Under consideration for publication in Math. Struct. in Comp. Science

Proof Nets for Multiplicative Cyclic Linear Logic and Lambek Calculus

V. Michele Abrusci and Roberto Maieli[†]

Department of Mathematics and Physics – "Roma Tre" University – Rome (Italy)

email: {abrusci, maieli}@uniroma3.it

Received 1 September 2017

Abstract – This paper presents a simple and intuitive syntax for proof nets of the multiplicative cyclic fragment (McyLL) of linear logic (LL). The main technical achievement of this work is to propose a correctness criterion that allows for sequentialization (recovering a proof from a proof net) for all McyLL proof nets, including those containing cut links. This is achieved by adapting the idea of contractibility (originally introduced by Danos to give a quadratic time procedure for proof nets correctness) to cyclic linear logic. This paper also gives a characterization of McyLL proof nets for Lambek Calculus and thus a geometrical (i.e., non inductive) way to parse phrases or sentences by means of Lambek proof nets.

Keywords – categorial grammars, cyclic orders, Lambek calculus, language parsing, linear logic, non-commutative logic, proof nets, sequent calculus.

1. Introduction

Proof nets (PNs) are one of the most innovative inventions of linear logic (LL, [Girard 1987]): they are used to represent demonstrations in a geometric (i.e., non inductive) manner, abstracting away from the technical bureaucracy of sequential proofs. Proof nets quotient classes of derivations that are equivalent up to some irrelevant permutations of inference rule instances. Following this spirit, we present a simple syntax for proof nets of the multiplicative cyclic fragment of LL (shortly, McyLL). In particular, we introduce a new correctness criterion for McyLL proof nets which can be considered as the non-commutative counterpart of the famous contraction criterion by Vincent Danos [Danos 1990] for proof nets of linear logic. The proposed syntax (i.e., the correctness criterion) is shown to be stable under (i.e., preserved by) cut elimination.

This work marks an important improvement compared to previous works on the same subject by the authors (see e.g., [Abrusci and Ruet 2000, Maieli 2003]). The proposed

[†] This work contains new original contributions and improvements w.r.t. the contents of a previous paper [Abrusci and Maieli 2015a] on the same subject presented by the authors at the *22nd Workshop on Logic, Language, Information and Computation (WoLLIC2015)*, held at the Indiana University (Bloomington, USA) from the 20th to the 23rd of July 2015.

new syntax admits a sequentialization (that is, a way to associate a sequent proof to each proof net) for the full class of McyLL PNs including those ones with cuts.

1.1. *The multiplicative cyclic fragment of linear logic (McyLL)*

We briefly recall the necessary background of the McyLL fragment without units. We arbitrarily assume literals $a, a^\perp, b, b^\perp, ...$ with a polarity: *positive* $(+)$ for atoms, $a, b, ...$ and *negative* $(-)$ $a^\perp, b^\perp...$ for their duals. A *formula* is built from literals by means of two groups of *multiplicative connectives*: *negative*, $\triangledown$ ("par") and *positive*, $\otimes$ ("tensor"). For these connectives we have the following De Morgan laws: $(A \otimes B)^\perp = B^\perp \triangledown A^\perp$ and $(A \triangledown B)^\perp = B^\perp \otimes A^\perp$. A McyLL proof is any derivation tree built by the following inference rules where sequents Γ, Δ are lists of formulas occurrences endowed with a *total cyclic order* (or *cyclic permutation*) (see the formal Definition 1):

$$\frac{}{\vdash A, A^\perp}\,id \qquad \frac{\vdash \Gamma, A \qquad A^\perp \Delta}{\vdash \Gamma, \Delta}\,cut \qquad \frac{\vdash \Gamma, A \qquad \vdash B, \Delta}{\vdash \Gamma, A \otimes B, \Delta}\,\otimes \qquad \frac{\vdash \Gamma, A, B}{\vdash \Gamma, A \triangledown B}\,\triangledown$$

It is worth nothing that the formula $(A \otimes B) \multimap (B \otimes A) \equiv (A \otimes B)^\perp \triangledown (B \otimes A)$ is not provable in McyLL: that is the reason why this logic is called "non commutative". Negative (or *asynchronous*) connectives correspond to a kind of *true determinism* in the way we apply bottom-up their corresponding inference rules (the application of $\triangledown$-rule is completely deterministic). Vice-versa, positive (or *synchronous*) connectives correspond to a kind of *true non-determinism* in the way we apply bottom-up their corresponding rules (there is no deterministic way to split the context Γ, Δ in the $\otimes$-rule).

A *total cyclic order* can be thought of as follows; consider a set of points of an oriented circle; the orientation induces a total order on these points as follows: if a, b and c are three distinct points, then b is either between a and c $(a < b < c)$ or between c and a $(c < b < a)$. Moreover, $a < b < c$ is equivalent to $b < c < a$ or $c < a < b$.

Definition 1 (total cyclic order). A *total cyclic order* is a pair (X, σ) where X is a set and σ is a ternary relation over X satisfying the following properties:

1. $\forall a, b, c \in X, \sigma(a, b, c) \to \sigma(b, c, a)$ *(cyclic)*,
2. $\forall a, b \in X, \neg\sigma(a, a, b)$ *(anti-reflexive)*,
3. $\forall a, b, c, d \in X, \sigma(a, b, c) \wedge \sigma(c, d, a) \to \sigma(b, c, d)$ *(transitive)*,
4. $\forall a, b, c \in X, \sigma(a, b, c) \vee \sigma(c, b, a)$ *(total)*.

In the following we adopt the syntax $\sigma(X)$ to denote a total cyclic order on a set X.

1.2. *The quest of satisfactory syntaxes for McyLL proof nets*

The most simple and intuitive definition of McyLL PNs is given by [Moot and Retoré 2012]: "proof nets for the cyclic fragment of MLL are intuitively quite simple graphs (special kinds of MLL proof nets) which can be drawn on a plane without intersecting axioms and keeping the same design and top-down orientation for links. This condition is strictly stronger than being simply planar graphs because we ask for the links to be drawn observing the left-right and up-down orientation as shown in the figures".

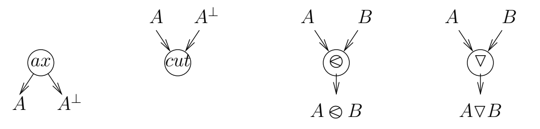
While there is a variety of nice (satisfying, indeed) syntaxes and correctness criteria for the commutative proof nets of MLL, this is not the case with proof nets of the non-commutative, cyclic, fragment of MLL. Actually, despite of the commutative MLL case, the presence of cut links is "quite tricky" in the non-commutative case, since cut links are not equivalent, from a topological point of view, to tensor links: these latter make appear new conclusions that may disrupt the original (i.e., in presence of cut links) conclusions cyclic order.

But, what is supposed to be in general a "satisfactory" correctness criterion for proof nets? There is not a so obvious answer. Let us say that a "good" correctness criterion should at least meet the following conditions, according to [Moot and Retoré 2012]:

1 (*de-sequentialization*) every sequent proof should be mapped to a correct proof structure in such a way that each instance of inference rule corresponds to a link; in particular, each proof with cuts (resp. cut-free) should be mapped to a proof structure with the corresponding cut links (resp., to a cut-free proof structure);
2 (*sequentialization*) every correct proof net should correspond to a sequent proof in such a way that each link corresponds to an instance of inference rule; in particular, each proof net with cuts (resp. cut-free) should be mapped to a sequent proof with the corresponding cut rule instances (resp., to a cut-free sequent proof);
3 (*canonicity*) sequent proofs which only differ up to permutations of some inference rule instances should be mapped to the same proof net;
4 (*stability*) the correctness criterion should be preserved by cut elimination.

There currently exist several syntaxes for McyLL proof nets, like notably those ones of [Abrusci and Ruet 2000], [Maieli 2003] and [Melliès 2004]. As far as the authors know, only the Melliès'syntax fully satisfies the four conditions above while both Abrusci-Ruet and Maieli's syntaxes only fully satisfy conditions 1 and 4 while they only partially satisfy conditions 2 and 3 (only when proof nets are cut-free). We find interesting to recall and compare in the following the two previous syntaxes, [Abrusci and Ruet 2000] and [Maieli 2003], given by the authors, highlighting the points where they mostly differ.

Definition 2 (concrete proof structure (PS)). A *(concrete) proof-structure* (shortly, PS) of McyLL is an oriented graph π, in which edges are labeled by formulas and nodes are labeled by connectives of McyLL , built by juxtaposing the following special graphs, called *links*, in which incident (resp., emergent) edges are called *premises* (resp., *conclusions*):



In a PS π each premise (resp., conclusion) of a link must be conclusion (resp., premise) of exactly (resp., at most) one link of π. We call *conclusion of* π any emergent edge that is not premises of any link.

We are interested in those McyLL PSs that correspond to McyLL proofs.

1.2.1. *"Trip-based" criterion by Abrusci and Ruet* – We recall some basic definitions of Abrusci-Ruet's syntax [Abrusci and Ruet 2000]. We consider as in [Girard 1987] formulas with decorations: $\uparrow$ (question) or $\downarrow$ (answer). A decorated formula is of the form $A^\uparrow$ or $A^\downarrow$, where A is a McyLL formula. For each link l we consider two sets of decorated formulas:

— l^{in} is the set of all decorated formulas A^x, where A is a premisse of l and x is $\downarrow$, or A is a conclusion of l and x is $\uparrow$;

— l^{out} is the set of all A^x, where A is a premisse of l and x is $\uparrow$, or A is a conclusion of l and x is $\downarrow$.

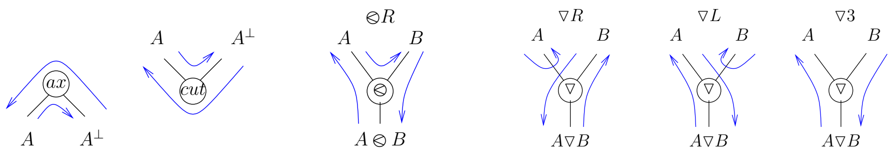
Definition 3 (switchings). For each link l we define a set $S(l)$ of (partial) functions from l^{in} to l^{out}, called *switching positions* of l, as follows (see also next picture):

— if l is an identity (or axiom) link $\overline{A^\perp \quad A}$, then $S(l) = \{id\}$ where $id : (A^\perp)^\uparrow \mapsto A^\downarrow, A^\uparrow \mapsto (A^\perp)^\downarrow$;

— if l is a cut link $\underline{A^\perp \quad A}$, then $S(l) = \{cut\}$ where $cut : (A^\perp)^\downarrow \mapsto A^\uparrow, A^\downarrow \mapsto (A^\perp)^\uparrow$;

— if l is a $\otimes$-link $\frac{A \quad B}{A \otimes B}$, then $S(l) = \{\otimes R\}$ where $\otimes R : (A \otimes B)^\uparrow \mapsto A^\uparrow, A^\downarrow \mapsto B^\uparrow, B^\downarrow \mapsto (A \otimes B)^\downarrow$;

— if l is a $\triangledown$-link $\frac{A \quad B}{A \triangledown B}$, then $S(l) = \{\triangledown R, \triangledown L, \triangledown 3\}$ where
$\triangledown R : (A \triangledown B)^\uparrow \mapsto B^\uparrow, A^\downarrow \mapsto A^\uparrow, B^\downarrow \mapsto (A \triangledown B)^\downarrow$,
$\triangledown L : (A \triangledown B)^\uparrow \mapsto A^\uparrow, A^\downarrow \mapsto (A \triangledown B)^\downarrow, B^\downarrow \mapsto B^\uparrow$,
$\triangledown 3 : (A \triangledown B)^\uparrow \mapsto A^\uparrow, B^\downarrow \mapsto (A \triangledown B)^\downarrow$.



A *switching* for a proof structure π is a function s s.t. for every link l of π, $s(l) \in S(l)$. Given a proof structure π and a switching s for π, the *switched proof structure* $s(\pi)$ is the oriented graph with the decorated formulas labeling π as vertices and with an oriented edge from A^x to B^y iff either $B^y = s(l)(A^x)$ for some link $l \in \pi$ or $A^x = C^\downarrow$ and $B^y = C^\uparrow$ for some conclusion C of π. Then we call *trip* any cycle or maximal path in $s(\pi)$. A cycle v in $s(\pi)$ is *bilateral* if v is not of the form $A^x, ..., B^y, ..., A^{\bar{x}}, ..., B^{\bar{y}}, ..., A^x$ where A and B are occurrences of formulas in π and $\bar{\uparrow} = \downarrow$ (resp., $\bar{\downarrow} = \uparrow$).

Definition 4 (Abrusci-Ruet's criterion). A PS π is *AR-correct* (i.e., it is a *McyLL proof net* by [Abrusci and Ruet 2000]) iff for every switching s for (π):

1 there exists exactly one cycle σ in $s(\pi)$,
2 σ contains all the conclusions of π,
3 σ is bilateral.

1.2.2. *"Seaweed-based" criterion by Maieli* – We recall some basic definitions of Maieli's syntax for proof nets as presented in [Maieli 2003].

Definition 5 (switchings & seaweeds). Assume a McyLL PS π with conclusions Γ.

— A *Danos-Regnier switching* (see [Danos and Regnier 1989]) S for π, denoted $S(\pi)$, is

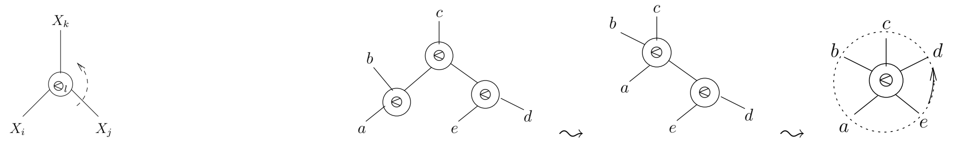


Fig. 1. seaweeds and total cyclic orders

the non oriented graph built on nodes and edges of π with the modification that for each ∇-node we take only one premise, that is called *left* or *right* ∇*-switch.*

— Let $S(\pi)$ be an acyclic and connected switching for π; $S(\pi)$ is the rootless planar tree[†] whose nodes are labeled by $\otimes$-nodes, and whose leaves $X_1, ..., X_n$ (with $\Gamma \subseteq X_1, ..., X_n$) are the terminal, i.e., pending, edges of $S(\pi)$; then $S(\pi)$ is a ternary relation, called *seaweed*, with *support* $X_1, ..., X_n$; we say that an ordered triple (X_i, X_j, X_k) belongs to the seaweed $S(\pi)$ iff:

  – the intersection of the three paths X_iX_j, X_jX_k and X_kX_i is a node $\otimes_l$;
  – the three paths $X_i\otimes_l, X_j\otimes_l$ and $X_k\otimes_l$ are in this cyclic order while moving anticlockwise around the $\otimes_l$-nod, like in the leftmost hand side picture of Figure 1:

  If A is an edge of the seaweed $S(\pi)$, then $S_i(\pi)\downarrow^A$ is the *restriction of the seaweed* $S(\pi)$, that is, the sub-graph of $S(\pi)$ obtained as follows:

  1 disconnecting the graph *below* (w.r.t. the orientation of π) the edge A;
  2 deleting the graph not containing A.

  The restriction of a seaweed can easily be extended to consider a set of formulas.

Fact 1 (seaweeds as cyclic orders). Any seaweed $S(\pi)$ can be viewed as a cyclic total order (Definition 1) on its support $X_1, ..., X_n$; in other words, if a triple $(X_i, X_j, X_k) \in S(\pi)$, then X_i, X_j, X_k are in this cyclic order $X_i < X_j < X_k$.

Naively, we may *contract* ("$\rightsquigarrow$") any seaweed (by associating the $\otimes$-nodes while preserving the order of the incident edges) until we get a (collapsed) single n-ary $\otimes$-node with n pending edges (its support), like the rightmost h.s. seaweed of Figure 1.

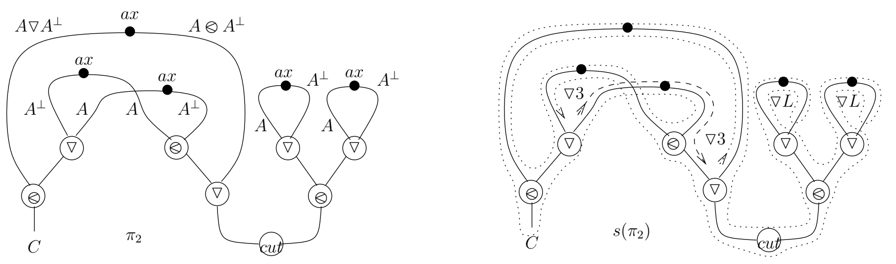
Definition 6 (Maieli's criterion). A PS π is *M-correct* (i.e., it is a *McyLL proof net* by [Maieli 2003]) iff:

1 π is a standard MLL PN, that is, by [Danos and Regnier 1989], any switching $S(\pi)$ is a connected and acyclic graph (therefore, $S(\pi)$ is a seaweed);
2 for any ∇-link $\frac{A \quad B}{A\nabla B}$ the triple (A, B, C) must occur with this cyclic order $A < B < C$ in any seaweed $S(\pi)$ restricted to A, B (i.e., $(A, B, C) \in S(\pi)\downarrow^{(A,B)}$) for every conclusion C (if any) in the support of $S(\pi)\downarrow^{(A,B)}$.

[†] In any switching we can consider as a simple edge every axiom, cut and ∇-link that remains after the mutilation of one of the two premises.

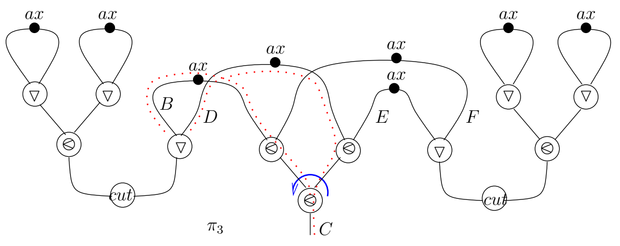
1.2.3. *Comparing the two previous syntaxes* – Abrusci-Ruet's criterion and Maieli's criterion are not equivalent; moreover, even though they are both stable under cut reduction, they suffer of the same drawback: they do not allow a direct sequentialization of proof nets with cuts; this means that proof nets must be normalized (reduced in cut-free normal form) before sequentialialization, as shown by the following proof structures:

π_1 is not correct according to Abrusci-Ruet; you can find a switching, like the $s(\pi_1)$ on the right hand side picture, containing only two paths, the dotted and the dashed one: neither of them is a cycle.



On the contrary, π_1 is correct according to Maieli even though it is not directly sequentializable: this proof structure contains only two terminal links, the leftmost $\otimes$-link and the (unique) cut link; neither of these links is "splitting" (sequentializable); as soon as you "remove" one of them, you get a sub-proof structure that is not correct. By the way, π_1 can be easily sequentialized once it has been reduced to cut-free form.

π_2 is not correct according to Maieli (assume $B = D = E = F = (A^\perp \otimes A)$): actually,



you can easily find a seaweed $S(\pi_2)\downarrow^{B,D}$, restricted to the premises of the $\frac{B \quad D}{B \triangledown D}$-link, containing a "bad" triple (B, C, D) (follow the anticlockwise intersection of the dotted lines above); on the contrary, π_2 is correct according to Abrusci-Ruet even though it is not directly sequentializable (it can only be sequentialized after cut elimination).

Anyway, as we will see in Section 2, these two proof structures, π_1 and π_2, are not recognized as correct by our new correctness criterion[‡] (see Definition 9). Following the original Danos terminology, [Danos 1990], these proof structures are not *contractible* (i.e., they do not reduce to a collapsed or elementary graph, made by a single node). The new contraction system Σ, for cyclic (abstract) proof structures, is quite simple and

‡ Indeed π_1 and π_2 are neither correct according to Melliès' criterion. Comparison with Melliès's condition is out of the scope of this work as it needs the introduction of some topological notions requiring more space than the one allowed here. By the way, we briefly recall the criterion in the Appendix A.1 and we postpone to a future work the precise correspondence between topological condition and retraction.

natural; it mainly differs from the original one Δ, for MLL (abstract) proof structures, by the following facts: *(i)* abstract proof structures are enriched with special handling nodes, denoted by "$\circ$", to distinguish the conclusions of a proof structure; *(ii)* for each node, the set of its incident edges is endowed with a total cyclic order (following the anticlockwise orientation); *(iii)* contraction steps are performed obeying some "natural" order constraints on the edges occurring in the redex graph. The stability of this new syntax under cut elimination is then proved in Section 2.1 while the direct sequentialization[§] of the full class of correct proof nets is shown in Sections 2.2.

1.3. *Lambek Calculus and Proof Nets as Parsing Structures*

McyLL can be considered as a classical extension of Lambek Calculus (LC, see [Lambek 1958], [Abrusci 2002] and [Moot and Retoré 2012]) one of the ancestors of LL. The LC represents the first attempt of the so called *parsing as deduction*, i.e., parsing of natural language by means of a logical system. Following [Andreoli and Pareschi 1991], in LC parsing is interpreted as type checking in the form of theorem proving of Gentzen sequents. Types (i.e. propositional formulas) are associated to words in the lexicon; when a string $w_1...w_n$ is tested as a sentence, the types $t_1, ..., t_n$ associated with the words are retrieved from the lexicon and then parsing reduces to proving the derivability of a one-sided sequent of the form $\vdash t_n^\perp, ..., t_1^\perp, s$, where s is the type associated with sentences. Moreover, forcing constraints on the Exchange rule by allowing only *cyclic permutations* over sequents of formulas, gives the required computational control needed to view theorem proving as parsing in Lambek Categorial Grammar style. Anyway, LC parsing presents some syntactical ambiguity problems; actually, there may be:

1 *non canonical proofs*, i.e., more than one cut-free proof for the same sequent;

2 *lexical polymorphism*, i.e., more than one type associated with a single word.

Now, multiplicative proof nets are commonly considered an elegant solution to the first problem of representing canonical proofs, since they allow to quotient classes of (cut-free) proofs that are equivalent up to irrelevant permutation of inference rules; in this sense, in Section 3.1, we also give an embedding of pure Lambek Calculus into McyLL proof nets. In Section 3.2, we show how McyLL proof structures can be used to parse phrases or sentences; some linguistic examples that can be also found in [Moot 2002].

Unfortunately, there is no an equally brilliant solution to the polymorphism problem mentioned above. However, we think that extending parsing by means of additive proof nets (MALL, [Girard 1996], [Hughes and van Glabbeek 2003] and [Maieli 2007]) could be a step towards a proof-theoretical solution to the problem of lexical polymorphism; technically speaking, the cyclic fragment of MALL proof nets allows to manage formulas (types) superposition (polymorphism) by means of the additive connectives & and $\oplus$ (see Appendix A.4).

[§] In Appendix A.2 we also discuss an alternative sequentialization procedure based on the parsing of abstract paired graphs labeled by possibly "open" sequent proofs.

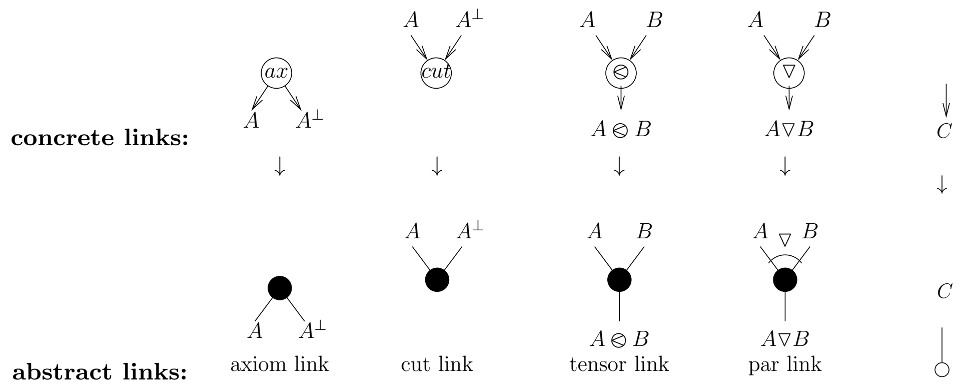


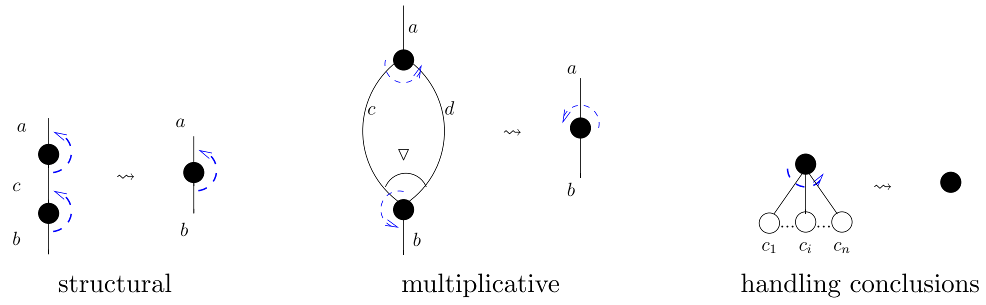
Fig. 2. transforming concrete proof structures into abstract proof structures

2. McyLL proof nets

Definition 7 (abstract structure (AS)). An *abstract structure* (shortly, *AS*) is a non oriented graph $\pi = \langle V, E\rangle$, equipped with a set $\mathcal{C}(\pi)$ of *pairs* of coincident edges graphically denoted by a crossing *arc* (with, possibly, "$\triangledown$" written above) close to the base. Edges are labeled by McyLL formulas. Nodes are displayed as bullets ($\bullet$) except the handling ones (conclusions) displayed as circles ($\circ$); all edges incident to a node are endowed with an total cyclic order displayed as an anticlockwise oriented dotted arrow around a node. An *abstract link* is any elementary AS made by a single node together with its incident (possibly paired) edges (premises). The *size* of an AS π (resp., of a PS) is given by the triple $\langle \sharp V, \sharp E, \mathcal{C}(\pi)\rangle$ (resp., by the pair $\langle \sharp V, \sharp E\rangle$).

We call *abstract proof structure (APS)* the AS π^{ab} obtained from a concrete PS π (Definition 2) by means of the *abstraction rules* of Figure 2: each rule maps a concrete link L of a PS π to an *abstract link* L^{ab} ($L \mapsto L^{ab}$) of π^{ab}; moreover, every edge of π labeled by a conclusion C is mapped in π^{ab} to a special handling node, denoted by a circle $\circ$, with a single incident edge labeled by C, called *conclusion abstract link.*

Definition 8 (Retraction System Σ for ASs). Given an AS π, a *retraction step*is a replacement (also, *deformation* or *rewriting*) of a subgraph S (called, *redex graph*) of π with a new graph S' (called, *reductum graph*), leading to an AS π' according to one of the following rules (of the *retraction system* Σ), preserving the anticlockwise orientation:



1 R_1 (*structural*) with the conditions that the edge c does not occur in any pair in π and the two displayed vertexes in the redex are distinct;

2 R_2 (*multiplicative*) with the conditions that the two vertexes in the redex are distinct and edges c and d does not occur in any pair except that one displayed in the redex; moreover, c must occur in this anticlockwise cyclic order, $c < d < a$, together with every (if any) edge a (with $a \neq c, d$) incident to the vertex opposite to the base of the pair c, d (as displayed in the figure);

3 R_3 (*handling conclusions*) with the conditions that every incident edges to the vertex $\bullet$ in the redex belongs to a handling conclusion link (at least one, i.e., $n \geq 1$).

Definition 9 (Σ-correctness by retraction (ΣCC)). Assume an AS π, then π is said:

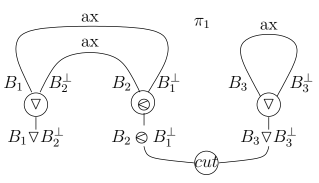
— *one step contractible*, if there exists an AS $\pi' \neq \pi$ s.t. $\pi \rightsquigarrow \pi'$ by an instance of one of the retraction rules of Σ, otherwise, π is said *terminal*;

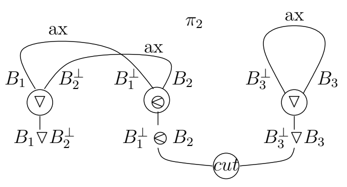
— *collapsed* (or *elementary*), if it consists of a single node $\bullet$ (with no incident edges);

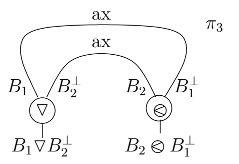
— *(full) contractible*, if there exists a non empty sequence of retraction steps starting at π and terminating with a *collapsed AS* (i.e., $\pi \rightsquigarrow^* \bullet$); we also say that π is *quasi-collapsed* when it collapses only by means of finite sequences of structural or conclusions retraction rules, R_1 and R_3 (in other words, no multiplicative instances).

A PS π is Σ*-correct (ΣCC)* so, it is a *proof net*, when its corresponding APS, π^{ab}, is contractible (i.e., it collapses). Equivalently, the end result of the retraction of a proof net (i.e., the last collapsed reductum) could be simply be one black node with any number of white nodes, instead of a single black node; this means that the final retraction rule R_3 could be considered redundant.

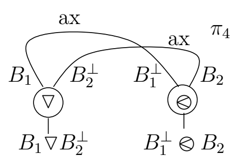
In the following we give below some instances, π_1, π_2 and π_3, of Σ-correct proof structures together with an instance π_4 of an incorrect proof structure.







Observe that the reason why π_2 is correct, but π_4 is not, it is because the sub-proof π_2 is cut against contracts and disappears, so that the edges of the tensor node can be rotated to disentangle the crossed axioms.



The retraction system Σ is a (non-commutative) refinement of the original Danos's retraction system Δ [Danos 1990]; this latter contains only two retraction rules: the structural R_1 and the multiplicative R_2, with the proviso that:

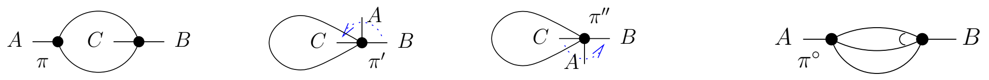
— abstract structures contain only standard nodes of type $\bullet$ (no handling nodes $\circ$);

— the anticlockwise order condition on the incident edges has been relaxed (both in the redex and in the reductum).

Theorem 1 (convergence of Σ). If π is a contractible AS then every retraction strategy ends up with the collapsed AS ($\bullet$).

Proof. The convergence (termination and confluence) of Σ is proven exactly like in in the standard commutative case [Danos 1990]. Observe that, like in the MLL case, since π is contractible then there are no pairs of instances of contraction rules that are *critical*[¶]. □

Observe that, in general, Σ is not confluent if we consider the full class of ASs including the non contractible ones. Consider the following AS π on the leftmost hand side: π may retract, by an instance of structural rules, either to π' or to π''; none of these latter is (one step) retractible; moreover, given the different cyclic orders of resp, their incident edges, π' and π'' diverge. Remind that retraction is neither confluent in the general case of MLL non retractible ASs as illustrated by the rightmost hand side AS π°.



Next fact is immediate once: *(i)* every handling node $\circ$ is replaced by a standard node $\bullet$ and *(ii)* any instance of conclusions contraction rule R_3 is mapped in to a possibly multiple instances of the structural rule R_1.

Fact 2 (Σ-contraction (ΣCC) $\Rightarrow$ Δ-contraction (ΔCC)). If π collapses by Σ then it also collapses by Δ; so we say that π is also *weakly* (also, Δ or Danos) *contractible.*

Fact 3 (switching and seaweeds for APSs). The notions of switching and seaweed (resp., restriction of a seaweed) of Definition 5 straightforwardly extended to the image APS π^{ab} of any proof net π (since the incident edges of each vertex in π^{ab} are naturally equipped with an anticlockwise strict cyclic order and each pair in π^{ab} can be switched by mutilating one of its paired edges).

Definition 10 (cyclic order conclusions). We can derive the order on the conclusions of a proof net from its structure. Assume π is a McyLL proof net with conclusions Γ; we call *order of conclusions* of π the cyclic order σ on Γ (denoted by $\sigma(\Gamma)$) induced by an arbitrary seaweed[‖] $S(\pi)$ restricted do Γ (i.e., $S(\pi)\downarrow^{\Gamma}$).

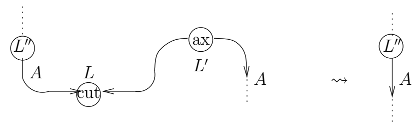
2.1. *Cut Reduction*

Definition 11 (cut reduction). Let L be a cut link in a proof net π whose premises A and $A^{\perp}$ are, resp., conclusions of links L', L''. Then we define the result π' (called *reductum*) of reducing this cut in π (called *redex*), as follows:

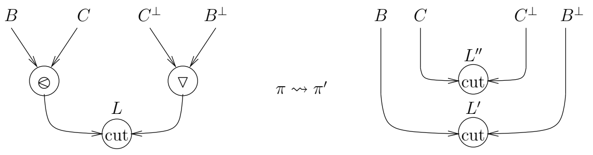
[¶] A pair of retraction rule instances, R_i and R_j, is called *critical pair* when the application of R_i prevents the application of R_j and the application of R_j prevents the application of R_i.

[‖] We will see that, by Lemma 6, in a proof net π with conclusion Γ, the order of conclusion $\sigma(\Gamma)$ as defined in Definition 10 is independent indeed from the choice of the seaweed $S(\pi)$.

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

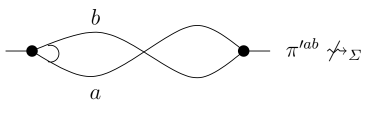


($\otimes/\triangledown$)-cut: if L' is a $\otimes$-link with premises B and C and L'' is a $\triangledown$-link with premises $C^\perp$ and $B^\perp$, then π' is obtained by removing in π the formulas A and $A^\perp$ as well as the cut link L with L' and L'' and by adding two new cut links with, resp., premises B, $B^\perp$ and $C, C^\perp$, as follows (this is also called *logical cut*):

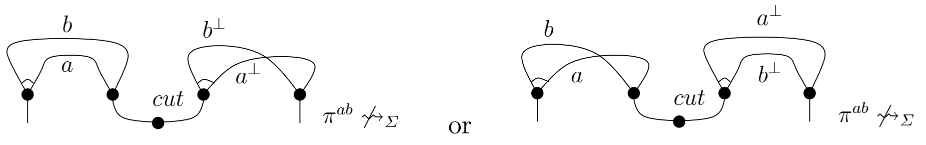


Theorem 2 (stability of correctness under cut reduction). If π is a proof net that reduces to π' by one step of cut reduction, $\pi \mapsto \pi'$, then π' still a proof net.

Proof. The case when π reduces to π' by an instance of axiom-cut reduction is trivial since an axiom-cut reduction step corresponds to an instance of structural rule R_1, $\pi^{ab} \rightsquigarrow_{R_1} \pi'^{ab}$. So assume that in π there are only logical cuts. π^{ab} is Σ-contractible and so, by Fact 2, π is also Δ-contractible, therefore by stability of Δ [Danos 1990], also the reductum π'^{ab} is Δ-contractible. Assume by absurdum that π^{ab} is Σ-contractible while π'^{ab} is only weakly contractible (π'^{ab} is not Σ-contractible). By convergence of Δ, π'^{ab} can be contracted following a strategy s that delays all the structural instances of R_1 contracting handling conclusion nodes $\circ$ (trivially, in Δ any instance of structural rule R_1 that contracts a handling conclusion node $\circ$ does not prevent the application of any other contraction rule, so this latter can be performed before the former one). Now, observe that each retraction step of this "delayed" strategy s can be mimicked (i.e., it can be performed as well) in Σ except in the case when this step consists of an instance of multiplicative retraction R_2 whose redex does not meet the (anticlockwise) cyclic order condition on the incident edges, as follows:



This means that, if we restore in π'^{ab} the just reduced logical cut then, by eventually exploiting the convergence of Σ (i.e., by permuting some instances of retraction rule, if necessary, by Theorem of 1), π^{ab} will contain one of the two sub structures; this means that in both cases π^{ab} is not Σ-contractible, contradicting the assumption:



b
a
cut
$b^\perp$
$a^\perp$
$\pi^{ab} \not\rightsquigarrow_\Sigma$
or
b
a
cut
$a^\perp$
$b^\perp$
$\pi^{ab} \not\rightsquigarrow_\Sigma$

□

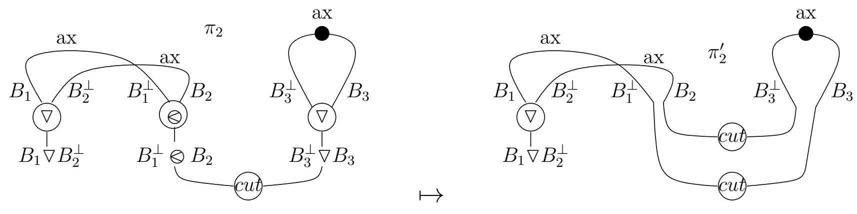
Lemma 1 (stability of order conclusions under cut reduction). If π is a proof net, with conclusions $\sigma(\Gamma)$, that reduces in one step of cut reduction to π', then π' has conclusions $\sigma(\Gamma)$ too.

Proof. See [Maieli 2003]. □

Theorem 3 (convergence of cut reduction). Cut reduction is *convergent* (i.e., terminating and confluent).

Proof. Easy, almost like in the standard MLL case [Danos 1990]. □

Example 1. Below, the proof net π_2 reduces to a proof structure π'_2 that is correct: we use indexed formulas B_1, B_2 and B_3 to distinguish several occurrences of B.

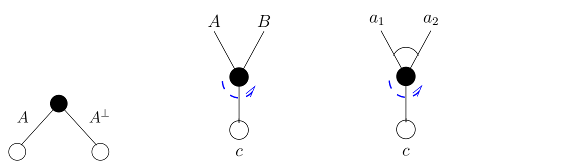


2.2. *Sequentialization* via *Splitting Lemma*

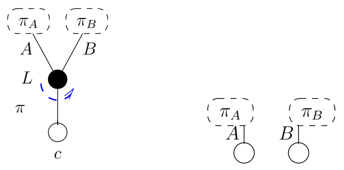
In this section we show a correspondence, also known in the literature as *sequentialization*, between McyLL proof nets and sequential proofs. A first sequentialization result, only for cut free non-commutative (McyLL) proof nets, can be found in [Retoré 1996]. Similarly to the standard (commutative) MLL case, the crucial point of the sequentialization procedure is given by the *Splitting Lemma 5.* Actually, as observed in [Bagnol *et al.* 2015] in the MLL case, there exists an alternative way (straightforward, indeed, that skips the Splitting Lemma) to sequentialize McyLL proof nets, based on the following "contraction as parsing" strategy: if an APS is contractible then, by convergence of Σ, there exists a "contraction strategy" which starts by contracting the axioms links and whose retraction steps can be interpreted as instances of inference rules of a (possibly open) sequential proof; in the case of success, the retraction sequence ends up with a collapsed graph (as usual, a •-node) labeled by a closed sequent proof (see details in the appendix A.2).

In order to simplify the syntax and when it is clear from the context, we sometimes denote simply π (resp., L) instead of π^{ab} (resp., L^{ab}) the APS (resp., the abstract link) corresponding to a concrete PS π (resp., to a concrete link L). Moreover, since we are going to reason on abstract structures that are immediate abstraction of concrete proof structures, it will be natural to use notions like "terminal links", "splitting links" and "cut-reduction steps" directly defined on these APSs.

Definition 12 (terminal and splitting link). An abstract tensor (resp., axiom or par) link L is *terminal* when its conclusion edge also belongs to a handling conclusion link $\circ$ (see next picture). Given a contractible APS π, a terminal tensor link L is said *splitting link* (resp., *weakly splitting*) when *(i)* we can delete its handling conclusion link labeled by C *(ii)* disconnect the premises, A and B of L, and *(iii)* get still two contractible (resp., weakly contractible) APSs, π_A and π_B, as follows (similarly, we define a *splitting cut link*):



terminal axiom, tensor and par (or pair) links



tensor splitting link

We say that π is *splitting* (resp., *weakly splitting*) when it contains at least a terminal tensor link or a cut link that is splitting (resp. weakly splitting). Finally, we say that π is in *splitting condition* when *(i)* it is not reduced to an axiom link, *(ii)* it does not contain any terminal par link and *(ii)* it contains at least a terminal tensor link or a cut-link.

Lemma 2 (APS splitting under weakly contraction). Let π^{ab} be the abstraction of a concrete PS π; assume π^{ab} is Σ-contractible and in splitting condition, then there exists at least one terminal tensor link (or a cut link) L^{ab} that is *weakly splitting.*

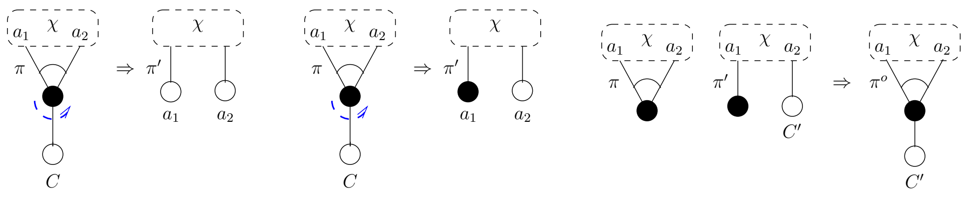
Proof. By Fact 2, π^{ab} is Δ-contractible; so, by the well known equivalence "Danos's MLL correctness $\Leftrightarrow$ Girard's MLL correctness" (see [Danos 1990]) π is also a standard MLL proof net by Girard, so by the MLL Splitting Lemma of [Girard 1987], we conclude that π is weakly splitting at a terminal tensor link (or cut link) L in two components, π_1 and π_2 whose abstractions, π_1^{ab} resp., π_2^{ab}, are clearly weakly contractible. $\square$

Lemma 3 (splitting tensor link). If π is a contractible APS that is weakly splitting at a tensor terminal link $L\frac{A \quad B}{A\otimes B}$, then π is also splitting at L, i.e. removing L splits π in two Σ-contractible APSs, π_A and π_B.

Proof. Immediate. $\square$

Proposition 1 (terminal pair link). Let π be a Σ-contractible AS:

1 assume π contains a terminal pair link L then π', obtained by replacing L with two handling $\circ$-links, a_1 and a_2, like in Case 1 below, is still contractible;
2 assume π contains a terminal pair link L then π' obtained by replacing L with an elementary $\bullet$-link a_1 and a handling $\circ$-link a_2, like in Case 2 below, is still contractible;
3 assume π contains a link L with an incident pair and assume a Σ-contractible AS π' obtained by replacing, in π, L with an elementary $\bullet$-link and a handling link C', like in Case 3; then π^o obtained by adding the handling link C' to π is still contractible.

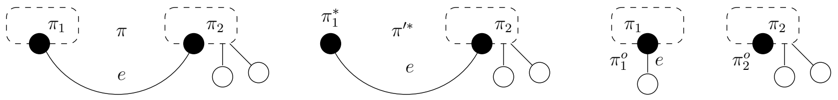


(Case 1) (Case 2) (Case 3)

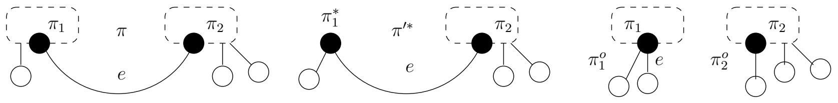
Proof. By induction of the size of π. □

Proposition 2 (handling links). Assume π is a Σ-contractible AS consisting of two disjoint components, π_1 and π_2, which are only connected through an edge e; then:

1 assume π_1 does not contain any conclusion and π contracts by Σ to π'^* like in the middle figure below, then π_1^o and $\pi_2^\bullet$ (on the rightmost hand side) are both Σ-contractible:



2 assume $\pi_1$contains at least a conclusion of π and assume π Σ-contracts to π'^* like in the middle figure below, then π_1^o and π_2^o (on the rightmost h. s.) are both Σ-contractible.



Proof. By induction on the size of π_1. □

Proposition 3 (weakly splitting cut link). Let π be a Σ-contractible APS that is only weakly splitting and let L be cut link $\frac{A \quad A^\perp}{}$ weakly splitting, like in Figure 3, then:

1 π is Σ-contractible by a sequence of retraction steps that contracts $\pi_{A^\perp}$ (resp., π_A) to a (quasi-)collapsed APS before starting with contracting π_A (resp., $\pi_{A^\perp}$); moreover,

2 either $\pi^o_{A^\perp}$ is a Σ-contractible APS not containing any conclusion of π, where $\pi^o_{A^\perp}$ is obtained by adding to $\pi_{A^\perp}$ the handling $\circ$-link, labeled by $A^\perp$, like in Figure 3, and $\pi^o_{A^\perp}$ has $A^\perp$ as single conclusion;

3 or π^o_A is a Σ-contractible APS not containing any conclusion of π, where π^o_A is obtained by adding to π_A the handling $\circ$-link, labeled by A, like in Figure 3, and π^o_A has A as single conclusion.

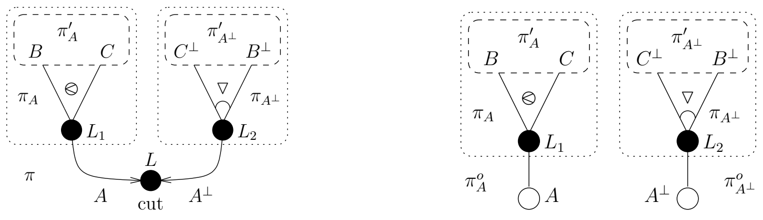
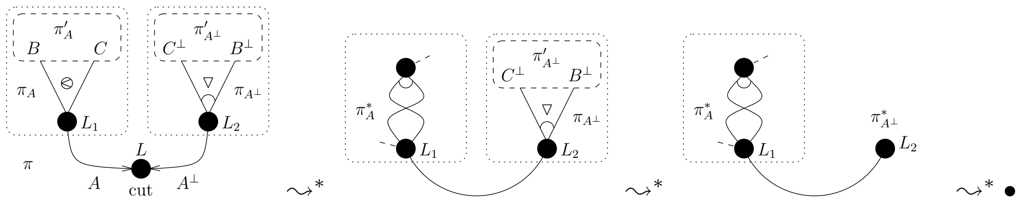


Fig. 3. (weakly) splitting cut link

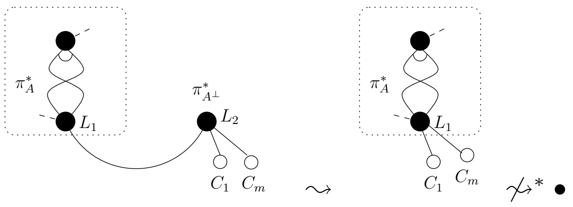
Proof. Since π is contractible and weakly splitting, it is always possible, by convergence of Σ (Theorem 1) to commute the order of the retraction steps in such a way that those ones involving (i.e., whose redex belongs to) π_A (resp., $\pi_{A^\perp}$) are performed before (i.e., independently from) those ones involving $\pi_{A^\perp}$ (resp., π_A); this strategy can be pursued until the retraction has turned π_A in to an AS π^*_A that is (there are two cases for π^*_A):

1 either in quasi-collapsed form (see Definition 9),
2 or in non retractible form (different from the quasi-collapsed one).

In case 1 we trivially proved statement 1. Moreover, observe that in this case, the quasi-collapsed π^*_A cannot contain any handling conclusion of π, otherwise by Proposition 2(2), both π^o_A and $\pi^o_{A^\perp}$ will be Σ-contractible, contradicting the assumption that π is only weakly splitting at L. Thus, by Proposition 2(1), π^o_A with A as single conclusion is contractible (this proves statement 3). Case 2 (i.e.π^*_A is non contractible but not quasi-collapsed) implies that π is retractible by means of a retraction sequence that starts by contracting $\pi_{A^\perp}$ to a quasi-collapsed $\pi^*_{A^\perp}$, before starting with contracting π_A, like below:



Now observe that $\pi^*_{A^\perp}$ cannot contain any handling conclusion link of of π, otherwise, since π^*_A is in a non retractible terminal form (it is not quasi-collapsed), we will conclude that π is not contractible, like illustrated below (a contradiction).



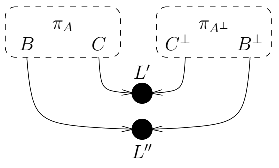
Thus we proved statements 1 and 2 (this latter, by Proposition 2(1)). The remaining case, when π contracts by a sequence starting with contracting $\pi_{A^\perp}$ to a (quasi-)collapsed $\pi^*_{A^\perp}$ before contracting π_A, is symmetric to the previous one and so omitted. □

Lemma 4 (splitting of Σ-contractible APSs). Let π be a Σ-contractible APS in splitting condition; assume π is only weakly splitting (i.e., π only contains terminal tensor links or cut links that are only weakly splitting), then there exists a splitting tensor link or a splitting cut link L.

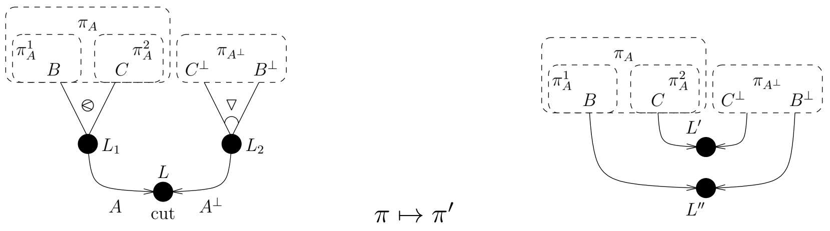
Proof. Assume, by absurdum, there exists such an APS π that it is only weakly splitting. By Lemma 3, π cannot contain any terminal tensor link that is only weakly splitting. Therefore, let π be a minimal APS (w.r.t. the size) containing only weakly splitting cut links and let L be a weakly splitting cut link like in Figure 3.

By Proposition 3, either $\pi^o_{A^\perp}$ (case 2) or π^o_A (case 3) must be Σ-contractible with a single conclusion $A^\perp$, resp., A. Assume $\pi^o_{A^\perp}$ is Σ contractible (case 2), therefore π^o_A must be only weakly retractible. We reason on link L_1 of π_A.

1 If L_1 is not (weakly) splitting then, we can reduce the cut link L of π and get a proof net π' (by Theorem 2) with two reduced cut links, L' and L'': since neither of these new cuts is weakly splitting and since all the other cut links (as well as the terminal tensor links) of π' remain not splitting, we found an APS π', strictly smaller than π, that is not splitting, contracting the assumption of minimality of π.

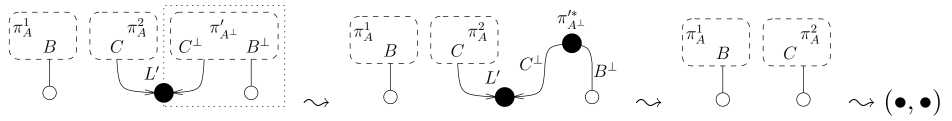


2 Otherwise, assume L_1 is weakly splitting, like in the next left hand side picture; then, as before, we can reduce the cut link L and get (by Theorem 2) an APS π' with two reduced cut links, L' and L'' like in the r.h.s. picture below:



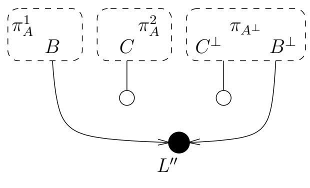
Now observe that, in the reductum π' neither L' nor L'' is splitting; otherwise:

(a) assume that L'' is splitting, then after splitting π' we get two separated components like in the leftmost hand side case below:



now, by assumption $\pi^o_{A^\perp}$ is contractible, then by Proposition 1 (Case 2), also $\pi'_{A^\perp}$ (enclosed in the dotted line above) will be so; this means that, both π^1_A and π^2_A are contractible (with abuse of notation "$(\bullet, \bullet)$" in the figure above), and so also π^o_A (Figure 3) is contractible contradicting the assumption L is only weakly splitting;

(b) assume that L' is splitting, then splitting π' produces two separated components like below; so, we get a contradiction by a similar argument to the case (2a).



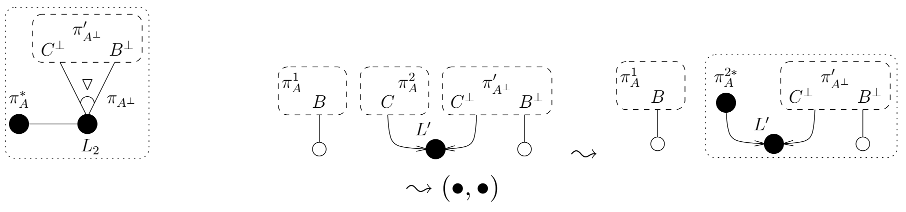
Now observe that, in π':

(i) none terminal tensor link is (weakly) splitting; otherwise, assume by absurdum there exists a terminal tensor link L_s that is weakly splitting in π'; now, since cut-reduction did not introduce new ones, all terminal tensor links in π' were already occurring in π; moreover, by Proposition 3(2), $\pi^o_{A^\perp}$ does not contain any terminal link, except L_2, thus L_s must occur in π_A; this means (simply, by reasoning on MLL APSs) that L_s was already weakly splitting in π, contradicting the assumption that (by minimality) π was not containing weakly splitting terminal tensor links;

(ii) none cut link is splitting; in particular, by 2a and 2b, L' and L'' are both not splitting; the remaining cut links in π' are not splitting for similar reasons seen in the previous case (i).

Therefore, we have found a non splitting π', strictly smaller than π, contradicting the assumption that π was a minimal non splitting APS.
In the other case, when by Proposition 3(3), π^o_A is Σ contractible with A as single conclusion, we proceed like before except for the fact that, in order to get a contradiction in the analogous of the sub-case 2a, we reason as follows:

— since by assumption π and π^o_A are both contractible, we deduce that the AS (enclosed in a dotted line) on left hand side of next picture is Σ-contractible, where π^*_A denotes the collapsed form of π_A; obviously, if π^o_A is contractible then also π_A (without its its single handling node) is contractible (it contracts exactly to π^*_A);
— since by assumption L'' is splitting, then after splitting π' we get two separated Σ-contractible components, like those ones in the middle side below;
— this means that, by Case 3 of Proposition 1 (applied to the ASs below enclosed in dotted rectangle frames, after a structural step), $\pi^o_{A^\perp}$ is contractible, contradicting the assumption the L in π is not splitting.



□

Lemma 5 (McyLL proof net splitting). Let π be a McyLL PN in splitting condition (π is not reduced to an axiom link, it contains at least a $\otimes$-link (resp., a *cut*-link) and it does not contain any ∇-conclusion) then, there must exist a $\otimes$-link $\frac{A \quad B}{A \otimes B}$ (resp., a *cut*-link $\frac{A \quad A^\perp}{}$) that splits π in two McyLL PNs, π_A and π_B (resp., π_A and $\pi_{A^\perp}$).

Proof. If π is a McyLL proof net in splitting condition then, by Lemma 4, its abstraction π^{ab} is splitting. Assume π^{ab} splits at the abstract terminal tensor link $L^{ab} : \frac{A \quad B}{A \otimes B}$ in two contractible π^{ab}_A and π^{ab}_B; clearly removing L in π produces two correct proof nets π_A and π_B. The case when π^{ab} is splitting at a cut link is similar. □

Lemma 6 (conclusions order of a proof net). Let π be a McyLL PN with conclusions Γ, then all seaweeds $S_i(\pi) \downarrow^\Gamma$, restricted to Γ, induce the same cyclic order σ on Γ ($\sigma(\Gamma)$).

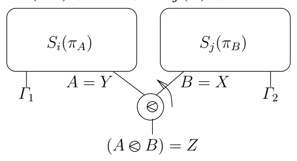
Proof. By induction on the size of π. If π is reduced to an axiom link, then obvious, otherwise we reason as follows.

1 If π contains at least a conclusion $A\nabla B$, then $\Gamma = \Gamma', A\nabla B$; by hypothesis of induction the sub-proof net π' with conclusion Γ', A, B has cyclic order $\sigma(\Gamma', A, B)$, and so, by condition 2 of Definition 6 applied to π, we know that each restricted seaweed $S_i(\pi)\downarrow^{(\Gamma',A,B)}$ induces the same cyclic order $\sigma(\Gamma', A, B)$; finally, by substituting $[A/A\nabla B]$ (resp., $[B/A\nabla B]$) in the restriction $S_i(\pi)\downarrow^{(\Gamma',A)}$ (resp., $S_i(\pi)\downarrow^{(\Gamma',B)}$), we get that each seaweed $S_i(\pi)\downarrow^{(\Gamma',A\nabla B)}$ induces the same cyclic order $\sigma(\Gamma', A\nabla B)$.

2 Otherwise π must contain a terminal splitting $\olessthan$-link or *cut*-link. Assume π contains a splitting $\olessthan$-link, $\frac{A \quad B}{A\olessthan B}$, and assume by absurdum that π is such a minimal (w.r.t. the size) PN with at least two seaweeds $S_i(\pi)$ and $S_j(\pi)$ s.t. $(X, Y, Z) \in S_i(\pi)$ and $(X, Y, Z) \notin S_j(\pi)$. We follow two sub-cases.

(a) It cannot be the case $X = B, Y = A$ and $Z = C$ otherwise, by definition of seaweeds, $S_i(\pi)$ and $S_j(\pi)$ will appear as follows:

$$S_i(\pi)\downarrow^{(\Gamma_1,A\olessthan B,\Gamma_2)}= S_i(\pi_A)\downarrow^{(\Gamma_1,A)} \olessthan S_i(\pi_B)\downarrow^{(B,\Gamma_2)}, \quad S_j(\pi)\downarrow^{(\Gamma_1,A\olessthan B,\Gamma_2)}= S_j(\pi_A)\downarrow^{(\Gamma_1,A)} \olessthan S_j(\pi_B)\downarrow^{(B,\Gamma_2)}$$

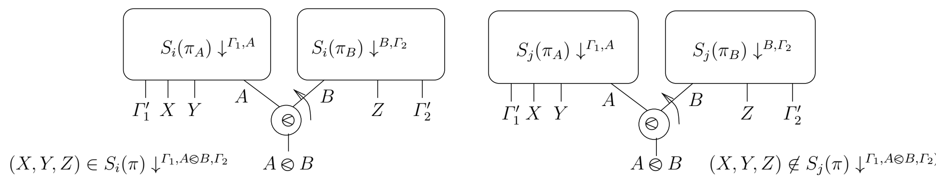


Now, by hypothesis of induction, all seaweeds on π_A (resp., all seaweeds on π_B) induce the same order on Γ_1, A (resp., Γ_2, B), then in particular,

$$S_i(\pi_A)\downarrow^{(\Gamma_1,A)}= S_j(\pi_A)\downarrow^{(\Gamma_1,A)} \quad \text{and} \quad S_i(\pi_B)\downarrow^{(B,\Gamma_2)}= S_j(\pi_B)\downarrow^{(B,\Gamma_2)}$$

but this implies $S_i(\pi)\downarrow^{(\Gamma_1,A\olessthan B,\Gamma_2)}= S_j(\pi)\downarrow^{(\Gamma_1,A\olessthan B,\Gamma_2)}$.

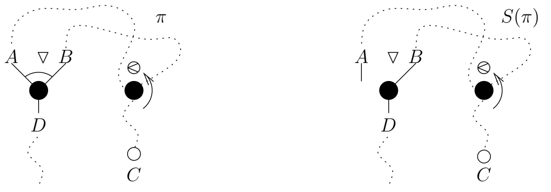
(b) Assume both X and Y belong to π_A (resp., π_B) and Z belongs to π_B (resp., π_A); moreover, assume for some i, j, $(X, Y, Z) \in S_i(\pi)\downarrow^{(\Gamma_1,A\olessthan B,\Gamma_2)}$ and $(X, Y, Z) \notin S_j(\pi)\downarrow^{(\Gamma_1,A\olessthan B,\Gamma_2)}$; by Splitting Lemma 5, each seaweeds for π, $S_i(\pi)$ and $S_j(\pi)$, must appear as follows:


$S_i(\pi_A)\downarrow^{\Gamma_1,A}$
$S_i(\pi_B)\downarrow^{B,\Gamma_2}$
Γ_1' X Y
A
B
Z Γ_2'
$\olessthan$
$(X,Y,Z) \in S_i(\pi)\downarrow^{\Gamma_1,A\olessthan B,\Gamma_2}$
$A \olessthan B$
$S_j(\pi_A)\downarrow^{\Gamma_1,A}$
$S_j(\pi_B)\downarrow^{B,\Gamma_2}$
Γ_1' X Y
A
B
Z Γ_2'
$\olessthan$
$A \olessthan B$
$(X,Y,Z) \notin S_j(\pi)\downarrow^{\Gamma_1,A\olessthan B,\Gamma_2)}$

so, by restriction, $(X, Y, A) \in S_i(\pi_A)\downarrow^{\Gamma_1,A}$ and $(X, Y, A) \notin S_j(\pi_A)\downarrow^{\Gamma_1,A}$, contradicting the assumption (by minimality) that π_A is a correct PN with a cyclic order on its conclusions $\Gamma_1', X, Y, A = \Gamma_1, A$.

The remaining case, when π contains a splitting cut, is similar and so omitted. $\square$

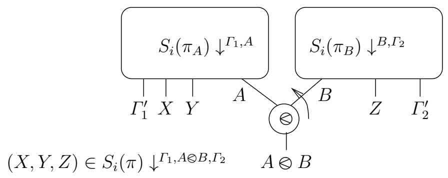
Lemma 7 (contractible pair). If π is a contractible APS, then there does not exist a pair (A, B) (an abstract link $A\nabla B$) in π and a seaweed $S(\pi)$ s.t. the three paths AC, BC and AB intersect in a $\bullet$-node with the anticlockwise order, for a conclusion C, as follows:



Proof. By induction of the size of π. □

Fact 4 (seaweed splitting). Assume π is a proof net in splitting condition with order conclusions $\sigma(\Gamma) = \Gamma_1 < A \otimes B < \Gamma_2$; assume π splits at a tensor (resp., cut) link $L : \frac{A \quad B}{A \otimes B}$ (resp., $L : \frac{A \quad A^\perp}{}$), then, after splitting, the proof net π_A has order conclusions $\sigma(\Gamma) \downarrow^{\Gamma_1, A} = \Gamma_1 < A$ and the proof net π_B (resp., $\pi_{A^\perp}$) has order conclusions $\sigma(\Gamma) \downarrow^{\Gamma_2, B} = \Gamma_2 < B$ (resp., $\sigma(\Gamma) \downarrow^{\Gamma_2, A^\perp} = \Gamma_2 < A^\perp$).

Proof. By Definition 10 of order conclusions $\sigma(\Gamma)$ of a proof net and by Lemma 6, each seaweed $S_i(\pi) \downarrow^{\Gamma}$ is given by the composition $S_i(\pi) \downarrow^{\Gamma_1} < A \otimes B < S_i(\pi) \downarrow^{\Gamma_2}$ like below:



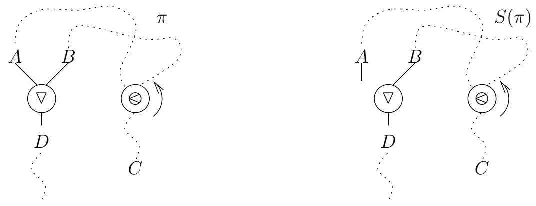
$S_i(\pi_A) \downarrow^{\Gamma_1, A}$
$S_i(\pi_B) \downarrow^{B, \Gamma_2}$
Γ_1'
X
Y
A
B
Z
Γ_2'
$\otimes$
$(X, Y, Z) \in S_i(\pi) \downarrow^{\Gamma_1, A \otimes B, \Gamma_2}$
$A \otimes B$

then, by restriction and substitution, each seaweed for π_A (resp., for π_B) will be given by $S_i(\pi_A) \downarrow^{\Gamma_1, A}$ (resp., $S_i(\pi_B) \downarrow^{\Gamma_2, B}$), that is, the order conclusions of π_A (resp., π_B) wil be $\sigma \downarrow^{\Gamma_1, A} = \Gamma_1 < A$ (resp., $\sigma \downarrow^{\Gamma_2, B} = \Gamma_2 < B$). □

Theorem 4 ((de-)sequentialization). If π is a McyLL proof net with conclusion $\sigma(\Gamma)$ then it sequentializes into a sequent proof with same conclusion $\sigma(\Gamma)$ and vice-versa.

Proof. The sequentialization part is proved by induction on the size of a PN π with conclusions order $\sigma(\Gamma)$. The case when π is reduced to an axiom link is immediate, otherwise we reason as follows.

1 Assume π contains a terminal ∇-link $L : \frac{A \quad B}{A \nabla B}$, so π has conclusions $\sigma(\Gamma) = \Gamma' < A \nabla B$, then we can remove L and consider the proof structure π' with conclusions Γ', A, B which is correct since its abstract image π'^{ab} is contractible by Proposition 1 (case 1) and therefore it is sequentializable into a proof of $\vdash \sigma'(\Gamma', A, B)$. It remains to show that $\sigma'(\Gamma', A, B) = \Gamma' < A < B = \sigma(\Gamma)[A \nabla B / A < B] = \Gamma' < A \nabla B$. Assume there exists a conclusion C in $S(\pi')$ s.t. $(A < B < C) \notin S(\pi')$, like in the next picture



that is absurdum, by Lemma 7. Therefore π sequentializes as follows:

$$\dfrac{\begin{matrix}\Pi' \\ \vdash \Gamma' < A < B = \sigma'(\Gamma', A, B)\end{matrix}}{\vdash \Gamma' < A \nabla B = \sigma(\Gamma' < A \nabla B)}$$

2 Otherwise, π must be splitting at some link L, by Splitting Lemma 5. Assume L is a terminal splitting $\otimes$-link $L : \frac{A \quad B}{A \otimes B}$ and π has conclusions $\sigma(\Gamma) = \Gamma' < A \otimes B$; then by hypothesis of induction on the size of π, $\pi^1_{\Gamma_1, A}$ and $\pi^2_{\Gamma_2, B}$ are correct proof nets that sequentialize into a proof of $\sigma'_1(\Gamma_1, A) = \Gamma_1 < A$, resp., $\sigma'_2(\Gamma_2, B) = \Gamma_2 < B$, from which we can conclude, by Fact 4, with the following sequentialization of π, where $\sigma(\Gamma) = \Gamma_1 < A \otimes B < \Gamma_2$ with $\Gamma' = \Gamma_2 < \Gamma_1$:

$$\dfrac{\begin{matrix}\Pi_1 \\ \vdash \Gamma_1 < A\end{matrix} \qquad \begin{matrix}\Pi_2 \\ \vdash \Gamma_2 < B\end{matrix}}{\vdash \Gamma_1 < A \otimes B < \Gamma_2}\otimes$$

The case when π is splitting at a cut link is similar and so omitted.

Finally, the adequacy-part (the fact the proof net syntax is adequate to represent sequent proofs) is proved by induction on the hight of the given sequent proof. □

3. Lambek Calculus and McyLL Proof Nets as parsing structures

3.1. *Proof nets for Lambek Calculus*

In this section we characterize those McyLL PNs that correspond to Lambek proofs. The first (sound) notion of Lambek cut-free proof net, without sequentialization, was given in [Roorda 1992]; see also [Retoré 1996] and [Moot and Retoré 2012] for an original discussion on the embedding of Lambek Calculus into PNs.

Definition 13 ((pure-)Lambek formulas and sequents of McyLL). Assume A and S are, respectively, a formula and a sequent of McyLL .

1 A is a *(pure) Lambek formula (pLF)* if it is a McyLL formula recursively built according to this grammar: A := positive atoms $\mid A \otimes A \mid A^\perp \nabla A \mid A \nabla A^\perp$.

2 S is a *Lambek sequent of McyLL* iff $S = (\Gamma)^\perp, A$ where A is a non void pLF and $(\Gamma)^\perp$ is a possibly empty finite sequence of negations of pLFs (i.e., $(\Gamma)^\perp$ is obtained by taking the negation of each pLF in Γ).

3 A *(pure) Lambek proof* is any derivation built by means of the McyLL inference rules in which premise(s) and the conclusions are Lambek sequents.

Definition 14 (Lambek McyLL proof net). We call *Lambek McyLL proof net* any McyLL PN whose edges are labeled by pure LFs or negation of pure LFs and whose conclusions form a Lambek sequent.

Corollary 1. Any Lambek McyLL PN π is stable under cut reduction, i.e., if π reduces in one step to π', then π' is a Lambek McyLL PN too.

Proof. Consequence of Theorem 2. Any reduction step preserves the property that each edge (resp., the conclusion) of the reductum is labeled by a Lambek formula or by a negation of a Lambek formula (resp., by a Lambek sequent). □

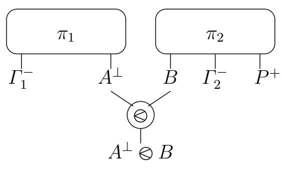
Theorem 5 (adequacy of Lambek PNs). Any Lambek proof of a sequent $\vdash \sigma(\Gamma^\perp, A)$ can be de-sequentialized in to a Lambek PN with same order conclusions $\sigma(\Gamma^\perp, A)$.

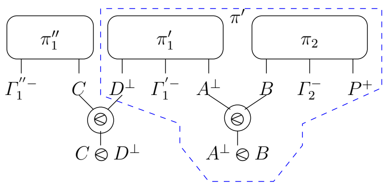
Proof. By induction on the height of the given sequent proof. □

Theorem 6 (sequentialization of Lambek PNs). Any Lambek McyLL proof net of $\sigma(\Gamma^\perp, A)$ sequentializes into a Lambek McyLL proof of the sequent $\vdash \sigma(\Gamma^\perp, A)$.

Proof. Assume by absurdum there exists a pure Lambek McyLL proof net π that does not sequentialize into a Lambek McyLL proof. We can chose π minimal w.r.t. the size. Clearly, π cannot be reduced to an axiom link; moreover π contains neither a negative conclusion of type $A^\perp \triangledown B^\perp$ nor a positive conclusion of type $A^\perp \triangledown B$ (resp., $A \triangledown B^\perp$), otherwise, we could remove this terminal $\triangledown$-link and get a strictly smaller (than π) proof net π' that is sequentializable, by minimality of π; this implies that also π is sequentializable (last inference rule of the sequent proof will be an instance of $\triangledown$-rule) contradicting the assumption. For same reasons (minimality), the unique positive conclusion (e.g. $A \otimes B$) of π cannot be splitting. Therefore, since π is not an axiom link $\overline{A^\perp \quad A}$, by Lemmas 5 and 6, there must exist either a (negative) splitting $\otimes$-link (Case 1) or a splitting cut-link (Case 2).

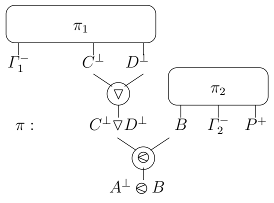
Case 1. Assume a negative splitting conclusion $A^\perp \otimes B$ (resp., $A \otimes B^\perp$). By minimality, π must split like in the next left hand side picture (we use A^+, resp. A^-, to denote positive, resp., negative, LF and Γ^- for sequence of negative LFs):

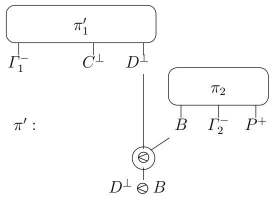




Now, let us reason on π_1 (reasoning on π_2 is symmetric): by minimality of π, π_1 cannot be reduced to an axiom link (otherwise Γ_1^- would not be negative); moreover, none of Γ_1^- is a (negative) splitting link, like e.g., $C \otimes D^\perp$, otherwise we could easily restrict to consider the sub-proof-net π', obtained by erasing from π the sub-proof-net π_1'' (with conclusions $\Gamma_1''^-, C$) together with the $C^\perp \otimes D$-link, like the graph enclosed in the dashed line above. Clearly, π' would be a non sequentializable Lambek proof net strictly smaller than π. In addition, π_1 must be cut-free, otherwise by minimality, after a cut-step reduction we could easily build a non sequentializable reductum PN π', strictly smaller than π, (π' will have same conclusions of π). Therefore, there are only two sub-cases:

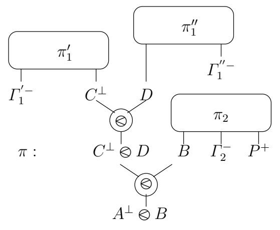
1 either $A^\perp = C^\perp \triangledown D^\perp$, then from the PN π on the l.h.s. of the next figure, we can easily get the non sequentializable PN π' (on the r.h.s.); π' is strictly smaller than π, contradicting the minimality assumption:

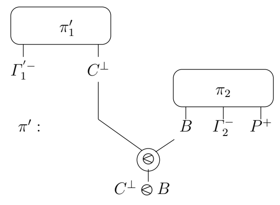




2 or $A^\perp = C^\perp \otimes D$, then this $C^\perp \otimes D$-link must split by Lemma 5, since π_1 is a cut-free

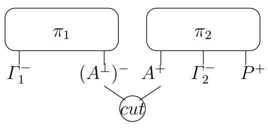
PN in splitting condition without other $\otimes$-splitting conclusion in Γ_1^-; so from π on the l.h.s., we can easily get the non sequentializable PN π' on r.h.s.; π' is strictly smaller of π, contradicting the minimality assumption:

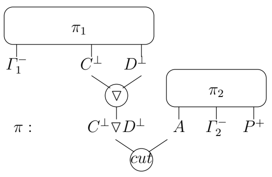


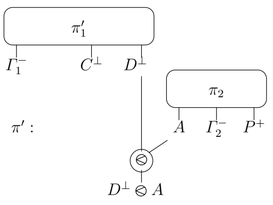


Case 2. Assume π contains a splitting cut link, like the leftmost hand side picture below, then we proceed like in Case 1. We reason on π_1 with two sub-cases:

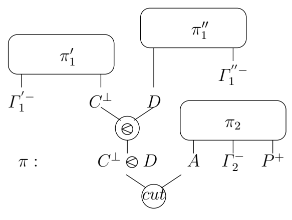
1 either $A^\perp = C^\perp \triangledown D^\perp$, then we can easily get, starting from the PN π on the middle side below, a non sequentializable PN π', like the rightmost hand side picture; $\pi'$is strictly smaller than π, contradicting the minimality assumption:

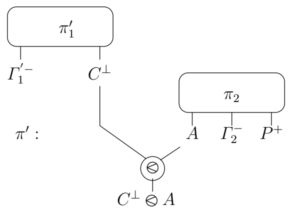






2 or $A^\perp = C^\perp \otimes D$, then this $A^\perp$-link must be splitting by Lemma 5, since π_1 is a cut-free PN in splitting condition without any other $\otimes$-splitting conclusion in Γ_1^-; so, we can easily get, starting from the PN π on the l.h.s., a non sequentializable PN π' that is strictly smaller than π (on the r.h.s.), contradicting the minimality assumption.





$\square$

3.2. *McyLL Proof Nets as parsing structures*

In this section we reformulate, in our syntax, some examples of linguistic parsing (some of them suggested by Richard Moot in his PhD thesis [Moot 2002]). We use s, np and n as the types expressing, respectively, a *sentence*, a *noun phrase* and a *common noun.* According to the "parsing as deduction style", when a string $w_1...w_n$ is tested for grammaticality, the types $t_1, ..., t_n$ associated with the words are retrieved from the lexicon and then parsing reduces to proving the derivability of a two-sided sequent of the form $t_1, ..., t_n \vdash s$. Remind that proving a two sided Lambek derivation $t_1, ..., t_n \vdash s$ is equivalent to prove the one-sided sequent $\vdash t_n^\perp, ...t_1^\perp, s$ where $t_i^\perp$ is the dual (i.e., linear negation) of type t_i. Any phrase or sentence should be read like in a mirror (with opposite direction).

Assume the following lexicon, where *linear implication* $\multimap$ (resp., $\multimapinv$) is traditionally used for expressing types in two-sided sequent parsing:

1	*Sollozzo, Vito*	$=$	np;				
2	*trusts*	$=$	$np \multimap (s \multimapinv np)$	$\equiv$	$np^\perp \nabla (s \nabla np^\perp)$		
		$=$	$(np \multimap s) \multimapinv np$	$\equiv$	$(np^\perp \nabla s) \nabla np^\perp$;		
3	*him*	$=$	$(s \multimapinv np) \multimap s$	$\equiv$	$(s \nabla np^\perp)^\perp \nabla s$	$\equiv$	$(np \otimes s^\perp) \nabla s$;

Cases of lexical ambiguity follow to words with several possible formulas A and B assigned it. For example, a verb like "*to believe*" can express a relation between two persons, np's in our interpretation, or between a person and a statement, interpreted as s, as in the following examples:

(1) *Sollozzo believes Vito.* (2) *Sollozzo believes Vito trusts him.*

We can express this verb ambiguity by two lexical assignments as follows:

4	*believes*	$=$	$(np \multimap s) \multimapinv np$	$\equiv$	$(np^\perp \nabla s) \nabla np^\perp$;
5	*believes*	$=$	$(np \multimap s) \multimapinv s$		$(np^\perp \nabla s) \nabla s^\perp$.

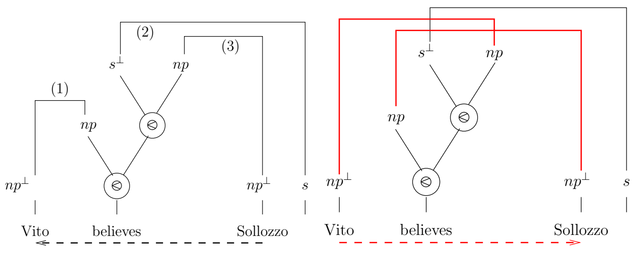
In order to parse sentence (1), "*Sollozzo believes Vito*", we may proceed in two ways:

— *via the sequent calculus*, building (bottom-up) a derivation tree in the sequent calculus:

— or *via proof structure*, by matching pairs of dual literals (i.e. linking) occurring in the top border of the syntactical trees of the types assigned to the lexical items, including the extra type for sentence s. Actually, there are two ways of linking dual pairs of literals $(np, np^\perp)$, both of them leading to correct proof nets:

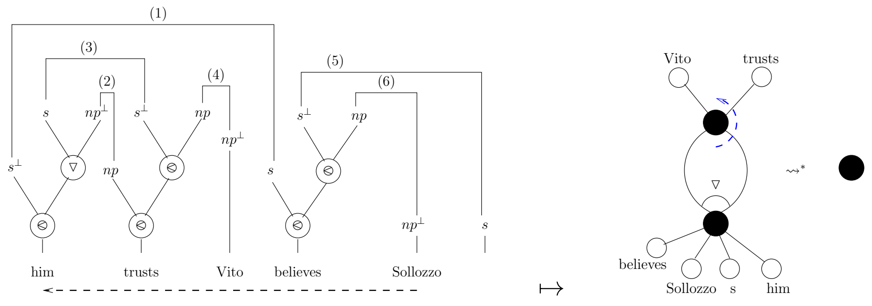
- the one in the middle side, with cyclic order conclusions "$s < Vito < believes < Sollozzo$", which sequentializes into the l.h.s. sequent proof, parsing of sentence 3.2;
- the one in the rightmost hand side, with cyclic order conclusions $s < Sollozzo < believes < Vito$, corresponding to the parsing of sentence "*Vito believes Sollozzo*".

$$\frac{\dfrac{}{np^\perp, np}\,id_1 \qquad \dfrac{\dfrac{}{s^\perp, s}\,id_2 \qquad \dfrac{}{np, np^\perp}\,id_3}{s^\perp \otimes np, np^\perp, s}\,\otimes}{np^\perp, np \otimes (s^\perp \otimes np), np^\perp, s}\,\otimes$$



Remind that, since we only consider one-side Lambek sequent proofs (proof nets), phrases or sentences should be read "like in a mirror" (following the dashed arrow below the conclusions), i.e., by inverting the "anticlockwise orientation" of the cyclic order conclusions.

Similarly, the parsing of sentence (2), "*Sollozzo believes Vito trusts him*", can be interpreted either by deriving (bottom up) a Lambek proof or by constructing, like below, a (bottom up) a Lambek PN (given together with its corresponding contractible APS):



Further examples can be found in Appendix A.3.

4. Conclusions and further works

In this paper we presented a correctness criterion for cyclic pure multiplicative (McyLL) proof nets satisfying a sequentialization for the full class of proof nets, including those ones with cut links. As shown in [Abrusci and Maieli 2015b]), the contraction criterion can be extended to consider the multiplicative and additive cyclic fragment of linear logic (MAcyLL), thereby allowing to parse superposition of phrases with lexical ambiguity (finite polymorphism). Intuitively it is enough to add an additive "box-like" contraction rule to the Σ system, with the proviso that the syntax of (abstract) proof structure has been enriched with links for the additive connectives & (with) and $\oplus$ (plus) together with extra (possibly n-ary) links for contraction (see Appendix A.4).

Retraction Systems represent a useful computational tool for:

— *proof search*, since we can chose special retraction strategies, like e.g. *parsing*, that are "optimal" w.r.t. e.g. complexity of search space, backtracking, ecc., (see [Maieli 2014]);
— classifying the *the complexity class* of correctness criteria; concerning the pure multiplicative fragment of linear logic, at this moment we know that deciding:
  - the correctness of a MLL proof structure is *linear* in the size of the input proof structure [Guerrini 2011] and *NL-complete* [de Naurois and Mogbil 2007];
  - correctness of McyLL proof structures, restricted to those ones that only allow a cut-free sequentialization (like e.g., [Maieli 2003]), is *quadratic* in the size of the input proof structure [Mogbil 2001].

As future work we aim at classifying the complexity class of the proposed new correctness criterion.

References

Abrusci, V. M. (2002) Classical conservative extensions of Lambek calculus. Studia Logica 71 (3):277 - 314 .

Abrusci, V. M. and Maieli, R. (2015a) Cyclic Multiplicative Proof Nets of Linear Logic with an Application to Language Parsing. In: Proc. of the Conference WoLLIC 2015, July 20-23, 2015, Bloomington, USA, LNCS 9160, pp. 53-68, 2015.

Abrusci, V. M. and Maieli, R. (2015b) Cyclic Multiplicative and Additive Proof Nets of Linear Logic with an Application to Language Parsing. In: Proc. of the Conference FG 2015. August 8-9, 2015, Barcelona. LNCS 9804, pp. 43-59 Springer, Heidelberg.

Abrusci, V. M. and Ruet, P. (2000) Non-commutative logic I: the multiplicative fragment. Annals of Pure and Applied Logic 101(1): 29-64.

Andreoli, J.-M. and Pareschi, R. (1991) From Lambek Calculus to word-based parsing. In: Proc. of Workshop on Substructural Logic and Categorial Grammar, CIS Munchen, Germany.

Bagnol, M., Doumane, A. and Saurin, A. (2015) On the Dependencies of Logical Rules. In: Proc. of the 18th Int.'l Conference, FoSSaCS 2015, London, UK, April 11-18, 2015, pp 436–450.

Danos, V. and Regnier, L. (1989) The Structure of Multiplicatives. AML, 28:181-203.

Danos, V. (1990) La Logique Linéaire appliquée à l'étude de divers processus de normalisation (principalment du λ-calcul). PhD Thesis, Univ. Paris VII.

de Naurois, P. J. and Mogbil, V. (2007) Correctness of Multiplicative (and Exponential) Proof Structures is *NL*-Complete. In: Proc. of CSL 2007, LNCS 4646, pp. 435–450, 2007.

Girard, J.-Y. (1987) Linear Logic. Theoretical Computer Science 50, 1–102.

Girard, J.-Y. (1996) Proof-nets: The parallel syntax for proof-theory Ursini, Agliano (Eds.), Logic and Algebra.

Girard, J-Y. (2006) Le point aveugle. Cours de Logique. Volume I. Ed. Hermann, Paris.

Guerrini, S. (2011) A linear algorithm for MLL proof net correctness and sequentialization. Theor. Comput. Sci. 412(20): 1958-1978.

Hughes, D. and van Glabbeek, R. (2003) Proof Nets for Unit-free Multiplicative-Additive Linear Logic. In: Proc. of the 18th IEEE Logic in Computer Science, Los Alamitos.

Lambek, J. (1958) The mathematics of sentence structure. American Mathematical Montly, 65.

Maieli, R. (2003) A new correctness criterion for multiplicative non commutative proof-nets. Archive for Mathematical Logic, vol.42, 205-220, Springer-Verlag.

Maieli, R. (2007) Retractile Proof Nets of the Purely Multiplicative and Additive Fragment of Linear Logic. In: Proc. of the 14th Int'l Conf. LPAR. LNAI 4790, pp. 363-377, Springer.

Maieli, R. (2014) Construction of Retractile Proof Structures. In: Proc. of the Int'l joint Conf. RTA-TLCA, July 14-17, 2014, Vienna. G. Dowek (ed.), LNCS 8560, pp. 319-333, Springer.

Melliès, P.-A. (2004) A topological correctness criterion for multiplicative non commutative logic. In: Ehrhard, T., Girard, J.-Y., Ruet, P., Scott, P. (eds.) Linear Logic in Computer Science. London Math. Soc. Lecture Notes, vol. 316, ch. 8, pp. 283-321. Cambridge University Press.

Mogbil, V. (2001) Quadratic Correctness Criterion for Non-commutative Logic. In: Proc. of CSL 2001, Paris, France, September 10-13. LNCS 2142, pp. 69–83. Springer.

Moot, R. and Retoré, Ch. (2012) The logic of categorial grammars: a deductive account of natural language syntax and semantics. Springer LNCS 6850.

Moot, R. (2002) Proof nets for linguistic analysis. PhD thesis, Utrecht University.

Pogodalla, S. and Retoré, C. (2004) Handsome Non-Commutative Proof-Nets: perfect matchings, series-parallel orders and Hamiltonian circuits. Tech. Rep. RR-5409, INRIA. In: Proc. of Categorial Grammars, Montpellier, France.

Retoré, C. (1996) Calcul de Lambek et logique linéaire. Traitement Automatique des Langues, vol. 37(2), pp. 39–70.

Retoré, C. (1997) A semantic characterization of the correctness of a proof net. Mathematical Structures in Computer Science, vol. 7(5), pp. 445-452.

Roorda, D. (1992) Proof nets for Lambek calculus. J. of Log. and Comp., vol. 2(2), pp. 21–233.

Appendix A.

A.1. *Melliès' topological criterion*

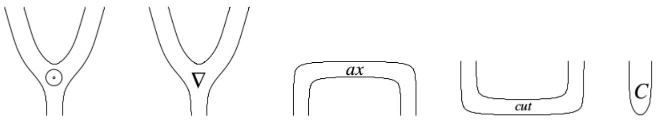
Definitions and examples of this section can be founded in [Melliès 2004].

Definition 15 (Melliès' correctness criterion). An McyLL (concrete) proof-structure π (Definition 2) is a proof-net iff:

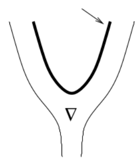
1 its commutative translation π^* is an MLL proof-net;
2 its surface **ribbon**(π) is planar with a unique external border σ;
3 σ contains all the conclusions of π.

The commutative translation π^* of an McyLL proof-structure π is the MLL proof-structure obtained as the result of replacing every $\odot$ and $\triangledown$ link by $\otimes$ and $\parr$, respectively, where according to Melliès' syntax "$\odot$" denotes the non commutative tensor (denoted in the rest of the paper by the usual "$\oslash$").

The second part of the criterion requires "planarity" of π, or more precisely planarity of the (orientable) surface **ribbon**(π) obtained by replacing every $\{\odot, \triangledown, axiom, cut\}$-link and conclusion C in π by the associated ribbon diagram as follows:



The criterion rejects the proof structure π of conclusion conclusion $\vdash (B \odot A) \multimap (A \odot B)$ because **ribbon**(π) in leftmost hand side of Figure 4 it is not planar. The crucial point is that planarity of **ribbon**(π) is not sufficient to characterize McyLL proofs among McyLL proof-structures. Typically, the McyLL proof-structure π of conclusion $\vdash (A^{\perp} \triangledown B^{\perp}), (A \odot B)$, in the middle side ribbon of Figure 4 is not sequentializable in McyLL, but its surface **ribbon**(π) is planar. One possible solution is to require that all conclusions of π lie on the same border of **ribbon**(π). Unfortunately, this criterion would be too weak to characterize proofs with cuts, as witnessed by the example of a non-sequentializable McyLL proof-structure, with a unique conclusion in the rightmost hand side of Figure 4. These (counter-)examples suggest to reinforce conditions 1 and 2 of Definition 15 by adding extra information (condition 3) concerning the $\triangledown$-link occurring in the border of **ribbon**(π). More precisely, given an McyLL proof structure π and a border σ of **ribbon**(π), we shall count the number of $\triangledown$-links visited by the border σ on their "thick side" displayed below:



We call this number the *index of σ*. A border of index 0 is called *external* and a border of index more than 1 is called *internal.* So, the criterion of Definition 15 rejects the two proof-structures on the right hand side of Figure 4 because one of their conclusions lies

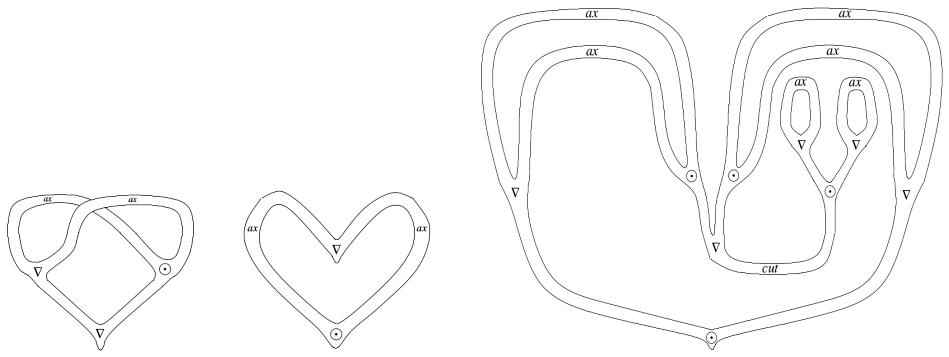


Fig. 4. examples of surface ribbons of proof structures

on an internal border. For similar reasons, proof structures π_1 and π_2 of Section 1.2.3 are non correct according to Definition 15.

A.2. *Sequentialization as parsing*

In spite of other syntaxes based on correction graphs (set of tests) like "switchings" or "trips", retraction gives a direct (simpler, indeed) sequentialization procedure of correct proof nets without passing through a Splitting Lemma. The genuine idea [Bagnol *et al.* 2015] is that each retraction step represents itself an inference of a possibly open sequent proof. For this purpose we adopt the intermediate syntax of *labeled abstract proof structures* (a "medium" between abstract proof structures and sequent proofs):

— nodes of abstract paired graphs are labeled with *open proofs* containing *context variables* $\Gamma^?$;

— *open proofs* correspond to *partial sequentializations*, which become larger and larger as contraction progresses, until reaching a full (complete, closed) McyLL proof;

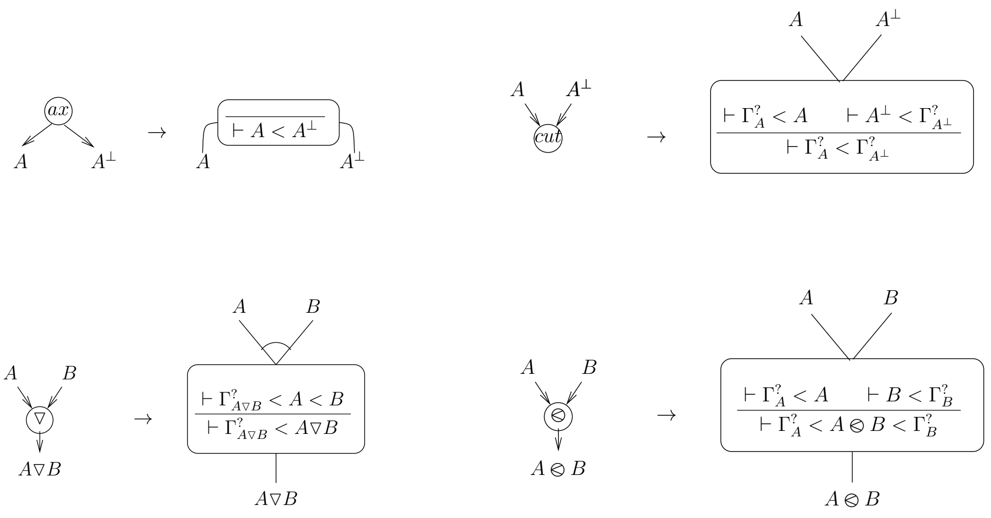
— open proofs are constructed on (i) cyclic ordered sequents S (i.e., S is endowed with a total cyclic order σ) with context variables, generated by the following syntax, where F is a McyLL formula and $\Gamma^?$ is a context variable:

$$S \;:=\; \emptyset \;\mid\; S, F \;\mid\; S, \Gamma^?$$

(ii) by the following inference rules:

$$\frac{}{\vdash A < A^\perp}\,id \qquad \vdash S \qquad \frac{\vdash S_1 < A \qquad A^\perp < S_2}{\vdash S_1 < S_2}\,cut \qquad \frac{\vdash S_1 < A \qquad \vdash B < S_2}{\vdash S_1 < A \otimes B < S_2}\,\otimes \qquad \frac{\vdash S < A < B}{\vdash S < A \triangledown B}\,\triangledown$$

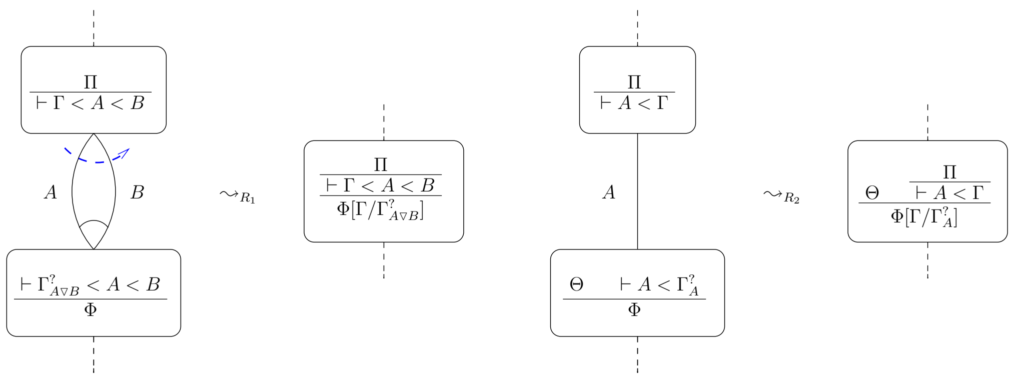
Given a concrete proof structure π, the *labeled paired graph* π^{lab} is a paired graph whose vertexes are labeled by open proofs (displayed inside rectangles) whose edges are labeled by formulas and obtained by applying the following *transformation rules*, from concrete proof structures (CPSs) to labeled paired graphs, also called *labeled abstract proof structures* (LAPSs):



ax
A
$A^\perp$
$\vdash A < A^\perp$
cut
$\vdash \Gamma^?_A < A \quad \vdash A^\perp < \Gamma^?_{A^\perp}$
$\vdash \Gamma^?_A < \Gamma^?_{A^\perp}$
B
$\triangledown$
$A \triangledown B$
$\vdash \Gamma^?_{A\triangledown B} < A < B$
$\vdash \Gamma^?_{A\triangledown B} < A\triangledown B$
$\otimes$
$A \otimes B$
$\vdash \Gamma^?_A < A \quad \vdash B < \Gamma^?_B$
$\vdash \Gamma^?_A < A \otimes B < \Gamma^?_B$
$\rightarrow$

(**transformation rules:** $CPSs \rightarrow LAPSs$)

Definition 16 (labelled retraction system Σ_l). The following rewriting or contraction rules are applied with (partial) substitution of context variables; where Π, Φ, Θ stay for possibly open sub-proofs and $\Phi[\Gamma/\Gamma^?_F]$ denotes the substitution of a context variable $\Gamma^?_F$ by a (closed) sequent Γ along an open proof Φ:



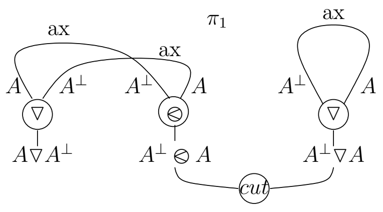
Π
$\vdash \Gamma < A < B$
A
B
$\vdash \Gamma^?_{A\triangledown B} < A < B$
Φ
$\rightsquigarrow_{R_1}$
$\Phi[\Gamma/\Gamma^?_{A\triangledown B}]$
$\vdash A < \Gamma$
$\Theta \quad \vdash A < \Gamma^?_A$
$\rightsquigarrow_{R_2}$
$\Phi[\Gamma/\Gamma^?_A]$

Definition 17 (Σ_l-correctness criterion ($\Sigma_l CC$)). A concrete proof structure π is Σ_l-correct when the corresponding LAPS π^{lab} collapses by Σ_l in to a single node labeled by a closed sequent proof Π (derived from only logical axioms, *id*).

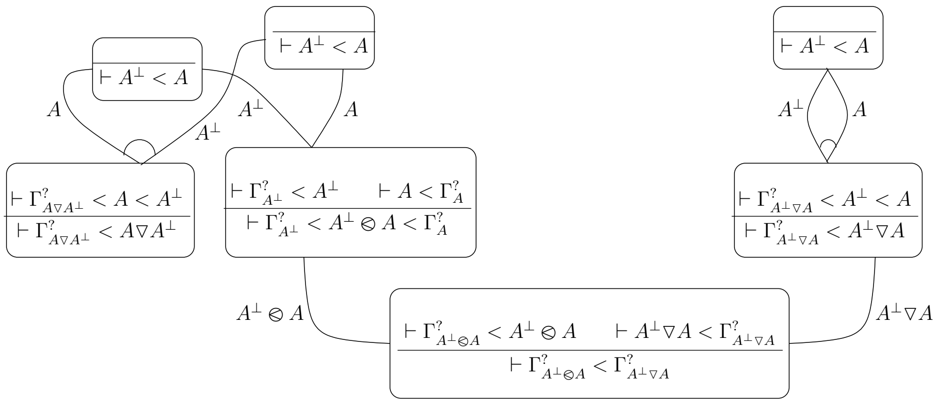
Theorem 7 ($\Sigma CC \equiv \Sigma_l CC$). A concrete proof structure π is Σ-correct (ΣCC, Definition 9) iff it is Σ_l-correct ($\Sigma_l CC$); moreover, π and the (closed) sequent proof Π, labeling the collapsed graph (resulting from the Σ_l-contraction of π^{lab}), have the same conclusions Γ endowed by the same cyclic order $\sigma(\Gamma)$.

Proof. By induction on π. $\square$

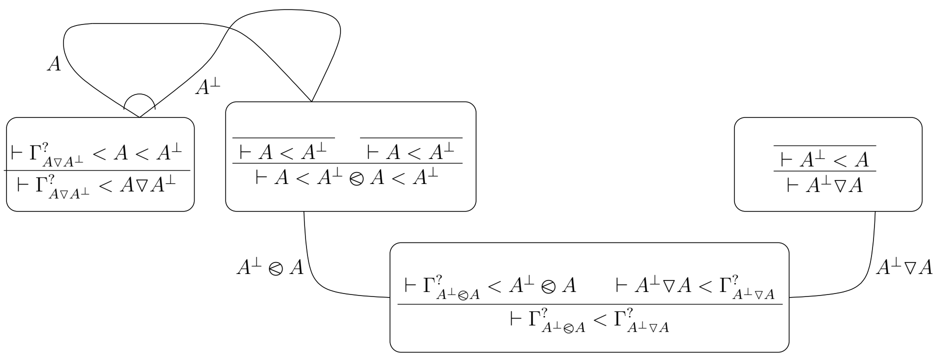
Example 2. Assume the following concrete proof structure π_1:



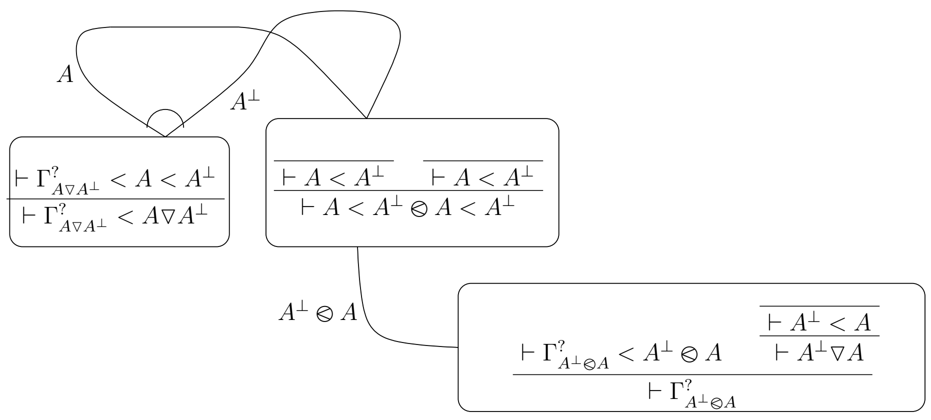
In order to check Σ_l-correctness of π, we first transform it into the LAPS π^{lab} below:


$\vdash A^\perp < A$
$\vdash A^\perp < A$
$\vdash A^\perp < A$
A
$A^\perp$
$A^\perp$
A
$A^\perp$
A
$\vdash \Gamma^?_{A\triangledown A^\perp} < A < A^\perp$ / $\vdash \Gamma^?_{A\triangledown A^\perp} < A\triangledown A^\perp$
$\vdash \Gamma^?_{A^\perp} < A^\perp$ $\vdash A < \Gamma^?_A$ / $\vdash \Gamma^?_{A^\perp} < A^\perp \otimes A < \Gamma^?_A$
$\vdash \Gamma^?_{A^\perp\triangledown A} < A^\perp < A$ / $\vdash \Gamma^?_{A^\perp\triangledown A} < A^\perp\triangledown A$
$A^\perp \otimes A$
$A^\perp\triangledown A$
$\vdash \Gamma^?_{A^\perp\otimes A} < A^\perp \otimes A$ $\vdash A^\perp\triangledown A < \Gamma^?_{A^\perp\triangledown A}$ / $\vdash \Gamma^?_{A^\perp\otimes A} < \Gamma^?_{A^\perp\triangledown A}$

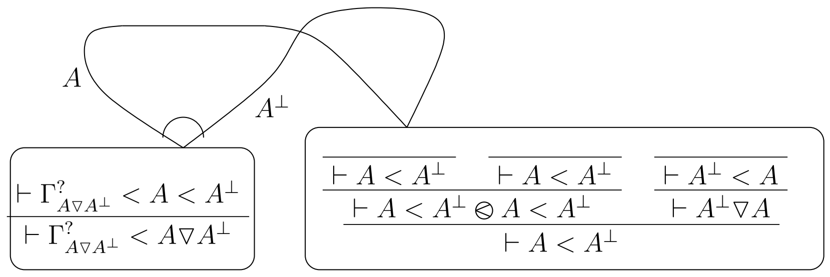
then, after a couple instances of R_1 (with substitution $[A/\Gamma^?_{A^\perp}]$ resp., $[A^\perp/\Gamma^?_A]$) and one instance of R_2 (with substitution $[\emptyset/\Gamma^?_{A^\perp\triangledown A}]$) we get the following LAPS:


A
$A^\perp$
$\vdash \Gamma^?_{A\triangledown A^\perp} < A < A^\perp$ / $\vdash \Gamma^?_{A\triangledown A^\perp} < A\triangledown A^\perp$
$\vdash A < A^\perp$ $\vdash A < A^\perp$ / $\vdash A < A^\perp \otimes A < A^\perp$
$\vdash A^\perp < A$ / $\vdash A^\perp\triangledown A$
$A^\perp \otimes A$
$A^\perp\triangledown A$
$\vdash \Gamma^?_{A^\perp\otimes A} < A^\perp \otimes A$ $\vdash A^\perp\triangledown A < \Gamma^?_{A^\perp\triangledown A}$ / $\vdash \Gamma^?_{A^\perp\otimes A} < \Gamma^?_{A^\perp\triangledown A}$

from which, after an instance of R_1 (with substitution $[\emptyset/\Gamma^?_{A^\perp\triangledown A}]$) we get next structure:


A
$A^\perp$
$\vdash \Gamma^?_{A\bigtriangledown A^\perp} < A < A^\perp$
$\vdash \Gamma^?_{A\bigtriangledown A^\perp} < A\bigtriangledown A^\perp$
$\vdash A < A^\perp$
$\vdash A < A^\perp$
$\vdash A < A^\perp \olessthan A < A^\perp$
$A^\perp \olessthan A$
$\vdash A^\perp < A$
$\vdash \Gamma^?_{A^\perp\olessthan A} < A^\perp \olessthan A$
$\vdash A^\perp\bigtriangledown A$
$\vdash \Gamma^?_{A^\perp\olessthan A}$

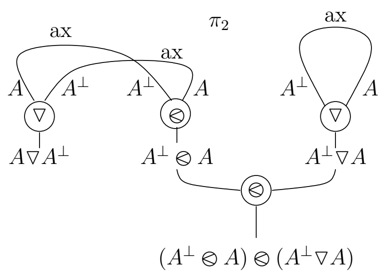
then, by an other instance of R_1 (with substitution $[(A^\perp < A)/\Gamma^?_{A^\perp\olessthan A}]$ we get:


A
$A^\perp$
$\vdash \Gamma^?_{A\bigtriangledown A^\perp} < A < A^\perp$
$\vdash \Gamma^?_{A\bigtriangledown A^\perp} < A\bigtriangledown A^\perp$
$\vdash A < A^\perp$
$\vdash A < A^\perp$
$\vdash A^\perp < A$
$\vdash A < A^\perp \olessthan A < A^\perp$
$\vdash A^\perp\bigtriangledown A$
$\vdash A < A^\perp$

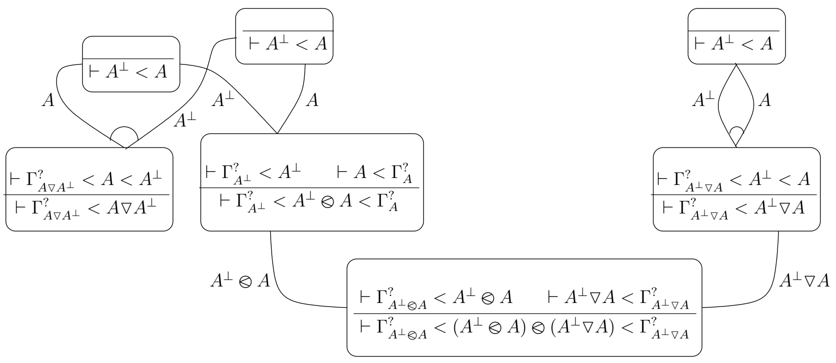
from which, finally, after an instance of R_2 (with substitution $[\emptyset/\Gamma^?_{A^\perp\bigtriangledown A}]$ we get a collapsed LAPS (a single node) labeled by the following closed proof Π (a sequentialization of π):

$$\boxed{\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash A < A^\perp} \qquad \dfrac{}{\vdash A < A^\perp}}{\vdash A < A^\perp \olessthan A < A^\perp} \qquad \dfrac{\dfrac{}{\vdash A^\perp < A}}{\vdash A^\perp\bigtriangledown A}}{\vdash A < A^\perp}}{\vdash A^\perp\bigtriangledown A}}$$

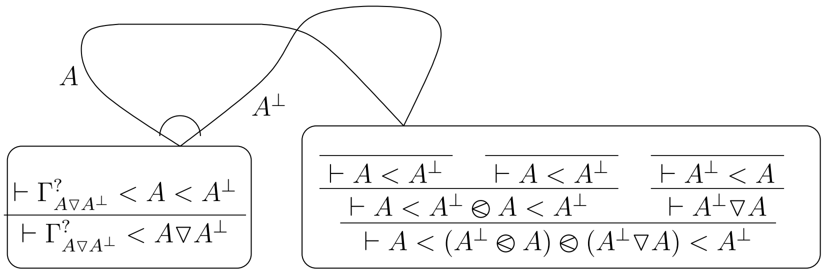
Example 3. Assume we want to test for correctness the following (incorrect indeed) proof structure π_2:


π_2
ax
ax
ax
A
$A^\perp$
$A^\perp$
A
$A^\perp$
A
$A\bigtriangledown A^\perp$
$A^\perp \olessthan A$
$A^\perp\bigtriangledown A$
$(A^\perp \olessthan A) \olessthan (A^\perp\bigtriangledown A)$

first, let us transform π_2 into the labeled proof structure below:


$\vdash A^\perp < A$
$\vdash A^\perp < A$
$\vdash A^\perp < A$
A
$A^\perp$
$A^\perp$
A
$A^\perp$
A
$\vdash \Gamma^?_{A\nabla A^\perp} < A < A^\perp$ / $\vdash \Gamma^?_{A\nabla A^\perp} < A\nabla A^\perp$
$\vdash \Gamma^?_{A^\perp} < A^\perp \quad \vdash A < \Gamma^?_A$ / $\vdash \Gamma^?_{A^\perp} < A^\perp \otimes A < \Gamma^?_A$
$\vdash \Gamma^?_{A^\perp\nabla A} < A^\perp < A$ / $\vdash \Gamma^?_{A^\perp\nabla A} < A^\perp\nabla A$
$A^\perp \otimes A$
$A^\perp\nabla A$
$\vdash \Gamma^?_{A^\perp\otimes A} < A^\perp \otimes A \quad \vdash A^\perp\nabla A < \Gamma^?_{A^\perp\nabla A}$ / $\vdash \Gamma^?_{A^\perp\otimes A} < (A^\perp \otimes A) \otimes (A^\perp\nabla A) < \Gamma^?_{A^\perp\nabla A}$

then, after a couple instances of R_1, one instance of R_2 and one more instance of R_1 (with the same substitutions seen before for π_1) we get the structure below which is not contractible anymore:


A
$A^\perp$
$\vdash \Gamma^?_{A\nabla A^\perp} < A < A^\perp$ / $\vdash \Gamma^?_{A\nabla A^\perp} < A\nabla A^\perp$
$\vdash A < A^\perp \quad \vdash A < A^\perp \quad \vdash A^\perp < A$ / $\vdash A < A^\perp \otimes A < A^\perp \quad \vdash A^\perp\nabla A$ / $\vdash A < (A^\perp \otimes A) \otimes (A^\perp\nabla A) < A^\perp$

observe that, the only possible substitution $[(A^\perp \otimes A) \otimes (A^\perp\nabla A)/\Gamma^?_{A\nabla A^\perp}]$ does not led to a cut-free derivable proof of the sequent $\vdash (A^\perp \otimes A) \otimes (A^\perp\nabla A) < A\nabla A^\perp$, as below.

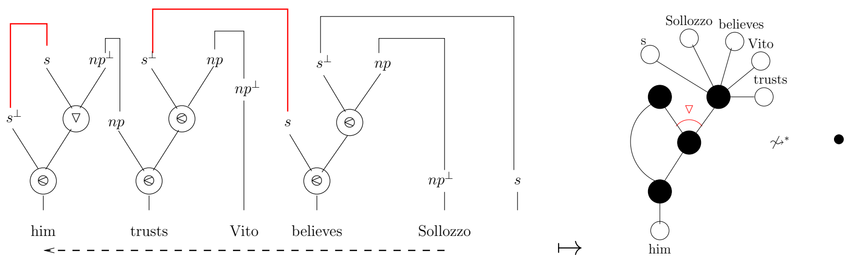
$$\dfrac{\dfrac{\dfrac{}{\vdash A < A^\perp} \qquad \dfrac{}{\vdash A < A^\perp}}{\vdash A < A^\perp \otimes A < A^\perp} \qquad \dfrac{\dfrac{}{\vdash A^\perp < A}}{\vdash A^\perp\nabla A}}{\vdash (A^\perp \otimes A) \otimes (A^\perp\nabla A) < A^\perp < A}$$

non (cut-free) derivable!

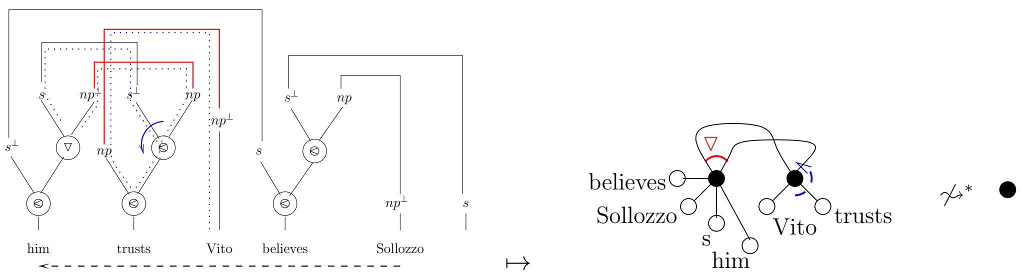
$$\vdash (A^\perp \otimes A) \otimes (A^\perp\nabla A) < A\nabla A^\perp$$

A.3. *Examples of Parsing via Lambek McyLL proof nets*

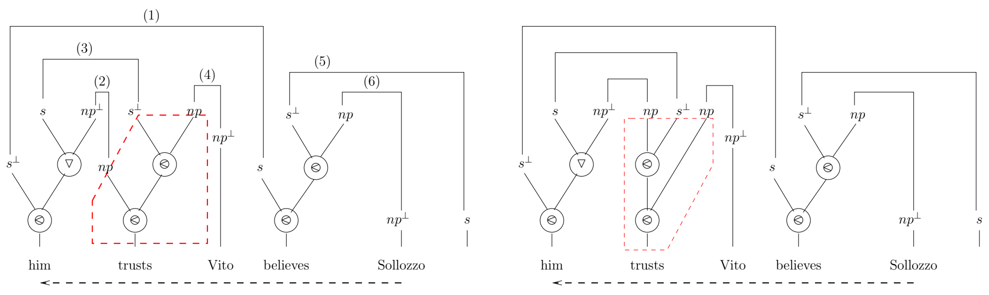
In order to better understand how the contraction criterion checks the incorrectness of wrong parsing of Lambek proof structure, we consider in the following a couple of incorrect parsing structures of sentence (2): the first one results from a wrong choice of linking (by axioms) some pairs of literal $s, s^\perp$



while the second one results from a incorrect linkings of some pairs of literals $np, np^{\perp}$



Proof nets are modular objects, that is, you can always replace in a proof-net a portion (also called "module") with an other graph with same "behavior" (in term of local correction) and get still a correct proof net. For instance, consider the two alternative parsing structures for sentence (2): they only differ for the module enclosed within dashed lines: two different syntactical types (even though logically equivalent!) for the same lexical item "trusts"$= np\multimap(s\circ\!\!- np) \ = \ np^{\perp}\nabla(s\nabla np^{\perp})$.



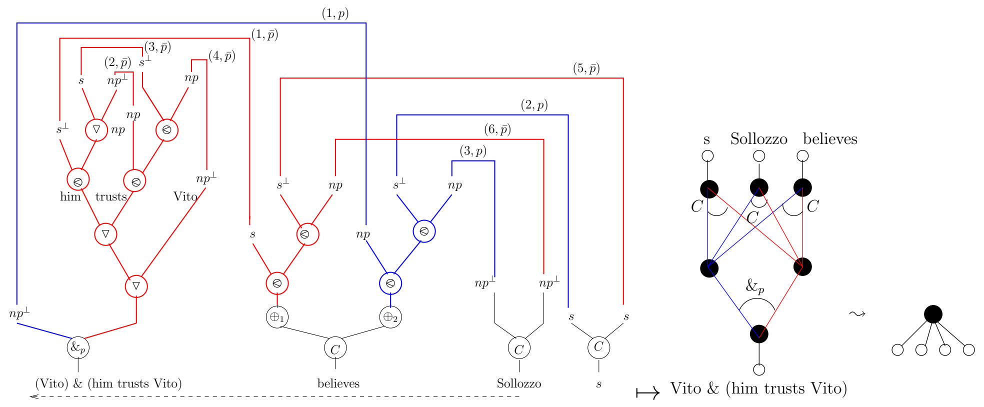
A.4. *Further work: parsing via Lambek MAcyLL proof nets*

Additive connectives, & and $\oplus$, allow superpositions of types (lexical ambiguity); in particular we may collapse the previous assignment items 4 and 5 of Section 3.2 for the lexical entry "*believes*" into a single additive assignment as follows:

6 believes $= ((np\multimap s)\circ\!\!- np)\&((np\multimap s)\circ\!\!- s) = ((np^{\perp}\nabla s)\nabla np^{\perp})\&((np^{\perp}\nabla s)\nabla s^{\perp})$.

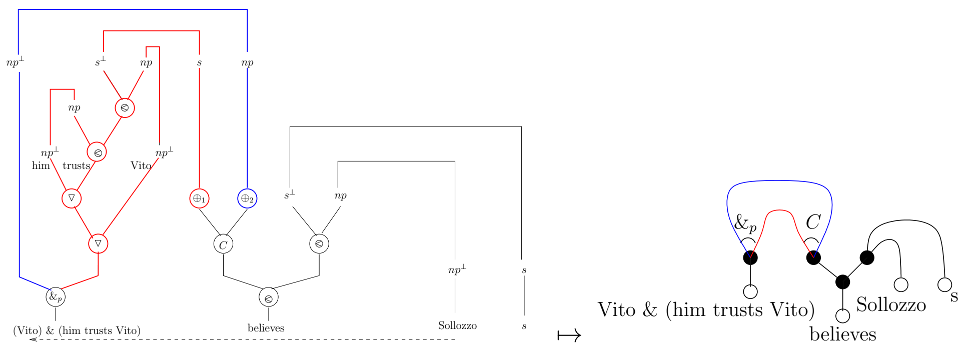
Then, as parsing structure of the superposition of sentences (1) and (2) we may build:

1 either a MAcyLL proof net with "minimal superposition of links"; this proof net is very close to the "sequent style" parsing since it makes use of "additive-boxes":



2 or a more abstract proof net (with ”maximal superposition of links“), which exploits a more compact lexical entry (due to the *distributivity law* of negative connectives)

$$\begin{aligned} \text{believes} \quad &= \quad ((np \multimap s) \circ\!\!- np) \& ((np \multimap s) \circ\!\!- s) \equiv ((np^{\perp} \nabla s) \nabla np^{\perp}) \& ((np^{\perp} \nabla s) \nabla s^{\perp}) \\ &= \quad ((np \multimap s) \circ\!\!- (s \oplus nps)) = ((np^{\perp} \nabla s) \nabla (np^{\perp} \& s^{\perp})). \end{aligned}$$


him
trusts
Vito
(Vito) & (him trusts Vito)
believes
Sollozzo
↦
&p
C
Vito & (him trusts Vito)
Sollozzo
s
believes