L-Nets, parallel strategies and proof-nets

Claudia Faggian

University of Padova (Italy) & PPS–Paris7 (France)

< > - +

L-Nets

Work in collaboration with
F. Maurel (LICS 2005) and P.-L. Curien (CSL 2005)

The context

- **Ludics** (Girard): an abstract game model of *sequential* interaction.
  (but maybe bridge with concurrency?)
- **Game Semantics,** Several proposal towards strategies where sequentiality is relaxed:
  Abramsky and Melliès, Hyland, Schalk, Melliès, Mimram, McCusker and Wall . . .

L-Nets

Work in collaboration with
F. Maurel (LICS 2005) and P.-L. Curien (CSL 2005)

The context

- **Ludics** (Girard): an abstract game model of *sequential* interaction.
  (but maybe bridge with concurrency?)
- **Game Semantics,** Several proposal towards strategies where sequentiality is relaxed:
  Abramsky and Melliès, Hyland, Schalk, Melliès, Mimram, McCusker and Wall . . .

Intuitions underlying L-nets:

MALL proof-nets

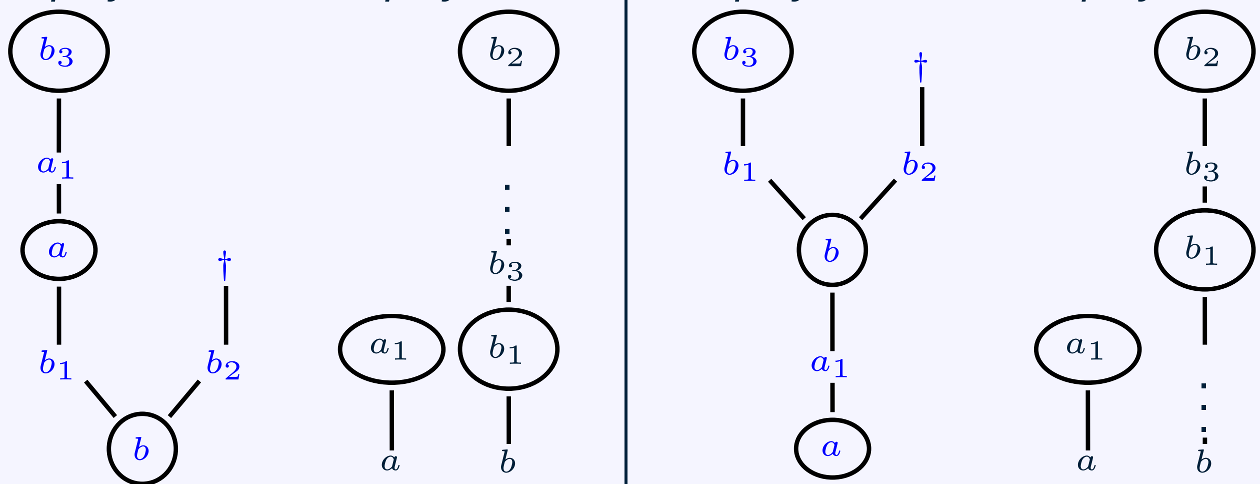
Event Structures (Winskel)

Multi-focalization (Andreoli)

Sketch: strategies and interaction (views presentation)

Strategies:

P-player: O-player: | P-player: O-player:



Interaction (plays)

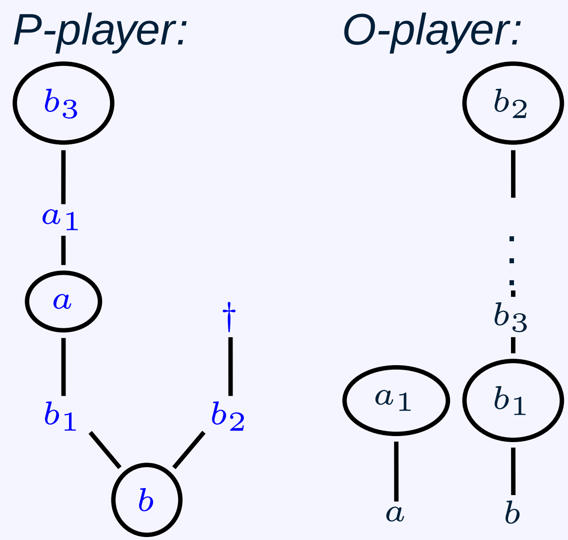
$bb_1aa_1b_3\ldots b_2\dagger$

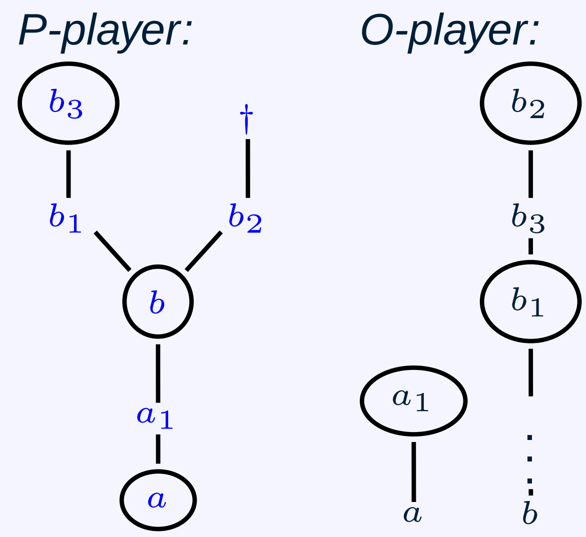
$aa_1b\ldots b_1b_3b_2\dagger$

< > - +

Sketch: strategies and interaction (views presentation)

Strategies:





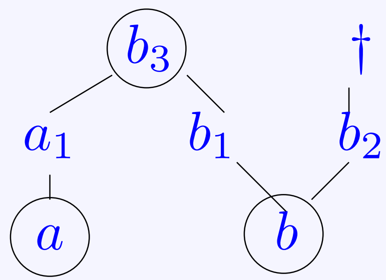
Interaction (plays)

$bb_1aa_1b_3\ldots b_2\dagger$

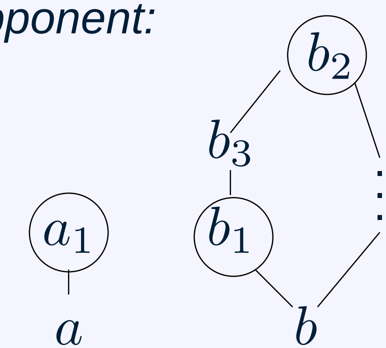
$aa_1b\ldots b_1b_3b_2\dagger$

Strategies :

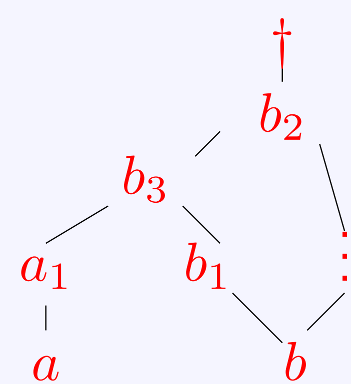
Player:



Opponent:



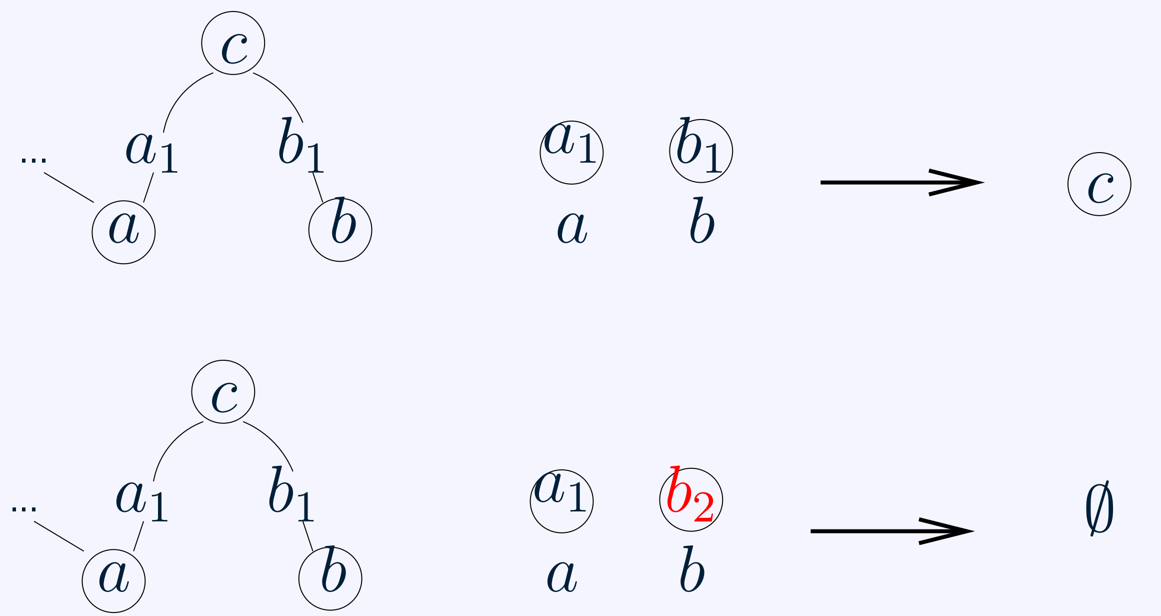
Interaction :



< > - +

Interaction (computation)

Computation accesses an action c only if all actions below c are reached.

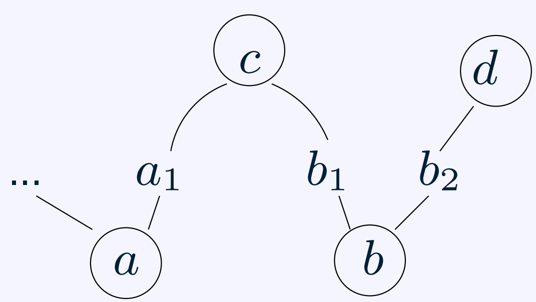


L-nets

a **bipartite D.A.G.** (... with conditions)

- *Nodes:* actions *(cluster of operations which can be performed at the same time)*

- *Edges:* *dependency* / *enabling* / *scheduling* between actions



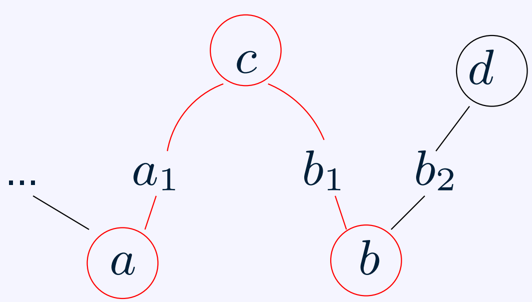
a, b can be performed in parallel, or scheduled in any order, as long as before c.
c: sync. point

< > - +

L-nets

a **bipartite D.A.G.** (... with conditions)

- *Nodes:* actions *(cluster of operations which can be performed at the same time)*
- *Edges:* *dependency* / *enabling* / *scheduling* between actions



a, b can be performed in parallel, or scheduled in any order, as long as before c.
c: sync. point

view of c: $\{k', k' \leq c\}$

Overall picture

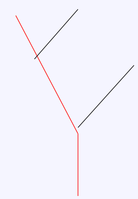
Have an arena: a forest of actions (parent relation)

TREE STRATEGIES

set of views

view: sequence of actions

play: total order



abstract

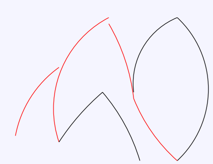
sequent calculus

PARALLEL STRATEGIES

set of p.o. views

p.o.view of k: $\{k', k' \leq k\}$

play: partial order



abstract

proof-net

< > - +

L-Arena

- an **interface** (the addresses)
- a finite **forest of actions** (with a parent relation $a \vdash b$).

The **interface** *of the arena provides the names on which a strategy on that arena can communicate with the rest of the world.*

The **forest** *on a set of actions with the polarity induced by the interface*

- (σ, I) is initial iff the address σ belongs to the interface.
- $a \vdash b$ iff a generates the address of b.

Simply think of the addresses as a precise way to deal with different copies of the same arena (a matter usually dealt with using indices).

L-nets as dag's

directed acyclic bipartite graph on a set of nodes labelled by polarized **actions of the arena.**

- *Justification.* Each k is either initial or there exists a preceding c (the justifier) such that $c \vdash k$.
- *Innocence (view)* If k is negative, and c its (immediate) predecessor, then $c \vdash k$.
- *Lin.* All the addresses in $k^{\downarrow}$ are distinct.
- *Positivity.* If a is maximal (leaf), then a is positive.
- *Additive.*

L-nets as set of views.

- **View $\mathfrak{c}$:** a partial order on polarized actions, such that: has a top element, Alternation, Justification and Innocence.
- **Strategy:** a set of views Δ closed under restriction which satisfies Positivity and Additive.

< > - +

L-nets as set of views.

- **View $\mathfrak{c}$:** a partial order on polarized actions, such that: has a top element, Alternation, Justification and Innocence.
- **Strategy:** a set of views Δ closed under restriction which satisfies Positivity and Additive.

$$\textit{Views}(\mathfrak{D}) = \{\ulcorner n \urcorner : \ n \text{ is a node of } \mathfrak{D}\}$$

conversely

$$\textit{Graph}(\Delta) \textit{is an L-net}$$

Overall picture

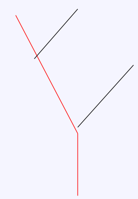
Have an arena: a forest of actions (parent relation)

SEQUENTIAL STRATEGIES

set of views

view: sequence of actions

play: total order



abstract

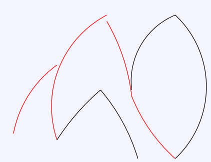
sequent calculus

PARALLEL STRATEGIES

set of p.o. views

p.o.view of k: $\{k', k' \leq k\}$

play: partial order



abstract

proof-net

< > - +

Transfer of technology from proof-nets to Games

(with Curien)

deseq :	Π	$\longmapsto$	*deseq* (Π)
	(sequential)		*(parallel)*
seq :	$\mathfrak{R}$	$\longmapsto$	$\{\Pi_i\}$
	(parallel)		*(sequential)*

in such a way that there is no loss of information: all dependency (sequentialization) which is taken away, can be (non-det.) restored:

Thm. $\Pi \in$ *seq* (*deseq* (Π))

< > - +

Why parallel strategies?

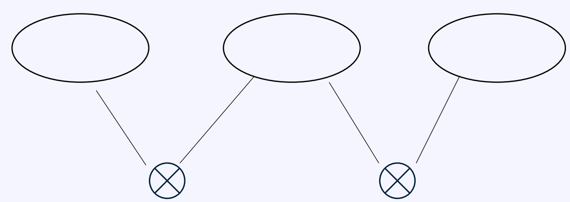
- quest for a more parallel form of interaction in Games
- approach to equational theory of sequential strategies

< > - +

Which equational theory?

“Sequential” strategies are too close to the sequent calculus syntax. Parallel strategies may play a role similar to that of proof-nets w.r.t. sequent calculus.

$$\dfrac{\dfrac{\vdash a, a^{\perp} \quad \vdash b, b^{\perp}}{\vdash a^{\perp} \otimes b, a, b^{\perp}}\,(a^{\perp} \otimes b) \qquad \vdash c, c^{\perp}}{\vdash a^{\perp} \otimes b, b^{\perp} \otimes c, \ldots}\,(b^{\perp} \otimes c) \qquad\qquad \dfrac{\vdash a, a^{\perp} \qquad \dfrac{\vdash b, b^{\perp} \quad \vdash c, c^{\perp}}{\vdash b, c^{\perp}, b^{\perp} \otimes c}\,(b^{\perp} \otimes c)}{\vdash a^{\perp} \otimes b, b^{\perp} \otimes c, \ldots}\,(a^{\perp} \otimes b)$$



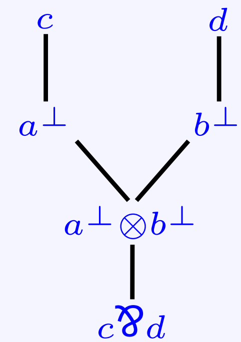
Different permutations correspond to different scheduling (different sequentialization)

Tree strategies are abstract sequent calculus

$$
\dfrac{
  \dfrac{
    \dfrac{
      \dfrac{\ldots}{\vdash \textcolor{black}{a_0, c}}\;\textcolor{blue}{c}
    }{\vdash a^{\perp}, c}\;\textcolor{blue}{a^{\perp}}
    \qquad
    \dfrac{
      \dfrac{\ldots}{\vdash b_0, d}\;\textcolor{blue}{d}
    }{\vdash b^{\perp}, d}\;\textcolor{blue}{b^{\perp}}
  }{\vdash c, d, a^{\perp} \otimes b^{\perp}}\;\textcolor{blue}{a^{\perp} \otimes b^{\perp}}
}{\vdash c \parr d, a^{\perp} \otimes b^{\perp}}\;\textcolor{blue}{c \parr d}
$$

Tree strategies are abstract sequent calculus

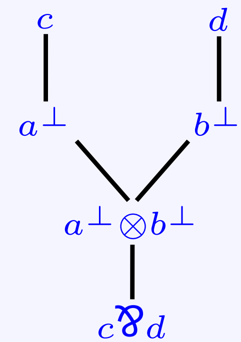
$$\dfrac{\dfrac{\dfrac{\dfrac{\dots}{\vdash a_0, c}\ c}{\vdash a^\perp, c}\ a^\perp \qquad \dfrac{\dfrac{\dots}{\vdash b_0, d}\ d}{\vdash b^\perp, d}\ b^\perp}{\vdash c, d, a^\perp \otimes b^\perp}\ a^\perp \otimes b^\perp}{\vdash c \parr d, a^\perp \otimes b^\perp}\ c \parr d$$



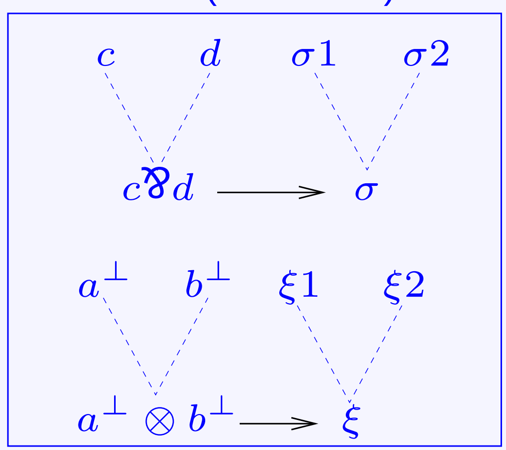
< > - +

Tree strategies are abstract sequent calculus

$$
\dfrac{\dfrac{\dfrac{\dfrac{\ldots}{\vdash a_0, c}\; c}{\vdash a^\perp, c}\; a^\perp \qquad \dfrac{\dfrac{\ldots}{\vdash b_0, d}\; d}{\vdash b^\perp, d}\; b^\perp}{\vdash c, d, a^\perp \otimes b^\perp}\; a^\perp \otimes b^\perp}{\vdash c \parr d, a^\perp \otimes b^\perp}\; c \parr d
$$

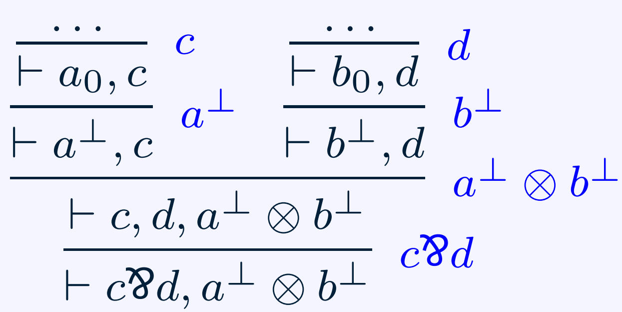


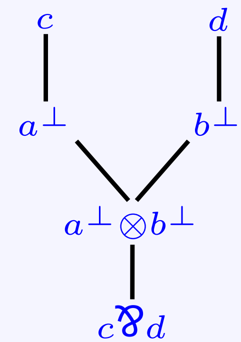
Arena (addresses)

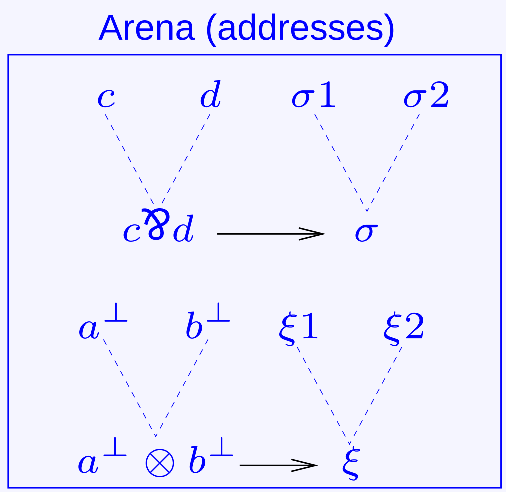


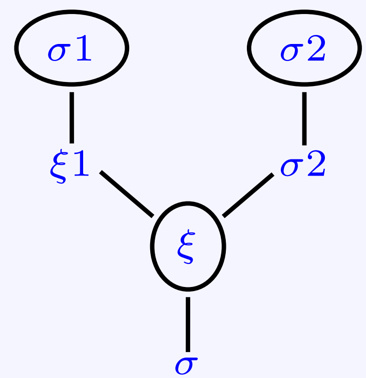
< > - +

Tree strategies are abstract sequent calculus



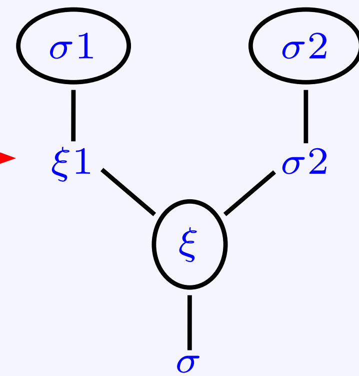






Syntax/semantics

$$\dfrac{\dfrac{\dfrac{\dots}{\vdash a^{\perp}, c} \qquad \dfrac{\dots}{\vdash b^{\perp}, d}}{\vdash c, d, a^{\perp} \otimes b^{\perp}}}{\vdash c \invamp d, a^{\perp} \otimes b^{\perp}}$$

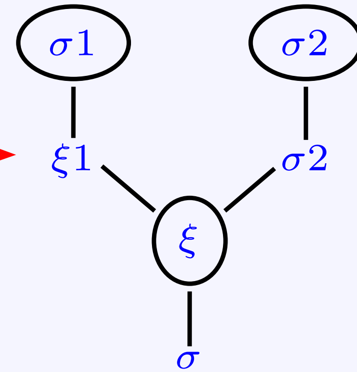


$$\dfrac{\dfrac{\dfrac{}{\vdash} \qquad \dfrac{}{\vdash}}{\vdash}}{\vdash \sigma, \xi}$$

$\xi : a^{\perp} \otimes b^{\perp}$

Syntax/semantics

$$\dfrac{\dfrac{\dfrac{\dots}{\vdash a^{\perp}, c} \qquad \dfrac{\dots}{\vdash b^{\perp}, d}}{\vdash c, d, a^{\perp} \otimes b^{\perp}}}{\vdash c \parr d, a^{\perp} \otimes b^{\perp}}$$



$$\dfrac{\dfrac{\dfrac{}{\vdash} \qquad \dfrac{}{\vdash}}{\vdash}}{\vdash \sigma, \xi}$$

$$\xi : a^{\perp} \otimes b^{\perp}$$

L-nets are abstract* MALL *proof-nets

< > - +

L-nets :

- Multiplicative structure (*partial order*)
- Additive structure (*superposition of p.orders)*

< > - +

Additive structure

Key technology: SLICES

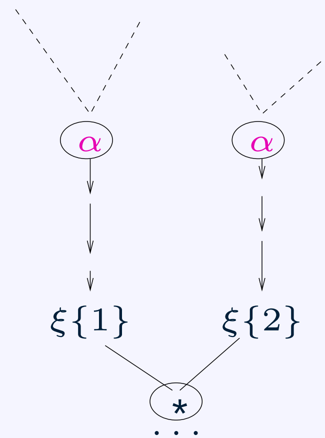
"Slices: the perfect syntax for $additive$ proof-nets"

- Think an $\&$ rule as superposition of unary rules: $\&_1, \&_2$

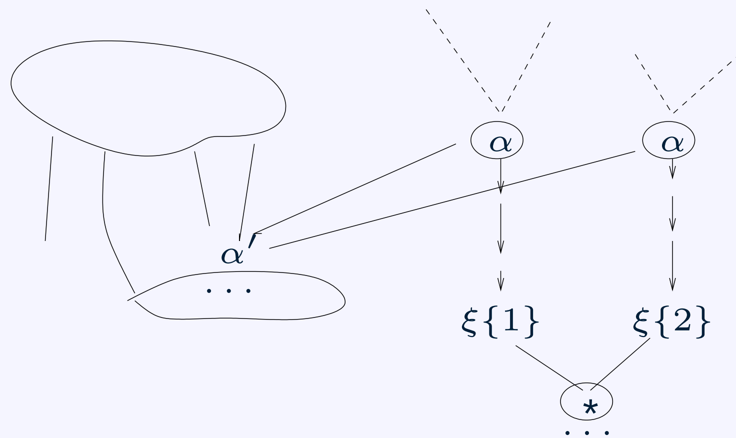
- Given a sequent calculus derivation, select one premiss for each $\&$-rule: you have a *slice*.

$$\dfrac{\dfrac{\vdots}{A, \Gamma} \qquad \dfrac{\vdots}{B, \Gamma}}{A\&B, \Gamma}\ \&$$

Additives and slices

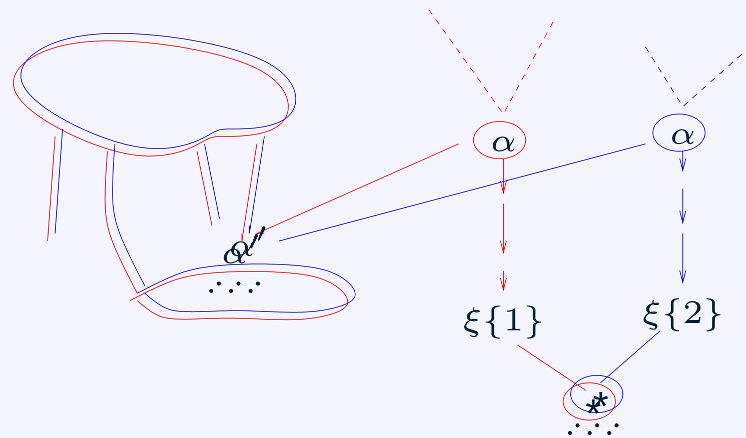


Additives and slices

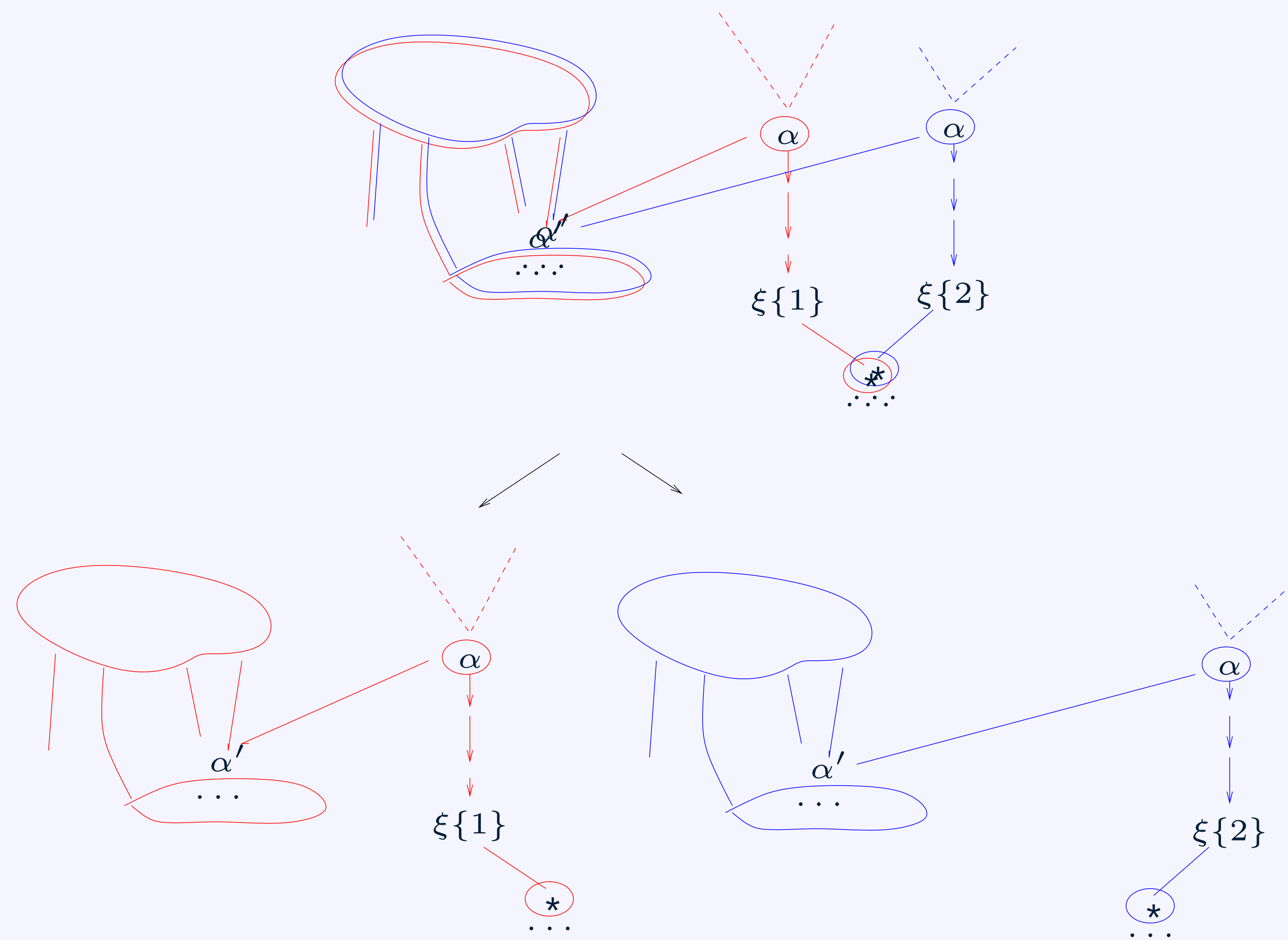


< > - +

Additives and slices



Additives and slices



< > - +

Slices: composition is easy

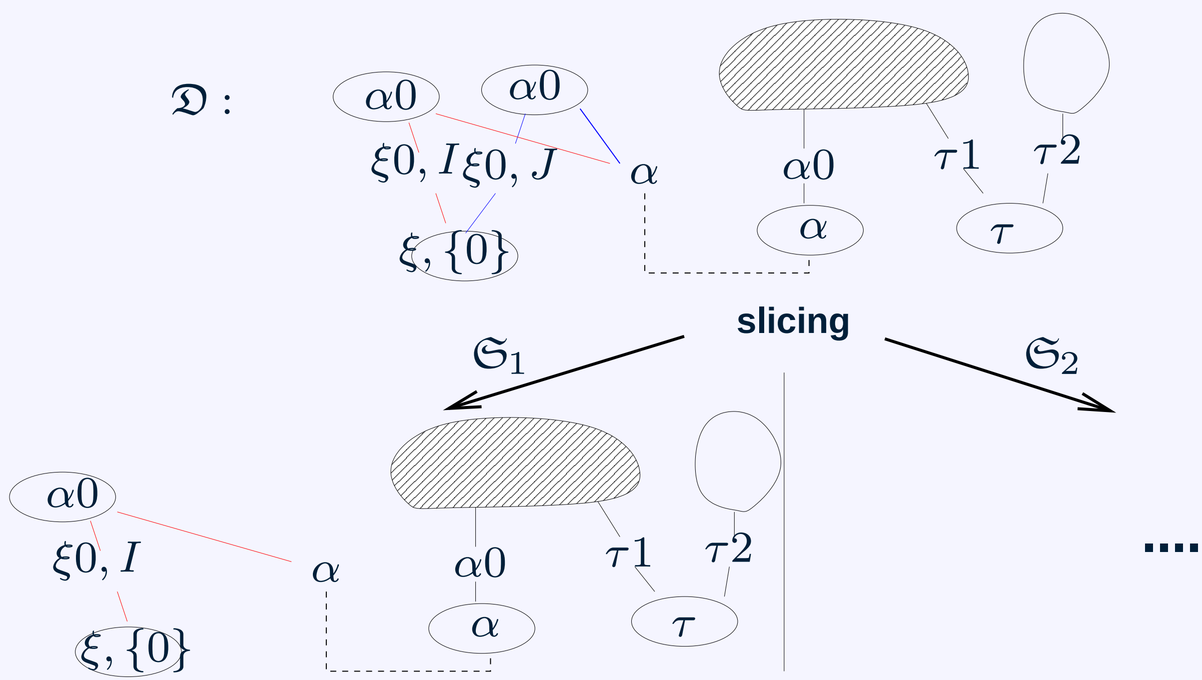
- merging of p.orders (as in Hyland)

- **MLL** proof-net style rewriting

- abstract machine (event structures reminescent)

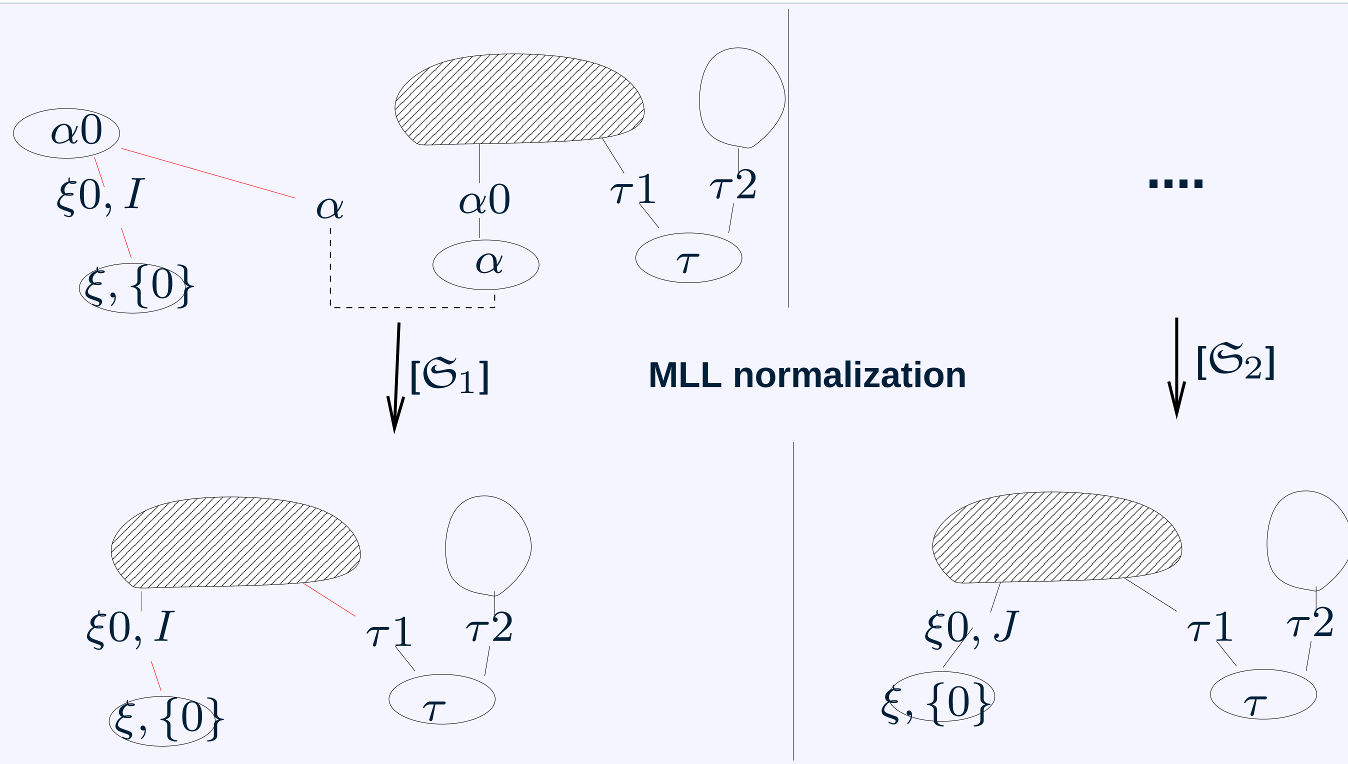
< > - +

L-nets composition



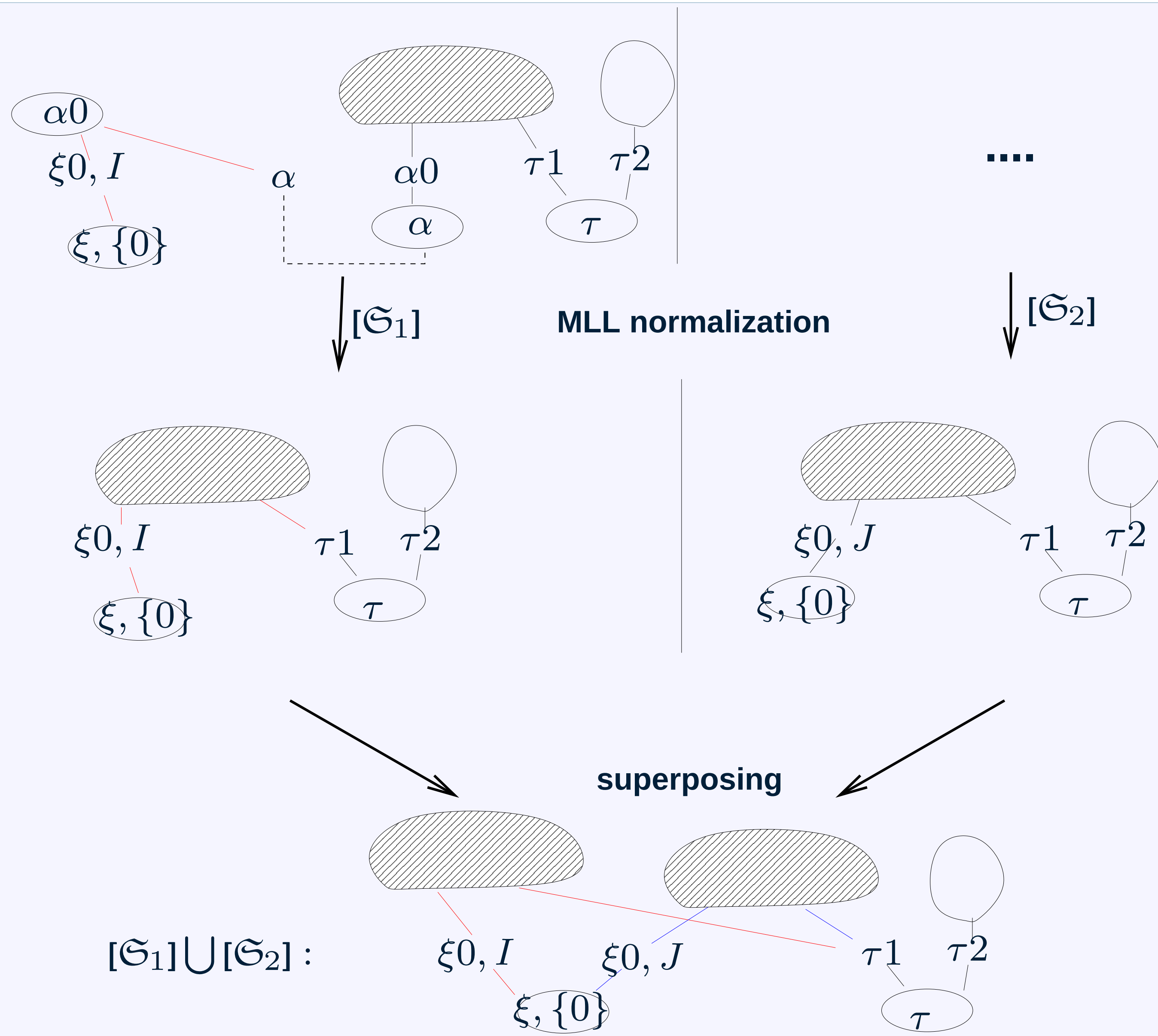
< > - +

L-nets composition (cont.)



< > - +

L-nets composition (cont.)



< > - +

(co)inductively define sequential and parallel L-nets

A (co)algebra of operators

Constructors:

- rooting, boxing, union, additive union

Deconstructors:

- root removal, splitting, scoping

Rooting and boxing

Two choices for adding a new negative node:

- **Rooting** $x^{-} \circ \mathfrak{D}$ **is a parallel operation,**
  adds the minimal amount of sequentiality (order) necessary to Justification.
- **Boxing** $x^{-}.\mathfrak{D}$ **is a serial (sequential) operation**
  which adds a maximal amount of sequentiality.
  (in terms of proof-nets, *boxing corresponds to enclosing $\mathfrak{D}$ in a box, which has x as principal port.*

Consistent use of rooting and boxing will produce (abstract versions of) proof-nets and sequent calculus derivations, respectively.

< > - +

Well-formed L-nets

$\mathfrak{D}^+$ is a positive L-net, $\mathfrak{D}^-_\sigma$ a negative L-net with negative conclusion σ.
k^+ is a positive action (axiom)

$$\begin{array}{lcl} \mathfrak{D} & := & \mathfrak{D}^+ \mid \mathfrak{D}^-_\sigma \\ \mathfrak{D}^+ & := & k^+ \mid \bigcup_i \; (\xi, I)^+ \circ \mathfrak{D}^-_{\xi_i} \\ \mathfrak{D}^-_\sigma & := & (\sigma, J)^- \triangleleft \mathfrak{D}^+ \end{array}$$

$$Order(x \circ \mathfrak{D}) \subseteq Order(x \triangleleft \mathfrak{D}) \subseteq Order(x.\mathfrak{D})$$

We are interested in the two canonical choices: consistentely using rooting (minimal sequentiality) or boxing (maximal sequentiality).

< > - +

Well-formed L-nets

$\mathfrak{D}^+$ is a positive L-net, $\mathfrak{D}^-_\sigma$ a negative L-net with negative conclusion σ.
k^+ is a positive action (axiom)

$$\begin{array}{lcl} \mathfrak{D} & := & \mathfrak{D}^+ \mid \mathfrak{D}^-_\sigma \\ \mathfrak{D}^+ & := & k^+ \mid \bigcup_i \ (\xi, I)^+ \circ \mathfrak{D}^-_{\xi_i} \\ \mathfrak{D}^-_\sigma & := & \biguplus_J \ (\sigma, J)^- \triangleleft \mathfrak{D}^+ \end{array}$$

$$Order(x \circ \mathfrak{D}) \subseteq Order(x \triangleleft \mathfrak{D}) \subseteq Order(x.\mathfrak{D})$$

We are interested in the two canonical choices: consistentely using rooting (minimal sequentiality) or boxing (maximal sequentiality).

Sequential and Parallel L-nets

Sequential L-nets (*Abstract sequent calculus derivation*)

$$
\begin{array}{lll}
\mathfrak{D} & := & \mathfrak{D}^{+} \mid \mathfrak{D}^{-}_{\sigma} \\
\mathfrak{D}^{+} & := & k^{+} \mid \bigcup_i \ (\xi, I)^{+} \circ \mathfrak{D}^{-}_{\xi_i} \\
\mathfrak{D}^{-}_{\sigma} & := & \bigcup_J \ (\sigma, J)^{-} . \mathfrak{D}^{+}
\end{array}
$$

Parallel L-nets (*Abstract proof-net*)

$$
\begin{array}{lll}
\mathfrak{D} & := & \mathfrak{D}^{+} \mid \mathfrak{D}^{-}_{\sigma} \\
\mathfrak{D}^{+} & := & k^{+} \mid \bigcup_i \ (\xi, I)^{+} \circ \mathfrak{D}^{-}_{\xi_i} \\
\mathfrak{D}^{-}_{\sigma} & := & \bigcup\!\!\!\!\downarrow_J \ (\sigma, J)^{-} \circ \mathfrak{D}^{+}
\end{array}
$$

< > - +

An application

A Game model of MALL

HS

Positive connectives: $\bigoplus_{I \in \mathcal{N}}(\bigotimes_{i \in I}(\downarrow N_i))$

$$\frac{\dots \quad \vdash N_i, \Delta_i \quad \dots \quad \vdash N_j, \Delta_j \quad \dots}{\vdash P, \dots, \Delta_i, \dots, \Delta_j, \dots} \ (P, N_I)$$

where $N_I = \{N_i : \ i \in I\}$

Negative connectives: $\&_{I \in \mathcal{N}}(\invamp_{i \in I}(\uparrow P_i))$

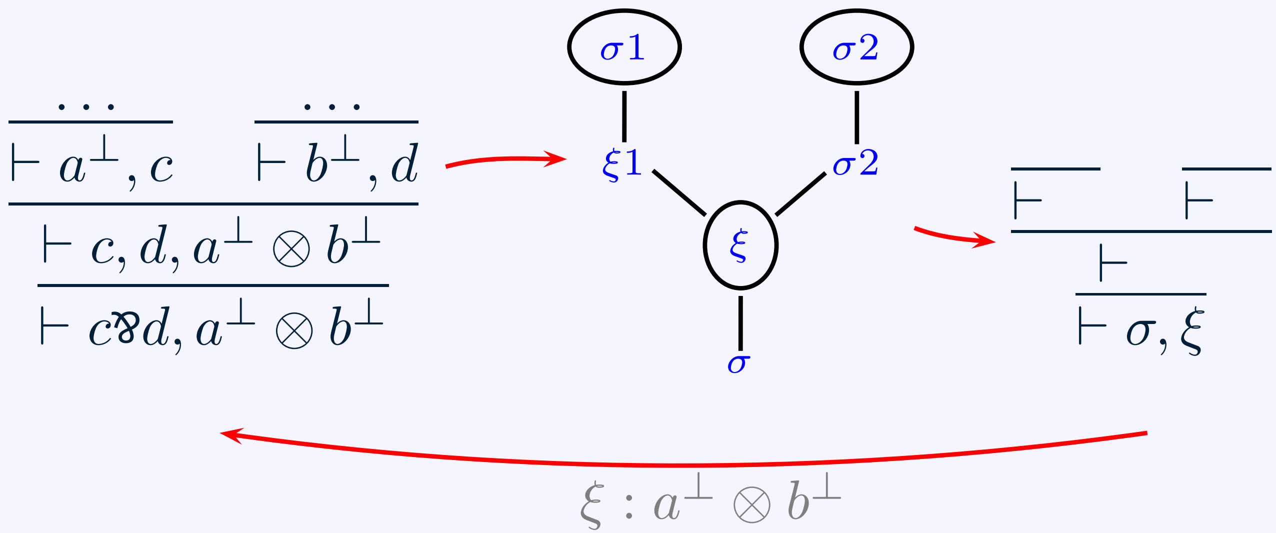
$$\frac{\dots \quad \vdash P_I, \Delta \quad \vdash P_J, \Delta \dots}{\vdash N, \Delta} \ \{\dots, (N, P_I), (N, P_J), \dots\}$$

where $P_I = \{P_i : \ i \in I\}$

< > - +

Syntax/semantics

Sequential interpretation:



Parallel interpretation:

L-nets are abstract ***MALL*** *proof-nets*

Ingredients

Interpret **formulas as arenas.**

Interpret **proofs as L-nets (sequential or parallel).**

Π *(sequential)* is **total** if for each negative action a^-

- if a^- is initial (in the arena), then $a^- \in \Pi$;
- if the chronicle $\mathfrak{c} = \ulcorner k \urcorner \in \Pi$ and $\overline{k} \vdash a$ (in the arena), then $\mathfrak{c}.a \in \Pi$

< > - +

Ingredients

Interpret **formulas as arenas.**

Interpret **proofs as L-nets (sequential or parallel).**

Π *(sequential)* is **total** if for each negative action a^-

- if a^- is initial (in the arena), then $a^- \in \Pi$;
- if the chronicle $\mathfrak{c} = \ulcorner k \urcorner \in \Pi$ and $\overline{k} \vdash a$ (in the arena), then $\mathfrak{c}.a \in \Pi$

$\mathfrak{D}$ *(any L-net)* is **total** if for any slice S of the arena,
$\mathfrak{D} \upharpoonright S$ satisfies the following, for each negative action a^-:

- if a^- is initial (in S), then $a^- \in \mathfrak{D}$;
- if the chronicle $\mathfrak{c} = \ulcorner k \urcorner \in \mathfrak{D}$ and $\overline{k} \vdash a$, then $\mathfrak{c}.a \in \mathfrak{D}$
  (where $\mathfrak{c}.a$ is obtained by extending $\mathfrak{c}$ with a new top action a).

< > - +

Back to the initial picture

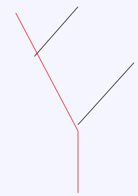
Have an arena: a forest of actions (parent relation)

TREE STRATEGIES

set of views

view: sequence of actions

play: total order



abstract

sequent calculus

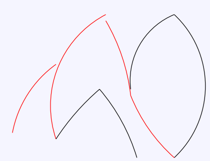
total $\Rightarrow MALL^{foc}$
(full completeness)

PARALLEL STRATEGIES

set of p.o. views

p.o.view of k: $\{k', k' \leq k\}$

play: partial order



abstract

proof-net

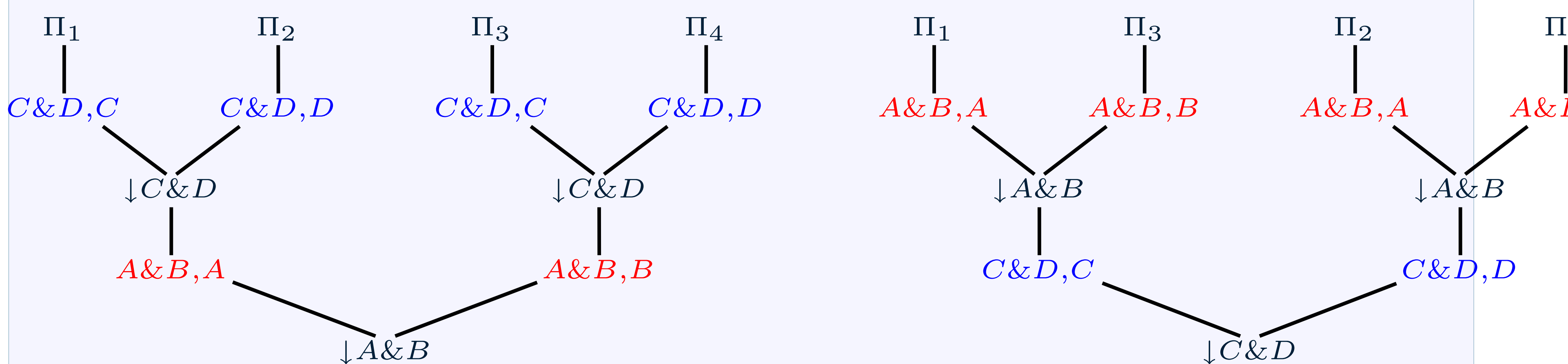
total $\Rightarrow$ proof-net is
$MALL^{foc}$ typable

< > - +

An additive example

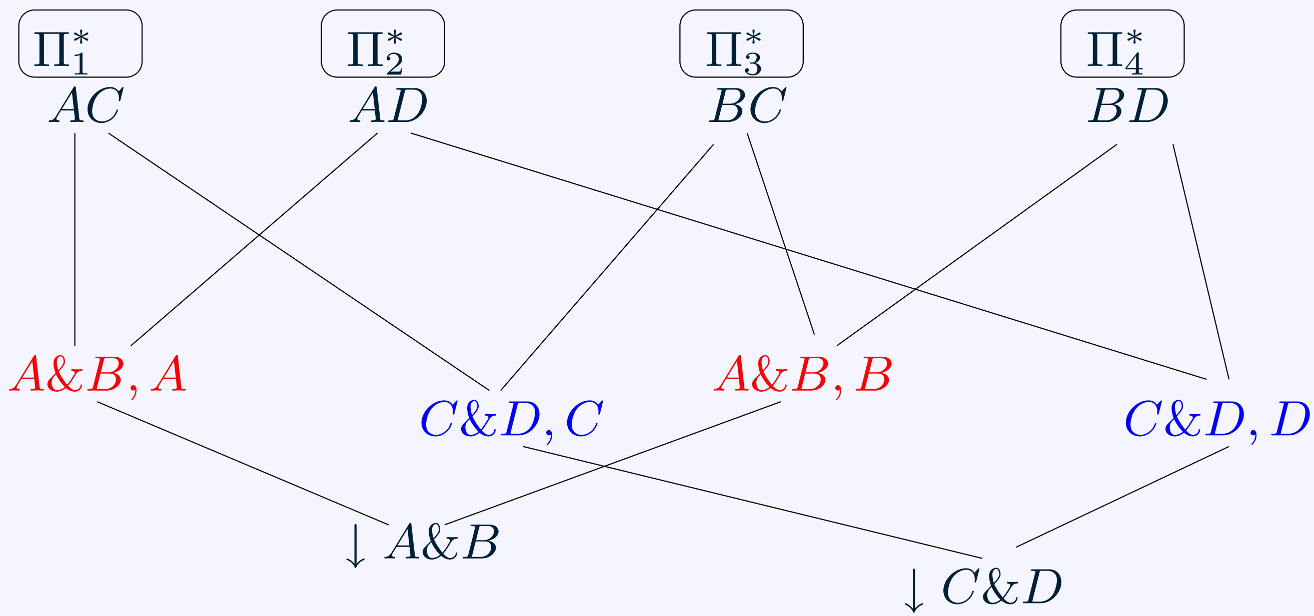
$$\dfrac{\dfrac{\dfrac{\Pi_1}{\vdash A, C} \quad \dfrac{\Pi_2}{\vdash A, D}}{\vdash A, C\&D}\ {\scriptstyle C\&D} \qquad \dfrac{\dfrac{\Pi_3}{\vdash B, C} \quad \dfrac{\Pi_4}{\vdash B, D}}{\vdash B, C\&D}\ {\scriptstyle C\&D}}{\vdash A\&B, C\&D}\ {\scriptstyle A\&B}$$

$$\dfrac{\dfrac{\dfrac{\Pi_1}{\vdash A, C} \quad \dfrac{\Pi_2}{\vdash A, D}}{\vdash A\&B, C}\ {\scriptstyle A\&B} \qquad \dfrac{\dfrac{\Pi_3}{\vdash B, C} \quad \dfrac{\Pi_4}{\vdash B, D}}{\vdash A\&B, D}\ {\scriptstyle A\&B}}{\vdash A\&B, C\&D}\ {\scriptstyle C\&D}$$



< > - +

Additive example, cont.



< > - +

(Some) further and future work

- Proof-nets
  - Which proof-nets? close to Hughes and van Glabbeek; extension of Andreoli 02.
  - move from MALL proof-nets to sequent calculus derivations in a continuum (Girard):
    vary the amount of sequentiality (order) on the graphs from the most-parallel to most-sequential.
    With Di Giamberardino: multiplicatives.
- Beyond the core: more expressivity...
  - Fix points (with Saurin)
- What about concurrency?
  - event structures (see Varacca-Yoshida.)
  - positive non-determinism, with Baillot (idea: HOPLA)

(Most) related work: *Abramsky and Melliès, Hyland, Schalk, Melliès and Mimram*

< > - +

About point 3: concurrency

- Multiplicative structure (*parallelism*)
- Additive structure (*“non-determinism” ?)*[a]

Can we make the proof-theoretical intuition precise?

The matter:
Strategies are event structures, additive components express conflict, a slice is a conflict-free confi guration...

BUT:

In event structures, conflict models non-determinism

vs.

in MALL strategies, only external choice.

Q: Can we have positive non-determinism in strategies?
Maybe yes: key ideas in Varacca-Yoshida.

[a] [see Mairson-Terui 03]

A remind on event structures

Regard distributed computation as

- event occurrences
- a relation expressing causal dependency (partial order).

To model nondeterminism adjoin further structure: a conflict relation between events, to express how the occurrence of certain events rules out the occurrence of others.

$$(E, \leq, \sharp)$$

< > - +

Event structures (2)

Computation state: set x of events which have occurred in computation

- if an event has occurred then all events on which it depends have occurred too
- not two conflicting events can occur together in the same computation

Given $(E, \leq, \sharp)$, its configurations are those $x \subseteq E$ which are

- conflict-free
- downwards-closed

Two events are *concurrent* if they are neither comparable nor in conflict.

< > - +

More details

http://www.math.unipd.it/~claudia

F., Maurel "Ludics Nets: a game model of concurrent interaction", LICS 05.

Curien, F. "A graph approach to innocent strategies" (45 pages), submitted.

F. "Linear Logic Game Semantics: sequential and parallel," submitted.

< > - +