Computational Paradigms: focusing proof nets construction

Roberto Maieli

Università degli Studi "Roma Tre"
maieli@uniroma3.it

joint research topic with Jean-Marc Andreoli
(Xerox Research Centre Europe, Grenoble)

GLGC Workshop "Ricerche logiche ed epistemologiche" – 9 July, 2010

computational paradigms based on logic

computational concepts $\Leftrightarrow$ proof-theoretical concepts

Computation	**Proof theory**	
paradigm	*functional (Curry-Howard)*	*logic programming (?)*
state (q_i, s_j)	proof: with cuts without proper axioms	proof: cut-free with proper axioms
transformation $(q_i, s_j) \rightarrow (s_{j'}, L/R, q_{i'})$	proof reduction: i.e. cut-elimination	proof construction: i.e. proper axioms elimination
final state q_0	cut-free proof	proof without proper axioms
type	formula	formula/sequent

computational paradigms based on Linear Logic

- **Classical** and **Intuitionistic Logic** deal with **stable truths**.
- **Linear Logic** deals with **actions** or **resources**.
- Consider the pure multiplicative-additive fragment of LL (**MALL**):

identity : $\dfrac{}{A, A^{\perp}}\,\mathrm{ax}$ $\qquad$ $\dfrac{\Gamma, A \qquad \Delta, A^{\perp}}{\Gamma, \Delta}\,\mathrm{cut}$

multiplicatives : $\dfrac{\Gamma, A \qquad \Delta, B}{\Gamma, \Delta, A \otimes B}\otimes$ $\qquad$ $\dfrac{\Gamma, A, B}{\Gamma, A \parr B}\parr$

additives : $\dfrac{\Gamma, A \qquad \Gamma, B}{\Gamma, A \& B}\&$ $\qquad$ $\dfrac{\Gamma, A}{\Gamma, A \oplus_1 B}\oplus_1$ $\qquad$ $\dfrac{\Gamma, B}{\Gamma, A \oplus_2 B}\oplus_2$

LL: determinism vs non-determinism

$$\frac{\Gamma, A \qquad \Delta, B}{\Gamma, \Delta, A \otimes B}\otimes \qquad\qquad \frac{\Gamma, A \qquad \Gamma, B}{\Gamma, A\&B}\&$$

$[\otimes, \&]$ they correspond to two radically different uses of the word "and". Both conjunctions express the **availability of two actions**:
– in the case of $\otimes$ both actions will be done,
– in the case of $\&$ only one of them will be performed (we decide which).
Assume an action of type $\vdash A^{\perp}, B$ and an action of type $\vdash A^{\perp}, C$ then there will be no way of performing an action $\vdash A^{\perp}, B \otimes C$ whereas there will be an *additive* action of type $\vdash A^{\perp}, B\&C$, namely the *superposition* of both actions. In order to perform this additive action we have first to choose which among the two possible actions we want to perform, and then to do the one selected. This is an exact analogue of

the **computer science instruction IF...THEN...ELSE**
both the two parts "THEN..." and "ELSE..." are available,
but only one of them will be done.

A typical misconception is to view the connective "&" as a disjunction.

LL: determism vs non-determinism

$$\frac{\Gamma, A}{\Gamma, A \oplus_1 B}\ \oplus_1 \qquad \frac{\Gamma, B}{\Gamma, A \oplus_2 B}\ \oplus_2$$

[$\oplus$] this is the dual of $\&$, which expresses the **choice of one action between two possible types**.
In terms of computer science, the distinction $\&/\oplus$ is a reminiscient of the distinction **outer/inner non-determinism**.
In an action of type $\vdash A^{\perp}, B \oplus C$ we can no longer decide which resource among B and C we shall get ("inner" or "true" non-determinism).

$$\frac{(1)\ \Gamma, A \qquad (2)\ \Gamma, B}{\Gamma, A \& B}\ \&$$

if ... then ... else ...
outer non-determinism

$$\frac{(3)\ \Delta, A^{\perp}}{\Delta, A^{\perp} \oplus B^{\perp}}\ \oplus_1$$

choice of the action A
inner non-determinism

$$\frac{(4)\ \Delta, B^{\perp}}{\Delta, A^{\perp} \oplus B^{\perp}}\ \oplus_2$$

choice of the action B
inner non-determinism

LL: determism vs non-determinism

$$\frac{\Gamma, A}{\Gamma, A \oplus_1 B}\ \oplus_1 \qquad\qquad \frac{\Gamma, B}{\Gamma, A \oplus_2 B}\ \oplus_2$$

[$\oplus$] this is the dual of $\&$, which expresses the **choice of one action between two possible types**.
In terms of computer science, the distinction $\&/\oplus$ is a reminiscient of the distinction **outer/inner non-determinism**.
In an action of type $\vdash A^{\perp}, B \oplus C$ we can no longer decide which resource among B and C we shall get ("inner" or "true" non-determinism).

$$\dfrac{\dfrac{\overset{(1)}{\Gamma, A} \qquad \overset{(2)}{\Gamma, B}}{\Gamma, A\&B}\ \& \qquad \dfrac{\overset{(3)}{\Delta, A^{\perp}}}{\Delta, A^{\perp} \oplus B^{\perp}}\ \oplus_1}{\Gamma, \Delta}\ cut \qquad \mapsto \qquad \dfrac{\overset{(1)}{\Gamma, A} \qquad \overset{(3)}{\Delta, A^{\perp}}}{\Gamma, \Delta}\ cut$$

"IF ... THEN ..." part

LL: determism vs non-determinism

$$\frac{\Gamma, A}{\Gamma, A \oplus_1 B}\,\oplus_1 \qquad\qquad \frac{\Gamma, B}{\Gamma, A \oplus_2 B}\,\oplus_2$$

[$\oplus$] this is the dual of $\&$, which expresses the **choice of one action between two possible types**.
In terms of computer science, the distinction $\&/\oplus$ is a reminiscient of the distinction **outer/inner non-determinism**.
In an action of type $\vdash A^{\perp}, B \oplus C$ we can no longer decide which resource among B and C we shall get ("inner" or "true" non-determinism).

$$\dfrac{\dfrac{(1)\quad \Gamma, A}{\Gamma, A\&B}\;\&\;\dfrac{(2)\quad \Gamma, B \qquad\qquad \dfrac{(4)\quad \Delta, B^{\perp}}{\Delta, A^{\perp} \oplus B^{\perp}}\,\oplus_1}{\Gamma, \Delta}\,cut \qquad \mapsto \qquad \dfrac{\dfrac{(2)\quad \Gamma, B \qquad\qquad (4)\quad \Delta, B^{\perp}}{\Gamma, \Delta}\,cut$$

"ELSE ..." part

LL: determism vs non-determinism

$$\frac{\Gamma, A, B}{\Gamma, A \parr B} \parr$$

[$\parr$] this is the dual of $\otimes$;

it expresses **a dependency between two types of actions**:

an action of type $\vdash A \parr B$ can either be read as $\vdash A^{\perp} \multimap B$ or as $\vdash B^{\perp} \multimap A$, where "$\multimap$" is the linear implication;

Computational Paradigm based on Proof-Construction

permutations of rules are "problematic" w.r.t. the sequent calculus

$$\dfrac{\dfrac{\dfrac{(1)}{\dfrac{A, A^\perp}{A, A^\perp \oplus A^\perp}\,\oplus_2} \qquad \dfrac{\dfrac{(2)}{A, A^\perp}}{A, A^\perp \oplus A^\perp}\,\oplus_1}{A \& A, A^\perp \oplus A^\perp}\,\& \qquad \dfrac{\dfrac{(3)}{C, C^\perp}}{C \parr C^\perp}\,\parr}{A \& A, (A^\perp \oplus A^\perp) \otimes (C \parr C^\perp)}\,\otimes$$

$$\dfrac{\dfrac{\dfrac{(1)}{A, A^\perp} \qquad \dfrac{(2)}{A, A^\perp}}{A \& A, A^\perp}\,\&}{A \& A, A^\perp \oplus A^\perp}\,\oplus_i \qquad \dfrac{\dfrac{(3)}{C, C^\perp}}{C \parr C^\perp}\,\parr}{A \& A, (A^\perp \oplus A^\perp) \otimes (C \parr C^\perp)}\,\otimes$$

$$\dfrac{\dfrac{\dfrac{(1)}{\dfrac{A, A^\perp}{A, A^\perp \oplus A^\perp}\,\oplus_2} \qquad \dfrac{\dfrac{(3)}{C, C^\perp}}{C \parr C^\perp}\,\parr}{A, (A^\perp \oplus A^\perp) \otimes (C \parr C^\perp)}\,\otimes \qquad \dfrac{\dfrac{\dfrac{(2)}{A, A^\perp}}{A, A^\perp \oplus A^\perp}\,\oplus_1 \qquad \dfrac{\dfrac{(3)}{C, C^\perp}}{C \parr C^\perp}\,\parr}{A, (A^\perp \oplus A^\perp) \otimes (C \parr C^\perp)}\,\otimes}{A \& A, (A^\perp \oplus A^\perp) \otimes (C \parr C^\perp)}\,\&$$

these three proofs are "equivalent",
modulo duplication/contraction of the sub-proof (3)

Question: there exist "canonical" representations for proofs?

proof nets, focusing and polarities

These problems may be overcame by exploiting the following tools:

- **proof nets**: parallel proofs, instead of sequential proofs;
- **polarities (+/-)**: consequence of the **focusing property**
  - **negative**: $\parr$ (multiplicative), & (additive)
  - **positive**: $\otimes$ (multiplicative), $\oplus$ (additive)

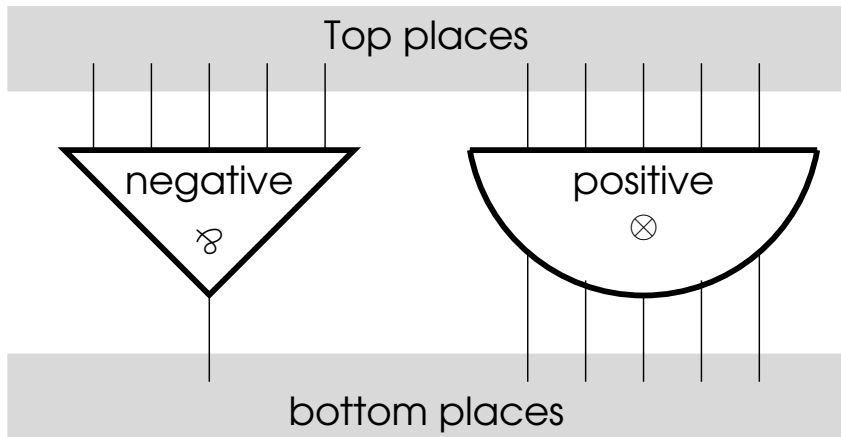
proof net construction paradigm: outline

- the construction of Focusing Bipolar Proof Nets by a set of concurrent agents (links);
- this construction (expansion) must be performed in an incremental and local (possibly efficient) way by link juxtaposition.
- agents are elementary components (positive or negative links) whose role is to guarantee that the overall structure they build is correct (still a proof net);
- checking correctness of a PN is a task that may require to explore (travel and freeze) a large portion of the structure, resulting in conflicts between agents and forcing them to take turn (scheduling).
- we try to identify "precisely" (a lower bound) the portion of structure must be checked (and locked) by a candidate (to the expansion) agent in such a way to reduce possible conflicts with other concurrent candidates.

links

– assume an infinite set $\mathcal{L}$ of (possibly untyped) *places* (or *loci*) $l_1, l_2, \ldots$;
– a *link* L consists in two disjoint sets of loci, *top* and *bottom*, together with a polarity $+/-$ and s.t.

- a **positive link** must have *at least one* bottom place;
- a **negative link** must have *exactly one* bottom place.



– negative links correspond to generalized (n-ary) $\parr$-links
– positive links correspond to generalized $\otimes$-links.

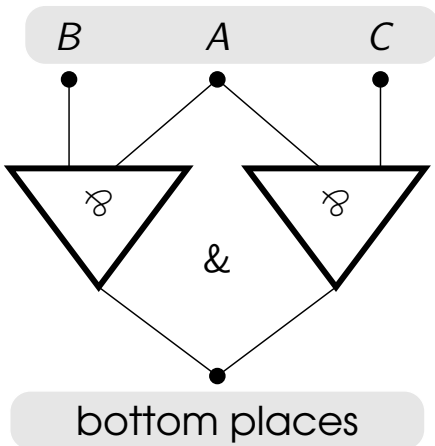
hyperlinks (or cells)

An *hyperlink* λ is a set of links that share some (at least one) places;

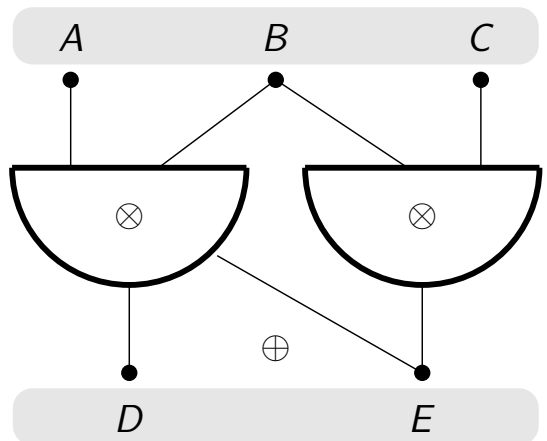
negative hyperlink contains only negative links and it has *exactly one* bottom place (all the negative links occurring in it share the same bottom place);

positive hyperlink contains only positive links and it has *at least one* bottom place.

Examples (distributivity) :



$A \parr (B \& C) = (A \parr B) \& (A \parr C)$



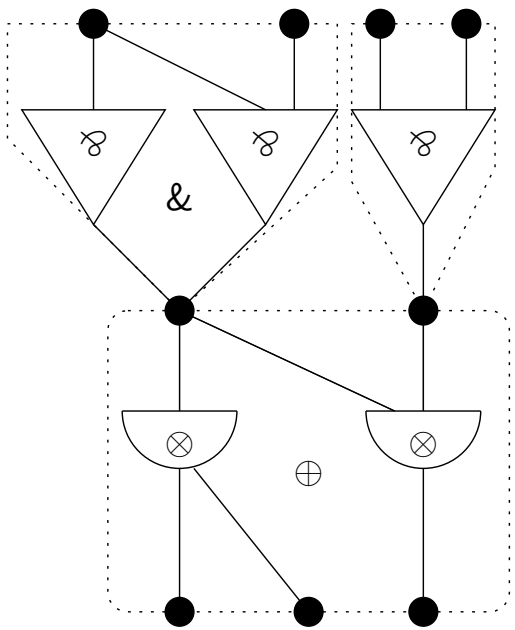
$((A \otimes D^{\perp}) \oplus C)) \otimes B \otimes E^{\perp} = (A \otimes D^{\perp} \otimes B \otimes E^{\perp}) \oplus (C \otimes B \otimes E^{\perp})$

(focusing) bipolar proof structures

a **proof structure** (PS) is a set π of hyperlinks s.t.:

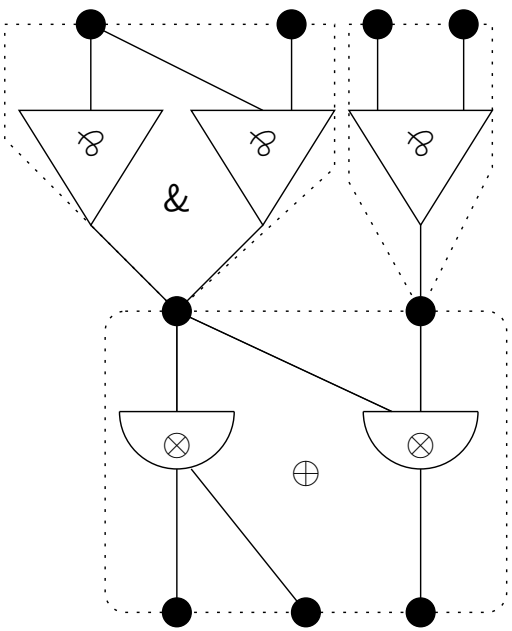
1. the sets of top (resp., bottom) places of any two hyperlinks of π are disjoint;
2. if two hyperlinks are adjacent, then they have opposite polarity.

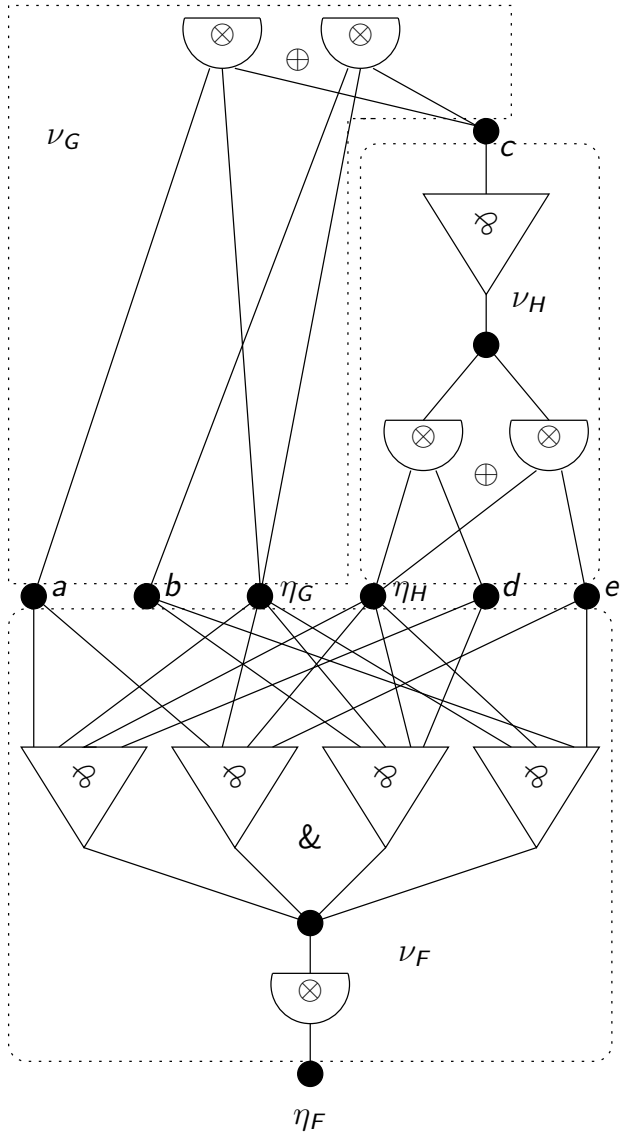
A PS is said **bipolar** (BPS) if any place occurring at the top of some positive hyper-link of π also occurs at the bottom of some negative hyper-link of π and vice-versa (the place occurring at the bottom of any negative hyper-link, also occurs at the top of some positive hyper-link).



examples of BPS

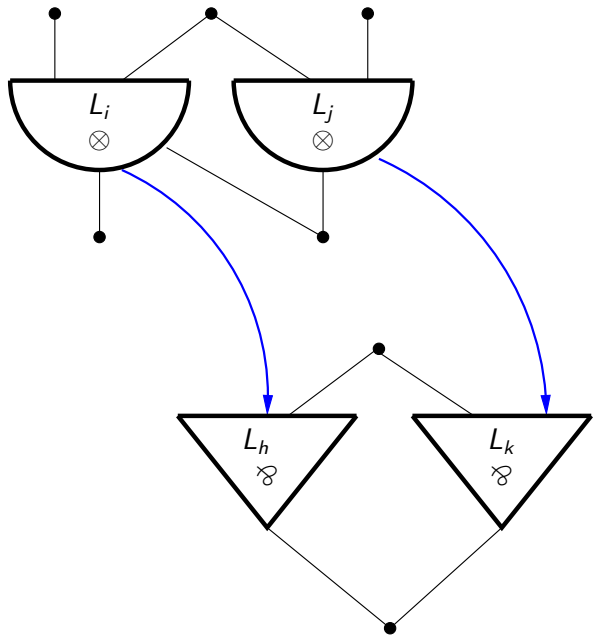
A BPS is *elementary* (**bipole**) if it contains exactly a +hyperlink.
Below: a bipole and a BPS with three bipoles.



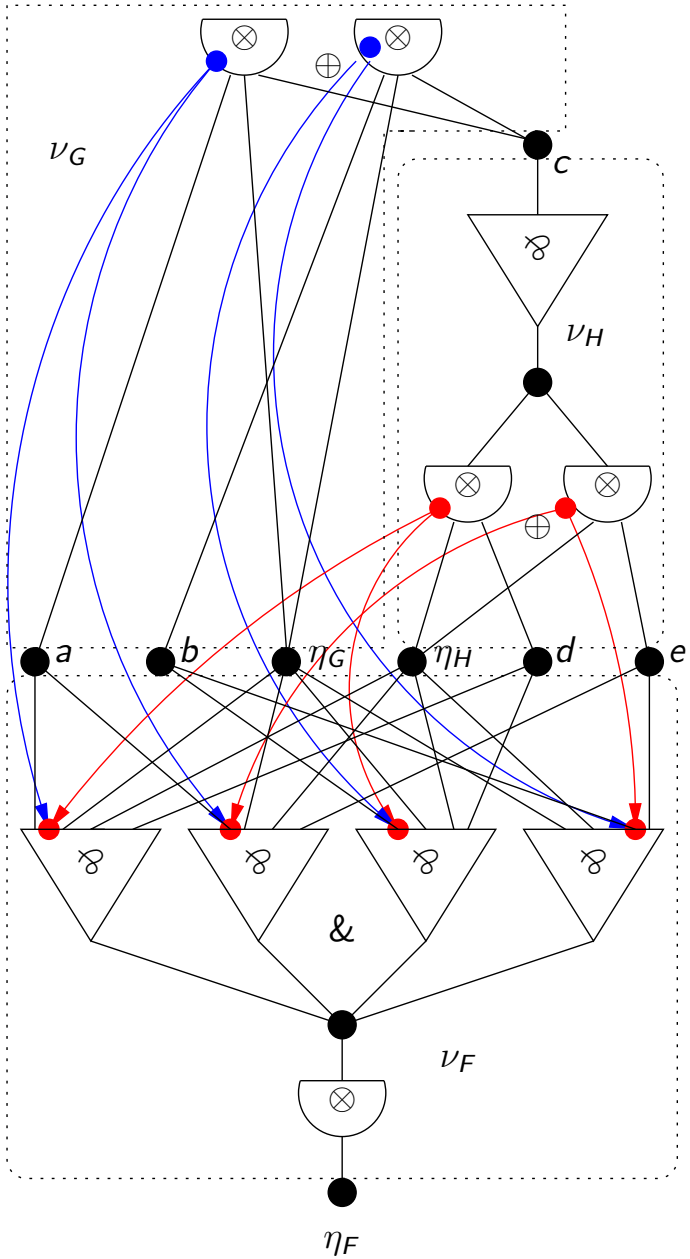


(1) – jumping a BPS

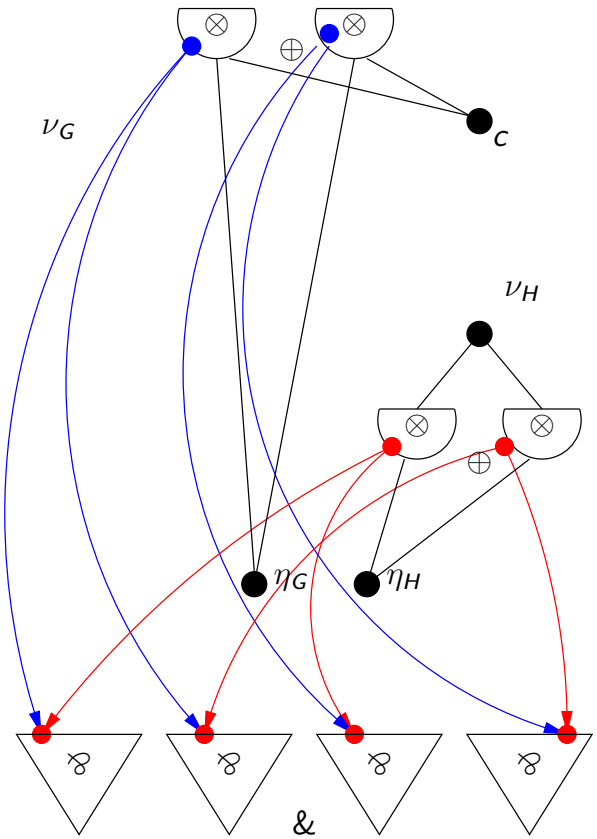
- A **jump** is an oriented edge going from the bottom of a positive $\otimes$-link to the top of a negative $\parr$-link.
- Given a BPS π a **jumping** J for π is the graph π in which we added jumps as follows: whenever two positive links L_i, L_j (belonging to a same positive hyperlink) share a top or bottom place, there exist two negative links L_h, L_k (belonging to a same negative hyperlink) s.t. there is a jump from L_i to L_h and a jump from L_j to L_k.



example of jumping for a BPS



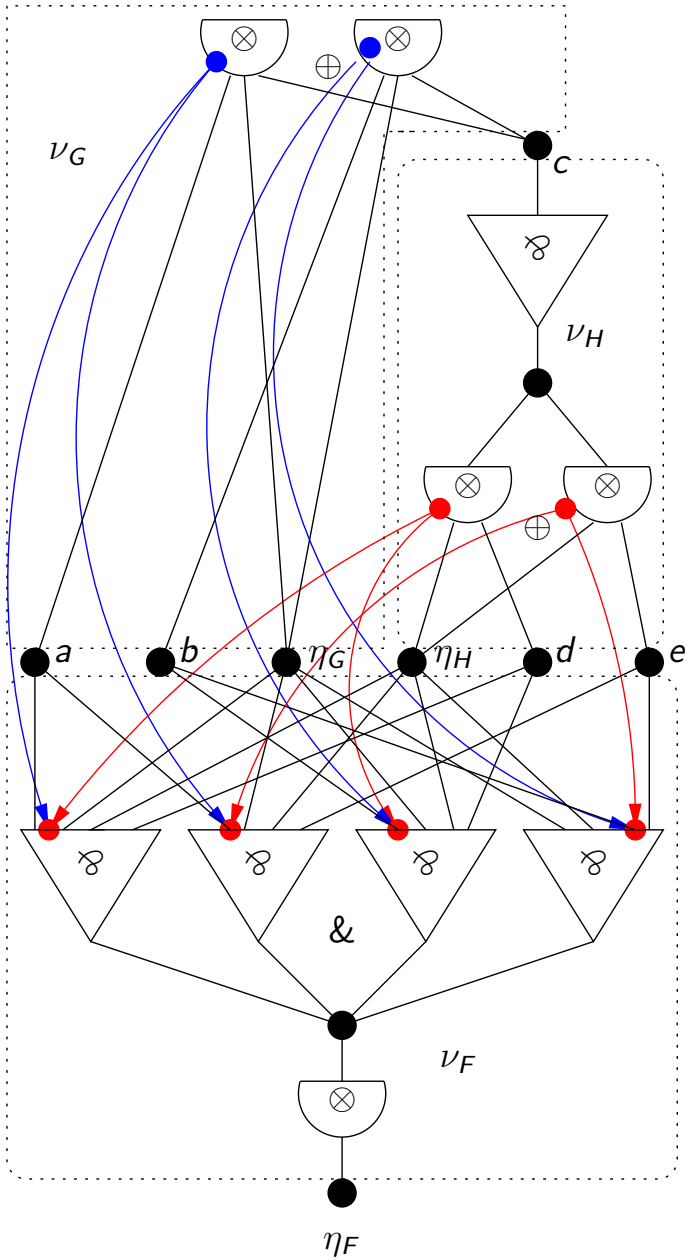
$\mapsto J$



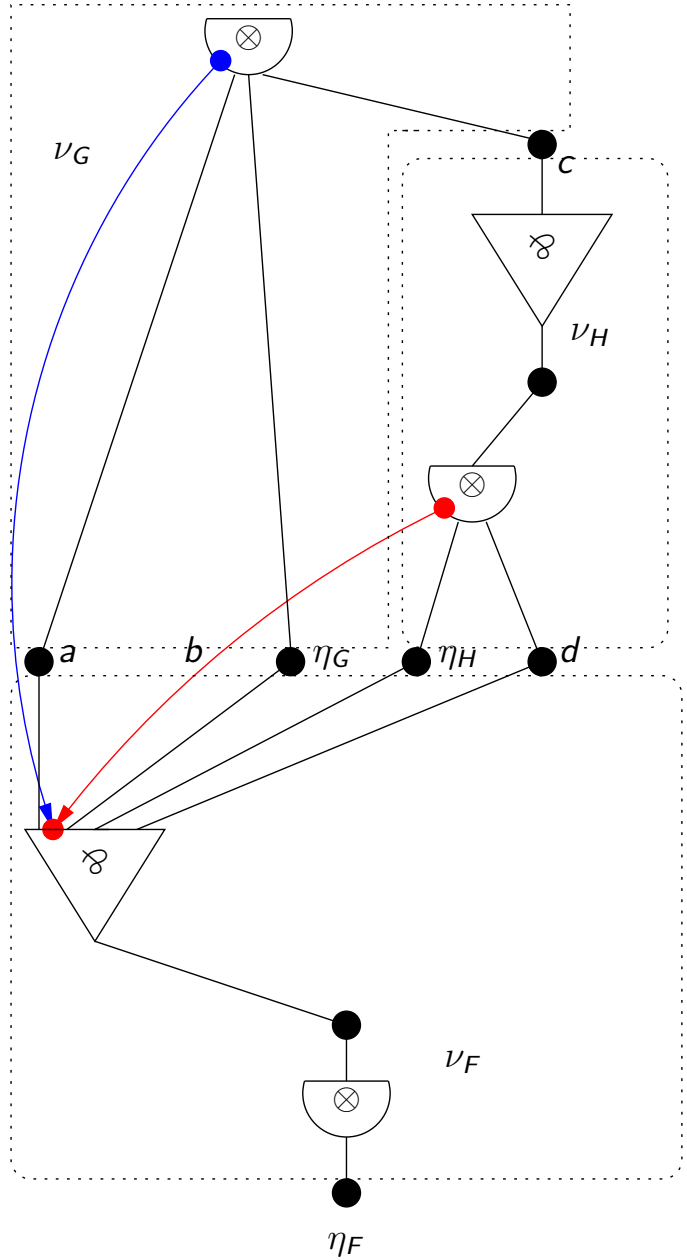
(2) – slicing a proof structure

- a &-**resolution** is a choice of one negative link for each negative hyperlink (all the other negative links will be erased).
- given a jumping J for π, a **slice** $S_J(\pi)$ for $J(\pi)$ is the graph obtained from $J(\pi)$ by the "natural erasing" induced by a &-resolution.
  - a place is destroyed when all top (resp. bottom) links sharing it are erased;
  - a $+$ or - link is erased when at least one of its places is erased.

example of slicing for a BPS

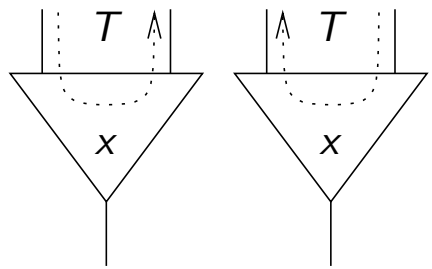


$\mapsto_S$



(3) – traveling a BPS (correctness)

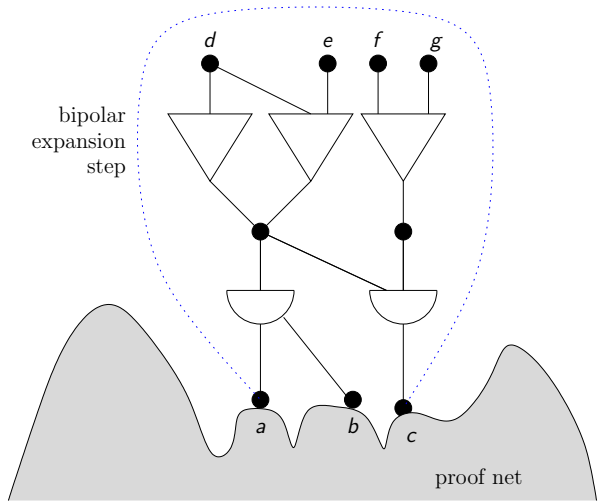
- **trip** T in a slice S for π is a non-empty binary relation on $|S|$ (the set of links of S) which is finite, connected and s.t. any link $x \in |S|$ has at most one successor (resp., predecessor), if it exists;
- a *negative middle link* x (with a predecessor and a successor) of a *proper trip* T (not reduced to a loop with only two links) is a **singularity** for T iff T *enters* x *downwards* and *exists* x *upwards*



- a BPS π is correct (**proof net**) if there exists a jumping J for π s.t. :
  – every &-resolution for $J(\pi)$ induces a unique slice S of π and
  – every trip T in S contains (at least) a singularity (i.e., S is a multiplicative PN!).
- a BPN corresponds to a **sequentializable proof** and vice-versa.

PN construction as middleware computational paradigm

– Bipoles drive the construction bottom-up.
– In the building BPN, places are decorated by type informations (occurrences of negative atoms);
– An expansion of the proof net is obtained by adding a bipole from the places whose types match the trigger (bottom places) of the given bipole: this expansion must preserve the correctness.

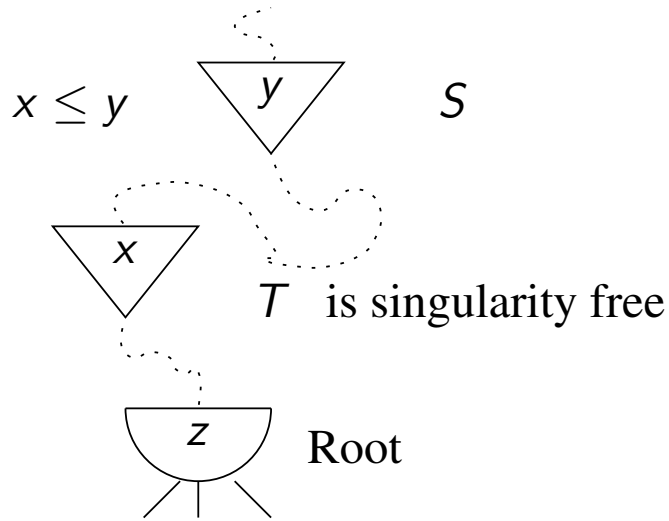


– During this construction we must:

1. restrict the traveling region (reducing conflicts among agents);
2. lock the gathered information against attempts of other agents;
3. increment, in case of success, the locked information for transition.

expansion under domination (1/2)

- Assume x, y are two negative links in a slice S for π;
  a *root* of S is any (positive) link of S that has no link below it; then:
  $x \leq y$ (x **dominates** y) if any singularity-free trip starting at a root and stopping upwards at y visits x upwards.



- The relation $\leq$ on $-$links of S is a **forest order**; it is reflexive, antisymmetric, transitive and it satisfies the following on $-$links:

$$\forall x, y, z \;\; \textit{if} \;\; (x \leq z \;\wedge\; y \leq z) \;\; \textit{then} \;\; (x \leq y \;\vee\; y \leq x).$$

- The **joint dominator of** N, $\bigwedge(N)$, is the *greatest lower bound* (if it exists), by $\leq$, of a set of negative links N.

expansion under domination (2/2)

assume x, y are two negative links and T a singularity-free trip of a slice S for π starting downwards at x and stopping upwards at y; then, any negative link z visited by T is *strictly dominated* by the joint dominator

$$\forall z^- \in |T|, \bigwedge\{x, y\} < z.$$

