Cut Elimination for Monomial Proof Nets of the Purely Multiplicative and Additive Fragment of Linear Logic

Roberto Maieli

Cut Elimination for Monomial Proof Nets of the Purely Multiplicative and Additive Fragment of Linear Logic*

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

Revisited Version: April 28, 2015
...
Earlier Version: December 29, 2007

Abstract

We present a simple cut-elimination procedure for MALL proof nets with monomial weights (*à la Girard*) and explicit contraction links, based on an almost local cut reduction steps. This procedure preserves correctness of proof nets and it is strong normalizing and confluent.

Keywords : Proof Theory, Sequent Calculus, Cut Elimination, Proof Nets, Linear Logic.

Contents

*Partially supported by the CNR-CNRS Research Programme *Interaction Complexity* and the MIUR Research Programme *Rete Italo-Francese di Ricerca in Logica e Geometria della Computazione*. The author thanks Olivier Laurent and Paul Ruet for their remarks and suggestions.

1 Introduction

Proof Nets (PNs) are *parallel presentations* of sequential proofs (SP) of Linear Logic; they *quotient classes of equivalent proofs*, modulo irrelevant permutations of derivation rules. The standard key ingredients of a PN syntax are

- a graph syntax, i.e., *proof structures* (PSs);
- a *correctness criterion* defining *PNs* among PSs;
- an *interpretation* of the sequent calculus proofs;
- a *cut elimination* procedure

The **correctness criterion** should be:

- *geometrical*: an intrinsic (non-inductive) characterisation of those PS that *sequentialise* to SP (they are PN);
- *stable*: under cut elimination;
- *efficient*: *checking correctness* and *sequentialization* should be *P-time* with respect to the number of nodes.
- *Sequentializable*: (P-time), i.e., each PN must be the image of at least one SP.

The **interpretation** (translation) of SP into PS should be:

- *sound*: the PS associated to a SP, must be *correct* (a PN);
- *function*: $SP \mapsto PN$;
- *canonical surjection*: SP equal up to (reasonable) commutations of rules must be identified upon translation to a PN;
- *efficient*: P-time in the size (of the proofs).
  - we should preserve the computational complexity of the interpreted proofs;
  - we should respect the notion that a *semantics* (PN) is a *structure-preserving map* or some kind of *homomorphism* from proofs.

The **cut elimination** procedure should be:

- *defined* directly on PS;
- *complete*: any cut node of a PS, reduces in one step;
- *local*: a cut elimination step only affects the nodes (immediately) connected to the reducing cut node;
- *strong normalising*: terminating and (locally) confluent;
- *efficient*: P-time in size.

1.1 Multiplicative-Additive fragment of LL

Sequent Calculus – *Formulas* $A, B, ...$ are built from *literals* by the binary connectives $\otimes$ (*tensor*), $⅋$ (*par*), $\&$ (*with*) and $\oplus$ (*plus*). *Negation* $(.)^\perp$ extends to any formula by de Morgan laws:

$$\begin{array}{ll} (A \otimes B)^\perp = (B^\perp ⅋ A^\perp) & \quad (A ⅋ B)^\perp = (B^\perp \otimes A^\perp) \\ (A\&B)^\perp = (B^\perp \oplus A^\perp) & \quad (A \oplus B)^\perp = (B^\perp \& A^\perp) \end{array}$$

MALL (resp., MLL) *Sequents* Γ, Δ are sets of formula occurrences $A_1, ..., A_{n\geq 1}$, proved using the following rules (resp., only identity and multiplicative rules):

- identity: $\dfrac{}{A, A^\perp}\,\text{ax} \qquad \dfrac{\Gamma, A \qquad \Delta, A^\perp}{\Gamma, \Delta}\,\text{cut}$
- multiplicatives: $\dfrac{\Gamma, A \qquad \Delta, B}{\Gamma, \Delta, A \otimes B}\,\otimes \qquad \dfrac{\Gamma, A, B}{\Gamma, A ⅋ B}\,⅋$

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

Proof Structures – The problem is to cope with the &-rule

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

for which a *superimposition* of two proof nets must be made. A *solution* is to introduce for each &-link a *boolean variable* (called *eigen-wight*)

$$\frac{\Gamma, A \qquad \Gamma, B}{\Gamma, A\&B}\,\&_p$$

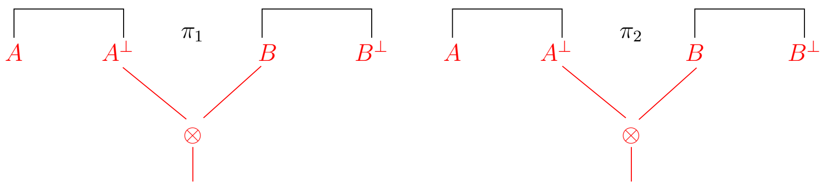
which distinguishes between two *slices* of the superimposition:

$$\begin{array}{cc} \bar{p}\ \textit{slice} & p\ \textit{slice} \\ \dfrac{\Gamma, A}{\Gamma, A\&B}\,\&_p & \dfrac{\Gamma, B}{\Gamma, A\&B}\,\&_p \end{array}$$

But this immediately opens to the problem of which kind of superposition can be performed over already desequentialized PSs? Let's illustrate this by an example. Assume a sequential proof as follows:

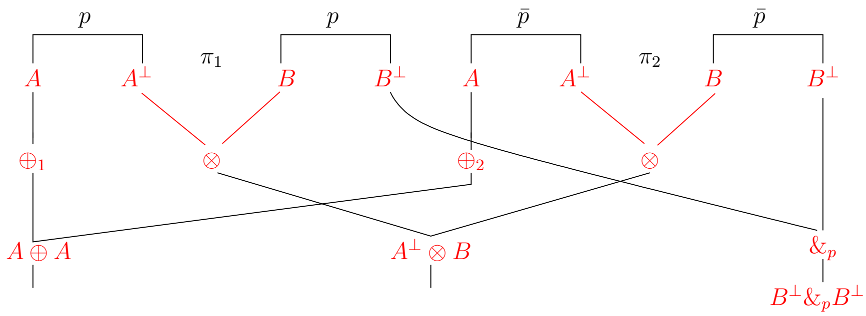
$$\Pi : \dfrac{\Pi_1 : \dfrac{\dfrac{\overline{A, A^\perp} \qquad \overline{B, B^\perp}}{A, A^\perp \otimes B, B^\perp}\,\otimes}{A \oplus A, A^\perp \otimes B, B^\perp}\,\oplus_1 \qquad \Pi_2 : \dfrac{\dfrac{\overline{A, A^\perp} \qquad \overline{B, B^\perp}}{A, A^\perp \otimes B, B^\perp}\,\otimes}{A \oplus A, A^\perp \otimes B, B^\perp}\,\oplus_2}{A \oplus A, A^\perp \otimes B, B^\perp \&_p B^\perp}\,\&_p$$

By hypothesis of induction we may assume Π_1 desequentializes in to (the MLL PN) π_1 and Π_2 desequentializes in to (the MLL PN) π_2 as follows:

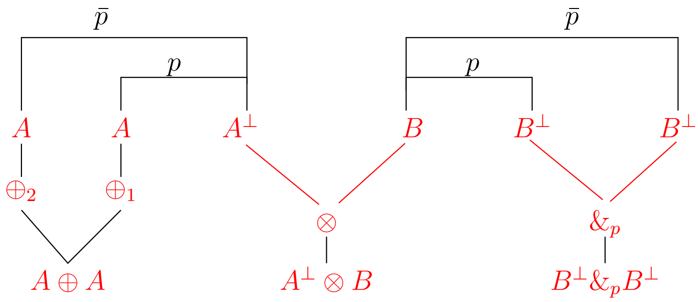


Then there are different possibilities of superposing pi_1 and π_2 in order to get a proof structure π that is a desequentialization of Π.

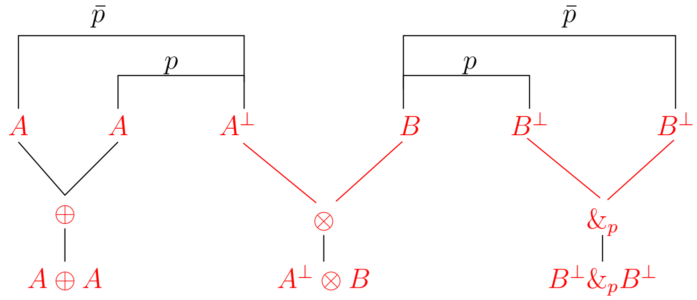
1. (Girard's solution) minimal superposition: only conclusions superpose



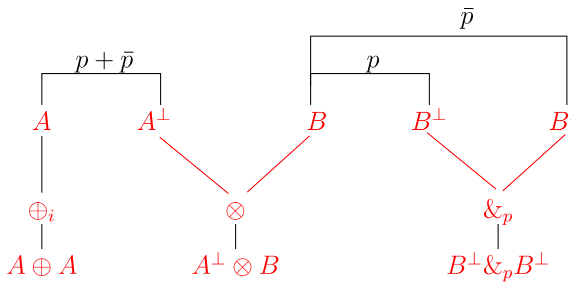
2. intermediate superposition with unary $\oplus$-links: only some links superpose



3. (Hughes-Van Glabbeek solution) maximal superposition with binary $\oplus$-links: the whole formula tree superpose (like in the MLL case).



Observe, the following is an example of wrong superposition: the $\oplus_i$ cannot simultaneolsy be of type $\oplus_1$ and $\oplus_2$.



So, given the *Bool*-algebra generated by the eigen-weights, we can get different notions of PN in which links are *weighted* by non-zero

- (Girard, 1998): *monomials* in case of *dependence condition* (*if L depends on p then* $w(L) \leq w(\&_p)$),
- or more general *polynomials* when *no dependence* at all is given.

1.2 Monomial interpretation

There is no canonical surjection from SP to Monomial PS. There is, indeed, only a (canonical) non-surjective mapping s.t.:

- nodes are weighted by non-zero monomials generated by the eigen-variables indexing the &'s nodes (satisfying the crucial dependency condition (if a node ν depends on a variable p then $w(\nu) \leq w(\&_p)$);
- a *minimal* superimposition of slices is allowed.

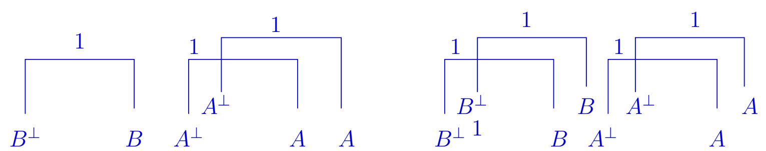
We illustrate that by means of an example. Assume we want de-sequentialize a SP as follows

$$\dfrac{\dfrac{\dfrac{}{B^\perp, B}\,\text{ax} \qquad \dfrac{\dfrac{}{A^\perp, A}\,\text{ax} \qquad \dfrac{}{A^\perp, A}\,\text{ax}}{A^\perp, A\&A}\,\&}{B^\perp, B\otimes A^\perp, A\&A}\,\otimes \qquad \dfrac{\dfrac{\dfrac{}{B^\perp, B}\,\text{ax} \qquad \dfrac{}{A^\perp, A}\,\text{ax}}{B^\perp, B\otimes A^\perp, A}\,\otimes \qquad \dfrac{\dfrac{}{B^\perp, B}\,\text{ax} \qquad \dfrac{}{A^\perp, A}\,\text{ax}}{B^\perp, B\otimes A^\perp, A}\,\otimes}{B^\perp, B\otimes A^\perp, A\&A}\,\&}{B^\perp \& B^\perp, B\otimes A^\perp, A\&A}\,\&$$

then, axioms (in blue)

$$\dfrac{\dfrac{\dfrac{}{\color{blue}{B^\perp, B}}\,\text{ax} \qquad \dfrac{\dfrac{}{\color{blue}{A^\perp, A}}\,\text{ax} \qquad \dfrac{}{\color{blue}{A^\perp, A}}\,\text{ax}}{A^\perp, A\&A}}{B^\perp, B\otimes A^\perp, A\&A} \qquad \dfrac{\dfrac{\dfrac{}{\color{blue}{B^\perp, B}}\,\text{ax} \qquad \dfrac{}{\color{blue}{A^\perp, A}}\,\text{ax}}{B^\perp, B\otimes A^\perp, A} \qquad \dfrac{\dfrac{}{\color{blue}{B^\perp, B}}\,\text{ax} \qquad \dfrac{}{\color{blue}{A^\perp, A}}\,\text{ax}}{B^\perp, B\otimes A^\perp, A}}{B^\perp, B\otimes A^\perp, A\&A}}{B^\perp \& B^\perp, B\otimes A^\perp, A\&A}$$

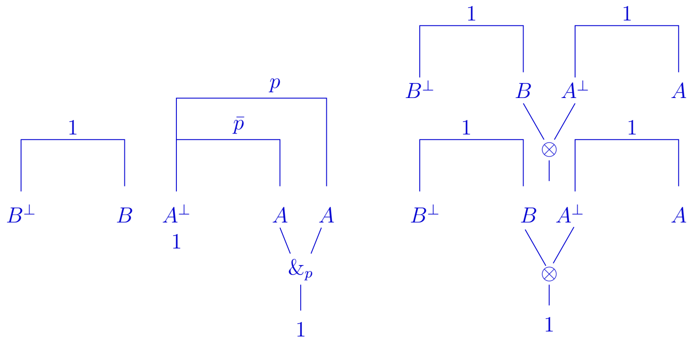
map to the following sub-proof nets (axioms links) with weight 1



the topmost & (with eigen-weight p) and $\otimes$-rules

$$\dfrac{\dfrac{\dfrac{}{B^\perp, B} \quad \dfrac{\dfrac{}{A^\perp, A} \quad \dfrac{}{A^\perp, A}}{A^\perp, A\&A}\,\&}{B^\perp, B\otimes A^\perp, A\&A} \quad \dfrac{\dfrac{\dfrac{}{B^\perp, B} \quad \dfrac{}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A}\,\otimes \quad \dfrac{\dfrac{}{B^\perp, B} \quad \dfrac{}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A}\,\otimes}{B^\perp, B\otimes A^\perp, A\&A}}{B^\perp\&B^\perp, B\otimes A^\perp, A\&A}$$

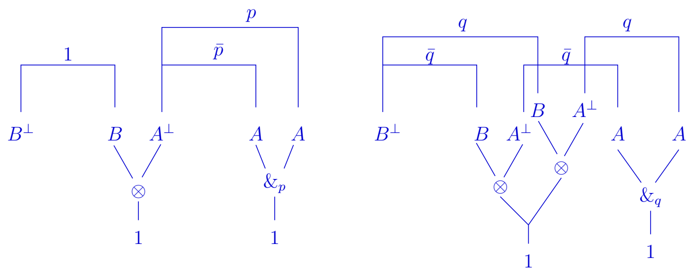
map to the following PNs



then, the middle $\otimes$ and &-rules (with eigen-weight q)

$$\dfrac{\dfrac{\dfrac{}{B^\perp, B} \quad \dfrac{\dfrac{}{A^\perp, A} \quad \dfrac{}{A^\perp, A}}{A^\perp, A\&A}}{B^\perp, B\otimes A^\perp, A\&A}\,\otimes \quad \dfrac{\dfrac{\dfrac{}{B^\perp, B} \quad \dfrac{}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A} \quad \dfrac{\dfrac{}{B^\perp, B} \quad \dfrac{}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A}}{B^\perp, B\otimes A^\perp, A\&A}\,\&}{B^\perp\&B^\perp, B\otimes A^\perp, A\&A}$$

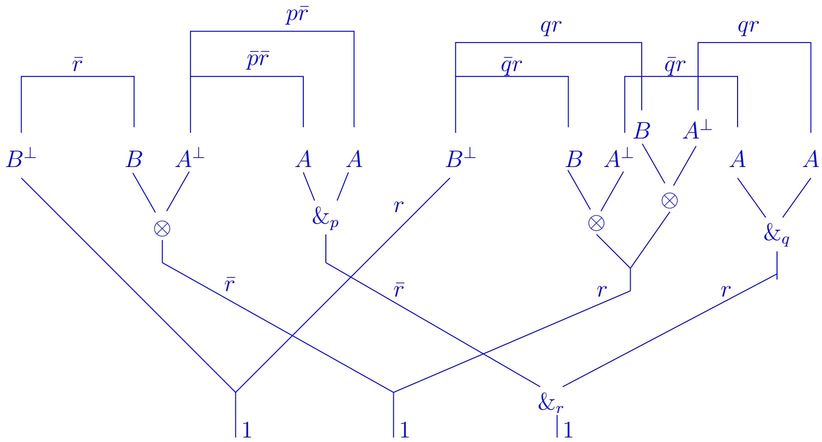
map to the following PN with only monomial weights



and, finally, the lowest &-rule (with eigen-weight r)

$$\dfrac{\dfrac{\dfrac{}{B^\perp, B} \quad \dfrac{\dfrac{}{A^\perp, A} \quad \dfrac{}{A^\perp, A}}{A^\perp, A\&A}}{B^\perp, B\otimes A^\perp, A\&A} \quad \dfrac{\dfrac{\dfrac{}{B^\perp, B} \quad \dfrac{}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A} \quad \dfrac{\dfrac{}{B^\perp, B} \quad \dfrac{}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A}}{B^\perp, B\otimes A^\perp, A\&A}}{B^\perp\&B^\perp, B\otimes A^\perp, A\&A}\,\&$$

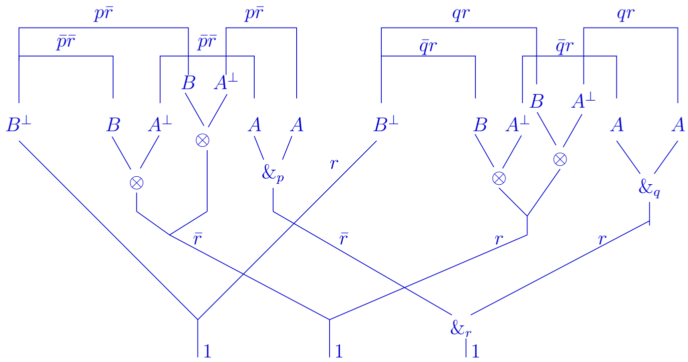
maps to the following PN with only monomial weights



Observe, this de-sequentialization is *not invariant* under the raising of the ⅋, $\otimes$, $\oplus$, & over the &-rule. Actually, if we raise, in the previous SP, the $\otimes$ over the $\&_p$-rule, as follows

$$\dfrac{\dfrac{\dfrac{\dfrac{}{B^\perp, B} \quad \dfrac{}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A}\otimes \quad \dfrac{\dfrac{}{B^\perp, B} \quad \dfrac{}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A}}{B^\perp, B\otimes A^\perp, A\&A}\,\& \quad \dfrac{\dfrac{\dfrac{}{B^\perp, B} \quad \dfrac{}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A} \quad \dfrac{\dfrac{}{B^\perp, B} \quad \dfrac{}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A}}{B^\perp, B\otimes A^\perp, A\&A}}{B^\perp\& B^\perp, B\otimes A^\perp, A\&A}$$

we get an equivalent PS which de-sequentialise into the following different PN



When we add a &-link, we don't know if a link L_1 of π_1 is the same as another link L' of π_2: in general, $p.w_1(L) + \bar{p}.w_2(L_2)$ is not a monomial, except when L_1, L_2 are conclusions.

1.3 Polynomial interpretation

There is a canonical surjection from MALL SP to Polynomial PN as illustrated in the following example. Assume the following SP:

$$\dfrac{\dfrac{\dfrac{}{B^\perp, B} \quad \dfrac{\dfrac{}{A^\perp, A} \quad \dfrac{}{A^\perp, A}}{A^\perp, A\&A}}{B^\perp, B\otimes A^\perp, A\&A} \quad \dfrac{\dfrac{\dfrac{}{B^\perp, B} \quad \dfrac{}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A} \quad \dfrac{\dfrac{}{B^\perp, B} \quad \dfrac{}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A}}{B^\perp, B\otimes A^\perp, A\&A}}{B^\perp\& B^\perp, B\otimes A^\perp, A\&A}$$

assign an eigen weight to each & in the sequent conclusion

$$\dfrac{\dfrac{\dfrac{}{B^\perp, B} \quad \dfrac{\dfrac{}{A^\perp, A} \quad \dfrac{}{A^\perp, A}}{A^\perp, A\&_{\color{blue}q}A}}{B^\perp, B\otimes A^\perp, A\&_{\color{blue}q}A} \quad \dfrac{\dfrac{\dfrac{}{B^\perp, B} \quad \dfrac{}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A} \quad \dfrac{\dfrac{}{B^\perp, B} \quad \dfrac{}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A}}{B^\perp, B\otimes A^\perp, A\&_{\color{blue}q}A}}{B^\perp\&_{\color{red}p} B^\perp, B\otimes A^\perp, A\&_{\color{blue}q}A}$$

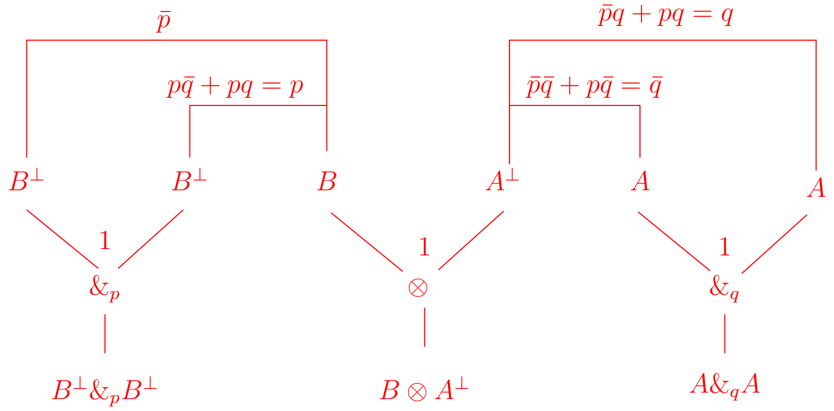
and propagate the eigen-weights upwards as follows

$$\dfrac{\dfrac{\dfrac{}{B^\perp, B}\qquad \dfrac{\dfrac{\bar q}{A^\perp, A}\qquad \dfrac{q}{A^\perp, A}}{A^\perp, A\&_q A}\,\&_q}{B^\perp, B\otimes A^\perp, A\&_q A}\qquad \dfrac{\dfrac{\dfrac{\bar q}{B^\perp, B}\qquad \dfrac{\bar q}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A}\qquad \dfrac{\dfrac{q}{B^\perp, B}\qquad \dfrac{q}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A}}{B^\perp, B\otimes A^\perp, A\&_q A}\,\&_q}{B^\perp\&_p B^\perp, B\otimes A^\perp, A\&_q A}$$

finally, separate, inductively (top-down), each slice by monomial weights as follows

$$\dfrac{\dfrac{\dfrac{\bar p}{B^\perp, B}\qquad \dfrac{\dfrac{\bar p\bar q}{A^\perp, A}\qquad \dfrac{\bar p q}{A^\perp, A}}{A^\perp, A\&_q A}\,\&}{B^\perp, B\otimes A^\perp, A\&_q A}\qquad \dfrac{\dfrac{\dfrac{p\bar q}{B^\perp, B}\qquad \dfrac{p\bar q}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A}\,\otimes\qquad \dfrac{\dfrac{pq}{B^\perp, B}\qquad \dfrac{pq}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A}\,\otimes}{B^\perp, B\otimes A^\perp, A\&_q A}\,\&}{B^\perp\&_p B^\perp, B\otimes A^\perp, A\&_q A}\,\&_p$$

The resulting corresponding de-sequentialized PN is a sequent forest with weighted axioms. We may replace parallel axioms $ax_1, ax_2, ..ax_n$ with, resp., weights $w_1, w_2, ..., w_n$, by a signle ax link with weight $w = \sum_i^n w_i$ as follows



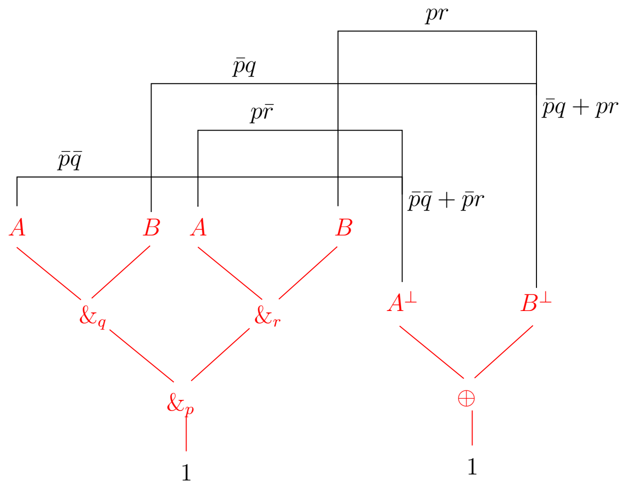
Observe, this "polynomial interpretation" is now invariant under the raising of the $⅋, \otimes, \oplus, \&$-rule over $\&$-rule; indeed the following(equivalent) SP maps to the same (previous) Polynomial PN.

$$\dfrac{\dfrac{\dfrac{\dfrac{\bar p\bar q}{B^\perp, B}\qquad \dfrac{\bar p\bar q}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A}\,\otimes\qquad \dfrac{\dfrac{\bar p q}{B^\perp, B}\qquad \dfrac{\bar p q}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A}\,\otimes}{B^\perp, B\otimes A^\perp, A\&_q A}\,\&\qquad \dfrac{\dfrac{\dfrac{p\bar q}{B^\perp, B}\qquad \dfrac{p\bar q}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A}\qquad \dfrac{\dfrac{pq}{B^\perp, B}\qquad \dfrac{pq}{A^\perp, A}}{B^\perp, B\otimes A^\perp, A}}{B^\perp, B\otimes A^\perp, A\&_q A}}{B^\perp\&_p B^\perp, B\otimes A^\perp, A\&_q A}$$

Observe that in general this surjective de-sequentialization may lead to PS labeled by polynomial weights; e.g., the next SP

$$\dfrac{\dfrac{\dfrac{\dfrac{\bar p\bar q}{A, A^\perp}\,\mathrm{ax}}{A, A^\perp\oplus B^\perp}\,\oplus_1\qquad \dfrac{\dfrac{\bar p q}{B, B^\perp}\,\mathrm{ax}}{B, A^\perp\oplus B^\perp}\,\oplus_2}{A\&_q B, A^\perp\oplus B^\perp}\,\&_q\qquad \dfrac{\dfrac{\dfrac{p\bar r}{A, A^\perp}\,\mathrm{ax}}{A, A^\perp\oplus B^\perp}\,\oplus_1\qquad \dfrac{\dfrac{pr}{B, B^\perp}\,\mathrm{ax}}{B, A^\perp\oplus B^\perp}\,\oplus_2}{A\&_r B, A^\perp\oplus B^\perp}\,\&_r}{(A\&_q B)\&_p(A\&_r B), A^\perp\oplus B^\perp}\,\&_p$$

de-sequentializes into a PS with some links weighted by "non-monomial" weighted. By the way observe the conclusion or terminal links must be labeled by the monomial weight 1 as in the following PN.



Finally, concerning the efficiency of the weight interpretations, observe that:

- both *monomial* and *polynomial* mapping are quite efficient: P-time in the size of the given sequent proof (there are $2.n$ slices, where n is the number of &-rules of the SP π);
- more efficient than *linkings mapping* (Hughes-Van Glabbeek, 2003) that is Exponential in the size of the sequent proof (there are 2^n, where n is the number of the & connective occurring in the sequent Γ).

1.4 Correctness Criterion for Monomial PNs

- **proof structure**: the crucial point is the *dependence condition* ("&-boxing"): if a link L depends on a variable p then $w(L) \leq w(\&_p)$.
- **proof net**: every boolean valuation, φ : *eigenweights* $\to \{0,1\}$, induces a (unique) slice S s.t. every *switching* on S, obtained by

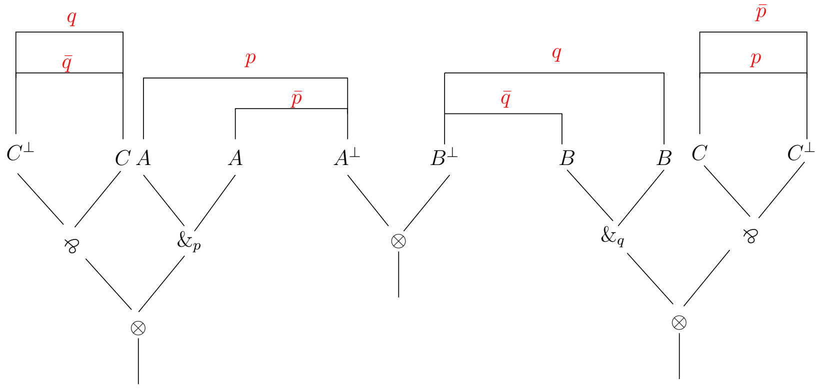
  – mutilating one premise in each $⅋$,

  – mutilating the unique &-premise in S,

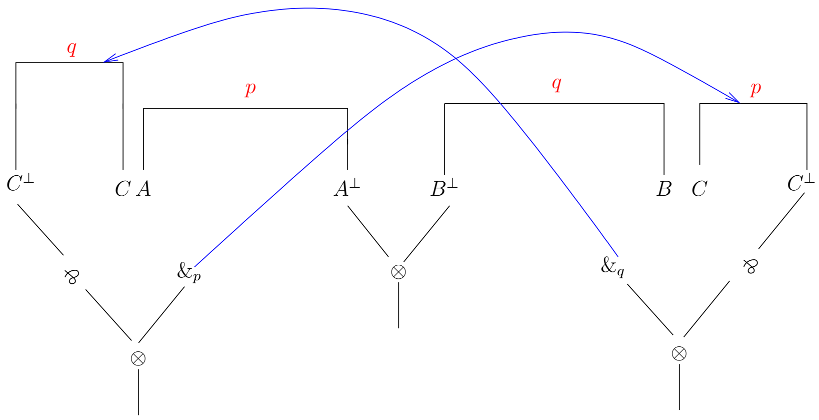
  – adding a jump (an edge) from a $\&_p$-node to a node depending on p,

  is an acyclic and connected (ACC) graph.

In the following we give an example of non correct PS.

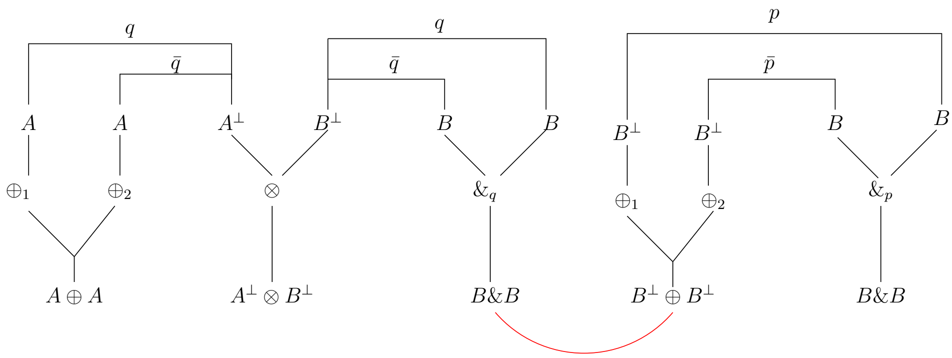


Actually, fixed a valuation $\varphi(p) = \varphi(q) = 1$, we get a non-ACC switching as follows.

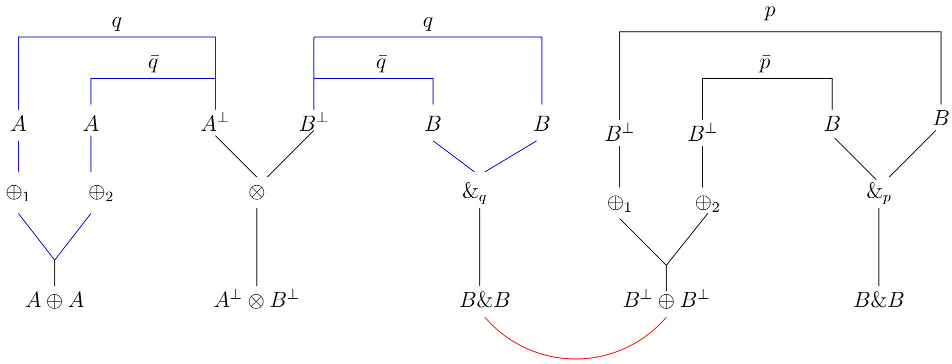


1.5 Cut-elimination for Monomial PSs

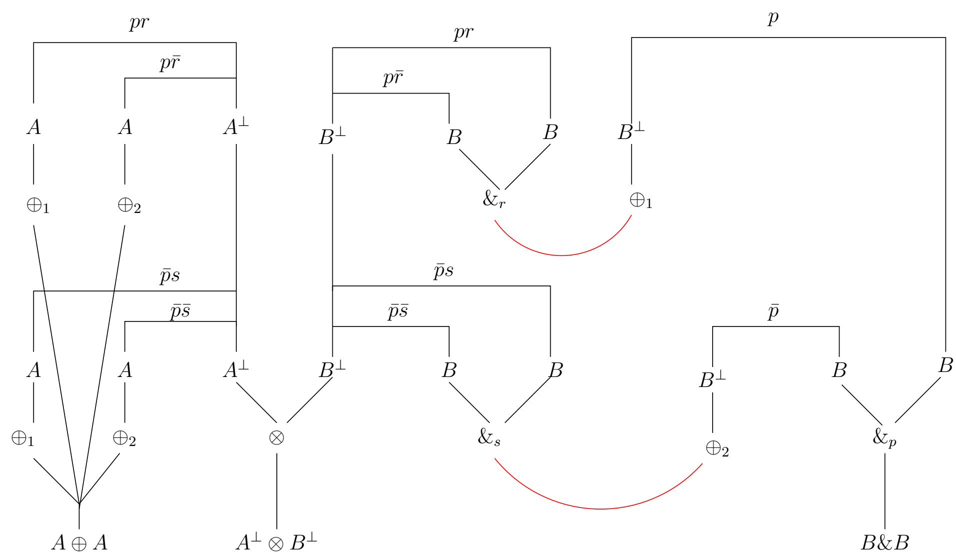
Assume a monomial PN π with cuts as follows.



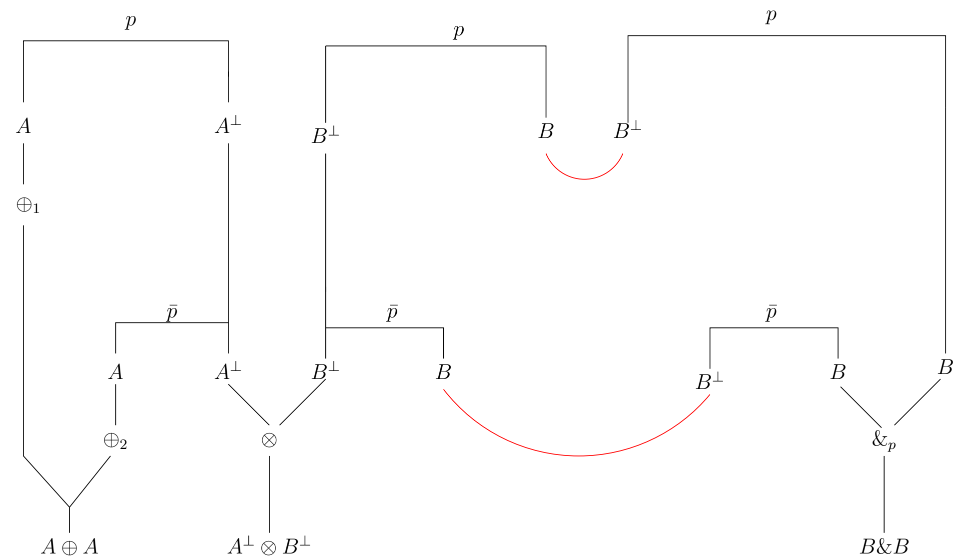
We call *dependency graph* of q w.r.t. π (Maieli, 2007) the (possibly disconnected) subgraph of π made by only those links and edges whose weight depends on q; e.g., the dependency graph of q w.r.t. π is the blue sub-graph as follows:



Then π reduces to the following π' via the *duplication* of the *dependency graph* of q: we replace in π' the eigen-weight q of π by two news (fresh) eigen-weights r and s as follows



Consequently, $\oplus_i/\&$ cuts reduce to the following PN by erasing slices $\bar{r}$ and s (i.e., by evaluating $\bar{r} = 0$ and $s = 0$).



Cut elimination procedure is shown to be terminating and confluent but with an unknown Complexity (P-time?).

2 Proof-structures with explicit binary contraction links

In this section we recall the basic notions of Girard's proof-net; we adopt the syntax with explicit contractions like in [Lau99].

Definition 1 (proof structure) *A* Girard proof structure π *of MALL, shortly* proof structure *(*PS*), is an oriented graph s.t. each edge is labelled by a MALL formula and built on the set of nodes (or vertexes) following the typing constraints of Figure 1. Pending edges are called* conclusions*; fixed a node, an entering edge is called* premise *while its unique emergent edge is called* conclusion*. We call* link *the graph made by a node together with its premise(s) and (possibly) its conclusion(s).*
If π involves the &-links $L_1, ..., L_k$ then:

1. *we associate a* Boolean variable *$(p, q, ...)$, called* eigen weight*, to each &-node of π (eigen weights are supposed to be different; we use ϵ_p to denote a variable p or its negation $\overline{p}$, and $\overline{\epsilon_p}$ for its orthogonal);*

2. *we associate a* weight *w, i.e., a product (conjunction) of eigen weights or negations of eigen weights of π $(p, \overline{p}, q, \overline{q}...)$, to each node with the constraint that two nodes have the same weight if they have a common edge, except when the edge is the premise of a & or C-node, in these cases we do as follows (see Figure 2):*

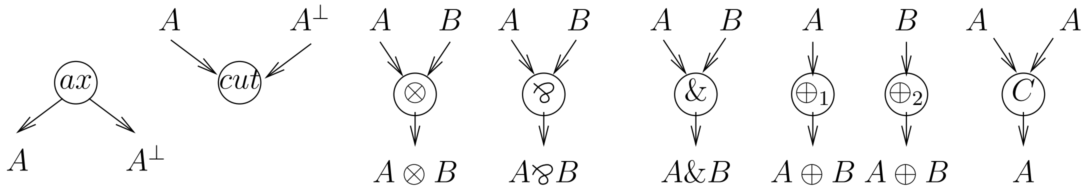


Figure 1: MALL links

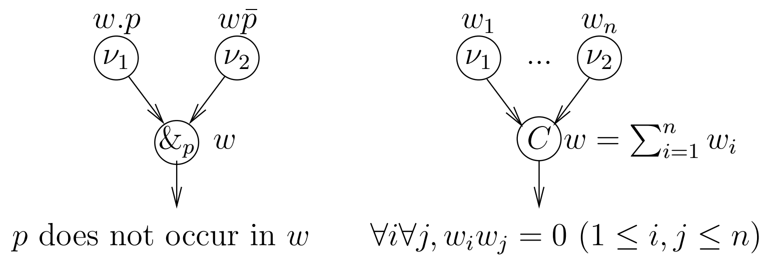


Figure 2: Weights for & and C links

(a) *if w is the weight of a &-link and p is its eigen weight then w does not* depends on p *and its premise links must have weights $w.p$ and $w.\overline{p}$ (we say that a weight w* depends on p *when p or $\overline{p}$ occurs in w);*

(b) *if w is the weight of a C-link and w_1, w_2 are the weights of its premise links then we must have $w = w_1 + w_2$ and $w_1 w_2 = 0$;*

3. *a* conclusion node *has weight* 1*;*

4. *if w is the weight of a &-link with eigen weight p and w' is a weight depending on p and appearing in the proof-structure then $w' \leq w$.*

A node L with weight w *depends on* the eigen weight $\&_p$ if w depends on p or L is a C-node and one of the weights just above it depends on p.

Remark 1 *:*

1. *(*splitting variable*) – Observe that, since the weights associated to a PS are products (*monomials*) of the Boolean algebra generated by the eigen weights associated to a proof structure, then, for each weight w associated to a contraction node, there exists a unique eigen weight p that* splits w into $w_1 = wp$ and $w_2 = w\overline{p}$. *We some times index a C-link with its* splitting variable *p, like in Figure 3.*

2. *(*dependency condition*) – Observe that the graph π of Figure 4 is not a proof structure since it violates condition 4 of Definition 1; actually, if $w = q$ is the weight of the $\&_p$-link and $w' = \overline{p}$ is a weight depending on p and appearing in the proof-structure then $\overline{p} \not\leq q$.*

Definition 2 (slice and switchings) *A* valuation *φ for a PS π is a function from the set of all weights of π into $\{0, 1\}$. Fixed a valuation φ for π then:*

- *the* slice *$\varphi(\pi)$ is the graph obtained from π by keeping only those nodes with weight 1 together its emerging edges;*

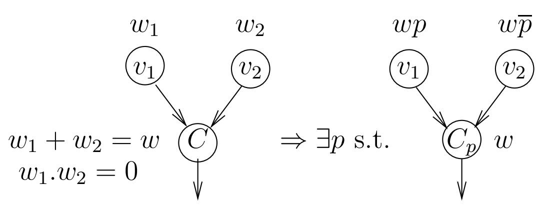


Figure 3: Splitting variable of a contraction link

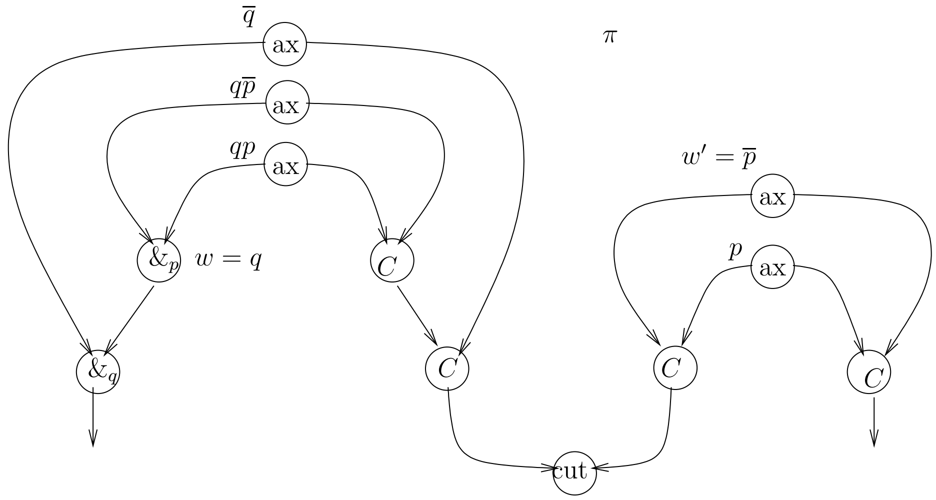


Figure 4: Violation of the dependency condition

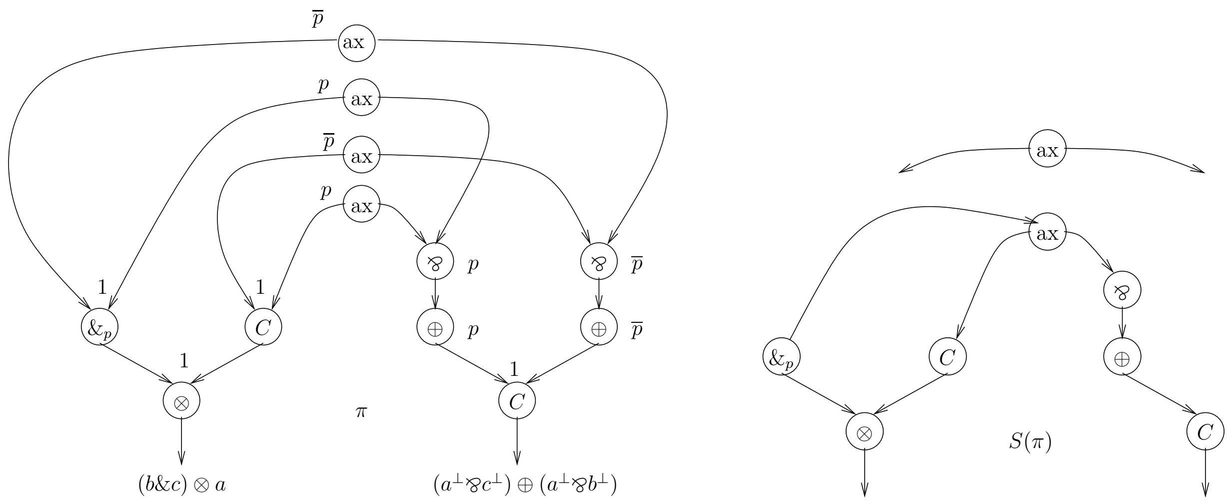


Figure 5: Example of non sequentializable PS

- *a* multiplicative switching *S for π is the non oriented graph built on the nodes and edges of $\varphi(\pi)$ with the modification that for each $\parr$-node we take only one premise and we cut the remaining one (*left/right $\parr$-switch*);*
- *an* additive switching *(or simply a* switching*) is a multiplicative switching where for each $\&$-node we cut the (unique) premise in $\varphi(\pi)$ and we add an oriented edge, called* jump*, from the $\&$-node to a L-node whose weight depends on the eigen weight of the $\&$-node.*

Definition 3 (Girard's proof-net) *A PS π is* correct *(it is a* proof-net, PN*), if any switching induced by a valuation of π is acyclic and connected (*ACC*).*

Theorem 1 ((de-)sequentialization) *A PN can be* sequentialized *into a MALL sequent proof with same conclusions and vice-versa (*de-sequentialization*).*

Proof — see[Gir96]. □

Remark 2 *The proof structure π on the left hand side of Figure 5 is not correct (not sequentializable): actually, fixed a valuation φ s.t. $\varphi(p) = 1$, then there exists a switching $S(\pi)$ with a remote jump that is not ACC (see the right hand side of Figure 5). Nevertheless, π is correct by multiplicative slices (all multiplicative switchings, without remote jumps, are ACC).*

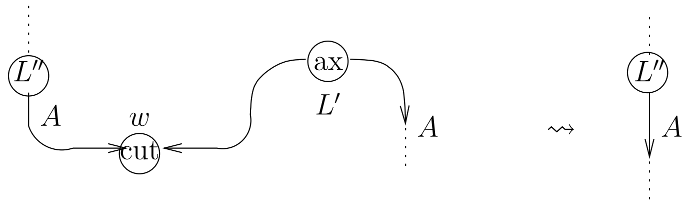


Figure 6: Axiom-cut reduction step

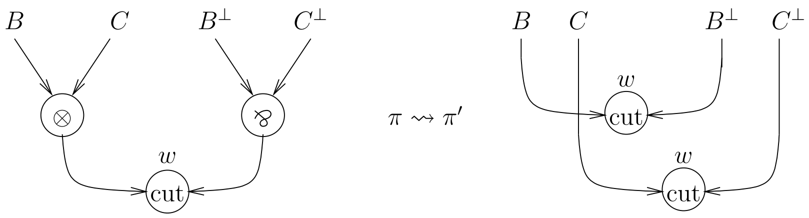


Figure 7: $(\otimes/\parr)$-cut reduction step

3 Cut-elimination

3.1 Ready cut-elimination

We follow [Gir96].

Definition 4 (ready cut reduction) *Let L be a cut in a proof net π whose premises A and $A^\perp$ are the respective conclusions of links L', L'' both different from the contraction C. Then we define the result π' (*reductum*) of reducing this* ready *cut in π (*redex*), as follows:*

***Ax*-cut:** *if L' (resp., L'') is an axiom link then π' is obtained by removing in π both formulas $A, A^\perp$ (as well as L) and giving a new conclusion to L'' (resp., L'), the other conclusion of L' (resp., L'') (see Figure 6).*

$(\otimes/\parr)$**-cut:** *if L' is a $\otimes$-link with premises B and C and L'' is a $\parr$-link with premises $B^\perp$ and $C^\perp$, then π' is obtained by removing in π the formulas A and $A^\perp$ as well as our cut link L with L' and L'' and adding two new cut links with respective premise B, $B^\perp$ and $C, C^\perp$ (see Figure 7).*

$(\&/\oplus_1)$**-cut:** *if L' is a $\&_p$-link with premises B and C and L'' is a $\oplus_1$-link with premise $B^\perp$, then π' is obtained in three steps: first we remove in π both formulas A, $A^\perp$ as well as our cut link L with L' and L'', then we replace the eigen weight p by 1 and keep only those links (vertexes and edges) that still have nonzero weight; finally we add a cut between B and $B^\perp$ (see Figure 8).*

$(\&/\oplus_2)$**-cut:** *if L' is a $\&_p$-link with premises B and C and L'' is a $\oplus_2$-link, with premise $C^\perp$, then π' is obtained in three steps: first we remove in π both formulas A, $A^\perp$ as well as our cut link L with L' and L'', then we replace the eigen weight $p_\&$ by 0 and keep only those links (vertexes and edges) that still have nonzero weight; finally we add a cut between C and $C^\perp$.*

Theorem 2 (stability under ready cut reduction) *If π is a proof net s.t. π reduces to π' in one step of ready cut reduction, then π' is still a proof net.*

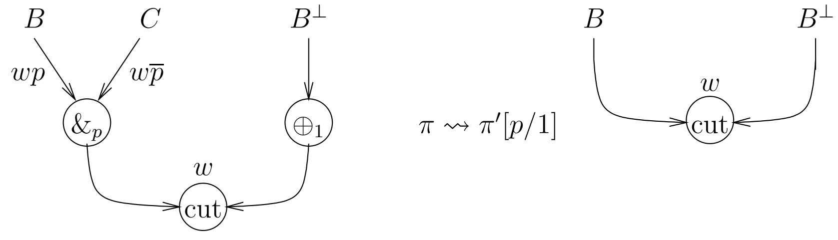


Figure 8: $(\&/\oplus_1)$-cut reduction step

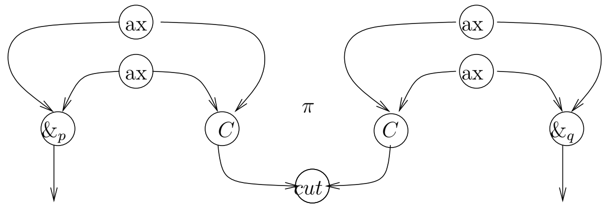


Figure 9: Reducing a commutative cut

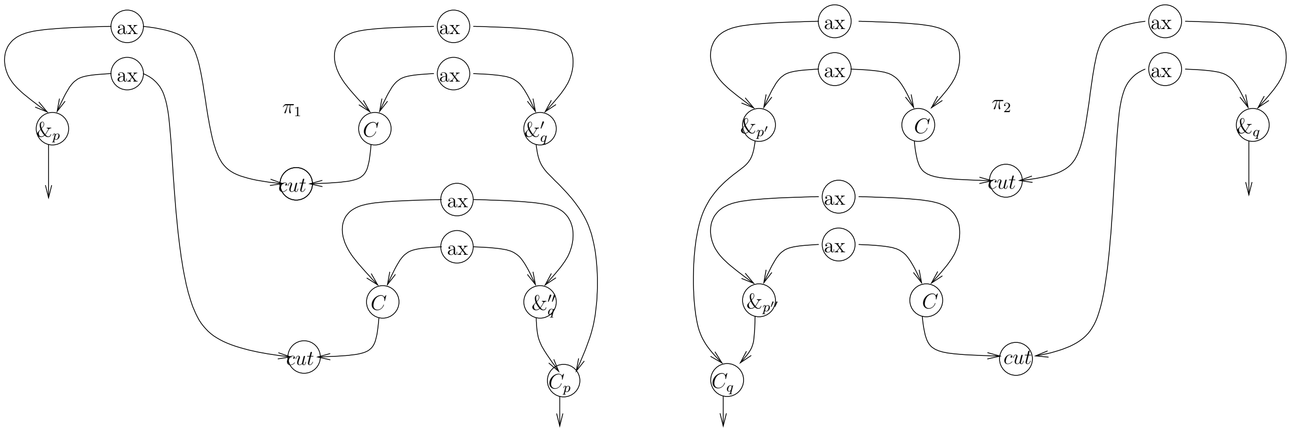


Figure 10: Non confluent, commutative, cut reduction

Proof — See [Gir96]. □

3.2 Commutative cut-elimination

3.2.1 The confluence problem

– In general, reducing a cut involving a contraction link as (at least) one of its premises from a proof structure π may lead to several solutions, depending on which sub-graph of π we decide to duplicate. For instance, reducing the commutative cut of π of Figure 9 may lead to solutions π_1 as well solution π_2, depicted in Figure 10, depending on which additive box ($\&_q$ or $\&_p$) we decide to duplicate. Of course these two resulting proof nets are different and there is no a-priori way to make them equal. Girard, in [Gir96], does not give any detail for the solution to this problem; indeed, a Church-Rosser procedure is only given for the ready cut-elimination.

Definition 5 (restriction, empire, spreading) *Assume a proof structure π, an eigen weight p and a weight w, then:*

- *the* restriction *of π w.r.t. p (resp., $\overline{p}$), denoted $\pi\downharpoonleft^p$ (resp., $\pi\downharpoonleft^{\overline{p}}$), is what remains of π when we replace p with 1 (resp., $\overline{p}$ with 1) and keep only those vertexes and edges whose weight is still non zero;*
- *the* empire *(or the* dependency graph*) of the eigen weight p w.r.t. π, denoted $\mathcal{E}_p$, is the (possibly disconnected) subgraph of π made by all links depending on p.*
- *the* spreading *of w over π, denoted by $w.[\pi]$, is the product of w for π, i.e., π where we replaced each weight v with the product of weights vw.*

Observe that, in general, the spreading does not preserve the property of being a proof structure; moreover, it can be defined also over an empire.

Lemma 1 (empire) *If a $\&_p$-node belongs to the empire of $\mathcal{E}_q$, then $\mathcal{E}(p) \subset \mathcal{E}(q)$.*

Proof — If the $\&_p$-node belongs to the empire of $\mathcal{E}_q$ then the weight w of the $\&_p$-node depends on q (i.e., $w = w'q$ or $w = w'\overline{q}$) then trivially, by the dependency condition 4 of Definition 1, each node v whose weight depends on p will also depends on q. □

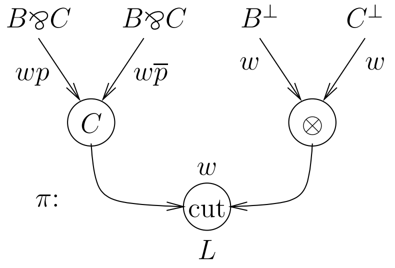


Figure 11: $(C/\otimes)$-redex

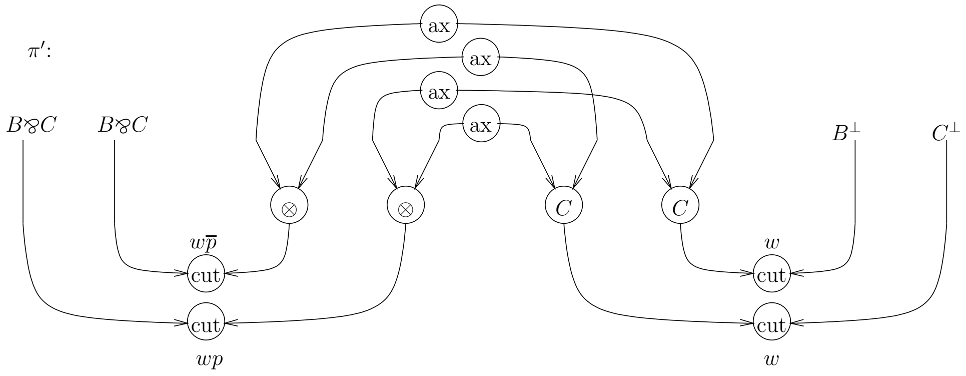


Figure 12: $(C/\otimes)$-reductum

Definition 6 (commutative cut reduction) *Let L be cut link in a proof net π whose premises A and $A^\perp$ are the respective conclusions of links L' and L'' s. t. at least one of them is a contraction link C. Then we define the result π'* (reductum) *of reducing this* commutative *cut in π* (redex) *, as follows:*

$(C/\otimes)$**-cut** *: if L' is a C-link and L'' is a $\otimes$-link, like in Figure 11, then π reduces in one $(C/\otimes)$ step to π', like in Figure 12.*

(C/C)**-cut** *: if both L' and L'' are contraction links, then we consider two sub-cases:*

- *the weight w of, resp., L' and L'' is split by the same p variable, then π reduces in one (C_p/C_p) step to π' like in Figure 13;*
- *the weight w of L', resp., L'', is split by the two different variables, p and q, like in Figure 14, then π reduces in one (C_p/C_q) step to π', like in Figure 15.*

$(C/\oplus_i)$**-cut** *: if L' is a C-link and L'' a $\oplus_{i=1,2}$-link, then π reduces in one $(C/\oplus)$ step to π', like in Figure 16.*

$(C/\&)$**-cut** *: if L' is a C-link and L'' a $\&_p$-link, like in Figure 17, then π reduces in one $(C/\&)$ step to π', like in Figure 18, with the assumptions that $\bar{q}.[\mathcal{E}'_p]$ and $q.[\mathcal{E}''_p]$ are obtained by spreading $\bar{q}$, resp., q, over two copies of the empire of p, $\mathcal{E}'_p$ and $\mathcal{E}''_p$, where we replaced any eigen weight with a new (fresh) one.*

Example 1 *Assume we want reduce the commutative $(C/\&)$-cut of the proof net in Figure 19 First, we calculate $\mathcal{E}'$ and $\mathcal{E}''$, like in Figure 20. Then we perform the cut reduction step $C/\&$, like in Figure 21.*

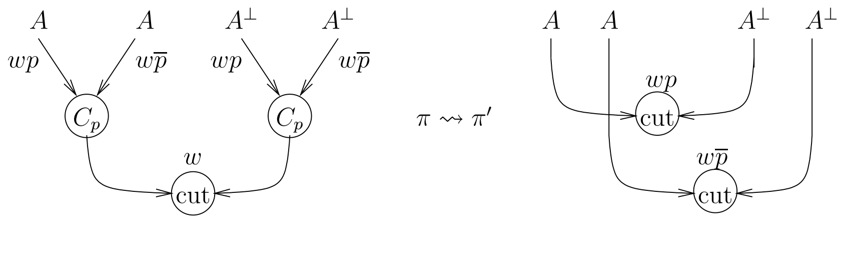


Figure 13: (C_p/C_p)-cut reduction step

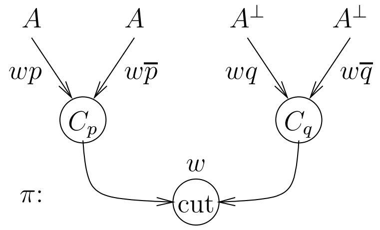


Figure 14: (C_p/C_q)-redex

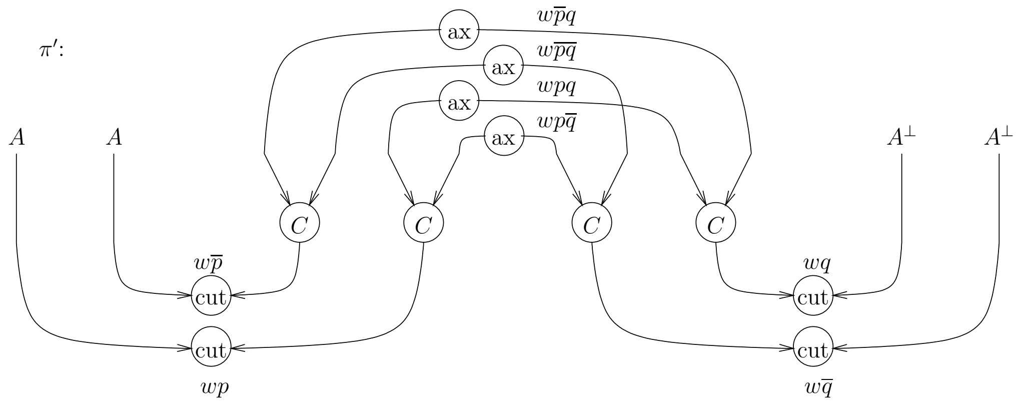


Figure 15: (C_p/C_q)-reductum

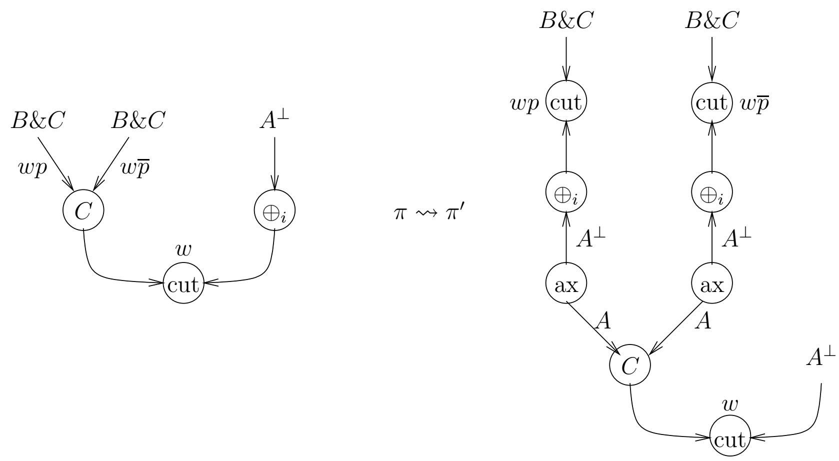


Figure 16: $(C/\oplus_i)$-cut reduction step

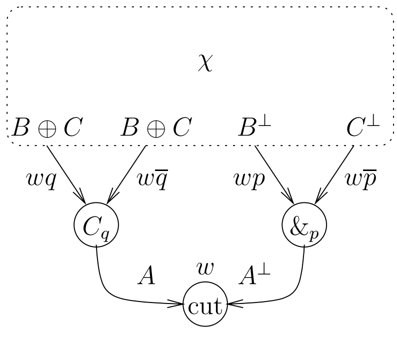


Figure 17: $(C/\&)$-redex

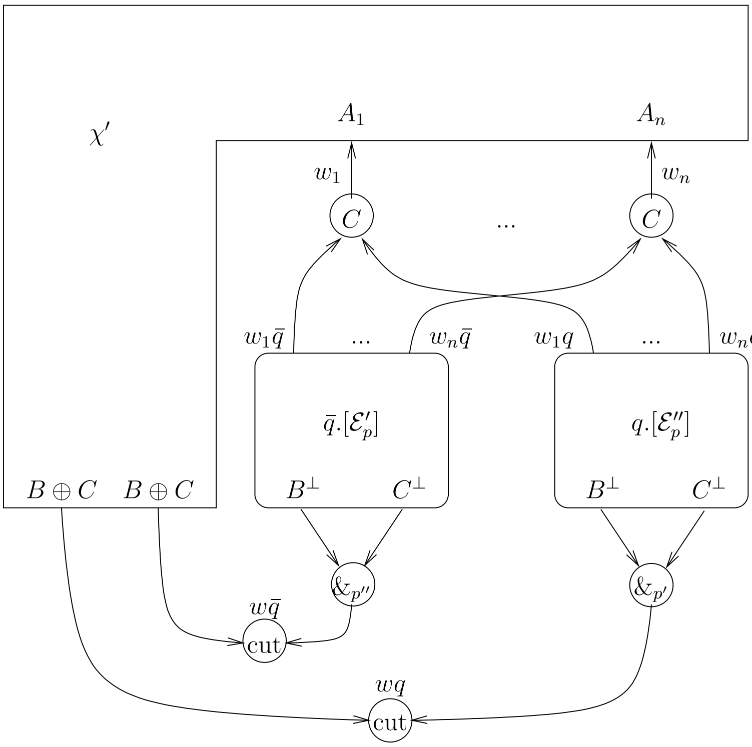


Figure 18: $(C/\&)$-reductum

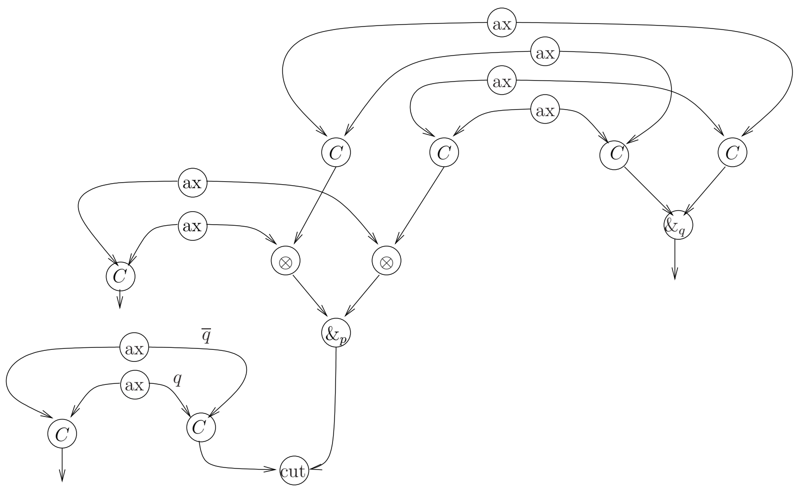


Figure 19: Redex π

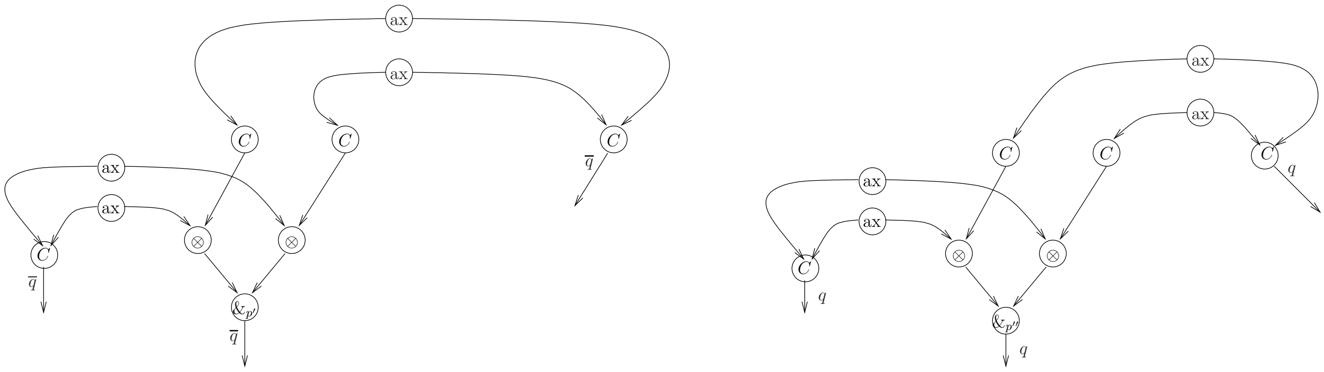


Figure 20: Graphs $\mathcal{E}'$ and $\mathcal{E}''$

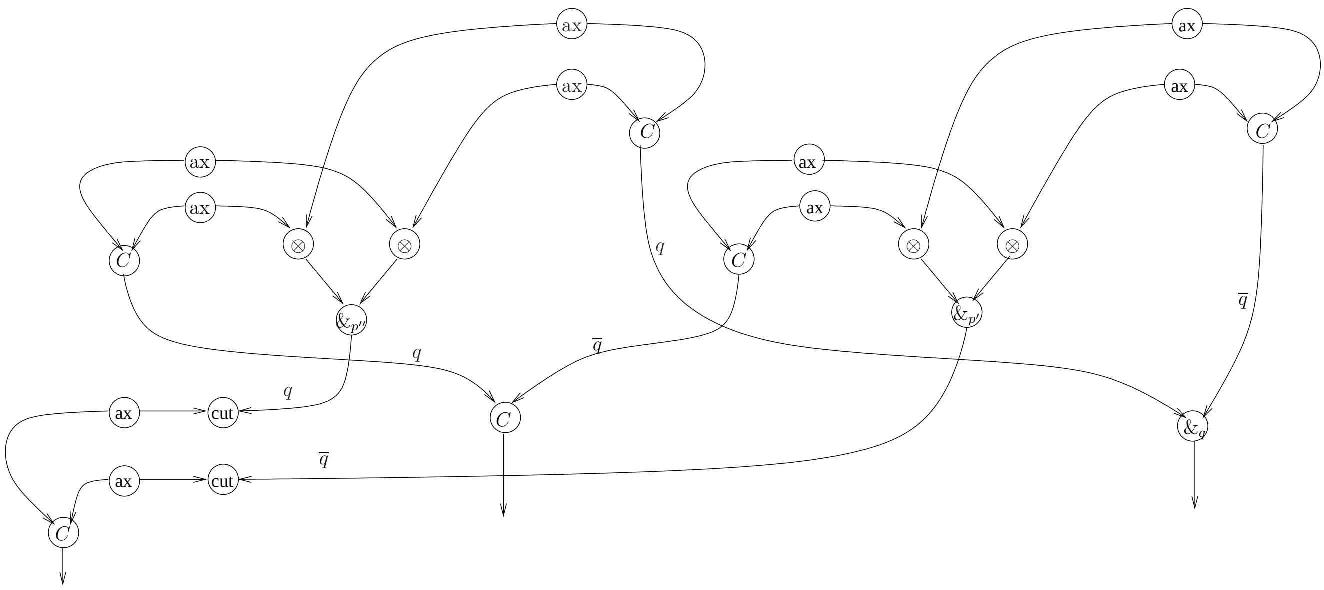


Figure 21: Reductum π'

3.3 Stability

Theorem 3 (stability under commutative cut reduction: I part) *If π is a proof net s.t. it reduces to π' in one step of commutative cut reduction that is different from the $C/\&$-case, then π' is still a proof net.*

Proof — (*sketch*) All cases are more or less immediate consequences of the next graph theoretical property (see also [Gir06], pages 250-251):

Property 1 (Euler-Poicaré invariance) *Given a graph $\mathcal{G}$, then*

$$\sharp CC - \sharp Cy = \sharp V - \sharp E$$

where $\sharp CC$, $\sharp Cy$, $\sharp V$ and $\sharp E$ denotes, respectively, the number of connected components, cycles, vertexes and edges of $\mathcal{G}$.

□

Lemma 2 (conservative/extensive switching) *Assume π is a proof net that reduces to π' in one step of commutative cut reduction $C_q/\&_p$. We say that a switching for π' is* extensive *(resp.,* conservative*) w.r.t. q if it makes (resp., does not make) use of at least a jump going from the $\&_q$-node to a node that was not previously depending on q in π. Then, for any conservative switching for π' that is not ACC we can find a corresponding switching for π that is not ACC too.*

Proof — Immediate. □

Lemma 3 (separation) *Assume π is a proof net containing two nodes, $\&_{p'}$ and $\&_{p''}$, with weights, resp., $w'q$ and $w''\overline{q}$, for some eigen weight q of π. Then, there cannot exist in π a node whose weight depends both on p' and on p''.*

Proof — By absurdum, assume a node v whose weight w depends both on p' and p'', that is, for instance, $w = w_1.p'.p''$. Now fix an evaluation φ s.t. $\varphi(w) = 1$, then by the dependency condition 4 of Definition 1, we have both $w_1.p'.p'' \leq w'q$ and $w_1.p'.p'' \leq w''\overline{q}$; but this is only possible when $\varphi(w) = 0$, contradicting the assumption $\varphi(w) = 1$. □

Theorem 4 (stability under commutative cut reduction: II part) *If π is a proof net that reduces to π' in one step of cut reduction $C/\&$, then π' is a proof net too.*

Proof — (*sketch*) First observe that each cut reduction step $C_q/\&_p$ preserves the property of being a proof structure. This follows by construction of π'. In particular Lemma 1 ensures that we can safely rename the sets of eigen weights in $\mathcal{E}'$ and $\mathcal{E}''$ and get still a proof structure.
Moreover, by Lemma 2 it is enough to only verify that all the extensive switchings, w.r.t. q, for π' are ACC. In the rest of proof, whenever it is not explicitly declared, each switching for π' is meant to be extensive w.r.t. q.
Connexion - Assume by absurdum there exists such a switching $S'(\pi')$ that is disconnected. Then there exists at least a node v_j occurring in a connected component γ_1 that is separated from the component γ_2 that contains the $\&_q$-node together with the jump ν directed to a node v_i (see the picture on the left hand side of Figure 22, where γ_1 and γ_2 are separated by dotted lines). This means that there not exist in $S'(\pi')$ a path from v_j to v_i. Now let $S''(\pi')$ be an other switching that is a copy of $S'(\pi')$ except for the jump ν from the $\&_q$-node to a v_h node that was already depending on q in π (in other words, S'' is a conservative switching, w.r.t. q, which differs from S' only for the jump ν, like in the picture on the right hand side of Figure 22). Clearly in $S''(\pi')$ there is no connection path between v_i and v_j, otherwise this path should go through the ν-jump (and so through the $\&_p$-node) contradicting the assumption that the $\&_q$-node and v_j were disconnected in $S'(\pi')$. But this contradicts, by Lemma 2, the hypothesis that π was correct.
Acyclicity - Assume there exists a cycle in $S'(\pi')$ going through a jump ν from the $\&_q$-node to a node v_i whose weight w_i depends on q, like in Figure 23 where γ is the path in $S'(\pi')$ from the conclusion of the $\&_q$-node to v_i. Now, the new variable p' or its negation (resp., p'' or its negation) occurs by definition of the reduction step $C_q/\&_p$ in the weight w_i (let us say $w_i = w_i'p'q$) so, by the condition 4 of Definition 1 and the connexion of $S'(\pi')$ showed before, the $\&_{p'}$-node (resp., the $\&_{p''}$-node) and the node v_i must be connected in $S'(\pi')$ in two possible ways:

1. either by a path going through the $\&_q$-node without accrossing γ, like a path γ_1 in Figure 24,
2. or like a path γ_2' or γ_2'') of Figure 24, that is, non going through the $\&_q$-node and possibly accrossing γ.

In both cases we can set a conservative switching $S''(\pi')$ that is identical to $S'(\pi')$ except for the jump ν going from the $\&_q$-node to the node v_j whose conclusion $A^\perp$ is the premise of the reductum cut, like in Figure 25. Then we get a cycle, contradicting, by Lemma 2, the hypothesis that π was correct. □

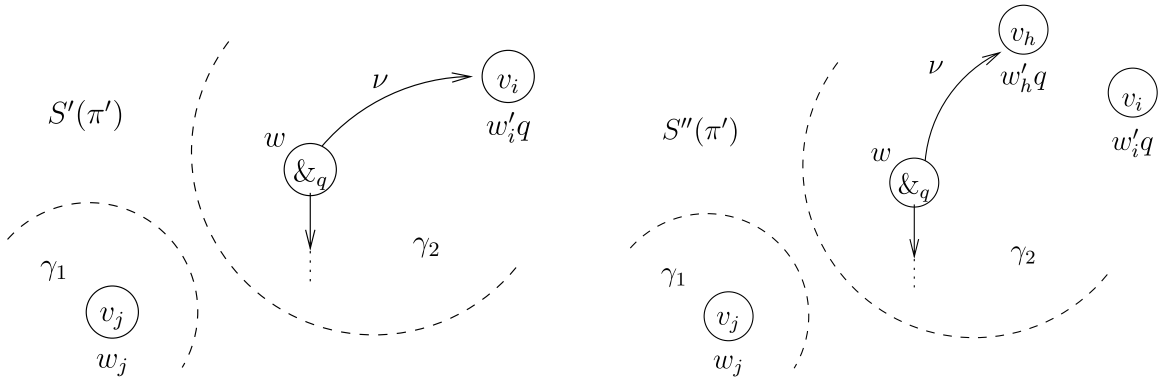


Figure 22:

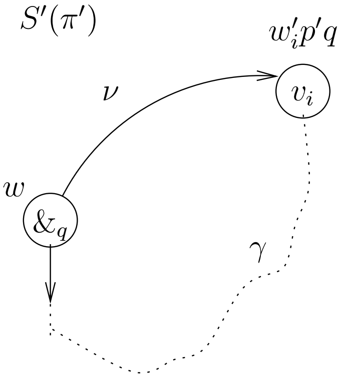


Figure 23:

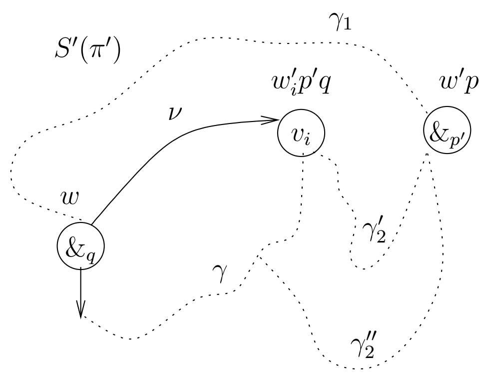


Figure 24:

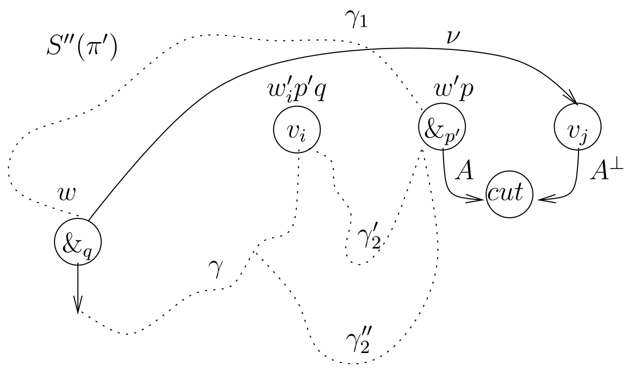


Figure 25:

3.4 Strong normalization

3.4.1 Weak normalization

We say that a cut link L is *safe* when reducing L does not make disappear any other cut link $L' \neq L$ from the reductum. Similarly, a cut reduction step is safe when it concerns a safe cut link. A cut reduction strategy is a finite sequence of cut reduction steps of Definition 4 and 6; a strategy is safe when it contains only safe cut reduction steps.

The *complexity* of a cut node is the logical complexity of its cut-formula[1]. We say that a cut-node with weight w has *depth* n if $|w| = n$, where $|w|$ is the number of variables of negations of variables occurring in w.

Lemma 4 (safe reduction) *If π is a proof net with only cut links of type $\&/\oplus_i$, then at least one of them is safe.*

Proof — By absurdum, assume π contains only $\&/\oplus_i$ cut links (at least one) and assume none of them is safe; then by the dependence condition we can easily find a switching with a cycle. □

Theorem 5 (weak normalization) *If π is a non cut-free proof net, then there exists a safe reduction strategy for π terminating with a cut-free (normal) proof net.*

Proof — Assume a safe reduction strategy consisting in applying a safe reduction step $\&/\oplus_i$ only when no other reduction choice is possible. By Lemma 4 we know that such a strategy exists. Then, termination of this such a safe strategy follows by induction on the lexicographic order of the *cut complexity sequence* $\sharp 1, ..., \sharp n$ of π, where n is the number of eigen-weight variables occurring in π and $\sharp i$, with $1 \leq i \leq n$, is the sum of the *complexities* of all cuts whose *depth* is i. It is immediate to check that the complexity sequence decreases at each reduction step, except for the reduction step $C/\&$, where in order to apply the hypothesis of induction we have to ensure that the length n of the complexity sequence does not increase. In other words we have to show that there cannot exists in π' a cut whose weight has depth greater than n. Now, assume by absurdum that, after one step of cut reduction $C/\&_p$, π reducts to π' with a cut node v_i whose depth is $n+1$, so the cut complexity sequence of π' is $\sharp 1, ..., \sharp n, \sharp(n+1)$. This could only be consequence of the fact that, after the $C/\&_p$-reduction step, the eigen weight p of π has been replaced in π' by two new eigen weights p' and p''. In other words, the weight w_i of v_i will depend both on p' and p'', like in Figure 26, with for instance $w_i'p'p'' \leq w_1 q$ and $w_i'p'p'' \leq w_1\overline{q}$, contradicting the Separation Lemma 3. □

3.4.2 Confluence

Theorem 6 (confluence) *Assume π is proof net s.t. it reduces in one step α to π' ($\pi \rightsquigarrow_\alpha \pi'$) and it reduces in an other step β to π'' ($\pi \rightsquigarrow_\beta \pi''$); then, there exists a proof net σ such that π' reduces, in a certain number of steps, to σ ($\pi' \rightsquigarrow^* \sigma$) and π'' reduces, in a certain number of steps, to σ ($\pi'' \rightsquigarrow^* \sigma$).*

Proof — (*sketch*) Assume $\pi \rightsquigarrow_\alpha \pi'$ and $\pi \rightsquigarrow_\beta \pi''$, then we proceed by cases according to α and β and we show that there always exists such a σ which both π' and π'' reduce to.
Case 1 - If neither α nor β is a reduction step involving a $\&$-node as a cut premise node, then it is easy to check that we get the confluence to σ by two more single reduction steps, $\pi' \rightsquigarrow_\beta \sigma$ and $\pi'' \rightsquigarrow_\alpha \sigma$, like in Figure 27 (the *diamond composition*).

[1] The *logical complexity* of a formula is inductively defined as follows: atoms have complexity 1; the complexity of $A^\perp$ is the same as the complexity of A; the complexity of $B \bullet C$, where $\bullet$ is any binary connective, is the sum of the complexities of B and C plus 1.

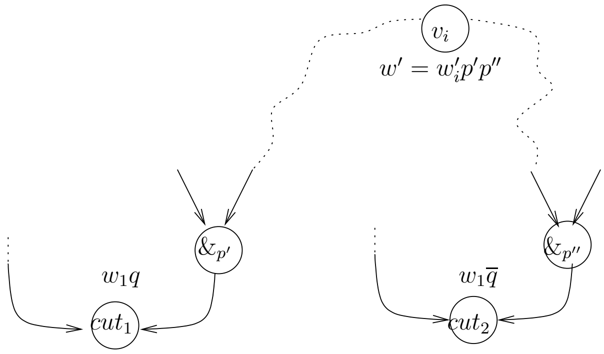


Figure 26:

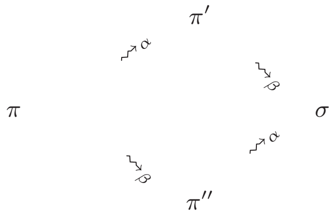


Figure 27: Diamond composition

Case 2 - Otherwise, if α or β is a reduction step involving a &-node as a cut premise node, then we split our reasoning in two main sub-cases:
Case 2.1 - α or β is a ready cut $\&/\oplus_i$. In these cases we get the confluence to σ by (possibly) two more single reduction steps, like in Figure 27. First, observe that the diamond composition can be partial, in the case that both α and β concern two $\&/\oplus_i$ cuts s.t. one of them depends on the eigen weight of the &-node that is premise of the other cut. Second, observe the case when α is a reduction step of a $(C/\&_p)$-cut link L_1 and β is a reduction step of a $(C/\&_p q)$-cut link L_2, with L_1 depending on q and L_2 depending on p, is excluded by the correctness of π.
Case 2.2 - Both α and β are two commutative reduction steps, then of course at least one of them must be a commutative reduction $C/\&$. (Observe the case when α is a reduction step $C_q/\&_p$ (with C depending on q) and β is a reduction step $C_p/\&_q$ (with C depending on p) is excluded by the correctness of π. We consider the remaining sub-cases:

1. Assume α and β concern the reduction of, respectively, cut L_1 and cut L_2 of Figure 28. Then, the two sequences of reductions, $\pi \rightsquigarrow_\alpha \pi_1 \rightsquigarrow_\beta \pi'_1$ of Figure 29 and $\pi \rightsquigarrow_\beta \pi_2 \rightsquigarrow_\alpha \pi'_2$ of Figure 30, converge to $\pi'_1 = \sigma$ by means of a sequence of two axiom reductions starting from π'_2 (i.e., $\pi'_2 \rightsquigarrow^*_{AX} \pi'_1 = \sigma$).

2. Assume α and β concern the reductions of, respectively, cut L_1 and cut L_2 of Figure 31. Then, the two reduction sequences, $\pi \rightsquigarrow_\alpha \pi_1 \rightsquigarrow_\beta \pi'_1$ of Figure 32 and $\pi \rightsquigarrow_\beta \pi_2 \rightsquigarrow_\alpha \pi'_2$ of Figure 33, can converge in to $\pi'_2 = \sigma$ by means of two more commutative reductions, $C_{p'}/C_{p'}$ and $C_{p''}/C_{p''}$ applied to π'_1 (i.e., $\pi'_1 \rightsquigarrow^*_{(C/C)} \pi'_2 = \sigma$). We omit the simpler particular sub-case when the reduction β concerns a commutative C_p/C_p reduction whose weight w_2 depends on q.

3. Assume α and β concern the reductions of, respectively, cut L_1 and cut L_2 of Figure 34. First observe that,

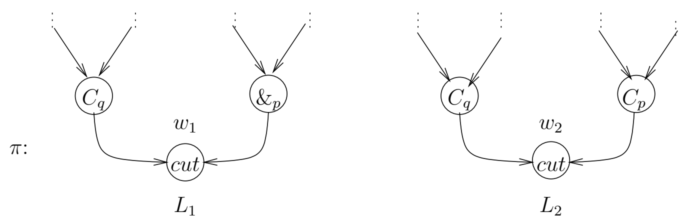


Figure 28:

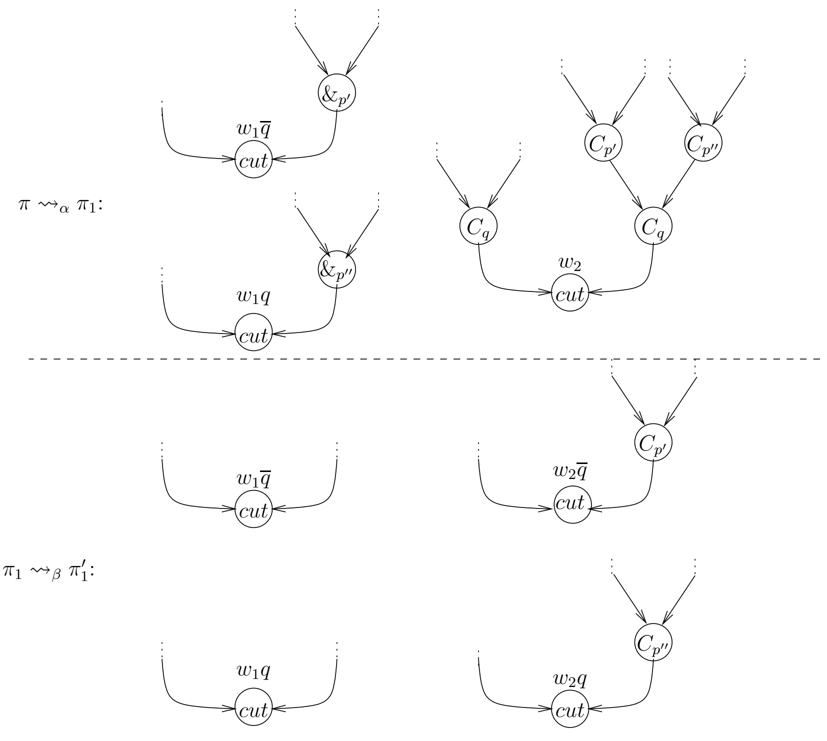


Figure 29:

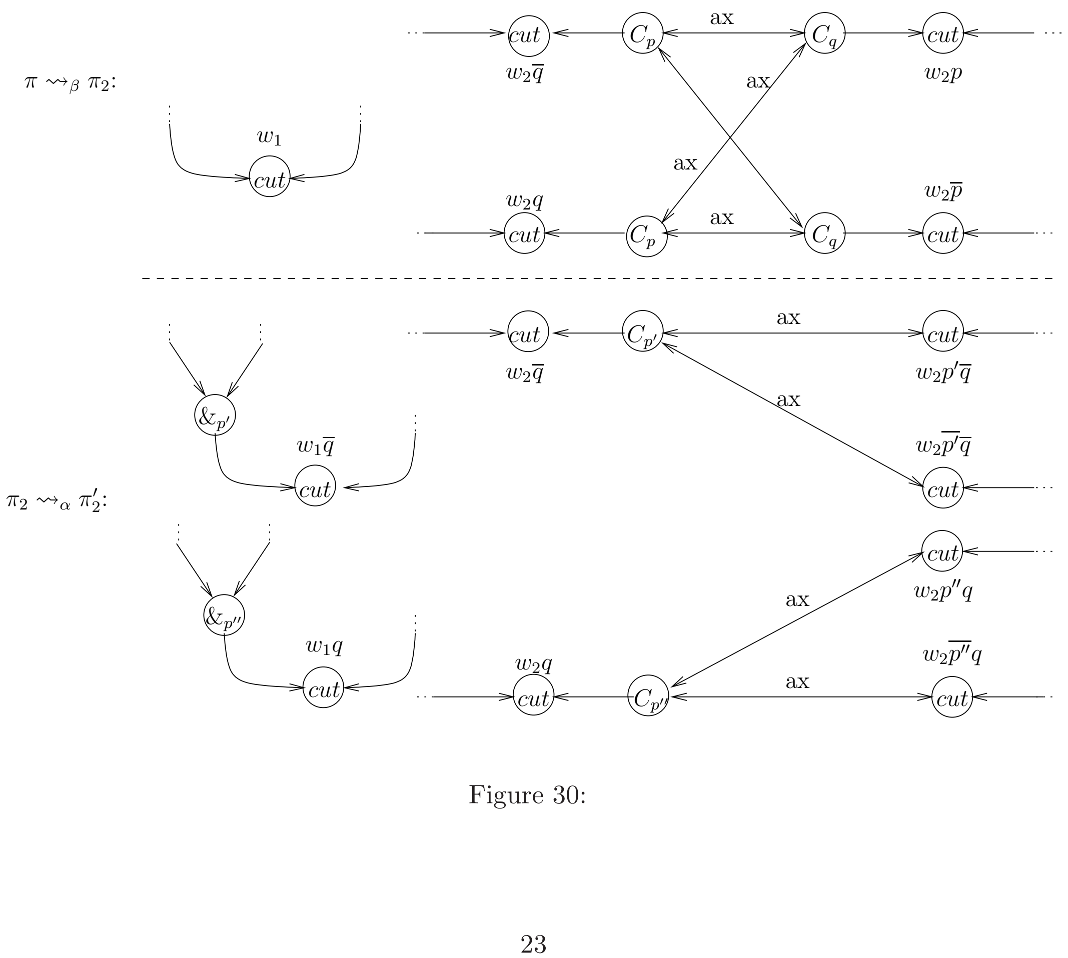


Figure 30:

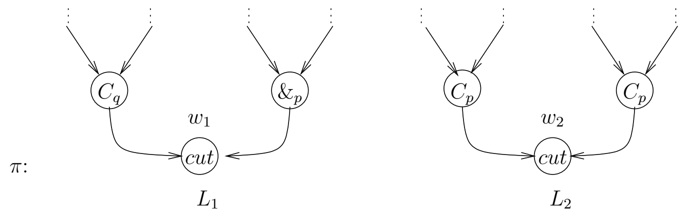


Figure 31:

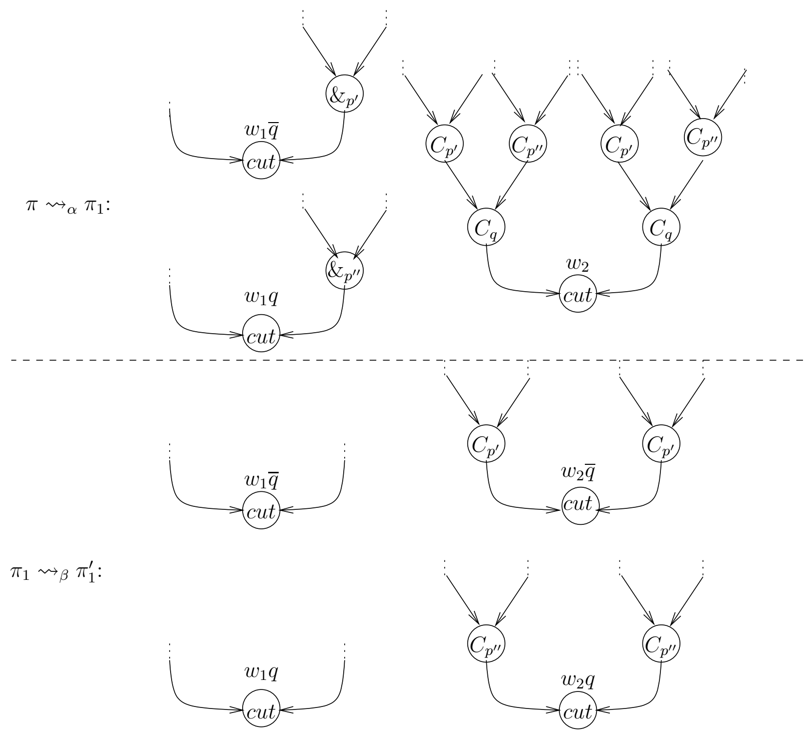


Figure 32:

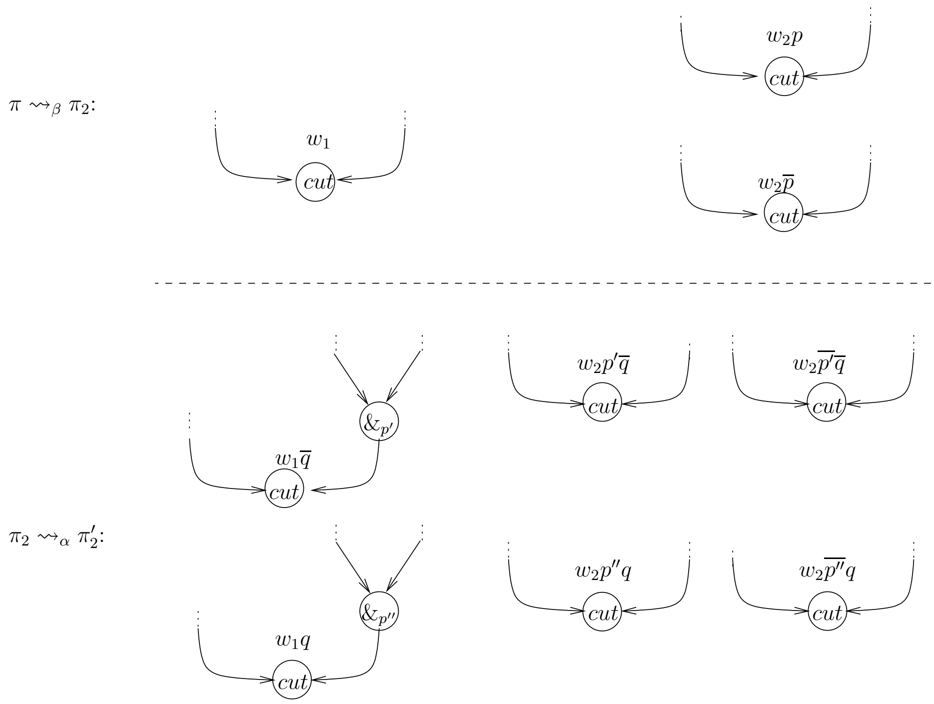


Figure 33:

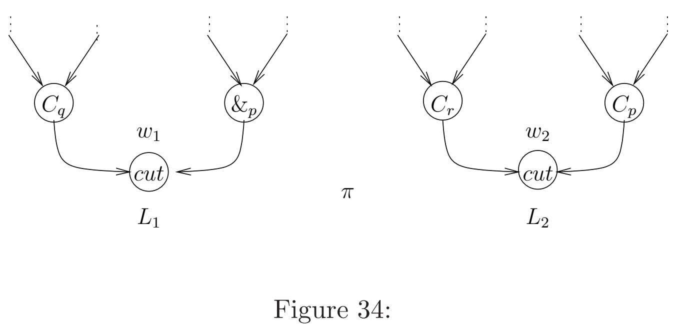


Figure 34:

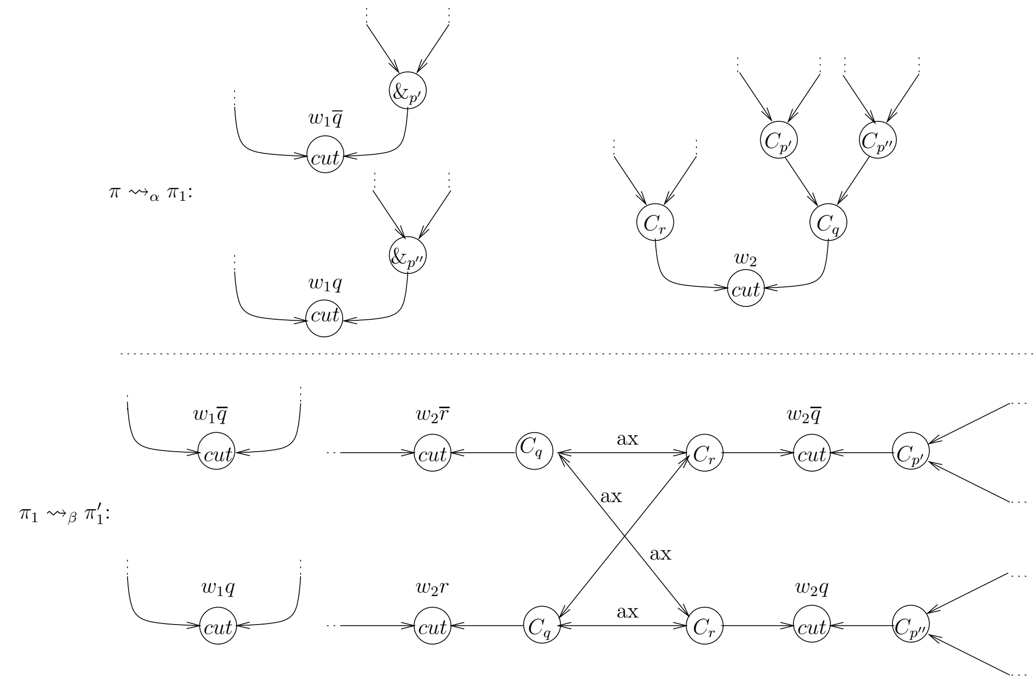


Figure 35:

by the dependency condition 4 of Definition 1, the $\&_r$-node cannot occur in the empire $\mathcal{E}(p)$, so the reduction α does not implies any duplication of the eigen weight r in π'. Now, consider the two reduction sequences, $\pi \rightsquigarrow_\alpha \pi_1 \rightsquigarrow_\beta \pi_1'$ of Figure 35 and $\pi \rightsquigarrow_\beta \pi_2 \rightsquigarrow_\alpha \pi_2'$ of Figure 36. Clearly π_1' converges to $\pi_2' = \sigma$ by means of two more commutative reduction steps, $(C_r/C_{p'})$ and $(C_r/C_{p''})$, modulo some some axiom reductions. We omit the particular (simpler) sub-cases when the weight w_2 of cut L_2 depends on q.

4. In all the remaining cases, illustrated in

   (a) Figure 37,

   (b) Figure 38,

   (c) Figure 39

   the confluence is obtained as simple diamond composition (see Figure 27).

□

Example 2 *According to our reduction system rules, the problematic (non confluent) reduction of the commutative cut of π of Figure 9 (see Section 3.2.1), has got now a (unique solution): the confluent reduction sequence is illustrated by Figure 40 and Figure 41.*

Theorem 7 (strong normalization) *Cut-elimination is strong normalizing.*

Proof — Weak-normalization (Theorem 5) and confluence (Theorem 6) imply strong normalization. □

4 Proof-structures with explicit n-ary contraction links

Proof-structures with explicit n-ary contraction links differ from those one defined in Definition 1 only for the use of n-ary contractions (defined in Figure 42) instead of the binary ones. Then, the notion of slices and switchings (Definitions 2) and proof-nets (Definitions 3) remain unchanged as well the (De-)Sequentialization Theorem 1.

Remark 3 (binary syntax vs. n-ary syntax for C-links) *Observe that in general the syntax with n-ary contraction is not equivalent to the former one with binary contraction, at least for the following two reasons:*

1. *Figure 43 shows an example of proof-net with an n-ary contraction link whose monomial sum, $\overline{pqr} + \overline{p}r + q\overline{r} + p\overline{q} + pqr$, cannot be associated in such a way to reduce this n-ary contraction into a sum of only monomial binary sums (contractions);*

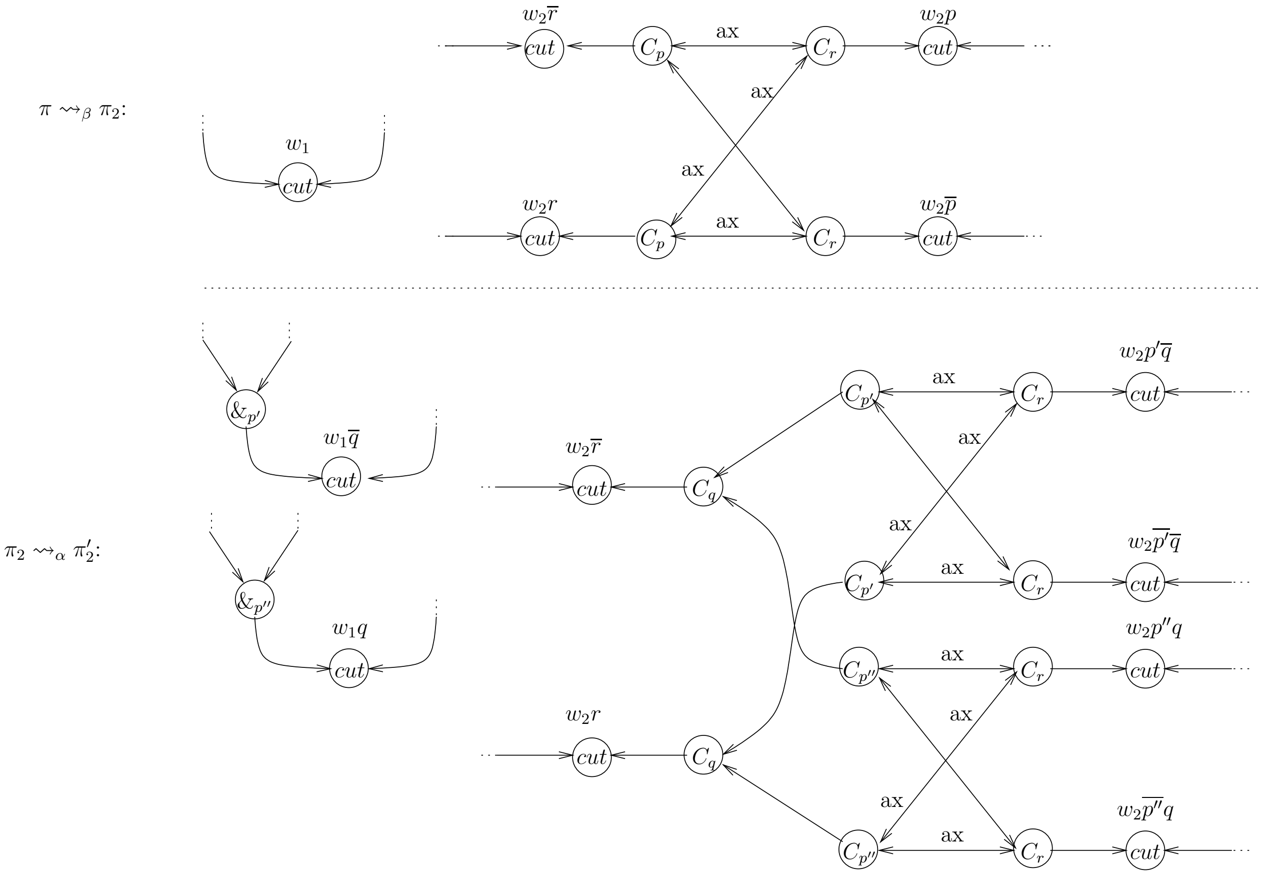


Figure 36:

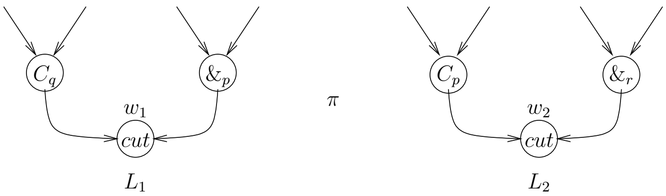


Figure 37:

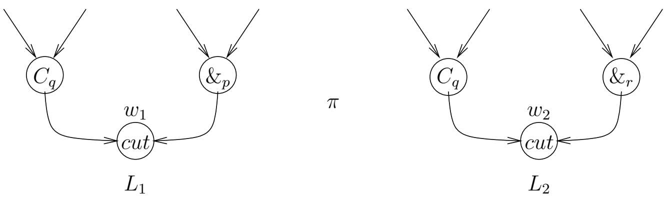


Figure 38:

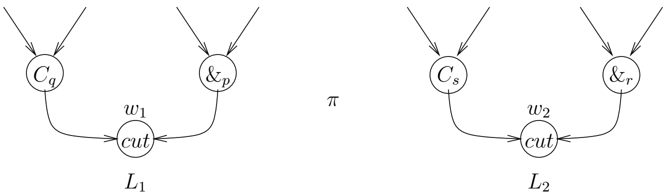


Figure 39:

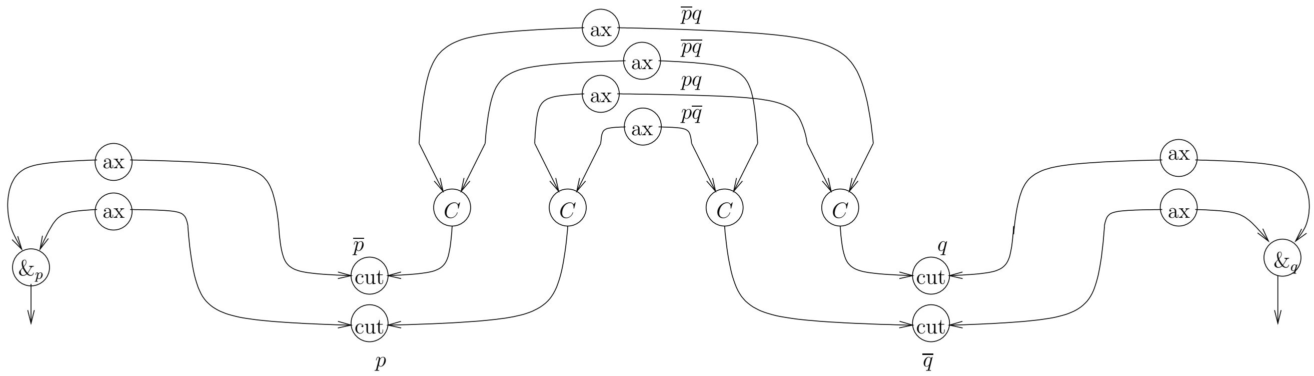


Figure 40:

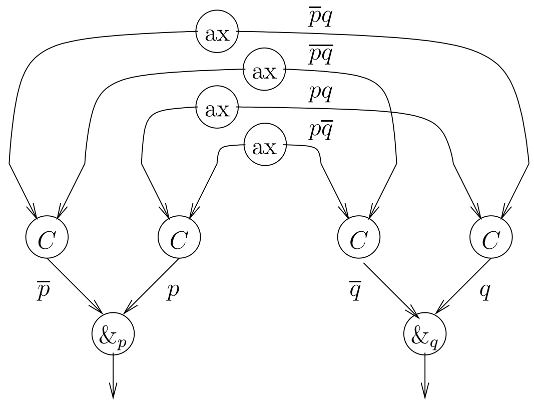


Figure 41:

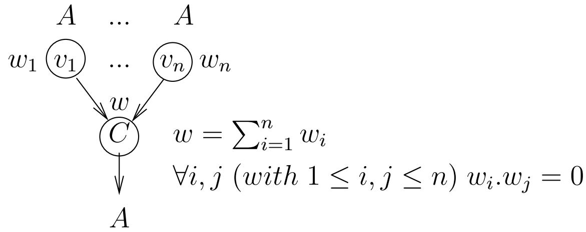


Figure 42: n-ary contraction link

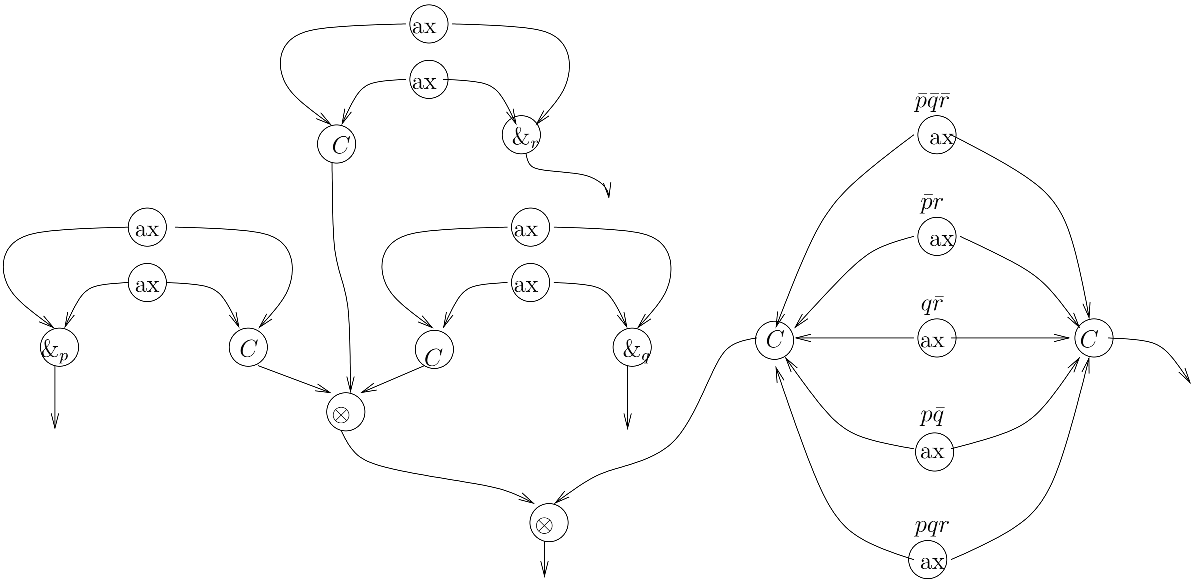


Figure 43: Proof-net with n-ary contraction

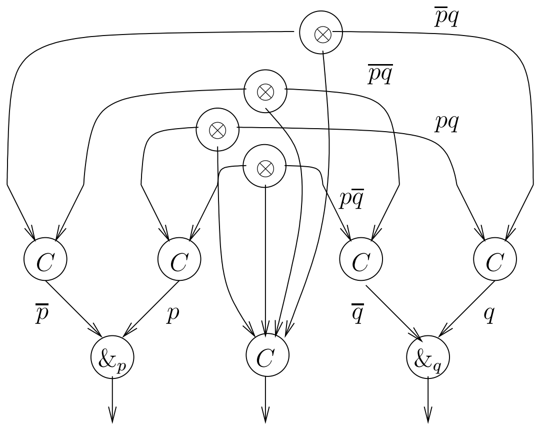


Figure 44: Proof-net with n-ary contraction

2. *moreover Figure 44 shows an example of proof-net that as soon its n-ary contraction is factorized into only binary contraction, the resulting proof-net will quotients less proofs than the previous one (in other words the new proof-net has less sequentialization than the original one with the n-ary contraction.*

4.1 Cut-elimination

We only consider here the reduction steps of commutative cuts, since the reduction of ready-cuts remains unchanged w.r.t. that one of Section 3.1

- the $C/\otimes$-cut of Figure 45, reduces in one step like in Figure 46.
- the C/C-cut, like in Figure 47, reduces in one step like in Figure 48 and we keep only those nodes and edges with nonzero weight.
- the $C/\oplus$-cut of Figure 49 is reduced in a way that is a trivial generalization of the simpler one illustrated in Figure 16.
- the $C/\&$-cut of Figure 50 reduces in one step like in Figure 51, where $w_i.[\mathcal{E}^i_p]$, $1 \leq i \leq n$ is obtained by spreading w_i over a copy of the empire of p, $\mathcal{E}i_p$, and replacing in it each eigen weight with a new (fresh) one (observe that $w = \sum_{i=1}^n w_i$ and by the dependency condition it must be $\forall j, w \subset v_{j=1,\ldots m}$, i.e. there exists a prefix σ s.t. $w\sigma = v_j$, therefore $v_j w = v_j$).

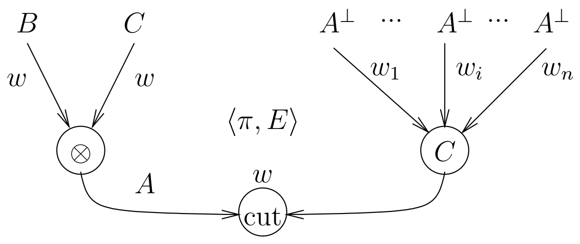


Figure 45:

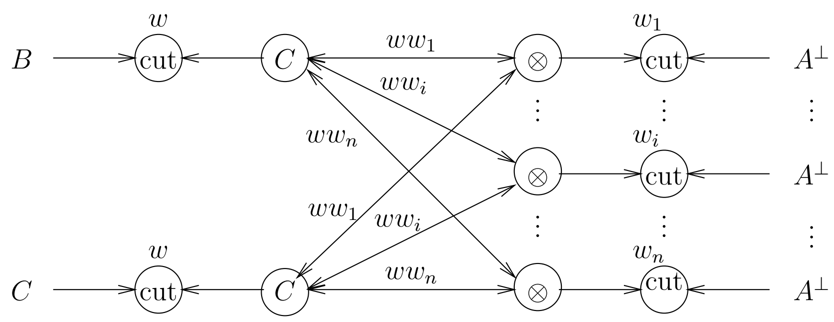


Figure 46:

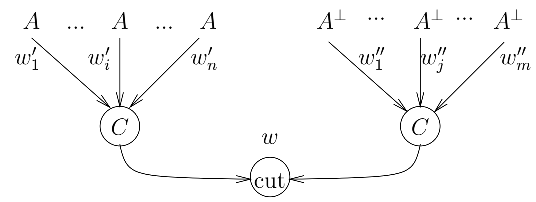


Figure 47:

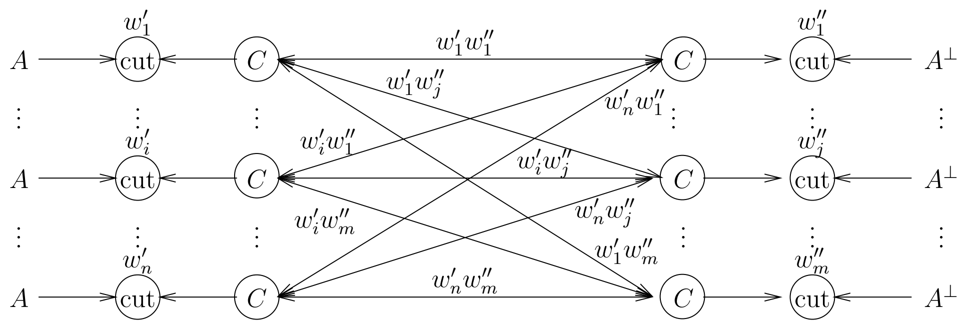


Figure 48:

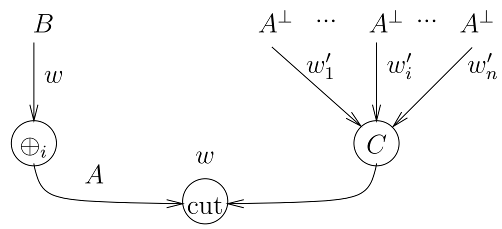


Figure 49:

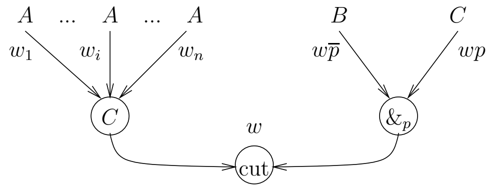


Figure 50:

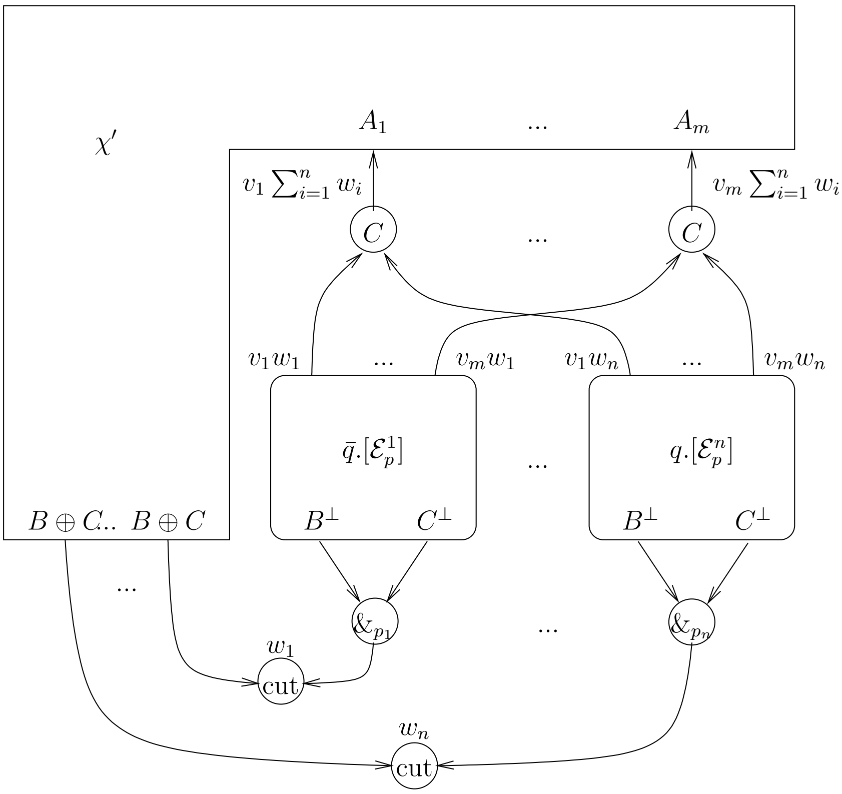


Figure 51:

4.2 Stability

Naively, we show that if there exists an extensive switching $S(\pi)$, with n extensive jumps. that is not ACC, then (by a reasoning like in Theorem 4) we can find an other non ACC switching $S'(\pi)$ with a number of extensive jumps strictly smaller than n. So, by iterating t reasoning for S' we get (t the end, when $n = 0$) a conservative switching $S*(\pi)$ (the base of the induction) that is not ACC, contradicting the (analogous) of Theorem 3.

4.3 Strong cut-elimination

Similar to Theorem 7.

References

[DR89] Danos, V. and Regnier, L. *The Structure of Multiplicatives.* Archive for Mathematical Logic, 1989, 28:181-203.

[Gir87] Girard, J-Y. *Linear Logic.* Theoretical Computer Science, 50, pp.1-102, 1987.

[Gir96] Girard, J-Y. *Proof-nets: the parallel syntax for proof theory.* Logic and Algebra. Marcel Dekker, 1996.

[Gir06] Girard, J-Y. *Le point aveugle.* Cours de Logique. Volume I, Vers la Perfection. Ed. Hermann, 2006, Paris.

[HVG03] Hughes, D. and Van Glabbeek, R. *Proof Nets for Unit-free Multiplicative-Additive Linear Logic.* In Proc. of IEEE Logic in Computer Science, 2003.

[Lau99] Laurent, O. *Polarized Proof-Nets: Proof-Nets for LC (Extended Abstract).* In J.-Y. Girard, editor, Typed Lambda Calculi and Applications 1999, LNCS 1581, pp. 213-227. Springer-Verlag. Avril 1999.