

Session types

Session types

Typing disciplines for the π-calculus.

Binary session types: strong restriction or syntactically complex

Multiparty session types: bad compositionality properties

Session types

Typing disciplines for the π-calculus.

Binary session types: strong restriction or syntactically complex

Multiparty session types: bad compositionality properties

Why?

Struggling to model interleaving.

Computatation-as-deduction

Computatation-as-deduction

Program	=	Inference system
Current state of a program	=	Sequent
Execution	=	Proof search

Computatation-as-deduction

Program	=	Inference system
Current state of a program	=	Sequent
Execution	=	Proof search

Non-determinism

$$\underbrace{\text{don't care}}_{\text{interleaving}} \text{ VS } \underbrace{\text{don't know}}_{\text{``conflict''}}$$

Good news everyone!

We have syntaxes capturing the "don't care" non-determinism!

$$
\dfrac{
  \dfrac{
    \dfrac{
      \dfrac{}{a, a^\perp}\,\text{ax}
      \quad
      \dfrac{}{b^\perp, b}\,\text{ax}
    }{a, a^\perp \otimes b^\perp, b}\,\otimes
  }{a \parr (a^\perp \otimes b^\perp), b}\,\parr
  \qquad
  \dfrac{
    \dfrac{}{c, c^\perp}\,\text{ax}
    \quad
    \dfrac{}{d^\perp, d}\,\text{ax}
  }{c, c^\perp \otimes d^\perp, d}\,\otimes
}{
  \dfrac{a \parr (a^\perp \otimes b^\perp), b \otimes c, d, c^\perp \otimes d^\perp}{a \parr (a^\perp \otimes b^\perp), (b \otimes c) \parr d, c^\perp \otimes d^\perp}\,\parr
}\,\otimes
\quad \simeq \quad
\dfrac{
  \dfrac{
    \dfrac{}{a, a^\perp}\,\text{ax}
    \quad
    \dfrac{
      \dfrac{
        \dfrac{
          \dfrac{}{b^\perp, b}\,\text{ax}
          \quad
          \dfrac{}{c, c^\perp}\,\text{ax}
        }{b^\perp, b \otimes c, c^\perp}\,\otimes
        \quad
        \dfrac{}{d^\perp, d}\,\text{ax}
      }{b^\perp, b \otimes c, c^\perp \otimes d^\perp, d}\,\otimes
    }{(b \otimes c) \parr d, c^\perp \otimes d^\perp}\,\parr
  }{a, a^\perp \otimes b^\perp, (b \otimes c) \parr d, c^\perp \otimes d^\perp}\,\otimes
}{a \parr (a^\perp \otimes b^\perp), (b \otimes c) \parr d, c^\perp \otimes d^\perp}\,\parr
$$

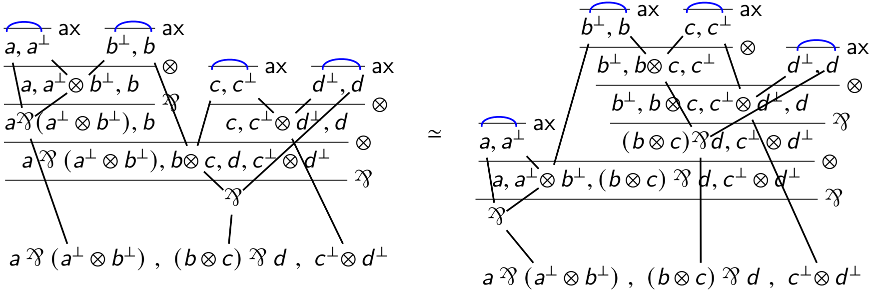
Good news everyone!

We have syntaxes capturing the “don’t care” non-determinism!

$$
\dfrac{\dfrac{\dfrac{\dfrac{}{a, a^\perp}\,\text{ax} \quad \dfrac{}{b^\perp, b}\,\text{ax}}{a, a^\perp \otimes b^\perp, b}\,\otimes}{a \parr (a^\perp \otimes b^\perp), b}\,\parr \quad \dfrac{\dfrac{}{c, c^\perp}\,\text{ax} \quad \dfrac{}{d^\perp, d}\,\text{ax}}{c, c^\perp \otimes d^\perp, d}\,\otimes}{\dfrac{a \parr (a^\perp \otimes b^\perp), b \otimes c, d, c^\perp \otimes d^\perp}{a \parr (a^\perp \otimes b^\perp), (b \otimes c) \parr d, c^\perp \otimes d^\perp}\,\parr}\,\otimes
\quad \simeq \quad
\dfrac{\dfrac{\dfrac{}{a, a^\perp}\,\text{ax} \quad \dfrac{\dfrac{\dfrac{\dfrac{}{b^\perp, b}\,\text{ax} \quad \dfrac{}{c, c^\perp}\,\text{ax}}{b^\perp, b \otimes c, c^\perp}\,\otimes \quad \dfrac{}{d^\perp, d}\,\text{ax}}{b^\perp, b \otimes c, c^\perp \otimes d^\perp, d}\,\otimes}{(b \otimes c) \parr d, c^\perp \otimes d^\perp}\,\parr}{a, a^\perp \otimes b^\perp, (b \otimes c) \parr d, c^\perp \otimes d^\perp}\,\otimes}{a \parr (a^\perp \otimes b^\perp), (b \otimes c) \parr d, c^\perp \otimes d^\perp}\,\parr
$$

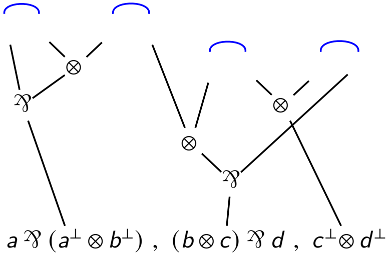
Good news everyone!

We have syntaxes capturing the "don't care" non-determinism!

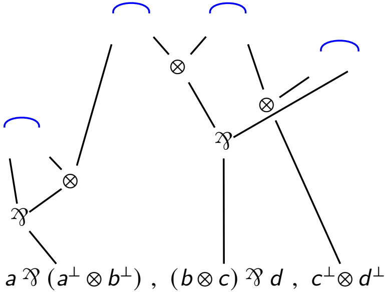


Good news everyone!

We have syntaxes capturing the “don’t care” non-determinism!



$\simeq$



The π-Calculus

Syntax and semantics of the π-calculus

$P, Q, R \coloneqq$ Nil terminated prog.

Syntax and semantics of the π-calculus

$P, Q, R \coloneqq$	Nil	terminated prog.			
$\mid$	$P \mid Q$	parallel	Par :	$P \mid Q \rightarrow P' \mid Q$	if $P \rightarrow P'$

Syntax and semantics of the π-calculus

$P, Q, R \coloneqq$	Nil	terminated prog.			
$\mid$	$P \mid Q$	parallel	Par :	$P \mid Q \to P' \mid Q$	if $P \to P'$
$\mid$	$(\nu x)P$	restriction (or nu)	Res :	$(\nu x)P \to (\nu x)P'$	if $P \to P'$

Syntax and semantics of the π-calculus

$P, Q, R \coloneqq$	Nil	terminated prog.			
$\mid$	$P \mid Q$	parallel	Par :	$P \mid Q \rightarrow P' \mid Q$	if $P \rightarrow P'$
$\mid$	$(\nu x)P$	restriction (or nu)	Res :	$(\nu x)P \rightarrow (\nu x)P'$	if $P \rightarrow P'$
$\mid$	$x!\langle y \rangle.P$	send (y on x)			
$\mid$	$x?(y).P$	receive (y on x)	Com :	$x!\langle a \rangle.P \mid x?(b).Q \rightarrow P \mid Q[a/b]$	

Syntax and semantics of the π-calculus

$P, Q, R \coloneqq$	Nil	terminated prog.			
$\mid$	$P \mid Q$	parallel	Par :	$P \mid Q \to P' \mid Q$	if $P \to P'$
$\mid$	$(\nu x)P$	restriction (or nu)	Res :	$(\nu x)P \to (\nu x)P'$	if $P \to P'$
$\mid$	$x!\langle y \rangle.P$	send (y on x)			
$\mid$	$x?(y).P$	receive (y on x)	Com :	$x!\langle a \rangle.P \mid x?(b).Q \to P \mid Q[a/b]$	
$\mid$	$P + Q$	Choice	Plus :	$(P + Q) \mid R \to P' \mid R'$	if $P \mid R \to P' \mid R'$

Syntax and semantics of the π-calculus

$$
\begin{array}{rlll}
P, Q, R \coloneqq & \mathsf{Nil} & \text{terminated prog.} & \\
\mid & P \mid Q & \text{parallel} & \text{Par}: \quad P \mid Q \to P' \mid Q \quad \text{if } P \to P' \\
\mid & (\nu x)P & \text{restriction (or nu)} & \text{Res}: \quad (\nu x)P \to (\nu x)P' \quad \text{if } P \to P' \\
\mid & x!\langle y\rangle.P & \text{send } (y \text{ on } x) & \\
\mid & x?(y).P & \text{receive } (y \text{ on } x) & \text{Com}: \quad x!\langle a\rangle.P \mid x?(b).Q \to P \mid Q[a/b] \\
\mid & x \triangleleft \{\ell : P_\ell\}_{\ell \in L} & \text{label-send (on } x) & \text{Choice}: \quad x \triangleleft \{\ell : P_\ell\}_{\ell \in L} \to x \triangleleft \left\{\ell_k : P_{\ell_k}\right\} \quad \text{if } \ell_k \in L \\
\mid & x \triangleright \{\ell : P_\ell\}_{\ell \in L} & \text{label-receive (on } x) & \text{Sel}: \quad x \triangleleft \left\{\ell_k : P_{\ell_k}\right\} \mid x \triangleright \{\ell : Q_\ell\}_{\ell \in L} \to P_{\ell_k} \mid Q_{\ell_k} \quad \text{if } \ell_k \in L
\end{array}
$$

Syntax and semantics of the π-calculus

$$
\begin{array}{rll}
P, Q, R \coloneqq & \text{Nil} & \text{terminated prog.} \\
\mid & P \mid Q & \text{parallel} \\
\mid & (\nu x)P & \text{restriction (or nu)} \\
\mid & x!\langle y\rangle.P & \text{send } (y \text{ on } x) \\
\mid & x?(y).P & \text{receive } (y \text{ on } x) \\
\mid & x \triangleleft \{\ell : P_\ell\}_{\ell\in L} & \text{label-send (on } x) \\
\mid & x \triangleright \{\ell : P_\ell\}_{\ell\in L} & \text{label-receive (on } x)
\end{array}
$$

$$
\begin{array}{rrl}
\text{Par}: & P \mid Q \to P' \mid Q & \text{if } P \to P' \\
\text{Res}: & (\nu x)P \to (\nu x)P' & \text{if } P \to P' \\
\text{Com}: & x!\langle a\rangle.P \mid x?(b).Q \to P \mid Q[a/b] & \\
\text{Choice}: & x \triangleleft \{\ell : P_\ell\}_{\ell\in L} \to x \triangleleft \left\{\ell_k : P_{\ell_k}\right\} & \text{if } \ell_k \in L \\
\text{Sel}: & x \triangleleft \left\{\ell_k : P_{\ell_k}\right\} \mid x \triangleright \{\ell : Q_\ell\}_{\ell\in L} \to P_{\ell_k} \mid Q_{\ell_k} & \text{if } \ell_k \in L
\end{array}
$$

$$
\begin{array}{rclrcl}
P \mid Q & \Leftrightarrow & Q \mid P & (P \mid Q) \mid R & \Leftrightarrow & P \mid (Q \mid R) \\
(\nu x)(\nu y)P & \Leftrightarrow & (\nu y)(\nu x)P & P \mid \text{Nil} & \Rrightarrow & P \\
x \triangleleft \{\ell : (\nu y)P_\ell\}_{\ell\in L} & \Leftrightarrow & (\nu y)\,(x \triangleleft \{\ell : P_\ell\}_{\ell\in L}) & x \triangleright \{\ell : (\nu y)P_\ell\}_{\ell\in L} & \Leftrightarrow & (\nu y)\,(x \triangleright \{\ell : P_\ell\}_{\ell\in L}) \\
(\nu x)S & \Rrightarrow & S & (\nu x)P \mid S & \Rrightarrow & (\nu x)(P \mid S)
\end{array}
$$

with $x \notin \text{free}(S)$

$$\text{Struc}: \quad P \to Q \quad P \equiv P' \to Q' \equiv Q$$

Syntax and semantics of the π-calculus

$P, Q, R \coloneqq$ Nil	terminated prog.			
$\mid P \mid Q$	parallel	Par :	$P \mid Q \to P' \mid Q$	if $P \to P'$
$\mid (\nu x)P$	restriction (or nu)	Res :	$(\nu x)P \to (\nu x)P'$	if $P \to P'$
$\mid x!\langle y\rangle.P$	send (y on x)			
$\mid x?(y).P$	receive (y on x)	Com :	$x!\langle a\rangle.P \mid x?(b).Q \to P \mid Q[a/b]$	
$\mid x \triangleleft \{\ell : P_\ell\}_{\ell\in L}$	label-send (on x)	Choice :	$x \triangleleft \{\ell : P_\ell\}_{\ell\in L} \to x \triangleleft \left\{\ell_k : P_{\ell_k}\right\}$	if $\ell_k \in L$
$\mid x \triangleright \{\ell : P_\ell\}_{\ell\in L}$	label-receive (on x)	Sel :	$x \triangleleft \left\{\ell_k : P_{\ell_k}\right\} \mid x \triangleright \{\ell : Q_\ell\}_{\ell\in L} \to P_{\ell_k} \mid Q_{\ell_k}$	if $\ell_k \in L$

$$\begin{array}{rclrcl}
P \mid Q & \Leftrightarrow & Q \mid P & (P \mid Q) \mid R & \Leftrightarrow & P \mid (Q \mid R) \\
(\nu x)(\nu y)P & \Leftrightarrow & (\nu y)(\nu x)P & P \mid \mathrm{Nil} & \Rrightarrow & P \\
x \triangleleft \{\ell : (\nu y)P_\ell\}_{\ell\in L} & \Leftrightarrow & (\nu y)\,(x \triangleleft \{\ell : P_\ell\}_{\ell\in L}) & x \triangleright \{\ell : (\nu y)P_\ell\}_{\ell\in L} & \Leftrightarrow & (\nu y)\,(x \triangleright \{\ell : P_\ell\}_{\ell\in L}) \\
(\nu x)S & \Rrightarrow & S & (\nu x)P \mid S & \Rrightarrow & (\nu x)(P \mid S)
\end{array}$$

with $x \notin \mathrm{free}(S)$

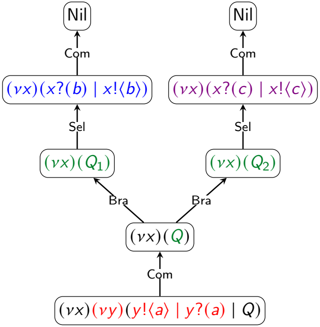
$$\mathrm{Struc}^{\Rrightarrow} : \quad P \to Q \quad \begin{cases} \text{if } P \Rrightarrow^* P' = \mathcal{P}[S] \to \mathcal{P}[S'] = Q \\ \text{with } P \neq P' \text{and} S \to S' \text{ not via Struc}^{\Rrightarrow} \end{cases}$$

Example

$$(\nu x)(\nu y)\left(y!\langle a\rangle \ \mid\ y?(a)\ \mid x \triangleright \left\{ \ell_1 \ :\ x?(b)\, ,\, \ell_2 \ :\ x!\langle c\rangle \right\} \mid x \triangleleft \left\{ \ell_1 \ :\ x!\langle b\rangle\, ,\, \ell_2 \ :\ x?(c) \right\}\right)$$

Example

$$(\nu x)(\nu y)\left(y!\langle a\rangle \mid y?(a) \mid x \triangleright \left\{\ell_1 : x?(b), \ell_2 : x!\langle c\rangle\right\} \mid x \triangleleft \left\{\ell_1 : x!\langle b\rangle, \ell_2 : x?(c)\right\}\right)$$



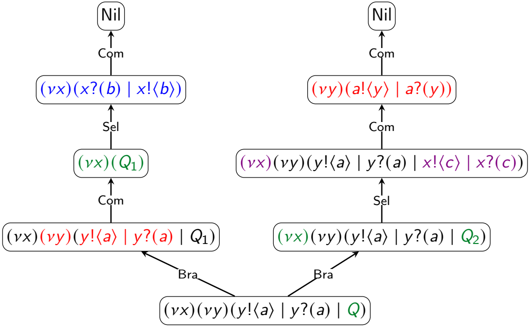
with

$$Q = x \triangleleft \begin{Bmatrix} \ell_1 : x?(b) \\ \ell_2 : c!\langle x\rangle \end{Bmatrix} \qquad R = x \triangleright \begin{Bmatrix} \ell_1 : x!\langle b\rangle \\ \ell_2 : x?(c) \end{Bmatrix}$$

$$Q_1 = x \triangleleft \{\ell_1 : x?(b)\} \qquad Q_2 = x \triangleleft \{\ell_2 : x!\langle c\rangle\}$$

Example

$$(\nu x)(\nu y)\left(y!\langle a\rangle \;|\; y?(a) \;|\; x \triangleright \left\{ \ell_1 \,:\, x?(b)\,,\, \ell_2 \,:\, x!\langle c\rangle \right\} \;|\; x \triangleleft \left\{ \ell_1 \,:\, x!\langle b\rangle\,,\, \ell_2 \,:\, x?(c) \right\}\right)$$

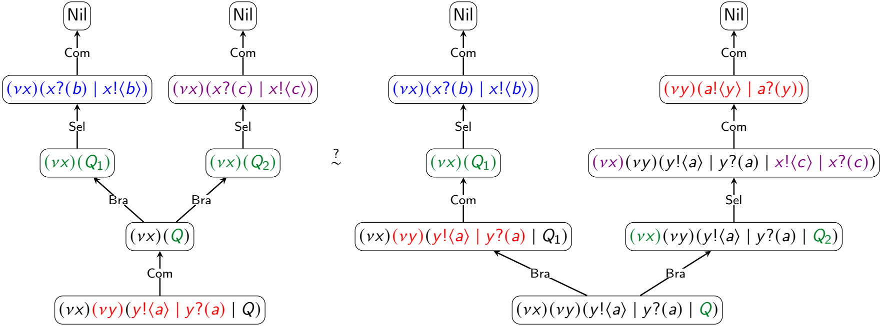


with

$$Q = x \triangleleft \begin{Bmatrix} \ell_1 : x?(b) \\ \ell_2 : c!\langle x\rangle \end{Bmatrix} \qquad R = x \triangleright \begin{Bmatrix} \ell_1 : x!\langle b\rangle \\ \ell_2 : x?(c) \end{Bmatrix}$$

$$Q_1 = x \triangleleft \{\ell_1 : x?(b)\} \qquad Q_2 = x \triangleleft \{\ell_2 : x!\langle c\rangle\}$$

Example



Nil
Nil
Nil
Nil
Com
Com
Com
Com
(νx)(x?(b) | x!⟨b⟩)
(νx)(x?(c) | x!⟨c⟩)
(νx)(x?(b) | x!⟨b⟩)
(νy)(a!⟨y⟩ | a?(y))
Sel
Sel
Sel
Com
(νx)(Q1)
(νx)(Q2)
?
~
(νx)(Q1)
(νx)(νy)(y!⟨a⟩ | y?(a) | x!⟨c⟩ | x?(c))
Bra
Bra
Com
Sel
(νx)(Q)
(νx)(νy)(y!⟨a⟩ | y?(a) | Q1)
(νx)(νy)(y!⟨a⟩ | y?(a) | Q2)
Com
Bra
Bra
(νx)(νy)(y!⟨a⟩ | y?(a) | Q)
(νx)(νy)(y!⟨a⟩ | y?(a) | Q)

Processes as Formulas

What do we need:

- a “linear” approach to resources;

What do we need:

- a "linear" approach to resources;
- parallel operator (commutative, associative);

$$[\![\mathsf{Nil}]\!] = \circ$$

$$[\![P \mid Q]\!] = [\![P]\!] \parr [\![Q]\!]$$

What do we need:

- a "linear" approach to resources;
- parallel operator (commutative, associative);
- choice operators (one with priority over the other).

$$[\![\mathsf{Nil}]\!] = \circ$$

$$[\![P \mid Q]\!] = [\![P]\!] \parr [\![Q]\!]$$

$$[\![x \triangleleft \{\ell : P_\ell\}_{\ell \in L}]\!] = \mathop{\&}_{\ell \in L} ([\![P_\ell]\!]) \qquad [\![x \triangleright \{\ell : P_\ell\}_{\ell \in L}]\!] = \bigoplus_{\ell \in L} ([\![P_\ell]\!])$$

What do we need:

- a "linear" approach to resources;
- parallel operator (commutative, associative);
- choice operators (one with priority over the other).
- prefix operator (non-commutative, non-associative);

$$[\![\mathsf{Nil}]\!] = \circ$$

$$[\![P \mid Q]\!] = [\![P]\!] \parr [\![Q]\!]$$

$$[\![x \triangleleft \{\ell : P_\ell\}_{\ell \in L}]\!] = \bigwith_{\ell \in L} ([\![P_\ell]\!]) \qquad [\![x \triangleright \{\ell : P_\ell\}_{\ell \in L}]\!] = \bigoplus_{\ell \in L} ([\![P_\ell]\!])$$

$$[\![x!\langle y\rangle.P]\!] = \begin{cases} \langle x!y\rangle & \text{if } P = \mathsf{Nil} \\ \langle x!y\rangle \blacktriangleleft [\![P]\!] & \text{if } P \neq \mathsf{Nil} \end{cases} \qquad [\![x?(y).P]\!] = \begin{cases} (x?y) & \text{if } P = \mathsf{Nil} \\ (x?y) \blacktriangleleft [\![P]\!] & \text{if } P \neq \mathsf{Nil} \end{cases}$$

What do we need:

- a "linear" approach to resources;
- parallel operator (commutative, associative);
- choice operators (one with priority over the other).
- prefix operator (non-commutative, non-associative);

$$[\![\mathsf{Nil}]\!] = \circ$$

$$[\![P \mid Q]\!] = [\![P]\!] \parr [\![Q]\!]$$

$$[\![x \triangleleft \{\ell : P_\ell\}_{\ell \in L}]\!] = \bigwith_{\ell \in L} (\langle x!\ell \rangle \blacktriangleleft [\![P_\ell]\!]) \qquad [\![x \triangleright \{\ell : P_\ell\}_{\ell \in L}]\!] = \bigoplus_{\ell \in L} ((x?\ell) \blacktriangleleft [\![P_\ell]\!])$$

$$[\![x!\langle y \rangle.P]\!] = \begin{cases} \langle x!y \rangle & \text{if } P = \mathsf{Nil} \\ \langle x!y \rangle \blacktriangleleft [\![P]\!] & \text{if } P \neq \mathsf{Nil} \end{cases} \qquad [\![x?(y).P]\!] = \begin{cases} (x?y) & \text{if } P = \mathsf{Nil} \\ (x?y) \blacktriangleleft [\![P]\!] & \text{if } P \neq \mathsf{Nil} \end{cases}$$

What do we need:

- a "linear" approach to resources;
- parallel operator (commutative, associative);
- choice operators (one with priority over the other).
- prefix operator (non-commutative, non-associative);
- variable binding for input (existential quantifier);

$$[\![\mathsf{Nil}]\!] = \circ$$

$$[\![P \mid Q]\!] = [\![P]\!] \parr [\![Q]\!] \qquad [\![(\nu x)(P)]\!] = \text{Ꮋ}x.\,[\![P]\!]$$

$$[\![x \triangleleft \{\ell : P_\ell\}_{\ell \in L}]\!] = \bigwith_{\ell \in L} (\langle x!\ell \rangle \blacktriangleleft [\![P_\ell]\!]) \qquad [\![x \triangleright \{\ell : P_\ell\}_{\ell \in L}]\!] = \bigoplus_{\ell \in L} ((x?\ell) \blacktriangleleft [\![P_\ell]\!])$$

$$[\![x!\langle y \rangle.P]\!] = \begin{cases} \langle x!y \rangle & \text{if } P = \mathsf{Nil} \\ \langle x!y \rangle \blacktriangleleft [\![P]\!] & \text{if } P \neq \mathsf{Nil} \end{cases} \qquad [\![x?(y).P]\!] = \begin{cases} \exists y.((x?y)) & \text{if } P = \mathsf{Nil} \\ \exists y.((x?y) \blacktriangleleft [\![P]\!]) & \text{if } P \neq \mathsf{Nil} \end{cases}$$

What do we need:

- a "linear" approach to resources;
- parallel operator (commutative, associative);
- choice operators (one with priority over the other).
- prefix operator (non-commutative, non-associative);
- variable binding for input (existential quantifier);
- restriction operator (nominal quantifier);

$$[\![\mathsf{Nil}]\!] = \circ$$

$$[\![P \mid Q]\!] = [\![P]\!] \parr [\![Q]\!] \qquad [\![(\nu x)(P)]\!] = \text{И} x.\,[\![P]\!]$$

$$[\![x \triangleleft \{\ell : P_\ell\}_{\ell \in L}]\!] = \bigwith_{\ell \in L} (\langle x!\ell\rangle \blacktriangleleft [\![P_\ell]\!]) \qquad [\![x \triangleright \{\ell : P_\ell\}_{\ell \in L}]\!] = \bigoplus_{\ell \in L} ((x?\ell) \blacktriangleleft [\![P_\ell]\!])$$

$$[\![x!\langle y\rangle.P]\!] = \begin{cases} \langle x!y\rangle & \text{if } P = \mathsf{Nil} \\ \langle x!y\rangle \blacktriangleleft [\![P]\!] & \text{if } P \neq \mathsf{Nil} \end{cases} \qquad [\![x?(y).P]\!] = \begin{cases} \exists y.((x?y)) & \text{if } P = \mathsf{Nil} \\ \exists y.((x?y) \blacktriangleleft [\![P]\!]) & \text{if } P \neq \mathsf{Nil} \end{cases}$$

Formulas

$A, B \coloneqq$

$\mid$	$\langle x!y \rangle$	atom
$\mid$	$(x?y)$	atom
$\mid$	$A \parr B$	par
$\mid$	$A \otimes B$	tensor

$$\text{ax}\ \frac{}{\mathcal{S} \vdash \langle x!y \rangle, (x?y)} \qquad \parr\ \frac{\mathcal{S} \vdash \Gamma, A, B}{\mathcal{S} \vdash \Gamma, A \parr B} \qquad \otimes\ \frac{\mathcal{S}_1 \vdash \Gamma, A \quad \mathcal{S}_2 \vdash B, \Delta}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, A \otimes B, \Delta} \qquad \boxed{\text{cut}\ \frac{\mathcal{S}_1 \vdash \Gamma, A \quad \mathcal{S}_2 \vdash A^\perp, \Delta}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, \Delta}}$$

$$\text{ax}\ \frac{}{\mathcal{S} \vdash \langle x!y\rangle, (x?y)} \qquad ⅋\ \frac{\mathcal{S} \vdash \Gamma, A, B}{\mathcal{S} \vdash \Gamma, A ⅋ B} \qquad \otimes\ \frac{\mathcal{S}_1 \vdash \Gamma, A \quad \mathcal{S}_2 \vdash B, \Delta}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, A \otimes B, \Delta} \qquad \boxed{\text{cut}\ \frac{\mathcal{S}_1 \vdash \Gamma, A \quad \mathcal{S}_2 \vdash A^\perp, \Delta}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, \Delta}}$$

Formulas

$A, B \coloneqq$	$\circ$	unit (atom)
$\mid$	$\langle x!y\rangle$	atom
$\mid$	$(x?y)$	atom
$\mid$	$A ⅋ B$	par
$\mid$	$A \otimes B$	tensor

$$\circ\ \frac{}{\mathcal{S} \vdash \circ} \qquad \text{mix}\ \frac{\mathcal{S}_1 \vdash \Gamma \quad \mathcal{S}_2 \vdash \Delta}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, \Delta}$$

Formulas

$A, B \coloneqq$	$\circ$	unit (atom)
$\mid$	$\langle x!y \rangle$	atom
$\mid$	$(x?y)$	atom
$\mid$	$A \parr B$	par
$\mid$	$A \otimes B$	tensor
$\mid$	$A \oplus B$	oplus
$\mid$	$A \,\&\, B$	with

$$\text{ax}\ \frac{}{\mathcal{S} \vdash \langle x!y \rangle, (x?y)} \qquad \parr\ \frac{\mathcal{S} \vdash \Gamma, A, B}{\mathcal{S} \vdash \Gamma, A \parr B} \qquad \otimes\ \frac{\mathcal{S}_1 \vdash \Gamma, A \quad \mathcal{S}_2 \vdash B, \Delta}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, A \otimes B, \Delta} \qquad \boxed{\text{cut}\ \frac{\mathcal{S}_1 \vdash \Gamma, A \quad \mathcal{S}_2 \vdash A^\perp, \Delta}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, \Delta}}$$

$$\oplus\ \frac{\mathcal{S} \vdash \Gamma, A_i}{\mathcal{S} \vdash \Gamma, A_1 \oplus B_2} \qquad \&\ \frac{\mathcal{S} \vdash \Gamma, A \quad \mathcal{S} \vdash \Gamma, B}{\mathcal{S} \vdash \Gamma, A \,\&\, B}$$

$$\circ\ \frac{}{\mathcal{S} \vdash \circ} \qquad \text{mix}\ \frac{\mathcal{S}_1 \vdash \Gamma \quad \mathcal{S}_2 \vdash \Delta}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, \Delta}$$

Formulas

$$A, B \coloneqq \circ \quad \text{unit (atom)}$$
$$\mid \langle x!y \rangle \quad \text{atom}$$
$$\mid (x?y) \quad \text{atom}$$
$$\mid A \parr B \quad \text{par}$$
$$\mid A \otimes B \quad \text{tensor}$$

$$\mid A \oplus B \quad \text{oplus}$$
$$\mid A \,\&\, B \quad \text{with}$$
$$\mid \forall x.A \quad \text{for all}$$
$$\mid \exists x.A \quad \text{exists}$$

$$\text{ax}\ \frac{}{\mathcal{S} \vdash \langle x!y \rangle, (x?y)} \qquad \parr\ \frac{\mathcal{S} \vdash \Gamma, A, B}{\mathcal{S} \vdash \Gamma, A \parr B} \qquad \otimes\ \frac{\mathcal{S}_1 \vdash \Gamma, A \quad \mathcal{S}_2 \vdash B, \Delta}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, A \otimes B, \Delta} \qquad \boxed{\text{cut}\ \frac{\mathcal{S}_1 \vdash \Gamma, A \quad \mathcal{S}_2 \vdash A^\perp, \Delta}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, \Delta}}$$

$$\oplus\ \frac{\mathcal{S} \vdash \Gamma, A_i}{\mathcal{S} \vdash \Gamma, A_1 \oplus B_2} \qquad \&\ \frac{\mathcal{S} \vdash \Gamma, A \quad \mathcal{S} \vdash \Gamma, B}{\mathcal{S} \vdash \Gamma, A \,\&\, B} \qquad \exists\ \frac{\mathcal{S} \vdash \Gamma, A[y/x]}{\mathcal{S} \vdash \Gamma, \exists x.A} \qquad \forall\ \frac{\mathcal{S} \vdash \Gamma, A[y/x]}{\mathcal{S} \vdash \Gamma, \forall x.A}\ \dagger$$

$$\circ\ \frac{}{\mathcal{S} \vdash \circ} \qquad \text{mix}\ \frac{\mathcal{S}_1 \vdash \Gamma \quad \mathcal{S}_2 \vdash \Delta}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, \Delta}$$

Formulas

$A, B \coloneqq$	$\circ$	unit (atom)
|	$\langle x!y \rangle$	atom
|	$(x?y)$	atom
|	$A \parr B$	par
|	$A \otimes B$	tensor
|	$A \triangleleft B$	prec
|	$A \oplus B$	oplus
|	$A \,\&\, B$	with
|	$\forall x.A$	for all
|	$\exists x.A$	exists

$$\text{ax}\,\frac{}{\mathcal{S} \vdash \langle x!y \rangle, (x?y)} \qquad \parr\,\frac{\mathcal{S} \vdash \Gamma, A, B}{\mathcal{S} \vdash \Gamma, A \parr B} \qquad \otimes\,\frac{\mathcal{S}_1 \vdash \Gamma, A \quad \mathcal{S}_2 \vdash B, \Delta}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, A \otimes B, \Delta} \qquad \boxed{\text{cut}\,\frac{\mathcal{S}_1 \vdash \Gamma, A \quad \mathcal{S}_2 \vdash A^\perp, \Delta}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, \Delta}}$$

$$\oplus\,\frac{\mathcal{S} \vdash \Gamma, A_i}{\mathcal{S} \vdash \Gamma, A_1 \oplus B_2} \qquad \&\,\frac{\mathcal{S} \vdash \Gamma, A \quad \mathcal{S} \vdash \Gamma, B}{\mathcal{S} \vdash \Gamma, A \,\&\, B} \qquad \exists\,\frac{\mathcal{S} \vdash \Gamma, A[y/x]}{\mathcal{S} \vdash \Gamma, \exists x.A} \qquad \forall\,\frac{\mathcal{S} \vdash \Gamma, A[y/x]}{\mathcal{S} \vdash \Gamma, \forall x.A}\,\dagger$$

$$\circ\,\frac{}{\mathcal{S} \vdash \circ} \qquad \text{mix}\,\frac{\mathcal{S}_1 \vdash \Gamma \quad \mathcal{S}_2 \vdash \Delta}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, \Delta} \qquad \triangleleft\,\frac{\mathcal{S}_1 \vdash \Gamma, A, C \quad \mathcal{S}_2 \vdash \Delta, B, D}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, \Delta, A \triangleleft B, C \triangleleft D} \qquad \triangleleft_\circ\,\frac{\mathcal{S}_1 \vdash \Gamma, A \quad \mathcal{S}_2 \vdash \Delta, B}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, \Delta, A \triangleleft B}$$

Formulas

$A, B \coloneqq$	$\circ$	unit (atom)
$\mid$	$\langle x!y \rangle$	atom
$\mid$	$(x?y)$	atom
$\mid$	$A \parr B$	par
$\mid$	$A \otimes B$	tensor
$\mid$	$A \blacktriangleleft B$	prec
$\mid$	$A \oplus B$	oplus
$\mid$	$A \mathbin{\&} B$	with
$\mid$	$\forall x.A$	for all
$\mid$	$\exists x.A$	exists
$\mid$	$\text{И}x.A$	new
$\mid$	$\text{Я}x.A$	ya

$$\text{ax}\,\frac{}{\mathcal{S} \vdash \langle x!y \rangle, (x?y)} \qquad \parr\,\frac{\mathcal{S} \vdash \Gamma, A, B}{\mathcal{S} \vdash \Gamma, A \parr B} \qquad \otimes\,\frac{\mathcal{S}_1 \vdash \Gamma, A \quad \mathcal{S}_2 \vdash B, \Delta}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, A \otimes B, \Delta} \qquad \boxed{\text{cut}\,\frac{\mathcal{S}_1 \vdash \Gamma, A \quad \mathcal{S}_2 \vdash A^{\perp}, \Delta}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, \Delta}}$$

$$\oplus\,\frac{\mathcal{S} \vdash \Gamma, A_i}{\mathcal{S} \vdash \Gamma, A_1 \oplus B_2} \qquad \&\,\frac{\mathcal{S} \vdash \Gamma, A \quad \mathcal{S} \vdash \Gamma, B}{\mathcal{S} \vdash \Gamma, A \mathbin{\&} B} \qquad \exists\,\frac{\mathcal{S} \vdash \Gamma, A[y/x]}{\mathcal{S} \vdash \Gamma, \exists x.A} \qquad \forall\,\frac{\mathcal{S} \vdash \Gamma, A[y/x]}{\mathcal{S} \vdash \Gamma, \forall x.A}\,\dagger$$

$$\circ\,\frac{}{\mathcal{S} \vdash \circ} \qquad \text{mix}\,\frac{\mathcal{S}_1 \vdash \Gamma \quad \mathcal{S}_2 \vdash \Delta}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, \Delta} \qquad \blacktriangleleft\,\frac{\mathcal{S}_1 \vdash \Gamma, A, C \quad \mathcal{S}_2 \vdash \Delta, B, D}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, \Delta, A \blacktriangleleft B, C \blacktriangleleft D} \qquad \blacktriangleleft_{\circ}\,\frac{\mathcal{S}_1 \vdash \Gamma, A \quad \mathcal{S}_2 \vdash \Delta, B}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, \Delta, A \blacktriangleleft B}$$

$$\text{И}_{\circ}\,\frac{\mathcal{S} \vdash \Gamma, A[y/x]}{\mathcal{S} \vdash \Gamma, \text{И}x.A}\,\dagger \qquad \text{И}_{\text{load}}\,\frac{\mathcal{S}, x^{\text{И}} \vdash \Gamma, A}{\mathcal{S} \vdash \Gamma, \text{И}x.A}\,\dagger \qquad \text{И}_{\text{pop}}\,\frac{\mathcal{S} \vdash \Gamma, A[y/x]}{\mathcal{S}, y^{\text{И}} \vdash \Gamma, \text{Я}x.A}$$

$$\text{Я}_{\circ}\,\frac{\mathcal{S} \vdash \Gamma, A[y/x]}{\mathcal{S} \vdash \Gamma, \text{Я}x.A}\,\dagger \qquad \text{Я}_{\text{load}}\,\frac{\mathcal{S}, x^{\text{Я}} \vdash \Gamma, A}{\mathcal{S} \vdash \Gamma, \text{Я}x.A}\,\dagger \qquad \text{Я}_{\text{pop}}\,\frac{\mathcal{S} \vdash \Gamma, A[y/x]}{\mathcal{S}, y^{\text{Я}} \vdash \Gamma, \text{И}x.A}$$

Formulas

$A, B \coloneqq$	$\circ$	unit (atom)
$\mid$	$\langle x!y \rangle$	atom
$\mid$	$(x?y)$	atom
$\mid$	$A \parr B$	par
$\mid$	$A \otimes B$	tensor
$\mid$	$A \blacktriangleleft B$	prec
$\mid$	$A \oplus B$	oplus
$\mid$	$A \,\&\, B$	with
$\mid$	$\forall x.A$	for all
$\mid$	$\exists x.A$	exists
$\mid$	$\text{И}x.A$	new
$\mid$	$\text{Я}x.A$	ya

$$\text{ax}\,\frac{}{\mathcal{S} \vdash \langle x!y \rangle, (x?y)} \qquad \parr\,\frac{\mathcal{S} \vdash \Gamma, A, B}{\mathcal{S} \vdash \Gamma, A \parr B} \qquad \otimes\,\frac{\mathcal{S}_1 \vdash \Gamma, A \quad \mathcal{S}_2 \vdash B, \Delta}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, A \otimes B, \Delta} \qquad \boxed{\text{cut}\,\frac{\mathcal{S}_1 \vdash \Gamma, A \quad \mathcal{S}_2 \vdash A^\perp, \Delta}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, \Delta}}$$

$$\oplus\,\frac{\mathcal{S} \vdash \Gamma, A_i}{\mathcal{S} \vdash \Gamma, A_1 \oplus B_2} \qquad \&\,\frac{\mathcal{S} \vdash \Gamma, A \quad \mathcal{S} \vdash \Gamma, B}{\mathcal{S} \vdash \Gamma, A \,\&\, B} \qquad \exists\,\frac{\mathcal{S} \vdash \Gamma, A[y/x]}{\mathcal{S} \vdash \Gamma, \exists x.A} \qquad \forall\,\frac{\mathcal{S} \vdash \Gamma, A[y/x]}{\mathcal{S} \vdash \Gamma, \forall x.A}\,\dagger$$

$$\circ\,\frac{}{\mathcal{S} \vdash \circ} \qquad \text{mix}\,\frac{\mathcal{S}_1 \vdash \Gamma \quad \mathcal{S}_2 \vdash \Delta}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, \Delta} \qquad \blacktriangleleft\,\frac{\mathcal{S}_1 \vdash \Gamma, A, C \quad \mathcal{S}_2 \vdash \Delta, B, D}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, \Delta, A \blacktriangleleft B, C \blacktriangleleft D} \qquad \blacktriangleleft_\circ\,\frac{\mathcal{S}_1 \vdash \Gamma, A \quad \mathcal{S}_2 \vdash \Delta, B}{\mathcal{S}_1, \mathcal{S}_2 \vdash \Gamma, \Delta, A \blacktriangleleft B}$$

$$\text{И}_\circ\,\frac{\mathcal{S} \vdash \Gamma, A[y/x]}{\mathcal{S} \vdash \Gamma, \text{И}x.A}\,\dagger \qquad \text{И}_{\text{load}}\,\frac{\mathcal{S}, x^{\text{И}} \vdash \Gamma, A}{\mathcal{S} \vdash \Gamma, \text{И}x.A}\,\dagger \qquad \text{И}_{\text{pop}}\,\frac{\mathcal{S} \vdash \Gamma, A[y/x]}{\mathcal{S}, y^{\text{И}} \vdash \Gamma, \text{Я}x.A}$$

$$\text{Я}_\circ\,\frac{\mathcal{S} \vdash \Gamma, A[y/x]}{\mathcal{S} \vdash \Gamma, \text{Я}x.A}\,\dagger \qquad \text{Я}_{\text{load}}\,\frac{\mathcal{S}, x^{\text{Я}} \vdash \Gamma, A}{\mathcal{S} \vdash \Gamma, \text{Я}x.A}\,\dagger \qquad \text{Я}_{\text{pop}}\,\frac{\mathcal{S} \vdash \Gamma, A[y/x]}{\mathcal{S}, y^{\text{Я}} \vdash \Gamma, \text{И}x.A}$$

$A \multimap B \coloneqq A^\perp \parr B$

PiL = all above rules except cut

Theorem

Let F be a formula. Then $\vdash_{\mathsf{PiL}\cup\{\mathsf{cut}\}} F \iff \vdash_{\mathsf{PiL}} F$.

Theorem

If $\vdash_{\mathsf{PiL}} A \multimap B$ *and* $\vdash_{\mathsf{PiL}} B \multimap C$, *then* $\vdash_{\mathsf{PiL}} A \multimap C$.

Theorem

The following logical equivalences and implications are derivable in PiL.

$$(A \blacktriangleleft B) \multimap (A \parr B) \qquad (A \parr \circ) \multimapboth (A \blacktriangleleft \circ) \multimapboth (\circ \blacktriangleleft A) \multimapboth (A \otimes \circ) \multimapboth A$$

$$((A \odot B) \odot C) \multimapboth (A \odot (B \odot C)) \qquad (A \odot B) \multimapboth (B \odot A) \quad \text{for all } \odot \in \{\parr, \otimes, \oplus, \&\}$$

$$((A \& C) \parr (B \& C)) \multimap (A \& B) \parr C \qquad \text{И}x.(A \parr B) \multimap (\text{И}x.A) \parr B \quad \text{if } x \notin \mathrm{free}(B)$$

Proofs in PiL as computation trees

Theorem

- *If* $P \Rrightarrow Q$, *then* $[\![Q]\!] \multimap [\![P]\!]$
- *If* $P \rightarrow Q$ *via* Com *or* Sel, *then* $[\![Q]\!] \multimap [\![P]\!]$
- *If* "$P \rightarrow \{Q_\ell\}_{\ell \in L}$" *via* Bra, *then* $(\&_{\ell \in L} [\![Q_\ell]\!]) \multimap [\![P]\!]$;

Theorem

If P *is race-free, then* P *is deadlock-free iff* $\vdash_{\mathsf{PiL}} [\![P]\!]$.

Lemma

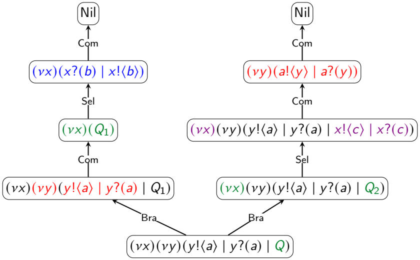
For each computation tree $\mathcal{T}$ of P there is a derivation $\mathcal{D}_{\mathcal{T}}$ with conclusion $\llbracket P \rrbracket$ in PiL.

Idea:

$$\left\llbracket \begin{array}{c} \boxed{(\nu x)(\nu \widetilde{y})\,(P \mid Q[a/b] \mid R)} \\ \uparrow \text{Com} \\ \boxed{(\nu x)(\nu \widetilde{y})\,(x!\langle a\rangle.P \mid x?(b).Q \mid R)} \end{array} \right\rrbracket = \dfrac{\dfrac{\dfrac{}{\vdash \langle x!a\rangle, (x?a)}\,\mathsf{ax} \qquad \dfrac{\dfrac{\vdash \lfloor\!\lfloor (\nu x)(\nu \widetilde{y})\,(P \mid Q[a/b] \mid R) \rfloor\!\rfloor}{\vdash \lfloor\!\lfloor P \rfloor\!\rfloor, \lfloor\!\lfloor Q[a/b] \rfloor\!\rfloor, \lfloor\!\lfloor R \rfloor\!\rfloor}\,= \quad \big\Vert \{\mathbin{\rotatebox[origin=c]{180}{\&}}, \text{И}_\circ, \mathsf{mix}, \circ\}}{\vdash \llbracket P \rrbracket, \llbracket Q[a/b] \rrbracket, \lfloor\!\lfloor R \rfloor\!\rfloor}}{\vdash \langle x!a\rangle \blacktriangleleft \llbracket P \rrbracket, (x?a) \blacktriangleleft \llbracket Q[a/b] \rrbracket, \lfloor\!\lfloor R \rfloor\!\rfloor}\,\blacktriangleleft}{\dfrac{\vdash \langle x!a\rangle \blacktriangleleft \llbracket P \rrbracket, \exists b.\,((x?b) \blacktriangleleft \llbracket Q \rrbracket), \Gamma}{\vdash \lfloor\!\lfloor (\nu x)(\nu \widetilde{y})\,(x!\langle a\rangle.P \mid x?(b).Q \mid R) \rfloor\!\rfloor}\,=}\,\exists$$

$$\left\llbracket \begin{array}{c} \boxed{(\nu x)(\nu \widetilde{y})\,(P_{\ell_k} \mid Q_{\ell_k} \mid R)} \\ \uparrow \text{Sel} \\ \boxed{(\nu x)(\nu \widetilde{y})\,\left(x \triangleleft \{\ell_k : P_{\ell_k}\} \mid x \triangleright \{\ell : Q_\ell\}_{\ell \in L} \mid R\right)} \end{array} \right\rrbracket = \dfrac{\dfrac{\dfrac{}{\vdash \langle x!\ell_k\rangle, (x?\ell_k)}\,\mathsf{ax} \qquad \dfrac{\dfrac{\vdash \lfloor\!\lfloor (\nu x)(\nu \widetilde{y})\,(P_{\ell_k} \mid Q_{\ell_k} \mid R) \rfloor\!\rfloor}{\vdash \lfloor\!\lfloor P_{\ell_k} \rfloor\!\rfloor, \lfloor\!\lfloor Q_{\ell_k} \rfloor\!\rfloor, \lfloor\!\lfloor R \rfloor\!\rfloor}\,= \quad \big\Vert \{\mathbin{\rotatebox[origin=c]{180}{\&}}, \text{И}_\circ, \mathsf{mix}, \circ\}}{\vdash \llbracket P_{\ell_k} \rrbracket, \llbracket Q_{\ell_k} \rrbracket, \lfloor\!\lfloor R \rfloor\!\rfloor}}{\vdash \langle x!\ell_k\rangle \blacktriangleleft \llbracket P_{\ell_k} \rrbracket, (x?\ell_k) \blacktriangleleft \llbracket Q_{\ell_k} \rrbracket, \lfloor\!\lfloor R \rfloor\!\rfloor}\,\blacktriangleleft}{\dfrac{\vdash \langle x!\ell_k\rangle \blacktriangleleft \llbracket P_{\ell_k} \rrbracket, \bigoplus_{\ell \in L} ((x?\ell) \blacktriangleleft \llbracket Q_\ell \rrbracket), \lfloor\!\lfloor R \rfloor\!\rfloor}{\vdash \lfloor\!\lfloor (\nu x)(\nu \widetilde{y})\,\left(x \triangleleft \{\ell_k : P_{\ell_k}\} \mid x \triangleright \{\ell : Q_\ell\}_{\ell \in L} \mid R\right) \rfloor\!\rfloor}\,=}\,\oplus$$

$\square$



=

$$\dfrac{\dfrac{\dfrac{}{\langle y!a\rangle, (y?a)}\,\text{ax}}{\langle y!a\rangle, \exists a.(y?a)}\,\exists \qquad \dfrac{\dfrac{\dfrac{}{\langle x!\ell_1\rangle, (x?\ell_1)}\,\text{ax} \quad \dfrac{\dfrac{}{(x?b)\langle x!b\rangle}\,\text{ax}}{\exists b.(x?b), \langle x!b\rangle}\,\exists}{\langle x!\ell_1\rangle \blacktriangleleft \exists b.(x?b), (x?\ell_1) \blacktriangleleft \langle x!b\rangle}\,\blacktriangleleft}{\langle x!\ell_1\rangle \blacktriangleleft \exists b.(x?b), \begin{pmatrix}(x?\ell_1) \blacktriangleleft \langle x!b\rangle \\ \oplus \\ (x?\ell_2) \blacktriangleleft \exists c.(x?c)\end{pmatrix}}\,\oplus}{\langle y!a\rangle, \exists a.(y?a), \langle x!\ell_1\rangle \blacktriangleleft \exists b.(x?b), \begin{pmatrix}(x?\ell_1) \blacktriangleleft \langle x!b\rangle \\ \oplus \\ (x?\ell_2) \blacktriangleleft \exists c.(x?c)\end{pmatrix}}\,\text{mix}$$

$$\dfrac{\dfrac{\dfrac{}{\langle y!a\rangle, (y?a)}\,\text{ax}}{\langle y!a\rangle, \exists a.(y?a)}\,\exists \qquad \dfrac{\dfrac{\dfrac{}{\langle x!\ell_2\rangle, (x?\ell_2)}\,\text{ax} \quad \dfrac{\dfrac{}{\langle x!c\rangle, (x?c)}\,\text{ax}}{\langle x!c\rangle, \exists c.(x?c)}\,\exists}{\langle x!\ell_2\rangle \blacktriangleleft \langle x!c\rangle, (x?\ell_2) \blacktriangleleft \exists c.(x?c)}\,\blacktriangleleft}{\langle x!\ell_2\rangle \blacktriangleleft \langle x!c\rangle, \begin{pmatrix}(x?\ell_1) \blacktriangleleft \langle x!b\rangle \\ \oplus \\ (x?\ell_2) \blacktriangleleft \exists c.(x?c)\end{pmatrix}}\,\oplus}{\langle y!a\rangle, \exists a.(y?a), \langle x!\ell_2\rangle \blacktriangleleft \langle x!c\rangle, \begin{pmatrix}(x?\ell_1) \blacktriangleleft \langle x!b\rangle \\ \oplus \\ (x?\ell_2) \blacktriangleleft \exists c.(x?c)\end{pmatrix}}\,\text{mix}$$

$$\dfrac{\cdots \qquad \cdots}{\langle y!a\rangle, \exists a.(y?a), \begin{pmatrix}\langle x!\ell_1\rangle \blacktriangleleft \exists b.(x?b) \\ \& \\ \langle x!\ell_2\rangle \blacktriangleleft \langle x!c\rangle\end{pmatrix}, \begin{pmatrix}(x?\ell_1) \blacktriangleleft \langle x!b\rangle \\ \oplus \\ (x?\ell_2) \blacktriangleleft \exists c.(x?c)\end{pmatrix}}\,\&$$

$$\Big\Vert\ \{\text{И}^\circ, ⅋\}$$

$$\text{И}x.\text{И}y.\left(\langle y!a\rangle ⅋ \exists a.(y?a) ⅋ \begin{pmatrix}\langle x!\ell_1\rangle \blacktriangleleft \exists b.(x?b) \\ \& \\ \langle x!\ell_2\rangle \blacktriangleleft \langle x!c\rangle\end{pmatrix} ⅋ \begin{pmatrix}(x?\ell_1) \blacktriangleleft \langle x!b\rangle \\ \oplus \\ (x?\ell_2) \blacktriangleleft \exists c.(x?c)\end{pmatrix}\right)$$

Completeness of Choreographic Programming

Theorem

Let P be deadlock-free, then there is a choreography C such that $P = \mathrm{EPP}(C)$.

Proof sketch

- P be deadlock-free $\Rightarrow$ exists a successful comptuation tree $\mathcal{T}$
- $\mathcal{T}$ successful computation tree $\Rightarrow$ exists $\mathcal{D}_{\mathcal{T}}$ derivation in PiL of $[\![P]\!]$
- using permutations $\Rightarrow$ there is a derivation $\mathcal{D}$ of $[\![P]\!]$ made of blocks as the following [also other shapes]:

$$
\exists\,\dfrac{\blacktriangleleft\,\dfrac{\mathsf{ax}\,\dfrac{}{\vdash (\langle k!x\rangle)_{\mathsf{p}}, ((k?x))_{\mathsf{q}}} \qquad \overset{\mathcal{D}'}{\vdash \Gamma, ([\![S]\!])_{\mathsf{p}}, ([\![S']\!]\,[x/y])_{\mathsf{q}}}}{\vdash \Gamma, (\langle k!x\rangle \blacktriangleleft [\![S]\!])_{\mathsf{p}}, ((k?x) \blacktriangleleft ([\![S']\!]\,[x/y]))_{\mathsf{q}}}}{\vdash \Gamma, (\langle k!x\rangle \blacktriangleleft [\![S]\!])_{\mathsf{p}}, (\exists y.((k?y) \blacktriangleleft [\![S']\!]))_{\mathsf{q}}}
$$

$$
\&\,\dfrac{\left\{\dfrac{\mathsf{ax}\,\dfrac{}{\vdash (\langle k!\ell\rangle)_{\mathsf{p}}, ((k?\ell))_{\mathsf{q}}} \qquad \overset{\mathcal{D}'}{\vdash ([\![S_\ell]\!])_{\mathsf{p}}, ([\![S'_\ell]\!])_{\mathsf{q}}}}{\vdash \Gamma, (\langle k!\ell\rangle \blacktriangleleft [\![S_\ell]\!])_{\mathsf{p}}, ((k?\ell) \blacktriangleleft [\![S'_\ell]\!])_{\mathsf{q}}}\right\}_{\ell\in L}}{\vdash \Gamma, \left(\mathop{\&}_{\ell\in L} (\langle k!\ell\rangle \blacktriangleleft [\![S_\ell]\!])\right)_{\mathsf{p}}, \left(\bigoplus_{\ell\in L'} ((k?\ell) \blacktriangleleft [\![S'_\ell]\!])\right)_{\mathsf{q}}}
$$

Theorem

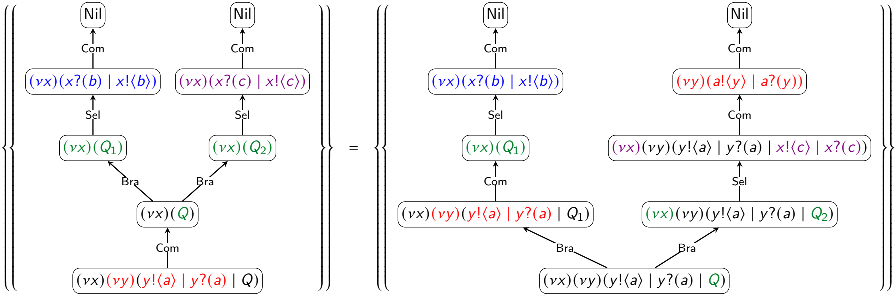
Let P be deadlock-free, then there is a choreography C such that $P = \mathrm{EPP}(C)$.

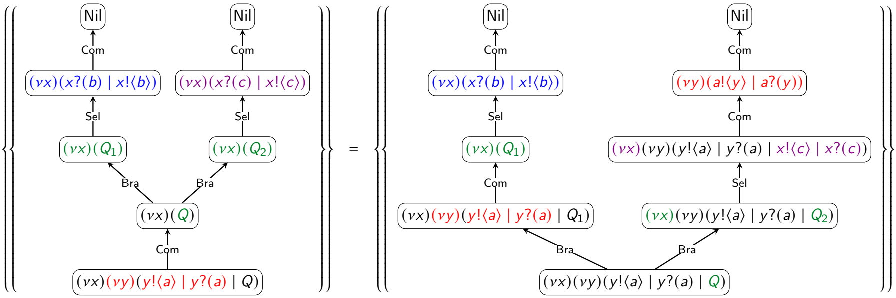
Proof sketch

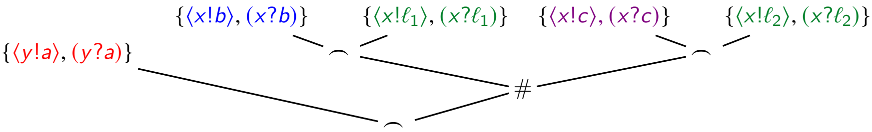
- P be deadlock-free $\Rightarrow$ exists a successful comptuation tree $\mathcal{T}$
- $\mathcal{T}$ successful computation tree $\Rightarrow$ exists $\mathcal{D}_{\mathcal{T}}$ derivation in PiL of $\llbracket P \rrbracket$
- using permutations $\Rightarrow$ there is a derivation $\mathcal{D}$ of $\llbracket P \rrbracket$ made of blocks as the following [also other shapes]:

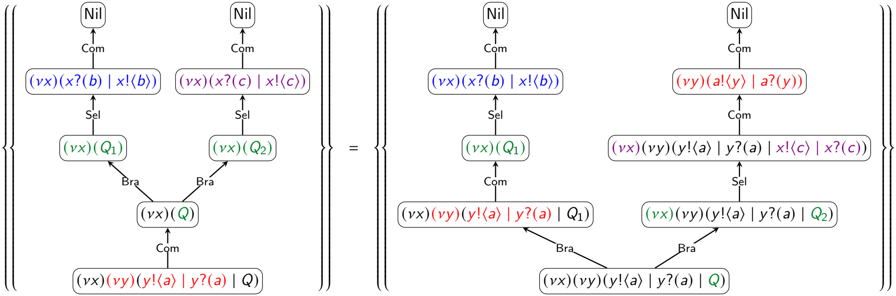
$\exists\,\dfrac{\blacktriangleleft\,\dfrac{\mathsf{ax}\,\dfrac{}{\vdash (\langle k!x\rangle)_{\mathsf{p}}, ((k?x))_{\mathsf{q}}} \qquad \overset{\Vert\, \mathcal{D}'}{\vdash \Gamma, (\llbracket S \rrbracket)_{\mathsf{p}}, (\llbracket S' \rrbracket\,[x/y])_{\mathsf{q}}}}{\vdash \Gamma, (\langle k!x\rangle \blacktriangleleft \llbracket S \rrbracket)_{\mathsf{p}}, ((k?x) \blacktriangleleft (\llbracket S' \rrbracket\,[x/y]))_{\mathsf{q}}}}{\vdash \Gamma, (\langle k!x\rangle \blacktriangleleft \llbracket S \rrbracket)_{\mathsf{p}}, (\exists y.((k?y) \blacktriangleleft \llbracket S' \rrbracket))_{\mathsf{q}}}$	$\&\,\dfrac{\left\{\dfrac{\mathsf{ax}\,\dfrac{}{\vdash (\langle k!\ell\rangle)_{\mathsf{p}}, ((k?\ell))_{\mathsf{q}}} \qquad \overset{\Vert\, \mathcal{D}'}{\vdash (\llbracket S_\ell \rrbracket)_{\mathsf{p}}, (\llbracket S'_\ell \rrbracket)_{\mathsf{q}}}}{\vdash \Gamma, (\langle k!\ell\rangle \blacktriangleleft \llbracket S_\ell \rrbracket)_{\mathsf{p}}, ((k?\ell) \blacktriangleleft \llbracket S'_\ell \rrbracket)_{\mathsf{q}}}\right\}_{\ell \in L}}{\vdash \Gamma, \left(\mathop{\&}_{\ell \in L} (\langle k!\ell\rangle \blacktriangleleft \llbracket S_\ell \rrbracket)\right)_{\mathsf{p}}, \left(\bigoplus_{\ell \in L'} ((k?\ell) \blacktriangleleft \llbracket S'_\ell \rrbracket)\right)_{\mathsf{q}}}$
$[\mathsf{p}.x \rightarrow \mathsf{q}.y];\, C_{\mathcal{D}'}$	$\mathsf{p}.L \rightarrow \mathsf{q}.L' : k \left\{ \begin{array}{l} \ell : C_{\mathcal{D}_\ell} \\ \ell : S'_\ell \end{array} \middle\vert \begin{array}{l} \ell \in L \\ \ell \in L' \setminus L \end{array} \right\}$

Proof nets


Nil
Nil
Com
Com
(νx)(x?(b) | x!⟨b⟩)
(νx)(x?(c) | x!⟨c⟩)
Sel
Sel
(νx)(Q1)
(νx)(Q2)
Bra
Bra
(νx)(Q)
Com
(νx)(νy)(y!⟨a⟩ | y?(a) | Q)
=
Nil
Nil
Com
Com
(νx)(x?(b) | x!⟨b⟩)
(νy)(a!⟨y⟩ | a?(y))
Sel
Com
(νx)(Q1)
(νx)(νy)(y!⟨a⟩ | y?(a) | x!⟨c⟩ | x?(c))
Com
Sel
(νx)(νy)(y!⟨a⟩ | y?(a) | Q1)
(νx)(νy)(y!⟨a⟩ | y?(a) | Q2)
Bra
Bra
(νx)(νy)(y!⟨a⟩ | y?(a) | Q)


Nil
Com
(νx)(x?(b) | x!⟨b⟩)
Sel
(νx)(Q1)
Bra
(νx)(Q)
Com
(νx)(νy)(y!⟨a⟩ | y?(a) | Q)
Nil
Com
(νx)(x?(c) | x!⟨c⟩)
Sel
(νx)(Q2)
Bra
=
Nil
Com
(νx)(x?(b) | x!⟨b⟩)
Sel
(νx)(Q1)
Com
(νx)(νy)(y!⟨a⟩ | y?(a) | Q1)
Bra
(νx)(νy)(y!⟨a⟩ | y?(a) | Q)
Nil
Com
(νy)(a!⟨y⟩ | a?(y))
Com
(νx)(νy)(y!⟨a⟩ | y?(a) | x!⟨c⟩ | x?(c))
Sel
(νx)(νy)(y!⟨a⟩ | y?(a) | Q2)
Bra


{⟨y!a⟩, (y?a)}
{⟨x!b⟩, (x?b)}
{⟨x!ℓ1⟩, (x?ℓ1)}
{⟨x!c⟩, (x?c)}
{⟨x!ℓ2⟩, (x?ℓ2)}
⌢
#
⌢
⌢



$$\left\{ \begin{array}{l} \{\{\langle y!a\rangle, (y?a)\}, \{\langle x!b\rangle, (x?b)\}, \{\langle x!\ell_1\rangle, (x?\ell_1)\}\}\ , \\ \{\{\langle y!a\rangle, (y?a)\}, \{\langle x!c\rangle, (x?c)\}, \{\langle x!\ell_2\rangle, (x?\ell_2)\}\} \end{array} \right\}$$

Conclusion and Future work

Main results:

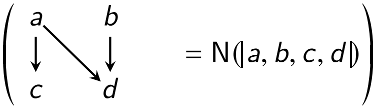
- PiL + cut-free sequent calculus
- Computation trees of $P \rightsquigarrow$ Derivations in PiL of $[\![P]\!]$;
- Deadlock-freedom P (for race-free) = provability of $[\![P]\!]$ in PiL;
- Choreographic extraction using PiL (choreographies-as-proofs);
- Completeness result for choreographies wrt deadlock-freedom;
- Conflict nets and slice nets for PiL;
- Correspondence between computation trees (modulo) and proof nets.

What next?

- *Recursion* using fixpoints operators

Replication	Iteration	recursion
$\dfrac{\mathcal{S} \vdash \Gamma, A, ?A}{\mathcal{S} \vdash \Gamma, ?A}\ ?\mathrm{b}$	$\dfrac{\mathcal{S} \vdash \Gamma, A \blacktriangleleft \text{⁊}A}{\mathcal{S} \vdash \Gamma, \text{⁊}A}\ \text{⁊}\mathrm{b}$	$\dfrac{\mathcal{S} \vdash \Gamma, P(\mu A)}{\mathcal{S} \vdash \Gamma, \mu A.P(A)}\ \mu$

- *Test preorders* as notions of orthogonality for proof nets;
- Asynchronous π-calculus: use message buffers (similar to Concurrent Constraint Programming)
  Note: graphical connectives may be needed to model non-SP behaviors



Thanks

Thanks

Questions?