

A concurrent model for linear logic

Emmanuel Beffara

PPS, Université Paris VII & CNRS

MFPS – May 20, 2005

Linear logic vs concurrency

Interactive interpretations of linear logic:

- ▶ interpretations in CCS (Abramsky)
- ▶ proof nets as processes (Bellin-Scott)
- ▶ game models, ludics

Logics for concurrency:

- ▶ modal logics (Hennessy-Milner, for CCS and π)
- ▶ spatial logics (Caires-Cardelli)
- ▶ relevant logic, bunched implication, etc.

Plan

1. Realizability in the π -calculus.
2. Observation and orthogonality.
3. Logical connectives.
4. Interpretation of classical systems.

Realizability and phase semantics

Traditional realizability interprets intuitionistic logic with

- ▶ a combinatory algebra M (i.e. a set with application)
- ▶ truth values = parts of M
- ▶ $A \rightarrow B = \{x \mid \forall y \in A, xy \in B\}$
- ▶ realize basic laws with elementary combinators (S, K, I, \dots)
- ▶ traditionally, $\perp = \emptyset$ and $\neg A$ is always \top or \perp

The crucial novelty in phase semantics is the meaning of \perp :

- ▶ \perp is an arbitrary subset of M
- ▶ negation is $A^\perp = A \multimap \perp = \{x \mid \forall y \in A, xy \in \perp\}$
- ▶ double-negation is a closure operator

Yet another π -calculus

We work with a kind of π I calculus with routers:

Actions:	$\alpha := \mathbf{u}^\varepsilon(\vec{x})$	with $\varepsilon \in \{\downarrow, \uparrow\}$
Processes:	$P, Q := \alpha.P \quad !\alpha.P$	actions
	$1 \quad (P \mid Q) \quad (\mathbf{v}x)P$	composition
	$\mathbf{u} \rightarrow \mathbf{v}$	routers

Where $\mathbf{u} \rightarrow \mathbf{v}$ can forward messages from \mathbf{u} to \mathbf{v} .

$$\bar{\mathbf{u}}(x).P \mid \mathbf{u} \rightarrow \mathbf{v} \mid \mathbf{v}(x).Q \longrightarrow (\mathbf{v}x)(P \mid Q) \mid \mathbf{u} \rightarrow \mathbf{v}$$

$\mathbf{u} \rightarrow \mathbf{v} \mid \mathbf{v} \rightarrow \mathbf{u}$ is an explicit fusion of the names \mathbf{u} and \mathbf{v} .

Interfaces

Channel types impose arities and polarities:

Channel types: $s := \varepsilon(s_1 \dots s_n) \mid \dots$

Interfaces: $I := x_1 : s_1, \dots, x_n : s_n$

The main typing rules are:

$$\frac{P :: I, u : \varepsilon(s_1 \dots s_n), \{x_i : s_i\}}{u^\varepsilon(x_1 \dots x_n).P :: I, u : \varepsilon(s_1 \dots s_n)} \qquad \frac{P :: I, x_i : s, x_o : \bar{s}}{(\nu x)P[x/x_i, x_o] :: I}$$

All reductions happen on private channels.

Composition of processes

Interfaces provide a composition structure:

- ▶ For I and J disjoint interfaces, define $I \rightarrow J = \bar{I} \cup J$.
- ▶ Define composition as

$$(P :: I \rightarrow J) \cdot (Q :: J \rightarrow K) = (\nu \vec{x})(P \mid Q) :: I \rightarrow K$$

with \vec{x} is the domain of J .

- ▶ Associative (up to structural congruence),
routers provide delocated identities.

Orthogonality

Choose \perp as a set of closed processes and deduce a notion of test between processes of opposite interfaces:

termination: $P \perp Q$ if $P \mid Q$ cannot diverge

must-testing: $P \perp Q$ if any maximal run of $P \mid Q$ reaches $\mathbf{0}$,
where $\mathbf{0}$ is a set of “accepting” states.

and so on...

$(\cdot)^{\perp\perp}$ is a closure operator,

closed sets form a complete lattice of **behaviours**.

Constants

For any interface I , define

$$\begin{aligned} \mathbf{0}_I &= (\emptyset : I)^{\perp\perp} & \top_I &= \text{all processes with interface } I \\ \mathbf{1}_I &= (\{1\} : I)^{\perp\perp} & \perp_I &= \{P \mid P \in \perp, P :: I\} \end{aligned}$$

For termination:

$$\begin{array}{ccc} \begin{array}{c} \top_I \\ \perp_I \\ \diagdown \quad \diagup \\ \dots \\ \diagup \quad \diagdown \\ \mathbf{1}_I \\ \mathbf{0}_I \end{array} & \Longrightarrow & \begin{array}{c} \top_\emptyset \\ | \\ \mathbf{1}_\emptyset = \perp_\emptyset \\ | \\ \mathbf{0}_\emptyset \end{array} \end{array}$$

Constants

For any interface I , define

$$\begin{aligned} \mathbf{0}_I &= (\emptyset : I)^{\perp\perp} & \top_I &= \text{all processes with interface } I \\ \mathbf{1}_I &= (\{1\} : I)^{\perp\perp} & \perp_I &= \{P \mid P \in \perp, P :: I\} \end{aligned}$$

For must-testing:

$$\begin{array}{ccc} & \top_I & \\ & \mathbf{1}_I \vee \perp_I & \\ / & & \backslash \\ & \dots & \\ \backslash & & / \\ & \mathbf{1}_I \wedge \perp_I & \\ & \mathbf{0}_I & \end{array} \quad \Longrightarrow \quad \begin{array}{c} \top_{\emptyset} \\ | \\ \mathbf{1}_{\emptyset} \\ | \\ \perp_{\emptyset} \\ | \\ \mathbf{0}_{\emptyset} \end{array}$$

Space

Interfaces yield a composition of behaviours:

- Composition of behaviours:

for I, J, K disjoint and $\mathcal{A} : I \rightarrow J, \mathcal{B} : J \rightarrow K,$

$$\mathcal{A} \cdot \mathcal{B} = \{ (\nu \vec{x})(P \mid Q) \mid P \in \mathcal{A}, Q \in \mathcal{B} \}^{\perp\perp} : I \rightarrow K$$

Characterizes orthogonality: $\mathcal{A} \perp \mathcal{B}$ iff $\mathcal{A} \cdot \mathcal{B} \subseteq \perp_{\emptyset}$

- Spatial connectives (for $\mathcal{A} : I, \mathcal{B} : J$ with I and J disjoint):

$$\mathcal{A} \otimes \mathcal{B} = \mathcal{A} \cdot \mathcal{B}$$

$$\mathcal{A} \wp \mathcal{B} = (\mathcal{A}^{\perp} \otimes \mathcal{B}^{\perp})^{\perp}$$

$$\mathcal{A} \multimap \mathcal{B} = \mathcal{A}^{\perp} \wp \mathcal{B}$$

Time

We define modalities à la Hennessy-Milner:

- ▶ For \mathcal{A} with interface $\mathbf{u} : \varepsilon(\vec{s}), \{x_i : s_i\}$,

$$P \in [\mathbf{u}^\varepsilon(\vec{x})]\mathcal{A} \quad \text{if} \quad \begin{cases} P \text{ has interface } \mathbf{u} : \varepsilon(s_1 \dots s_n) \\ P \text{ does not diverge} \\ P \xrightarrow{\tau} \xrightarrow{\mathbf{u}^\varepsilon(\vec{x})} Q \text{ implies } Q \in \mathcal{A} \end{cases}$$

- ▶ Modalities are self-dual:

$$([\alpha]\mathcal{A})^\perp = [\bar{\alpha}]\mathcal{A}^\perp$$

The core type system

Axiom and cut:

$$\frac{}{u \rightarrow v \vdash u : \downarrow A^\perp, v : \uparrow A} \quad \frac{P \vdash \Gamma, \vec{x} : A \quad Q \vdash \vec{x} : A^\perp, \Delta}{(\nu \vec{x})(P \mid Q) \vdash \Gamma, \Delta}$$

Multiplicatives:

$$\frac{P \vdash \Gamma, \vec{x} : A \quad Q \vdash \vec{y} : B, \Delta}{P \mid Q \vdash \Gamma, \vec{x}\vec{y} : A \otimes B, \Delta} \quad \frac{P \vdash \Gamma, \vec{x} : A, \vec{y} : B}{P \vdash \Gamma, \vec{x}\vec{y} : A \wp B}$$

Actions:

$$\frac{P \vdash \Gamma, \vec{x} : A}{u(\vec{x}).P \vdash \Gamma, u : \downarrow A} \quad \frac{P \vdash \Gamma, \vec{x} : A}{\bar{u}(\vec{x}).P \vdash \Gamma, u : \uparrow A}$$

The core type system

Axiom and cut:

$$\frac{}{u \rightarrow v \vdash u : \downarrow A^\perp, v : \uparrow A} \quad \frac{P \vdash \Gamma, \vec{x} : A \quad Q \vdash \vec{x} : A^\perp, \Delta}{(\nu \vec{x})(P \mid Q) \vdash \Gamma, \Delta}$$

Multiplicatives:

$$\frac{P \vdash \Gamma, \vec{x} : A \quad Q \vdash \vec{y} : B, \Delta}{P \mid Q \vdash \Gamma, \vec{x}\vec{y} : A \otimes B, \Delta} \quad \frac{P \vdash \Gamma, \vec{x} : A, \vec{y} : B}{P \vdash \Gamma, \vec{x}\vec{y} : A \wp B}$$

Actions:

$$\frac{P \vdash \Gamma, \vec{x} : A}{u(\vec{x}).P \vdash \Gamma, u : \downarrow A} \quad \frac{P \vdash \Gamma, \vec{x} : A}{\bar{u}(\vec{x}).P \vdash \Gamma, u : \uparrow A}$$

A fixed point

A guarded replication $! \alpha . P$ is a generator of independent copies of P guarded by α . Define this:

$$\mathcal{E}_{\alpha, \mathcal{A}}(\mathbf{X}) = [\alpha](\mathbf{X} \otimes \mathcal{A}) \quad \text{and} \quad [! \alpha] \mathcal{A} = \bigcap_{k \in \mathbb{N}} \mathcal{E}_{\alpha, \mathcal{A}}^k(\top)$$

$$[? \alpha] \mathcal{A} = ([! \bar{\alpha}](\mathcal{A}^\perp))^\perp$$

One checks that this satisfies

contraction: $\mathbf{u} \rightarrow \mathbf{w} \mid \mathbf{v} \rightarrow \mathbf{w} \in [! \mathbf{w}(\vec{x})] \mathcal{A} \multimap [! \mathbf{u}(\vec{x})] \mathcal{A} \otimes [! \mathbf{v}(\vec{x})] \mathcal{A}$

co-contraction: $\mathbf{w} \rightarrow \mathbf{u} \mid \mathbf{w} \rightarrow \mathbf{v} \in [! \mathbf{u}(\vec{x})] \mathcal{A} \otimes [! \mathbf{v}(\vec{x})] \mathcal{A} \multimap [! \mathbf{w}(\vec{x})] \mathcal{A}$

Exponential rules

Replicated actions:

$$\frac{P \vdash ?\Gamma, \vec{x} : A}{!u(\vec{x}).P \vdash ?\Gamma, u : !A} \qquad \frac{P \vdash \Gamma, u : \uparrow A}{P \vdash \Gamma, u : ?A}$$

$$\frac{P \vdash \Gamma, u : ?A, v : ?A}{P[w/u, v] \vdash \Gamma, w : ?A} \qquad \frac{P \vdash \Gamma}{P \vdash \Gamma, u : ?A}$$

Exponential axiom:

$$\frac{}{u \rightarrow v \vdash u : !A^\perp, v : ?A}$$

Optional rules

The following more “concurrent” rules hold:

- ▶ Mix:

$$\frac{P \vdash \Gamma \quad Q \vdash \Delta}{P \mid Q \vdash \Gamma, \Delta}$$

- ▶ Non-deterministic choice:

$$\frac{P \vdash \Gamma \quad Q \vdash \Gamma}{P \oplus Q \vdash \Gamma}$$

- ▶ Co-contraction:

$$\frac{P \vdash \Gamma, u : !A \quad Q \vdash \Delta, u : !A}{P \mid Q \vdash \Gamma, \Delta, u : !A}$$

Normalisation

Theorem

Typed processes cannot diverge.

Theorem

*Typing is preserved by reduction
(up to structural congruence).*

Classical systems

A modal translations of intuitionistic and classical logic is:

- ▶ A pair of modalities,

$$\mu, \nu \in \{\downarrow, \uparrow, !, ?\}^* \Rightarrow \text{interaction protocols}$$

E.g. $P \vdash \vec{x} : A$ implies $u(\nu).!v(w).\bar{w}(\vec{x}).P \vdash u : \downarrow!A$.

- ▶ A translation of formulas,

$$(A \rightarrow B)^t = \mu A^t \multimap \nu B^t$$

- ▶ A translation of inference rules.

$$\Rightarrow \text{translation of } \lambda\mu\text{-terms}$$

The translations

Call by name, intuitionistic: $(A \rightarrow B)^{\text{in}} = !\downarrow A^{\text{in}} \multimap \downarrow B^{\text{in}}$

$$\llbracket \mu\alpha[\beta]t \rrbracket_{\gamma}^{\text{in}} = \llbracket t \rrbracket_{\beta}[\gamma/\alpha]$$

$$\llbracket x \rrbracket_{\alpha}^{\text{in}} = \bar{x}\langle\alpha\rangle$$

$$\llbracket \lambda x.t \rrbracket_{\alpha}^{\text{in}} = \alpha(x\beta)\llbracket t \rrbracket_{\beta}^{\text{in}}$$

$$\llbracket tu \rrbracket_{\alpha}^{\text{in}} = (\nu\beta x)(\llbracket t \rrbracket_{\beta}^{\text{in}} \mid !x(\gamma).\llbracket u \rrbracket_{\gamma}^{\text{in}} \mid \bar{\beta}\langle x\alpha\rangle)$$

Milner/Sangiorgi translation

The translations

Call by name, classical:

$$(A \rightarrow B)^n = !?A^n \multimap ?B^n$$

$$\llbracket \mu\alpha[\beta]t \rrbracket_\gamma^n = \llbracket t \rrbracket_\beta[\gamma/\alpha]$$

$$\llbracket x \rrbracket_\alpha^n = \bar{x}\langle\alpha\rangle$$

$$\llbracket \lambda x.t \rrbracket_\alpha^n = \bar{\alpha}(x\beta)\llbracket t \rrbracket_\beta^n$$

$$\llbracket tu \rrbracket_\alpha^n = (\mathbf{v}\beta)(\llbracket t \rrbracket_\beta^n \mid !\beta(xy).(!x(\gamma).\llbracket u \rrbracket_\gamma^n \mid y\rightarrow\alpha))$$

(any reference ?)

The translations

Call by value, intuitionistic: $(A \rightarrow B)^{iv} = !A^{iv} \multimap !B^{iv}$

$$\llbracket \mu\alpha[\beta]t \rrbracket_{\gamma}^{iv} = \llbracket t \rrbracket_{\beta}[\gamma/\alpha]$$

$$\llbracket x \rrbracket_{\alpha}^{iv} = \alpha \rightarrow x$$

$$\llbracket \lambda x.t \rrbracket_{\alpha}^{iv} = !\alpha(x\beta). \llbracket t \rrbracket_{\beta}^{iv}$$

$$\llbracket tu \rrbracket_{\alpha}^{iv} = (\nu\beta\gamma)(\llbracket t \rrbracket_{\beta}^{iv} \mid \llbracket u \rrbracket_{\gamma}^{iv} \mid \bar{\beta}\langle\gamma\alpha\rangle)$$

(any reference ?)

The translations

Call by value, classical:

$$(A \rightarrow B)^v = !A^v \multimap ?!B^v$$

$$\llbracket \mu\alpha[\beta]t \rrbracket_\gamma^v = \llbracket t \rrbracket_\beta[\gamma/\alpha]$$

$$\llbracket x \rrbracket_\alpha^v = \bar{\alpha}\langle x \rangle$$

$$\llbracket \lambda x.t \rrbracket_\alpha^v = \bar{\alpha}(y)!y(x\beta).\llbracket t \rrbracket_\beta^v$$

$$\llbracket tu \rrbracket_\alpha^v = (\mathbf{v}\gamma)\left(!\gamma(x).\mathbf{v}\beta\right)\left(\llbracket t \rrbracket_\beta^v \mid !\beta(w).\bar{w}\langle x\alpha \rangle\right) \mid \llbracket u \rrbracket_\gamma^v$$

Honda/Yoshida/Berger translation

So what ?

What we have:

- ▶ a meaningful concurrent model of LL + shifts
- ▶ a Curry-Howard style connection with some π -calculus
- ▶ a unifying framework for translations of classical logic

What works too:

- ▶ additive connectives = guarded choice
- ▶ new concurrent combinators for λ and $\lambda\mu$

What next ?

Comparisons that should be clarified:

- ▶ with spatial and modal logics
- ▶ with other interactive models (games, ludics, etc)

Some possible directions:

- ▶ describe concurrent inhabitants of known types (new rules)
- ▶ extend typing to catch more concurrency (new connectives)
- ▶ develop logic-based denotational semantics for processes (geometry of interaction?)

Thank you.