Proof Nets with Explicit Negation for Multiplicative Linear Logic

Quintijn Puite*

November 24, 1998

Abstract

Multiplicative linear logic (**MLL**) was introduced in [Gi87] as a one-sided sequent calculus: linear negation is a notion that is *defined*, via De Morgan identities. One obtains *proof nets* for **MLL** by identifying derivations in the one-sided calculus that are equal up to a permutation of inference rules.

In this paper we consider a similar quotient for the formulation of **MLL** as a *two-sided* sequent calculus: to the usual set of links we add links also for the *left* rules. As a consequence, negation need no longer be defined, but can be treated as a basic connective.

We develop the fundamental theory (substructures, empires and sequentialization) for this variation on the notion of proof net, and show how to obtain Girard's sequentialization theorem for the standard proof nets in one-sided sequent calculus as a corollary.

*`puite@math.uu.nl` URL: `http://www.math.uu.nl/people/puite/`

Contents

1 Introduction

Proof structures for multiplicative linear logic are usually defined as the smallest set containing axiom-links $\overline{a^{\perp} \qquad a}$ and closed under disjoint union and under the lower attachment of the links $\dfrac{a \qquad b}{a \otimes b}$, $\dfrac{a \qquad b}{a \parr b}$ and $\underline{a^{\perp} \qquad a}$ (cf. [Gi87]). The definition of substructure is a bit intricate, and the existence of empires does not follow from an easy argument. In this paper we generalize the notion of proof structure in two directions.

First, we allow proof structure to have open hypotheses (cf. [Da88]); roughly said, we consider the smallest set containing sole formulas, and closed under disjoint union and under the (lower and upper) attachment of links like $\overline{a^{\perp} \qquad a}$, $\dfrac{a \qquad b}{a \otimes b}$, et cetera. This makes the definition of substructure more natural. Since a formula may be considered as a proof net having itself as a conclusion (a is a proof net of $a \vdash a$), the existence of empires is immediate.

Furthermore, this generalization enables us to introduce links corresponding to left rules (e.g. $\dfrac{a \otimes b}{a \qquad b}$) as well as links corresponding to 'new' connectives, e.g. to $\cdot \multimap \cdot$ (definable as $(\cdot)^{\perp} \parr \cdot$), with link $\dfrac{b}{a \qquad a \multimap b}$.

In this generalized setting the Danos-Regnier correctness criterion continues to define the subset of proof nets. Contrary however to the one-sided calculus we can forget about the De Morgan quotient on formulas, since the De Morgan laws have become provable in our proof net syntax.

We are very close to natural deduction now: the attachment of $\dfrac{a \qquad a \multimap b}{b}$ to two disjoint proof nets is actually $\multimap$-elimination, while the attachment of $\dfrac{b}{a \qquad a \multimap b}$ to a proof net with conclusion b and open hypothesis a is $\multimap$-introduction, where a is canceled.

The translation of a derivation into a proof net again is defined inductively. However, the direction of this inductive definition is not any longer from up (the axiom-links) to down, but from middle (the axiomatic *formulas*) to border (the hypotheses and the conclusions). The right rules translate in links going down, as usual, but the left rules are mapped to links going up.

There is an obvious embedding of the usual proof structures into the new ones, viz. the identity map. However, this map does not preserve the meaning of the links. E.g., an axiom-link becomes a right negation-link in the new setting, while a cut-link becomes its dual: a left negation-link. Nevertheless we do obtain Girard's sequentialization theorem for the standard proof nets in one-sided sequent calculus as a corollary, by means of a 'meaning preserving' translation.

2 Proof structures of MLL

Starting from an infinite denumerable set of atoms $p_1, p_2, p_3, \ldots$, the *syntactic formulas* of multiplicative linear logic (**MLL**) are built up with the unary connective $(\cdot)^\perp$ and with the binary connectives $\otimes$, $⅋$ and $\multimap$.

Let S be a multiset of syntactic formulas (e.g. $[a, a, a, b, \ldots]$), i.e. a set of occurrences of syntactic formulas $\{[a]_1, [a]_2, [a]_3, [b]_1, \ldots\}$. From now on we will use the word *formula* for such an occurrence of a syntactic formula, so S is a set of formulas. We write[1] $x = y$ when the formulas $x = [a]_i$ and $y = [b]_j$ are the same, i.e. when $a = b$ and $i = j$.

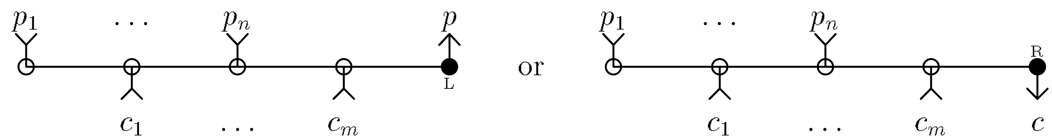
We define a *link* L in S to be a quadruple

$$\langle P_{\text{in}}, P_{\text{out}}, C_{\text{in}}, C_{\text{out}} \rangle$$

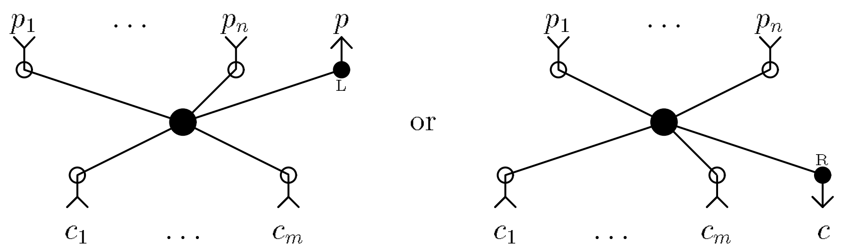
of multisets of formulas of S, with the restriction that $P_{\text{out}} \cup C_{\text{out}}$ consists of exactly one formula, called the *main* formula or the *output* formula of L. The remaining formulas (in $P_{\text{in}} \cup C_{\text{in}}$) are the *active* formulas or the *input* formulas of L. Let $|L|$ stand for the total number of formula occurrences in L.

The formulas in $P_{\text{in}} \cup P_{\text{out}}$ are the *premises* of L, and the formulas in $C_{\text{in}} \cup C_{\text{out}}$ are the *conclusions* of L. When the output formula of $P_{\text{out}} \cup C_{\text{out}}$ is actually in P_{out}, the link is called a *left* link; in this case the conclusions are called *positive* active formulas of L, and the remaining premises are called *negative*. Dually, when the output formula is in C_{out}, the link is called a *right* link; now the premises are called *positive*, and the remaining conclusions are called *negative*. See fig. 1 for an overview.

Links can be graphicly represented by a picture like



or (contracting the horizontal bar)



but one has to keep in mind that P_{in} and C_{in} are orderless, and moreover that it is not a priori the case that all occurring formulas $[a]_i$ are different. Recall that $|L| = n + m + 1$ for those links.

[1] Of course we will usually denote a formula $[a]_i$ by its underlying syntactic formula a.

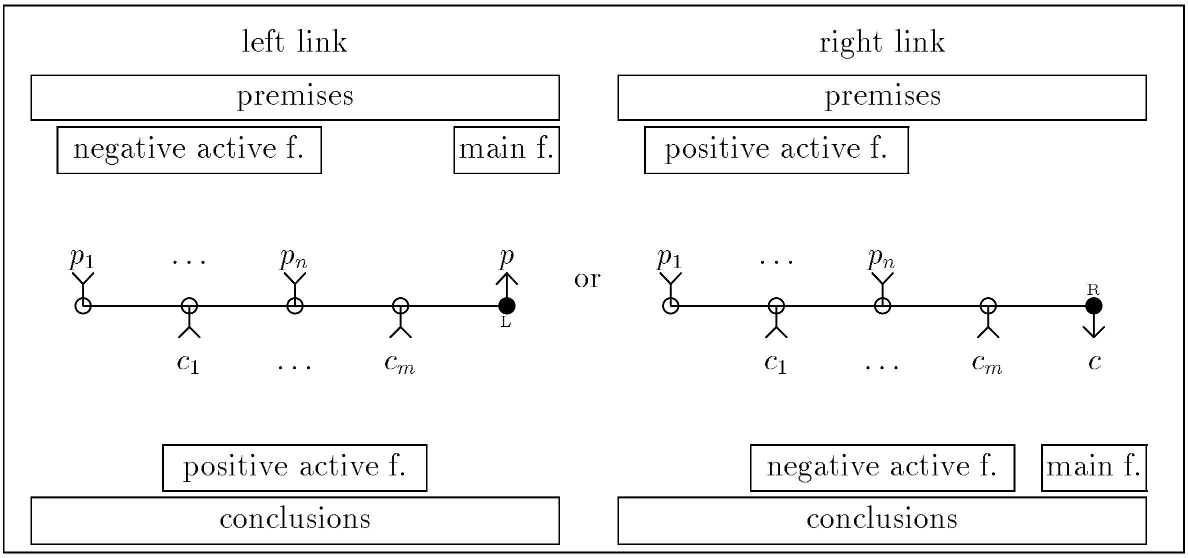
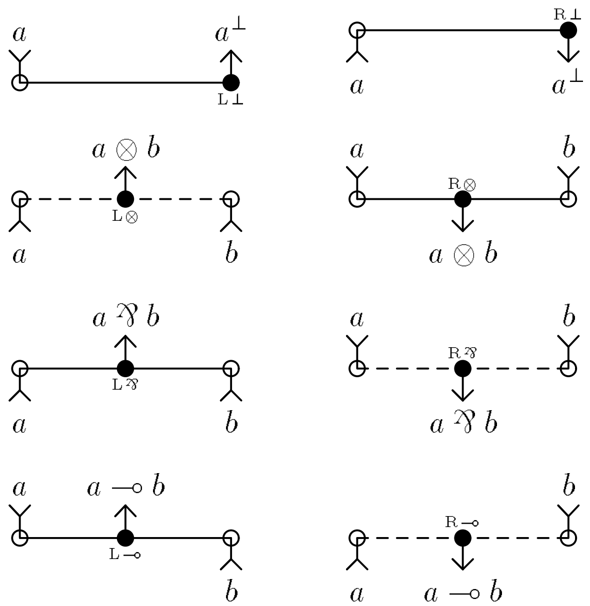


Figure 1: Predicates on the formulas of a link $L = \langle P_{\text{in}}, P_{\text{out}}, C_{\text{in}}, C_{\text{out}} \rangle$.

Definition 2.1 A *(multiplicative) proof structure (with hypotheses)* $\langle S, \mathcal{L} \rangle$ consists of a finite set S of formulas together with a set $\mathcal{L}$ of links in S of the following forms:



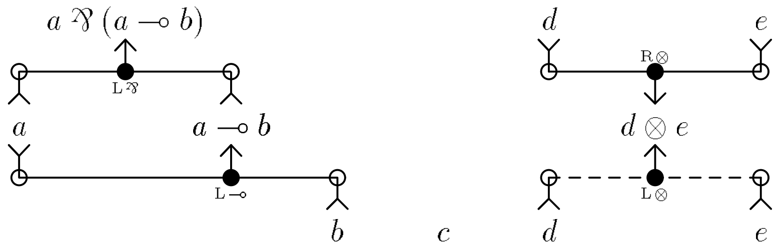
such that the following holds:

1. every formula of S is at most once a conclusion of a link;
2. every formula of S is at most once a premise of a link.

♦

Given a formula a of a proof structure $\mathcal{S}$, we denote by L_a (L^a) the unique link a is a conclusion (premise) of, if it exists.

Example 2.2 The following is an example of a proof structure $\mathcal{S}_0$ with four links:



♦

Observe that in a proof structure the links have the property that the main formula is obtained as a connective applied to the active formula(s); hence the length of formulas increases along the direction of the arrows in our proof structure. Moreover it is clear that every link is of exactly one of the mentioned forms: first determine whether it is a left or right link, and then differentiate on the main connective of the main formula.

The formulas which are not the conclusion of a link are the *hypotheses* H of $\mathcal{S} = \langle S, \mathcal{L} \rangle$, while those that are not the premise of a link are the *conclusions* Q of $\mathcal{S}$. This is also expressed by saying that $\mathcal{S}$ is a proof structure of the *sequent* $H \vdash Q$. Being subsets of S, H and Q are multisets of syntactic formulas. E.g. the proof structure $\mathcal{S}_0$ of example 2.2 is a proof structure of $a \parr (a \multimap b), c, d, e \vdash b, c, d, e$.

A formula that is a hypothesis or a conclusion of $\mathcal{S}$ is called a *leaf* of $\mathcal{S}$; otherwise it is called an *internal formula* of $\mathcal{S}$. A formula of $\mathcal{S}$ is *unlinked* when it is both hypothesis and conclusion of $\mathcal{S}$, i.e. if it is neither a conclusion of a link, nor a premise of a link. $\mathcal{S}_0$ consist of 10 formulas, 4 hypotheses, 4 conclusions, 7 leaves, 1 unlinked formula (c) and 3 internal formulas.

Alternatively, we may partition the formulas of $\mathcal{S}$ according to the number of links a formula is the main formula of. If a formula is not the main formula of any link, it is called an *axiomatic* formula; if it is the main formula of two links, it is a *cut* formula; in all other cases it is a *flow* formula. In $\mathcal{S}_0$ $d \otimes e$ is a cut formula; $a \multimap b$ and $a \parr (a \multimap b)$ are flow formulas, and all remaining formulas (a, b, c, d, e, d, e) are axiomatic formulas.

In fig. 2 we have abstracted all these predicates, where $\Box$ is one of the four connectives. Observe that in case of an axiomatic internal formula the two links L_1 and L_2 may coincide, which would imply that we are dealing with a (left or right) $\multimap$-link, the only form having positive and negative active formulas. In the other three cases of an internal formula the two links L_1 and L_2 are different, because of monotonicity (in the

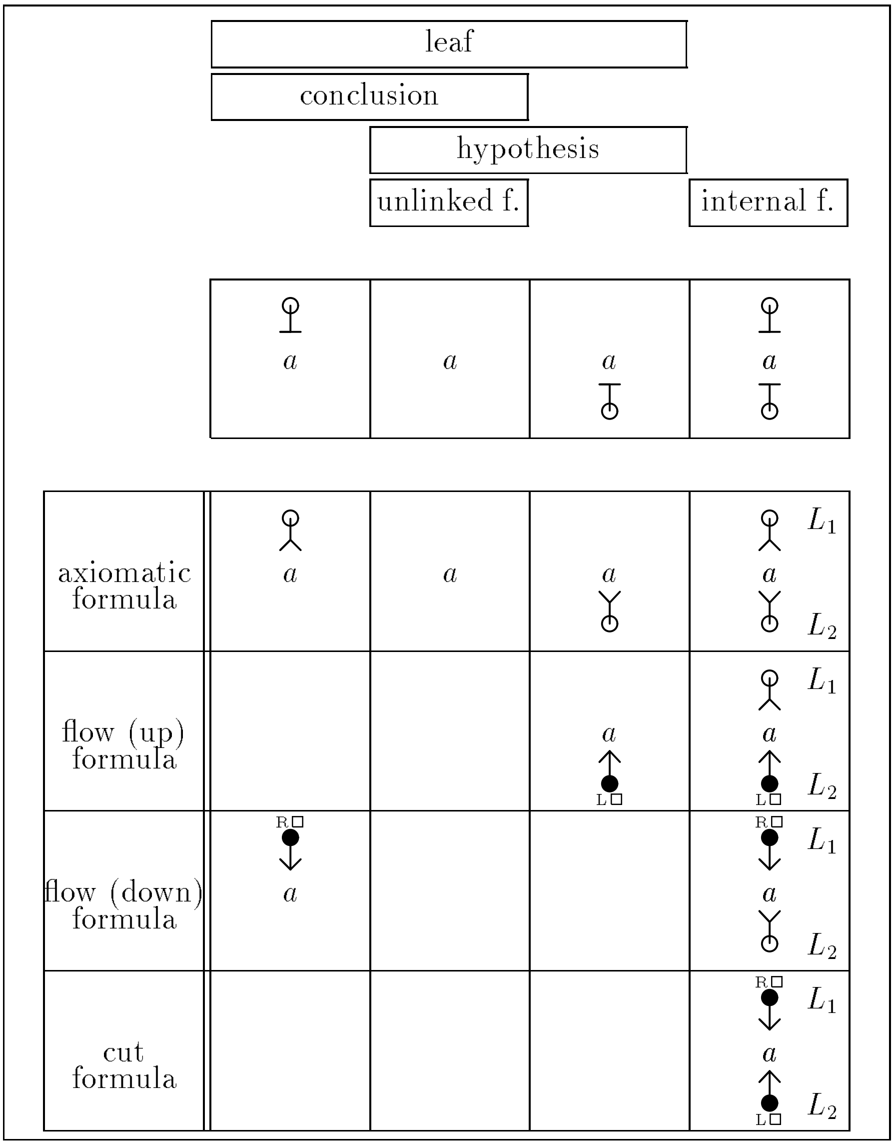
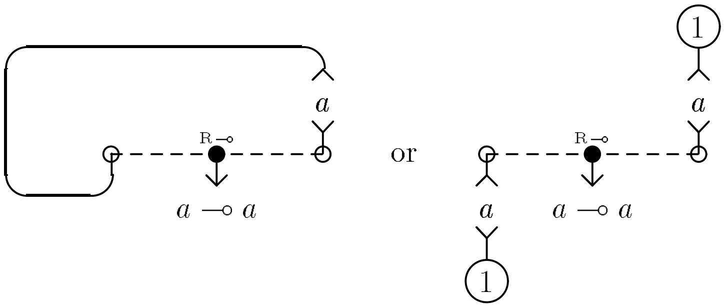


Figure 2: Predicates on the formulas of a proof structure $\mathcal{S} = \langle S, \mathcal{L} \rangle$.

case of a flow internal formula) or because of the fact that each link has only one main formula (in the case of a cut internal formula).

Every non-axiomatic formula is the main formula of at least one link, and hence is non-atomic, and its main connective corresponds to the link. In particular a cut formula is the main formula of a R$\Box$-link and a L$\Box'$-link, where $\Box$ and $\Box'$ must coincide. We call such links *dual* links.

From the requirements it follows that the conclusions of a link are distinct formulas; otherwise a formula would be more than once a conclusion of a link. The same holds for the premises of a link. However, a formula can be premise and conclusion of one and the same link, viz. (as mentioned) in case of a $\multimap$-link. We will indicate this in our graphic representation as follows:



If $\mathcal{S} = \langle S, \mathcal{L} \rangle$ has $S = \emptyset$, there can be no link in S, since a link has at least one premise or conclusion, viz. the main formula. Hence $\mathcal{S} = \langle \emptyset, \emptyset \rangle$, which we will denote by $\varnothing$.

A *substructure* of a proof stucture $\langle S, \mathcal{L} \rangle$ is a pair $\langle S', \mathcal{L}' \rangle$ such that $S' \subseteq S$ and $\mathcal{L}' \subseteq \mathcal{L}$ and such that all $L \in \mathcal{L}'$ are actually links in S'. Obviously it is a proof structure itself, since every formula of S' is still at most one time a conclusion (premise) of a link in $\mathcal{L}' \subseteq \mathcal{L}$. We write $\mathcal{S}' \subseteq \mathcal{S}$.

Given a collection of substructures $\{\mathcal{S}_i\}$ of a given proof structure $\mathcal{S}$, we can define their intersection and union in the obvious way:

- $\bigcap \mathcal{S}_i := \langle \bigcap S_i, \bigcap \mathcal{L}_i \rangle$, which is actually a substructure of $\mathcal{S}$ again, since given a link $L \in \bigcap \mathcal{L}_i$, it is a link in S_i for each i, i.e. a link in $\bigcap S_i$. In particular, starting from the empty collection, we obtain the substructure $\mathcal{S}$ of $\mathcal{S}$.

- $\bigcup \mathcal{S}_i := \langle \bigcup S_i, \bigcup \mathcal{L}_i \rangle$, which also is a substructure of $\mathcal{S}$ again, since given a link $L \in \bigcup \mathcal{L}_i$, it is a link in S_i for some i, i.e. a link in $\bigcup S_i$. In particular, starting from the empty collection, we obtain the substructure $\varnothing$ of $\mathcal{S}$.

Given a subset $\mathcal{L}'$ of links, the intersection of all substructures of $\mathcal{S}$ containing those links is called the *substructure generated by* $\mathcal{L}'$, denoted by $\langle\!\langle \mathcal{L}' \rangle\!\rangle$. It is a substructure containing $\mathcal{L}'$, and contained in every other substructure containing $\mathcal{L}'$, whence it is the smallest substructure containing $\mathcal{L}'$. Equivalently we may define

$$\langle\!\langle \mathcal{L}' \rangle\!\rangle = \left\langle \bigcup \{ P_{\text{in}} \cup P_{\text{out}} \cup C_{\text{in}} \cup C_{\text{out}} \mid \langle P_{\text{in}}, P_{\text{out}}, C_{\text{in}}, C_{\text{out}} \rangle \in \mathcal{L}' \}, \mathcal{L}' \right\rangle$$

There are two other canonical substructures of a given proof structure $\langle S, \mathcal{L} \rangle$, given the subset of formulas S':

- the *discrete* substructure generated by S':

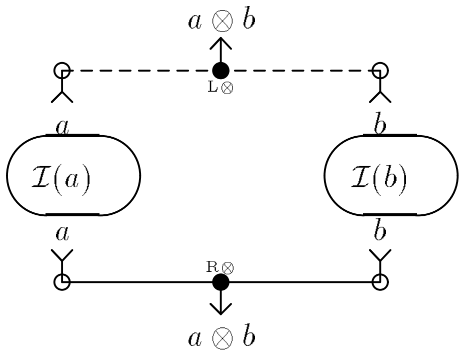
$$\langle S', \emptyset \rangle$$

- the *full* substructure generated by S':

$$\langle S', \{ L \in \mathcal{L} \mid L \text{ is actually a link in } S' \} \rangle$$

The binary links of the form R$\otimes$, L$⅋$ and L$\multimap$ are called *thick*, while the other binary links (L$\otimes$, R$⅋$ and R$\multimap$) are called *thin*. Observe that for every binary connective we have a left and its dual right link, one of the two being thick and the other being thin.

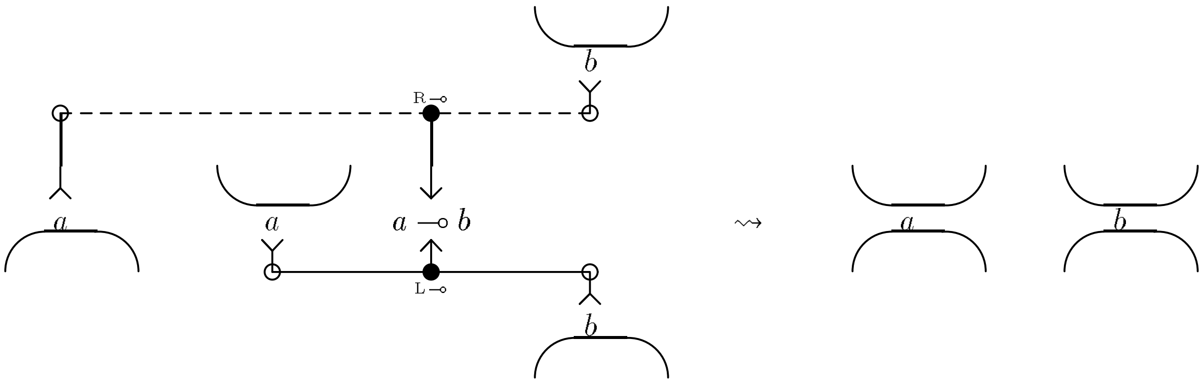
The existence of dual links admits *identity* proof structures $\mathcal{I}(a)$ of $a \vdash a$ for every syntactic formula a using only **atomic** axiomatic formulas (called an η-*expanded* proof structure): for an atom p we can take the proof structure consisting of the sole formula p and no links, while for a complex syntactic formula $a \,\square\, (b)$ we can paste the two dual links L$\square$ and R$\square$ to the inductively obtained identity proof structure(s) for a (and b), as in:

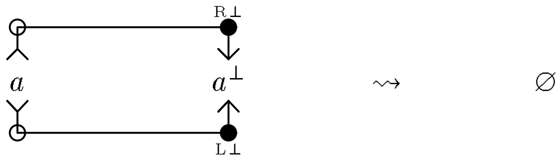


Moreover, the duality of links provides us with a *cut elimination* procedure, where a reduction step is defined in the obvious way: a (necessarily internal) cut formula is the main formula of two dual links L and L^*. Now delete these links and the cut formula, while pairwise identifying the active formulas in case they are different (as occurrence of the same syntactic formula), or deleting them if they are identical. It is clear that

$|\mathcal{L}|$ decreases by 2 (and that $|S|$ decreases by $|L| = |L^*|$), implying that this reduction is noetherian. Moreover it is CR, whence normal forms are unique.

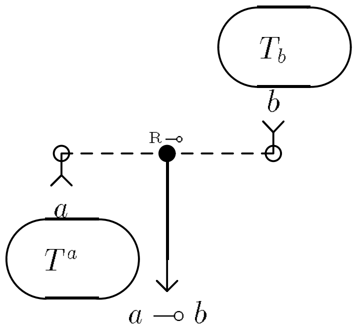
We give two examples:





To each syntactic formula a we can inductively assign two proof structures, called the *upper-* and *lower construction tree* of a. Denoting the set of positive **atomic** subformulas of a by $P(a)$ and the set of negative ones by $N(a)$, the upper construction tree T_a is a proof structure of $P(a) \vdash N(a), a$, while the lower construction tree T^a is a proof structure of $N(a), a \vdash P(a)$.

$T_{a \multimap b} =$

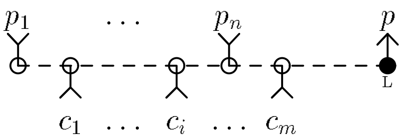


In these trees all formulas are flow formulas, except for the atomic leaves, which are axiomatic. The two trees T_a and T^a of a formula a may be pasted into one proof structure by identifying the atomic leaves. This is another way to obtain the identity proof structure $\mathcal{I}(a)$ of $a \vdash a$.

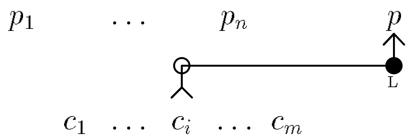
For a *balanced* formula a (i.e. a formula in which each syntactic atom p appears positively the same number of times as it does negatively), we may — in at least one way — pairwise identify such atoms in T_a, in order to get a proof structure of $\vdash a$.

3 Proof nets of MLL

Given a proof structure $\mathcal{S} = \langle S, \mathcal{L} \rangle$, we define a *switching* σ for $\mathcal{S}$ to be a choice for each thin link $L \in \mathcal{L}$ of one of the active formulas of L. Disconnecting for each thin link all the non-chosen formulas from the link, as in



which is turned into



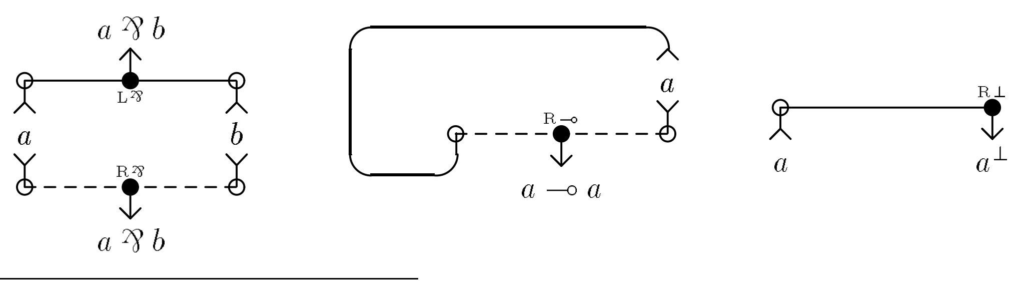
(in case $\sigma(L) = c_i$), we obtain the *correction graph* $\sigma\mathcal{S}$ with vertices the formulas of S and edges as indicated by the graphic representations of the (replaced) links. (It may be instructive to contract the horizontal bar of each link into a point; see page 3.)[2]

For each (binary) thin link there is a choice between the *two* active formulas. Recall that we call $\mathcal{S}$ a proof structure of $X \vdash Y$ if X and Y are the multisets of syntactic formulas that are the hypotheses respectively the conclusions of $\mathcal{S}$.

Definition 3.1 A proof structure $\mathcal{S}$ of $X \vdash Y$ with n **thin** links is a *proof net* iff all its 2^n correction graphs are trees, i.e. non-empty, acyclic and connected. In this case we call $X \vdash Y$ *provable*. A substructure of a proof structure (in particular of a proof net) $\mathcal{S}$ is called a *subnet* of $\mathcal{S}$ iff, regarded as a proof structure, it is a proof net. ♦

Alternatively we can define the notion of proof net by means of a generalization of the long trip condition ([Gi87], [DR89]; see also [Tr92]) or by means of a homological characterization (cf. [Me94]). The only thing we have to do is to treat our thin links as if they were R⅋-links, and our thick links as if they were R⊗-links, while the X⊥-links may be treated as Ax- or Cut-links.

Example 3.2 (Proof nets) Some examples of proof nets are given by

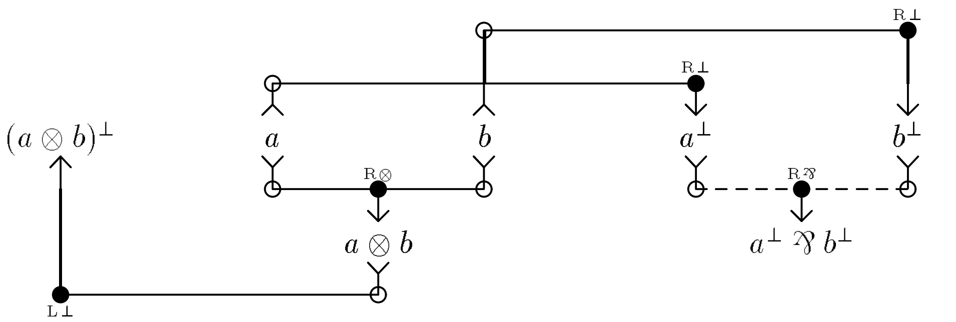


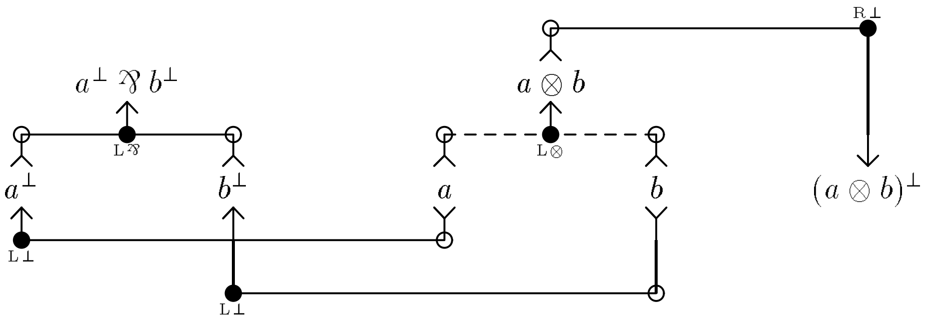
♦

[2] If we would extend the notion of thick and thin links to the (unary) ⊥-link, the notion of switching and correction graph does not depend on whether we consider such a link as a thick link or as a thin link, since there is only one choice for the active formula.

Example 3.3 (Rotatable proof nets)

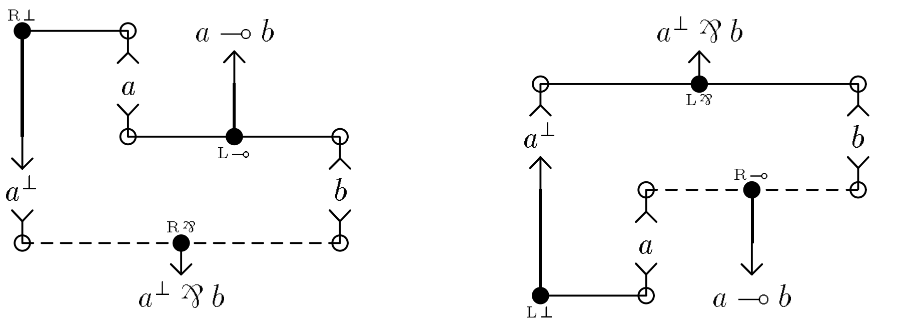
1. Provability of the De Morgan laws follows from the following proof nets.



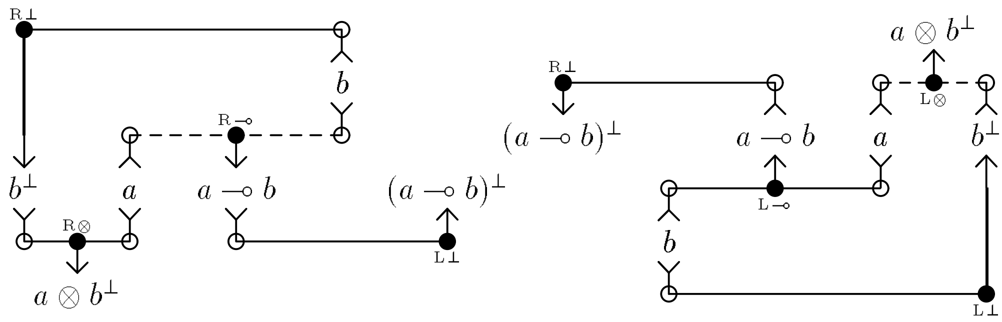


Observe that the second proof net may be obtained from the first one by rotating it, replacing each link by its dual link.

2. The same holds for the next proof nets, that prove the relation between $\multimap$ and ⅋.



3. And also the next pair of proof nets is an example of this phenomenon.



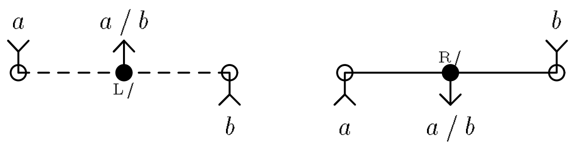
♦

Rotating a proof structure $\mathcal{S}$ of $X \vdash Y$ and replacing each link by its dual link yields a proof structure $\mathcal{S}^*$ of $Y \vdash X$, which we call the *rotation* or *dualization* of $\mathcal{S}$. In the previous example $\mathcal{S}$ and $\mathcal{S}^*$ were proof nets, but in general a proof net will not yield a proof net under this transformation (because $\vdash$ is clearly not symmetric; $X \vdash Y$ does not generally imply $Y \vdash X$). And even if both $X \vdash Y$ and $Y \vdash X$ are provable, it need not be the case that they are provable by a *rotatable* proof net (e.g. in section 7 it will be shown that both $(p \multimap p) \otimes p \vdash p$ and $p \vdash (p \multimap p) \otimes p$ are provable, but not by a rotatable proof net; see example 7.10.3). If it happens to be the case that there is a rotatable proof net of $X \vdash Y$ (and $Y \vdash X$), we will write $X \dashv_{\mathrm{r}}\vdash Y$. We will have a closer look at rotatable proof nets in section 7, where we will show them to be — roughly speaking — the identity proof structures modulo the 'De Morgan laws', associativity and commutativity. Observe that the dualization of the proof structure generated by $\{L\}$ is the proof structure generated by $\{L^*\}$, where L^* is the dual link of L.

We must not confuse this transformation with the transformation on proof nets without $\multimap$-links, that rotates a proof net $\mathcal{S}$ and interchanges $\otimes$ and $\parr$ in all formulas. Since the underlying links remain of the same type (thick or thin), this transformation does always yield a proof net $\mathcal{S}^\perp$, called the *flip-over* of $\mathcal{S}$. Hence provability of $X \vdash Y$ implies provability of $Y[\otimes \leftrightarrow \parr] \vdash X[\otimes \leftrightarrow \parr]$.

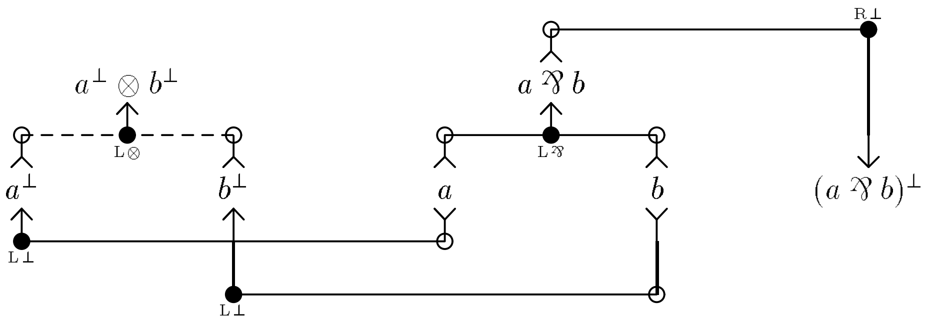
Proposition 3.4 If $\mathcal{S}$ is a $\multimap$-link-free[3] proof net of $X \vdash Y$, then $\mathcal{S}^\perp$ is a proof net of $Y[\otimes \leftrightarrow \parr] \vdash X[\otimes \leftrightarrow \parr]$. ♦

[3]This result may be extended to proof nets with $\multimap$-links when we extend our language with the binary connective $/$, and our admissible links with



Now we define the flip-over of a proof net $\mathcal{S}$ to be the rotation of $\mathcal{S}$ in which we interchange $\otimes$ and $\parr$ as well as $\multimap$ and $/$. The flip-overs of the proof nets of example 3.3.2 show that $a^\perp \otimes b \dashv_{\mathrm{r}}\vdash a \,/\, b$, so that the intuitive meaning of $a \,/\, b$ is 'b but not a'.

Example 3.5 (The flip-over of a proof net) As an example we give the application of this transformation to the first proof net of example 3.3.1:



♦

Let $\mathcal{S}'$ be a substructure of a proof structure $\mathcal{S}$. A switching σ for $\mathcal{S}$ has a unique *restriction* to a switching σ' for $\mathcal{S}'$, and comparing the corresponding correction graphs we observe that $\sigma'\mathcal{S}' \subseteq \sigma\mathcal{S}$. The other way around, given a switching σ' for $\mathcal{S}'$ we may *extend* it to a switching σ for $\mathcal{S}$ by choosing an active formula for each of the remaining thin links, and once again $\sigma'\mathcal{S}' \subseteq \sigma\mathcal{S}$. But this implies that if $\mathcal{S}$ is a proof net, then $\mathcal{S}'$ can not have cyclic correction graphs. Hence a substructure of a **proof net** is a subnet if and only if it is non-empty and all its correction graphs are connected.

Lemma 3.6 Let $\mathcal{S}_1$ and $\mathcal{S}_2$ be subnets of a given proof net $\mathcal{S}$. Then

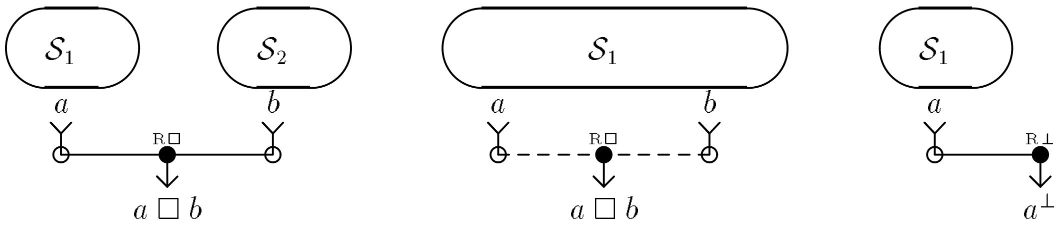
1. $\mathcal{S}_1 \cap \mathcal{S}_2$ is a subnet if and only if $\mathcal{S}_1 \cap \mathcal{S}_2 \neq \varnothing$.

2. $\mathcal{S}_1 \cup \mathcal{S}_2$ is a subnet if and only if $\mathcal{S}_1 \cap \mathcal{S}_2 \neq \varnothing$. ♦

Proof of 1: Suppose $\mathcal{S}_1 \cap \mathcal{S}_2 \neq \varnothing$, which is non-emptyness. Given a switching σ' for $\mathcal{S}_1 \cap \mathcal{S}_2$, extend it to a switching σ for $\mathcal{S}$. Given two vertices $s, t \in S_1 \cap S_2$, between those there is a $\sigma_1\mathcal{S}_1$-path, as well as a $\sigma_2\mathcal{S}_2$-path (where σ_i are the restrictions of σ to $\mathcal{S}_i$). When these two paths would not coincide, they would constitute a cycle in $\sigma\mathcal{S}$, which can not be the case. Hence the path is in $\sigma_1\mathcal{S}_1$ as well as in $\sigma_2\mathcal{S}_2$, so its in $\sigma'(\mathcal{S}_1 \cap \mathcal{S}_2)$, which shows connectedness of the correction graphs.
The other way around is clear since a subnet is non-empty.

Proof of 2: Suppose $\mathcal{S}_1 \cap \mathcal{S}_2 = \varnothing$. Choose $s_i \in S_i$ (possible because the $\mathcal{S}_i$ are proof nets), then those vertices can not be connected in any correction graph of $\mathcal{S}_1 \cup \mathcal{S}_2$, hence $\mathcal{S}_1 \cup \mathcal{S}_2$ is not a proof net.
The other way around, suppose $\mathcal{S}_1 \cap \mathcal{S}_2 \neq \varnothing$, say $s \in S_1 \cap S_2$. Non-emptyness of $\mathcal{S}_1 \cup \mathcal{S}_2$ follows from non-emptyness of S_1. Now let σ' be a switching for $\mathcal{S}_1 \cup \mathcal{S}_2$, which we may restrict to σ_i of $\mathcal{S}_i$. Given two vertices $s_i \in S_1 \cup S_2$: when both are in S_j there is a path in $\sigma_j\mathcal{S}_j$ and hence in $\sigma'(\mathcal{S}_1 \cup \mathcal{S}_2)$; otherwise $s_i \in S_i$, and there are paths from s_i to s in $\sigma_i\mathcal{S}_i$, which may be combined into a single path from s_1 to s_2 in $\sigma'(\mathcal{S}_1 \cup \mathcal{S}_2)$. ■

In case of a binary link X$\Box$ (a link with two active formulas: X $\in$ {L,R} and $\Box \in \{\otimes, \parr, \multimap\}$) we define two operations as follows: for two proof structures $\mathcal{S}_1$ and $\mathcal{S}_2$, the proof structure X$\Box(\mathcal{S}_1, \mathcal{S}_2)$ is obtained by pasting the proof structures $\mathcal{S}_i$ to the corresponding active formulas of the X$\Box$-link, while the proof structure X$\Box(\mathcal{S}_1)$ is obtained by pasting the two active formulas of the X$\Box$-link to $\mathcal{S}_1$. For a unary link ($\Box = \bot$) these two kind of operations collapse into the obvious one: X$\Box(\mathcal{S}_1)$ is obtained by pasting the active formula of the X$\Box$-link to $\mathcal{S}_1$.



R$\Box(\mathcal{S}_1, \mathcal{S}_2)$ R$\Box(\mathcal{S}_1)$ R$\bot(\mathcal{S}_1)$

Proposition 3.7 Let $\mathcal{S}_1$ and $\mathcal{S}_2$ be two (disjoint) proof structures, let X$\Box$ be a link, and let X$\Box(\mathcal{S}_1, \mathcal{S}_2)$ and X$\Box(\mathcal{S}_1)$ be defined as above.

1. In case X$\Box$ is a thick link, then X$\Box(\mathcal{S}_1, \mathcal{S}_2)$ is a proof net if and only if $\mathcal{S}_1$ and $\mathcal{S}_2$ are proof nets.

2. In case X$\Box$ is a thin link, then X$\Box(\mathcal{S}_1)$ is a proof net if and only if $\mathcal{S}_1$ is a proof net.

3. In case X$\Box$ is a unary link, then X$\Box(\mathcal{S}_1)$ is a proof net if and only if $\mathcal{S}_1$ is a proof net. ♦

Proof of 1: Suppose $\mathcal{S}_1$ and $\mathcal{S}_2$ are proof nets. Then every correction graph for $\mathcal{S} :=$ X$\Box(\mathcal{S}_1, \mathcal{S}_2)$ consists in two trees, connected via the X$\Box$-link, whence is itself a tree. Hence $\mathcal{S}$ is a proof net.
The other way around, given a correction graph for $\mathcal{S}_1$, we may extend it to a correction graph for $\mathcal{S}$, which is a tree. Cutting this tree off in the one active formula, we see that our original correction graph was a tree. Hence $\mathcal{S}_i$ is a proof net.

Proof of 2: Suppose $\mathcal{S}_1$ is a proof net. Then a correction graph of $\mathcal{S} :=$ X$\Box(\mathcal{S}_1)$ is just a correction graph of $\mathcal{S}_1$, extended with one edge, hence a tree.
The other way around, pruning a correction graph (tree) of $\mathcal{S}$ yields again a tree. This gives all correction graphs of $\mathcal{S}_1$, which therefore is a proof net as well.

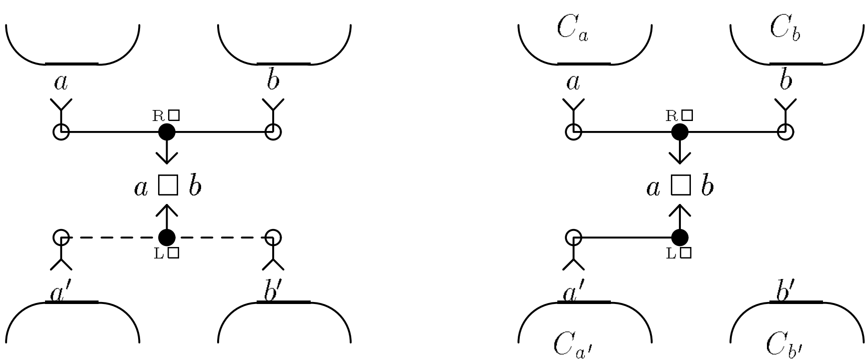
Proof of 3: Immediate. ■

Proposition 3.8 1. The identity proof structure $\mathcal{I}(a)$ of $a \vdash a$ (defined on page 8) is actually a proof net. Moreover, it is invariant under rotation (defined on page 13).

2. Cut elimination is sound, i.e. if $\mathcal{S}$ is a proof net, then so is $\mathcal{S}'$, which is obtained from $\mathcal{S}$ by a reduction step. ⧫

Proof of 1: A sole formula p is a proof net.
For a complex syntactic formula $a \,\Box\, b$ first paste the thick link to the identity proof nets for a and b (yielding a proof net by prop. 3.7.1), and then paste the dual link to the resultant (yielding a proof net by prop. 3.7.2). Rotating this yields the same proof net.
For a complex syntactic formula $a^\perp$ extend the identity proof net for a with the R$\perp$ and L$\perp$ in an arbitrary order, the result of which is also invariant under rotating.

Proof of 2: If the cut formula is $a^\perp$, cut elimination is nothing else but contraction of two edges, which clearly preserves correctness.
In the binary case, let a switching of $\mathcal{S}'$ be given.



Extend it to a switching of $\mathcal{S}$. Deleting the two concerning links L and L^* in our correction graph, we obtain three components (trees again). C_a, C_b and $C_{a'}$ are pairwise disjoint, and changing the switching in L these components does not change, whence we see that also C_a, C_b and $C_{b'}$ are pairwise disjoint. Hence $C_{a'} = C_{b'}$. Now it is clear that the original correction graph of $\mathcal{S}'$ is C_a, (in one point) pasted to $C_{a'} = C_{b'}$, (in one point) pasted to C_b, which is a tree again. ■

Suppose $X \vdash Y$ is provable, i.e. there exists a proof net $\mathcal{S}$ of $X \vdash Y$. Applying cut elemination, we obtain a unique normal (i.e. cut-free) proof structure $\widehat{\mathcal{S}}$ of $X \vdash Y$ which is again a proof net. Hence a sequent $X \vdash Y$ is provable iff there exists a cut-free proof net of $X \vdash Y$.

Definition 3.9 A *derivation rule* of **MLL** is an expression

$$\frac{H_1 \vdash Q_1 \quad \ldots \quad H_n \vdash Q_n}{H \vdash Q}$$

indicating that from provability of $H_1 \vdash Q_1$ up to $H_n \vdash Q_n$ it follows that $H \vdash Q$ is provable. (Remind that we call $X \vdash Y$ provable if there is a proof structure of $X \vdash Y$ which is actually a proof net; see def. 3.1.) A derivation rule with $n = 0$ is called an *axiom rule.* ⧫

We will only consider derivation rules that are *constructive*, in the sense that they appear accompanied with a witness RULE that transforms the given proof nets $\mathcal{S}_i$ of $H_i \vdash Q_i$ into a proof net RULE$(\mathcal{S}_1, \ldots, \mathcal{S}_n)$ of $H \vdash Q$. As this operation RULE only yields proof structures that are a proof net, we call it a *sound* operation.

From the definition it follows that derivation rules may be composed to form other derivation rules. A *derivation* is a closed composite of rules. Moreover, every proof net of some sequent $X \vdash Y$ yields an axiom rule

$$\frac{}{X \vdash Y}$$

Lemma 3.10 The following are derivation rules of **MLL**.

Identity rules:

$$\frac{}{a \vdash a}\,\text{Ax}$$

$$\frac{X_1 \vdash a, Y_1 \qquad X_2, a \vdash Y_2}{X_1, X_2 \vdash Y_1, Y_2}\,\text{Cut}$$

Negation rules:

$$\frac{X \vdash a, Y}{X, a^\perp \vdash Y}\,\text{L}\perp \qquad\qquad \frac{X, a \vdash Y}{X \vdash a^\perp, Y}\,\text{R}\perp$$

Multiplicative logical rules:

$$\frac{X_1 \vdash a, Y_1 \qquad X_2 \vdash b, Y_2}{X_1, X_2 \vdash a \otimes b, Y_1, Y_2}\,\text{R}\otimes \qquad\qquad \frac{X, a, b \vdash Y}{X, a \otimes b \vdash Y}\,\text{L}\otimes$$

$$\frac{X_1, a \vdash Y_1 \qquad X_2, b \vdash Y_2}{X_1, X_2, a \parr b \vdash Y_1, Y_2}\,\text{L}\parr \qquad\qquad \frac{X \vdash a, b, Y}{X \vdash a \parr b, Y}\,\text{R}\parr$$

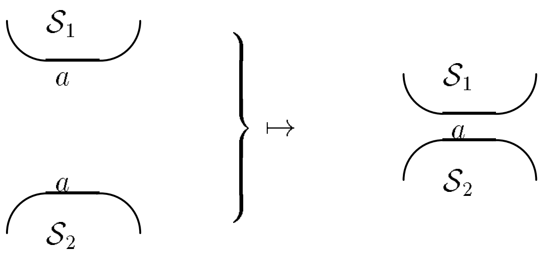
$$\frac{X_1 \vdash a, Y_1 \qquad X_2, b \vdash Y_2}{X_1, X_2, a \multimap b \vdash Y_1, Y_2}\,\text{L}\multimap \qquad\qquad \frac{X, a \vdash b, Y}{X \vdash a \multimap b, Y}\,\text{R}\multimap$$

⧫

Proof: We only have to sum up the respective witnesses.

The AX-rule has as witness the proof net $\left(\quad a \quad \right)$.

The Cut-rule is a consequence of the operation



The result is indeed a proof net, which is proved in the same way as prop. 3.7.1; now the two proof nets $\mathcal{S}_1$ and $\mathcal{S}_2$ are connected via the formula a (for the matter not necessarily a *cut formula*) instead of via the thick link.
The negation rules follow by prop. 3.7.3 and hence have witness $\mathrm{X}\bot(\cdot)$.
The binary multiplicative logical rules follow by prop. 3.7.1 since $\mathrm{R}\otimes$, $\mathrm{L}⅋$ and $\mathrm{L}\multimap$ are thick. So the operation $\mathrm{X}\Box(\cdot,\cdot)$ is the desired witness.
The unary multiplicative logical rules follow by prop. 3.7.2 since $\mathrm{L}\otimes$, $\mathrm{R}⅋$ and $\mathrm{R}\multimap$ are thin. The operation $\mathrm{X}\Box(\cdot)$ is the witness. ■

Observe that the thick binary links correspond to the binary multiplicative logical rules, which are irreversible. The thin links correspond to the unary multiplicative logical rules, which are reversible.

Let us recapitulate: we defined proof structures and a subset of them, viz. proof nets, on which the operations as described by the rules in lemma 3.10 are well-defined: i.e. starting with one or two proof nets, we get a proof net again.

Now let a class $\mathfrak{R}$ of rules be given. Given two multisets X and Y of syntactic formulas, we call $X \vdash Y$ $\mathfrak{R}$-derivable iff there is a $\mathfrak{R}$-derivation ending with $X \vdash Y$. By induction on the derivation it is clear that

$$\text{if } X \vdash Y \text{ is } \mathfrak{R}\text{-derivable, then } X \vdash Y \text{ is provable.}$$

Indeed, $\overline{X \vdash Y}$ being an axiom rule (of $\mathfrak{R}$) means by definition that $X \vdash Y$ is provable so that there is a proof net of $X \vdash Y$, and since all other rules correspond to sound operations, the conclusions remain provable. Moreover, as our derivation rules are constructive, each derivation $\mathcal{D}$ has also a witness, which we call the proof net $\mathcal{P}(\mathcal{D})$ of the derivation.

Of course now the question becomes: What about the converse? Does $\mathfrak{R}$ have so many derivable sequents, that the notions 'provable' and 'derivable' coincide, i.e. is $\mathfrak{R}$ *complete*?

Our aim in the following sections is to show that the scheme of rules of lemma 3.10 is a complete class, by showing that every proof net is the proof net of some derivation (in these rules). This is enough: if $X \vdash Y$ is provable, then there is a proof net of

$X \vdash Y$, which is the proof net of some derivation ending with $X \vdash Y$, which means that $X \vdash Y$ is derivable.

First we will investigate this operation, which assigns a proof net to a derivation.

4 The proof net of a derivation

In this section we will only consider **MLL**-derivations, where we mean the scheme of rules of lemma 3.10.

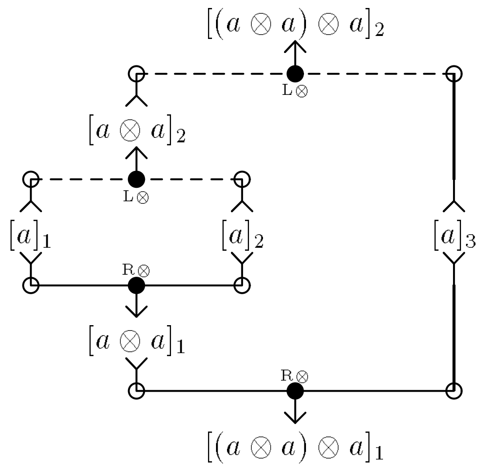
As the sequents in our derivations consist of multisets of syntactic formulas rather than sequences, it will (because of our abuse of notation writing down only the underlying syntactic formula) not always be clear on which occurrences a rule in the derivation acts.

Example 4.1 The following is a derivation with ambiguities, since it is not clear on which occurrences of a the first L$\otimes$-rule acts.

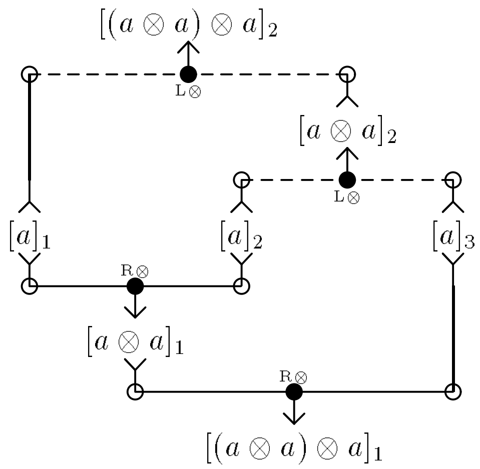
$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{a \vdash a}\,\text{Ax} \qquad \dfrac{}{a \vdash a}\,\text{Ax}}{a, a \vdash a \otimes a}\,\text{R}\otimes \qquad \dfrac{}{a \vdash a}\,\text{Ax}}{a, a, a \vdash (a \otimes a) \otimes a}\,\text{R}\otimes}{a \otimes a, a \vdash (a \otimes a) \otimes a}\,\text{L}\otimes}{(a \otimes a) \otimes a \vdash (a \otimes a) \otimes a}\,\text{L}\otimes$$

There are three possibilities:

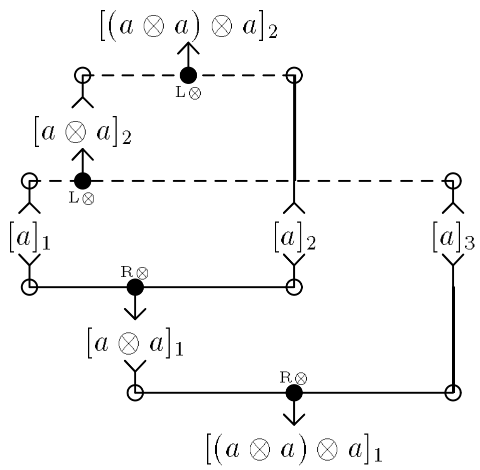
$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{[a]_1 \vdash [a]_1}\,\text{Ax} \qquad \dfrac{}{[a]_2 \vdash [a]_2}\,\text{Ax}}{[a]_1, [a]_2 \vdash [a \otimes a]_1}\,\text{R}\otimes \qquad \dfrac{}{[a]_3 \vdash [a]_3}\,\text{Ax}}{\mathbf{[a]_1}, \mathbf{[a]_2}, [a]_3 \vdash [(a \otimes a) \otimes a]_1}\,\text{R}\otimes}{[a \otimes a]_2, [a]_3 \vdash [(a \otimes a) \otimes a]_1}\,\text{L}\otimes}{[(a \otimes a) \otimes a]_2 \vdash [(a \otimes a) \otimes a]_1}\,\text{L}\otimes$$



$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{[a]_1 \vdash [a]_1}\,{\scriptstyle \mathrm{Ax}} \qquad \dfrac{}{[a]_2 \vdash [a]_2}\,{\scriptstyle \mathrm{Ax}}}{[a]_1, [a]_2 \vdash [a \otimes a]_1}\,{\scriptstyle \mathrm{R}\otimes} \qquad \dfrac{}{[a]_3 \vdash [a]_3}\,{\scriptstyle \mathrm{Ax}}}{[a]_1, \boldsymbol{[a]_2}, \boldsymbol{[a]_3} \vdash [(a \otimes a) \otimes a]_1}\,{\scriptstyle \mathrm{R}\otimes}}{[a \otimes a]_2, [a]_1 \vdash [(a \otimes a) \otimes a]_1}\,{\scriptstyle \mathrm{L}\otimes}}{[(a \otimes a) \otimes a]_2 \vdash [(a \otimes a) \otimes a]_1}\,{\scriptstyle \mathrm{L}\otimes}$$



$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{[a]_1 \vdash [a]_1}\,{\scriptstyle \mathrm{Ax}} \qquad \dfrac{}{[a]_2 \vdash [a]_2}\,{\scriptstyle \mathrm{Ax}}}{[a]_1, [a]_2 \vdash [a \otimes a]_1}\,{\scriptstyle \mathrm{R}\otimes} \qquad \dfrac{}{[a]_3 \vdash [a]_3}\,{\scriptstyle \mathrm{Ax}}}{\boldsymbol{[a]_1}, [a]_2, \boldsymbol{[a]_3} \vdash [(a \otimes a) \otimes a]_1}\,{\scriptstyle \mathrm{R}\otimes}}{[a \otimes a]_2, [a]_2 \vdash [(a \otimes a) \otimes a]_1}\,{\scriptstyle \mathrm{L}\otimes}}{[(a \otimes a) \otimes a]_2 \vdash [(a \otimes a) \otimes a]_1}\,{\scriptstyle \mathrm{L}\otimes}$$

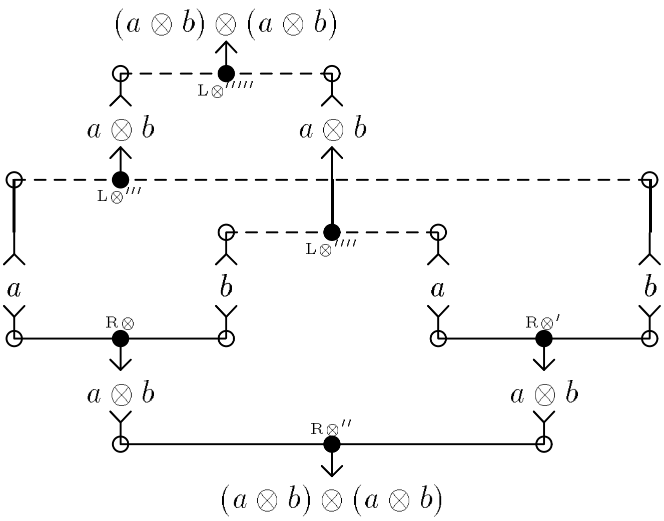


♦

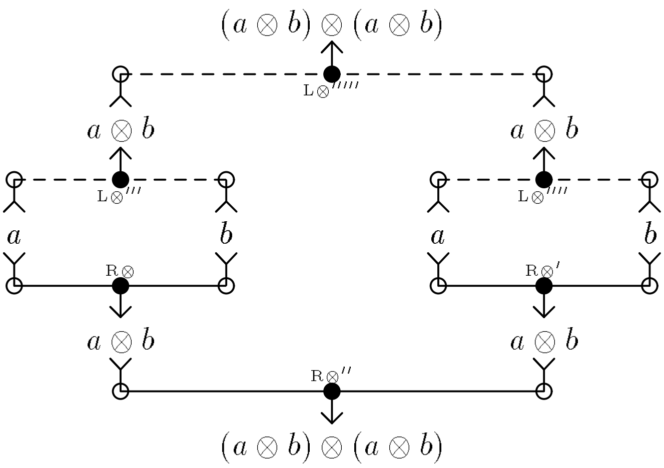
Example 4.2 The same holds for the following non-well-defined derivation.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{a \vdash a}\,{\scriptstyle \mathrm{Ax}_a} \quad \dfrac{}{b \vdash b}\,{\scriptstyle \mathrm{Ax}_b}}{a, b \vdash a \otimes b}\,{\scriptstyle \mathrm{R}\otimes_{a,b}} \qquad \dfrac{\dfrac{}{a \vdash a}\,{\scriptstyle \mathrm{Ax}_{a'}} \quad \dfrac{}{b \vdash b}\,{\scriptstyle \mathrm{Ax}_{b'}}}{a, b \vdash a \otimes b}\,{\scriptstyle \mathrm{R}\otimes'_{a',b'}}}{a, a, b, b \vdash (a \otimes b) \otimes (a \otimes b)}\,{\scriptstyle \mathrm{R}\otimes''_{\otimes,\otimes'}}}{a \otimes b, a, b \vdash (a \otimes b) \otimes (a \otimes b)}\,{\scriptstyle \mathrm{L}\otimes'''_{a,b} \text{ or } \mathrm{L}\otimes'''_{a,b'}}}{a \otimes b, a \otimes b \vdash (a \otimes b) \otimes (a \otimes b)}\,{\scriptstyle \mathrm{L}\otimes''''_{a',b'} \text{ or } \mathrm{L}\otimes''''_{a',b}}}{(a \otimes b) \otimes (a \otimes b) \vdash (a \otimes b) \otimes (a \otimes b)}\,{\scriptstyle \mathrm{L}\otimes'''''_{\otimes''',\otimes''''}}$$

Depending on which derivation is meant, its proof net is



or the identity proof net $\mathcal{I}((a \otimes b) \otimes (a \otimes b))$ (see page 8):



Observe that this last proof net is also the proof net of many other derivations, like (we abbreviate a bit)

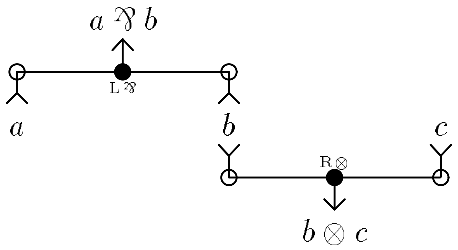
$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\mathrm{Ax}_a \qquad \mathrm{Ax}_b}{\mathrm{R}\otimes_{a,b}}}{\mathrm{L}\otimes'''_{a,b}} \qquad \dfrac{\mathrm{Ax}_{a'} \qquad \mathrm{Ax}_{b'}}{\mathrm{R}\otimes'_{a',b'}}}{\mathrm{R}\otimes''_{\otimes,\otimes'}}}{\mathrm{L}\otimes''''_{a',b'}}}{\mathrm{L}\otimes'''''_{\otimes''',\otimes''''}} \qquad \dfrac{\dfrac{\dfrac{\dfrac{\mathrm{Ax}_a \qquad \mathrm{Ax}_b}{\mathrm{R}\otimes_{a,b}}}{\mathrm{L}\otimes'''_{a,b}} \qquad \dfrac{\dfrac{\mathrm{Ax}_{a'} \qquad \mathrm{Ax}_{b'}}{\mathrm{R}\otimes'_{a',b'}}}{\mathrm{L}\otimes''''_{a',b'}}}{\mathrm{R}\otimes''_{\otimes,\otimes'}}}{\mathrm{L}\otimes'''''_{\otimes''',\otimes''''}}$$

which are derivations that differ from the first one only in the order of the rules. ♦

Example 4.3 Another example of this phenomenon is given by the two derivations:

$$\dfrac{\dfrac{\dfrac{}{a \vdash a}\,\text{Ax} \qquad \dfrac{}{b \vdash b}\,\text{Ax}}{a \parr b \vdash a, b}\,\text{L}\parr \qquad \dfrac{}{c \vdash c}\,\text{Ax}}{a \parr b, c \vdash a, b \otimes c}\,\text{R}\otimes \qquad\qquad \dfrac{\dfrac{}{a \vdash a}\,\text{Ax} \qquad \dfrac{\dfrac{}{b \vdash b}\,\text{Ax} \qquad \dfrac{}{c \vdash c}\,\text{Ax}}{b, c \vdash b \otimes c}\,\text{R}\otimes}{a \parr b, c \vdash a, b \otimes c}\,\text{L}\parr$$

with coinciding proof net:



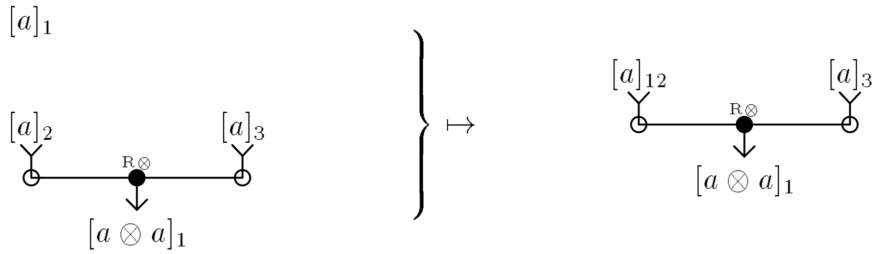
♦

A complete answer to the question which derivations have the same proof net may be found in theorem 6.4.

So far we have not paid any attention to the Cut-rule. As we saw in the proof of lemma 3.10, we have to identify two different occurrences of the same syntactic formula when pasting the two proof nets together. Again, this fact has its consequences for the meaning of derivations.

Example 4.4

$$\dfrac{\dfrac{}{[a]_1 \vdash \mathbf{[a]_1}}\,\text{Ax} \qquad \dfrac{\dfrac{}{[a]_2 \vdash [a]_2}\,\text{Ax} \qquad \dfrac{}{[a]_3 \vdash [a]_3}\,\text{Ax}}{\mathbf{[a]_2}, [a]_3 \vdash [a \otimes a]_1}\,\text{R}\otimes}{[a]_{12}, [a]_3 \vdash [a \otimes a]_1}\,\text{Cut}$$

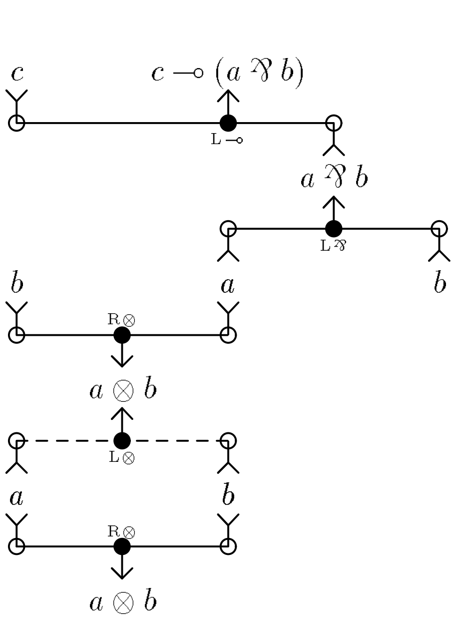


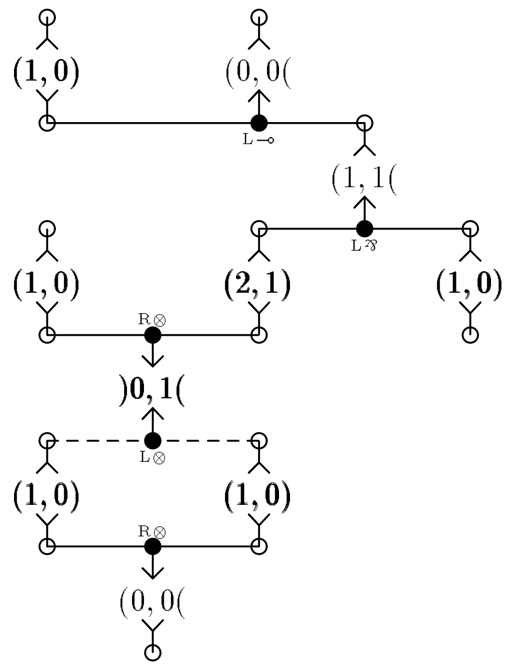
♦

Example 4.5 Another example with an application of the Cut-rule is the following. Next to the proof net we have indicated for every formula the number of Ax- respectively Cut-rules it corresponds with, viz. for axiomatic formulas as (n, m); for cut formulas as $)n, m($; and for flow formulas as $(n, m($.

$$
\dfrac{\dfrac{\dfrac{}{c \vdash c}\,\text{Ax} \qquad \dfrac{}{a \parr b \vdash a \parr b}\,\text{Ax}}{c, c \multimap (a \parr b) \vdash \boldsymbol{a \parr b}}\,\text{L}\multimap \qquad \dfrac{\dfrac{\dfrac{}{a \vdash a}\,\text{Ax} \quad \dfrac{}{b \vdash b}\,\text{Ax}}{a \parr b \vdash \boldsymbol{a}, b}\,\text{L}\parr \qquad \dfrac{\dfrac{}{a \vdash a}\,\text{Ax} \quad \dfrac{}{b \vdash b}\,\text{Ax}}{\boldsymbol{a}, b \vdash a \otimes b}\,\text{R}\otimes}{\boldsymbol{a \parr b}, b \vdash a \otimes b, b}\,\text{Cut}}{b, c, c \multimap (a \parr b) \vdash \boldsymbol{a \otimes b}, b}\,\text{Cut} \qquad \dfrac{\dfrac{\dfrac{}{a \vdash a}\,\text{Ax} \quad \dfrac{}{b \vdash b}\,\text{Ax}}{a, b \vdash a \otimes b}\,\text{R}\otimes}{\boldsymbol{a \otimes b} \vdash a \otimes b}\,\text{L}\otimes
$$

$$
\dfrac{b, c, c \multimap (a \parr b) \vdash \boldsymbol{a \otimes b}, b \qquad \boldsymbol{a \otimes b} \vdash a \otimes b}{b, c, c \multimap (a \parr b) \vdash a \otimes b, b}\,\text{Cut}
$$





Observe that this proof net has six axiomatic formulas (a, a, b, b, b and c) and one cut formula[4] ($a \otimes b$), while our derivation contains eight Ax-rules (on the formulas a, a, a, b, b, b, c and $a \parr b$) and three Cut-rules (on the formulas a, $a \parr b$ and $a \otimes b$). This is a consequence of the following proposition. ♦

Proposition 4.6 Every formula a of the proof net $\mathcal{P}$ of a derivation $\mathcal{D}$ corresponds to n Ax-rules on a and m Cut-rules on a, and the following holds:

$$
n - m = \begin{cases} 1 & \text{if } a \text{ is an axiomatic formula,} \\ 0 & \text{if } a \text{ is a flow formula,} \\ -1 & \text{if } a \text{ is a cut formula.} \end{cases}
$$

Moreover, every link L of $\mathcal{P}$ of the form X□ corresponds to a X□-rule. ♦

Proof: The proof is by induction on the definition of the derivation.
The proof net of $\dfrac{}{a \vdash a}\,\text{Ax}$ is the sole formula which is an axiomatic formula. It clearly

[4] See fig. 2 for the definition of axiomatic formula and of cut formula.

corresponds with 1 AX-rule and no CUT-rules, while there are no other links as well as no other rules, which proves this case.
If $\mathcal{D} = \mathrm{R}\otimes(\mathcal{D}_1, \mathcal{D}_2)$, by induction hypothesis the links of $\mathcal{P}_i$ correspond to the X□-rules of $\mathcal{D}_i$, hence the links of $\mathcal{P}$ correspond to the X□-rules of $\mathcal{D}$. Moreover, all old formulas of $\mathcal{P}$ correspond to the same AX- and CUT-rules as before while $a \otimes b$ correspond to no AX- or CUT-rules. This proves the proposition, since $a \otimes b$ is a flow formula, while all other formulas remain of the same type. (Viz. a (and b) were of type

a	ax. form.	a	ax. form.	R□ a	flow form.

and become of type

a	ax. form.	a	ax. form.	R□ a	flow form.

so the type is unchanged. For all other formulas this is trivial.)
The cases of a negation rule or another multiplicative logical rule are proved similar to the previous case.
In case of an application of the CUT-rule (say on the formula a), the second part of the proposition is clear, and also the first part w.r.t. all formulas different from a. When in $\mathcal{P}_i$ a corresponds to n_i AX-rules and m_i CUT-rules of $\mathcal{D}_i$, then in $\mathcal{P}$ a corresponds to $n := n_1 + n_2$ AX-rules and $m := m_1 + m_2 + 1$ CUT-rules of $\mathcal{D}$.

$\mathcal{P}_1$ $n_1 - m_1$	$\mathcal{P}_2$ $n_2 - m_2$	$\mathcal{P}$ $n - m$
a 1	a 1	a 1
a 1	a 0 L□	a 0 L□
R□ a 0	a 1	R□ a 0
R□ a 0	a 0 L□	R□ a -1

Depending on whether a is an axiomatic formula or a flow formula in $\mathcal{D}_1$ respectively $\mathcal{D}_2$, we know by the induction hypothesis the value of $n_i - m_i$, from which we can compute $n - m = (n_1 - m_1) + (n_2 - m_2) - 1$, and this number turns out to correspond in the desired way with the type of the new formula a in $\mathcal{P}$. ■

Definition 4.7 A derivation $\mathcal{D}$ is called a *sober* derivation if every formula a of its proof net $\mathcal{P}$ corresponds to 0 Ax-rules on a or 0 Cut-rules on a. ♦

In the light of prop. 4.6 we hence know that an axiomatic formula of the proof net of a sober derivation corresponds to exactly one Ax-rule; a cut formula corresponds to exactly one Cut-rule, while a flow formula corresponds to no Ax- or Cut-rules.

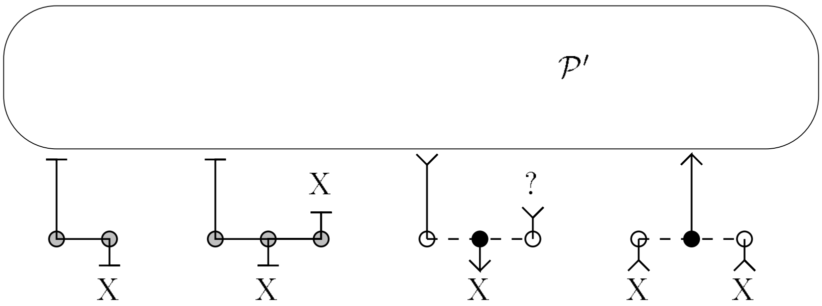
5 Subnets of proof nets

In this section $\mathcal{P}$ stands for a proof **net**. We recall that a substructure of $\mathcal{P}$ is a subnet iff it is non-empty and all its correction graphs are connected. We call a subnet *trivial* if it consists of a sole formula, and clearly $\mathcal{P} = \langle P, \mathcal{L} \rangle$ has exactly $|P|$ trivial subnets. The next lemma shows that a very rough upperbound for the number of non-trivial subnets is given by $2^{|\mathcal{L}|} - 1$, since every non-trivial subnet is determined by its links.

Lemma 5.1 Suppose $\mathcal{P}' = \langle P', \mathcal{L}' \rangle$ is a subnet of $\mathcal{P} = \langle P, \mathcal{L} \rangle$, then P' is a singleton in case $\mathcal{L}' = \emptyset$ and $\mathcal{P}' = \langle\!\langle \mathcal{L}' \rangle\!\rangle$ otherwise. ⧫

Proof: Suppose $\mathcal{L}' = \emptyset$. A subnet is non-empty, hence it contains at least one formula. If it would contain another formula as well, there would not be a path between them in the unique correction graph.
Suppose now that $\mathcal{L}'$ is non-empty, which means that $\langle\!\langle \mathcal{L}' \rangle\!\rangle$ is non-empty. It is clear that $\langle\!\langle \mathcal{L}' \rangle\!\rangle \subseteq \mathcal{P}' = \langle P', \mathcal{L}' \rangle$. If there would be some other formula in P', this is an $\mathcal{P}'$-unlinked formula, again contradicting connectedness of any correction graph of $\mathcal{P}'$. ■

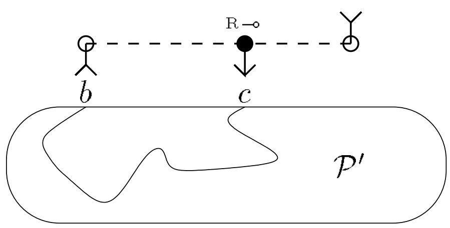
We call two formulas of a link L *opposite* when for some switching they are connected by the (replaced) link. That is, any two formulas of a link are opposite except for the two active formulas of a thin link.



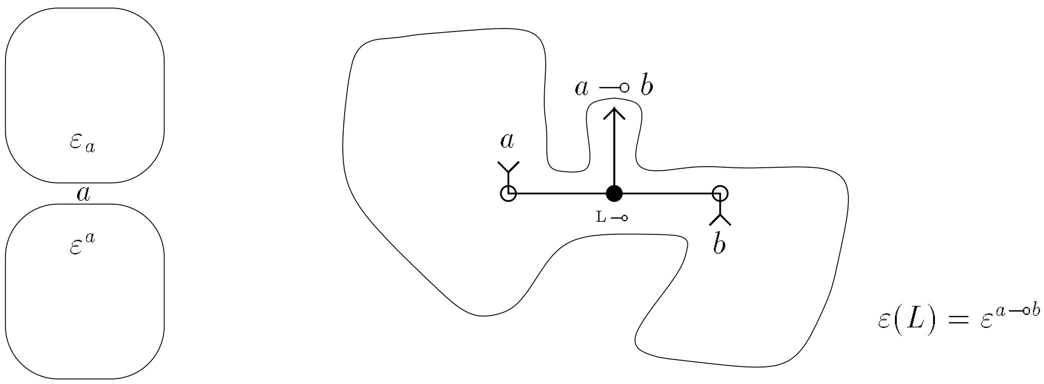
X can not belong to $\mathcal{P}'$;
? may belong to $\mathcal{P}'$.

Lemma 5.2 Let $\mathcal{P}'$ be a subnet of $\mathcal{P}$, and let L be a link not belonging to $\mathcal{P}'$. Then $\mathcal{P}'$ does not contain a pair of opposite formulas of L. ⧫

Proof: Suppose it contains the pair of opposite formulas b and c, then for a switching σ' of $\mathcal{P}'$ there is a path from b to c in $\sigma'\mathcal{P}'$ (since $\mathcal{P}'$ is a subnet), and extending σ' to an appropriate switching σ for $\mathcal{P}$ (such that b and c are connected by the (replaced) version of L), we obtain a cycle in $\sigma\mathcal{P}$; contradiction. ■



Definition 5.3 Given a formula a of $\mathcal{P}$, the *upper (lower) empire* ε_a (ε^a) of a is the largest subnet of $\mathcal{P}$ containing a as a conclusion (hypothesis).
Given a link L of $\mathcal{P}$, its unique main formula c is a conclusion of L or a premise of L, and we define the empire $\varepsilon(L)$ of L to be ε_c respectively ε^c. ⧫



Since $\left(\quad a \quad\right)$ is a subnet of $\mathcal{P}$ containing a as a conclusion, the existence of ε_a is clear by lemma 3.6.2; just put

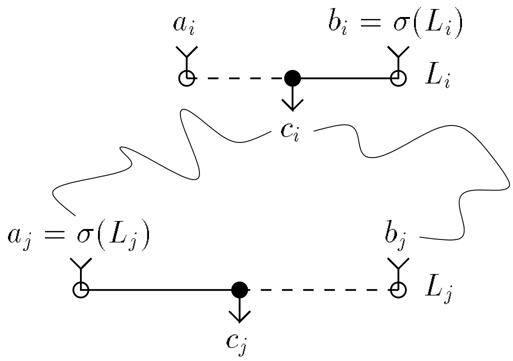
$$\varepsilon_a := \bigcup \{\mathcal{P}' \mid \mathcal{P}' \text{ is a subnet of } \mathcal{P} \text{ with } a \text{ as a conclusion}\}$$

which is well-defined because it is a finite non-empty union of non-disjoint subnets of $\mathcal{P}$. The same holds for the existence of ε^a.

At the moment our aim[5] is to show that for any link L of $\mathcal{P}$ the empire $\varepsilon(L)$ contains L. This is immediate for unary and thick links L, since then $\widehat{L} := \langle\!\langle \{L\} \rangle\!\rangle$ is a subnet already.

Let σ be a switching for $\mathcal{P}$. Then the correction graph $\sigma\mathcal{P}$ is a tree. Hence for each pair of vertices $a, b \in \mathcal{P}$ there is a (unique) path $\pi_\sigma(a, b)$ from a to b in $\sigma\mathcal{P}$. We define the following relation[6] on the n thin links L_i. Suppose that L_i has active formulas a_i and b_i and main formula $c_i = a_i \,\Box\, b_i$.

$$L_i \prec_\sigma L_j \quad \text{iff} \quad c_i \in \pi_\sigma(a_j, b_j) :$$



Lemma 5.4 The relation $\prec_\sigma$ is well-founded. ⧫

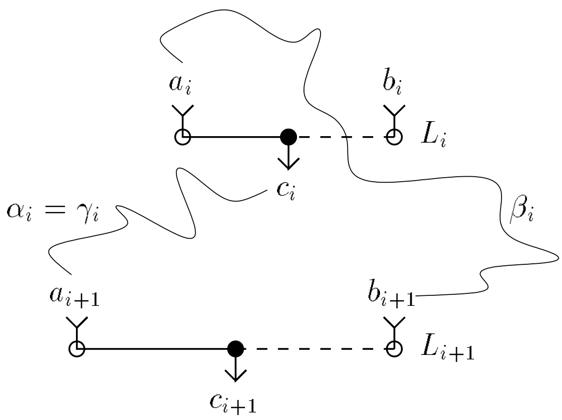
[5] In [BW95] the difficulty is to prove the existence of the empire; in our approach the problem is turned into showing non-triviality of the empire of a link.

[6] I thank Tonny Hurkens for pointing out to me this way of defining the relation.

Proof: W.l.o.g. we assume that $\sigma(L_i) = a_i$ for all i. Suppose that there is a cycle

$$L_0 \prec_\sigma L_1 \prec_\sigma \ldots \prec_\sigma L_p = L_0$$

with $p > 0$ minimal. Then for $i = 0, \ldots, p-1$ we have $c_i \in \pi_\sigma(a_{i+1}, b_{i+1})$, while no other $c_j \in \pi_\sigma(a_{i+1}, b_{i+1})$ $(j = 0, \ldots, \hat{i}, \ldots, p-1)$ by minimality of p. Let $\alpha_i := \pi_\sigma(a_{i+1}, c_i)$ and $\beta_i := \pi_\sigma(b_{i+1}, c_i)$, then at most one of those two paths contains the edge $a_i c_i$; define γ_i to be the one **not** containing $a_i c_i$. It is clear that γ_i does not contain the other $a_j c_j$ either, as $c_j \notin \pi_\sigma(a_{i+1}, b_{i+1})$ $(j = 0, \ldots, \hat{i}, \ldots, p-1)$. Hence γ_i is not affected when we change the value of the switching on the links $L_0, \ldots, L_{p-1}$. So let us define



$$\sigma'(L) = \begin{cases} \sigma(L) & \text{if } L \notin \{L_0, \ldots, L_{p-1}\}, \\ a_{i+1} & \text{if } L = L_{i+1} \text{ and } \gamma_i = \alpha_i, \\ b_{i+1} & \text{if } L = L_{i+1} \text{ and } \gamma_i = \beta_i, \end{cases}$$

then we obtain a cycle

$$\sigma'(L_0)c_0; \gamma_0; \sigma'(L_1)c_1; \gamma_1; \sigma'(L_2)c_2; \ldots; \sigma'(L_{p-1})c_{p-1}; \gamma_{p-1}; [\sigma'(L_0)c_0; \ldots]$$

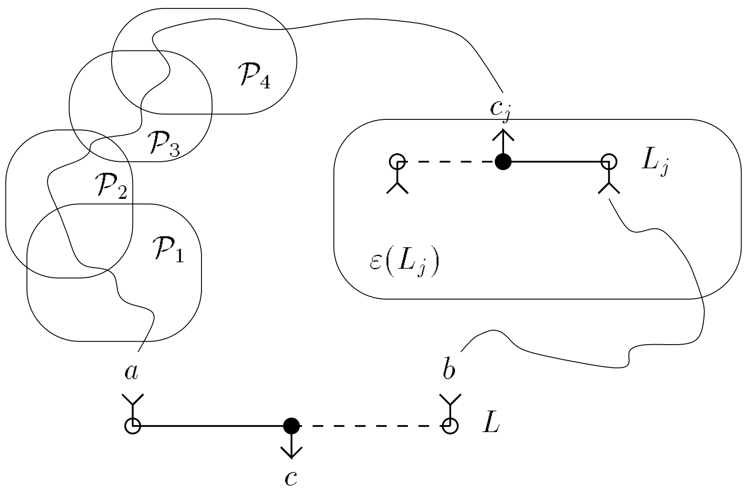
which contradicts the fact that $\sigma'\mathcal{P}$ is a tree. ■

This lemma shows that we may apply induction w.r.t. $\prec_\sigma$ for statements Φ about the thin links of a proof net:

$$\forall L \quad \Big((\forall K \prec_\sigma L \quad \Phi(K)) \Rightarrow \Phi(L) \Big) \quad \Longrightarrow \quad \forall L \quad \Phi(L)$$

Lemma 5.5 For any thin link L the empire $\varepsilon(L)$ contains L. ♦

Proof: Let σ be an arbitrary switching. We apply induction on $\prec_\sigma$.
Let a thin link L be given and suppose the lemma holds for all thin links $K_i \prec_\sigma L$ (induction hypothesis). We have to show that it holds for L as well.
The active formulas a and b of L are connected in $\sigma\mathcal{P}$ by a unique path $\pi_\sigma(a, b)$. Any edge $x_j y_j$ of this path belongs to a link L_j, and we define a subnet $\mathcal{P}_j$ containing this link as follows: in case L_j is unary or thick, we put $\mathcal{P}_j := \widehat{L}$ $(= \langle\!\langle \{L\} \rangle\!\rangle)$; in case L_j is a thin link, clearly $L_j \prec_\sigma L$, whence it is one of the K_i, and we can put $\mathcal{P}_j := \varepsilon(L_j)$, containing L_j by induction hypothesis. The union $\mathcal{P}' := \Big(\quad a \quad \Big) \cup \bigcup \mathcal{P}_j$ is a subnet again, being a finite non-empty union of sequentially intersecting subnets. By construction $\sigma\mathcal{P}'$ contains $\pi_\sigma(a, b)$, but it is not a priori clear that a and b are leaves of $\mathcal{P}'$.



Suppose $\mathcal{P}'$ contains L, then $L \in \mathcal{P}_j$ for some j. So for some thin link $L_j \prec_\sigma L$ we have $L \in \varepsilon(L_j)$. Change the switching in L into σ' such that the path from c to c_j does not contain an edge of L_j. Now in $\sigma'\varepsilon(L_j)$ there is also a path from c to c_j, but this path approaches c_j via the link L_j (by definition of empire). Hence we have in $\sigma'\mathcal{P}$ two different paths from c to c_j, constituting a cycle; contradiction.
So $L \notin \mathcal{P}'$, which by lemma 5.2 means that also $c \notin L$, whence we may attach L to $\mathcal{P}'$, obtaining a subnet again by prop. 3.7.2. But now we have found a non-trivial subnet, by definition contained in $\varepsilon(L)$, which shows that $\varepsilon(L)$ is non-trivial. ∎

With the help of lemma 5.2 and lemma 5.5 it is possible to determine how empires are embedded in $\mathcal{P}$. Let us fix a formula a.

If a is a hypothesis of $\mathcal{P}$ then ε_a is clearly trivial. Otherwise a is conclusion of a link L_a. If L_a is unary or thick, $\widehat{L_a}$ is a subnet with a as conclusion, whence ε_a contains L_a. If L_a is thin and a is the main formula of L_a, then by definition ε_a equals $\varepsilon(L_a)$, and we have just seen that in this case ε_a contains L_a as well. But in case L_a is thin and a is an active formula of L_a, it turns out that ε_a is trivial, since if it would contain L_a, there would be a switching in which a is disconnected from the rest.

In case a is a conclusion of $\mathcal{P}$, then ε_a equals $\mathcal{P}$.

Now let us consider the ε_a-leaves. We call a (being the conclusion of ε_a) the *main leaf*, while we call the other leaves *active leaves* of ε_a.[7] In particular, when ε_a is trivial, a is a main leaf as a conclusion of ε_a, and a is an active leaf as a hypothesis of ε_a.

Suppose a link $L \notin \varepsilon_a$ of $\mathcal{P}$ is connected to an active leaf b. We differentiate on the three following cases.

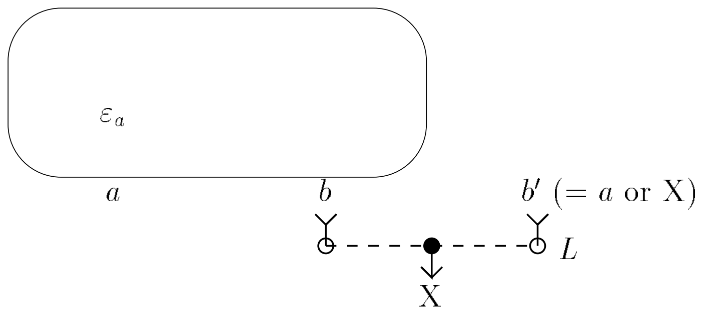
[7] Intuitively we may consider ε_a as a thick link, whith ε_a-hypotheses and ε_a-conclusions (see fig. 2) as its premises and conclusions respectively (see fig. 1); with the main leaf as its main formula and with the active leaves as its active formulas.

- L is a unary or thick link. In this case the other formulas of L do not belong to ε_a by lemma 5.2. Taking the union of the subnets ε_a and $\widehat{L}$ (intersecting in the formula b) we get a strictly larger subnet with a as conclusion, in contradiction with the definition of empire. So this cannot happen.

- L is a thin link and b is its main formula. Consider $\varepsilon(L)$, which is not trivial by lemma 5.5. Suppose it contains a, then (for an arbitrary switching σ for $\mathcal{P}$) there is a path in $\sigma\varepsilon(L)$ from b to a. However, in $\sigma\varepsilon_a$ there is a different path from b to a; contradiction. So $a \notin \varepsilon(L)$, and we may take the union $\varepsilon(L) \cup \varepsilon_a$, which is a strictly larger proof net, and still has a as conclusion. Again, this is in contradiction with the definition of empire.

- L is a thin link and b is one of its active formulas. The main formula of L does not belong to ε_a by lemma 5.2. The other active formula b' may belong to ε_a. However, if b' is an active leaf of ε_a, then we could enlarge ε_a by prop. 3.7.2, in contradiction with the definition of empire. So b' coincides with a, or $b' \notin \varepsilon_a$

We have established the following proposition[8].



Proposition 5.6 If some link $L \notin \varepsilon_a$ (ε^a) of $\mathcal{P}$ is connected to an active leaf b of ε_a (ε^a), then L is a thin link; b is an active formula of L; the other active formula b' coincides with a or is outside ε_a (ε^a); and the main formula of L is outside ε_a (ε^a). ♦

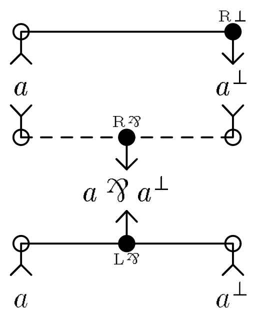
[8] This is the analogue of the principle switching theorem of [BW95]. It shows that we can isolate our empire by choosing a partial switching, viz. defined on the thin links mentioned in the proposition. Put $\sigma(L) = b'$, then perhaps deleting L^a (the link under a) yields ε_a as a connected component of $\sigma\mathcal{P}$.

6 Splitting formulas and sequentialization

Throughout this section $\mathcal{P}$ will again stand for a proof net.

We call a link of $\mathcal{P}$ *terminal* iff its main formula is a leaf of $\mathcal{P}$.

In prop. 3.7 we have proved that terminal unary and thin links can be removed, still keeping a proof net. For a terminal thick link the situation is different: how do we know that the removal of the thick link yields two **disjoint** proof structures...? Moreover, appearingly in contrast to the one-sided counterpart, a non-trivial proof net does not always contain a terminal link at all, e.g. in:



Another approach in the literature ([Da88]) to sequentialization for the one-sided theory is by considering a thin non-terminal link having a main formula that is splitting in the following sense.

Definition 6.1 A formula c is *splitting* if it is an internal formula such that:

- $\varepsilon_c \cap \varepsilon^c = \Big(\quad c \quad \Big)$
- $\varepsilon_c \cup \varepsilon^c = \mathcal{P}$

♦

Observe that this definition of splitting formula generalizes the existing notions of both 'splitting tensor' and 'splitting par'. A terminal $\otimes$-link is 'splitting' iff one of its active formulas is splitting iff both of its active formulas are splitting; a non-terminal $⅋$-link is 'splitting' iff its main formula is splitting.

Lemma 6.2 (Splitting Lemma) When $\mathcal{P}$ has at least two links, and moreover it has no terminal thin links, then it has a splitting formula. ♦

Proof: Let I be the set of internal formulas. I is non-empty because there are at least two links. Consider the set

$$\{\varepsilon_a \,|\, a \in I\} \cup \{\varepsilon^a \,|\, a \in I\}$$

and let $\mathcal{P}' = \varepsilon_c$ (or: ε^c) be a maximal element of this set ordered by inclusion. We claim that c is a splitting formula.
It is clear that $\varepsilon_c \cap \varepsilon^c \supseteq \Big(\quad c \quad \Big)$. Suppose also another $b \in \varepsilon_c \cap \varepsilon^c$, then for any switching σ there are two different paths (viz. in $\sigma\varepsilon_c$ and $\sigma\varepsilon^c$). This contradicts the fact that $\mathcal{P}$ is a proof net. Hence $\varepsilon_c \cap \varepsilon^c = \Big(\quad c \quad \Big)$.
Now consider $\varepsilon_c \cup \varepsilon^c$. As usual we denote by L_c and L^c the link c is a conclusion respectively premise of. Both links exist as c is an internal formula.
Suppose $\varepsilon_c \cup \varepsilon^c$ is not all of $\mathcal{P}$, then there is a link L that is attached to ε_c or ε^c. We claim that it is attached to an active leaf. For suppose it is attached to ε_c, then it is attached to an active leaf of ε_c or to its main leaf c; but in the last case $L = L^c$, so ε^c (not containing L) is trivial, and L is attached to the unique active leaf of ε^c.
Let us suppose w.l.o.g. that L is attached to an active leaf b of ε_c. Applying prop. 5.6 and lemma 5.2 (to $\varepsilon_c \cup \varepsilon^c$) we know L is a thin link; b is an active formula of L; the other active formula b' coincides with c or is outside ε_c; and the main formula d of L is outside $\varepsilon_c \cup \varepsilon^c$. As $\varepsilon(L)$ is the largest subnet with d as main leaf, and it contains L (lemma 5.5), it also contains the union $\varepsilon(L) \cup \varepsilon_c \cup \varepsilon^c$, so $\varepsilon_c \cup \varepsilon^c \subsetneq \varepsilon(L)$. Now by assumption $\mathcal{P}$ has no terminal thin links, so d is an internal formula and $\mathcal{P}' \subsetneq \varepsilon(L)$ contradicts the maximality of $\mathcal{P}'$. Hence $\varepsilon_c \cup \varepsilon^c = \mathcal{P}$. ■

This is the main ingredient of the following theorem:

Theorem 6.3 (Sequentialization) Every proof net $\mathcal{P}$ is the proof net of an **MLL**-derivation $\mathcal{D}$ (called a *sequentialization*). Moreover, this sequentialization is a sober (see def. 4.7) derivation, so that every cut formula of $\mathcal{P}$ corresponds to one application of the Cut-rule, every axiomatic formula corresponds to one application of the Ax-rule, while every link corresponds to an application of a logical rule. ♦

Proof: We apply induction on the size of $\mathcal{P}$ (equivalent: the number of links $|\mathcal{L}|$).

If $|\mathcal{L}| = 0$ then $\mathcal{P}$ consists of a sole (and hence axiomatic) formula (being a proof net, hence non-empty and connected), which is the image of the identity rule Ax.

If $|\mathcal{L}| \geq 1$ and $\mathcal{P}$ has a thin terminal link, then $\mathcal{P} = \mathrm{X}\Box(\mathcal{P}_1)$ where $\mathcal{P}_1$ is a proof net too. Now by induction hypothesis $\mathcal{P}_1$ is the image of a derivation, and applying the corresponding unary multiplicative logical rule gives a derivation that translates into $\mathcal{P}$.
Now suppose $\mathcal{P}$ does not have a thin terminal link.

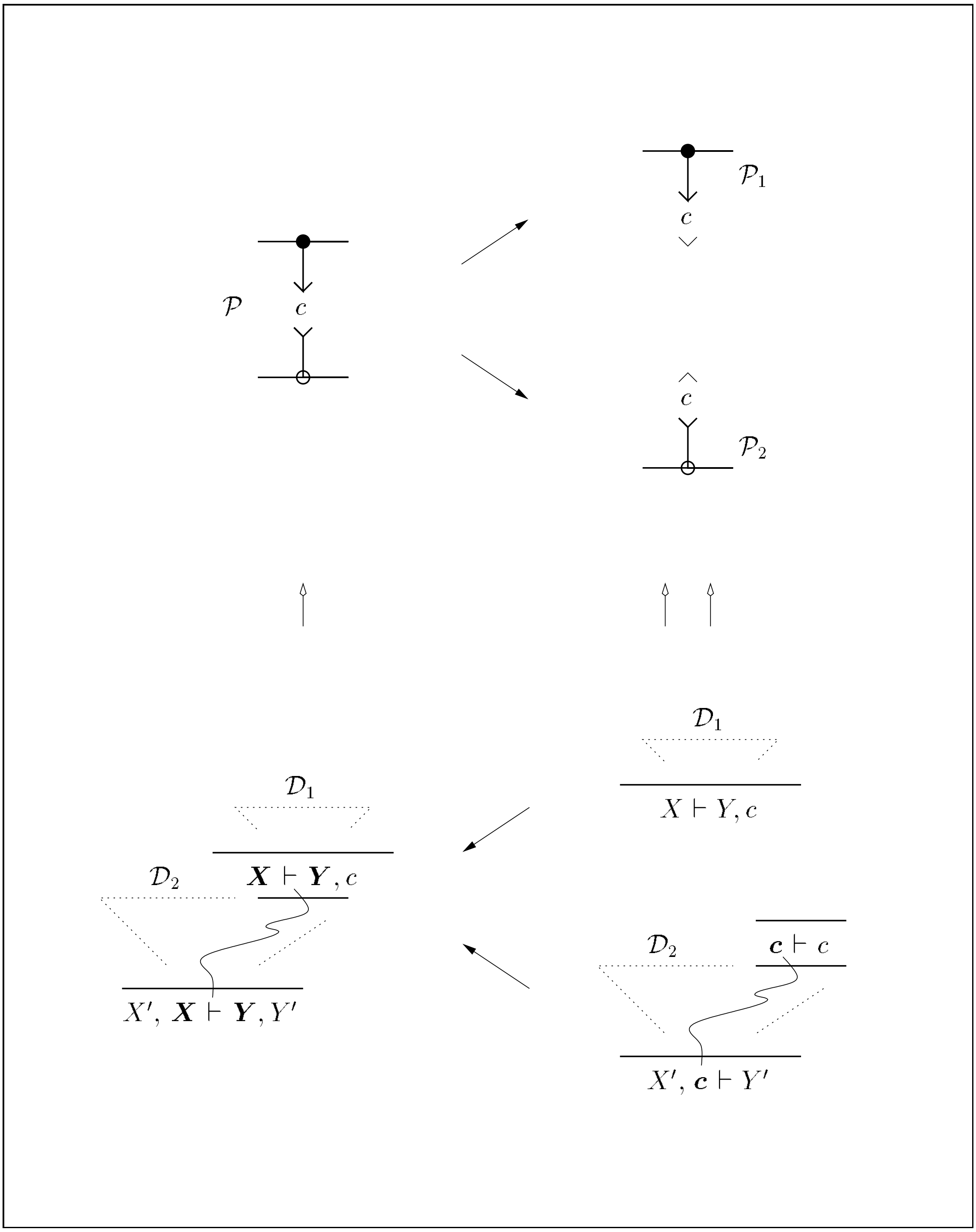


Figure 3: Substitution of $\mathcal{D}_1$ into $\mathcal{D}_2$.

In case $|\mathcal{L}| = 1$, then the unique link L is terminal and hence L is unary or thick. Now $\mathcal{P} = \widehat{L}$ is clearly the image of a derivation, viz. one or two AX-rules composed with a negation rule or with a binary multiplicative logical rule.

In case $|\mathcal{L}| \geq 2$, we are in the conditions of lemma 6.2, which gives us a splitting formula c and a partition of $\mathcal{P}$ into two non-trivial proof nets $\mathcal{P}_1 := \varepsilon_c$ and $\mathcal{P}_2 := \varepsilon^c$. By induction hypothesis those $\mathcal{P}_i$ are images of derivations $\mathcal{D}_i$. As c is an internal formula, by fig. 2 we can distinguish 4 cases:

- c is a cut formula.
  Application of a CUT does the job, and this is o.k. since c is a cut formula.
- c is a flow (down) formula.
  We could apply CUT again in this case, but then there are more CUT's in our derivation than cut formulas in our proof net. However, in $\mathcal{P}_2$ the leaf c has become an axiomatic formula now, so $\mathcal{D}_2$ uses an AX-rule $\dfrac{}{c \vdash c}$. Substituting the left (hypothesis) c by the other leaves X and Y of $\mathcal{D}_1$ through the whole derivation $\mathcal{D}_2$, together with a replacement of the AX-rule $\dfrac{}{c \vdash c}$ (that has become $\dfrac{}{X \vdash Y, c}$) by $\mathcal{D}_1$, we obtain a derivation $\mathcal{D}$ that translates into $\mathcal{P}$ (see fig. 3).
- c is a flow (up) formula.
  This case is also treated with substitution.
- c is an axiomatic formula.
  Here we may substitute $\mathcal{D}_1$ in $\mathcal{D}_2$, or the other way around.

■

The last clause shows that the process of sequentialization of a proof net may yield different derivations, which however by soberness all contain the same 'set of rules' (since every logical rule corresponds to a link; every AX-rule corresponds to an axiomatic formula; and every CUT-rule corresponds to a cut formula). We know that there are many more other derivations that also have this proof net (e.g. replace $\dfrac{}{a \vdash a}{\scriptstyle \mathrm{AX}}$ by $\dfrac{\dfrac{}{a \vdash a}{\scriptstyle \mathrm{AX}} \quad \dfrac{}{a \vdash a}{\scriptstyle \mathrm{AX}}}{a \vdash a}{\scriptstyle \mathrm{CUT}}$) and these do satisfy the conditions of proposition 4.6, but they need not satisfy the extra condition of soberness.

The next theorem will characterize all derivations that have the same proof net. We define a *strong* CUT-inference to be an application of the CUT-rule on a formula a which is main formula (for some *logical* link) in both the left and the right subderivation above the CUT-rule. We define a *weak* CUT-inference on a to be a CUT-rule that is not a strong CUT-inference. This means that in at least one of the two subderivations a originates from an AX-rule. By substitution (see fig. 4) of the one subderivation in the other we get rid of one AX-rule and one (weak) CUT-rule. Such a substitution will be called a *weak* CUT *replacement*.

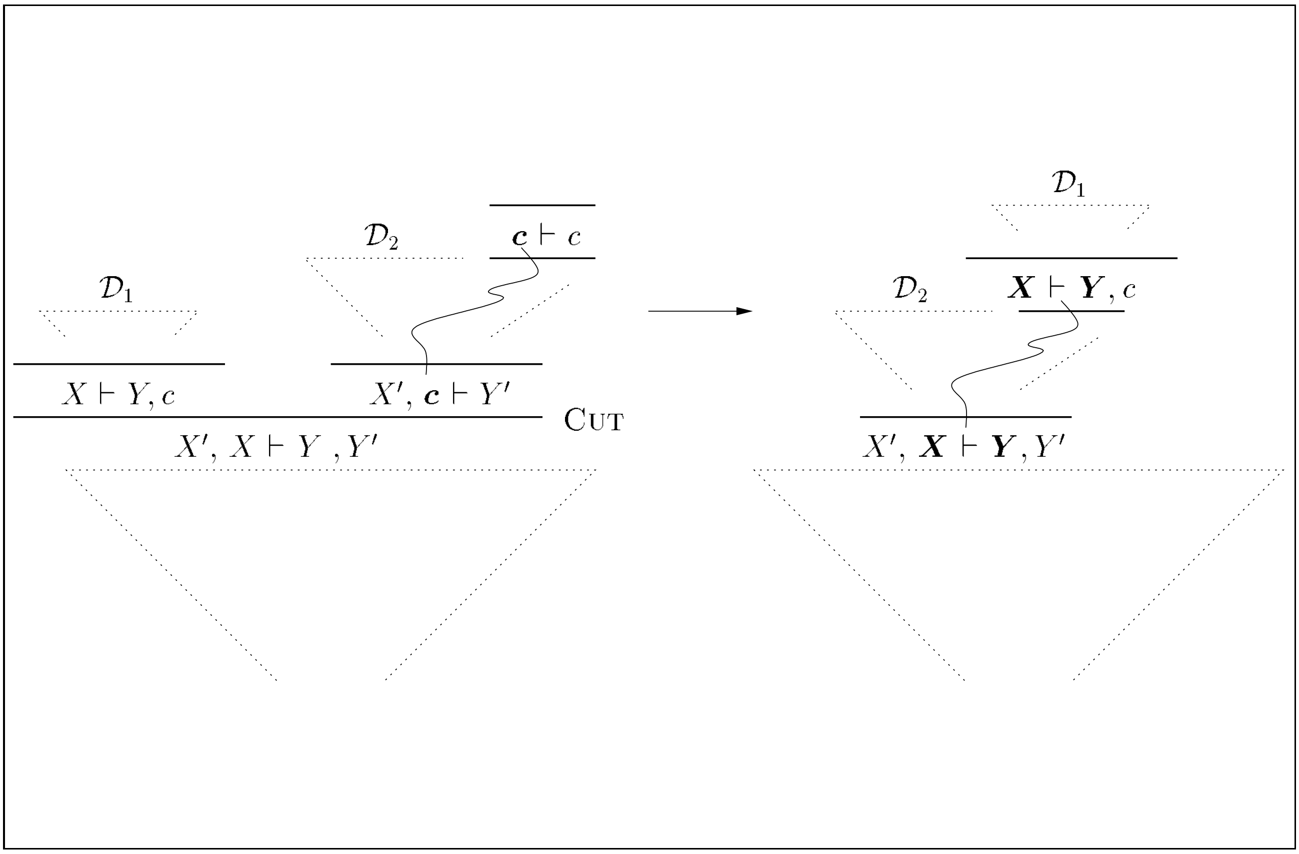


Figure 4: Weak CUT replacement (by a substitution of $\mathcal{D}_1$ into $\mathcal{D}_2$).

Theorem 6.4 Let $\mathcal{D}$ and $\mathcal{D}'$ be derivations of the same sequent $X \vdash Y$. Then their respective proof nets $\mathcal{P}$ and $\mathcal{P}'$ are equal if and only if there exists a sequence of derivations $\mathcal{D} = \mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_n = \mathcal{D}'$ such that $\mathcal{D}_i$ and $\mathcal{D}_{i+1}$ differ only for a permutation of two consecutive inferences, or $\mathcal{D}_i$ is obtained from $\mathcal{D}_{i+1}$ (or the other way around) by a weak Cut replacement. ♦

Proof: The if-part is clear: attaching links in a different order does not give a different proof net, and neither does weak Cut replacement.

The other way around, suppose $\mathcal{P}$ and $\mathcal{P}'$ are equal. From prop. 4.6 we derive that $\mathcal{D}$ and $\mathcal{D}'$ have the same logical rules, and the same surplus of Ax-rules w.r.t. Cut-rules (for each formula). In both derivations we can replace all weak Cut-inferences, so that we may assume that both $\mathcal{D}$ and $\mathcal{D}'$ are sober derivations (i.e. with only strong Cut-rules). This implies that $\mathcal{D}$ and $\mathcal{D}'$ have the same logical rules (one for each link of $\mathcal{P} = \mathcal{P}'$), the same (strong) Cut-rules (one for each cut formula of $\mathcal{P}$) and the same Ax-rules (one for each axiomatic formula).
The rest of the proof is almost similar to that of [BW95]:

Let I be the last inference of $\mathcal{D}$. Via a link L of $\mathcal{P} = \mathcal{P}'$ we know this inference corresponds to a *similar* inference I' of $\mathcal{D}'$ (i.e. an inference of the same type (R$\otimes$, L$\otimes$, et cetera) and with the same active and/or main formulas). By induction on the level l of I' in the deduction tree $\mathcal{D}'$ we first show that after some permutations of two consecutive inferences $\mathcal{D}'$ may be turned into a derivation $\mathcal{D}''$ with last inference I'' similar to I and I'.
If $l = 0$ I' is already the last inference of $\mathcal{D}'$.
If $l > 0$, denote by I_1' the inference below I' in $\mathcal{D}'$. This inference has active formulas distinct from the main formula(s) of I', and moreover it corresponds to a similar inference I_1 of $\mathcal{D}$. We distinguish four cases.
In case I' is an Ax-rule, an active formula of I_1' would be a main formula of I'; contradiction.
In case I' is a negation rule, we can permute it with I_1' (yielding two similar inferences in the other order).
In case I' is a unary multiplicative logical rule, we can also permute it with I_1'.
In case I' is a binary multiplicative logical rule or a Cut-rule, we can permute it in the subcases that I_1' is a negation rule, a binary multiplicative logical rule or a Cut-rule. As I_1' cannot be an Ax-rule, we only have to consider the subcase that I_1' is a unary multiplicative logical rule.
E.g. suppose I' is a R$\otimes$-rule with active formulas a_1 and a_2, and I_1' is a R$\parr$-rule with active formulas b_1 and b_2. We have two prove that b_1 and b_2 originate from the same

subderivation $\mathcal{D}'_j$ of $\mathcal{D}'$ above I'.

$$
\dfrac{\dfrac{\begin{array}{c}\mathcal{D}_1\\ \ddots\ \vdots\ \iddots\\ \vdash b_1, b_2\end{array}}{\vdash b_1 \parr b_2}\,{\scriptstyle I_1} \atop \begin{array}{c}\ddots\ \vdots\ \iddots\\ \vdash a_1, b_1 \parr b_2\end{array} \qquad \begin{array}{c}\mathcal{D}_2\\ \vdash a_2\end{array}}{\vdash a_1 \otimes a_2, b_1 \parr b_2}\,{\scriptstyle I}
\qquad\qquad
\begin{array}{c}
\dfrac{\dfrac{\begin{array}{c}\mathcal{D}'_1\\ \ddots\ \vdots\ \iddots\\ \vdash a_1\end{array} \qquad \begin{array}{c}\mathcal{D}'_2\\ \ddots\ \vdots\ \iddots\\ \vdash a_2\end{array}}{\vdash a_1 \otimes a_2, b_1, b_2}\,{\scriptstyle I'}}{\vdash a_1 \otimes a_2, b_1 \parr b_2}\,{\scriptstyle I'_1}\\
\ddots\qquad \vdots\qquad \iddots
\end{array}
$$

As a_j is a leaf of the subnet $\mathcal{P}(\mathcal{D}'_j)$ of $\mathcal{P}$ (actually a conclusion), we know that $\mathcal{P}(\mathcal{D}'_j) \subseteq \varepsilon_{a_j} = \mathcal{P}(\mathcal{D}_j)$. If w.l.o.g. I_1 belongs to $\mathcal{D}_1$, then the corresponding link L_1 of $\mathcal{P}$ occurs in $\mathcal{P}(\mathcal{D}_1)$. But then b_1 and b_2 must belong to $\mathcal{D}'_1$. This means that we can permute I' and I'_1, as desired.
Now after this permutation, l has decreased by 1, so that we know by induction hypothesis that after some (more) permutations of two consecutive inferences $\mathcal{D}'$ is turned into a derivation $\mathcal{D}''$ with last inference I''.

Moreover, this last inference I'' of $\mathcal{D}''$ actually coincides with I. This is clear for the sequent below the bar which consist of the leaves of $\mathcal{P}$. Hence we are ready in the case of an AX-rule, while the result is immediately clear for a negation rule or a unary multiplicative logical rule. For a binary multiplicative logical rule or a CUT-rule the result follows by inspection of the empires of the active formulas.

Next we show by induction on the coinciding subtree of $\mathcal{D}$ and $\mathcal{D}'$ that after some permutations of two consecutive inferences $\mathcal{D}'$ may be turned into $\mathcal{D}$.
If the coinciding subtree is $\mathcal{D}$ we are ready.
Otherwise there is a branch of $\mathcal{D}$ with an inference I of minimal level such that $\mathcal{D}$ and $\mathcal{D}'$ coincide below I. Let us call the subderivation above this coinciding subbranch $\widehat{\mathcal{D}}$ and $\widehat{\mathcal{D}'}$ respectively. Then applying the previous result we know that after some permutation of consecutive inferences $\widehat{\mathcal{D}'}$ may be turned into a derivation with last inference exactly the same as I. Hence the coinciding subtree has increased and we may apply the induction hypothesis. ∎

7 Rotatable proof nets of MLL

In this section we will study the predicates $\dashv\vdash$ and $\dashv_{\mathrm{r}}\vdash$ on multisets of syntactic formulas of **MLL**:

$$
\begin{array}{lll}
X \dashv\vdash Y & \Longleftrightarrow & X \vdash Y \text{ is provable} \quad \text{and} \quad Y \vdash X \text{ is provable} \\
 & \Longleftrightarrow & \text{there is a proof net } \mathcal{P}_1 \text{ of } X \vdash Y \\
 & & \quad \text{and a proof net } \mathcal{P}_2 \text{ of } Y \vdash X \\
 & \Longleftrightarrow & \text{there is a cut-free proof net } \mathcal{P}_1 \text{ of } X \vdash Y \\
 & & \quad \text{and a cut-free proof net } \mathcal{P}_2 \text{ of } Y \vdash X \\
 & \Longleftrightarrow & \text{there is a cut-free and } \eta\text{-expanded proof net } \mathcal{P}_1 \text{ of } X \vdash Y \\
 & & \quad \text{and a cut-free and } \eta\text{-expanded proof net } \mathcal{P}_2 \text{ of } Y \vdash X \\
X \dashv_{\mathrm{r}}\vdash Y & \Longleftrightarrow & \text{there is a proof net } \mathcal{P} \text{ of } X \vdash Y \text{ such that} \\
 & & \quad \text{its rotation } \mathcal{P}^* \text{ is a proof net of } Y \vdash X \\
 & \overset{(*)}{\Longleftrightarrow} & \text{there is a cut-free proof net } \mathcal{P} \text{ of } X \vdash Y \text{ such that} \\
 & & \quad \text{its rotation } \mathcal{P}^* \text{ is a cut-free proof net of } Y \vdash X \\
 & \Longleftrightarrow & \text{there is a cut-free and } \eta\text{-expanded proof net } \mathcal{P} \text{ of } X \vdash Y \text{ such that} \\
 & & \quad \text{its rotation } \mathcal{P}^* \text{ is a cut-free and } \eta\text{-expanded proof net of } Y \vdash X
\end{array}
$$

Here we mean by an *η-expanded* proof structure a proof structure with only **atomic** axiomatic formulas. The last equivalence in both definitions above is a consequence of the fact that we can replace a non-atomic axiomatic formula a by the identity proof net $\mathcal{I}(a)$ having only atomic axiomatic formulas and being invariant under rotation. After this operation the proof structure remains a proof net (the rotation of which is also obtained by a replacement of a by $\mathcal{I}(a)$ and hence is also a proof net).

The equivalence marked by $(*)$ is a consequence of the fact that rotating a proof structure commutes with a cut elimination step: Suppose $\mathcal{P}$ is a rotatable proof net of $X \vdash Y$ (i.e. $\mathcal{P}$ and $\mathcal{P}^*$ are proof nets). Then its reduct $\mathcal{P}'$ is a proof net (by the soundness of cut-elimination) the rotation of which ($(\mathcal{P}')^*$) is nothing else but the reduct $(\mathcal{P}^*)'$ of $\mathcal{P}^*$, and hence a proof net itself. This implies that $\mathcal{P}'$ is a rotatable proof net. By induction we may conclude that $\widehat{\mathcal{P}}$ is a rotatable cut-free proof net of $X \vdash Y$, the rotation of which is automatically cut-free as well.

$$
\begin{array}{ccc}
\mathcal{P} & \rightsquigarrow & \mathcal{P}' \\
\downarrow (\cdot)^* & & \downarrow (\cdot)^* \\
\mathcal{P}^* & \rightsquigarrow & (\mathcal{P}^*)'
\end{array}
$$

Lemma 7.1 Let $\mathcal{P}$ be a proof net of $X \vdash Y$ with t thin binary links and T thick binary links. Define $f := |X| + |Y|$. Then the following holds:

$$f + t = 2 + T.$$

If moreover also $Y \vdash X$ is provable, then

$$|X| + |Y| = 2.$$

♦

Proof: We apply induction on the size of $\mathcal{P}$ (equivalent: the number of links $|\mathcal{L}|$).

If $|\mathcal{L}| = 0$ then $\mathcal{P}$ consist of a sole (and hence axiomatic) formula (being a proof net, hence non-empty and connected), for which $f = 2$ and $T = t = 0$.

If $|\mathcal{L}| \geq 1$ and $\mathcal{P}$ has a thin terminal link, then $\mathcal{P} = \mathrm{X}\Box(\mathcal{P}_1)$ where $\mathcal{P}_1$ is a proof net too. Now by induction hypothesis $\mathcal{P}_1$ satisfies $f_1 + t_1 = 2 + T_1$, and attaching the thin link yields $f = f_1 - 1$ and $t = t_1 + 1$. Hence also $\mathcal{P}$ satisfies the relation.
Now suppose $\mathcal{P}$ does not have a thin terminal link.

In case $|\mathcal{L}| = 1$, then the unique link L is terminal and hence L is unary or thick. Now $\mathcal{P} = \widehat{L}$ has $f = 2$, $t = 0$, $T = 0$ or $f = 3$, $t = 0$, $T = 1$ respectively, both satisfying the relation.

In case $|\mathcal{L}| \geq 2$, we are in the conditions of lemma 6.2, which gives us a splitting formula c and a partition of $\mathcal{P}$ into two non-trivial proof nets $\mathcal{P}_1 := \varepsilon_c$ and $\mathcal{P}_2 := \varepsilon^c$. By induction hypothesis those $\mathcal{P}_i$ satisfy $f_i + t_i = 2 + T_i$, and since $f = f_1 + f_2 - 2$ we get

$$\begin{aligned} f + t &= (f_1 + f_2 - 2) + (t_1 + t_2) \\ &= (f_1 + t_1) + (f_2 + t_2) - 2 \\ &= (2 + T_1) + (2 + T_2) - 2 \\ &= 2 + (T_1 + T_2) = 2 + T \end{aligned}$$

as desired. This proves the first part.

Now let us suppose that there is a proof net $\mathcal{P}_1$ of $X \vdash Y$ and moreover a proof net $\mathcal{P}_2$ of $Y \vdash X$. Look at the cut-free and η-expanded normal forms $\widehat{\mathcal{P}_1}$ and $\widehat{\mathcal{P}_2}$ of both. Observe that $\widehat{\mathcal{P}_1}$ may be constructed as the union of the lower construction trees T^a $(a \in X)$ and the upper construction trees T_b $(b \in Y)$, followed by an identification of the atomic formulas, while for $\widehat{\mathcal{P}_2}$ a similar construction applies. Hence if $\widehat{\mathcal{P}_1}$ has t thin binary links and T thick binary links, then the other normal form $\widehat{\mathcal{P}_2}$ must have T thin binary links and t thick binary links. (By the way, $\widehat{\mathcal{P}_2}$ does not have to be the *rotation* of $\widehat{\mathcal{P}_1}$; see example 7.10.3.) This gives us the relation $f + T = 2 + t$. Together with $f + t = 2 + T$ we conclude that $T = t$ and hence that $f = 2$. ■

This lemma shows that is it no restriction to study the predicates $\dashv\vdash$ and $\dashv_{\mathrm{r}}\vdash$ on

1MLL	
connectives	
0-ary	p_i, $p_i^{\perp}$
1-ary	
2-ary	$\otimes$, $\parr$
operations	
0-ary	
1-ary	$[\cdot]^{\perp}$
2-ary	$\multimap$

$$\xrightarrow{\nu}$$

$$p_i^{\perp} \mapsto (p_i)^{\perp}$$

$$\xleftarrow{\pi}$$

MLL	
connectives	
0-ary	p_i
1-ary	$(\cdot)^{\perp}$
2-ary	$\otimes$, $\parr$, $\multimap$
operations	
0-ary	
1-ary	
2-ary	

Figure 5: Definitions of syntactic formulas for both **1MLL** and **MLL**.

multisets of length one only (i.e. on formulas). Indeed,

$$\begin{array}{llllllll}
X \dashv\vdash Y & \iff & X \dashv\vdash Y & \text{and} & |X|+|Y|=2 & & & \\
 & \iff & X=a,b & \text{and} & Y=\emptyset & \text{and} & a,b \dashv\vdash & \text{or} \\
 & & X=a & \text{and} & Y=b & \text{and} & a \dashv\vdash b & \text{or} \\
 & & X=\emptyset & \text{and} & Y=a,b & \text{and} & \dashv\vdash a,b & \\
 & \iff & X=a,b & \text{and} & Y=\emptyset & \text{and} & a \dashv\vdash (b)^{\perp} & \text{or} \\
 & & X=a & \text{and} & Y=b & \text{and} & a \dashv\vdash b & \text{or} \\
 & & X=\emptyset & \text{and} & Y=a,b & \text{and} & (a)^{\perp} \dashv\vdash b &
\end{array}$$

which expresses $\dashv\vdash$ completely in terms of its behaviour on formulas. A similar result holds for $\dashv_{\mathrm{r}}\vdash$.

We say a is *(rotatable)-provably equivalent* to b iff $a \dashv\vdash b$ ($a \dashv_{\mathrm{r}}\vdash b$), and it is easy to check that both relations are congruence relations on **MLL**-formulas.

In order to completely characterize the congruence relation $\dashv_{\mathrm{r}}\vdash$, we will now introduce the language of the one-sided multiplicative linear logic (**1MLL**), which will turn out to be the quotient of **MLL**-formulas modulo the 'De Morgan laws'.

Starting from an infinite denumerable set of atoms $p_1, p_2, p_3, \ldots$ and their formal negations $p_1^{\perp}, p_2^{\perp}, p_3^{\perp}, \ldots$, the syntactic formulas of one-sided multiplicative linear logic (**1MLL**) are built up with the binary connectives $\otimes$ and $\parr$.

Linear negation $[\cdot]^\perp$ of a formula is inductively defined by the 'De Morgan laws':

$$\begin{aligned}
[p_i]^\perp &:= p_i^\perp \\
[p_i^\perp]^\perp &:= p_i \\
[x \otimes y]^\perp &:= [x]^\perp \parr [y]^\perp \\
[x \parr y]^\perp &:= [x]^\perp \otimes [y]^\perp
\end{aligned}$$

and linear implication is defined by

$$x \multimap y := [x]^\perp \parr y.$$

Observe that linear negation is an involution: $[[x]^\perp]^\perp = x$.

Every syntactic formula x of **1MLL** is just a syntactic formula of **MLL** when we replace the formal negations of atoms $p_i^\perp$ by actual negations of atoms $(p_i)^\perp$. The resulting formula $\nu(x)$ is $\multimap$-free and *'almost $\perp$-free'* (i.e. with no negations but negations of atoms), and may be formally defined by:

$$\begin{aligned}
\nu(p_i) &:= p_i \\
\nu(p_i^\perp) &:= (p_i)^\perp \\
\nu(x \otimes y) &:= \nu(x) \otimes \nu(y) \\
\nu(x \parr y) &:= \nu(x) \parr \nu(y)
\end{aligned}$$

The other way around we have a surjection

$$\begin{aligned}
\pi(p_i) &:= p_i \\
\pi(a \mathbin{\Box} b) &:= \pi(a) \mathbin{\Box} \pi(b) \\
\pi((a)^\perp) &:= [\pi(a)]^\perp
\end{aligned}$$

which actually computes the 'De Morgan quotient' on **MLL**, in a sense to be made precise in the following definition and lemma.

Definition 7.2 Let $\equiv$ be the smallest equivalence relation on **MLL**-formulas satisfying:

$$\begin{aligned}
a \equiv a', \quad b \equiv b' \quad &\Longrightarrow \quad a \mathbin{\Box} b \equiv a' \mathbin{\Box} b' && (0\Box) \\
a \equiv a' \quad &\Longrightarrow \quad (a)^\perp \equiv (a')^\perp && (0\perp) \\
&(a \otimes b)^\perp \equiv (a)^\perp \parr (b)^\perp && (1) \\
&(a \parr b)^\perp \equiv (a)^\perp \otimes (b)^\perp && (2) \\
&a \multimap b \equiv (a)^\perp \parr b && (3) \\
&((a)^\perp)^\perp \equiv a && (4)
\end{aligned}$$

♦

Lemma 7.3 The relation $\equiv$ on **MLL**-formulas is the kernel of the map π, i.e. for all syntactic **MLL**-formulas a and b the following holds:

$$a \equiv b \qquad \text{if and only if} \qquad \pi(a) = \pi(b)$$

♦

Proof: $\boxed{\Longrightarrow}$ Consider $\sim := \{(a,b) \mid \pi(a) = \pi(b)\}$. This is an equivalence relation satisfying:

- (0) (since π commutes with $\Box$ and $(\cdot)^\perp$);
- (1) and (2) (by definition of $[\cdot]^\perp$);
- (3) (by definition of $\cdot \multimap \cdot$);
- (4) (by the fact that $[\cdot]^\perp$ is an involution).

As $\equiv$ it the smallest such equivalence relation, we must have that $\equiv \subseteq \sim$, i.e. if $a \equiv b$ then $\pi(a) = \pi(b)$.

$\boxed{\Longleftarrow}$ We prove this by induction on the total length of the formulas a and b, distinguishing the nine cases $a = p_i$; $a = a_1 \otimes a_2$; $a = a_1 \parr a_2$; $a = a_1 \multimap a_2$; $a = (p_i)^\perp$; $a = (a_1 \otimes a_2)^\perp$; $a = (a_1 \parr a_2)^\perp$; $a = (a_1 \multimap a_2)^\perp$ and $a = ((a_1)^\perp)^\perp$. Suppose $\pi(a) = \pi(b)$. If $a = ((a_1)^\perp)^\perp$ then

$$\begin{aligned} \pi a_1 &= [[\pi a_1]^\perp]^\perp && \text{(since } [\cdot]^\perp \text{ is an involution)} \\ &= \pi((a_1)^\perp)^\perp \\ &= \pi a = \pi b \end{aligned}$$

whence

$$\begin{aligned} a = ((a_1)^\perp)^\perp &\equiv a_1 && \text{(by (4))} \\ &\equiv b && \text{(by induction hypothesis)} \end{aligned}$$

The case $b = ((b_1)^\perp)^\perp$ is treated similarly.
Now suppose a and b are of the other eight forms.
If $\pi a = p_i$, then $a = p_i = b$ whence $a \equiv b$.
If $\pi a = p_i^\perp$, then $a = (p_i)^\perp = b$ whence $a \equiv b$.
If $\pi a = x \otimes y$ for some x and y, then $a = a_1 \otimes a_2$; $a = (a_1 \parr a_2)^\perp$ or $a = (a_1 \multimap a_2)^\perp$, while b is of one of the similar forms, yielding nine subcases.
E.g. in the subcase that $a = a_1 \otimes a_2$ and $b = (b_1 \multimap b_2)^\perp$, then from $\pi a = \pi b$ it follows that

$\pi a_1 \otimes \pi a_2 = \pi((b_1 \multimap b_2)^\perp) = \pi b_1 \otimes [\pi b_2]^\perp$, whence $\pi a_1 = \pi b_1$ and $\pi a_2 = [\pi b_2]^\perp = \pi(b_2)^\perp$. This yields by induction hypothesis $a_1 \equiv b_1$ and $a_2 \equiv (b_2)^\perp$, whence

$$\begin{aligned} a = a_1 \otimes a_2 \equiv b_1 \otimes (b_2)^\perp & \qquad \text{(by } (0\otimes)) \\ \equiv ((b_1)^\perp)^\perp \otimes (b_2)^\perp & \qquad \text{(by (4) and } (0\otimes)) \\ \equiv ((b_1)^\perp \parr b_2)^\perp & \qquad \text{(by (2))} \\ \equiv (b_1 \multimap b_2)^\perp = b & \qquad \text{(by (3) and } (0\perp)) \end{aligned}$$

The other subcases are proved analogously.
If $\pi a = x \parr y$ for some x and y, then $a = a_1 \parr a_2$; $a = a_1 \multimap a_2$ or $a = (a_1 \otimes a_2)^\perp$, while b is of one of the similar forms, yielding again nine subcases that are also proved in a straightforward way. ■

From

$$\pi\nu = \mathrm{id}_{\mathbf{1MLL}}$$

we see that $\pi a = \pi b$ is equivalent to $\nu\pi a = \nu\pi b$ (which is a condition of **MLL**-formulas), whence we can also formulate lemma 7.3 as

$$a \equiv b \qquad \text{if and only if} \qquad \nu\pi(a) = \nu\pi(b)$$

We call $\nu\pi(a)$ the *normal form* of a. As $\pi(\nu\pi a) = (\pi\nu)\pi a = \pi a$, we see that $\nu\pi a \equiv a$.

Corollary 7.4 **1MLL** = **MLL**/$\equiv$ ⧫

Proof: We only use the result of lemma 7.3 and the fact that π is epi. Let us denote the congruence class of $a \in \mathbf{MLL}$ by $[a] \in \mathbf{MLL}/\equiv$. As π is epi, for every $x \in \mathbf{1MLL}$ there is a $\nu' x \in \mathbf{MLL}$ such that $\pi\nu' x = x$. Let us consider the following maps:

$$\begin{aligned} \mathbf{1MLL} &\leftrightarrows \mathbf{MLL}/\equiv \\ x &\mapsto [\nu' x] \\ \pi a &\leftarrow\!\shortmid [a] \end{aligned}$$

The last map is well-defined, because $[a] = [b]$ implies $\pi a = \pi b$. The one composite reads

$$x \mapsto [\nu' x] \mapsto \pi\nu' x = x$$

while the other one reads

$$[a] \mapsto \pi a \mapsto [\nu'\pi a] = [a]$$

(since $\pi(\nu'\pi a) = (\pi\nu')\pi a = \pi a$, implying $\nu'\pi a \equiv a$). This proves the bijective correspondence between the formulas of **1MLL** and the objects of **MLL**/$\equiv$. ■

At first sight dividing **MLL** by $\equiv$ destroys the inductive construction of the objects; the connectives have become operations:

$$[a] \Box [b] := [a \Box b]$$
$$[\, [a]\,]^{\perp} := [(a)^{\perp}]$$

which are well-defined by the requirements ($0\Box$) and ($0\perp$). The importance of this corollary is the fact that this quotient still has an inductive construction of the objects, viz. the same as **1MLL**.

Definition 7.5 Let $\simeq$ be the smallest equivalence relation on **MLL**-formulas satisfying (0) up to (4) and moreover:

$$a \otimes (b \otimes c) \simeq (a \otimes b) \otimes c \tag{5$\otimes$}$$
$$a \parr (b \parr c) \simeq (a \parr b) \parr c \tag{5$\parr$}$$
$$a \otimes b \simeq b \otimes a \tag{6$\otimes$}$$
$$a \parr b \simeq b \parr a \tag{6$\parr$}$$

♦

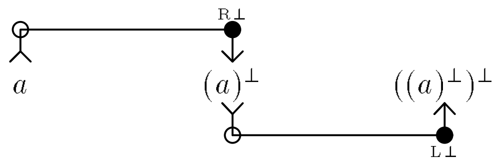
With the help of the next lemmas we will establish the fact that $\simeq$ and $\dashv_{\mathrm{r}}\vdash$ coincide.

Lemma 7.6 Let a and b be two syntactic **MLL**-formulas.
1. If $a \equiv b$, then $a \simeq b$.
2. If $a \simeq b$, then a and b are rotatable-provably equivalent.
3. a is rotatable-provably equivalent with its normal form $\nu\pi(a)$. ♦

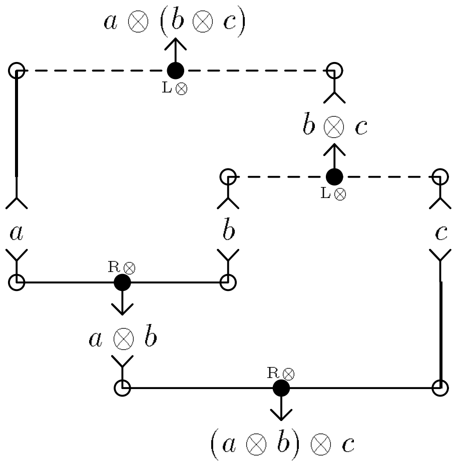
Proof of 1: The relation $\simeq$ is an equivalence relation satisfying at least (0) up to (4). As $\equiv$ is the smallest such equivalence relation, we must have that $\equiv \subseteq \simeq$, i.e. if $a \equiv b$ then $a \simeq b$.

Proof of 2: The relation $\dashv_{\mathrm{r}}\vdash := \{(a,b) \mid a \dashv_{\mathrm{r}}\vdash b\}$ is an equivalence relation satisfying

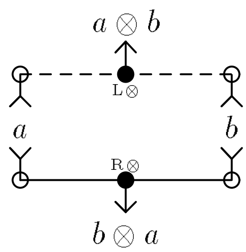
- (0), by pasting the two dual links L$\Box$ and R$\Box$ in the right order to the rotatable proof net(s);
- (1); see example 3.3.1;
- (2); similarly; see example 3.5;
- (3); see example 3.3.2;

- (4), by the proof net 

- (5), by the proof net



  for (5⊗) and by a similar proof net for (5⅋);

- (6), by the proof net 

 for (6⊗) and by a similar proof net for (6⅋).

As $\simeq$ it the smallest such equivalence relation, we must have that $\simeq \subseteq \dashv_{\mathrm{r}}\vdash$, i.e. if $a \simeq b$ then $a \dashv_{\mathrm{r}}\vdash b$.

Proof of 3: We know that $\nu\pi a \equiv a$, whence by part 1 we obtain $\nu\pi a \simeq a$, which yields $\nu\pi a \dashv_{\mathrm{r}}\vdash a$ by part 2. ■

Lemma 7.7 1. Let a and b be two $\otimes$-only ($⅋$-only) **MLL**-formulas with the same sequence of atoms. Then $a \simeq b$.
2. Let a and b be two $\otimes$-only ($⅋$-only) **MLL**-formulas with the same multiset of atoms. Then $a \simeq b$. ♦

Proof of 1: With induction on the number n of atoms.
If $n = 1$, then $a = p_1 = b$.

If $n \geq 2$, then $a = a' \otimes a''$ and $b = b' \otimes b''$.
If a' and b' are of the same length, they have the same sequence of atoms (since a and b do). Hence the induction hypothesis applies, yielding $a' \simeq b'$, and similarly $a'' \simeq b''$, whence by (0$\otimes$) also $a = a' \otimes a'' \simeq b' \otimes b'' = b$.
In the other case that (w.l.o.g.) a' is shorter than b', we can split up the sequence of atoms of b' into those of a' and of another formula c, such that $b' \simeq a' \otimes c$ (by induction hypothesis). Also $a'' \simeq c \otimes b''$, whence by (0$\otimes$) and (5$\otimes$) we get

$$\begin{aligned} a = a' \otimes a'' &\simeq a' \otimes (c \otimes b'') \\ &\simeq (a' \otimes c) \otimes b'' \\ &\simeq b' \otimes b'' = b \end{aligned}$$

Proof of 2: Let a and b be two such formulas with respective sequences of atoms

$$p_1, \ldots, p_{k-1}, p_k, p_{k+1}, p_{k+2}, \ldots, p_n$$

and

$$p_1, \ldots, p_{k-1}, p_{k+1}, p_k, p_{k+2}, \ldots, p_n$$

($1 \leq k < n$). Then by (6$\otimes$) we know that $p_k \otimes p_{k+1} \simeq p_{k+1} \otimes p_k$, whence by (0$\otimes$) we get (in case $k \neq 1, n-1$) $(c_1 \otimes (p_k \otimes p_{k+1})) \otimes c_2 \simeq (c_1 \otimes (p_{k+1} \otimes p_k)) \otimes c_2$. (In case $k = 1$ or $k = n-1$ the expression become even simpler.) On both sides we can apply part 1, yielding $a \simeq b$.

Now let a and b be two general such formulas with the same multiset of atoms. Then their sequences of atoms differ by a permutation, which can be decomposed into a number of transpositions. So there are formulas $a_0, \ldots, a_m$ such that $a_0 = a$; $a_m = b$; and a_i and a_{i+1} have sequences of atoms differing only by a transposition. Above we proved $a_i \simeq a_{i+1}$, so that we can conclude that also $a \simeq b$ holds.

For $⅋$-only formulas the result is proved analogously. ∎

Lemma 7.8 Suppose $a \dashv_{\mathrm{r}}\vdash b$ and let $\mathcal{P}$ be a cut-free and η-expanded rotatable proof net of $a \vdash b$. Then each (atomic) axiomatic formula is a subformula of both a and b. As a consequence, a and b have the same multiset of atoms. ◆

Proof: We know that $\mathcal{P}$ is the union of T^a and T_b, followed by an identification of the atomic formulas. Now suppose that an atomic subformula p of a is identified with another atomic subformula p of a. Let a' be the smallest subformula of a containing both occurrences of p. If a' is the main formula of a thick link L, then there is a switching of T^a yielding a path from the one occurrence of p to a' as well as a path from the other occurrence of p to a'. But this yields (after identification of the two occurrences of p) a cycle since L is thick. If L is thin, then L^* is thick, so the same argument applies and yields a cycle in $\mathcal{P}^*$. Hence every atomic subformula p of a is identified with an atomic subformula p of b. ∎

Theorem 7.9 For all syntactic **MLL**-formulas a and b the following holds:

$$a \simeq b \qquad \text{if and only if} \qquad a \dashv_{\mathrm{r}}\vdash b$$

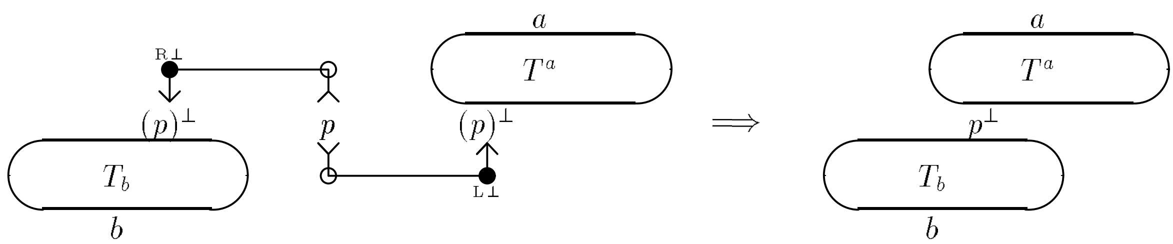
♦

Proof: $\boxed{\Longrightarrow}$ This is lemma 7.6.2.

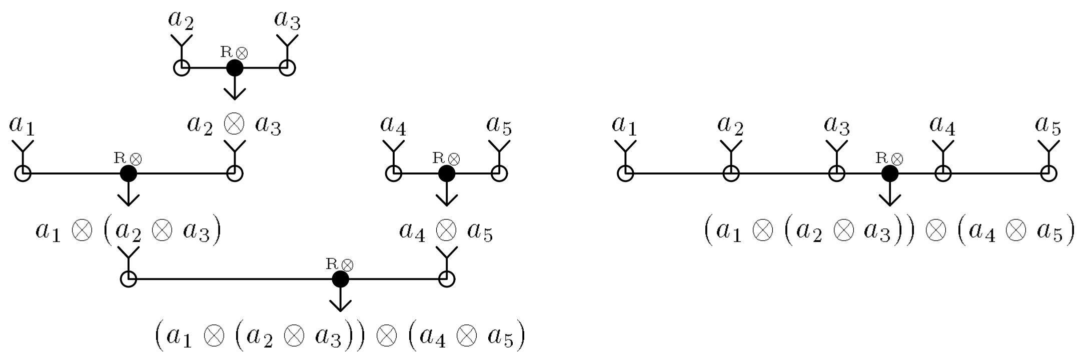
$\boxed{\Longleftarrow}$ We first prove this direction for $\multimap$-free and almost $\perp$-free **MLL**-formulas. This will be done by induction on the size of a cut-free and η-expanded rotatable proof net of $a \vdash b$.

Suppose $a \dashv_{\mathrm{r}}\vdash b$ where a and b are $\multimap$-free and almost $\perp$-free **MLL**-formulas.

Let $\mathcal{P}$ be a cut-free and η-expanded rotatable proof net of $a \vdash b$. Then we know $\mathcal{P}$ is the union of T^a and T_b containing only $\otimes$- and $⅋$-links, or $\perp$-links applied to atoms, followed by an identification of the atomic formulas, which is pairwise by lemma 7.8. If $(p)^\perp$ is a subformula of a, then p is a hypothesis of T^a. Hence it is a conslusion of T_b, yielding that $(p)^\perp$ is a subformula of b. Contracting the two $\perp$-links and replacing $(p)^\perp$ by the new atom $p^\perp$ yields a proof net which moreover is $\perp$-free.

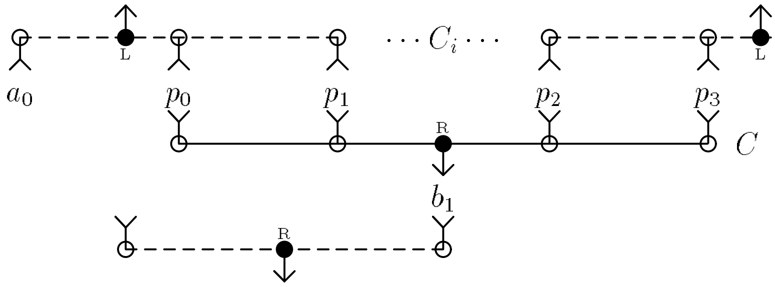


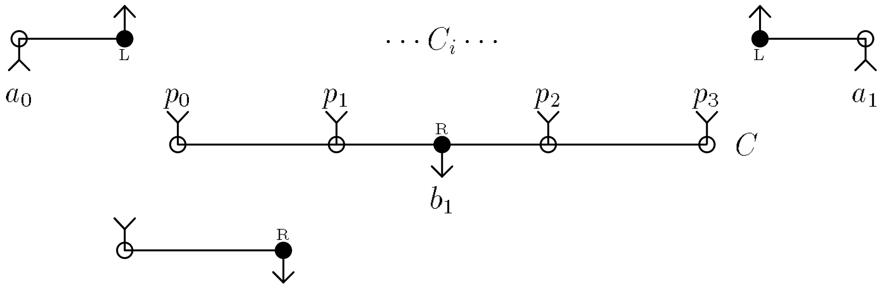
Hence let $\mathcal{P}$ be a cut-free and η-expanded rotatable proof net of $a \vdash b$, where $\mathcal{P}$ is the union of T^a and T_b containing only $\otimes$- and $⅋$-links, followed by a pairwise identification of the (new) atomic formulas. We will call a maximal connected component of $\mathcal{P}$ consisting entirely of thick (thin) links a thick (thin) cluster. By the absence of $\perp$-links, every link belongs to exactly one cluster. Every internal formula of a cluster is a flow formula: for if it was an axiomatic formula, then for some switching of the thin version of this cluster (in $\mathcal{P}$ or $\mathcal{P}^*$) this formula would be disconnected; and it can neither be a cut formula, since we assumed $\mathcal{P}$ to be cut-free. We call a leaf of a cluster an active (main) formula, if it is an active (main) formula of some link of the cluster. Each cluster contains exactly one main formula, while by the absence of $\multimap$-links we know that all active formulas are positive active formulas.



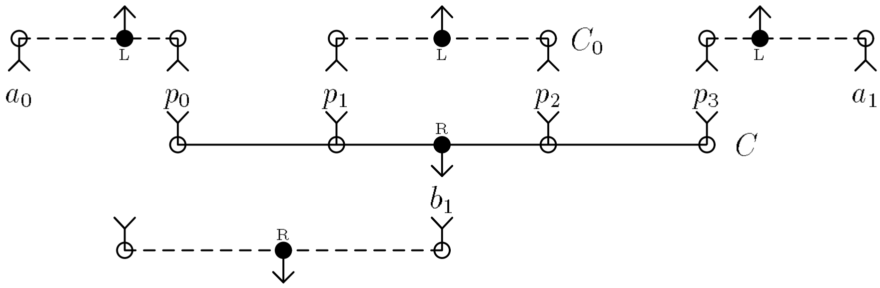
If there are no clusters, we get $a = p = b$, whence $a \simeq b$. Now suppose there is at least one cluster. Then there is a cluster with only **atomic** active formulas. For if not, then for every cluster C_i we could choose a non-atomic active formula, which can not be an axiomatic formula in our η-expanded rotatable proof net, and hence is a flow formula. So it is the main formula of another cluster C_{i+1}, yielding an infinite descending chain of subformulas; contradiction.

We may assume there is a thick cluster C with only atomic active formulas (because there is a cluster with only atomic active formulas in $\mathcal{P}$, which is thick or thin, and hence thin or thick in $\mathcal{P}^*$). The active formulas are active formulas of thin clusters C_i (since by the definition of cluster they cannot be thick), and we want to show that at least one of these thin clusters C_0 has all its active formulas among those of C. Well, if this would not be the case, then for each thin cluster choosing one of its active formula not among those of C can be extended to a switching where C is disconnected (after disconnecting its main formula as well in case b_1 is a **strict** subformula of b; see picture below); contradiction.





So there is a thin cluster C_0 having all its active formulas among those of C:



Now repeating the same story in $\mathcal{P}^*$ yields that the active formulas of C are among those of C_0. Hence C and C_0 face each other, so their main formulas c and c_0 are $\otimes$-only ($\wp$-only) **MLL**-formulas with the same multiset of atoms. This gives $c \simeq c_0$ by lemma 7.7.2.

Replacing C and C_0 by a unique new atom p_∞ results in a strictly smaller rotatable proof net $\mathcal{P}'$, yielding $a[p_\infty/c] \simeq b[p_\infty/c_0]$ by induction hypothesis. Backsubstituting c and c_0 (for which $c \simeq c_0$) we get $a[p_\infty/c][c/p_\infty] \simeq b[p_\infty/c_0][c_0/p_\infty]$, i.e. $a \simeq b$.

Now let arbitrary a and b be given for which $a \dashv_{\mathrm{r}}\vdash b$. Then by lemma 7.6.3 $\nu\pi a \dashv_{\mathrm{r}}\vdash a \dashv_{\mathrm{r}}\vdash b \dashv_{\mathrm{r}}\vdash \nu\pi b$, hence $\nu\pi a \dashv_{\mathrm{r}}\vdash \nu\pi b$. By the result established above we obtain $\nu\pi a \simeq \nu\pi b$, whence also $a \simeq \nu\pi a \simeq \nu\pi b \simeq b$, i.e. $a \simeq b$. ■

We summarize the results of this section by

$$\begin{array}{ccccccc}
a = b \implies & a \equiv b & \overset{\text{lemma 7.6.1}}{\Longrightarrow} & a \simeq b & & & \\
& \Updownarrow \text{lemma 7.3} & & \Updownarrow \text{th. 7.9} & & & \\
& \pi a = \pi b & & a \dashv_{\mathrm{r}}\vdash b & \implies & a \dashv\vdash b & \overset{\text{lemma 7.1}}{\Longrightarrow} \quad T_{T^a \cup T_b} = t_{T^a \cup T_b}
\end{array}$$

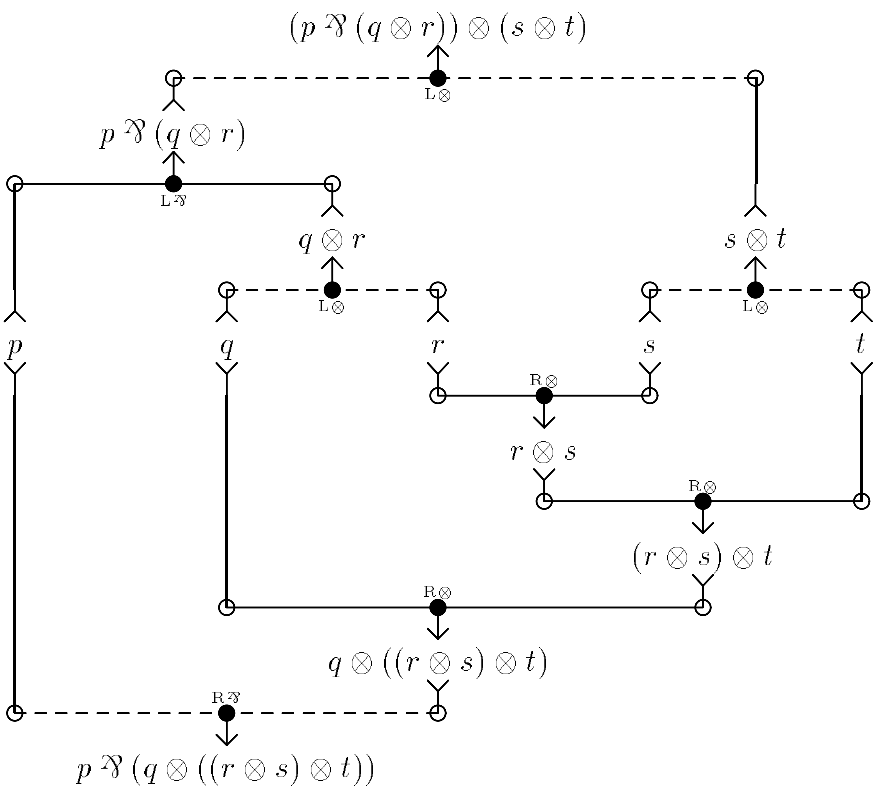
Observe that all implications above are strict:

Example 7.10 1. $(p \otimes q)^\perp \equiv (p)^\perp \parr (q)^\perp$, but $(p \otimes q)^\perp \neq (p)^\perp \parr (q)^\perp$.
2. $p \otimes q \simeq q \otimes p$ but $p \otimes q \not\equiv q \otimes p$ since $\pi(p \otimes q) = p \otimes q \neq q \otimes p = \pi(q \otimes p)$.
3. $(p \multimap p) \otimes p \dashv\vdash p$ but not by a rotatable proof net, since then both formulas would have the same multiset of atoms by lemma 7.8, which is not the case.
4. The first proof net in example 7.12 (say, of $a \vdash b$) has 4 thick and 4 thin links, but $b \vdash a$ is not provable. ♦

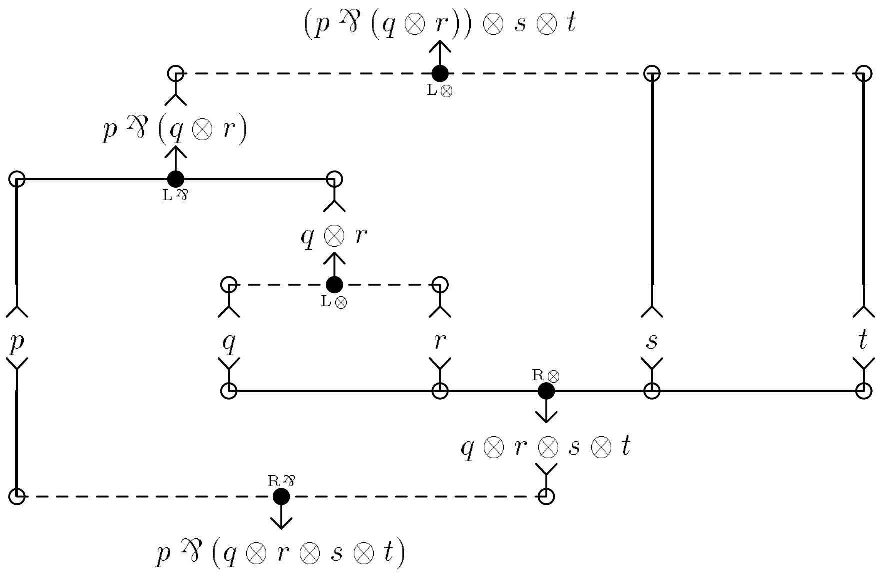
Lemma 7.11 Suppose a and b are **MLL**-formulas with coinciding multisets of atomic subformulas, in which moreover each atom has multiplicity one. Then $a \dashv\vdash b$ implies $a \dashv_{\mathrm{r}}\vdash b$. ♦

Proof: Given $a \dashv\vdash b$, there is a cut-free and η-expanded proof net $\mathcal{P}_1$ of $a \vdash b$ and a cut-free and η-expanded proof net $\mathcal{P}_2$ of $b \vdash a$. We know that $\mathcal{P}_1$ is the union of T^a and T_b followed by an identification of the atoms, while $\mathcal{P}_2$ has a similar description. From the requirement it follows that the mentioned identifications are unique in both cases, and hence they are the same. But then $\mathcal{P}_2 = \mathcal{P}_1^*$, which means that $a \dashv_{\mathrm{r}}\vdash b$. ■

Example 7.12 Consider the proof net



The clusters of this proof net may be represented as in the following diagram:



This proof net can not be rotatable, since the $\otimes$-cluster on q, r, s and t does not face exactly one thin cluster, while we know from the proof of theorem 7.9 that this is a necessary condition for rotatable proof nets. Moreover, from lemma 7.11 we now can deduce that

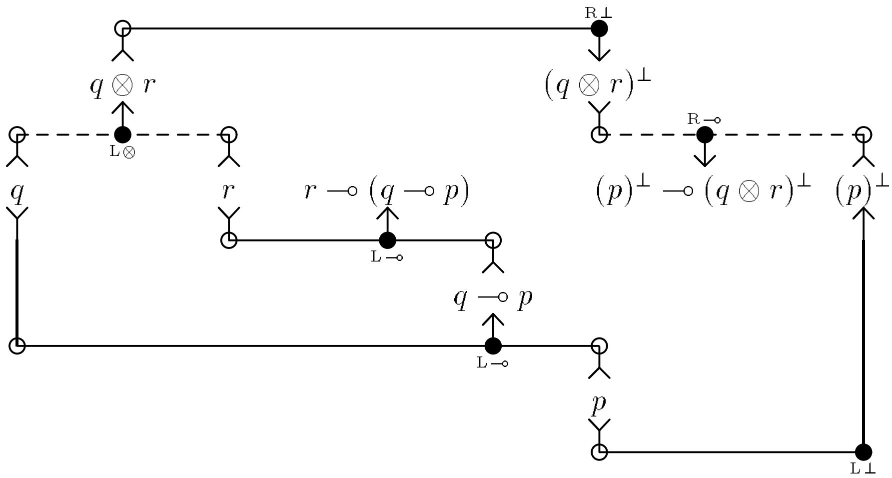
$$(p ⅋ (q \otimes r)) \otimes (s \otimes t) \vdash p ⅋ (q \otimes ((r \otimes s) \otimes t))$$

is provable, and that

$$p ⅋ (q \otimes ((r \otimes s) \otimes t)) \vdash (p ⅋ (q \otimes r)) \otimes (s \otimes t)$$

is not provable.

One easily checks that the next proof net is rotatable:



By theorem 7.9 this implies that $(p)^{\perp} \multimap (q \otimes r)^{\perp}$ and $r \multimap (q \multimap p)$ must be $\simeq$-equivalent. We will show this by computing their normal forms.

The normal form of $(p)^\perp \multimap (q \otimes r)^\perp$ is

$$\begin{aligned}
\nu\pi((p)^\perp \multimap (q \otimes r)^\perp) &= \nu([\pi(p)]^\perp \multimap [\pi(q) \otimes \pi(r)]^\perp) \\
&= \nu([p]^\perp \multimap [q \otimes r]^\perp) \\
&= \nu(p^\perp \multimap (q^\perp \parr r^\perp)) \\
&= \nu([p^\perp]^\perp \parr (q^\perp \parr r^\perp)) \\
&= \nu(p \parr (q^\perp \parr r^\perp)) \\
&= p \parr ((q)^\perp \parr (r)^\perp)
\end{aligned}$$

while the normal form of $r \multimap (q \multimap p)$ is

$$\begin{aligned}
\nu\pi(r \multimap (q \multimap p)) &= \nu(\pi(r) \multimap (\pi(q) \multimap \pi(p)) \\
&= \nu(r^\perp \parr (q^\perp \parr p)) \\
&= (r)^\perp \parr ((q)^\perp \parr p)
\end{aligned}$$

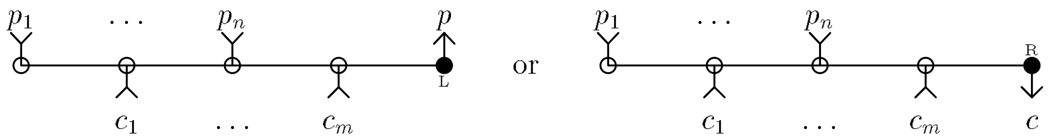
and these two normal forms are $\simeq$-equivalent by ($5\parr$) and ($6\parr$) (and ($0\parr$)). As we also know that formulas are $\simeq$-equivalent to their normal forms (actually by (0) up to (4)), we indeed find that $(p)^\perp \multimap (q \otimes r)^\perp$ and $r \multimap (q \multimap p)$ are $\simeq$-equivalent. ♦

8 Graphical representations

Formally we defined a *link* L in S to be a quadruple

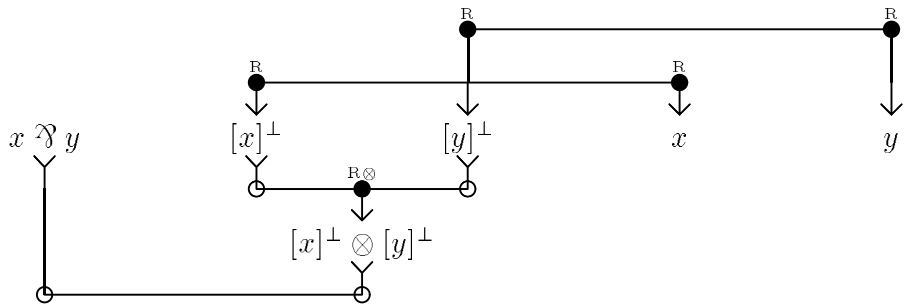
$$\langle P_{\mathrm{in}}, P_{\mathrm{out}}, C_{\mathrm{in}}, C_{\mathrm{out}} \rangle$$

of multisets of formulas of S, which we represented by a picture like

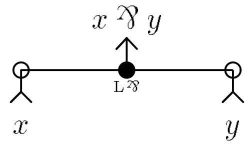


(where we indicate a thin link by a dotted horizontal bar). This representation is inspired by abbreviations in one-sided nets (to be treated in section 9).

Example 8.1 We can abbreviate the following part of a one-sided proof net



by



♦

The receipt for the representation of a proof structure $\langle S, \mathcal{L} \rangle$ is the following. In the plane, write down the formulas belonging to S, then draw a horizontal bar for each link L of $\mathcal{L}$, and finally connect each formula a with the bar (corresponding with the link) it is a conclusion (premise) of, by means of a vertical line segment 'above' (respectively 'below') a. As every formula of S is at most once a conclusion (premise) of a link (see def. 2.1), every formula is connected to at most two links: at most one 'above' and at most one 'below'. We have to assign a direction to each line segment in order to distinguish between an active formula of a link (line segment points to the bar) and the main formula of a link (line segment points away from the bar). Observe that on many occasions our vertical line segments have to be bent in order to reach the bar, as in the three proof nets of example 10.2.

In this section we will define two other representations in which we do not have to bend any vertical line segment anymore, and in which moreover the direction of all line segments is downward.

Given a proof structure $\langle S, \mathcal{L}\rangle$, we define the relation *a is active in b* on S by:

$$a \text{ is active in } b \quad \text{iff} \quad \text{for some link } L \in \mathcal{L}: \; a \text{ is active and } b \text{ is main}$$

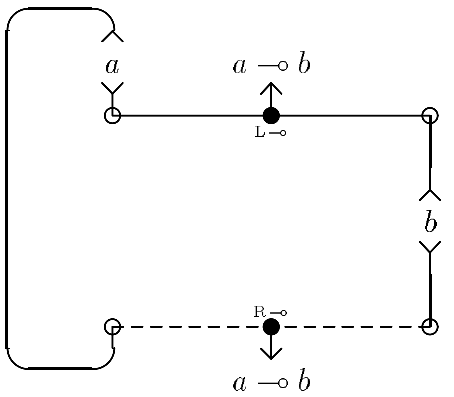
Let $\cdot \prec \cdot$ $(\cdot \preceq \cdot)$ be its transitive (and reflexive) closure. Since a is active in b implies — given the fact that links are of the forms as in def. 2.1 — that a is a strict subformula of b, it is clear that $\preceq$ is antisymmetric, whence a partial order. In case $a \preceq b$ we say a is *hereditary active* in b, or a is *above* b.

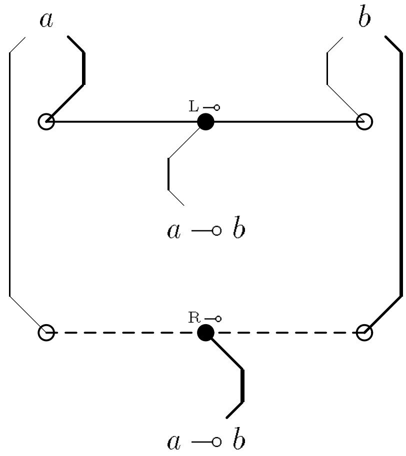
Let us now write down the formulas of S, such that (in the plane) a is put above b whenever a is hereditary active in b. Now draw a horizontal bar (in the plane) for each link L of $\mathcal{L}$ and locate it between its active formulas and its main formula. Finally connect a formula a to a bar above (corresponding to a link L) iff a is the main formula of L. We do this by means of a vertical line segment *in front of* the plane if a is a conlusion of L (L a right link), and by means of a vertical line segment *behind* the plane if a is a premise of L (L a left link). Similar, connect a formula a to a bar below (corresponding to a link L) iff a is an active formula of L. We do this by means of a vertical line segment *in front of* the plane if a is a premise of L, and by means of a vertical line segment *behind* the plane if a is a conclusion of L.

A direction for each vertical line segment corresponding to the plane representation, would be such that for an active formula of a link the line segment points to the bar (i.e. downward) and for the main formula of a link the line segment points away from the bar (i.e. downward as well). So this would not add any more information. However, the information which was contained in the direction of the vertical line segments in the plane representation is now contained in the possibility of a line segment to be either in front of or behind the plane.

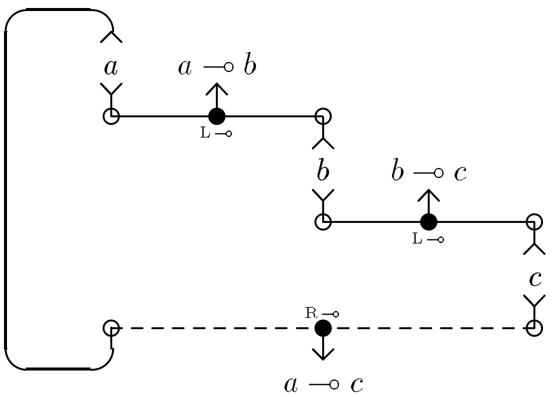
Example 8.2 We give three examples of this way of representing a proof net.

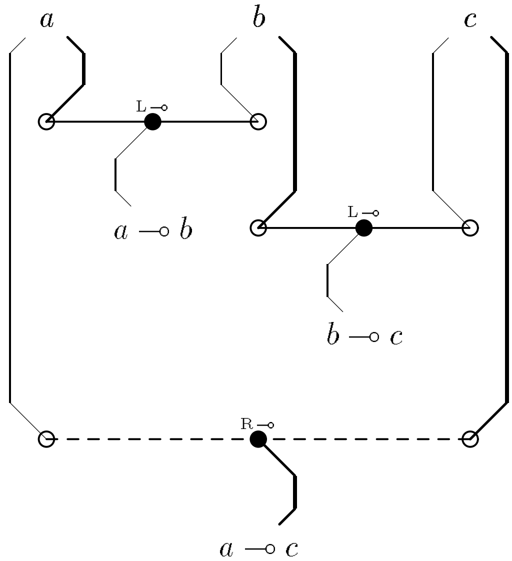
1.



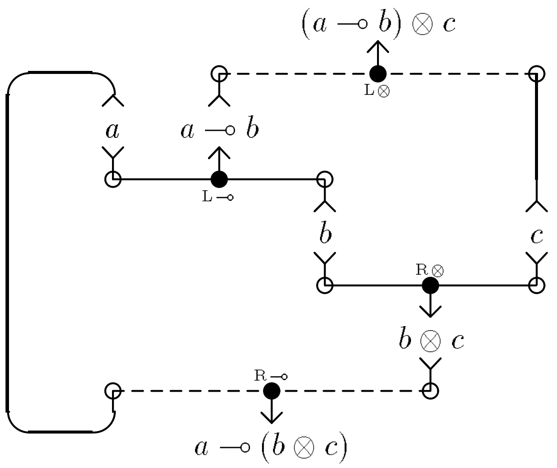


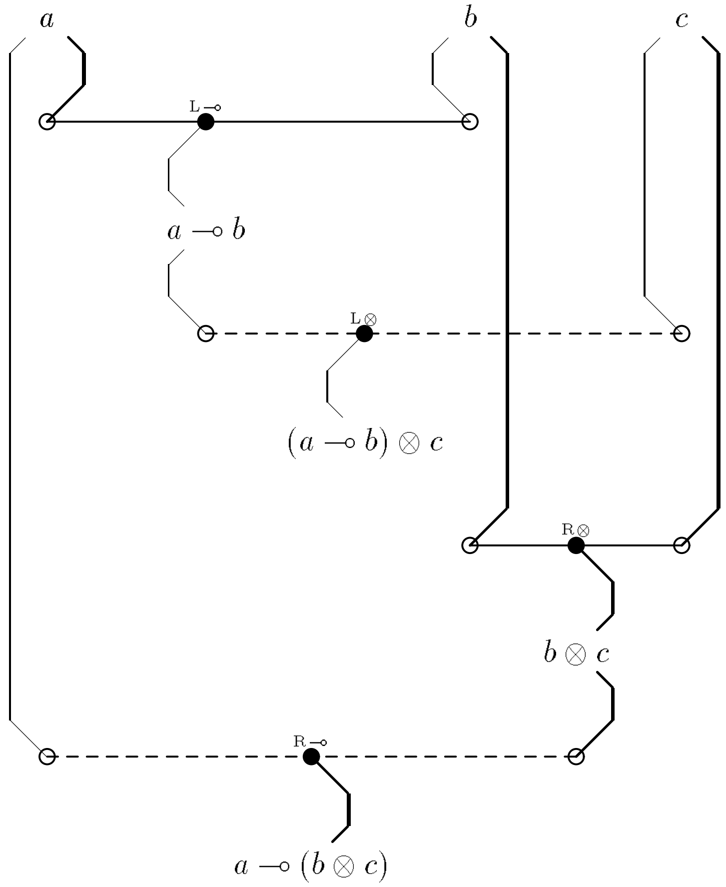
2.





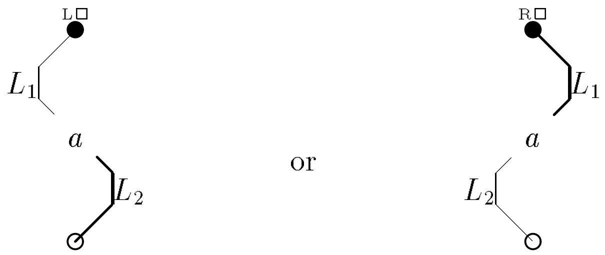
3.





♦

In fig. 6 we show what the different kinds of formulas of a proof structure look like in this new representation (cf. fig. 2). In particular we should realize that a formula a can not be connected to two links in one of the following ways:



In this situation a would be a premise (conclusion) of L_1 as well as of L_2, which is impossible by our definition of proof structure.

We obtain another variant of a representation when we draw the vertical line segments in the *plane*, and label them by a '+' or '−' in order to indicate whether it was in front of or below the plane. Moreover, replacing each axiomatic and cut formula a by a (horizontal) line segement a——a, the labeling can be extended to the formulas

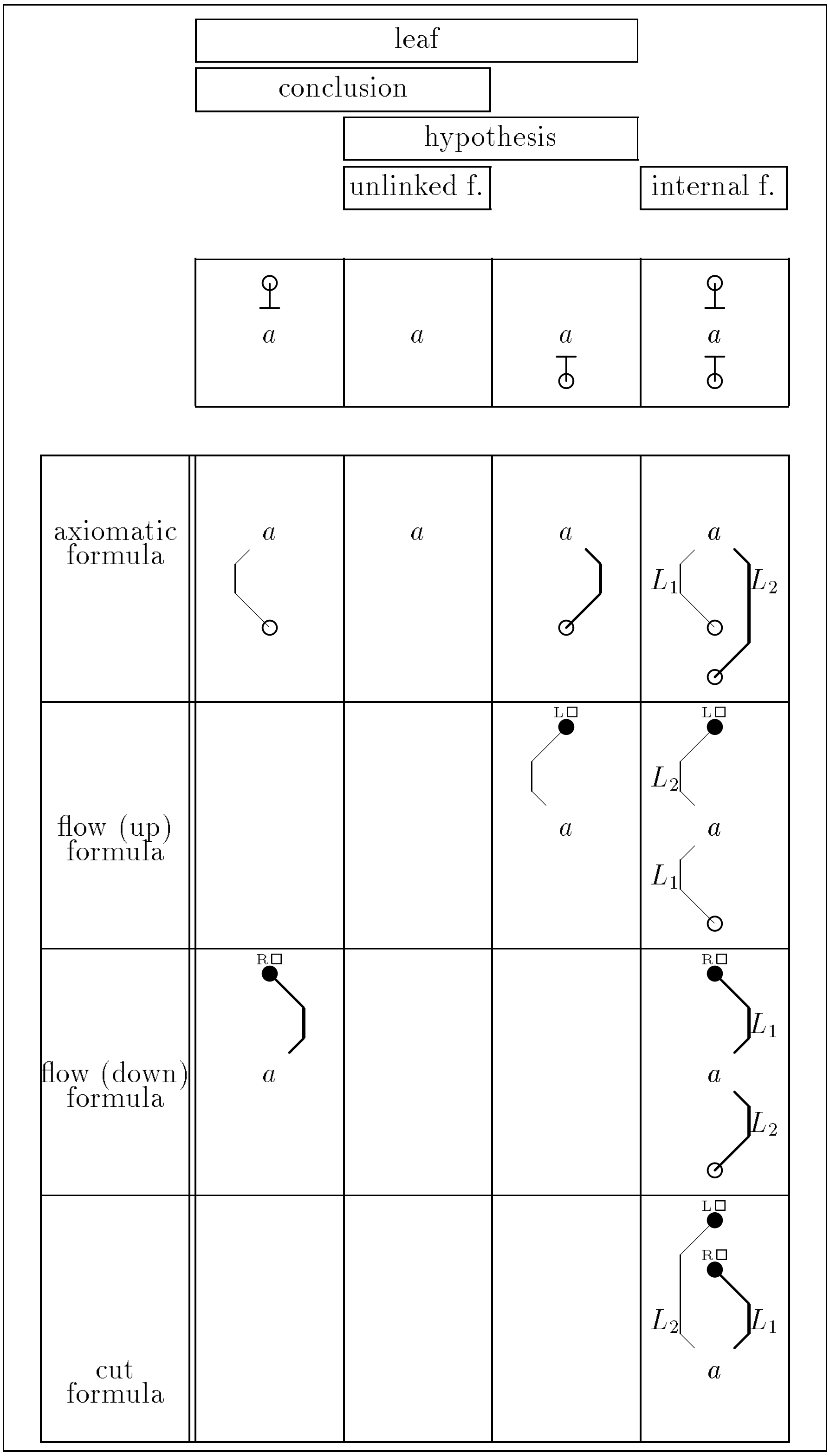
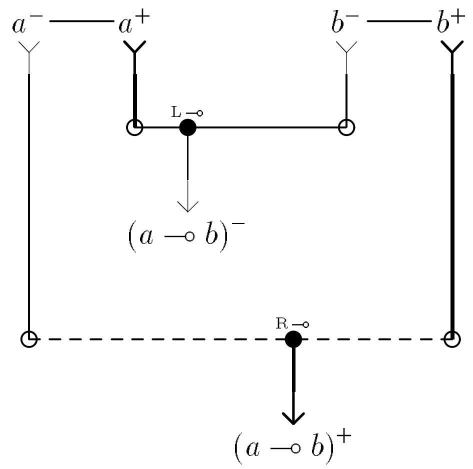


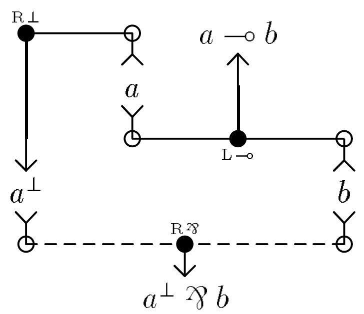
Figure 6: Predicates on the formulas of a proof structure $\mathcal{S} = \langle S, \mathcal{L} \rangle$.

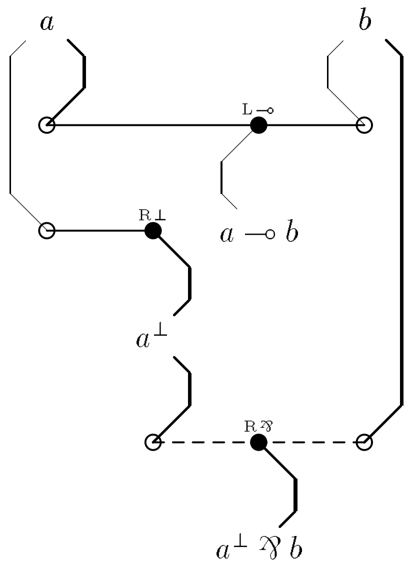
such that each vertical line segment is of the same sign as its formula: label flow down formulas by a '+'; label flow up formulas by a '−'; axiomatic and cut formulas are duplicated, and we label the one by '−' and the other by '+'.

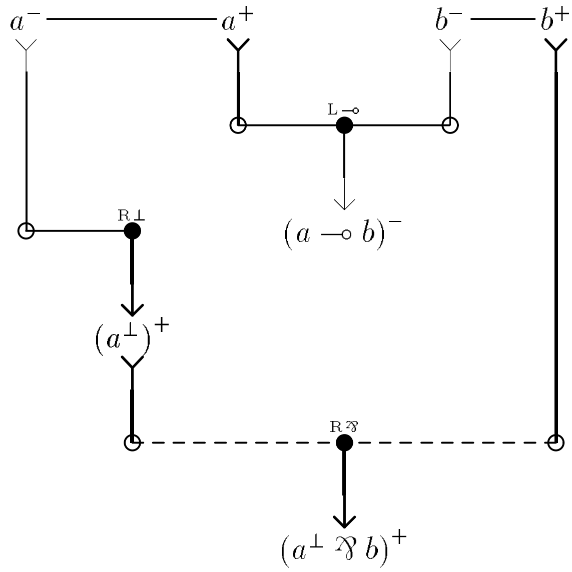
Example 8.3 The representation of the proof net of example 8.2.1 reads



The next three diagrams are representations of one and the same proof net







♦

Now every labeled formula is an endpoint of at least one (horizontal or vertical) line segment. If $a^\square$ is an endpoint of exactly one line segment, its underlying formula a is a hypothesis of $\mathcal{S}$ in case $a^\square = a^-$, and it is a conclusion of $\mathcal{S}$ in case $a^\square = a^+$. So $\mathcal{S}$ is a proof structure of $L_- \vdash L_+$, where $L_\square$ denotes the subset of S of $\square$-labeled leaves.

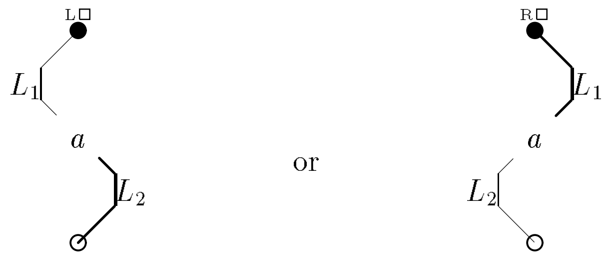
Observe that — viewed modulo De Morgan, and after interpreting the '$-$' by negation — these are actually nets with conclusions only. We define one-sided proof structures and proof nets in the next section. A translation from two-sided proof structures to one-sided proof structures is given by applying π on the formulas, and replacing every horizontal line segment by an identity link (Ax or Cut), replacing every thick (thin) binary link by R$\otimes$ (R$\parr$), and contracting every negation link. This is a well-defined operation, since the π-images of the formulas linked by a horizontal line segment are negations of each other, and since the π-images of the formulas of a link are really the formulas of a one-sided link, and since the π-images of the formulas of a $\perp$-link coincide. In the next section we will find a partial right inverse $(\cdot)^\bullet$ for π (viz. such that $\pi((\mathcal{S})^\bullet) = \mathcal{S}$ for one-sided proof structures without so-called trivial Ax-links).

We finish this section by giving the three equivalent formulations of def. 2.1 using the mentioned representations.

Proposition 8.4 The following are equivalent.

(i) A proof structure $\langle S, \mathcal{L}\rangle$ consists of a finite set S of formulas together with a set $\mathcal{L}$ of links in S of the forms as mentioned below (I), such that every formula of S is connected to at most one link 'above' and at most one link 'below';

(ii) A proof structure $\langle S, \mathcal{L}\rangle$ consists of a finite set S of formulas together with a set $\mathcal{L}$ of links in S of the forms as mentioned below (II), such that every formula of S is connected to at most one link in each of the four directions ('above & in front', et cetera), and moreover to at most two links in total, but not like:



(iii) A proof structure $\langle S, \mathcal{L}\rangle$ consists of a finite set S_{flow} of $\{-,+\}$-labeled formulas and a finite set $S_{\text{ax/cut}}$ of formulas, together with

- a set $\mathcal{L}$ of links in $S_{\text{flow}} \cup S^-_{\text{ax/cut}} \cup S^+_{\text{ax/cut}}$ of the forms as mentioned below (III);

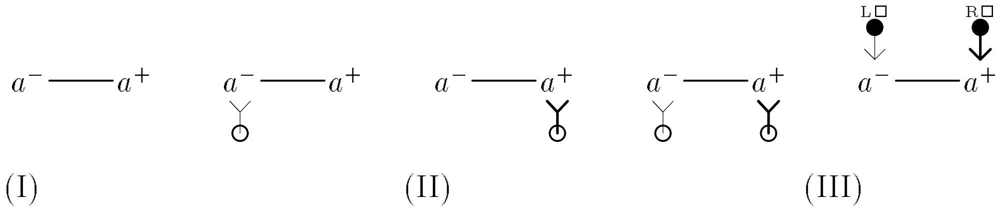
– for each $a \in S_{\mathrm{ax/cut}}$ exactly one horizontal line segment a^{-}——a^{+}

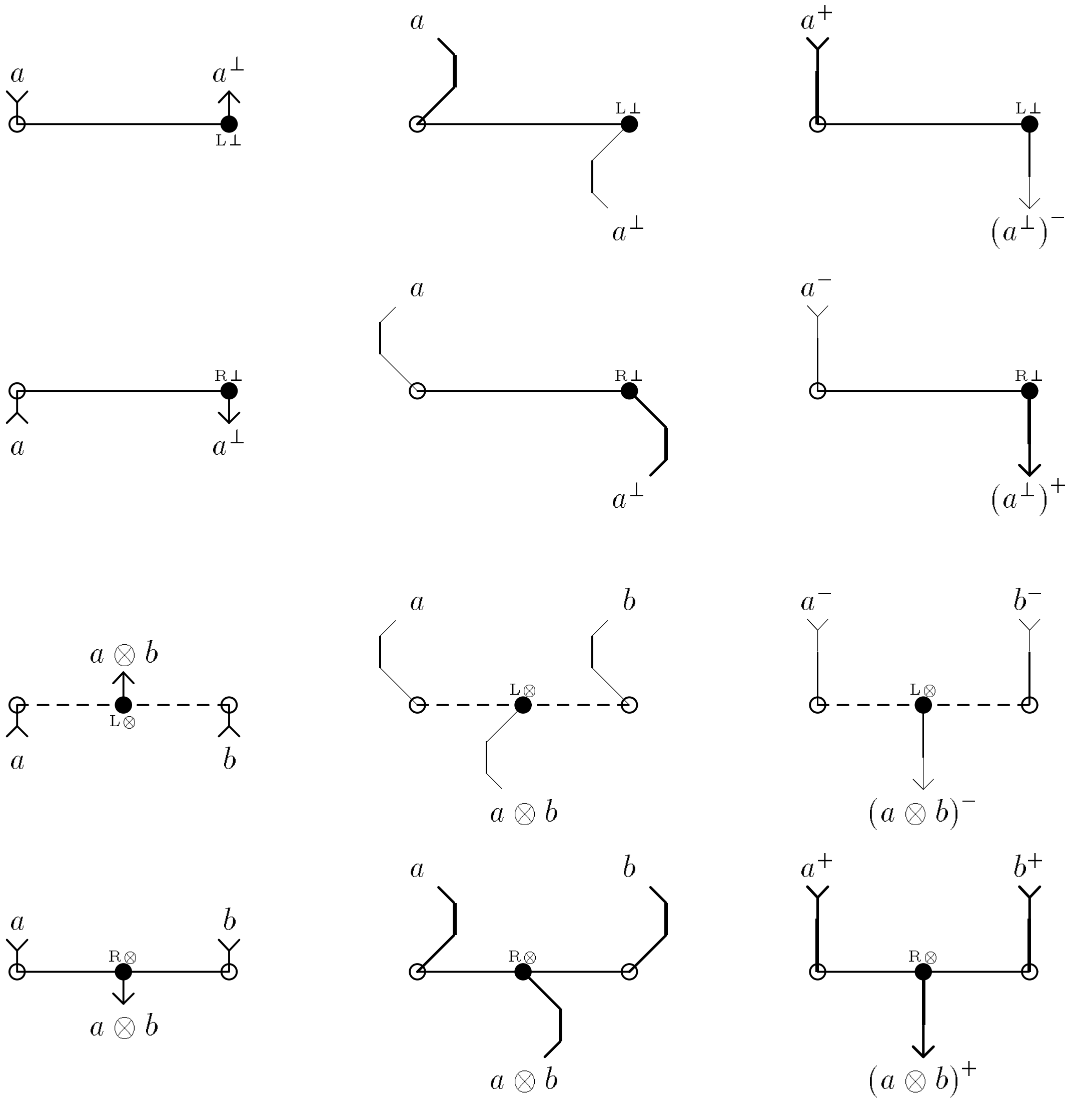
such that

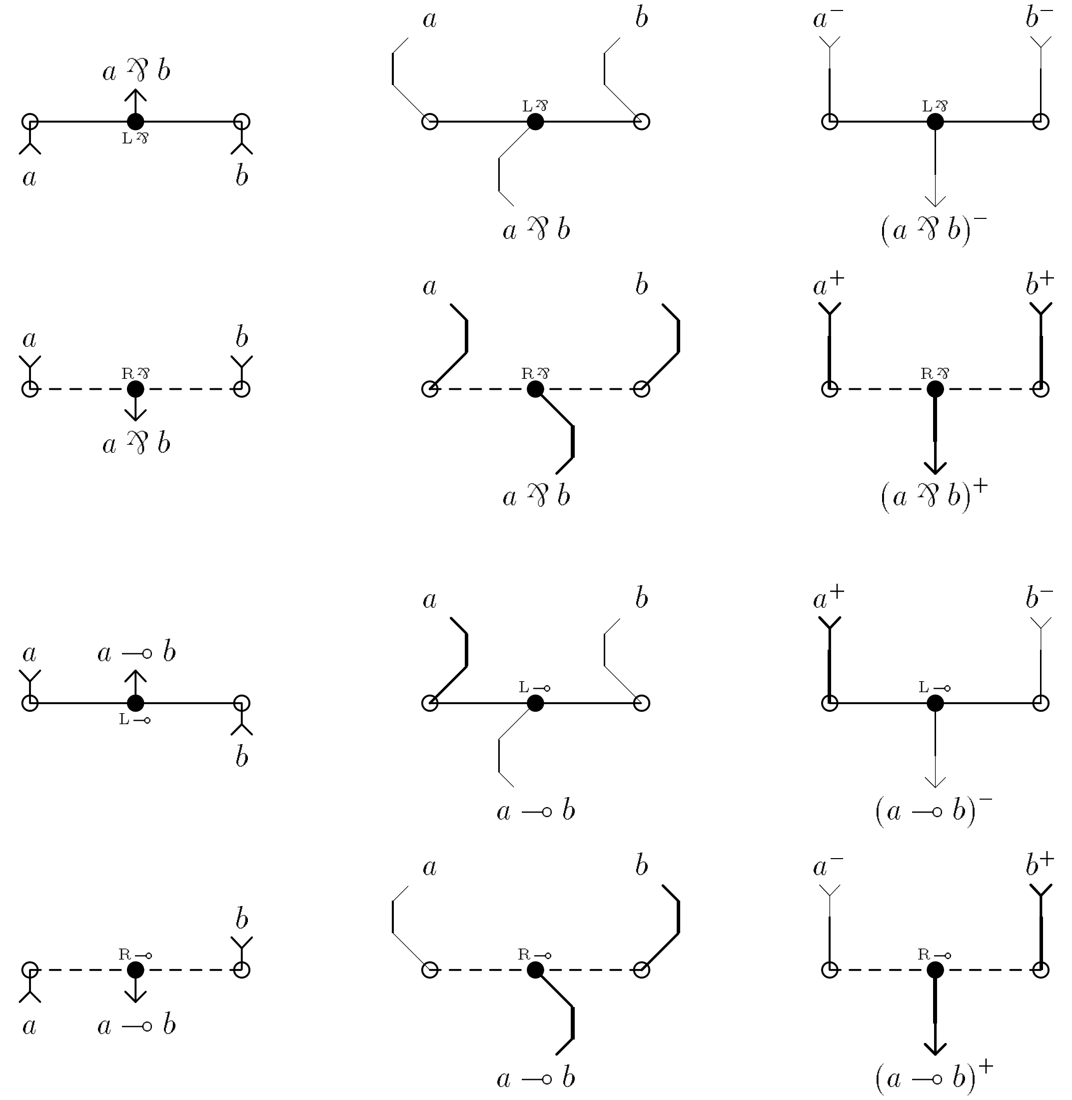
– every labeled formula of S_{flow} is attached to exactly one link above and to at most one link below;

– every labeled formula of $S^{-}_{\mathrm{ax/cut}} \cup S^{+}_{\mathrm{ax/cut}}$ is attached to at most one link;

– every horizontal line segment is between formulas a^{-} and a^{+} such that a^{-} is attached to a link above if and only if a^{+} is, i.e. these a^{-} and a^{+} are attached to links as in:



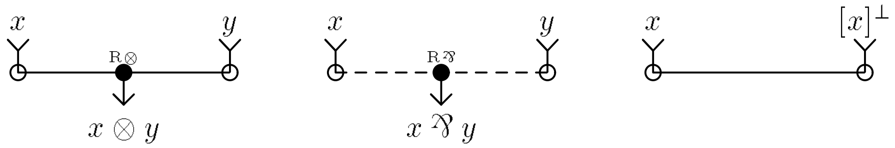



a ⅋ b
L ⅋
a
b
L ⅋
a ⅋ b
a⁻
b⁻
(a ⅋ b)⁻
R ⅋
a ⅋ b
a⁺
b⁺
(a ⅋ b)⁺
a ⊸ b
L ⊸
(a ⊸ b)⁻
R ⊸
(a ⊸ b)⁺

9 One-sided MLL

Proof structures of **1MLL** are defined as the smallest set containing AX-links

and closed under disjoint union and the lower attachment of R$\otimes$-, R$\parr$- and CUT-links:



Observe that AX- and CUT-links are not links in our original sense, since they have two and zero main formulas respectively. Moreover, the notions 'main formula' and 'conclusion' coincide now, as well as the notions 'active formula' and 'premise'.

We call an AX-link *trivial* if at least one of its main formulas is the active formula of a CUT-link. In that case also the concerning CUT-link is called trivial.

A switching σ for a proof structure is a choice for each R$\parr$-link of one of the premises. Proof nets of **1MLL** are defined similarly as proof nets of **MLL** (see def. 3.1).

Similar to lemma 3.10 we derive:

Lemma 9.1 The following are derivation rules of **1MLL**.

Identity rules:

$$\frac{}{\vdash x, [x]^{\perp}}\ \text{AX}$$

$$\frac{\vdash x, Y_1 \qquad \vdash [x]^{\perp}, Y_2}{\vdash Y_1, Y_2}\ \text{CUT}$$

Multiplicative logical rules:

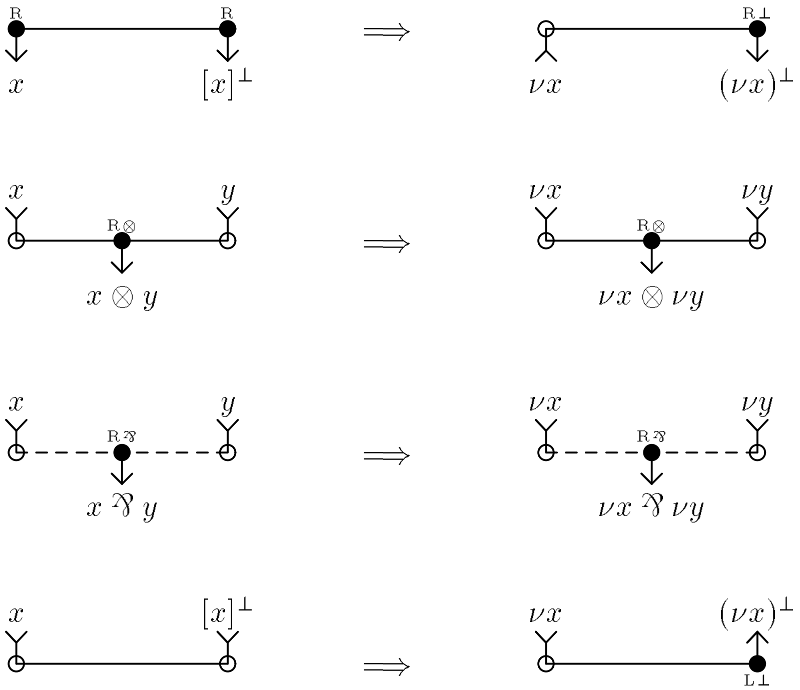
$$\frac{\vdash x, Y_1 \qquad \vdash y, Y_2}{\vdash x \otimes y, Y_1, Y_2}\ \text{R}\otimes \qquad\qquad \frac{\vdash x, y, Y}{\vdash x \parr y, Y}\ \text{R}\parr$$

♦

The aim of this section is to prove sequentialization for **1MLL** with the help of sequentialization for **MLL** (theorem 6.3).

To this end we have to map a **1MLL** proof net into an **MLL** proof net, which has an **MLL** sequentialization that we have to transform into a **1MLL** derivation.

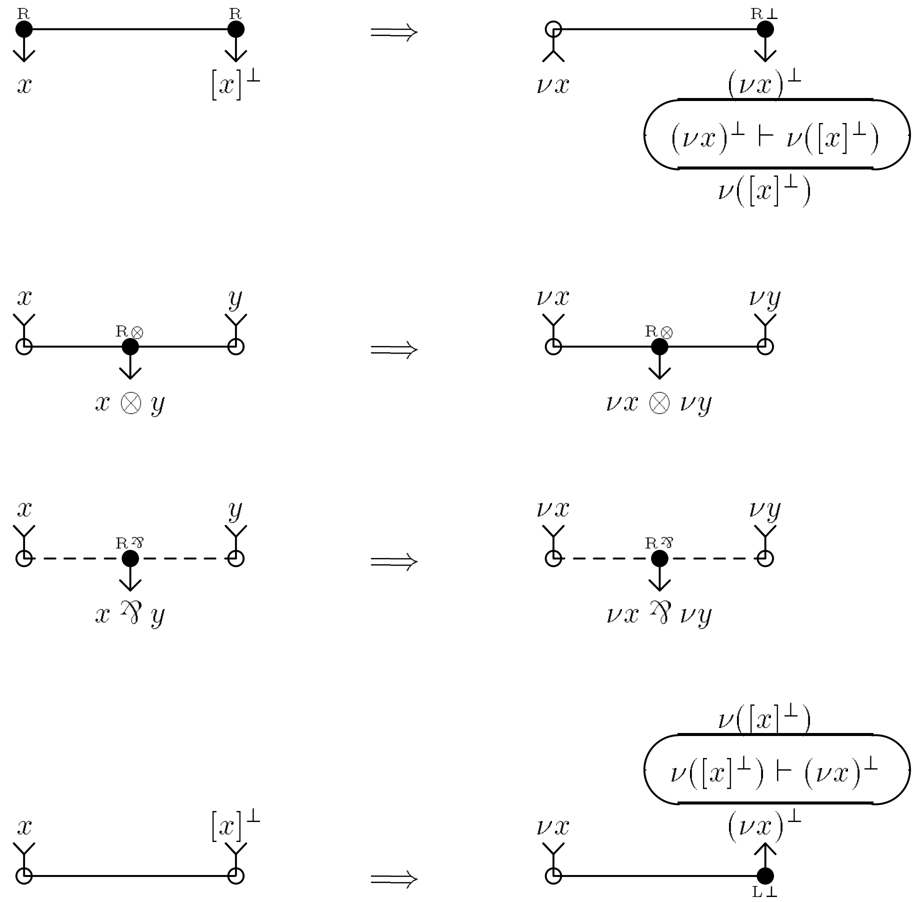
Example 9.2 We can try to translate each link into a link of def. 2.1 in the following way:



Observe that a choice has to be made for the AX- and the CUT-link which formula plays the active role (x), and which one the main role ($[x]^{\perp}$). Moreover, we encounter the problem that the main formula in the translation of a link may not coincide with the ν-image of the main formula in the original. For the R$\otimes$- and R$\parr$-link this is not the case, since $\nu(x \otimes y) = \nu x \otimes \nu y$ and $\nu(x \parr y) = \nu x \parr \nu y$. But for the AX- and CUT-link $[x]^{\perp}$ is translated into $(\nu x)^{\perp}$ instead of $\nu([x]^{\perp})$. However, from lemma 7.6.3 (applied to $(\nu x)^{\perp}$) we can deduce that

$$
\begin{aligned}
(\nu x)^{\perp} &\dashv_{\mathrm{r}}\vdash \nu\pi((\nu x)^{\perp}) \\
&= \nu([\pi(\nu x)]^{\perp}) \\
&= \nu([x]^{\perp})
\end{aligned}
$$

and hence we can adjust our translation as follows:



But now the translated proof net will have a sequentialization from which we can hardly obtain a **1MLL**-sequentialization of the original proof net: for many **1MLL**-formulas we have inserted an **MLL**-proof net which gives rise to many rules in the **MLL**-derivation that are superfluous in the **1MLL**-derivation. ♦

We will therefore define another translation that respects all **MLL**-links; i.e. non-trivial AX- and CUT-links will be translated into AX- and CUT-formulas (instead of $\perp$-links!), while the logical links will be translated into logical links of the same type (thick or thin). First we have to extend the whole theory above with the symbol $\multimapinv$. So we add the binary connective $\multimapinv$ to the language of **MLL**, while we define for syntactic formulas of **1MLL**:

$$x \multimapinv y := x \parr [y]^\perp.$$

Recall that $y \multimap x = [y]^\perp \parr x$, which is hence provably equivalent with $x \multimapinv y$.

We enrich **MLL** with the following dual links

1MLL

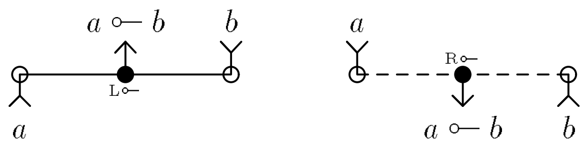
connectives	
0-ary	p_i, $p_i^\perp$
1-ary	
2-ary	$\otimes$, $\parr$
operations	
0-ary	
1-ary	$[\cdot]^\perp$
2-ary	$\multimap$, $\multimapinv$

$$\xrightarrow{\psi}$$
$$x \mapsto (x^\bullet, \overline{x})$$
$$\xleftarrow{\pi}$$

MLL$^+$

connectives	
0-ary	p_i
1-ary	$(\cdot)^\perp$
2-ary	$\otimes$, $\parr$, $\multimap$, $\multimapinv$
operations	
0-ary	
1-ary	
2-ary	

Figure 7: Definitions of syntactic formulas for both **1MLL** and **MLL$^+$**.



that give rise to the following extra rules:

$$\frac{X_1 \vdash b, Y_1 \qquad X_2, a \vdash Y_2}{X_1, X_2, a \multimapinv b \vdash Y_1, Y_2}\;\text{L}{\multimapinv} \qquad\qquad \frac{X, b \vdash a, Y}{X \vdash a \multimapinv b, Y}\;\text{R}{\multimapinv}$$

As $a \multimapinv b \dashv_{\mathrm{r}}\vdash b \multimap a$, this new connective $\multimapinv$ does not add anything new to the theory. It is just a variant of $\multimap$ that we need[9] in order to derive lemma 9.4.

[9]Without this new connective we would have been forced to define

$$\begin{aligned} \psi(x \otimes y) &= (y^\bullet \multimap x^\bullet, 1) && \text{if } \overline{x} = 1 \text{ and } \overline{y} = 0 \\ \psi(x \parr y) &= (y^\bullet \multimap x^\bullet, 0) && \text{if } \overline{x} = 0 \text{ and } \overline{y} = 1 \end{aligned}$$

instead of

$$\begin{aligned} \psi(x \otimes y) &= (x^\bullet \multimapinv y^\bullet, 1) && \text{if } \overline{x} = 1 \text{ and } \overline{y} = 0 \\ \psi(x \parr y) &= (x^\bullet \multimapinv y^\bullet, 0) && \text{if } \overline{x} = 0 \text{ and } \overline{y} = 1 \end{aligned}$$

but then lemma 9.4 would have held only modulo commutativity; e.g.

$$\pi((p \parr q^\perp)^\bullet) = \pi(q \multimap p) = q^\perp \parr p$$

Next we define a translation on syntactic **1MLL**-formulas that maps x and $[x]^\perp$ to the same $(\cdot)^\perp$-free **MLL**$^+$-formula $x^\bullet$ that is rotatable-provably equivalent to $\nu(x)$ or $\nu([x]^\perp)$. Simultaneously we define a label $\overline{x}$ that keeps track of whether $\pi(x^\bullet) = x$ ($\overline{x} = 0$) or $\pi(x^\bullet) = [x]^\perp$ ($\overline{x} = 1$).

Definition 9.3 We define $\psi(x) = (x^\bullet, \overline{x})$ as follows:

$$
\begin{aligned}
p_i &\mapsto (p_i, 0) \\
p_i^\perp &\mapsto (p_i, 1) \\
x \otimes y &\mapsto \begin{cases} (x^\bullet \otimes y^\bullet, 0) & \text{if } \overline{x} = 0 \text{ and } \overline{y} = 0, \\ (x^\bullet \multimap y^\bullet, 1) & \text{if } \overline{x} = 0 \text{ and } \overline{y} = 1, \\ (x^\bullet \multimapinv y^\bullet, 1) & \text{if } \overline{x} = 1 \text{ and } \overline{y} = 0, \\ (x^\bullet \parr y^\bullet, 1) & \text{if } \overline{x} = 1 \text{ and } \overline{y} = 1 \end{cases} \\
x \parr y &\mapsto \begin{cases} (x^\bullet \parr y^\bullet, 0) & \text{if } \overline{x} = 0 \text{ and } \overline{y} = 0, \\ (x^\bullet \multimapinv y^\bullet, 0) & \text{if } \overline{x} = 0 \text{ and } \overline{y} = 1, \\ (x^\bullet \multimap y^\bullet, 0) & \text{if } \overline{x} = 1 \text{ and } \overline{y} = 0, \\ (x^\bullet \otimes y^\bullet, 1) & \text{if } \overline{x} = 1 \text{ and } \overline{y} = 1 \end{cases}
\end{aligned}
$$

♦

Lemma 9.4 For every syntactic **1MLL**-formula x the following holds:

$$\pi(x^\bullet) = \begin{cases} x & \text{if } \overline{x} = 0, \\ [x]^\perp & \text{if } \overline{x} = 1. \end{cases} \tag{1}$$

$$x^\bullet = ([x]^\perp)^\bullet \tag{2}$$

$$\overline{x} = 1 - \overline{[x]^\perp} \tag{3}$$

♦

Proof of (1):

$\pi(p_i^\bullet) = \pi(p_i) = p_i$ which is o.k. since $\overline{p_i} = 0$.

$\pi((p_i^\perp)^\bullet) = \pi(p_i) = p_i = [p_i^\perp]^\perp$ which is o.k. since $\overline{p_i^\perp} = 1$.

For $x \otimes y$, in case $\overline{x} = 0$ and $\overline{y} = 0$:

$$\pi((x \otimes y)^\bullet) = \pi(x^\bullet \otimes y^\bullet) \ = \pi(x^\bullet) \otimes \pi(y^\bullet) \ \overset{\text{ih}}{=} x \otimes y$$

which is o.k. since $\overline{x \otimes y} = 0$. If $\overline{x} = 0$ and $\overline{y} = 1$:

$$\pi((x \otimes y)^\bullet) = \pi(x^\bullet \multimap y^\bullet) = \pi(x^\bullet) \multimap \pi(y^\bullet) \stackrel{\text{ih}}{=} x \multimap [y]^\perp \quad = [x]^\perp \parr [y]^\perp = [x \otimes y]^\perp$$

which is o.k. since $\overline{x \otimes y} = 1$. If $\overline{x} = 1$ and $\overline{y} = 0$:

$$\pi((x \otimes y)^\bullet) = \pi(x^\bullet \multimapinv y^\bullet) = \pi(x^\bullet) \multimapinv \pi(y^\bullet) \stackrel{\text{ih}}{=} [x]^\perp \multimapinv y \quad = [x]^\perp \parr [y]^\perp = [x \otimes y]^\perp$$

which is o.k. since $\overline{x \otimes y} = 1$. If $\overline{x} = 1$ and $\overline{y} = 1$:

$$\pi((x \otimes y)^\bullet) = \pi(x^\bullet \parr y^\bullet) \;= \pi(x^\bullet) \parr \pi(y^\bullet) \;\stackrel{\text{ih}}{=} [x]^\perp \parr [y]^\perp = [x \otimes y]^\perp$$

which is o.k. since $\overline{x \otimes y} = 1$.

For $x \parr y$ the four cases are treated analogously, and we have to use the fact that $[\cdot]^\perp$ is an involution.

We prove (2) and (3) by simultanious induction on a:
If $x = p_i$ or $x = p_i^\perp$ the result is clear.
For $x \otimes y$ we have $(x \otimes y)^\bullet = x^\bullet \,\Box\, y^\bullet$ where $\Box$ depends on $\overline{x}$ and $\overline{y}$. Now observe (by inspection of the four cases in definition 9.3) that

$$\begin{aligned} ([x \otimes y]^\perp)^\bullet &= ([x]^\perp \parr [y]^\perp)^\bullet \\ &= ([x]^\perp)^\bullet \,\Box\, ([y]^\perp)^\bullet \\ &= x^\bullet \,\Box\, y^\bullet = (x \otimes y)^\bullet \end{aligned}$$

(since by induction hypothesis $\overline{[x]^\perp} = 1 - \overline{x}$ and $\overline{[y]^\perp} = 1 - \overline{y}$) and moreover that

$$\begin{aligned} \overline{[x \otimes y]^\perp} = \overline{[x]^\perp \parr [y]^\perp} = 0 &\text{ iff } \overline{[x]^\perp} = 0 \text{ or } \overline{[y]^\perp} = 0 \\ &\text{ iff } \overline{x} = 1 \text{ or } \overline{y} = 1 \\ &\text{ iff not } (\overline{x} = 0 \text{ and } \overline{y} = 0) \\ &\text{ iff not } \overline{x \otimes y} = 0 \\ &\text{ iff } \overline{x \otimes y} = 1 \end{aligned}$$

whence

$$\overline{[x \otimes y]^\perp} = \overline{[x]^\perp \parr [y]^\perp} = 1 - \overline{x \otimes y}.$$

This also proves the case $x \parr y$ (by substituting $[x]^\perp$ for x and $[y]^\perp$ for y and using the fact that $[\cdot]^\perp$ is an involution). ■

The importance of this lemma is the fact that it divides the syntactic formulas of **1MLL** into two parts: the *positive* x with $\overline{x} = 0$ and the *negative* x with $\overline{x} = 1$. Moreover, in this sense x and $[x]^\perp$ are of opposite sign. Hence now we are able to assign a unique **MLL**$^+$-formula to every Ax- or Cut-link, viz. $x^\bullet$ (that coincides with $([x]^\perp)^\bullet$).

Example 9.5 An atom p is positive, while its formal negation $p^\perp$ is negative.

The formulas in the upper part of this box are positive, and the others are negative.

$p \otimes q$	$p \parr q$
$p \otimes q^\perp$	$p \parr q^\perp$
$p^\perp \otimes q$	$p^\perp \parr q$
$p^\perp \otimes q^\perp$	$p^\perp \parr q^\perp$

These formulas may also be expressed modulo outermost $[\cdot]^\perp$ in positive atoms by means of $\otimes$ and $\parr$ and the defined operations $\multimap$ and $\multimapinv$, which gives respectively

$p \otimes q$	$p \parr q$
$[p \multimap q]^\perp$	$p \multimapinv q$
$[p \multimapinv q]^\perp$	$p \multimap q$
$[p \parr q]^\perp$	$[p \otimes q]^\perp$

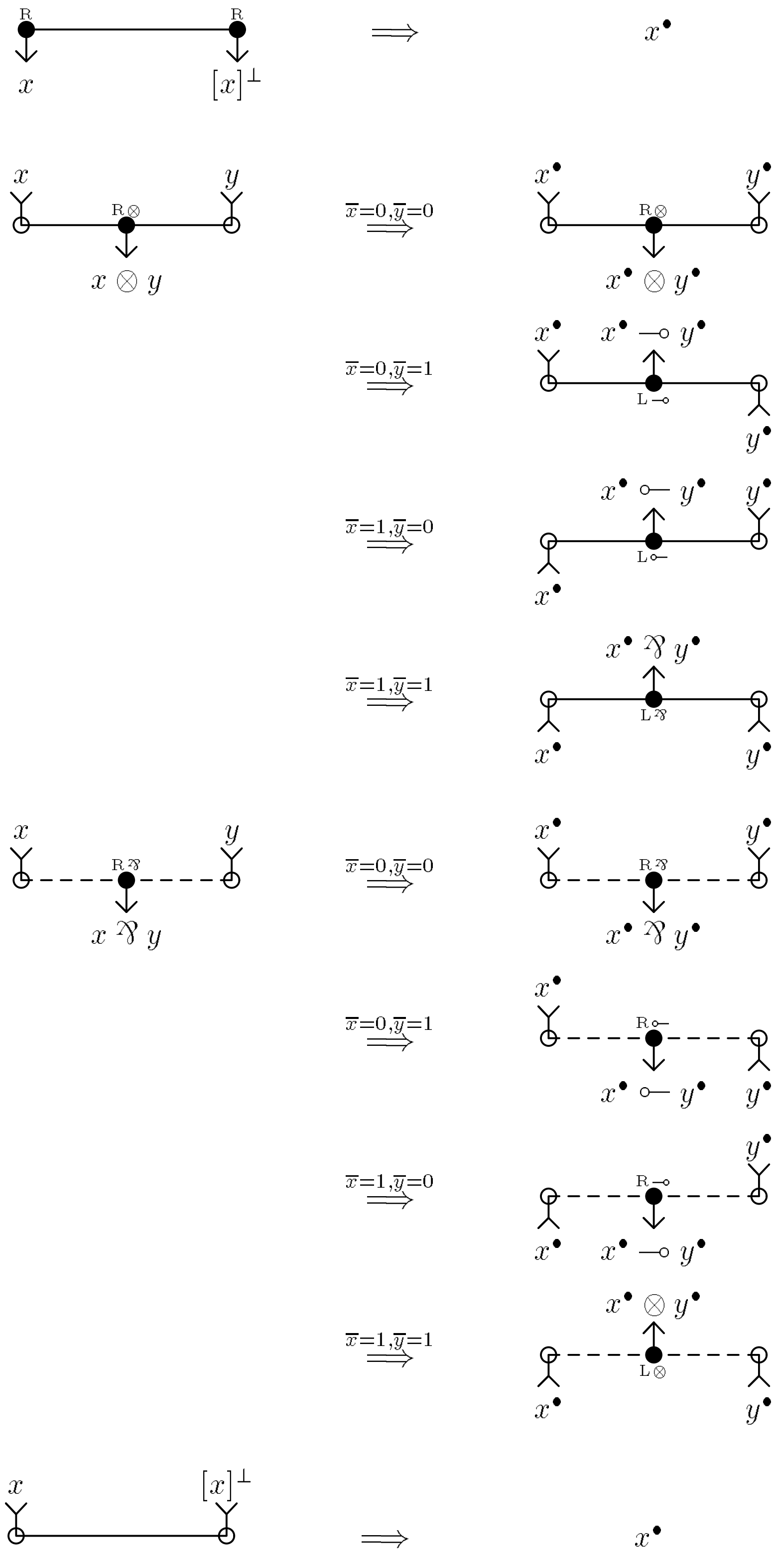
which explains the words 'positive' and 'negative'. In theorem 10.9 we will see that this way of expressing **1MLL**-formulas is unique. ♦

From def. 9.3 it is immediately clear that the following boolean characterization may be used in order to compute the label $\overline{x}$:

$$\overline{x \otimes y} = 0 \text{ iff } \overline{x} = 0 \text{ and } \overline{y} = 0;$$
$$\overline{x \parr y} = 0 \text{ iff } \overline{x} = 0 \text{ or } \overline{y} = 0.$$

We will translate the logical links such that positive formulas stream down, while negative formulas stream up. That is, if x is conclusion of L_1 and/or premise of L_2, then $x^\bullet$ is conclusion of $L_1^\bullet$ and/or premise of $L_2^\bullet$ in case x is positive, while $x^\bullet$ is premise of $L_1^\bullet$ and/or conclusion of $L_2^\bullet$ in case x is negative.

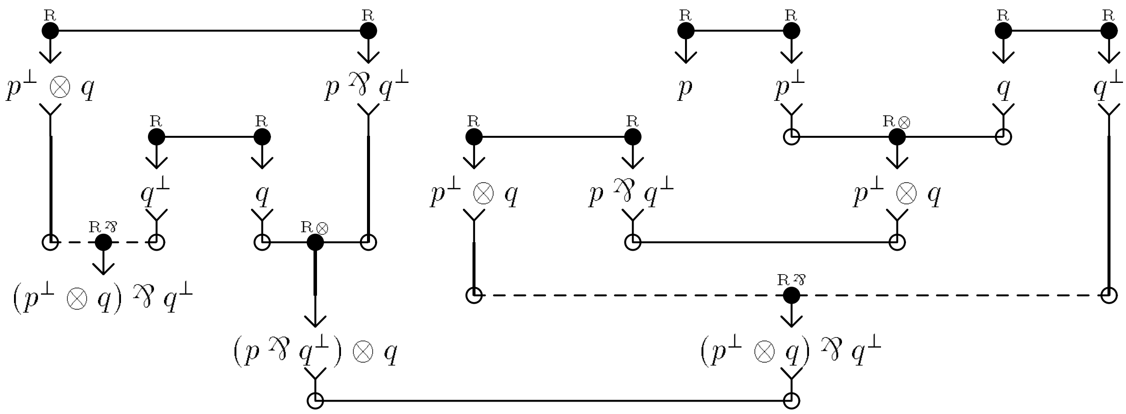
Definition 9.6 We translate each link L into a formula or into a link $L^\bullet$ of def. 2.1 in the following way:



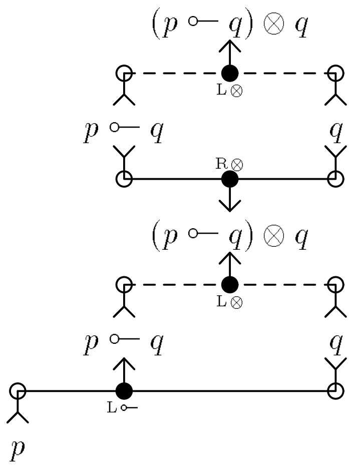
♦

Observe that this translation is well-defined on proof structures, since the main formula of $L^\bullet$ is really the image of the main formula of L. Moreover, it maps proof nets to proof nets, sinse the unary links are being contracted while the binary links remain of the same type.

Example 9.7 Let us consider the following **1MLL**-proof net of $\vdash p, (p^\perp \otimes q) \parr q^\perp$.



Under $(\cdot)^\bullet$ it translates into the depictured $\perp$-link-free **MLL**$^+$-proof net of $(p \multimapinv q) \otimes q \vdash p$. As a sequentialization we find e.g. the following derivation:



$$\dfrac{\dfrac{\dfrac{\dfrac{}{p \multimapinv q \vdash p \multimapinv q} \qquad \dfrac{}{q \vdash q}}{p \multimapinv q, q \vdash (p \multimapinv q) \otimes q}\,{\scriptstyle \mathrm{R}\otimes}}{(p \multimapinv q) \otimes q \vdash (p \multimapinv q) \otimes q}\,{\scriptstyle \mathrm{L}\otimes} \qquad \dfrac{\dfrac{\dfrac{}{q \vdash q} \qquad \dfrac{}{p \vdash p}}{p \multimapinv q, q \vdash p}\,{\scriptstyle \mathrm{L}\multimapinv}}{(p \multimapinv q) \otimes q \vdash p}\,{\scriptstyle \mathrm{L}\otimes}}{(p \multimapinv q) \otimes q \vdash p}\,{\scriptstyle \textsc{Cut}}$$

which may be transformed into a **1MLL**-derivation in the obvious way:

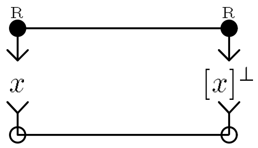
$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash p \parr q^\perp, p^\perp \otimes q} \qquad \dfrac{}{\vdash q, q^\perp}}{\vdash (p \parr q^\perp) \otimes q, p^\perp \otimes q, q^\perp}\,{\scriptstyle \mathrm{R}\otimes}}{\vdash (p \parr q^\perp) \otimes q, (p^\perp \otimes q) \parr q^\perp}\,{\scriptstyle \mathrm{R}\parr} \qquad \dfrac{\dfrac{\dfrac{}{\vdash q, q^\perp} \qquad \dfrac{}{\vdash p, p^\perp}}{\vdash p^\perp \otimes q, p, q^\perp}\,{\scriptstyle \mathrm{R}\otimes}}{\vdash (p^\perp \otimes q) \parr q^\perp, p}\,{\scriptstyle \mathrm{R}\parr}}{\vdash (p^\perp \otimes q) \parr q^\perp, p}\,{\scriptstyle \textsc{Cut}}$$

After some additions for each trivial Ax- or Cut-link, we find a derivation that is a sequentialization of the original **1MLL**-proof net.

$$
\begin{prooftree}
\AxiomC{}
\UnaryInfC{$\vdash p \parr q^{\perp}, p^{\perp} \otimes q$}
\AxiomC{}
\UnaryInfC{$\vdash q, q^{\perp}$}
\RightLabel{\scriptsize R$\otimes$}
\BinaryInfC{$\vdash (p \parr q^{\perp}) \otimes q, p^{\perp} \otimes q, q^{\perp}$}
\RightLabel{\scriptsize R$\parr$}
\UnaryInfC{$\vdash (p \parr q^{\perp}) \otimes q, (p^{\perp} \otimes q) \parr q^{\perp}$}
\AxiomC{}
\UnaryInfC{$\vdash p \parr q^{\perp}, p^{\perp} \otimes q$}
\AxiomC{}
\UnaryInfC{$\vdash q, q^{\perp}$}
\AxiomC{}
\UnaryInfC{$\vdash p, p^{\perp}$}
\RightLabel{\scriptsize R$\otimes$}
\BinaryInfC{$\vdash p^{\perp} \otimes q, p, q^{\perp}$}
\RightLabel{\scriptsize Cut}
\BinaryInfC{$\vdash p^{\perp} \otimes q, p, q^{\perp}$}
\RightLabel{\scriptsize R$\parr$}
\UnaryInfC{$\vdash (p^{\perp} \otimes q) \parr q^{\perp}, p$}
\RightLabel{\scriptsize Cut}
\BinaryInfC{$\vdash (p^{\perp} \otimes q) \parr q^{\perp}, p$}
\end{prooftree}
$$

♦

The two conclusions x and $[x]^{\perp}$ of an AX-link are translated into a sole formula $x^{\bullet}$, which surely will be an axiomatic formula in case neither x nor $[x]^{\perp}$ is the premise of a CUT-link. In the other case that at least one of the two formulas is the premise of a CUT-link (i.e. in case the concerning AX- and CUT-link are trivial), $x^{\bullet}$ can be everything (an axiomatic formula, a flow formula or a cut formula) or it can even disappear, as in



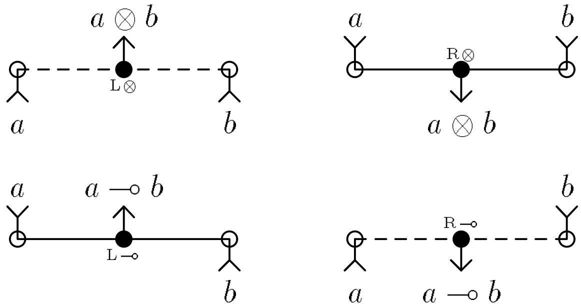
which translates into the empty proof structure. Similar we can see that a non-trivial CUT-link translates into a cut formula, while a trivial CUT-link (with at least one of the two premises being the conclusion of an AX-link) can be anything or disappear.

With this motivation the following is easily derived:

Corollary 9.8 (One-sided sequentialization) Every **1MLL**-proof net $\mathcal{P}$ is the proof net of a **1MLL**-derivation $\mathcal{D}$ (called a *sequentialization*), in which each rule corresponds with a link of the same type (AX, CUT, R$\otimes$ or R$\parr$). ♦

10 Intuitionistic MLL

We define the language of intuitionistic multiplicative linear logic (**iMLL**) to be the $\bot$- and $\parr$-free fragment of **MLL**. A proof structure of **iMLL** is a proof structure of **MLL** such that all its formulas belong to the language of **iMLL**. In particular it contains only four kinds of links, viz.:



We define a proof net of **iMLL** to be a proof structure of **iMLL** that, regarded as a proof structure of **MLL**, is a proof net of **MLL**. From lemma 3.10 we immediately derive:

Lemma 10.1 The following are derivation rules of **iMLL**.

Identity rules:

$$\dfrac{}{a \vdash a}\ \textsc{Ax}$$

$$\dfrac{X_1 \vdash a \qquad X_2, a \vdash y}{X_1, X_2 \vdash y}\ \textsc{Cut}$$

Multiplicative logical rules:

$$\dfrac{X_1 \vdash a \qquad X_2 \vdash b}{X_1, X_2 \vdash a \otimes b}\ \mathrm{R}\otimes \qquad \dfrac{X, a, b \vdash y}{X, a \otimes b \vdash y}\ \mathrm{L}\otimes$$

$$\dfrac{X_1 \vdash a \qquad X_2, b \vdash y}{X_1, X_2, a \multimap b \vdash y}\ \mathrm{L}\multimap \qquad \dfrac{X, a \vdash b}{X \vdash a \multimap b}\ \mathrm{R}\multimap$$

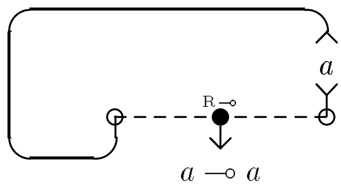
♦

From the rules in this lemma it is clear that any derivation of **MLL** in the language of **iMLL** is actually a derivation in the above rules; indeed, all rules preserve the property that the conclusion Q of a sequent $H \vdash Q$ is a multiset consisting of exactly one formula.

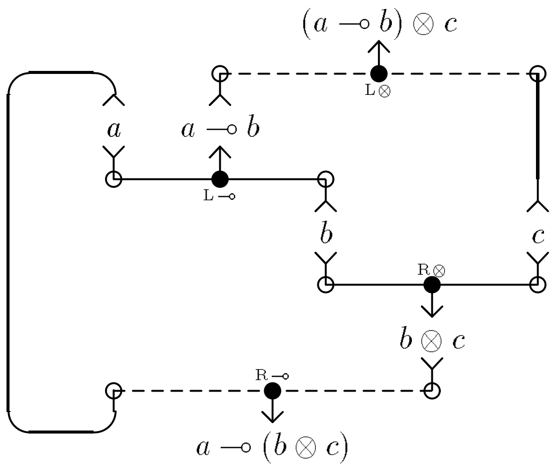
But then this scheme of rules is a complete class: given a proof net $\mathcal{P}$ of **iMLL**, an **MLL**-sequentialization of $\mathcal{P}$ will be a $\bot$- and $\parr$-free derivation $\mathcal{D}$, and hence a derivation in the rules of the above lemma.

We conclude that proof nets of **iMLL** have exactly one conclusion, and hence are nothing but deductions in the linearized version of the ($\wedge$,$\rightarrow$)-fragment of the system of Natural Deduction introduced by Gentzen [Ge35].

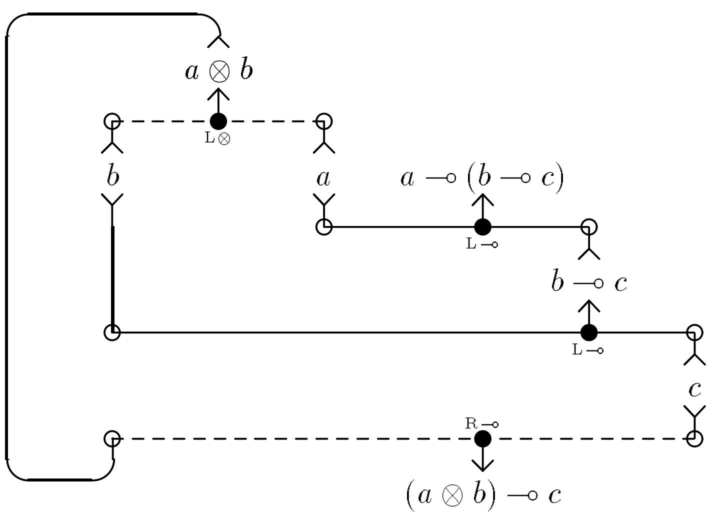
Example 10.2 (iMLL-Proof nets) 1.



2.



3.



♦

For multisets of syntactic **iMLL**-formulas we write

$$\begin{aligned}
X \dashv\vdash_{\mathrm{i}} Y \quad &\iff \quad X \vdash Y \text{ is } \mathbf{iMLL}\text{-provable} \quad \text{and} \quad Y \vdash X \text{ is } \mathbf{iMLL}\text{-provable} \\
&\iff \quad \text{there is a cut-free and } \eta\text{-expanded } \mathbf{iMLL}\text{-proof net } \mathcal{P}_1 \text{ of } X \vdash Y \\
&\qquad\quad \text{and a cut-free and } \eta\text{-expanded } \mathbf{iMLL}\text{-proof net } \mathcal{P}_2 \text{ of } Y \vdash X \\
X \dashv_{\mathrm{r}}\vdash_{\mathrm{i}} Y \quad &\iff \quad \text{there is an } \mathbf{iMLL}\text{-proof net } \mathcal{P} \text{ of } X \vdash Y \text{ such that} \\
&\qquad\quad \text{its rotation } \mathcal{P}^* \text{ is an } \mathbf{iMLL}\text{-proof net of } Y \vdash X \\
&\iff \quad \text{there is a cut-free and } \eta\text{-expanded } \mathbf{iMLL}\text{-proof net } \mathcal{P} \text{ of } X \vdash Y \text{ such that} \\
&\qquad\quad \text{its rotation } \mathcal{P}^* \text{ is a cut-free and } \eta\text{-expanded } \mathbf{iMLL}\text{-proof net of } Y \vdash X
\end{aligned}$$

and since cut-free and η-expanded **MLL**-proof nets of **iMLL**-sequents are automatically **iMLL**-proof nets, we see that $\dashv\vdash_{\mathrm{i}}$ and $\dashv_{\mathrm{r}}\vdash_{\mathrm{i}}$ are just the restrictions of $\dashv\vdash$ and $\dashv_{\mathrm{r}}\vdash$.

Example 10.3 By definition 7.5 we have the following chain of $\simeq$-equivalent **MLL**-formulas:

$$\begin{aligned}
a \multimap (b \multimap c) &\simeq (a)^{\perp} \parr (b \multimap c) \\
&\simeq (a)^{\perp} \parr ((b)^{\perp} \parr c) \\
&\simeq ((a)^{\perp} \parr (b)^{\perp}) \parr c \\
&\simeq (a \otimes b)^{\perp} \parr c \\
&\simeq (a \otimes b) \multimap c
\end{aligned}$$

and hence by lemma 7.6.2 we know that

$$a \multimap (b \multimap c) \dashv_{\mathrm{r}}\vdash (a \otimes b) \multimap c$$

Now both sides are **iMLL**-formulas, whence also

$$a \multimap (b \multimap c) \dashv_{\mathrm{r}}\vdash_{\mathrm{i}} (a \otimes b) \multimap c$$

Indeed one easily checks that the proof net of example 10.2.3 is rotatable. ♦

Definition 10.4 Let $\simeq_{\mathrm{i}}$ be the smallest equivalence relation on **iMLL**-formulas satisfying:

$$\begin{aligned}
a \simeq_{\mathrm{i}} a', \quad b \simeq_{\mathrm{i}} b' \quad &\Longrightarrow \quad a \otimes b \simeq_{\mathrm{i}} a' \otimes b' && (0\otimes) \\
a \simeq_{\mathrm{i}} a', \quad b \simeq_{\mathrm{i}} b' \quad &\Longrightarrow \quad a \multimap b \simeq_{\mathrm{i}} a' \multimap b' && (0\multimap) \\
&a \otimes (b \otimes c) \simeq_{\mathrm{i}} (a \otimes b) \otimes c && (5\otimes) \\
&a \multimap (b \multimap c) \simeq_{\mathrm{i}} (a \otimes b) \multimap c && (5\multimap) \\
&a \otimes b \simeq_{\mathrm{i}} b \otimes a && (6\otimes)
\end{aligned}$$

♦

Lemma 10.5 Let $a = a_n \multimap (a_{n-1} \multimap (\ldots (a_1 \multimap p)\ldots))$ and $b = b_n \multimap (b_{n-1} \multimap (\ldots (b_1 \multimap p)\ldots))$ be two **iMLL**-formulas where all a_i and b_i are $\multimap$-free (i.e. they are $\otimes$-only). Suppose $a_n \otimes a_{n-1} \otimes \ldots \otimes a_1$ and $b_n \otimes b_{n-1} \otimes \ldots \otimes b_1$ have the same multiset of atoms. Then $a \simeq_{\mathrm{i}} b$. ♦

Proof: By induction on n we first show that

$$a \simeq_{\mathrm{i}} (a_n \otimes (a_{n-1} \otimes (\ldots (a_2 \otimes a_1)\ldots))) \multimap p$$

For $n = 1$ this is true, since even equality holds. Suppose the result holds for $n = k$. Then for $n = k + 1$:

$$\begin{aligned} a &= a_{k+1} \multimap (a_k \multimap (a_{k-1} \multimap (\ldots (a_2 \multimap (a_1 \multimap p))\ldots))) \\ &= a_{k+1} \multimap [(a_k \otimes (a_{k-1} \otimes (\ldots (a_2 \otimes a_1)\ldots))) \multimap p] \qquad \text{(by the i.h. and } (0\multimap)) \\ &= [a_{k+1} \otimes (a_k \otimes (a_{k-1} \otimes (\ldots (a_2 \otimes a_1)\ldots)))] \multimap p \qquad \text{(by } (5\multimap)) \end{aligned}$$

which is the result for $n = k + 1$.

Now let a and b be given as described in the lemma, then

$$\begin{aligned} a &\simeq_{\mathrm{i}} (a_n \otimes (a_{n-1} \otimes (\ldots (a_2 \otimes a_1)\ldots))) \multimap p \\ b &\simeq_{\mathrm{i}} (b_n \otimes (b_{n-1} \otimes (\ldots (b_2 \otimes b_1)\ldots))) \multimap p \end{aligned}$$

As the proof of lemma 7.7 only refers to $(0\otimes)$, $(5\otimes)$ and $(6\otimes)$, we know that $a_n \otimes (a_{n-1} \otimes (\ldots (a_2 \otimes a_1)\ldots))$ and $b_n \otimes (b_{n-1} \otimes (\ldots (b_2 \otimes b_1)\ldots))$ (having the same multiset of atoms) are $\simeq_{\mathrm{i}}$-equivalent. Hence by $(0\multimap)$ also the above righthand sides are $\simeq_{\mathrm{i}}$-equivalent, yielding $a \simeq_{\mathrm{i}} b$. ■

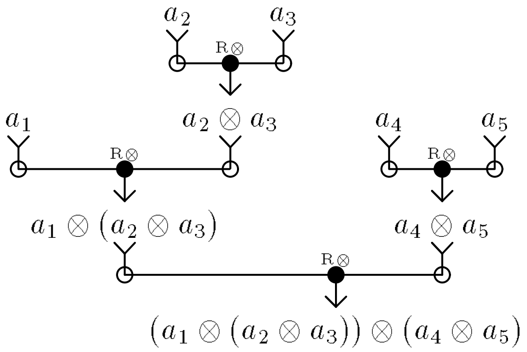
Theorem 10.6 For all syntactic **iMLL**-formulas a and b the following holds:

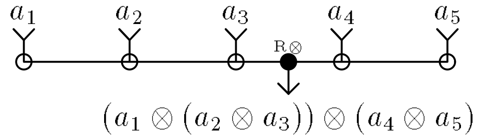
$$a \simeq_{\mathrm{i}} b \qquad \text{if and only if} \qquad a \dashv_{\mathrm{r}}\vdash_{\mathrm{i}} b$$

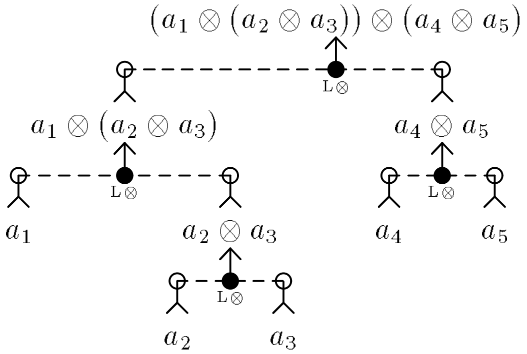
♦

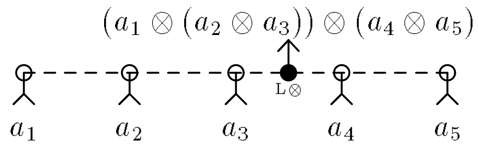
Proof: $\boxed{\Longrightarrow}$ As $\simeq$ is an equivalence relation satisfying $(0\otimes)$, $(0\multimap)$, $(5\otimes)$, $(6\otimes)$, and also $(5\multimap)$ (see ex. 10.3), its restriction $\simeq\restriction_{\mathrm{i}}$ is such an equivalence relation as well. On the other hand $\simeq_{\mathrm{i}}$ is by definition the smallest such an equivalence relation, whence $\simeq_{\mathrm{i}} \subseteq \simeq\restriction_{\mathrm{i}}$. So suppose $a \simeq_{\mathrm{i}} b$, then $a \simeq b$, and hence by lemma 7.6.2 $a \dashv_{\mathrm{r}}\vdash b$. But then also $a \dashv_{\mathrm{r}}\vdash_{\mathrm{i}} b$ as we argued before.

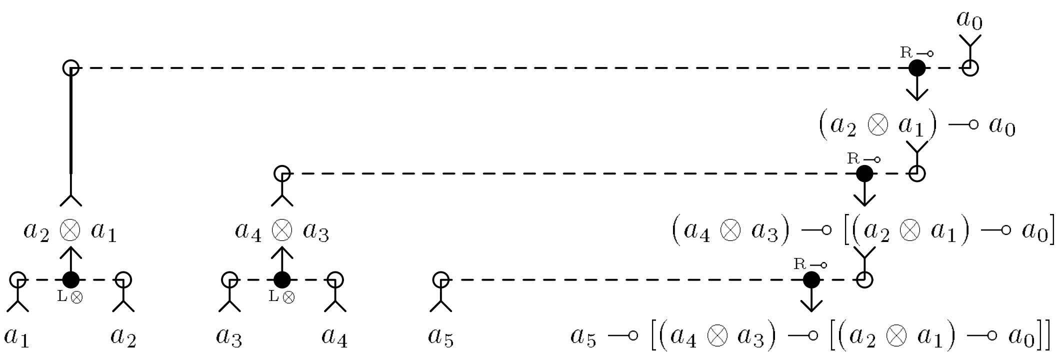
$\boxed{\Longleftarrow}$ This proof is completely similar to the proof of theorem 7.9.
Let $\mathcal{P}$ be a cut-free and η-expanded rotatable proof net of $a \vdash b$. Then we know $\mathcal{P}$ is the union of T^a and T_b containing only $\otimes$- and $\multimap$-links, followed by an identification of the atomic formulas, which is pairwise by lemma 7.8. The clusters are now of the form:

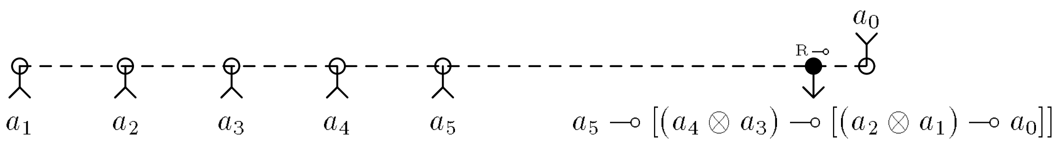

a_2 a_3
R$\otimes$
$a_2 \otimes a_3$
a_1 a_4 a_5
$a_1 \otimes (a_2 \otimes a_3)$
$a_4 \otimes a_5$
$(a_1 \otimes (a_2 \otimes a_3)) \otimes (a_4 \otimes a_5)$

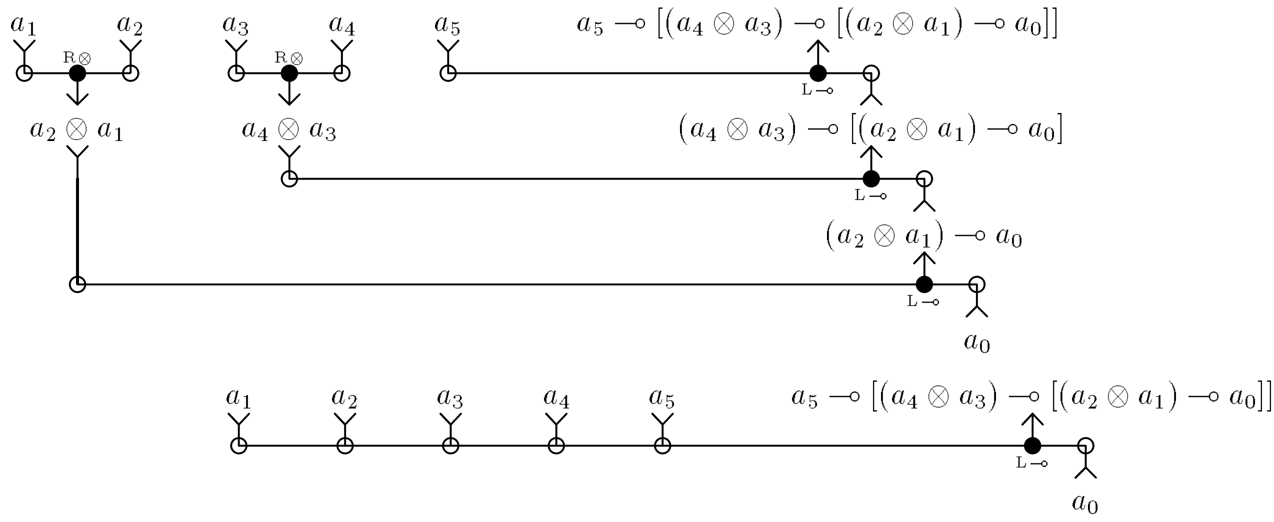

a_1 a_2 a_3 a_4 a_5
R$\otimes$
$(a_1 \otimes (a_2 \otimes a_3)) \otimes (a_4 \otimes a_5)$


$(a_1 \otimes (a_2 \otimes a_3)) \otimes (a_4 \otimes a_5)$
L$\otimes$
$a_1 \otimes (a_2 \otimes a_3)$
$a_4 \otimes a_5$
a_1 a_4 a_5
$a_2 \otimes a_3$
a_2 a_3


$(a_1 \otimes (a_2 \otimes a_3)) \otimes (a_4 \otimes a_5)$
L$\otimes$
a_1 a_2 a_3 a_4 a_5


a_0
R$\multimap$
$(a_2 \otimes a_1) \multimap a_0$
$(a_4 \otimes a_3) \multimap [(a_2 \otimes a_1) \multimap a_0]$
$a_2 \otimes a_1$
$a_4 \otimes a_3$
L$\otimes$
a_1 a_2 a_3 a_4 a_5
$a_5 \multimap [(a_4 \otimes a_3) \multimap [(a_2 \otimes a_1) \multimap a_0]]$


a_0
R$\multimap$
a_1 a_2 a_3 a_4 a_5
$a_5 \multimap [(a_4 \otimes a_3) \multimap [(a_2 \otimes a_1) \multimap a_0]]$



Again there is a cluster with only atomic active formulas, which we moreover may suppose to be thick. This cluster is hence a generalized R⊗-link or a generalized L⊸-link. It faces exactly one thin cluster (which is hence a generalized L⊗-link respectively a generalized R⊸-link), and the result follows by induction by means of lemma 7.7.2 respectively lemma 10.5. ■

The sequel of this section is devoted to the question which **MLL**-formulas are intuitionistic (⊥- and ⅋-free) modulo ≡-equivalence.

On page 48 we defined the normal form of an **MLL**-formula a as $\nu\pi a$. Here we will define the *binary normal form*[10] of an **MLL**-formula a as the **MLL**$^+$-formula $\chi\pi a$, where χ is defined by

$$\chi x := \begin{cases} x^\bullet & \text{if } \overline{x} = 0 \\ (x^\bullet)^\perp & \text{if } \overline{x} = 1 \end{cases}$$

(See def. 9.3 for the meaning of $x^\bullet$ and $\overline{x}$ for a **1MLL**-formula x.)

Lemma 10.7 Let a and b be two **MLL**$^+$-formulas.
1. $\pi\chi = \mathrm{id}_{\mathbf{1MLL}}$;
2. $\chi\pi a \equiv a$;
3. $a \equiv b$ if and only if $\chi\pi a = \chi\pi b$;
4. Suppose a is of de form b or $(b)^\perp$ where b is ⊥-free. Then $\chi\pi a = a$. ♦

Proof of 1: Let x be a **1MLL**-formula. If $\overline{x} = 0$ then by lemma 9.4 $\pi(\chi x) = \pi((x)^\bullet) = x$. If $\overline{x} = 1$ then $\pi(\chi x) = \pi((x^\bullet)^\perp) = [\pi(x^\bullet)]^\perp = [[x]^\perp]^\perp = x$.

[10]We call this the **binary** normal form since all occuring connectives are binary (eventually except for the outermost).

1MLL	
connectives	
0-ary	p_i, $p_i^\perp$
1-ary	
2-ary	$\otimes$, $⅋$
operations	
0-ary	
1-ary	$[\cdot]^\perp$
2-ary	$\multimap$, $\multimapinv$

$$\xrightarrow{\chi} \quad x \mapsto (x^\bullet)^{\overline{x}} \qquad \xleftarrow{\pi}$$

MLL$^+$	
connectives	
0-ary	p_i
1-ary	$(\cdot)^\perp$
2-ary	$\otimes$, $⅋$, $\multimap$, $\multimapinv$
operations	
0-ary	
1-ary	
2-ary	

iMLL	
connectives	
0-ary	p_i
1-ary	
2-ary	$\otimes$, $\multimap$
operations	
0-ary	
1-ary	
2-ary	

Figure 8: Definitions of syntactic formulas for **1MLL**, **MLL$^+$** and **iMLL**.

Proof of 2: From part 1 we see that $\pi(a) = \pi\chi(\pi a) = \pi(\chi\pi a)$. Now the result follows from lemma 7.3.

Proof of 3: Suppose that $a \equiv b$. From lemma 7.3 we see that then $\pi a = \pi b$, whence $\chi\pi a = \chi\pi b$. The other way around, suppose $\chi\pi a = \chi\pi b$, then by part 2 $a \equiv \chi\pi a = \chi\pi b \equiv b$, whence $a \equiv b$.

Proof of 4: Define $\psi(x) = (x^\bullet, \overline{x})$ as in def. 9.3. Now it is enough to show that $\psi(\pi b) = (b, 0)$ and $\psi(\pi(b)^\perp) = (b, 1)$. Indeed, then as a consequence, if $a = b$, then $\chi\pi a = (\pi a)^\bullet = b = a$, while if $a = (b)^\perp$, then $\chi\pi a = ((\pi a)^\bullet)^\perp = (b)^\perp = a$.
First we prove $\psi(\pi b) = (b, 0)$ by induction on b.
If $b = p_i$ then

$$\psi(\pi b) = \psi(\pi p_i) = \psi(p_i) = (p_i, 0) = (b, 0)$$

If $b = b' \mathbin{\Box} b''$, then the induction hypothesis reads $\psi(\pi b') = (b', 0)$ and $\psi(\pi b'') = (b'', 0)$. That is, $\overline{\pi b'} = 0 = \overline{\pi b''}$ and $(\pi b')^\bullet = b'$ and $(\pi b'')^\bullet = b''$. Hence (using lemma 9.4)

$$\begin{aligned}
\psi(\pi b) &= \psi(\pi(b' \mathbin{\Box} b'')) = \psi(\pi b' \mathbin{\Box} \pi b'') \\
&= \begin{cases}
\psi(\pi b' \otimes \pi b'') = ((\pi b')^\bullet \otimes (\pi b'')^\bullet, 0) = (b, 0) & \text{if } \Box = \otimes \\
\psi([\pi b']^\perp ⅋ \pi b'') = ((\pi b')^\bullet \multimap (\pi b'')^\bullet, 0) = (b, 0) & \text{if } \Box = \multimap \\
\psi(\pi b' ⅋ [\pi b'']^\perp) = ((\pi b')^\bullet \multimapinv (\pi b'')^\bullet, 0) = (b, 0) & \text{if } \Box = \multimapinv \\
\psi(\pi b' ⅋ \pi b'') = ((\pi b')^\bullet ⅋ (\pi b'')^\bullet, 0) = (b, 0) & \text{if } \Box = ⅋
\end{cases}
\end{aligned}$$

which finishes the proof of $\psi(\pi b) = (b, 0)$.
From this result and lemma 9.4 it follows that

$$\psi(\pi(b)^\perp) = \psi([\pi b]^\perp) = (([\pi b]^\perp)^\bullet, \overline{[\pi b]^\perp}) = ((\pi b)^\bullet, 1 - \overline{\pi b}) = (b, 1)$$

$\blacksquare$

Theorem 10.8 Let a be an **MLL**-formula. Then a is intuitionistic modulo $\equiv$-equivalence if and only if $\chi\pi a$ is intuitionistic. ♦

Proof: $\boxed{\Longrightarrow}$ Suppose $a \equiv b$ where b is $\perp$- and $⅋$-free. Then by lemma 10.7.3 $\chi\pi a = \chi\pi b$ and by lemma 10.7.4 $\chi\pi b = b$, whence $\chi\pi a = b$, i.e. $\chi\pi a$ is intuitionistic.

$\boxed{\Longleftarrow}$ Suppose $\chi\pi a$ is intuitionistic, then by lemma 10.7.2 we see that $a \equiv \chi\pi a$, i.e. a is intuitionistic modulo $\equiv$-equivalence. ■

The next theorem shows that the definition of the map ψ in def. 9.3 is the only possible one for our purposes (viz., to get rid of $\perp$ in the inner part of a formula). Example 9.5 serves also as an example of the next theorem.

Theorem 10.9 Let x be a **1MLL**-formula. Then there is a unique $\perp$-free **MLL**$^+$-formula b such that x equals πb ór $[\pi b]^\perp$. ♦

Proof: Take $b := x^\bullet$. Then, depending on $\overline{x}$, we obtain by lemma 9.4.1

$$\pi b = \pi x^\bullet = x \qquad (\overline{x} = 0)$$

or

$$\pi b = \pi x^\bullet = [x]^\perp \text{ implying } [\pi b]^\perp = x \qquad (\overline{x} = 1)$$

Now suppose also b' does the job. Then there are three possibilities: $\pi b = \pi b'$; $\pi b = [\pi b']^\perp$ or $[\pi b]^\perp = [\pi b']^\perp$.
If $\pi b = \pi b'$ then $\chi\pi b = \chi\pi b'$. Now b and b' are both $\perp$-free, so lemma 10.7.4 yields $b = \chi\pi b = \chi\pi b' = b'$, whence $b = b'$.
If $\pi b = [\pi b']^\perp = \pi((b')^\perp)$, the same argument yields $b = (b')^\perp$, which is in contradiction with the fact that b is $\perp$-free.
If $[\pi b]^\perp = [\pi b']^\perp$, then also $\pi b = \pi b'$, which is treated already and yields $b = b'$.
We conclude that there exists exactly one b which does the job, and moreover in exactly one way. ■

References

[BW95] G. Bellin and J. van de Wiele. Subnets of proof nets in $\boldsymbol{MLL^-}$. In J.-Y. Girard, Y. Lafont and L. Regnier editors, *Advances in Linear Logic,* pages 249–270. Cambridge University Press. London Mathematical Society Lecture Note Series 222, 1995.

[Da88] V. Danos. Correctness of proof nets. *Draft, université de Paris VII,* March 1988.

[DR89] V. Danos and L. Regnier. The structure of multiplicatives. *Archive for Mathematical Logic,* 28:181–203, 1989.

[Ge35] G. Gentzen. Untersuchungen über das logische Schließen. *Mathematische Zeitschrift,* 39:176–210, 405–431, 1935.

[Gi87] J.-Y. Girard. Linear logic. *Theoretical Computer Science,* 50:1–102, 1987.

[Me94] F. Métayer. Homology of proof nets. *Archive for Mathematical Logic,* 33:169–188, 1994.

[Tr92] A.S. Troelstra. *Lectures on Linear Logic.* CSLI Lecture Notes 29, Center for the Study of Language and Information, Stanford, California, 1992.