

Jump from parallel to sequential proofs: additives.

Paolo Di Giamberardino

`digiambe@uniroma3.it`

Dip. di Filosofia, Università Roma Tre - Institute de Mathématiques de Luminy

Introduction

We introduce a new class of multiplicative/additive proof nets, *J-proof nets*, which are a typed version of Faggian and Maurel's L-nets.

In J-proof nets, we can characterize nets with different degrees of sequentiality, by gradual insertion of sequentiality constraints (jumps). As a byproduct, we obtain a simple proof of the sequentialisation theorem.

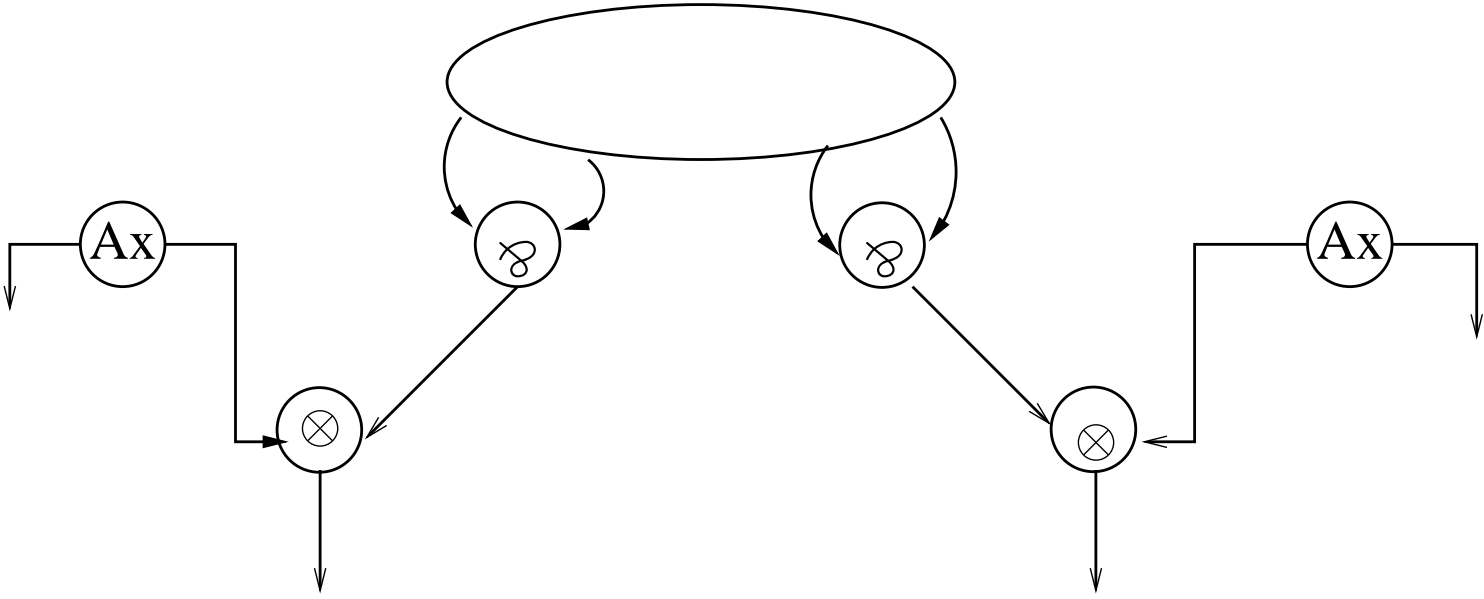
Due to the close connection between J-proof nets, L-nets (linear strategies), and event structures (models of concurrency), our sequentialisation method applies in all these domains.

Aims

1. give a simple proof of sequentialisation, using jumps (Girard);
2. pass (gradually) from proof nets to sequent calculus in the same setting;
3. understand what kind of proof nets are the L-nets.

Sequentialising with jump (1)

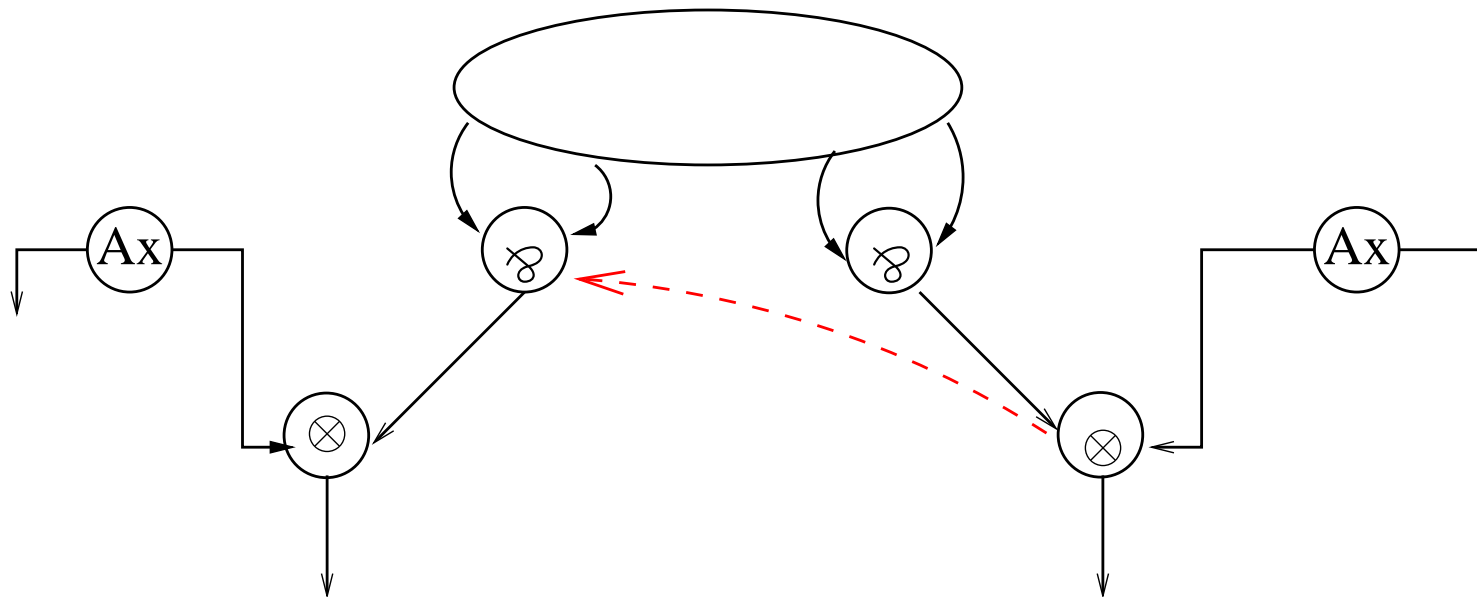
Proof nets: minimal information about the order of the rules (allowing *canonicity*)



Sequentialising with jump (2)

Jump (Girard): untyped oriented edge from one link a to another link b .

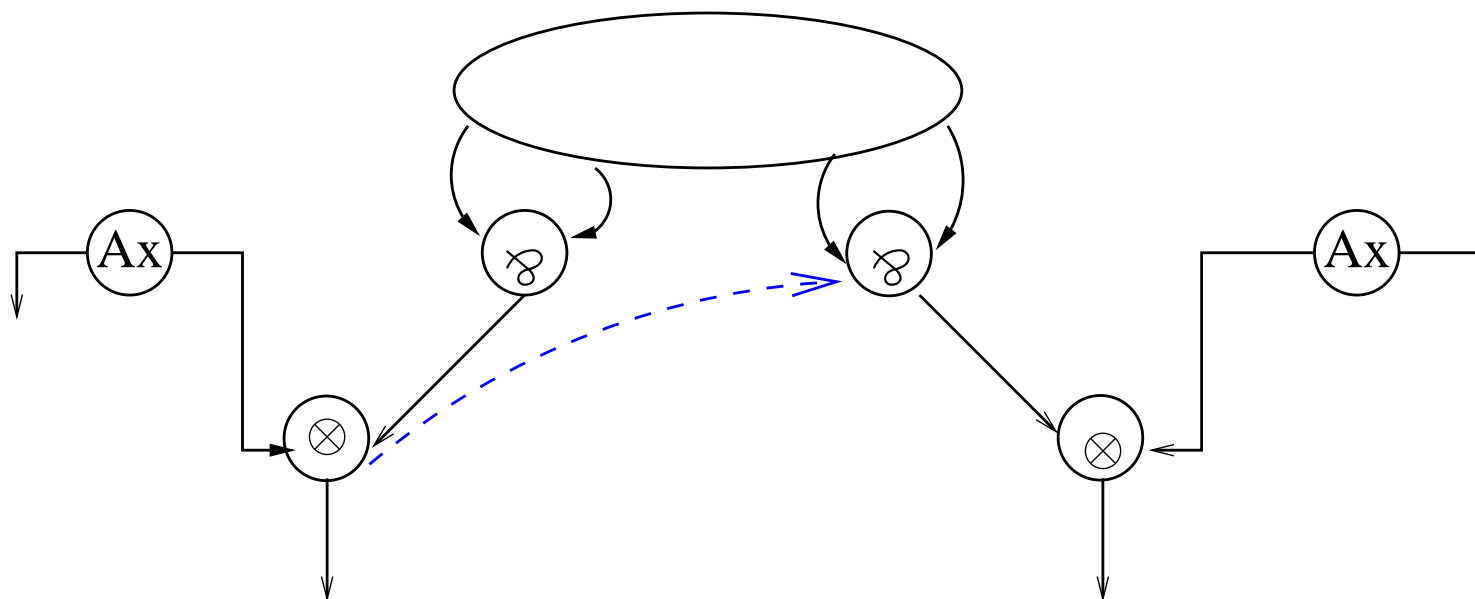
a jumps to b = as a rule, b precedes a (bottom-up) in a sequent calculus proof (dependency relation).



Sequentialising with jump (3)

Jump (Girard): untyped oriented edge from one link a to another link b .

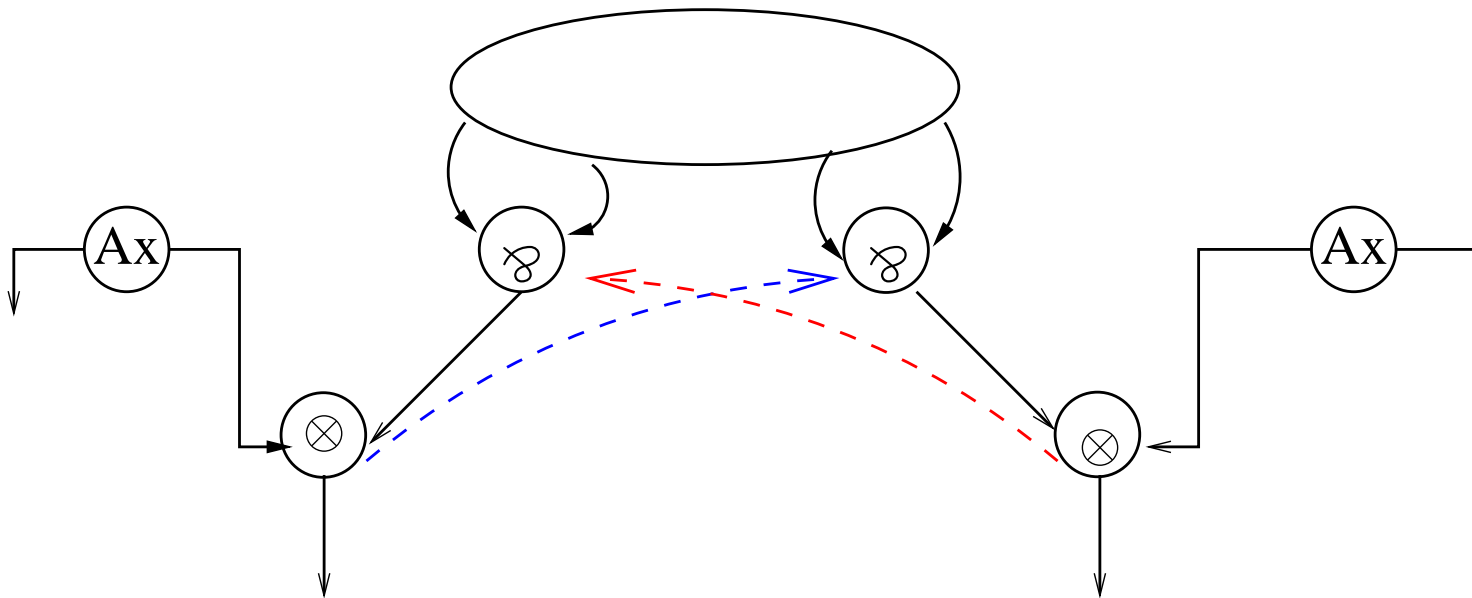
*a jumps to b = as a rule, b precedes a (bottom-up) in a sequent calculus proof
(dependency relation).*



Sequentialising with jump (4)

Jump (Girard): untyped oriented edge from one link a to another link b .

a jumps to $b =$ as a rule, b precedes a (bottom-up) in a sequent calculus proof
(dependency relation).



- Once put all the possible jumps on a proof net, can we directly retrieve a sequentialisation?

What kind of proof net an L-net is?

L-nets: parallel version of ludics designs (linear strategies)

	Graph	Games	Syntax
Designs	tree	set of views	proof of $MALL^{fo}$
L-nets	d.a.g	set of p.o. views	???

Partial sequentialisation/desequentialisation

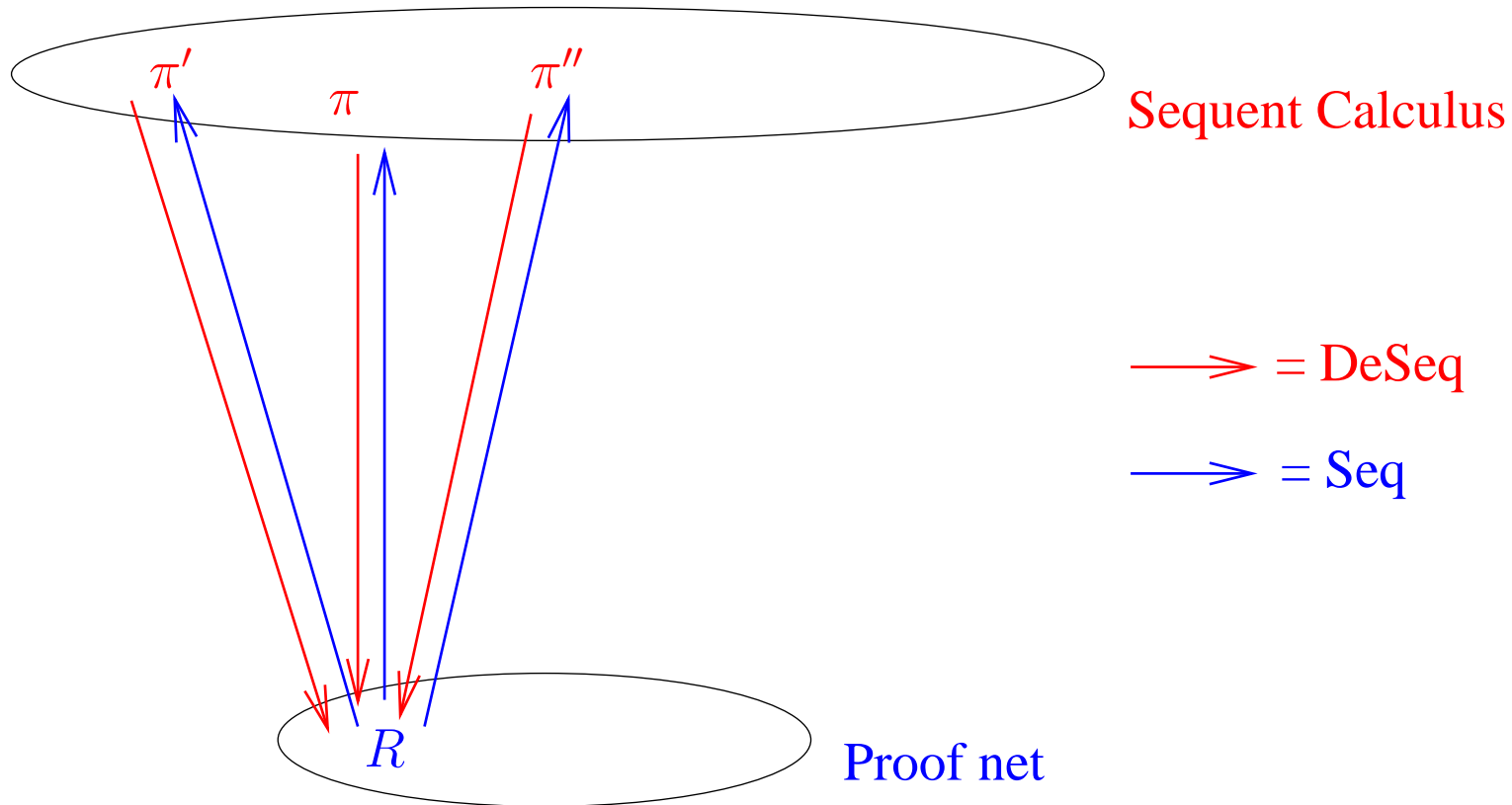
A sequentialisation (resp. desequentialisation) procedure usually inserts (resp. removes) sequentiality up to a maximum:

proof π $DeSeq(\pi) \rightsquigarrow$ proof net R (Many to one)

proof net R $Seq(R) \rightsquigarrow$ proof π (One to many)

Is it possible to add (resp. remove) sequentiality *à la carte*?

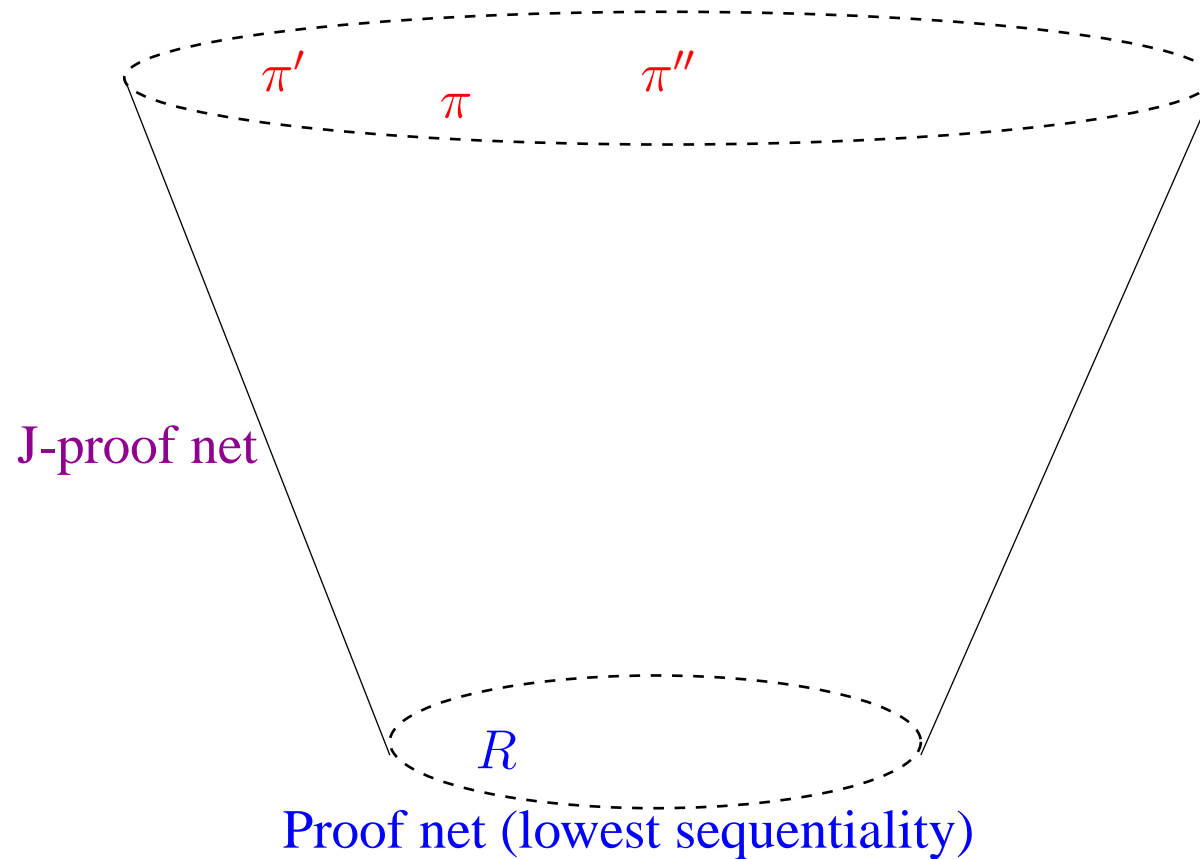
From proof nets...



- From proof net (high level of parallelism) to sequent calculus (high level of sequentiality) and back.
- Is there something in between?

...to J-proof-nets!

Sequent Calculus (highest sequentiality)



- Objects with different degrees of sequentiality, in the same setting.
- Simple way to approach sequentialisation.

Focalization

Distinction of Linear Logic connectives into

Positives: $\otimes, \oplus, 1, 0$.

Negative: $\wp, \&, \perp, \top$.

A sequent calculus proof is called *focusing* if its bottom-up construction satisfies the following discipline:

Negative first : first keep decomposing a negative formula (if any) and its subformulas, until one get to atoms or positive subformulas;

Choose a positive : choose a positive formula, and keep decomposing it up to atoms or negative subformulas.

Focalization (Andreoli):

Any *LL* proof π can be transformed, by solely permutation of rules, into a focussing proof π^{foc}

Multiplicatives

Multiplicative Hypersequentialised Calculus (MHS)

Hypersequentialised Calculus (Girard): *syntethic connectives (n-ary) + polarity alternation* (“Normal form” of focussing proofs).

MHS *formulas* (polarisation of the atoms: X denotes a positive atom)

$$N ::= X^\perp \quad | \quad \wp(P, \dots, P)$$

$$P ::= X \quad | \quad \otimes(N, \dots, N)$$

MHS *rules* (at most one negative formula):

$$\frac{}{\vdash X, X^\perp} (Ax)$$

$$\frac{\vdash \Gamma \quad \vdash \Delta}{\vdash \Gamma, \Delta} (Mix)$$

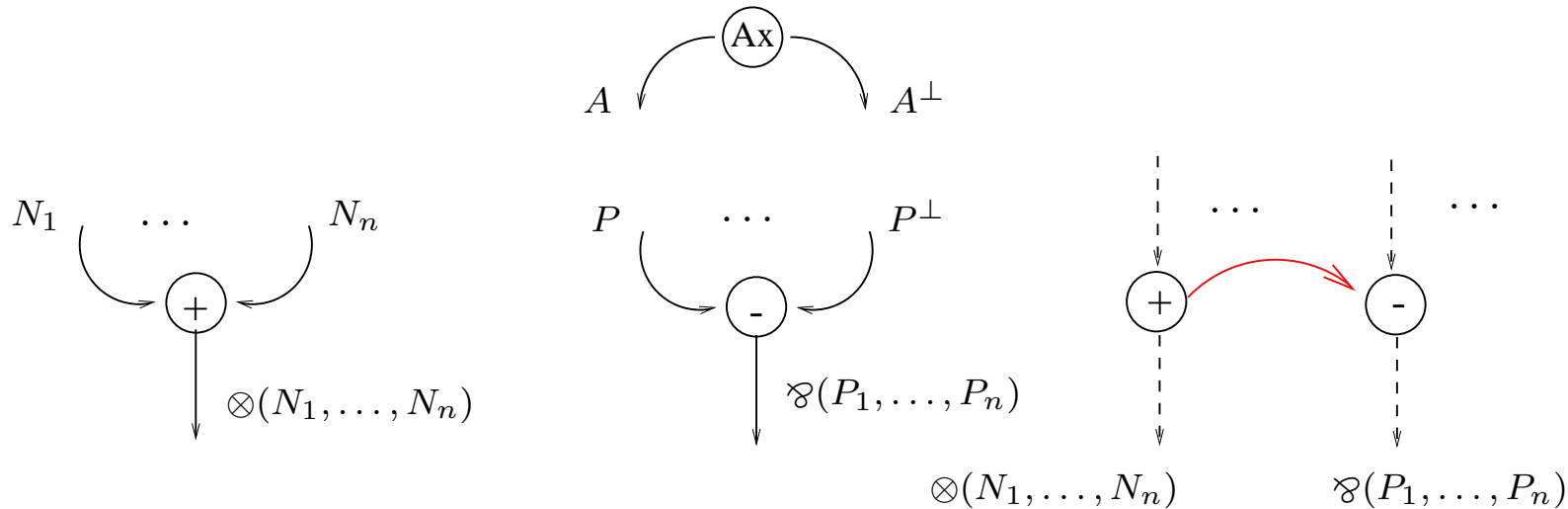
$$\frac{\vdash \Gamma_1, N_1 \dots \vdash \Gamma_n, N_n}{\vdash \Gamma_1, \dots, \Gamma_n, \otimes(N_1, \dots, N_n)} (+)$$

$$\frac{\vdash \Gamma, P_1, \dots, P_n}{\vdash \Gamma, \wp(P_1, \dots, P_n)} (-)$$

Unary \wp (resp. \otimes) = polarity inverter, \uparrow (resp. \downarrow).

J-proof net

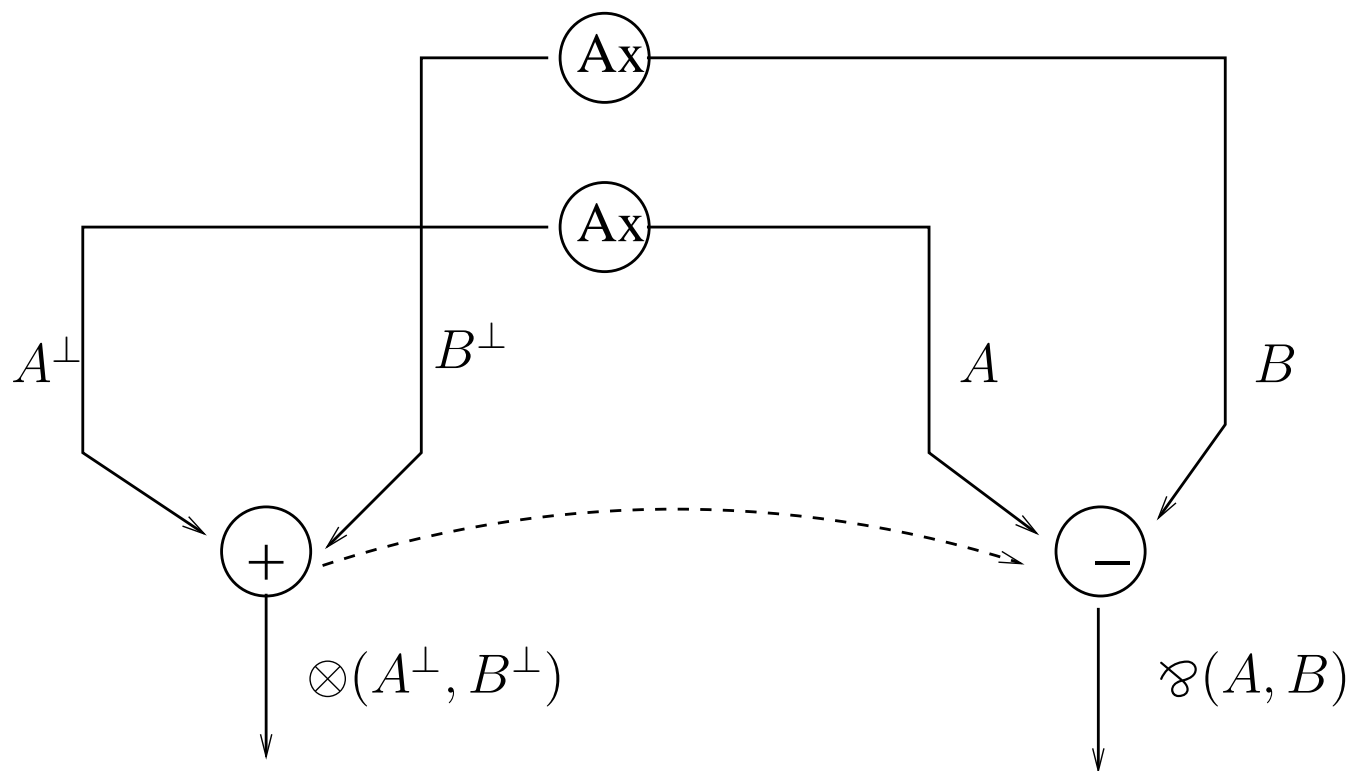
J-proof structure



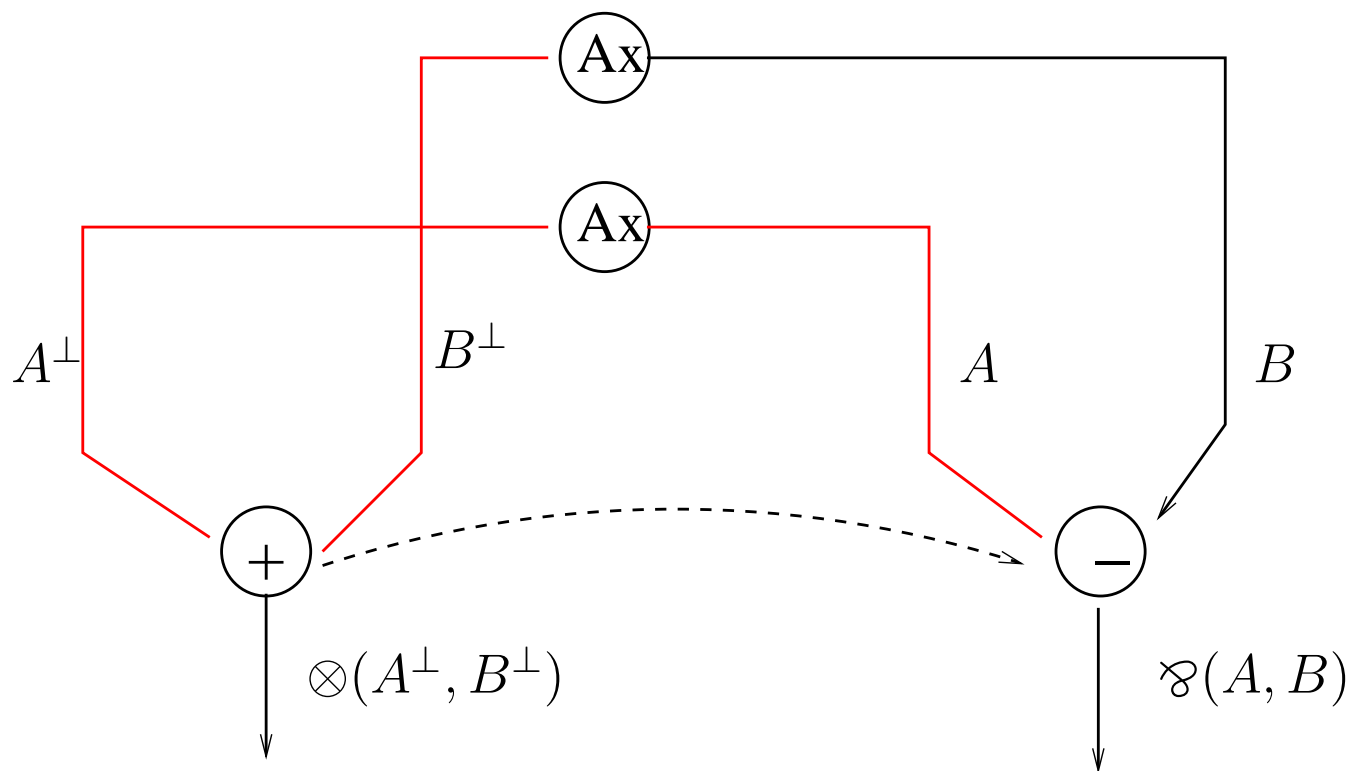
At most one negative terminal link

- A *switching path* is an unoriented path which never uses two incident edges of the same negative link (including jumps).
- A *J-proof net* is a J-proof structure which has no switching cycles.

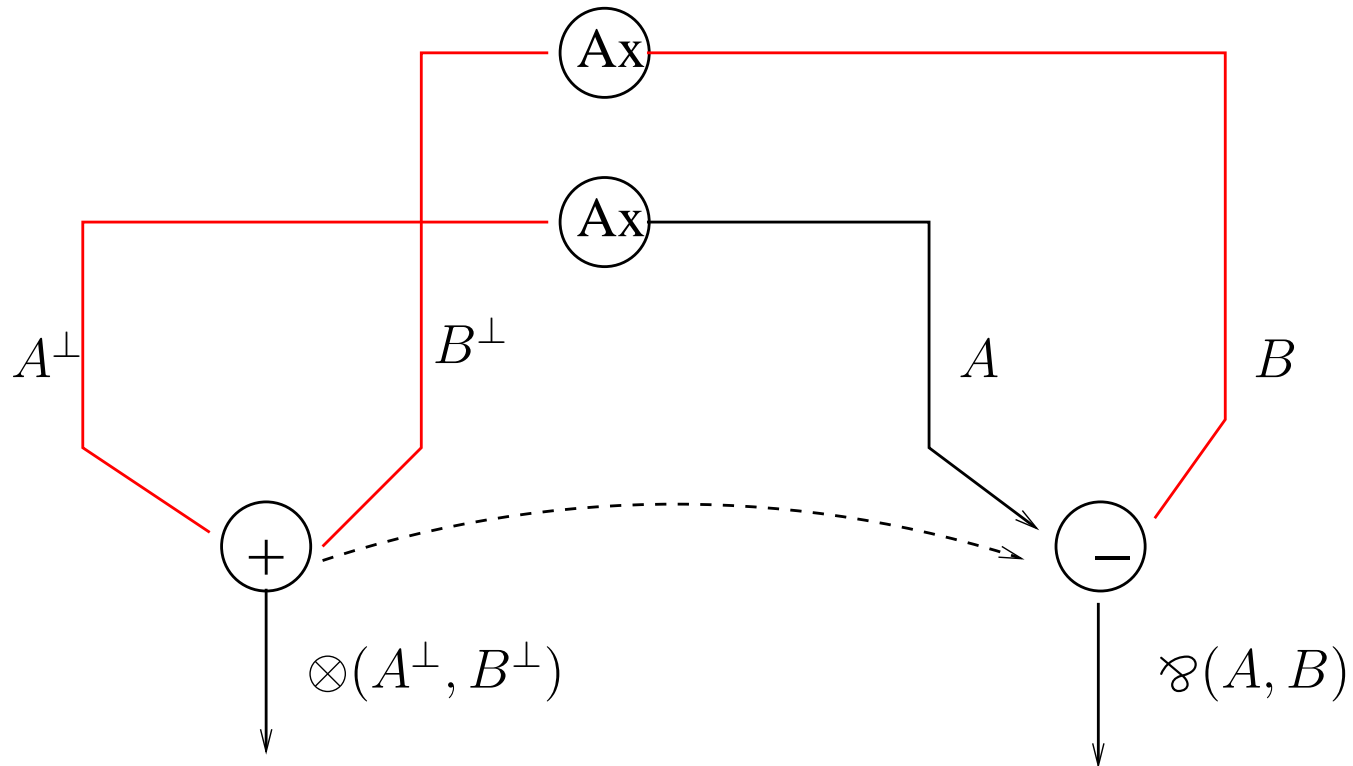
J-proof net: example (1)



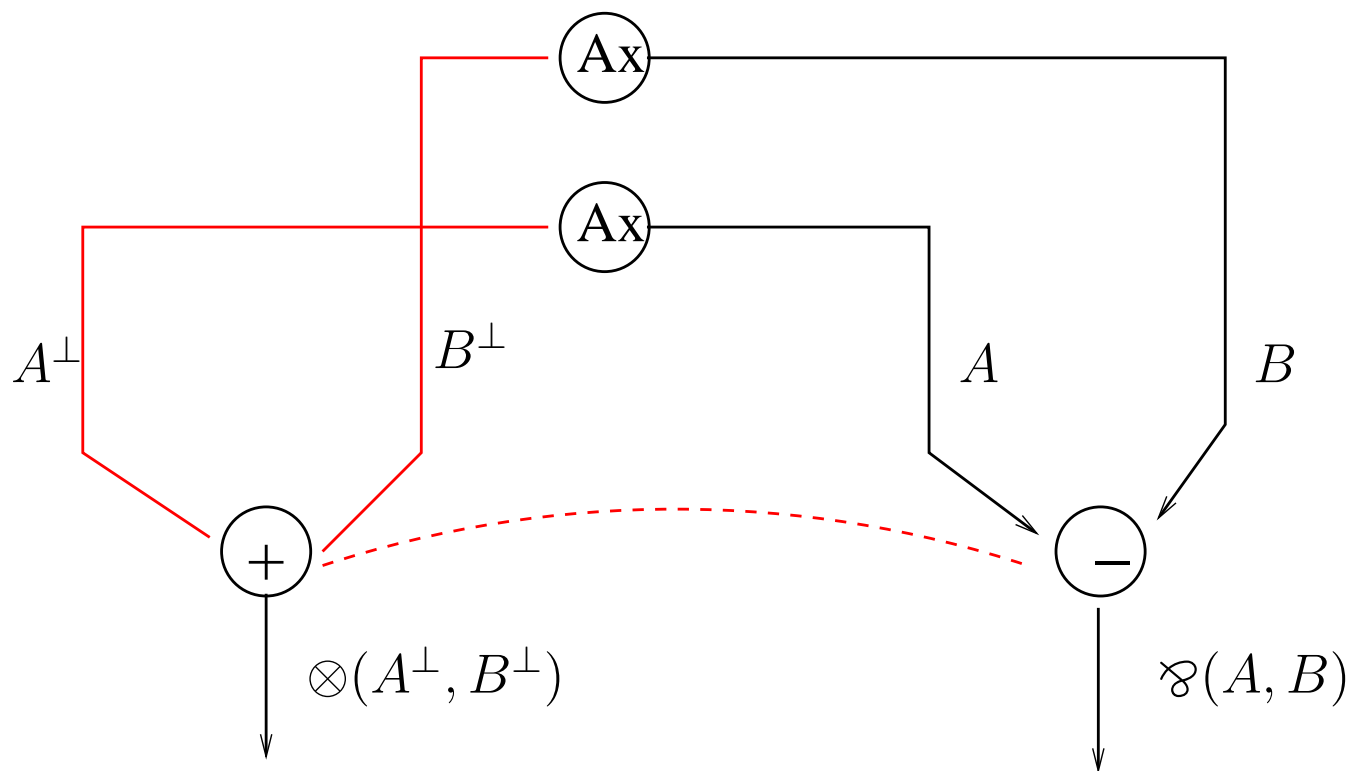
J-proof net: example (2)



J-proof net: example (3)



J-proof net: example (4)



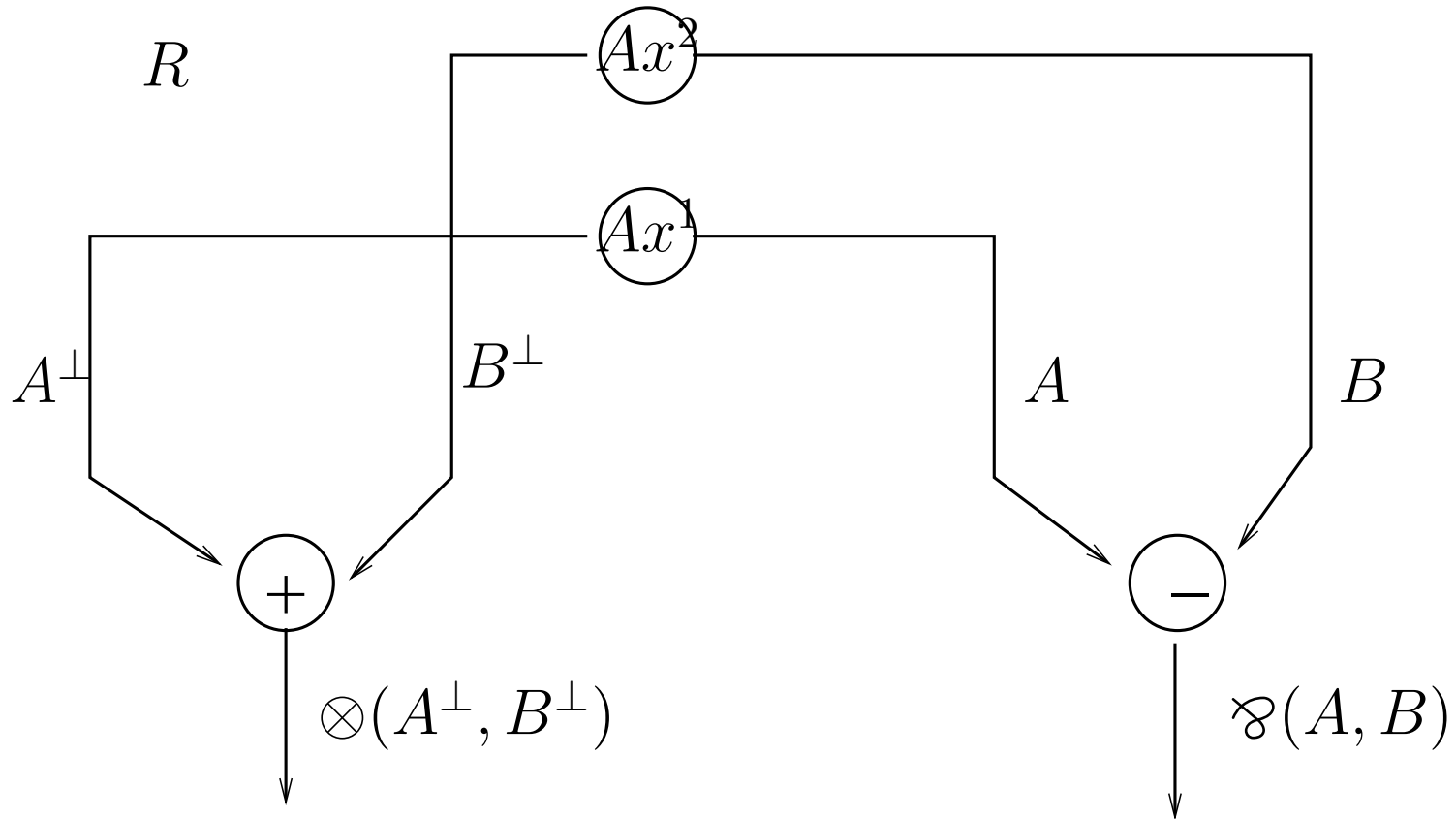
Characterizing sequentiality: partial order associated with a J-proof net

J-proof net R : directed acyclic graph (d.a.g.),

Order \prec_R : strict partial order on the nodes of R induced by the transitive closure of R as a d.a.g ($a \prec_R b$ iff there exists a directed path in R from b to a).

Skeleton of R ($Sk(R)$) : the smallest graph whose transitive closure is the same as that of R (R and $Sk(R)$ induce the same order).

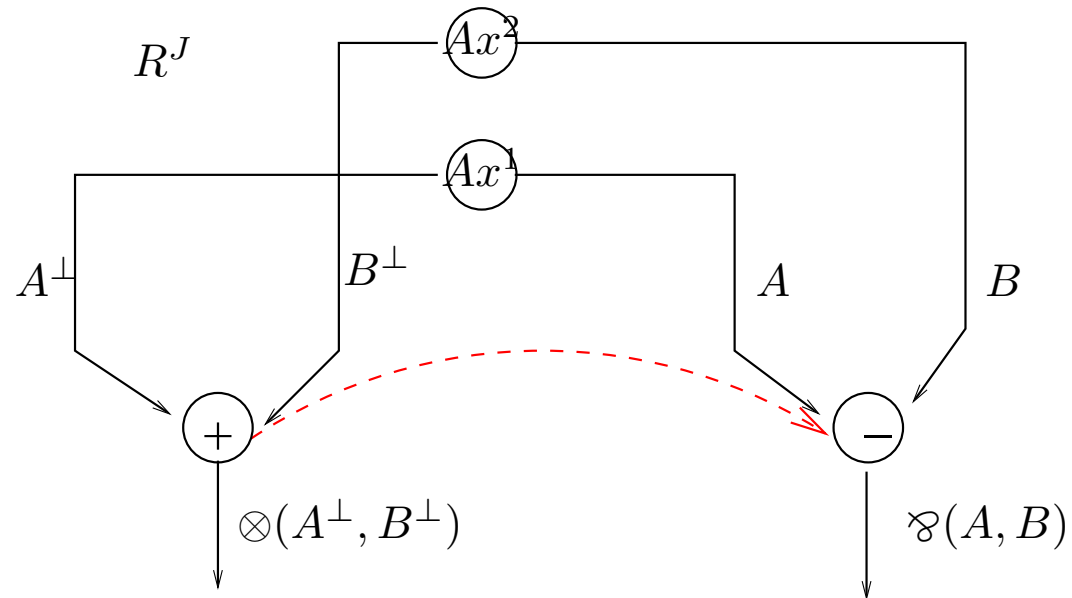
J-proof net and Sequent Calculus (1)



$$- \langle Ax^1 \rangle +$$

$$- \langle Ax^2 \rangle +$$

J-proof net and Sequent Calculus (2)



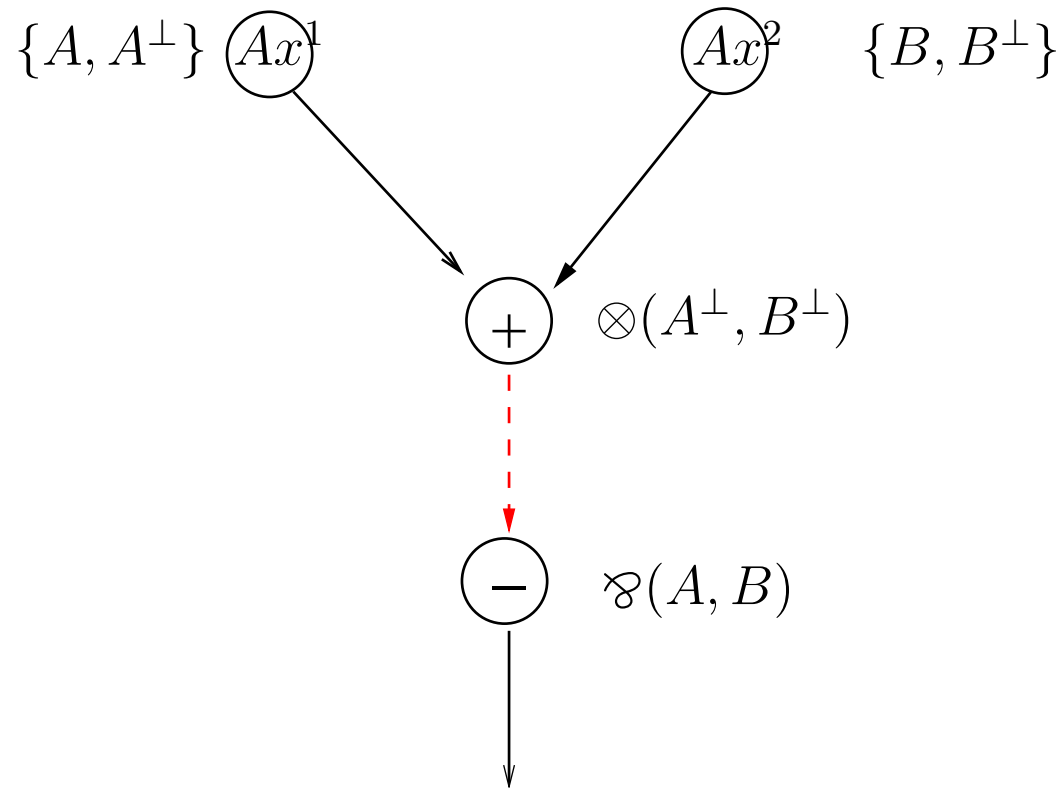
Arborescence : $\forall a, b$ if $\exists c$ s.t. $a < c \wedge b < c$ then $a < b \vee b < a$

$$- < + < Ax^1$$

$$- < + < Ax^2$$

J-proof net and Sequent Calculus (3)

$Sk(R^J)$



J-proof net and Sequent Calculus (4)

$$\frac{\frac{\overline{\vdash A, A^\perp} Ax^1 \quad \overline{\vdash B, B^\perp} Ax^2}{\vdash \otimes(A^\perp, B^\perp), A, B} (+)}{\vdash \otimes(A^\perp, B^\perp), \wp(A, B)} (-)$$

If \prec_R is arborescent, then we can directly associate with R a sequent calculus proof π^R

J-proof net sequentialisation

Definition (Saturated J-proof net). *A J-proof net R is saturated if for every negative link n and for every positive link p adding a jump between n and p creates a switching cycle or doesn't increase the order \prec_R .*

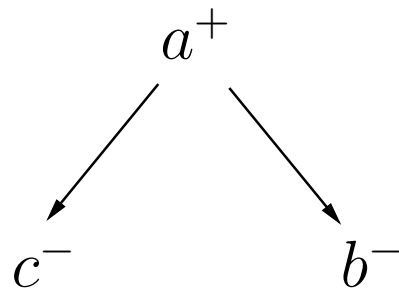
Our sequentialization argument is as follows:

- Any J-proof net can be saturated.
- **The order associated with a saturated J-proof net is arborescent.**
- If the order \prec_R associated with a J-proof net R is arborescent, we can associate with R a proof π^R in the sequent calculus.

Lemma (Arborisation). *Let R be a J-proof net. If R is saturated then \prec_R is arborescent.*

Proof of the arborisation lemma (1)

Proof by contraposition: \prec_R not arborescent $\implies R$ not saturated.



- b^- , c^- cannot be both conclusions.

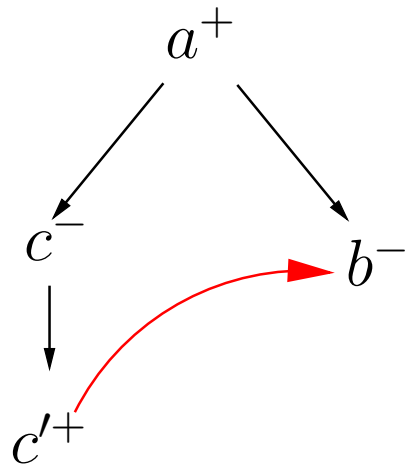
Proof of the arborisation lemma (2)

Two cases

1. b^- is a conclusion;
2. neither b^- and c^- are conclusions.

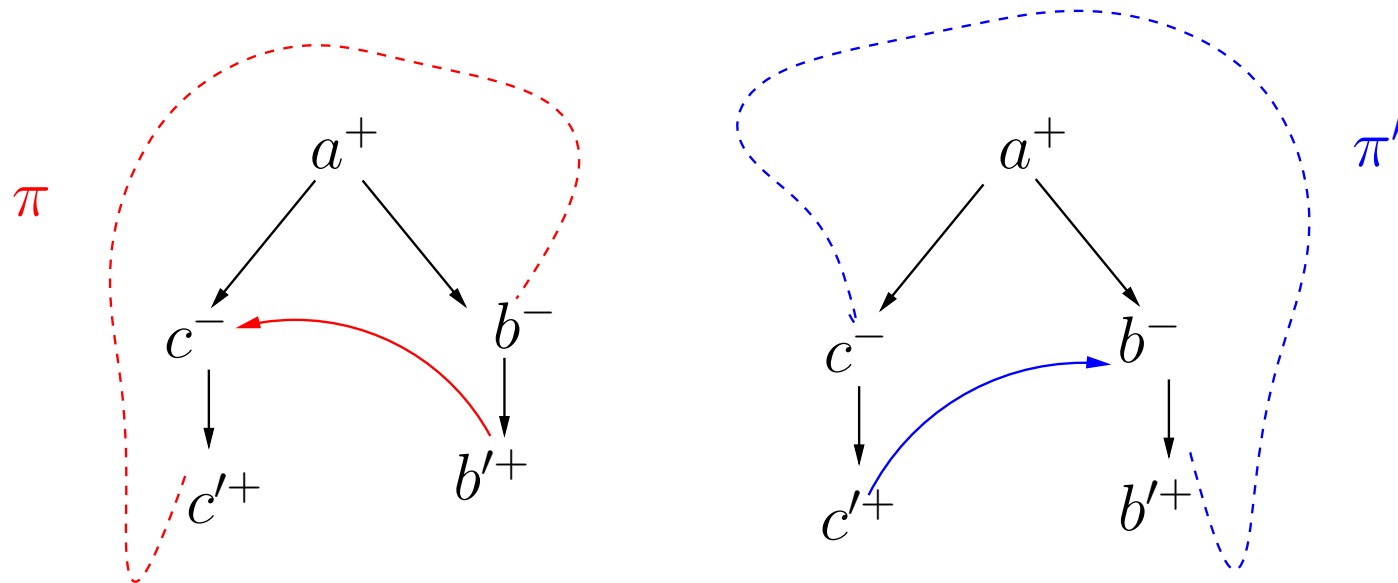
Proof of the arborisation lemma (3)

- b^- is a conclusion:

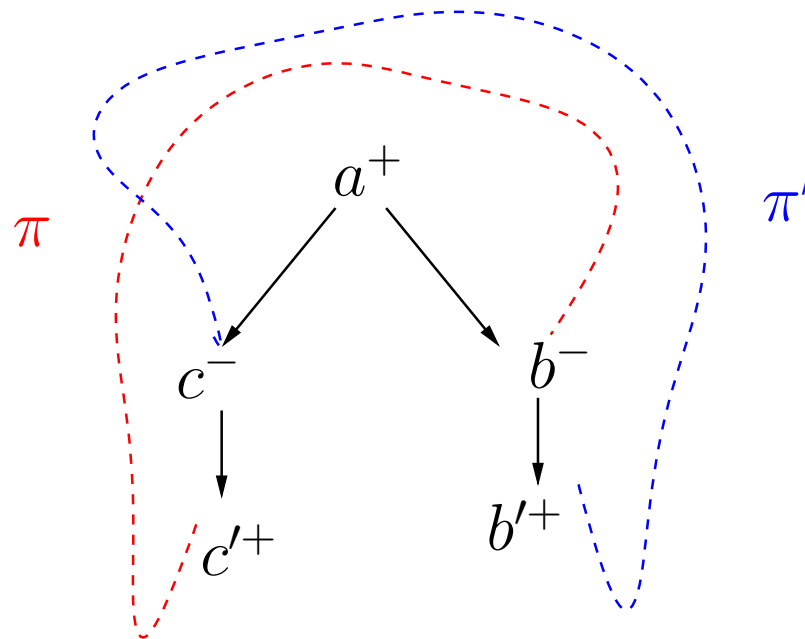


Proof of the arborisation lemma (4)

- neither b^- and c^- are conclusions; we reason by absurd:



Proof of the arborisation lemma (5)



Additives

Hypersequentialised Calculus (HS)

HS *formulas* (I : index set, \mathcal{N} : set of index set)

$$N ::= X^\perp \mid \&_{I \in \mathcal{N}} (\wp_{i \in I} (\uparrow P_i))$$

$$P ::= X \mid \oplus_{I \in \mathcal{N}} (\otimes_{i \in I} (\downarrow N_i))$$

HS *rules* (at most one negative formula):

$$\frac{}{\vdash X, X^\perp} (Ax)$$

$$\frac{\vdash \Gamma \quad \vdash \Delta}{\vdash \Gamma, \Delta} (Mix)$$

$$\frac{\vdash \Gamma_i, N_i \dots \vdash \Gamma_j, N_j}{\vdash \Gamma_i, \dots, \Gamma_j, \oplus_{I \in \mathcal{N}} (\otimes_{i \in I} (\downarrow N_i))} (+, I)$$

$$\frac{\vdash \Gamma, P_{j \in J} \dots \vdash \Gamma, P_{j' \in J'}}{\vdash \Gamma, \&_{I \in \mathcal{N}} (\wp_{i \in I} (\uparrow P_i))} (-, \mathcal{N})$$

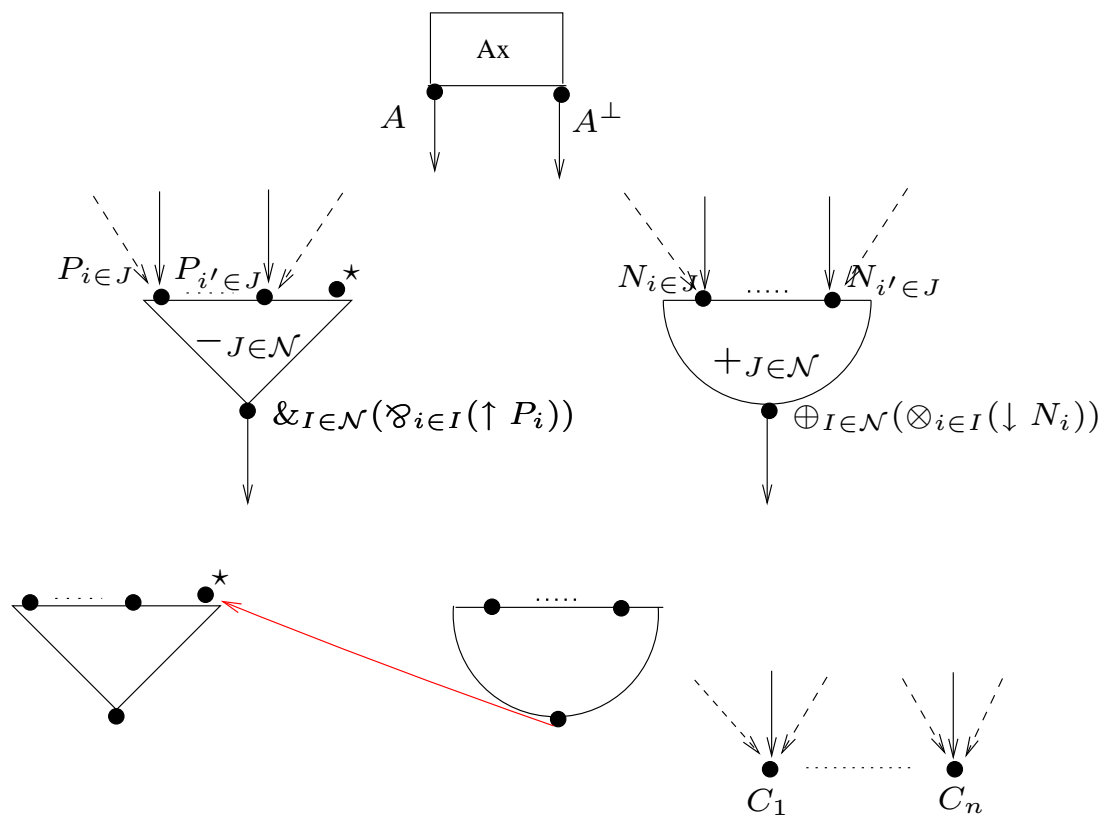
Decomposition of & rule

$$\frac{\vdash \Gamma, A \quad \vdash \Gamma, B}{\vdash \Gamma, A \& B} (A \& B) \rightsquigarrow \frac{\vdash \Gamma, A}{\vdash \Gamma, A \& B} (A \& B, A) \quad \text{and} \quad \frac{\vdash \Gamma, B}{\vdash \Gamma, A \& B} (A \& B, B)$$

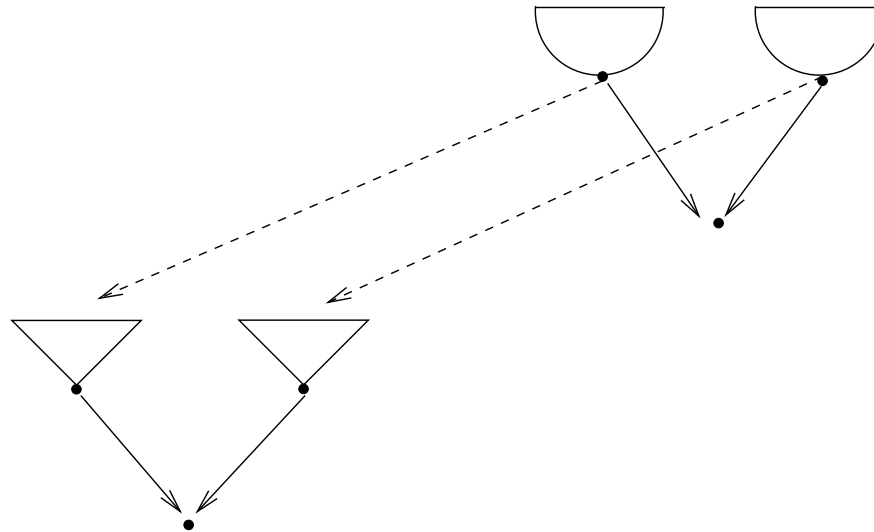
- & rule: *superposition* of rules (*slices*)

J-proof structure

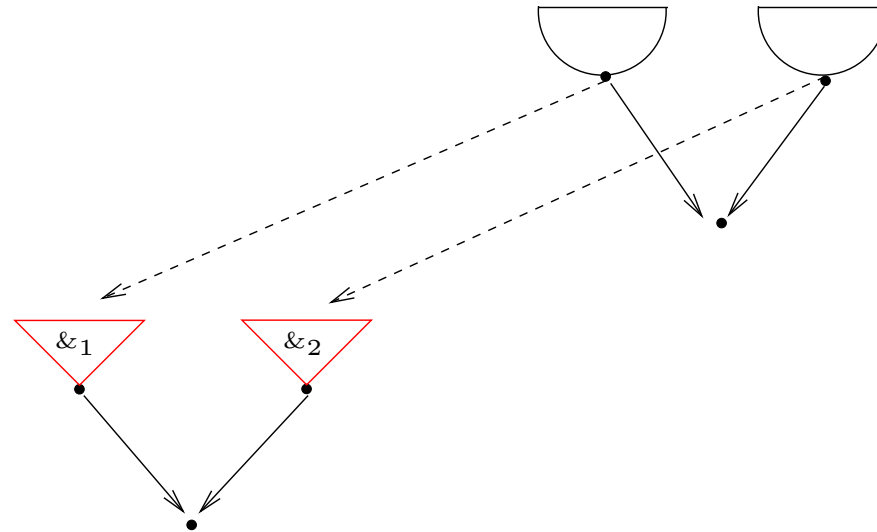
- Links enriched with *ports*
- **J-proof structure** : *links + edges + conclusion ports*



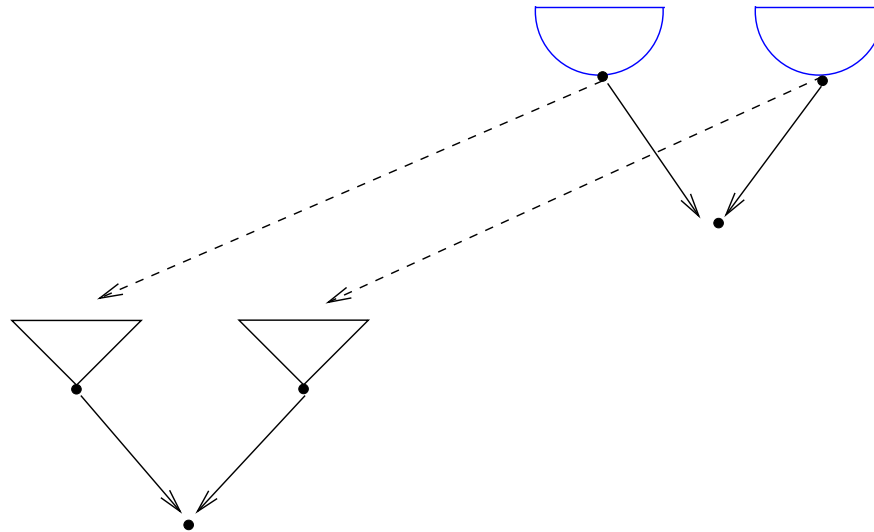
J-proof structure: sibling links



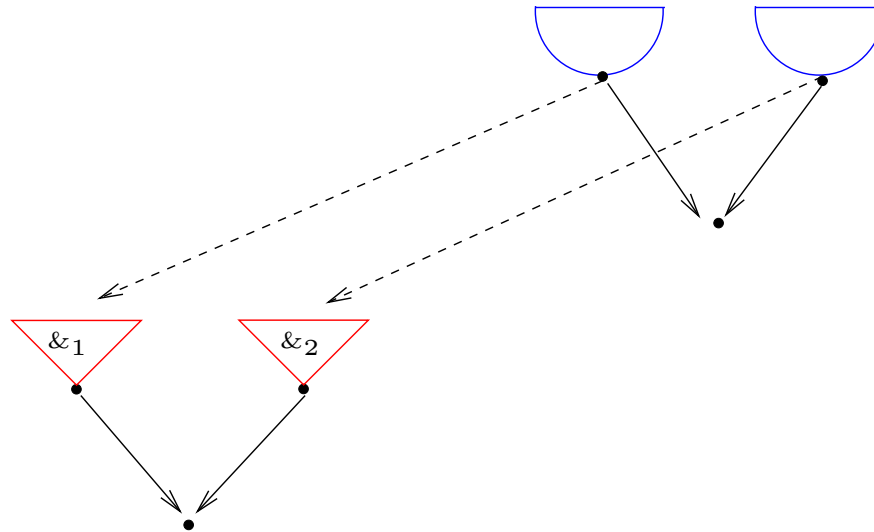
J-proof structure: negative siblings (additive rule)



J-proof structure: not negative siblings

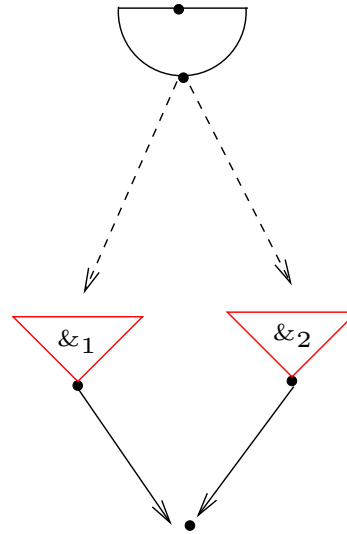


J-proof structure: not negative siblings



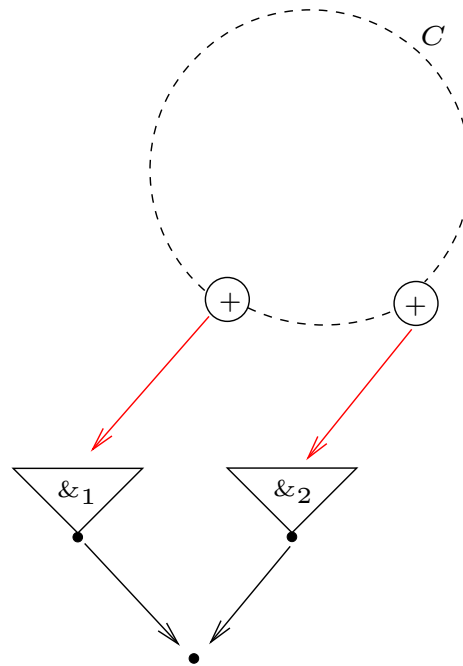
J-proof structure: views

NO!



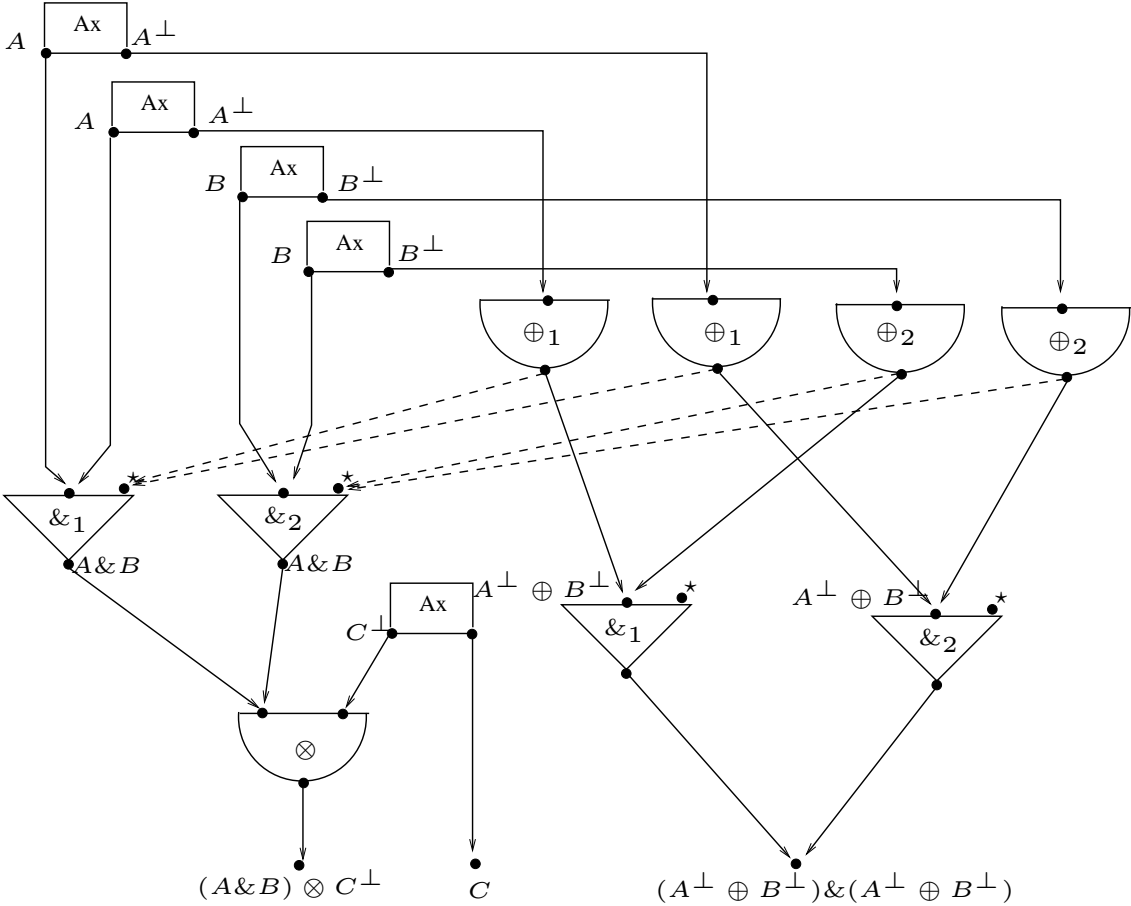
Correctness criterion

Definition (Correct J-proof-structure (Curien-Faggian)). A *J-proof structure* R is correct if and only if, given a non empty union C of switching cycles of R , there is an additive rule $W \in R$ not intersecting C and a pair $w_1, w_2 \in W$ such that for some links $c_1, c_2 \in C$, $w_1 \xleftarrow{+} c_1$ and $w_2 \xleftarrow{+} c_2$; in this case we say that the additive pair $w_1, w_2 \in W$ breaks C .



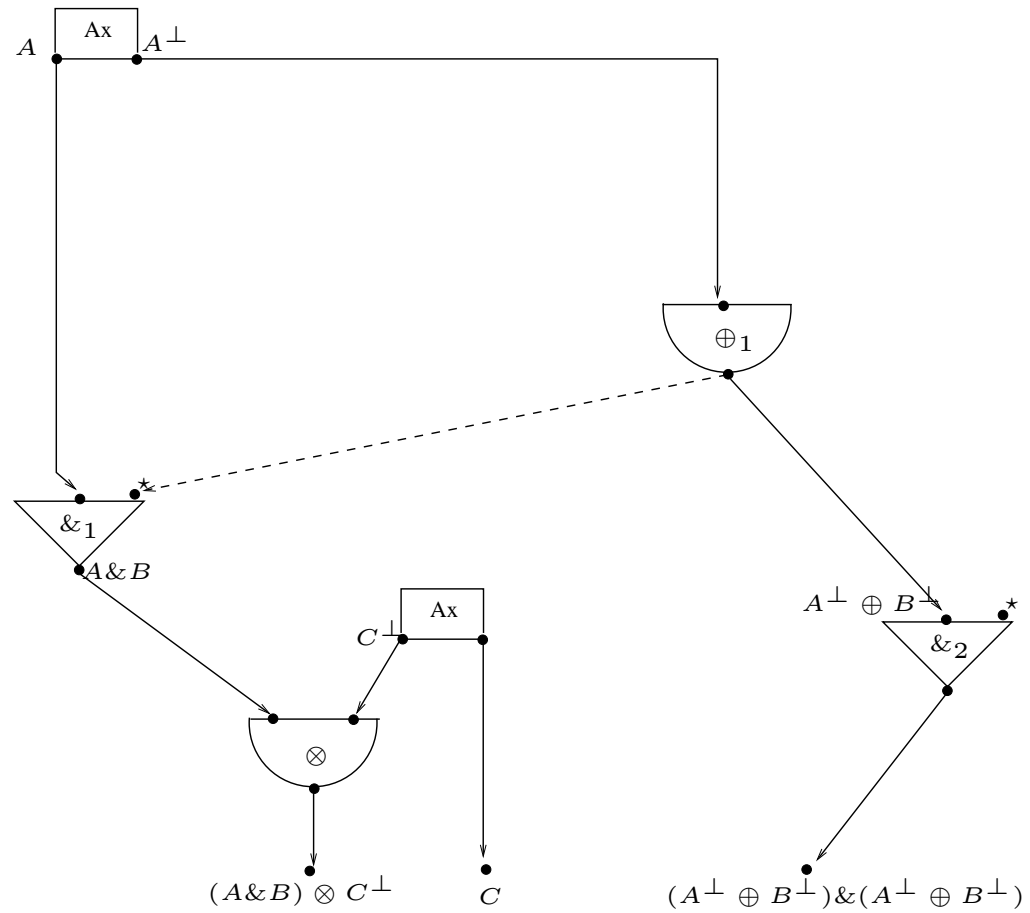
- Simplification of Hughes and Van Glabeek

Example: correct J-proof structure

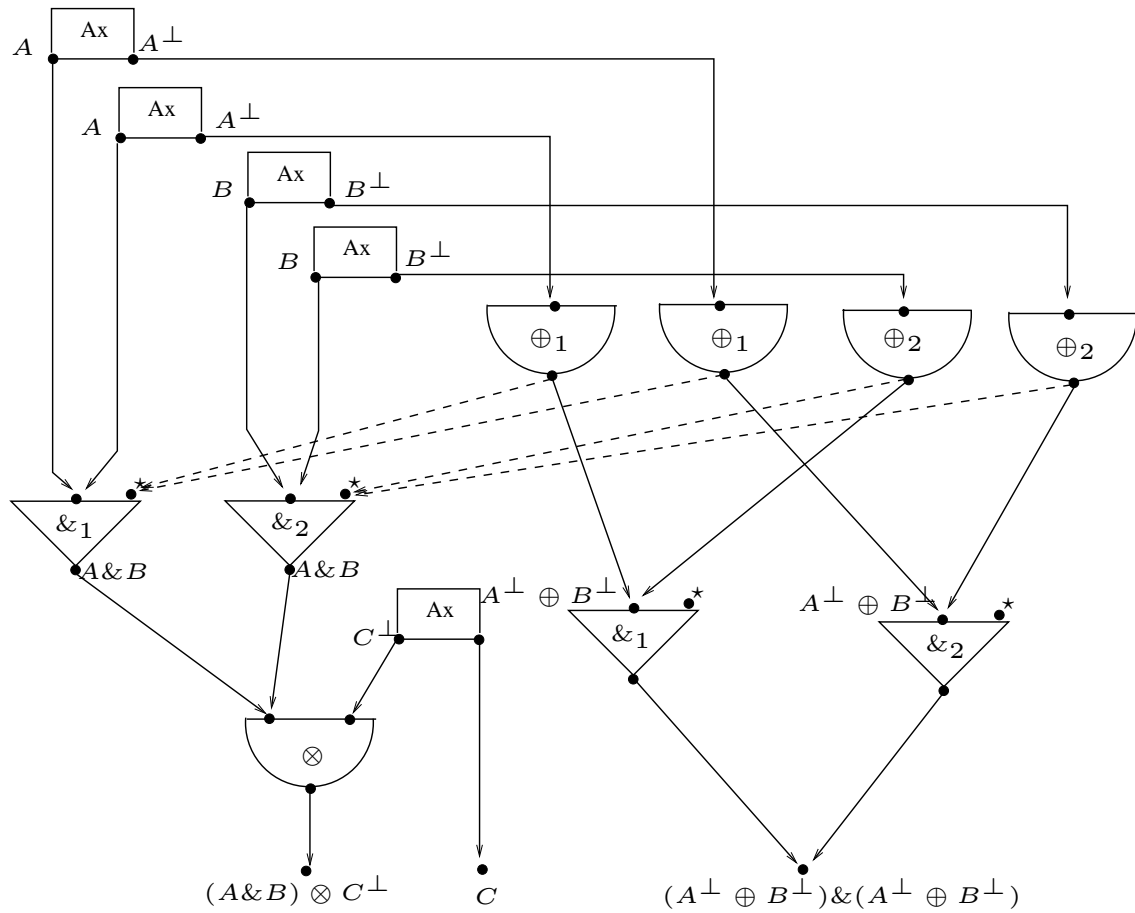


-We still need one ingredient: *totality*.

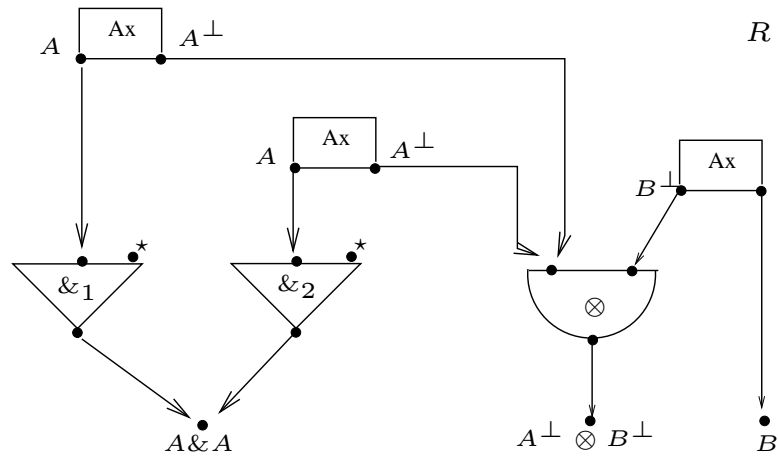
Totality: slices



J-proof net: correct and total J-proof structure

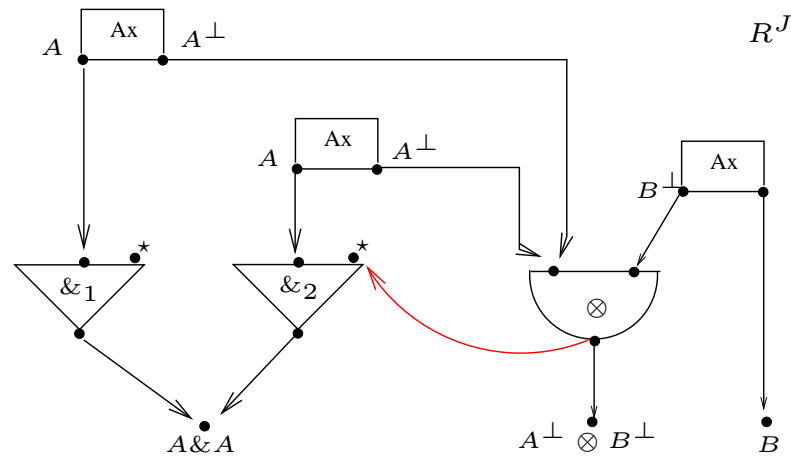


J-proof net and sequentialisation: the additive case



J-proof net and sequentialisation: the additive case

- A *correct jump* is a jump which doesn't violate the correctness criterion and increases the order.



Arborisation Lemma II

Lemma (Arborisation). *Let R be a J -proof net. If \prec_R is not arborescent, then we can still add a correct jump.*

Arborisation Lemma II

Lemma (Arborisation). *Let R be a J -proof net. If \prec_R is not arborescent, then we can still add a correct jump.*

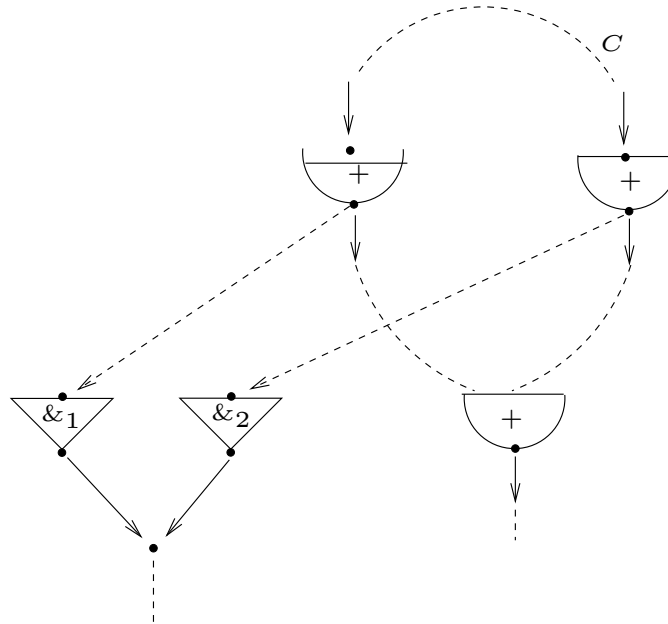
Lemma. *Let R be J -proof net; if R contains a cycle C , then we can still add a correct jump.*

Arborisation Lemma II

Lemma (Arborisation). *Let R be a J-proof net. If \prec_R is not arborescent, then we can still add a correct jump.*

Lemma. *Let R be J-proof net; if R contains a cycle C , then we can still add a correct jump.*

Proof. Let's suppose there is just a cycle in R :



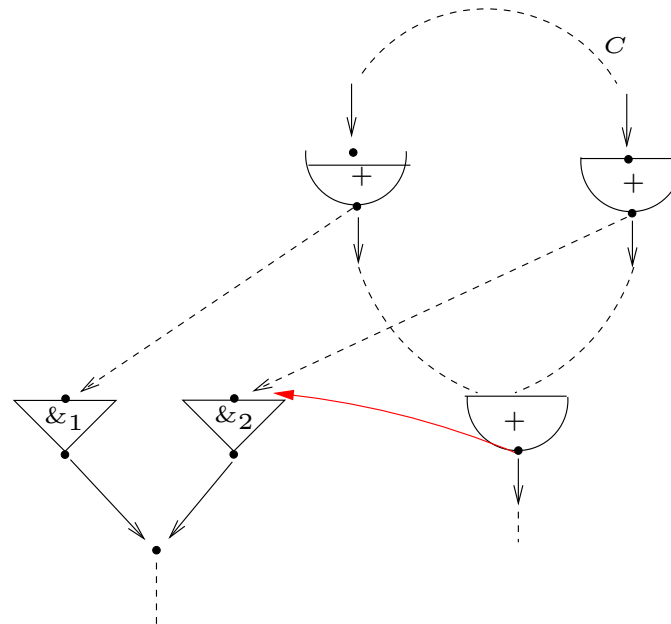
□

Arborisation Lemma II

Lemma (Arborisation). *Let R be a J-proof net. If \prec_R is not arborescent, then we can still add a correct jump.*

Lemma. *Let R be J-proof net; if R contains a cycle C , then we can still add a correct jump.*

Proof. Let's suppose there is just a cycle in R :



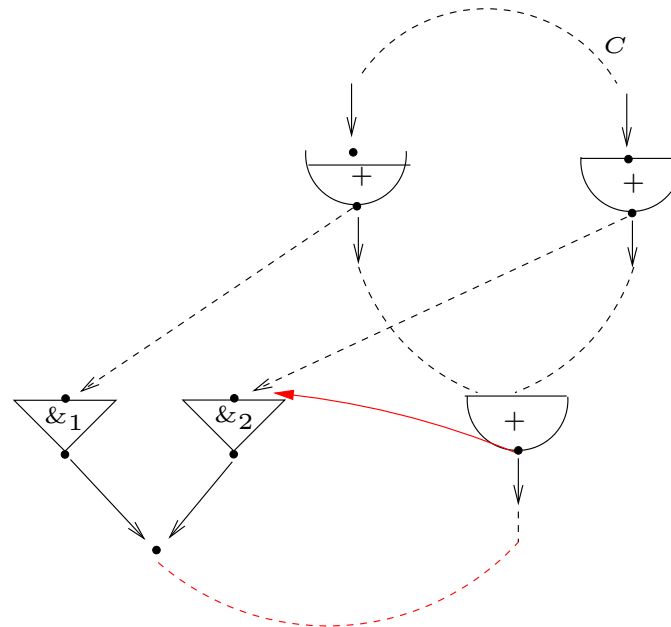
□

Arborisation Lemma II

Lemma (Arborisation). *Let R be a J-proof net. If \prec_R is not arborescent, then we can still add a correct jump.*

Lemma. *Let R be J-proof net; if R contains a cycle C , then we can still add a correct jump.*

Proof. Let's suppose there is just a cycle in R :



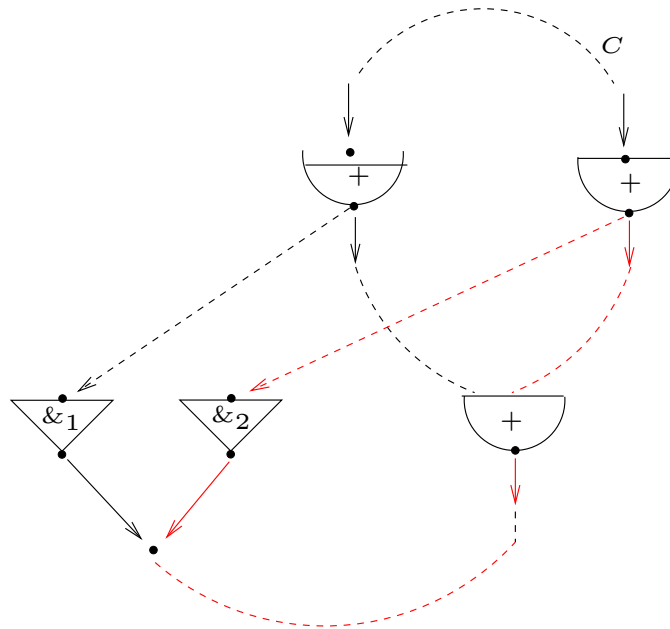
□

Arborisation Lemma II

Lemma (Arborisation). *Let R be a J-proof net. If \prec_R is not arborescent, then we can still add a correct jump.*

Lemma. *Let R be J-proof net; if R contains a cycle C , then we can still add a correct jump.*

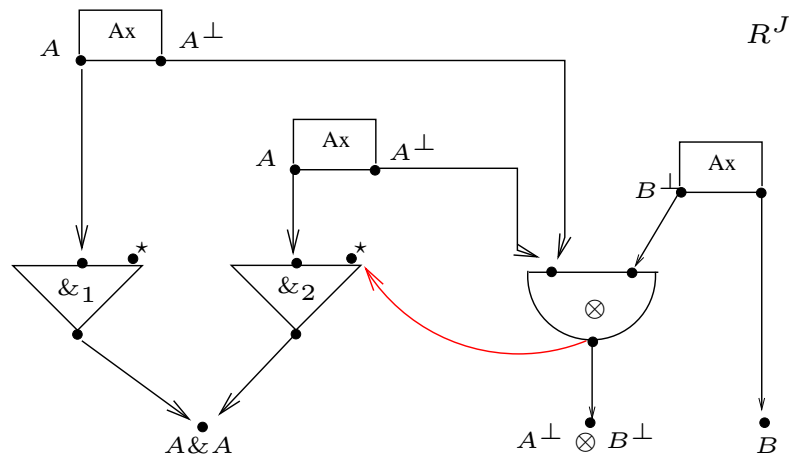
Proof. Let's suppose there is just a cycle in R :



□

J-proof net and sequentialisation: the additive case

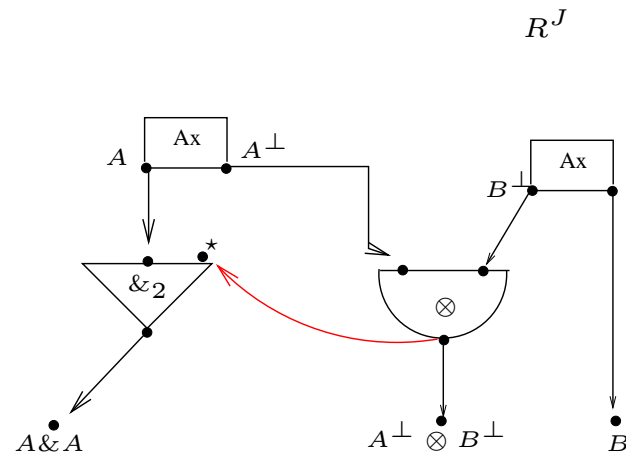
- A *correct jump* is a jump which doesn't violate the correctness criterion and increases the order.



- R^J is not total!

J-proof net and sequentialisation: the additive case

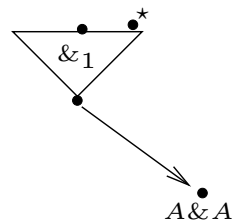
- A *correct jump* is a jump which doesn't violate the correctness criterion and increases the order.



- R^J is not total!

J-proof net and sequentialisation: the additive case

- A *correct jump* is a jump which doesn't violate the correctness criterion and increases the order.



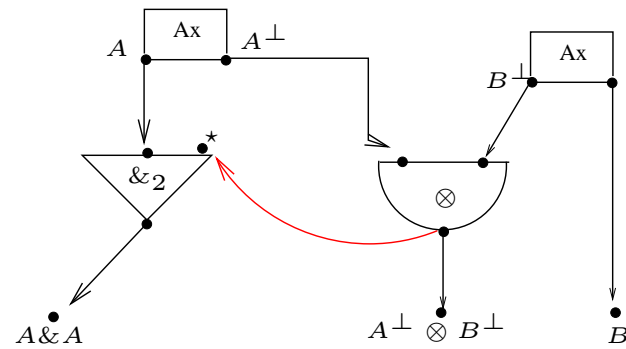
$$A^\perp \otimes B^\perp$$

R^J

- R^J is not total!

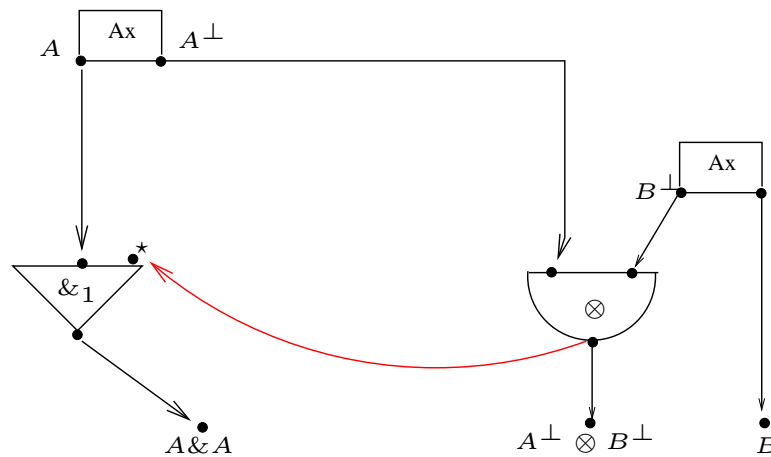
J-proof net and sequentialisation: the additive case

- Jumping in slices.



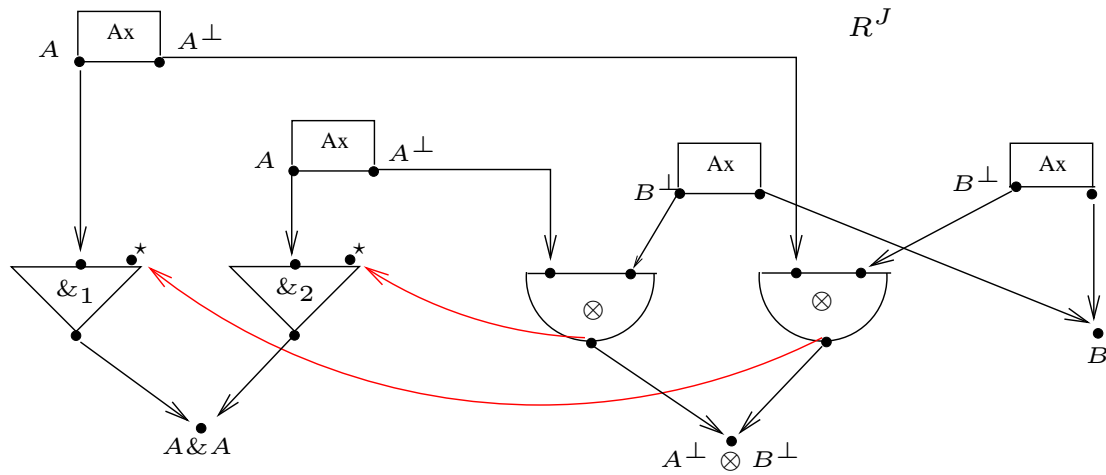
J-proof net and sequentialisation: the additive case

- Jumping in slices.



J-proof net and sequentialisation: the additive case

- We superpose slices using a notion of *sharing equivalence*.

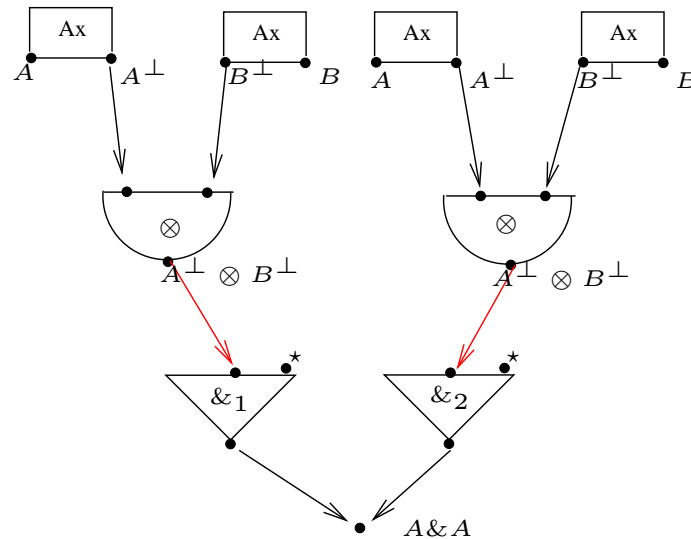


- R^J is total.

J-proof net and sequentialisation: the additive case

- We take $Sk(R^J)$:

$Sk(R^J)$



- If we can add a correct jump, we can also add a bundle of correct jumps which preserves totality.

J-proof net and sequentialisation: the additive case

To sequentialize a J- proof net R :

1. If \prec_R is not arborescent, we can add a correct jump;
2. if we can add a correct jump, we can add a bundle of correct jumps which preserves totality;
3. we add them to R , and iterate the procedure until the order becomes arborescent;
4. if the order is arborescent, we can directly associate to it an HS proof.

To sum up....

- Both parallel/sequential syntax for HS proofs;
- Simplification of sequentialisation theorem.

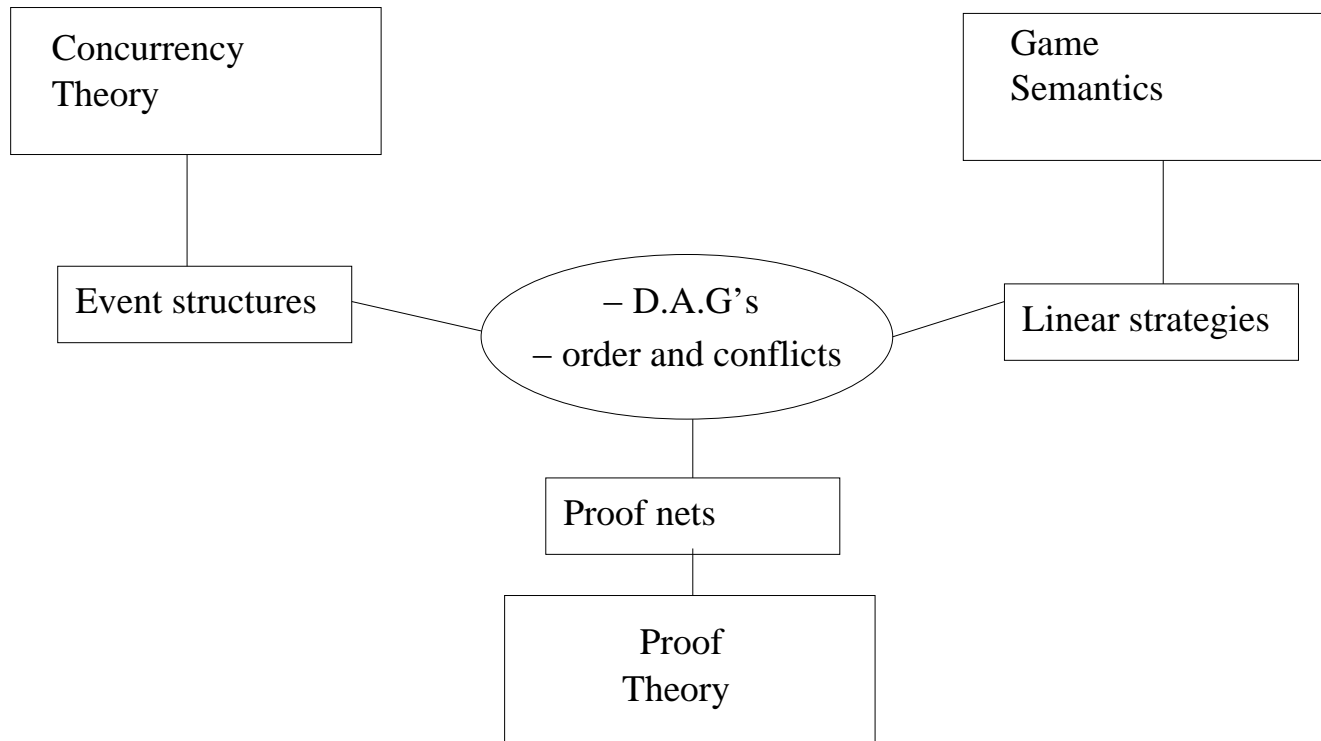
Things to do:

- cut-elimination by slices, as in L-nets;
- proving full completeness of L-nets w.r.t. J-proof nets;
- *naif* extension to exponentials by generalising axiom to *boxes*?

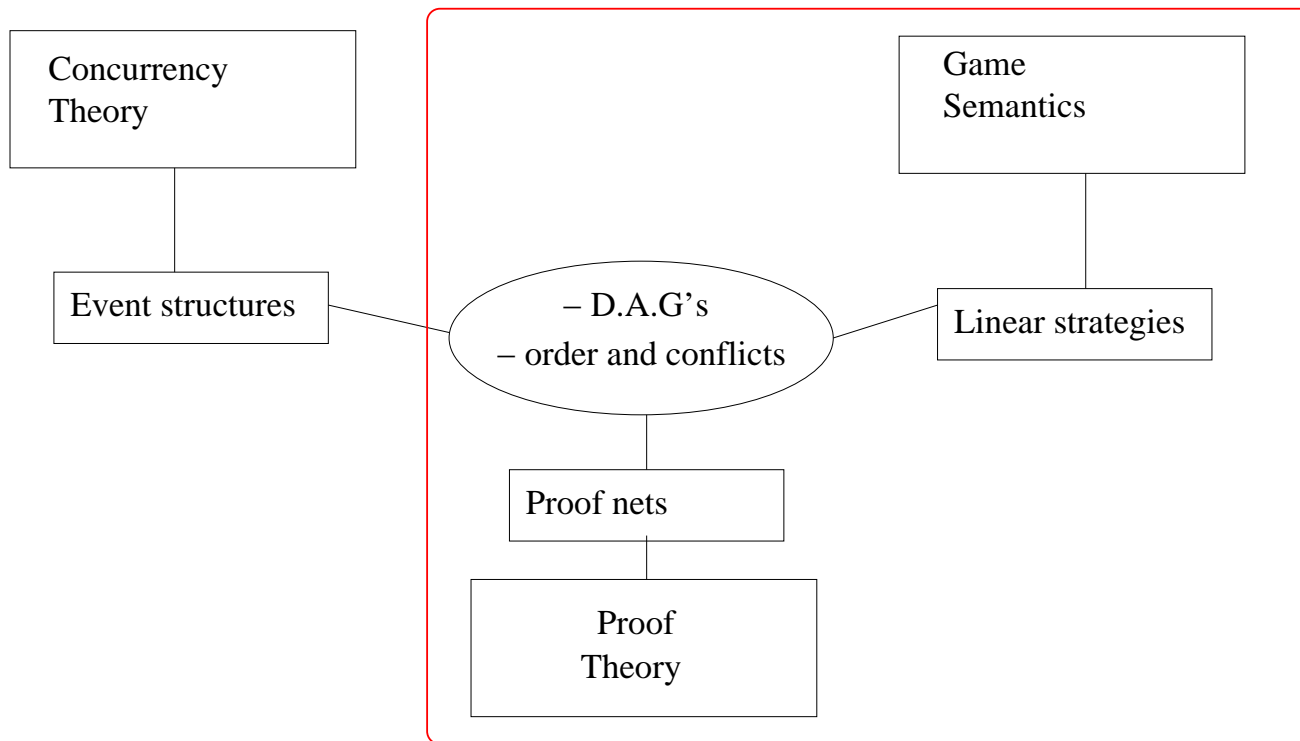
Questions:

- How much canonical are J-proof nets?
- Can we extend our sequentialisation proof to other syntaxes for *MALL* (HvG, proof nets with weights)?

General picture

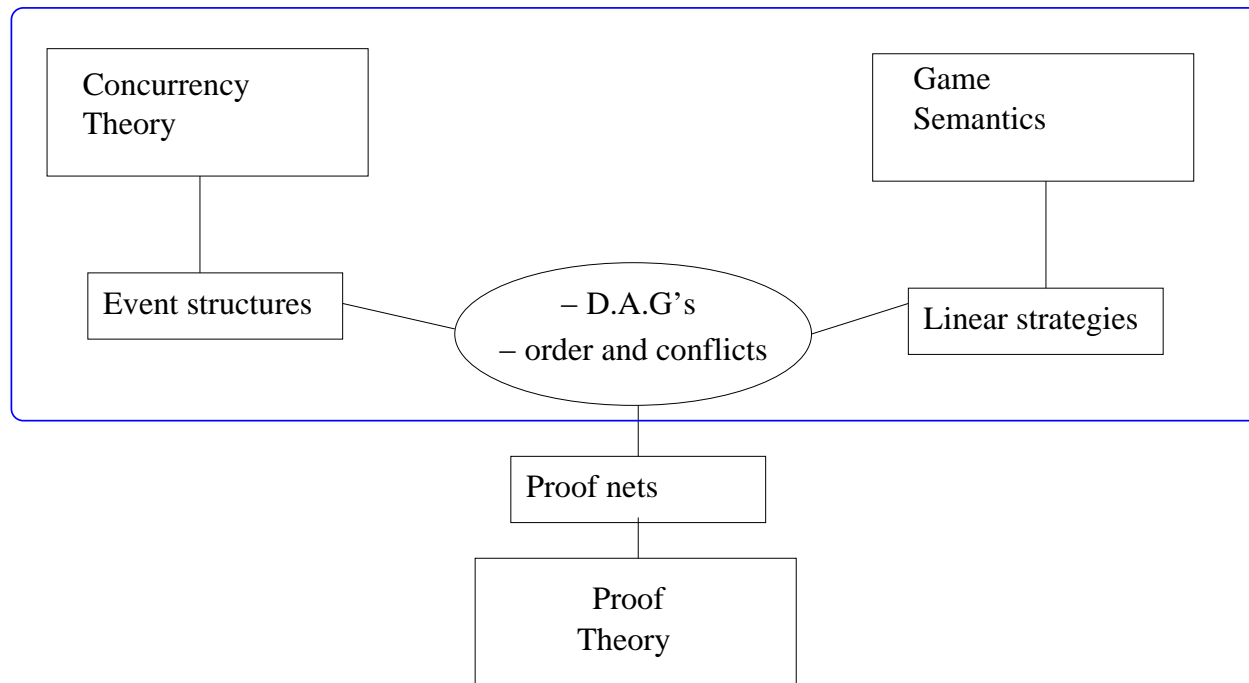


General picture

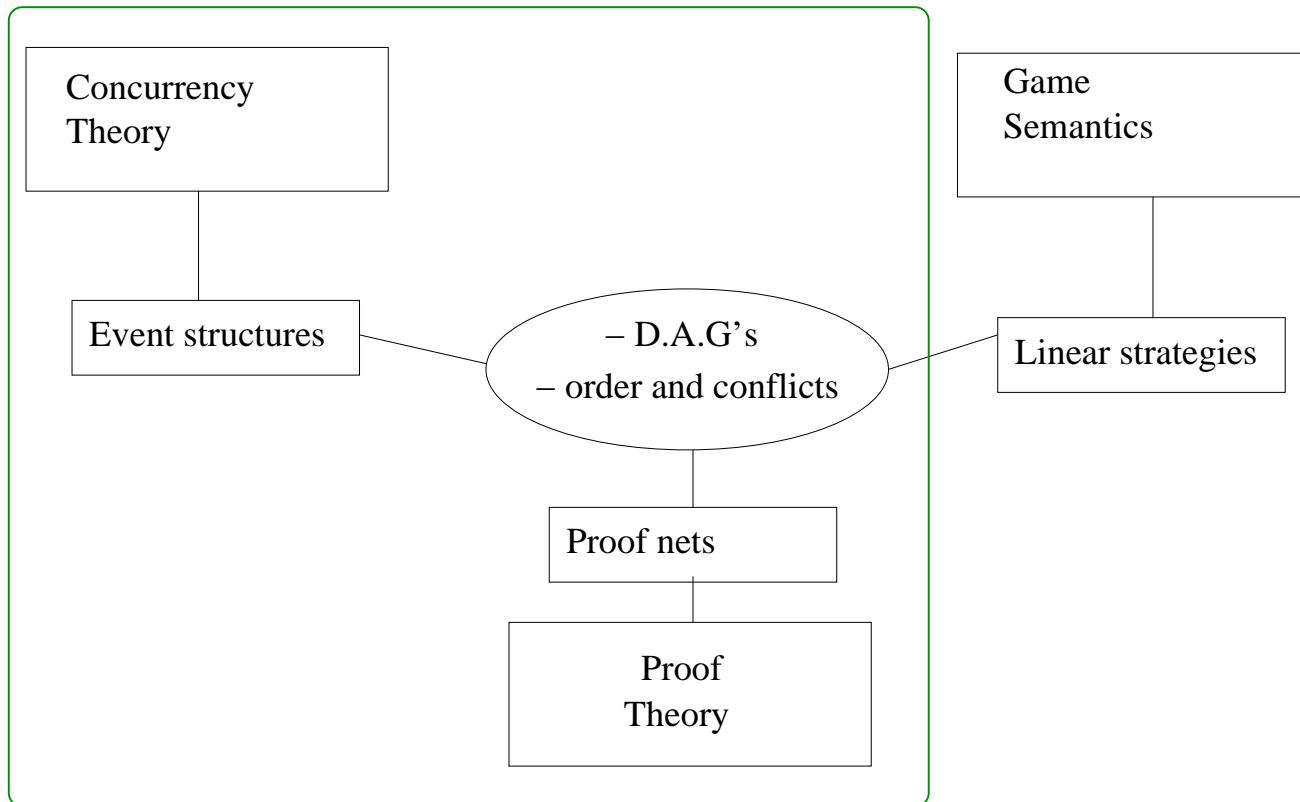


General picture

Faggian–Piccolo



General picture



Suggestions?