# 6

# Lambek Calculus and Linear Logic: Proof Nets as Parse Structures

**Summary.** This chapter, a large part of which is a translation of (Retoré, 1996), deals with the connection between Lambek categorial grammar and linear logic, the main objective being the presentation of proof nets which are excellent parse structures, because they identify linguistically equivalent analyses of a given sentence.

This graphical notation for proofs that are parse structures in categorial grammar is a not a mere variation for convenience. On a technical ground, it avoids the so-called spurious ambiguity problem of categorial grammars (the fact that we can find many different proofs/parse structures for what corresponds to a single analysis or lambda term). Conceptually, this proof syntax is a justification of the use of the expression *parsing as deduction* often associated with categorial grammar. Indeed proof nets only distinguish between proofs which correspond to different syntactic analyses.

We first give a rather complete presentation of the correspondence between the Lambek calculus and variants of multiplicative linear logic, since the Lambek calculus can be defined as non-commutative intuitionistic multiplicative linear logic without empty antecedents.

Next we define proof nets and establish their correspondence with the more traditional sequent calculus, present parsing as proof net construction and present some recent descriptions of non commutative proof nets.

As an evidence of their linguistic relevance, we explain how they provide a formal account of some performance questions, like the complexity of the processing of several intricate syntactic constructs, like center embedded relatives, garden path phenomena and preferred readings.

## 6.1 The Formula Language of Categorial Grammar and of Linear Logic

### 6.1.1 The Formula Language of Multiplicative Linear Logic

Let us recall the language of the Lambek calculus:

$$\mathsf{Lp} \quad ::= \quad \mathsf{P} \quad | \quad (\mathsf{Lp} \bullet \mathsf{Lp}) \quad | \quad (\mathsf{Lp} / \mathsf{Lp}) \quad | \quad (\mathsf{Lp} \backslash \mathsf{Lp})$$

R. Moot and C. Retoré: The Logic of Categorial Grammars 2012, LNCS 6850, pp. 193–238, 2012.

As we have seen in the previous chapters $\backslash$ and $/$ are implications, and the product $\bullet$ is a conjunction. All these connectives are linear logic connectives, but are rather denoted by: $\circ\!-, -\!\circ, \otimes$ in the linear logic community.

| Lambek calculus | $\backslash$ | $/$ | $\bullet$ |
|---|---|---|---|
| Linear logic | $-\!\circ$ | $\circ\!-$ | $\otimes$ |

Multiplicative linear logic is a classical calculus which extends the Lambek calculus by a negation denoted by $(\ldots)^{\perp}$ (*the orthogonal of*…) together with the symmetries induced by a classical negation: the familiar De Morgan identities of classical logic.

To be precise, Multiplicative Linear Logic extends the Lambek calculus without the non empty antecedent requirement, and allows for permutation (hypotheses can be permuted). In order to have a single involutive negation and two distinct implications $\circ\!-$ and $-\!\circ$, one must restrict the allowed permutations to *cyclic* permutations. In the absence of any form of permutation, there have to be two negations (Abrusci, 1991, 1995).

Because of the De Morgan identities, there will be a disjunction $\wp$ (*par*, standing for *in parallel with*) corresponding to the conjunction $\otimes$. As we are especially interested in having a non commutative conjunction, the disjunction, by duality, will be non commutative as well.

Such a disjunction and a classical negation allow the implication $A \backslash B$ to be defined as $A^{\perp} \wp B$ and the implication $B / A$ to be defined as $B \wp A^{\perp}$ — just like it is possible to define $A \Rightarrow B$ as $\neg A \vee B$ in classical logic. Notice that the non commutativity of the disjunction is necessary if we want to be able to distinguish between these two implications.

In the Lambek calculus, one has the following equivalence: $(C / B) / A \equiv C / (A \bullet B)$ : indeed $(C / B) / A$ is a formula which requires an $A$ and then a $B$ to obtain $C$, and $C / (A \bullet B)$ is a formula which requires an *A followed by a B*, to obtain a $C$. The formula $(C / B) / A$ can be written as $C \wp B^{\perp} \wp A^{\perp}$ using the (associative) disjunction and the formula $C / (A \bullet B)$ as $C \wp (A \otimes B)^{\perp}$. Therefore if there is a classical extension of the Lambek calculus then negation has to swap the components of a disjunction (and of a conjunction, by duality).

Linear logic, when seen as a classical extension of the Lambek calculus, has the following language:

$$\mathsf{Li}_{+} \quad ::= \quad \mathsf{P} \mid \mathsf{Li}_{+}^{\perp} \mid (\mathsf{Li}_{+} \wp \mathsf{Li}_{+}) \mid (\mathsf{Li}_{+} \otimes \mathsf{Li}_{+}) \mid (\mathsf{Li}_{+} \backslash \mathsf{Li}_{+}) \mid (\mathsf{Li}_{+} / \mathsf{Li}_{+})$$

and enjoys the elimination of double negation and the De Morgan identities, as shown below.

$$(A^{\perp})^{\perp} \equiv A \qquad (A \wp B)^{\perp} \equiv B^{\perp} \otimes A^{\perp} \qquad (A \otimes B)^{\perp} \equiv B^{\perp} \wp A^{\perp}$$

### 6.1.2 Reduced Linear Language (Negative Normal Form)

For every formula $X$ in $\mathsf{Li}_+$ there exists a unique equivalent formula $+X$ such that negation only applies to propositional variables, and its only connectives are conjunction and disjunction. In some books, the analogous of $+X$ for classical logic is called its negative normal form. The formula $+X$ is obtained by replacing its implication by its definition as a disjunction, and then applying De Morgan identities as rewriting rules from left to right, and, finally by cancelling double negations. Notice that, unlike disjunctive normal form and conjunctive normal form, this form does not require distributivity of $\wp$ w.r.t. $\otimes$ or $\otimes$ w.r.t. $\wp$ — these distributivity identities do not hold in linear logic.[1]

So every formula in $\mathsf{Li}_+$ is equivalent to a formula $+X$ in $\mathsf{Li}$, where $\mathsf{Li}$ is:

$$\mathsf{Li} ::= \mathsf{N} \mid \mathsf{Li} \wp \mathsf{Li} \mid \mathsf{Li} \otimes \mathsf{Li} \qquad \text{where } \mathsf{N} = \mathsf{P} \cup \mathsf{P}^\perp \text{ is the set of } \textit{atoms}.$$

Observe that if $F \in \mathsf{Li}$ then $+F = F$.

Let us denote by $-F$ the unique formula in $\mathsf{Li}$ equivalent to $(F)^\perp \in \mathsf{Li}_+$ — $-F = +(F^\perp)$. Given $+F$, $-F$ is obtained by replacing every propositional variable in $+F$ with its negation, every conjunction by a disjunction, every disjunction by a conjunction, and finally by *reversing the left to right order of the result*.

Given $F = (\alpha^\perp \wp \beta) \otimes \gamma^\perp$ one first obtains $F' = (\alpha \otimes \beta^\perp) \wp \gamma$, which yields $F^\perp \equiv -F = \gamma \wp (\beta^\perp \otimes \alpha)$ by rewriting $F'$ from right to left.

### 6.1.3 Relating Categories and Linear Logic Formulae: Polarities

Since $\mathsf{Lp}$ is a sublanguage of $\mathsf{Li}_+$, for every formula $L$ in $\mathsf{Lp}$ there exists a unique formula $+L$ in $\mathsf{Li}$ which is equivalent to $L$ and a unique formula $-L$ which is equivalent to $L^\perp$. These two maps from $\mathsf{Lp}$ to $\mathsf{Li}$ can be inductively defined as follows:

| $L$ | $\alpha \in \mathsf{P}$ | $L = M \bullet N$ | $L = M \backslash N$ | $L = N / M$ |
|---|---|---|---|---|
| $+L$ | $\alpha$ | $+M \otimes +N$ | $-M \wp +N$ | $+N \wp -M$ |
| $-L$ | $\alpha^\perp$ | $-N \wp -M$ | $-N \otimes +M$ | $+M \otimes -N$ |

*Example 6.1*

| $L$ | $+L$ | $-L$ | |
|---|---|---|---|
| $np$ | $np$ | $np^\perp$ | noun phrase |
| $np / n$ | $np^\perp \wp n$ | $n^\perp \otimes np$ | determiner |
| $n$ | $n$ | $n^\perp$ | common noun |
| $n \backslash n$ | $n^\perp \wp n$ | $n^\perp \otimes n$ | right adjective |
| $(n \backslash n) / (n \backslash n)$ | $(n^\perp \wp n) \wp (n^\perp \otimes n)$ | $(n^\perp \wp n) \otimes (n^\perp \otimes n)$ | left modifier for right adjectives |
| $\beta \backslash ((\alpha / \beta) \backslash \alpha)$ | $\beta^\perp \wp ((\beta \otimes \alpha^\perp) \wp \alpha)$ | $(\alpha^\perp \otimes (\alpha \wp \beta^\perp)) \otimes \beta)$ | type raising |

[1] Though the classical distributivities, such as the equivalences $A \wedge (B \vee C) \leftrightarrow (A \wedge B) \vee (A \wedge C)$ and $A \vee (B \wedge C) \leftrightarrow (A \vee B) \wedge (A \vee C)$ which are required for disjunctive and conjunctive normal form do not hold between tensor and par, we do have some weaker *implications*, eg. $A \wedge (B \vee C) \to (A \wedge B) \vee C$, or written using tensor and par: $A \otimes (B \wp C) \vdash (A \otimes B) \wp C$.

Let us consider the following sets of formulae, which enable us to recognize, among linear formulae the ones which are Lambek formulae or the negation of Lambek formulae.

$$\begin{array}{l} \mathsf{Li}^\circ = \{F \in \mathsf{Li}/\exists L \in \mathsf{Lp} \quad +L = F\} \text{ : positive linear formulae} \\ \mathsf{Li}^\bullet = \{F \in \mathsf{Li}/\exists L \in \mathsf{Lp} \quad -L = F\} \text{ : negative linear formulae} \\ \mathsf{Li}^\circ \cup \mathsf{Li}^\bullet \quad : \quad \text{intuitionistic or polarized linear formulae} \end{array}$$

We then have:

$$\begin{array}{l} F \in \mathsf{Li}^\bullet \Leftrightarrow -F \in \mathsf{Li}^\circ \qquad \text{and} \qquad F \in \mathsf{Li}^\circ \Leftrightarrow -F \in \mathsf{Li}^\bullet \\ \mathsf{Li}^\circ \cup \mathsf{Li}^\bullet \neq \mathsf{Li} \text{ — for instance } \alpha \wp \beta \notin \mathsf{Li}^\circ \cup \mathsf{Li}^\bullet \\ \mathsf{Li}^\bullet \cap \mathsf{Li}^\circ = \emptyset \text{ — because of the following proposition:} \end{array}$$

**Proposition 6.2.** *The sets of formulae* $\mathsf{Li}^\circ$ *and* $\mathsf{Li}^\bullet$ *are inductively defined by:*

$$\left[\begin{array}{l} \mathsf{Li}^\circ ::= \ \mathsf{P} \mid (\mathsf{Li}^\circ \otimes \mathsf{Li}^\circ) \mid (\mathsf{Li}^\bullet \wp \mathsf{Li}^\circ) \mid (\mathsf{Li}^\circ \wp \mathsf{Li}^\bullet) \\ \mathsf{Li}^\bullet ::= \mathsf{P}^\perp \mid (\mathsf{Li}^\bullet \wp \mathsf{Li}^\bullet) \mid (\mathsf{Li}^\circ \otimes \mathsf{Li}^\bullet) \mid (\mathsf{Li}^\bullet \otimes \mathsf{Li}^\circ) \end{array}\right.$$

*The maps* $+$ *and* $-$ *are bijections from* $\mathsf{Lp}$ *to* $\mathsf{Li}^\circ$ *and* $\mathsf{Li}^\bullet$ *respectively.*

If $(\ldots)^\circ_{\mathsf{Lp}}$ denotes the inverse bijection of $+$, from $\mathsf{Li}^\circ$ to $\mathsf{Lp}$ and if $(\ldots)^\bullet_{\mathsf{Lp}}$ denotes the inverse bijection of $-$ from $\mathsf{Li}^\bullet$ to $\mathsf{Lp}$. Then these two maps are inductively defined as follows:

| $F \in \mathsf{Li}^\circ$ | $\alpha \in \mathsf{P}$ | $(G \in \mathsf{Li}^\circ) \otimes (H \in \mathsf{Li}^\circ)$ | $(G \in \mathsf{Li}^\bullet) \wp (H \in \mathsf{Li}^\circ)$ | $(G \in \mathsf{Li}^\circ) \wp (H \in \mathsf{Li}^\bullet)$ |
|---|---|---|---|---|
| $F^\circ_{\mathsf{Lp}}$ | $\alpha$ | $G^\circ_{\mathsf{Lp}} \otimes H^\circ_{\mathsf{Lp}}$ | $G^\bullet_{\mathsf{Lp}} \setminus H^\circ_{\mathsf{Lp}}$ | $G^\circ_{\mathsf{Lp}} / H^\bullet_{\mathsf{Lp}}$ |
| $F \in \mathsf{Li}^\bullet$ | $\alpha^\perp \in \mathsf{P}^\perp$ | $(G \in \mathsf{Li}^\bullet) \wp (H \in \mathsf{Li}^\bullet)$ | $(G \in \mathsf{Li}^\circ) \otimes (H \in \mathsf{Li}^\bullet)$ | $(G \in \mathsf{Li}^\bullet) \otimes (H \in \mathsf{Li}^\circ)$ |
| $F^\bullet_{\mathsf{Lp}}$ | $\alpha$ | $H^\bullet_{\mathsf{Lp}} \otimes G^\bullet_{\mathsf{Lp}}$ | $H^\bullet_{\mathsf{Lp}} / G^\circ_{\mathsf{Lp}}$ | $H^\circ_{\mathsf{Lp}} \setminus G^\bullet_{\mathsf{Lp}}$ |

The inductive definition of $\mathsf{Li}^\circ$ and $\mathsf{Li}^\bullet$ yields an easy decision procedure to check whether a formula $F$ is in $\mathsf{Li}^\circ$ or $\mathsf{Li}^\bullet$ — if so, all subformulae of $F$ are in $\mathsf{Li}^\circ$ or in $\mathsf{Li}^\bullet$: replace every propositional variable with $\circ$ and every negation of a propositional variable with $\bullet$ and compute using $\wp$ and $\otimes$ as the following operations on $\star, \circ, \bullet$ :

| $\wp$ | $\star$ | $\circ$ | $\bullet$ |
|---|---|---|---|
| $\star$ | $\star$ | $\star$ | $\star$ |
| $\circ$ | $\star$ | $\star$ | $\circ$ |
| $\bullet$ | $\star$ | $\circ$ | $\bullet$ |

| $\otimes$ | $\star$ | $\circ$ | $\bullet$ |
|---|---|---|---|
| $\star$ | $\star$ | $\star$ | $\star$ |
| $\circ$ | $\star$ | $\circ$ | $\bullet$ |
| $\bullet$ | $\star$ | $\bullet$ | $\star$ |

The result of this simple computation is used as follows:

- $\star$ whenever the formula is neither in $\mathsf{Li}^\circ$ nor in $\mathsf{Li}^\bullet$
- $\circ$ whenever the formula is in $\mathsf{Li}^\circ$
- $\bullet$ whenever the formula is in $\mathsf{Li}^\bullet$

*Example 6.3*

| $F$ | computation | conclusion | $F^\circ_{\mathsf{Lp}}$ | $F^\bullet_{\mathsf{Lp}}$ |
|---|---|---|---|---|
| $(\alpha^\perp \wp \beta) \wp \alpha$ | $(\bullet \wp \circ) \wp \circ = \circ \wp \circ = \star$ | $F \notin \mathsf{Li}^\circ \cup \mathsf{Li}^\bullet$ | undefined | undefined |
| $(\alpha^\perp \wp \beta) \wp \alpha^\perp$ | $(\bullet \wp \circ) \wp \bullet = \circ \wp \bullet = \circ$ | $F \in \mathsf{Li}^\circ$ | $(\alpha \backslash \beta) / \alpha$ | undefined |
| $(\alpha^\perp \wp \beta) \otimes \alpha^\perp$ | $(\bullet \wp \circ) \otimes \bullet = \circ \otimes \bullet = \bullet$ | $F \in \mathsf{Li}^\bullet$ | undefined | $\alpha / (\alpha \backslash \beta)$ |

## 6.2 Two Sided Calculi

Here is the two sided linear calculus $\mathsf{MLL}_+$ for all connectives of the language $\mathsf{Li}_+$. In the Section 6.3.1, we shall see how it embeds the Lambek calculus.

| | | | |
|---|---|---|---|
| Exchange | $\dfrac{\Gamma,A,B,\Delta \vdash \Psi}{\Gamma,B,A,\Delta \vdash \Psi}\,(\mathbf{x})_h$ | | $\dfrac{\Theta \vdash \Gamma,A,B,\Delta}{\Theta \vdash \Gamma,B,A,\Delta}\,(\mathbf{x})_i$ |
| Axiom | $\dfrac{}{A \vdash A}\,ax \qquad A \in \mathsf{Li}_+$ | | |
| Logical rules | $\dfrac{\Gamma \vdash A,\Delta}{A^\perp,\Gamma \vdash \Delta}\,\perp_h$ | Negation | $\dfrac{A,\Gamma \vdash \Delta}{\Gamma \vdash A^\perp,\Delta}\,\perp_i$ |
| | $\dfrac{\Gamma,A \vdash \Theta \qquad B,\Gamma' \vdash \Theta'}{\Gamma,A \wp B,\Gamma' \vdash \Theta,\Theta'}\,\wp_h$ | Disjunction | $\dfrac{\Theta \vdash \Gamma,A,B,\Delta}{\Theta \vdash \Gamma,A \wp B,\Delta}\,\wp_h$ |
| | $\dfrac{\Gamma,A,B,\Delta \vdash \Psi}{\Gamma,A \otimes B,\Delta \vdash \Psi}\,\otimes_h$ | Conjunction | $\dfrac{\Theta \vdash \Phi,A \qquad \Theta' \vdash B,\Phi'}{\Theta,\Theta' \vdash \Phi,A \otimes B,\Phi'}\,\otimes_i$ |
| | $\dfrac{\Gamma \vdash \Phi,A \qquad \Gamma',B,\Delta' \vdash \Psi'}{\Gamma',\Gamma,A \backslash B,\Delta' \vdash \Phi,\Psi'}\,\backslash_h$ <br> $\dfrac{\Gamma \vdash \Phi,A \qquad \Gamma',B,\Delta' \vdash \Psi'}{\Gamma',B / A,\Gamma,\Delta' \vdash \Phi,\Psi'}\,/_h$ | Implications | $\dfrac{A,\Gamma \vdash C,\Phi}{\Gamma \vdash A \backslash C,\Phi}\,\backslash_i$ <br> $\dfrac{\Gamma,A \vdash \Phi,C}{\Gamma \vdash \Phi,C / A}\,/_i$ |

### 6.2.1 Properties of the Linear Two Sided Sequent Calculus

#### Cut Elimination

We left out the cut rule on purpose. There are two ways to formulate the cut rule in a classical calculus:

$$\frac{\Theta \vdash \Phi, A \qquad A, \Theta' \vdash \Psi'}{\Theta, \Theta' \vdash \Phi, \Psi'}\, cut \qquad \frac{\Theta \vdash \Phi, A \qquad \Theta' \vdash A^{\perp}, \Phi'}{\Theta, \Theta' \vdash \Phi, \Phi'}\, cut$$

As in the Lambek calculus, this rule is redundant, and the proof is more or less the same. As a consequence, the subformula property also holds for this calculus.

#### De Morgan Identities and Double Negation Elimination

As we claimed before, these identities hold for linear logic. For instance:

$$\dfrac{\dfrac{A \vdash A}{A^{\perp}, A \vdash}\, \perp_i}{A \vdash (A^{\perp})^{\perp}}\, \perp_i \qquad \dfrac{\dfrac{A \vdash A}{\vdash A^{\perp}, A}\, \perp_i}{(A^{\perp})^{\perp} \vdash A}\, \perp_i$$

#### Restriction to Atomic Axioms

As for the Lambek calculus, an easy induction on $A$, shows that every axiom $A \vdash A$ can be derived from axioms $\alpha \vdash \alpha$, where $\alpha$ is a propositional variable, without using the exchange rule. For instance let us show that $A \vdash A$ with $A = \alpha \wp \beta^{\perp}$ can be derived from the axioms $\alpha \vdash \alpha$ and $\beta \vdash \beta$:

$$\dfrac{\dfrac{\dfrac{}{\alpha \vdash \alpha}\, ax \qquad \dfrac{\dfrac{\dfrac{}{\beta \vdash \beta}\, ax}{\beta, \beta^{\perp} \vdash}\, \perp_h}{\beta^{\perp} \vdash \beta^{\perp}}\, \perp_i}{\alpha \wp \beta^{\perp} \vdash \alpha, \beta^{\perp}}\, \wp_h}{\alpha \wp \beta^{\perp} \vdash \alpha \wp \beta^{\perp}}\, \wp_i$$

#### Equality of the Two Implications

In this calculus, the implication $A \setminus B$ can be viewed as a shorthand for $A^{\perp} \wp B$, while $A / B$ is a shorthand for $B \wp A^{\perp}$. Indeed the rules for the implications can be derived when implications are defined this way. Furthermore, in the presence of a full exchange rule, one has: $A \setminus B \equiv B / A$.

$$\dfrac{\dfrac{\Gamma \vdash A}{\Gamma, A^{\perp} \vdash}\ \perp_h \qquad \Delta', B, \Gamma' \vdash \Theta'}{\Delta', \Gamma, A^{\perp} \wp B, \Gamma' \vdash \Theta'}\ \wp_h \qquad \dfrac{\dfrac{\Gamma, A \vdash B}{\Gamma \vdash A^{\perp}, B}\ \perp_h}{\Gamma \vdash A^{\perp} \wp B}\ \wp_i$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\overline{B \vdash B}\ ax \qquad \overline{A^{\perp} \vdash A^{\perp}}\ ax}{B \wp A^{\perp} \vdash B, A^{\perp}}\ \wp_h}{B \wp A^{\perp} \vdash A^{\perp}, B}\ (\mathbf{x})_i}{B \wp A^{\perp} \vdash A^{\perp} \wp B}\ \wp_i}{B / A \vdash A \backslash B}\ \equiv \qquad \dfrac{\dfrac{\dfrac{\dfrac{\overline{A^{\perp} \vdash A^{\perp}}\ ax \qquad \overline{B \vdash B}\ ax}{A^{\perp} \wp B \vdash A^{\perp}, B}\ \wp_h}{A^{\perp} \wp B \vdash B, A^{\perp}}\ (\mathbf{x})_i}{A^{\perp} \wp B \vdash B \wp A^{\perp}}\ \wp_i}{A \backslash B \vdash B / A}\ \equiv$$

**Negation and Symmetrical Rules**

If one considers formulae up to De Morgan identities, then right rules are enough.

For instance the rule $\wp_h$ can be simulated by the rule $\otimes_i$ as shown in the following derivation.

$$\dfrac{\dfrac{\dfrac{\dfrac{\Gamma, A \vdash \Theta}{\Gamma \vdash A^{\perp}, \Theta}\ \perp_i \qquad \dfrac{B, \Gamma' \vdash \Theta'}{\Gamma' \vdash B^{\perp}, \Theta'}\ \perp_i}{\Gamma, \Gamma' \vdash \Theta, A^{\perp} \otimes B^{\perp}, \Theta'}\ \otimes_i}{[A \wp B \equiv (A^{\perp} \otimes B^{\perp})^{\perp}], \Gamma', \Gamma \vdash \Theta', \Theta}\ (\mathbf{x})_h \text{ and } \perp_h}{\Gamma, [B \wp A \equiv (A^{\perp} \otimes B^{\perp})^{\perp}], \Gamma' \vdash \Theta', \Theta}\ (\mathbf{x})_h$$

In order to avoid the exchange rule, one has to consider a more subtle sequent calculus like the one of (Abrusci, 1991, p. 1415) but identifying the two negations — this actually forces a restricted form of the exchange rule known as cyclic exchange, that we shall present later on.

### 6.2.2 The Intuitionistic Two Sided Calculus $\mathsf{LP}_\varepsilon$

The calculus $\mathsf{LP}_\varepsilon$, that is Lambek calculus with permutation and empty antecedents is exactly intuitionistic multiplicative linear logic. This calculus is obtained from $\mathsf{MLL}_+$ by forcing sequents to always have exactly one formula on the right hand side.

By inspection of the rules, it is clear that restricting the right hand side of the sequent to one formula means that we can no longer formulate the rules for negation. Therefore the natural language for $\mathsf{LP}_\varepsilon$ is $\mathsf{Lp}$. The rules are obtained from the ones of $\mathsf{MLL}_+$ in Section 6.2, by replacing the sequences of formulae denoted by $\Phi$ and $\Phi'$ by the empty sequence, and the sequences of formulae denoted by $\Psi$ and $\Psi'$ by a single formula $F$ or $F'$. This yields the following rules:

| | | |
|---|---|---|
| Exchange | $\dfrac{\Gamma,A,B,\Delta\vdash F}{\Gamma,B,A,\Delta\vdash F}\,(\mathbf{x})_h$ | |
| Axiom | $\dfrac{}{A\vdash A}\,ax \qquad A\in \mathsf{Lp}$ | |
| Logical rules | $\dfrac{\Gamma,A,B,\Delta\vdash F}{\Gamma,A\otimes B,\Delta\vdash F}\,\otimes_h$ Conjunction $\dfrac{\Theta\vdash A \qquad \Theta'\vdash B}{\Theta,\Theta'\vdash A\otimes B}\,\otimes_h$ | |
| | $\dfrac{\Gamma\vdash A \qquad \Gamma',B,\Delta'\vdash F'}{\Gamma',\Gamma,A\backslash B,\Delta'\vdash F'}\,\backslash_h$ | $\dfrac{A,\Gamma\vdash C}{\Gamma\vdash A\backslash C}\,\backslash_i$ |
| | Implications | |
| | $\dfrac{\Gamma\vdash A \qquad \Gamma',B,\Delta'\vdash F'}{\Gamma',B/A,\Gamma,\Delta'\vdash F'}\,/_h$ | $\dfrac{\Gamma,A\vdash C}{\Gamma\vdash C/A}\,/_i$ |

This calculus $\mathsf{LP}_\varepsilon$ and its variants are studied in a slightly different perspective in (van Benthem, 1991), and is also the basis of works on the semantics of LFG in a series of articles like (Dalrymple et al, 1995).

This calculus allows for several variants according to the presence or absence of the exchange rule, or the allowance or prohibition of sequents with an empty antecedent, that is: the sequence of formulae $\Pi$ is not empty when the rule $\backslash_i$ or $/_i$ is applied or, equivalently, every sequent in a proof has a non empty antecedent.

This last restriction is harmless from a logical viewpoint, i.e. preserves cut-elimination, but is essential for a grammatical use of the Lambek calculus, as we have seen in Section 2.5. Let us give another example of an incorrect analysis due to empty antecedents:

*Example 6.4.* Look at the following small lexicon.

| Word | Type(s) | Translation |
|---|---|---|
| *exemple* | $n$ | *example* |
| *simple* | $n\backslash n$ | *simple* |
| *très* | $(n\backslash n)/(n\backslash n)$ | *very* |
| *un* | $np/n$ | *a* |

$$\dfrac{\dfrac{\dfrac{}{n\vdash n}\,ax}{\boxed{\varepsilon}\vdash n/n}\,/_i \qquad \dfrac{\dfrac{\dfrac{}{n\vdash n}\,ax \qquad \dfrac{}{np\vdash np}\,ax}{np/n,n\vdash np}\,/_h \qquad \dfrac{}{n\vdash n}\,ax}{np/n,n,n\backslash n\vdash np}\,\backslash_h}{np/n,\quad n,\quad (n\backslash n)/(n\backslash n)\vdash np}\,/_h$$

`un exemple très`

### 6.2.3 Proofs as Parse Structures: Too Many of Them

When we look at parsing a Lambek grammar, then, given that the Lambek calculus is a logic, a parse for a Lambek grammar is a proof in the Lambek calculus. However, sequent calculus proof search, which is one way of implementing parsing for the Lambek calculus is problematic: it is easy to find several proofs which should correspond to the same parse structure, but which nevertheless are distinct. For instance, with the previous lexicon, the following sequent calculus proofs are different

*Example 6.5*

$$\dfrac{\dfrac{\dfrac{\dfrac{}{n \vdash n}\,ax \quad \dfrac{}{n \vdash n}\,ax}{n, n \backslash n \vdash n}\,\backslash_h}{n \backslash n \vdash n \backslash n}\,\backslash_i \qquad \dfrac{\dfrac{\dfrac{}{n \vdash n}\,ax \quad \dfrac{}{n \vdash n}\,ax}{n, n \backslash n \vdash n}\,\backslash_h \quad \dfrac{}{np \vdash np}\,ax}{np / n, n, n \backslash n \vdash np}\,/_h}{\underset{\texttt{un}}{np / n}, \quad \underset{\texttt{exemple}}{n}, \quad \underset{\texttt{très}}{(n \backslash n) / (n \backslash n)}, \quad \underset{\texttt{simple}}{n \backslash n} \quad \vdash np}\,/_h$$

*Example 6.6*

$$\dfrac{\dfrac{\dfrac{\dfrac{}{n \vdash n}\,ax \quad \dfrac{}{n \vdash n}\,ax}{n, n \backslash n \vdash n}\,\backslash_h}{n \backslash n \vdash n \backslash n}\,\backslash_i \qquad \dfrac{\dfrac{\dfrac{}{n \vdash n}\,ax \quad \dfrac{}{np \vdash np}\,ax}{np/n, n \vdash np}\,/_h \quad \dfrac{}{n \vdash n}\,ax}{np / n, n, n \backslash n \vdash np}\,\backslash_h}{\underset{\texttt{un}}{np / n}, \quad \underset{\texttt{exemple}}{n}, \quad \underset{\texttt{très}}{(n \backslash n) / (n \backslash n)}, \quad \underset{\texttt{simple}}{n \backslash n} \quad \vdash np}\,/_h$$

In the two proofs above the order of the $\backslash_h$ and $/_h$ rules is reversed, but both rules have the same formula occurrences as their active and main formulae; the only way the two proofs differ is in the way the context variables of the rules are instantiated.

This problem, that there can be many proofs of what we would want to be the same *parse* is sometimes called the *spurious ambiguity problem*. Natural deduction is a bit better in this respect, though, as we have seen in Section 2.6.3 problems of multiple derivations corresponding to a single parse exist for the product formula. One of the main objective of this chapter is to find a notion of proof that yields one proof per parse structure; this is a key motivation for proof nets, to be introduced in Section 6.4: proof nets will solve the problem of multiple equivalent proofs which exists for the sequent calculus and, unlike natural deduction, will treat the product formulae as easily as the other connectives.

## 6.3 A One Sided Calculus for Linear Logic: MLL

As we have seen in the paragraph 6.1.2 for every formula $X$ of $\mathsf{Li}_+$ there exists a unique formula $+X$ of $\mathsf{Li}$ which is equivalent to it by De Morgan identities, and as explained in paragraph 6.2.1, right rules can be simulated by left rules. Therefore, if one considers formulae up to De Morgan identities then the following one sided sequent calculus, defined as follows, is enough:

| | | |
|---|---|---|
| Exchange | $\dfrac{\vdash \Gamma, A}{\vdash A, \Gamma}\ (\mathsf{cx})$ | $\dfrac{\vdash \Gamma, A, B}{\vdash \Gamma, B, A}\ (\mathsf{tx})$ |
| Axiom | $\dfrac{}{\vdash \alpha, \alpha^{\perp}}\ ax$ | $\alpha \in \mathsf{P}$ |
| Logical rules | $\dfrac{\vdash \Gamma, A, B, \Delta}{\vdash \Gamma, A \wp B, \Delta}\ \wp$ | $\dfrac{\vdash \Gamma, A \qquad \vdash B, \Gamma'}{\vdash \Gamma, A \otimes B, \Gamma'}\ \otimes$ |

The exchange rule $(\mathbf{x})_h$ of $\mathsf{MLL}_+$ has been split into two rules $(\mathsf{tx})$ (transposition exchange) and $(\mathsf{cx})$ (cyclic exchange). Therefore $(\mathbf{x})_h$ is derivable but, this formulation allows to consider the calculus $\mathsf{NC\text{-}MLL}$ of (Yetter, 1990), which only has the $(\mathsf{cx})$ exchange, but not the $(\mathsf{tx})$ exchange.

The simple calculus $\mathsf{MLL}$ whose language is $\mathsf{Li}$, proves exactly the same sequents as the bigger two sided calculus $\mathsf{MLL}_+$ :

**Proposition 6.7.** *Let $A_1, \ldots, A_n, B_1, \ldots, B_p$ be formulae in $\mathsf{Li}_+$; then one has:*

$$(A_1, \ldots, A_n \vdash_{MLL_+} B_1, \ldots, B_p) \Leftrightarrow (\vdash_{MLL} -A_n, \ldots, -A_1, +B_1, \ldots, +B_p)$$

*For the converse implication, notice that given a formula $F \in \mathsf{Li}$ there usually exist several formulae $X \in \mathsf{Li}_+$ such that $+X = F$ or $-X = F$.*

### 6.3.1 Variants

We are about to introduce several variants of MLL according to the following restrictions:

INTUI **intuitionistic calculi**
- in two sided presentation: one formula in the right hand side of every sequent
- in one sided presentation: only polarized formulae (formulae of $\mathsf{Li}^{\circ} \cup \mathsf{Li}^{\bullet}$)[2]

NC **non commutative calculi**
- in two sided presentation: no exchange at all
- in one sided presentation: cyclic exchange $(\mathsf{cx})$ only (no transposition exchange $(\mathsf{tx})$)

[2] Note that by Proposition 6.8 of the next section, we do not have to require explicitly that there is only one formula in $\mathsf{Li}^{\circ}$.

$\varepsilon$-FREE **no empty antecedent**

in two sided presentation: no empty antecedent, at least one formula on the left hand side of every sequent

in one sided presentation: at least two formulae in every sequent

The names for these calculi somehow differ in the categorial tradition and in the linear logic community, for instance, calculi without empty antecedents are never considered in linear logic and, though classical calculi are sometimes discussed in the categorial tradition (see, for example, Lambek, 1993; de Groote and Lamarche, 2002), there are, to the best of our knowledge, no linguistic applications of formulas *not* in $\mathsf{Li}^{\circ} \cup \mathsf{Li}^{\bullet}$. For linear calculi, the restriction which corresponds to forbidding empty antecedents will be denoted by $(\cdots)^{*}$. Conversely, for categorial grammar and Lambek calculus, allowing for empty antecedents will be denoted by $(\cdots)^{\varepsilon}$. The non-commutative restriction of a linear calculus will be denoted by a prefix NC, and the commutative extension of a Lambek style calculus will be denoted by a suffix P

Because of these two communities, we have two names for the intuitionistic calculi, and we hope it will not confuse the reader. Table 6.1 lists all the different systems, together with their different names and the restrictions which apply to them. Figure 6.1 portrays the relations between the logics by means of a commutative diagram. All these restrictions will appear again for describing the proof nets corresponding to each calculus.

Although this might be surprising we are able to provide a one sided formulation for intuitionistic calculi. So we will use the *linear name* $\cdots$ *MLL for one sided calculi* and the *categorial name L*$\cdots$ *for two sided calculi*.

**Table 6.1.** The different logical systems and their properties

| INTUI | NC | $\varepsilon$-FREE | Linear name | Categorial name |
|---|---|---|---|---|
| yes | yes | yes | NC-IMLL* | L |
| yes | yes | no | NC-IMLL | $\mathsf{L}_{\varepsilon}$ |
| yes | no | yes | IMLL* | LP |
| yes | no | no | IMLL | $\mathsf{LP}_{\varepsilon}$ |
| no | yes | yes | NC-MLL* | |
| no | yes | no | NC-MLL | |
| no | no | yes | MLL* | |
| no | no | no | MLL | |
| | | | one sided | two sided |

### 6.3.2 The Intuitionistic Restriction in One Sided Calculi

The two sided intuitionistic calculus $\mathsf{LP}_{\varepsilon}$ is a proper restriction of its classical counterpart MLL. For instance, if we look at the formula $F = (\beta \wp \alpha) \wp (\alpha^{\perp} \otimes \beta^{\perp})$ one

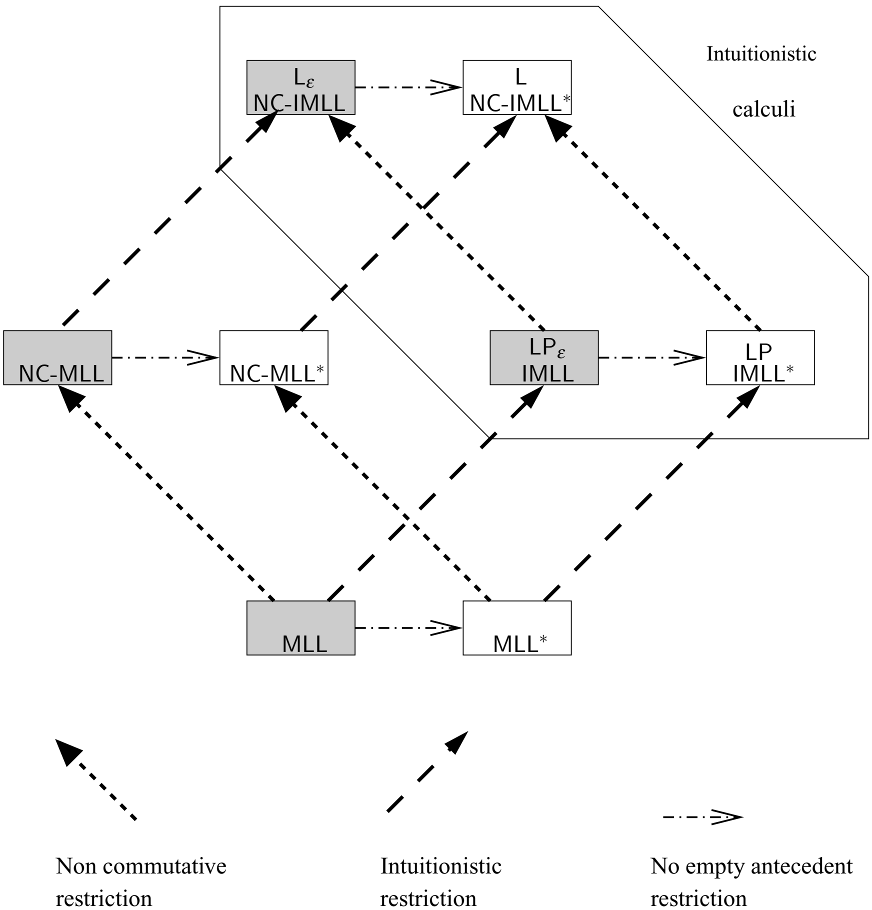



**Fig. 6.1.** Commutative diagram listing the relations between the different logics

has $\vdash_{\mathsf{MLL}} F$ but there is no formula $G$ equivalent to $F$ such that $\vdash_{\mathsf{LP}_\varepsilon} G$. Actually, this restriction is only a restriction of the *language*, which we have already studied in Section 6.1.3. Indeed, it is only because there is no formula in $\mathsf{Lp}$ equivalent to $F$, i.e. because $F \notin \mathsf{Li}^\bullet \cup \mathsf{Li}^\circ$ that $F$ is not a theorem of $\mathsf{IMLL}$. More precisely we have the following result.

**Proposition 6.8.** *If* $\forall i \in [1,n]\ A_i \in \mathsf{Li}^\bullet \cup \mathsf{Li}^\circ$ *then*

$$(\vdash_{MLL} A_1,\ldots,A_n) \Leftrightarrow (\vdash_{IMLL} A_1,\ldots,A_n)$$

*and whenever these properties hold, then exactly one formula of the sequent is in* $\mathsf{Li}^\circ$*, all others being in* $\mathsf{Li}^\bullet$*. This also holds for the variants* NC-MLL *and* NC-IMLL*.*

*Proof.* Easy induction on the proofs. □

Proposition 6.8 was first studied by van de Wiele in the typed case and then taken up by Bellin and Scott (1994) and by Danos and Regnier (Danos, 1990; Regnier, 1992) in the untyped case. This property has lead Lamarche to an interesting theory of intuitionistic proof nets (Lamarche, 1994) which is orthogonal to our presentation.

From the previous proposition we easily deduce the correspondence between one sided intuitionistic calculi and the two sided intuitionistic calculi:

**Proposition 6.9.** *If* $\vdash_{MLL} F_1,\ldots,F_n$*, with* $\forall i \in [1,n]\ F_i \in \mathsf{Li}^{\bullet} \cup \mathsf{Li}^{\circ}$*, then:*

- *there exists a unique index* $i_0 \in [1,n]$ *such that* $F_{i_0} \in \mathsf{Li}^{\circ}$ *and for every other index* $i \in [1,n]$ *we have* $F_i \in \mathsf{Li}^{\bullet}$ *because of the Proposition 6.8*
- *because of Section 6.1.3, every formula* $F_i^{\perp}$ *with* $i \neq i_0$ *is equivalent to a unique formula* $(F_i)^{\bullet}_{\mathsf{Lp}} \in \mathsf{Lp}$*, while* $F_{i_0}$ *is equivalent to a unique formula* $(F_{i_0})^{\circ}_{\mathsf{Lp}}$
- $(F_{i_0-1})^{\bullet}_{\mathsf{Lp}}, (F_{i_0-2})^{\bullet}_{\mathsf{Lp}}, \ldots, (F_1)^{\bullet}_{\mathsf{Lp}}, (F_n)^{\bullet}_{\mathsf{Lp}}, \ldots, (F_{i_0+1})^{\bullet}_{L} \vdash_{LP_\varepsilon} (F_{i_0})^{\circ}_{\mathsf{Lp}}$

*Conversely,* $(X_1,\ldots,X_n \vdash_{LP_\varepsilon} Y) \Rightarrow (\vdash_{MLL,IMLL} -X_n,\ldots,-X_1,+Y)$*.*

*If one replaces* MLL *with* NC-MLL *(resp.* NC-MLL* *) and* $LP_\varepsilon$ *with* $L_\varepsilon$ *(resp.* L*) the result also holds (As announced in the commutative diagram of Figure 6.1, the restrictions* INTUI,NC *and* ε-FREE *commute).*

*For these non commutative variants* NC-MLL*,* NC-MLL**,* $L_\varepsilon$ *and* L*, with a restricted exchange rule, one has to abide by the order between formulae: this order is reversed when formulae move from one side of the sequent's turnstile to the other.*

*Proof.* The "conversely" is obvious.

The direct implication is shown by induction on the proof. For the proof to work in the non commutative case, the rule (tx) is only used for the translation of the $(\mathbf{x})_h$ rule of IMLL. Here is, for instance, the translation of the $/_h$.

Assume that the sequences of formulae involved in $/_h$ are $\Gamma = G_1,\ldots,G_n$, $\Gamma' = G'_1,\ldots,G'_k$, $\Delta' = D'_1,\ldots,D'_l$. Here is the NC-MLL proof which simulates the rule $/_h$ of $\mathsf{L}_\varepsilon$ — remember that $+A \otimes -B = -(A \setminus B)$ (c.f. Section 6.1.3):

$$\dfrac{\dfrac{\dfrac{\vdash -G_n,\ldots,-G_1,+A \qquad \dfrac{\vdash -D'_l,\ldots,-D'_1,-B,-G'_k,\ldots,-G'_1,+C'}{\vdash -B,-G'_k,\ldots,-G'_1,+C',-D'_l,\ldots,-D'_1}\ l(EC)}{\vdash -G_n,\ldots,-G_1,+A\otimes -B,-G'_k,\ldots,-G'_1,+C',-D'_l,\ldots,-D'_1}\ \otimes}{\vdash -D'_l,\ldots,-D'_1,-G_n,\ldots,-G_1,+A\otimes -B,-G'_k,\ldots,-G'_1,+C'}\ l(EC)}{\vdash -D'_l,\ldots,-D'_1,-G_n,\ldots,-G_1,-(A\setminus B),-G'_k,\ldots,-G'_1,+C'}\ =$$

□

Let us provide the NC-MLL translation of the proofs or parse structures given in examples 6.5 and 6.6 :

*Example 6.10*

$$
\dfrac{
\dfrac{
\dfrac{
\dfrac{
\dfrac{\dfrac{}{\vdash n, n^{\perp}}\,ax \quad \dfrac{}{\vdash n, n^{\perp}}\,ax}{\vdash n, n^{\perp}\otimes n, n^{\perp}}\,\otimes
}{\vdash n^{\perp}\otimes n, n^{\perp}, n}\,CX
}{\vdash n^{\perp}\otimes n, n^{\perp}\wp n}\,\wp
\quad
\dfrac{
\dfrac{\dfrac{}{\vdash n, n^{\perp}}\,ax \quad \dfrac{}{\vdash n, n^{\perp}}\,ax}{\vdash n, n^{\perp}\otimes n, n^{\perp}}\,\otimes
}{\vdash n^{\perp}\otimes n, n^{\perp}, n}\,CX
}{\vdash n^{\perp}\otimes n, (n^{\perp}\wp n)\otimes(n^{\perp}\otimes n), n^{\perp}, n}\,\otimes
\quad
\dfrac{}{\vdash np^{\perp}, np}\,ax
}{\vdash \underset{n\backslash n}{\underset{\texttt{simple}}{n^{\perp}\otimes n}},\; \underset{(n\backslash n)/(n\backslash n)}{\underset{\texttt{très}}{(n^{\perp}\wp n)\otimes(n^{\perp}\otimes n)}},\; \underset{n}{\underset{\texttt{exemple}}{n^{\perp}}},\; \underset{np/n}{\underset{\texttt{un}}{n\otimes np^{\perp}}},\; np}\,\otimes
$$

*Example 6.11*

$$
\dfrac{
\dfrac{
\dfrac{
\dfrac{\dfrac{}{\vdash n, n^{\perp}}\,ax \quad \dfrac{}{\vdash n, n^{\perp}}\,ax}{\vdash n, n^{\perp}\otimes n, n^{\perp}}\,\otimes
}{\vdash n^{\perp}\otimes n, n^{\perp}, n}\,CX
}{\vdash n^{\perp}\otimes n, n^{\perp}\wp n}\,\wp
\quad
\dfrac{
\dfrac{
\dfrac{
\dfrac{\dfrac{}{\vdash n, n^{\perp}}\,ax \quad \dfrac{}{\vdash np, np^{\perp}}\,ax}{\vdash n^{\perp}, n\otimes np^{\perp}, np}\,\otimes
}{\vdash n\otimes np^{\perp}, np, n^{\perp}}\,CX
\quad
\dfrac{}{\vdash n, n^{\perp}}\,ax
}{\vdash n\otimes np^{\perp}, np, n^{\perp}\otimes n, n^{\perp}}\,\otimes
}{\vdash n^{\perp}\otimes n, n^{\perp}, n\otimes np^{\perp}, np}\,2\times CX
}{\vdash \underset{n\backslash n}{\underset{\texttt{simple}}{n^{\perp}\otimes n}},\; \underset{(n\backslash n)/(n\backslash n)}{\underset{\texttt{très}}{(n^{\perp}\wp n)\otimes(n^{\perp}\otimes n)}},\; \underset{n}{\underset{\texttt{exemple}}{n^{\perp}}},\; \underset{np/n}{\underset{\texttt{un}}{n\otimes np^{\perp}}},\; np}\,\otimes
$$

## 6.4 Proof Nets: Concise and Expressive Proofs

We now turn our attention to proof nets; they are for linear logic what natural deductions (or typed lambda terms) are for intuitionistic logic, in the sense that the contexts are not copied at each step of the proof.

From a logical viewpoint, they are much more compact than sequent calculus proofs: well-formedness is a global condition but easy (and fast) to verify, and cut-elimination is a local and efficient process. But their main advantage is that they are a better representation of proofs. Indeed, many sequent calculus proofs which only differ in the order of application of the rules convert to the same proof net. For example, the two proofs given in the Examples 6.10 and 6.11 will yield the same proof net. It should be noticed that when these proofs are viewed as a representation of syntactic analyses in the Lambek calculus (they correspond to the parses of Examples 6.5 and 6.6. in a Lambek grammar), they both describe the same linguistic analysis, so it is really a good feature of proof nets that we are able to describe this analysis by a single object.

### 6.4.1 Proof Nets for MLL

#### R&B Graphs

A *matching* in a graph is a subset of the set of edges such that no two edges of the matching are adjacent. The matching is said to be *perfect* whenever each vertex of the graph is incident to an edge of the matching – because it is a matching, each vertex is incident to exactly one edge of the matching.

**Definition 6.12 (R&B graphs).** *A R&B graph is an edge colored graph, whose edges either are of color B (blue or bold), or R (red or regular), such that the B edges define a perfect matching of the graph.*

B edges correspond to formulae and R edges to connectives. The recognition, among all such graphs, of the ones which are proofs, will involve the notion of alternate elementary path.

**Definition 6.13 (æ paths and cycles).** *An æ path in a R&B graph is an alternating elementary path, that is a path the edges of which are alternatively in B and in R which does not use twice the same edge — as B edges are a matching, this is equivalent to the property that the path does not contain the same vertex twice (except, possibly the first and last vertices that might be the same). More precisely, an æ path is a finite sequence of edges $(a_i)_{i\in[1,n]}$ such that:*

$$i \neq j \Longrightarrow a_i \neq a_j \qquad \#(a_i \cap a_{i+1}) = 1$$
$$a_i \in B \Longrightarrow a_{i+1} \in R \qquad a_i \in R \Longrightarrow a_{i+1} \in B$$

*An æ cycle is an æ path of even length, whose end vertices are equal.*

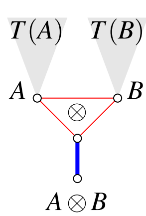
#### Prenets

**Definition 6.14 (Prenets or proof structures, links).** *Prenets are R&B graphs built from basic R&B graphs called links, which are shown in Figure 6.2 (where $\alpha$ denotes an atomic formula) in such a way that each formula is the conclusion of exactly one link and the premise of at most one link. Formulae that are not the premise of a link are called conclusions of the prenet.*

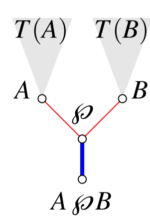
**Definition 6.15 (R&B subformula tree).** *Given a formula C, its* R&B subformula tree *$T(C)$ is a R&B graph defined inductively as follows.*

- *If $C = \alpha$ is a propositional variable then $T(C)$ is:* $\overset{\circ}{\alpha}$
- *given $T(A)$ and $T(B)$, $T(A \otimes B)$ and $T(A \wp B)$ are defined as follows.*

$T(A \otimes B)$ :




$T(A \wp B)$ :



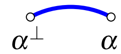
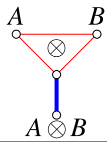
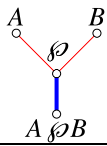
| Links | | | |
|---|---|---|---|
| Name | Graph | Premises | Conclusions |
| Axiom |  | none | $\alpha$ and $\alpha^\perp$ |
| Times |  | $A$ and $B$ | $A \otimes B$ |
| Par |  | $A$ and $B$ | $A \wp B$ |

**Fig. 6.2.** Links for constructing prenets

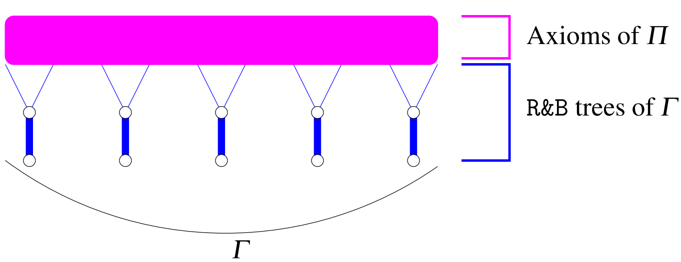
Beware that the R&B subformula tree of a formula $C$ is not, from a graph theoretical point of view, a tree: indeed, every *Times* link contains a cycle. We nevertheless chose this name because it is very similar to the subformula tree, and because of the fact that w.r.t. the æ paths, the only paths we shall use, the R&B subformulae trees are acyclic.

The vertices corresponding to propositional variables in a subformula tree will be called leaves of the subformula tree.

**Definition 6.16 (prenet with conclusions $\Gamma$).** *Given a sequence of formulae $\Gamma$, a prenet $\Pi$ with conclusions $\Gamma$ consists of:*

- *the R&B subformula trees of the formulae in $\Gamma$*
- *a set of B edges joining dual leaves, called axioms, such that each leaf is incident to exactly one axiom.*

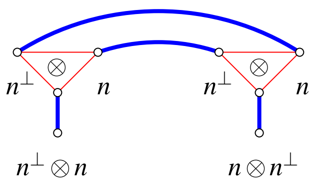
The structure of a prenet is the following.



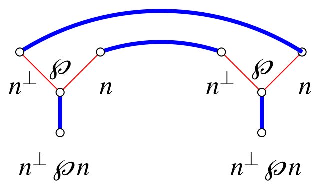

Notice that the order between formulae of $\Gamma$ or their subformula trees is not part of the structure, but because of the labeling of the vertices, R&B subformula trees make a distinction between their right and left subtrees.

The examples below — Example 6.17 to 6.23 — give some examples of prenets. Note that not all of these prenets correspond to sequent proofs: we will see how to distinguish the correct prenets, the *proof nets*, from the other prenets below.

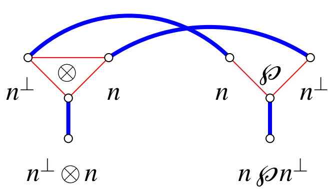
*Example 6.17*



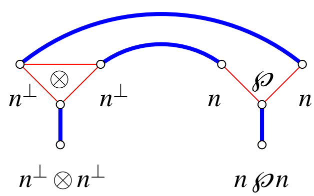

*Example 6.18*



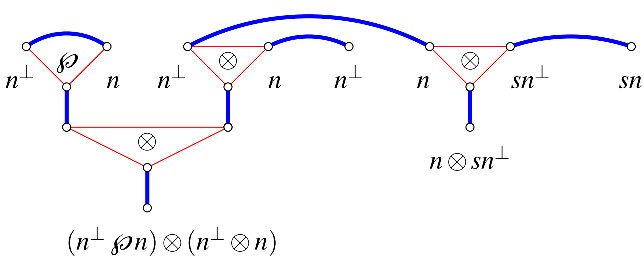

*Example 6.19*



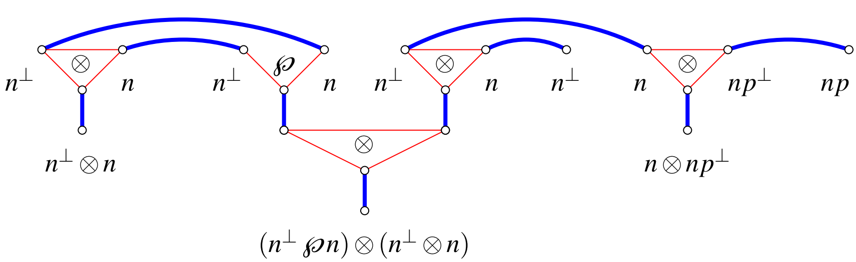

*Example 6.20*



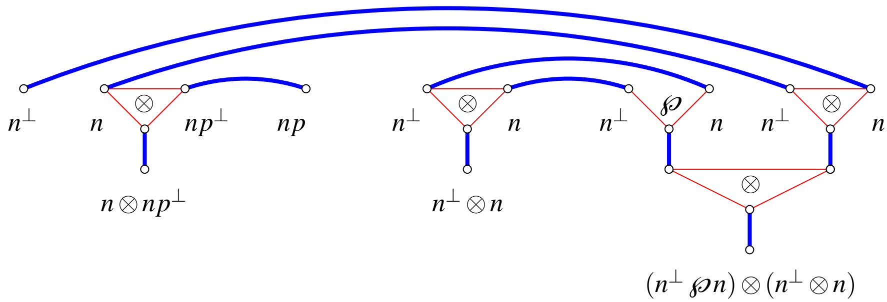

*Example 6.21*




*Example 6.22*



*Example 6.23*



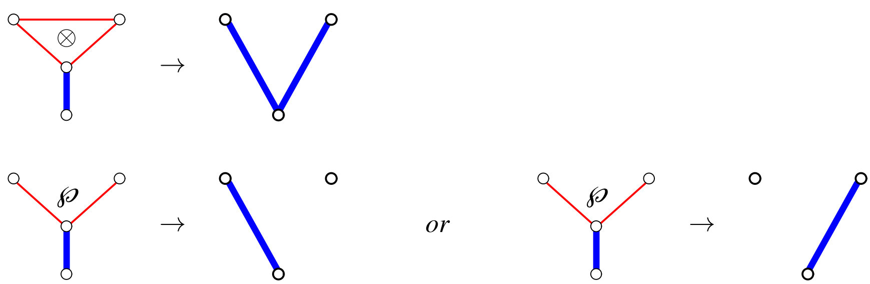

**Proof Nets**

**Definition 6.24 (proof net).** *A* proof net *is a prenet satisfying the following properties:*

ØÆ *there is no æ cycle.*
SAT *there exists an æ path between any two vertices.*

To facilitate a comparison with the well-known presentation of proof nets according to (Danos and Regnier, 1989; Girard, 1995), we will introduce the Danos-Regnier correctness condition, which is stated using correction graphs of prenets, defined as follows.

**Definition 6.25 (correction graph).** *From a prenet we obtain a* correction graph *by rewriting the logical links as follows.*




*Note that there are two ways of rewriting the par links, which means that for a prenet with p par links there are* $2^p$ *correction graphs. In addition, correction graphs only have a single type of edges (all edges are* B *edges) so correction graphs really* are *graphs (ie. a set of vertices and a set of edges connecting these vertices).*

**Definition 6.26 (Danos and Regnier (1989)).** *A prenet is a* proof net *iff all its correction graphs are acyclic and connected.*

Compared to the Danos and Regnier presentation of proof nets, the property ØÆ corresponds to the acyclicity of all correction graphs and the property SAT to their connectedness (see Fleury and Retoré, 1994; Retoré, 1996). The advantage of the current representation of proof nets is that the correctness condition can be verified by inspection of only a single graph.

The following result of (Retoré, 1996; Retoré, 2003) shows that verifying the correctness of prenets is rather easy from an algorithmic point of view — recently some linear algorithms have been provided on the Danos-Regnier presentation of proof nets, and they certainly can be adapted to our formalism (Guerrini, 1999, 2011; Murawski and Ong, 2000).

**Proposition 6.27.** *Given a prenet with $n$ vertices, their exists an algorithm which decides in $n^2$ steps whether the prenet is a proof net.*

Among the examples of prenets given above, only 6.19, 6.20, 6.21, 6.22 and 6.23 are proof nets. The prenet 6.17 contains an æ cycle, and the prenet 6.18 does not contain any æ path between the left most leaves $n^\perp$ and $n$.

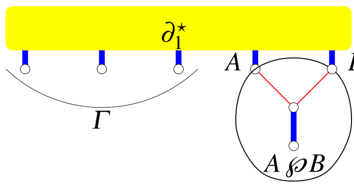
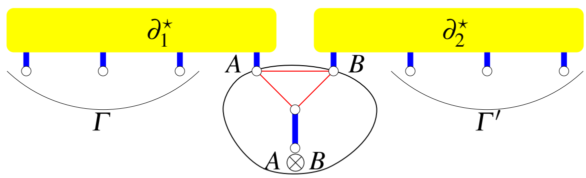
### 6.4.2 Sequent Calculus and Proof Nets

The following proposition gives a precise account of the correspondence between proof nets and sequent calculus proofs, and its proofs shows how sequent calculus proofs are mapped onto proof nets. The converse correspondence relies on graph theoretical properties, and we refer the reader to (Retoré, 1996; Retoré, 2003).

**Theorem 6.28.** *Every sequent calculus proof in MLL of a sequent $\vdash A_1,\ldots,A_n$ translates into a proof net with conclusions $A_1,\ldots,A_n$. Conversely, every proof net with conclusions $A_1,\ldots,A_n$ corresponds to at least one sequent calculus proof in MLL of $\vdash A_1,\ldots,A_n$ in NC-MLL — every such proof is called a sequentialisation of the proof net.*

*Proof.* As said above, we limit ourselves to the first part of this statement.

The translation from sequent calculus proofs to proof nets is defined inductively. As the exchange rule has no effect on proof nets, since for the time being we have no order on the conclusions, we simply skip it. The effect of this rule would be to produce crossings of axiom links, but up to now this is not part of our description of a proof net. For instance, the Examples 6.22 and 6.23 shown above are considered to be the same proof net: the three rightmost conclusions of Example 6.22 ($n^\perp$, $n \otimes np^\perp$ and $np$) are the three leftmost conclusions of Example 6.23 but they are connected in exactly the same way both to each other and to the rest of the prenet.

| Proof $\partial$ in MLL | Corresponding proof net $\partial^\star$ |
|---|---|
| $\dfrac{}{\vdash \alpha^\perp, \alpha}\,ax$ |  |
| $\dfrac{\begin{matrix}\vdots\ \partial_1\\ \vdash \Gamma, A, B\end{matrix}}{\vdash \Gamma, A \wp B}\,\wp$ |  |
| $\dfrac{\begin{matrix}\vdots\ \partial_1\\ \vdash \Gamma, A\end{matrix} \quad \begin{matrix}\vdots\ \partial_2\\ \vdash B, \Gamma'\end{matrix}}{\vdash \Gamma, A \otimes B, \Gamma'}\,\otimes$ |  |

It is easily checked by induction that the prenet corresponding to a sequent calculus proof are proof nets: no æ cycle can appear during the construction, and the fact that there always exists an æ path between any two vertices is also preserved during the construction. □

Using this inductive definition, the proofs of Example 6.10 and 6.11, both yield the proof net of Example 6.22, so a single proof net corresponds to a single parse structure.

Rules and links are in a one-to-one correspondence (that is, *ax* with *Axiom*, $\wp$ with *Par* and $\otimes$ with *Tensor*), and the last logical rule in the sequent calculus proof correspond to a final link in the prenet — a link which is the root of one of the subformula trees — while the converse does not hold. We nevertheless have the following property, that will be useful later on:

**Proposition 6.29.** *Let $\Pi$ be a proof net such that:*

- *all conclusions of $\Pi$ are the conclusions of Times or Axioms links*
- *there is at least one Times link, that is $\Pi$ is not a single Axiom*

*then at least one of the final Times links is* splitting, *that is each of the two premise* B *edges is a bridge — an edge the suppression of which increases the number of connected components.*

*Proof.* As we have a proof net, at least one sequent calculus proof translates into it. The final rule of the sequent calculus correspond to a final link, so is a *Times* link. From the translation given above, both the premise B edges of this link are bridges of the graph. □

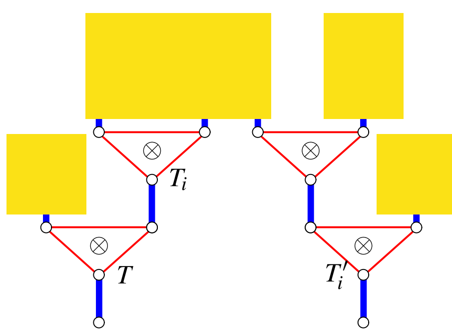
Observe that not all final *Times* links are splitting. For instance in the example 6.22 the final *Times* $n^{\perp} \otimes n$ is not splitting, and can not be the translation of the final rule of a corresponding sequent calculus proof. The final *Times* links $(n^{\perp} \wp n) \otimes (n^{\perp} \otimes n)$ and $n \otimes np^{\perp}$ are splitting *Times* links, and this is supported by the sequentialisations given in examples 6.10 and 6.11.

We can generalize the notion of splitting *Times* link to a *hereditary* splitting *Times* link as follows (Retoré, 1993).

**Proposition 6.30.** *Let $\Pi$ be a proof net and, as in Proposition 6.29, let all conclusions of $\Pi$ be the conclusions of Times and Axiom links with the number of Times links being at least one. $\Pi$ has a* hereditary splitting *Times link $T$; that is*

- *$T$ is a splitting Times link, and therefore removing $T$ from $\Pi$ splits the proof net into two proof nets $\Pi_1$ and $\Pi_2$*
- *For each of the premises $P_1$ and $P_2$ of $T$, if $P_i$ is the conclusion of a tensor link $T_i$ then $T_i$ is a hereditary splitting Times link in $\Pi_i$. Note that, since we know for both $\Pi_1$ and $\Pi_2$ that all conclusions are either the conclusions of Axiom or of Tensor links, it makes sense to talk about hereditary splitting Tensor links of these subnets.*

*Proof.* First, we remark that if one of the $\Pi_i$ has a hereditary splitting *Times* link $T_i' \neq T_i$, then $T_i'$ is a hereditary splitting *Times* link of $\Pi$. For suppose $T_i'$ were not a hereditary splitting *Times* link of $\Pi$, this would mean that there would be a path from two of the "leaves" of the tensor tree with $T_i'$ as its root passing through $T$ which contradicts $T$ being a splitting tensor link. The figure below illustrates the situation. Note that $T$ is a splitting tensor but (because of $T_i$) not a hereditary splitting tensor although $T_i'$ is a hereditary splitting tensor.




We proceed by induction on the number of tensor links in the proof net. Let $T$ be a splitting tensor link of $\Pi$.

If none of the premises of $T$ is the conclusion of a *Times* link, then $T$ is hereditary splitting and we are done.

If one of the two premises of $T$, say $P_1$ is the conclusion of a *Times* link $T_1$ then, by induction hypothesis $T_1$ has a hereditary splitting *Times* link $T_1'$. If $T_1' \neq T_1$ then $T_1'$ is a hereditary splitting *Times* link of $\Pi$ according to the remark at the start of the proof. Otherwise $T_1$ is a hereditary splitting *Times* link in $\Pi_1$. We therefore look at the other premise $P_2$ of $T$. If it is not the conclusion of a tensor link, then we are done. However, if it is the conclusion of a *Times* link $T_2$ we proceed as before: we know by induction hypothesis that $\Pi_2$ has a hereditary splitting *Times* link $T_2'$. If $T_2'$ is not equal to $T_2$ then $T_2'$ is a hereditary splitting *Times* link of $\Pi$. But if $T_2'$ is a hereditary splitting *Times* link of $\Pi_2$ then $T$ is a hereditary splitting *Times* link of $\Pi$. □

*A minimal representation of prenets and proof nets*

To define a prenet or a proof net $\Pi$ it is enough to give its conclusions and the pairs of propositional variables which are linked by an axiom link. These pairs can be depicted by a 2-permutation $\sigma_\Pi$ — that is a permutation such that $\sigma_\Pi^2 = Id$ and $\forall x\ \sigma_\Pi(x) \neq x$ — defined on the set of occurrences of atoms in the sequence of conclusions. This representation will become necessary when we will deal with proof nets for the Lambek calculus, that are parse structures for Lambek categorial grammars.

Up to now, representing the conclusions by a graph is needed to check whether a prenet is a proof net (Girard, 1987; Danos and Regnier, 1989; Asperti, 1991; Asperti and Dore, 1994; Métayer, 1993). This graph can be minimized in more abstract representation (Retoré, 2003). There exists an alternative criterion relying on denotational semantics (Retoré, 1997) which does not need such a graph, but, unfortunately, checking the correctness becomes exponential.

Let us give the description of the examples 6.22 and 6.19 by means of 2-permutations.

*Example 6.31*

| Proof Net $\Pi$ | Example 6.22 | | | | | | | | | | Example 6.19 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Conclusions of $\Pi$ | $n^\perp \otimes n$ | | $(n^\perp \wp n) \otimes (n^\perp \otimes n)$ | | | | $n^\perp$ | $n \otimes np^\perp$ | | $np$ | $n^\perp \otimes n$ | | $n \wp n^\perp$ | |
| Atom occurrences $x$ | $n_1^\perp$ | $n_2$ | $n_3^\perp$ | $n_4$ | $n_5^\perp$ | $n_6$ | $n_7^\perp$ | $n_8$ | $np_9^\perp$ | $np_{10}$ | $n_1^\perp$ | $n_2$ | $n_3$ | $n_4^\perp$ |
| $\sigma_\Pi(x)$ | $n_4$ | $n_3^\perp$ | $n_2$ | $n_1^\perp$ | $n_8$ | $n_7^\perp$ | $n_6$ | $n_5^\perp$ | $np_{10}$ | $np_9^\perp$ | $n_3$ | $n_4^\perp$ | $n_1^\perp$ | $n_2$ |

### 6.4.3 Intuitionistic Proof Nets

**Definition 6.32.** *An intuitionistic proof net with conclusions $F_1, \ldots, F_n$ is a proof net satisfying:*

INTUI: $\forall i \in [1,n]\ F_i \in \mathsf{Li}^\circ \cup \mathsf{Li}^\bullet$.

For instance the example 6.20 is not an intuitionistic proof net since $n \wp n \notin \mathsf{Li}^\bullet \cup \mathsf{Li}^\circ$.

**Theorem 6.33.** *Every sequent calculus proof $A_1, \dots, A_n \vdash B$ in IMLL translates into an intuitionistic proof net with conclusions $-A_n, \dots, -A_1, +B$.*

*Conversely, let $\Pi$ a proof net with conclusions $F_1, \dots, F_n \in \mathsf{Li}$. Then there exists a unique index $i_0$ in $[1,n]$ such that $F_{i_0} \in \mathsf{Li}^\circ$ and $F_i \in \mathsf{Li}^\bullet$, for $i \neq i_0$, and $\Pi$ is the translation of a proof in IMLL of*

$$(F_{i_0-1})^\bullet_{\mathsf{Lp}}, (F_{i_0-2})^\bullet_{\mathsf{Lp}}, \dots, (F_1)^\bullet_{\mathsf{Lp}}, (F_n)^\bullet_{\mathsf{Lp}}, \dots, (F_{i_0+1})^\bullet_{\mathsf{Lp}} \vdash (X_{i_0})^\circ_{\mathsf{Lp}}$$

*Proof.* The first part is obvious.

For the converse, we first have to justify the existence of $i_0$. This existence is justified by Theorem 6.28 (it shows that $\Pi$ is the translation of proof of MLL) and proposition 6.8 (which shows that a proof in MLL with all its conclusions in $\mathsf{Li}^\bullet \cup \mathsf{Li}^\circ$ has exactly one conclusion in $\mathsf{Li}^\circ$ and all the others in $\mathsf{Li}^\bullet$). Once the existence of $i_0$ is established, the result follows from proposition 6.9, which shows that given a sequentialisation of $\Pi$ in MLL, with conclusions $\vdash F_1, \dots, F_n$ (with $F_{i_0}$ in $\mathsf{Li}^\circ$ and all the others in $\mathsf{Li}^\bullet$) corresponds to a proof in IMLL of

$$(F_{i_0-1})^\bullet_{\mathsf{Lp}}, (F_{i_0-2})^\bullet_{\mathsf{Lp}}, \dots, (F_1)^\bullet_{\mathsf{Lp}}, (F_n)^\bullet_{\mathsf{Lp}}, \dots, (F_{i_0+1})^\bullet_{\mathsf{Lp}} \vdash (X_{i_0})^\circ_{\mathsf{Lp}} \qquad \square$$

### 6.4.4 Cyclic Proof Nets

We now turn our attention towards proof nets for NC-MLL. These are proof nets which can be drawn in the plane without intersecting axioms, keeping the same design and up-down orientation for links. This condition is strictly stronger than being a planar graph (because we ask for the links to be drawn respecting left-right and up-down as shown in the figures). Consequently we shall present this condition without any reference to an embedding of the graph in the plane, but by means of a 2-permutation (bracketings from formal language theory would work just the same). This restriction, combined with the restriction for intuitionistic proof nets from the previous paragraph, will give us a characterization of proof nets for the Lambek calculus, and therefore give us a way to parse phrases and sentences with proof nets.

#### Cyclic Permutations and Compatibility of a 2-Permutation

A permutation $\psi$ over a set $E$ with $n$ elements is said to be cyclic whenever:

$$\forall x, y \in E \quad \exists k \in [0, n-1] \quad y = \psi^k(x) \text{ (with } \psi^0(x) = x)$$

such a permutation $\psi$ can be described by an expression:

$$\triangleright\, x; \psi(x); \psi(\psi(x)); \cdots ; \psi^{n-1}(x) \,\triangleright$$

Given $x, y \in E$, and an index $k \in [0, n-1]$ such that $y = \psi^k(x)$, we write $[x, y]$ for the set $\{z \mid \exists j \in [0,k] \quad z = \psi^j(x)\}$; similarly $[x, y[$ is defined as $\{z \mid \exists j \in [0,k[ \quad z = \psi^j(x)\}$ etc.

Given a set $E$ endowed with a cyclic permutation $\psi$ and a 2-permutation $\sigma$ we can give an algebraic account of the following geometric fact: if we place the points of $E$ on a circle following the cyclic order $\psi$, the chords joining $x$ and $\sigma(x)$ *do not intersect* any other chord — in other words, $\sigma$ is a correct bracketing, w.r.t. the cyclic order $\psi$ over $E$.

**Definition 6.34.** *A 2-permutation $\sigma$ of $E$ is said to be* compatible *with a cyclic permutation $\psi$ of $E$ whenever $\forall x,y \in E\ x \in [y,\sigma(y)] \Rightarrow \sigma(x) \in [y,\sigma(y)]$.*

For instance the 2-permutation $\sigma_\Pi$ of the example 6.31 $(n_1^\perp,n_3),(n_2,n_4^\perp)$ is not compatible with the cyclic permutation $\rhd n_1^\perp;n_2;n_3;n_4^\perp \rhd$. Indeed, $n_2 \in [n_1^\perp,\sigma_\Pi(n_1^\perp){=}n_3]$ while $\sigma_\Pi(n_2) = n_4^\perp \not\in [n_1^\perp,n_3]$.

In the following definition the $E_i$'s should be viewed as the conclusions of a proof net $\Pi$, endowed with the cyclic permutation $\Psi_\Pi$. The induced cyclic permutation is the cyclic permutation induced on the atoms — thus, viewing $\sigma$ of the previous definition as the axioms of $\Pi$, we are able to express that axioms do not intersect.

**Definition 6.35.** *Let $\rhd E_1;\cdots;E_n \rhd$ be a cyclic permutation of $M = \{E_1,\ldots,E_n\}$ where each $E_i$ is a sequence of symbols $a_i^1,a_i^2,\ldots,a_i^{j_i}$. The cyclic permutation induced by $\Psi$ over the disjoint sum of the symbols of the $E_i$'s is the cyclic permutation defined by:*

$$\rhd a_1^1;a_1^2;\cdots;a_1^{j_1};a_2^1;a_2^2;\cdots;a_2^{j_2};\cdots;a_n^1;a_n^2;\cdots;a_n^{j_n} \rhd$$

In order to characterize the proof nets for the Lambek calculus we shall need the following proposition:

**Proposition 6.36.** *Let $\Psi$ be a cyclic permutation over a finite set $M$ of $n$ sequences of symbols $M = E_1,\ldots,E_n$. Let $\psi$ be the cyclic permutation induced on $E = \oplus E_i$, as in definition 6.35. Let $\sigma$ be a 2-permutation of $E$, compatible with $\psi$, as in definition 6.34. Let $\Sigma$ be the following (symmetric) relation over $M$: $E_i \Sigma E_j$ whenever there exists $x_i \in E_i$ such that $\sigma(x_i) \in E_j$. Let $\Sigma^*$ be the transitive closure of $\Sigma$; if $\Sigma^*$ has* exactly two equivalence classes *$\mathscr{G}$ and $\mathscr{D}$, then there exist $G \in \mathscr{G}$ and $D \in \mathscr{D}$ such that: $\mathscr{G} = [G,D[$ and $\mathscr{D} = [D,G[$.*

*Proof.* By induction on $\#E + n$.

If one of the class contains only one element, the result is obvious — this necessarily happens when a class has a single element, for instance when $n = 2$.
*There exists $z$ such that $\psi(z) = \sigma(z)$* Let $z$ be a point such that $\#]z,\sigma(z)[$ has the smallest number of elements, and let us show that $\#]z,\sigma(z)[= 0$ — hence $\psi(z) = \sigma(z)$. Assume that there exists $y \in ]z,\sigma(z)[$; since $\sigma$ is compatible with $\psi$, $\sigma(y) \in ]z,\sigma(z)[$. Thus one of the two intervals $]y,\sigma(y)[$ or $]\sigma(y),\sigma(\sigma(y)) = y[$ is a subset of $]z,\sigma(z)[$, and since none of them contains $y$, they have strictly less elements than $\#]z,\sigma(z)[$, contradiction.

Let $z$ be an element such that $\psi(z) = \sigma(z)$ and let $i$ be the index such that $z \in E_i$. Three cases can happen:

$\underline{\sigma(z) \in E_i \text{ and } E_i = z, \sigma(z)}$ In this case, $E_i$ is the only element in its equivalence class, and the result is clear.

$\underline{\sigma(z) \in E_i \text{ and } E_i = \ldots, z, \sigma(z), \ldots}$ In this case, replace $E_i$ with $E_i \setminus \{z, \sigma(z)\}$, restrict $\sigma$ and $\psi$ to $E \setminus \{z, \sigma(z)\}$. The induction hypothesis apply, and since $\Sigma^*$ remains unchanged, the $D$ and $G$ provided by the induction hypothesis are solutions for the original problem.

$\underline{\sigma(z) \notin E_i}$. In this case $\sigma(z)$ is the first symbol of $E_{i+1} = \Psi(E_i)$. Let us consider the following reduction problem:

let $\Psi'$ be the cyclic permutation $\rhd E_1; \ldots; E_{i-1}; E_{i(i+1)}; E_{i+2}; \ldots; E_n \rhd$ where $E_{i(i+1)}$ is the sequence of symbols $E_i, E_j$

Observe that $E$, $\psi$ and $\sigma$ remains unchanged, and therefore $\sigma$ is compatible with $\psi$. Since $E_i \Sigma E_{i+1}$ the equivalence relation $\Sigma'^*$ for this reduction problem also has exactly two classes.

Hence we are faced with a similar problem with $\#M' = n-1$. The induction hypothesis yields $G'$ and $D'$ such that $\mathscr{G}' = [G', D'[$ and $\mathscr{D}' = [D, G'[$. A solution to the original problem is given by $G = G'$ and $D = D'$ —- if $G'$ (resp. $D'$) is $E_{i(i+1)}$, then $G$ (resp. $D$) should be $E_i$. □

## Cyclic Proof Nets

**Definition 6.37.** *A cyclic prenet with conclusions $\Psi :\rhd A_1; \cdots; A_n \rhd$ is a prenet with conclusions $A_1, \ldots, A_n$ endowed with the cyclic permutation $\Psi_\Pi :\rhd A_1, \ldots, A_n \rhd$. We write $\Psi_\Pi^{at}$ for the cyclic permutation induced by $\Psi_\Pi$ on the atoms of $\Psi$ — in the sense of the definition 6.35.*

**Definition 6.38.** *A cyclic prenet with conclusion $\Psi :\rhd A_1, \cdots, A_n \rhd$ is a cyclic proof net if and only if it is a proof net with conclusion $A_1, \ldots, A_n$ (the conditions* ØÆ *and* SAT*are satisfied) and:*

NC*: $\sigma_\Pi$ is compatible with $\Psi_\Pi^{at}$*

For instance the example 6.19 is not a cyclic proof net. Indeed, $\Psi_\Pi = \rhd n_1^\perp \otimes n_2; n_3 \wp n_4^\perp \rhd$ (there are only two conclusions, so there is only one possible cyclic permutation), and $\Psi_\Pi^{at} = \rhd n_1^\perp; n_2; n_3; n_4^\perp \rhd$, while the 2-permutation $\sigma_\Pi$ of its axiom links, given in example 6.31, is not compatible with $\Psi_\Pi^{at}$ — as we have seen after the definition 6.34.

The proof nets of the examples 6.20, 6.21, 6.22 and 6.23 are cyclic proof nets.

**Theorem 6.39.** *Every sequent calculus proof of $\vdash A_1, \ldots, A_n$ in NC-MLL translates into a cyclic proof net with conclusions $\rhd A_1; \cdots; A_n \rhd$.*

*Conversely, every cyclic proof net with conclusion $n \rhd A_1; \cdots; A_n \rhd$ is the translation of at least a sequent calculus proof of $\vdash A_1, \ldots, A_n$ in NC-MLL.*

*Proof.* The first part is rather simple to establish by induction on the sequent calculus proof. Nevertheless one should take care of the compatibility of $\Psi_\Pi^{at}$ with $\sigma_\Pi$; to do so, one should place atoms on a circle, and draw axiom links as chords of this circle, and draw R&B subformula trees outside the circle. Observe that the cyclic exchange (cx) corresponds to the equality of the proof nets.

The converse is proved by induction on the number of links of the proof net $\Pi$. As it is a proof net, Proposition 6.29 applies.

If $\Pi$ is an axiom $\rhd \alpha, \alpha^\perp \rhd = \rhd \alpha, \alpha^\perp \rhd$ a sequentialisation is provided by the axiom $\vdash \alpha, \alpha^\perp$ of NC-MLL.

If $\Pi$ has a final *Par* link $A_i = A \wp A'$, let us consider $\Pi'$ the proof net obtained from $\Pi$ by suppressing this final *Par* link and endowed with the cyclic permutation $\rhd A_1;\ldots;A_{i-1};A;A';A_{i+1};\cdots A_n \rhd$. The proof net $\Pi'$ is a cyclic proof net as well, since $\Psi^{at}_{\Pi'} = \Psi^{at}_{\Pi}$ and $\sigma_{\Pi'} = \sigma_{\Pi}$. By induction hypothesis there exists a sequent calculus proof in NC-MLL corresponding to $\Pi'$, and applying a $\wp$ rule to this proof yields a sequentialisation of $\Pi$.

Otherwise, by Lemma 6.29, $\Pi$ has a splitting *Times*, say $A_i = A \otimes A'$. Suppressing this final link yields two proof nets $\Pi_A$ and $\Pi_{A'}$ with conclusions $\Gamma_A = A_{i_1},\ldots,A_{i_p},A$ and $\Gamma_{A'} = A_{j_1},\ldots,A_{j_q},A'$ with $\{i_1,\ldots,i_p,j_1,\ldots,j_q\} = [1,n] \setminus \{i\}$. Consider the prenet $\Pi' = \Pi_A \cup \Pi_{A'}$ and endow its conclusions with the cyclic permutation $\rhd A_1;\cdots;A_{i-1};A;A';A_{i+1};\cdots;A_n \rhd$. Since $\Psi^{at}_{\Pi'} = \Psi^{at}_{\Pi}$ and $\sigma_{\Pi'} = \sigma_{\Pi}$, the 2-permutation $\sigma_{\Pi'}$ is compatible with $\Psi^{at}_{\Pi'}$. Let $\Sigma$ be the (symmetric) relation between the conclusions of $\Pi'$ defined by: $\exists x \in C\ \sigma_{\Pi}(x) \in C'$ — in other words, this relation holds whenever $\Pi$ contains an axiom with a conclusion in $C$ and the other in $C'$. The link $A \otimes B$ is splitting in $\Pi$, means that $\Sigma^*$ has exactly two equivalence classes $\Gamma_A$ and $\Gamma_{A'}$. Because of Proposition 6.36 the cyclic permutation of the conclusions of $\Pi'$ can be written as $\rhd A_{i_1};\cdots,A_{i_p};A;A',A_{j_1};\cdots;A_{j_q} \rhd$. Thus $\Pi_A$ (resp. $\Pi_{A'}$) endowed with the cyclic permutation $\rhd A_{i_1};\cdots,A_{i_p};A \rhd$ (resp. $\rhd A',A_{j_1};\cdots;A_{j_q} \rhd$) is a cyclic proof net. Indeed $\Pi_A$ is a proof net and since $\sigma_{\Pi_A}$ and $\Psi^{at}_{\Pi_A}$ are the restrictions to $\Gamma_A$ of $\sigma_{\Pi}$ and $\Psi^{at}_{\Pi}$ compatibility is preserved — the same argument works for $\Pi_{A'}$.

Therefore, by induction hypothesis we have two sequent calculus proofs in NC-MLL with conclusions $\vdash A_{i_1};\cdots,A_{i_p};A$ and $\vdash A';A_{j_1};\cdots;A_{j_q}$ corresponding to $\Pi_A$ and $\Pi'_A$. Applying the rule $\otimes$ of NC-MLL yields a proof with conclusion $\vdash \Gamma_A, A \otimes B, \Gamma_B$ corresponding to $\Pi$. □

For instance the proofs of the examples 6.10 and 6.11 correspond to the cyclic proof net of the example 6.22, which is *equal* to the proof net of the example 6.23. Indeed expressions $\rhd n^\perp \otimes n; (n^\perp \wp n) \otimes (n^\perp \otimes n); n^\perp; n \otimes np^\perp; np \rhd$ and $\rhd n^\perp; n \otimes np^\perp; np; n^\perp \otimes n; (n^\perp \wp n) \otimes (n^\perp \otimes n) \rhd$ denotes the same cyclic permutation.

### 6.4.5 Proof Nets for the Lambek Calculus — With or Without Empty Antecedent

In order to characterize the proof nets of the Lambek calculus L, which exclude sequents with empty antecedents, we need the following proposition. It involves the notion of a sub-prenet and subproof net: a sub-prenet (sub proof net) is a subgraph of a prenet (proof net) which is itself a prenet (proof net). A sub-prenet of a proof net is not always a proof net: it is possible that SAT does not hold in the sub-prenet (but ØÆ holds).

**Proposition 6.40.** *Let $\Pi$ be a proof net; the following statements are all equivalent:*

1. *Every sub-prenet of $\Pi$ has at least two conclusions. ($\varepsilon$-FREE)*
2. *Every sub proof net of $\Pi$ has at least two conclusions.*
3. *Every sequentialisation of $\Pi$ contains only sequents with at least two conclusions.*
4. *There exists a sequentialisation of $\Pi$ which contains only sequents with at least two conclusions.*

*Proof.* Implications $1 \Rightarrow 2$, $2 \Rightarrow 3$ and $3 \Rightarrow 4$ are straightforward.

$4 \Rightarrow 1$ is shown by induction on the number of links in $\Pi$, which is equal to the number of axioms and logical rules of every sequentialisation of $\Pi$. Let us consider a sequentialisation $\Pi^*$ of $\Pi$, such that every sequent of it contains at least two formulae. We can assume the last rule of $\Pi^*$ is not an exchange rule: indeed the same proof without this exchange rule is also a sequentialisation of $\Pi$, with all sequents having at least two formulae.

If the last rule of $\Pi^*$ is an axiom, $\Pi^*$ consists of this axiom, which contains two formulae. In this case $\Pi$ is an axiom, whose only sub-prenet is itself, which has two conclusions.

If that rule of $\Pi^*$ is a two premise rule, applied to two proofs $\Pi'^*$ and $\Pi''^*$, the corresponding link of $\Pi$ is a splitting *Times* link: $\Pi$ is obtained from two smaller proof nets $\Pi'$ and $\Pi''$ connected by this *Times* link. The two proofs $\Pi'^*$ and $\Pi''^*$ are possible sequentialisations for $\Pi'$ and $\Pi''$ and these proofs also have sequents with at least two formulae. Thus the induction hypothesis can be applied to $\Pi'$ and $\Pi''$: every sub-prenet of $\Pi'$ or of $\Pi''$ has at least two conclusions. The intersection of a sub-prenet $s\Pi$ of $\Pi$, with $\Pi'$ (resp. $\Pi''$) is a sub-prenet of $\Pi'$ (resp. $\Pi''$) which has $p > 1$ (resp. $q > 1$) conclusions. If the *Times* link is part of $s\Pi$ then the number of conclusions of $s\Pi$ is $p+q-1 > 1$, and otherwise the number of conclusion of $s\Pi$ is $p+q > 1$. Thus, in any case $\Pi$ satisfies $\varepsilon$-FREE.

If the last rule of $\Pi^*$ is a one premise rule applied to some proof $\Pi'^*$, the corresponding link of $\Pi$ is a final *Par* link. Let $\Pi'$ be the proof net obtained from $\Pi$ by removing this final *Par* link; it is a proof net with strictly less links, which has a sequentialisation $\Pi'^*$ with sequents with more than one conclusions. Hence, by induction hypothesis every sub-prenet of $\Pi'$ has at least two conclusions. Given a sub-prenet $s\Pi$ of $\Pi$, its intersection $s\Pi'$ with $\Pi'$ has at least two conclusions. It is impossible that $s\Pi'$ has only the two conclusions $X$ and $Y$. Indeed we know that $\Pi$ has at least two conclusions, hence it has another conclusion $Z$ in addition to $X \wp Y$. Since $\Pi$ is a proof net it is connected, and there exists a path joining $s\Pi'$ to $Z$ conclusion, and this path can be assumed to lie outside $s\Pi'$ — by cutting the part inside $s\Pi'$. So there exists an edge of $\Pi$, incident to $s\Pi'$ starting this path. This edge can neither be the R edge below $X$, nor the one below $Y$, since any path starting by one of these edges has to enter again $s\Pi'$. But the only way to leave a sub-prenet is from one of its conclusions: therefore $s\Pi'$ has a conclusion which is neither $X$ nor $Y$. Let $p$ be the number of conclusions of $s\Pi'$. If $X$ and $Y$ are among the $p$ conclusions of $s\Pi'$, then $s\Pi'$ has another conclusion and $p > 2$. Therefore, either $s\Pi$ has $p > 2$

conclusions (when $X \wp Y$ is not one of its conclusions), or $s\Pi$ has $p-1>1$ conclusions (when $X \wp Y$ is one of its conclusions). If $X$ or $Y$ is not a conclusion of $s\Pi'$, then $X \wp Y$ is not a conclusion of $s\Pi$, and $s\Pi$ and $s\Pi'$ have the same number of conclusions $p>1$.

In any case $s\Pi$ has at least two conclusions. □

**Definition 6.41.** *A Lambek proof net of conclusion $\Psi_\Pi = \rhd F_1; \cdots ; F_n \rhd$ is an intuitionistic cyclic proof net, i.e. a prenet satisfying*

ØÆ*: there is no æ cycle* alternate elementary cycle.
SAT*: There always exists an æ path between any two vertices.*
INTUI*: Every conclusion $F_i$ is in* $\mathsf{Li}^\bullet \cup \mathsf{Li}^\circ$.
NC*: $\sigma_\Pi$ is compatible with $\Psi_\Pi^{at}$ — the axioms of $\Pi$ do not intersect.*

*A Lambek proof net is said to be without empty antecedent if, moreover:*

$\varepsilon$-FREE*: Every subprenet of $\Pi$ has at least two conclusions.*

Among the four equivalent statements given above, we have chosen the first one, because subprenets are easier to define. It is enough to chose a set of vertices of the proof net, and to close it by subformula and axiom links, without verifying SAT or ØÆ. When NC and ØÆ hold, this amounts to the following fact: for every subformula $G$ of a conclusion, the first and last atom of $G$ are never linked by an axiom. If $G = H \otimes H'$ then this holds, and if $G = H \wp H'$, this exactly means that there is no sub-net with a single conclusion.

**Theorem 6.42.** *Every sequent calculus proof with conclusion $A_1, \ldots, A_n \vdash B$ in $\mathsf{L}_\varepsilon$ (resp. $\mathsf{L}$) translates into a Lambek proof net (resp. a Lambek proof net without empty antecedent) with conclusions $\rhd -A_n; \cdots, -A_1; +B \rhd$.*

*Conversely, let $\Pi$ be a Lambek proof net (resp. a Lambek proof net without empty antecedent) with conclusions $\rhd F_1; \ldots; F_n \rhd$. and let $i_0$ be the unique index in $[1,n]$ such that $F_{i_0} \in \mathsf{Li}^\circ$ and $F_i \in \mathsf{Li}^\bullet$, for $i \neq i_0$. The proof net $\Pi$ is the translation of at least a sequent calculus proof in $\mathsf{L}_\varepsilon$ (resp. $\mathsf{L}$) of*

$$(F_{i_0-1})^\bullet_L, (F_{i_0-2})^\bullet_{\mathsf{Lp}}, \ldots, (F_1)^\bullet_{\mathsf{Lp}}, (F_n)^\bullet_{\mathsf{Lp}}, \ldots, (F_{i_0+1})^\bullet_{\mathsf{Lp}} \vdash (F_{i_0})^\circ_{\mathsf{Lp}}$$

*Proof.* The first part is a straightforward induction on the sequent calculus proof in $\mathsf{L}_\varepsilon$ (resp. $\mathsf{L}$).

For the second part, we know from Proposition 6.39 that there is a sequentialisation corresponding to $\Pi$ in NC-MLL, with conclusion $\vdash F_1, \cdots, F_n$. Because of Proposition 6.9, this sequent calculus proof in NC-MLL corresponds to a proof of

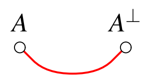
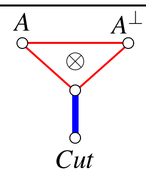
$$(F_{i_0-1})^\bullet_L, (F_{i_0-2})^\bullet_{\mathsf{Lp}}, \ldots, (F_1)^\bullet_{\mathsf{Lp}}, (F_n)^\bullet_{\mathsf{Lp}}, \ldots, (F_{i_0+1})^\bullet_{\mathsf{Lp}} \vdash (F_{i_0})^\circ_{\mathsf{Lp}}$$

in $\mathsf{L}_\varepsilon$. Using $1 \Rightarrow 3$ of Proposition 6.40, it is easily seen that whenever $\Pi$ is a Lambek proof net without empty antecedent, the sequentialisation in $\mathsf{L}_\varepsilon$ is in fact in $\mathsf{L}$, i.e. it does not contain sequents with only one formula. □

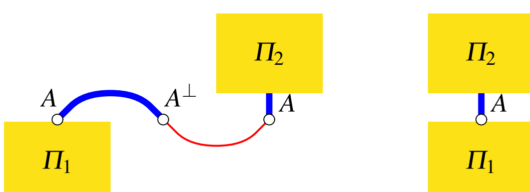
Among our proof net examples, only Examples 6.21, 6.22 and 6.23 are Lambek proof nets. Example 6.22 corresponds to the parse structures 6.5 and 6.6: we thus got rid of *spurious ambiguity* — a classical drawback of sequent proof search for categorial grammars, which provides too many proofs/parse structures for a single analysis. One advantage of working with cyclic permutation is that Examples 6.22 and 6.23 are *equal*. Example 6.21 is not a Lambek proof net without empty antecedent: indeed it contains a sub-net whose only conclusion is $n^{\perp} \wp n$. It corresponds to the Example 6.4 in $\mathsf{L}_{\varepsilon}$.

### 6.4.6 Cut Elimination for Proof Nets

We have deliberately excluded the cut links from our discussion of proof nets so far. We will present two versions of the cut link, the first is a simple connection of a formula with its negation. The second is a special kind of tensor link with the constant "Cut" as its conclusion. This second formulation has the advantage that we can treat the cut link just as a tensor link in sequentialisation proofs.

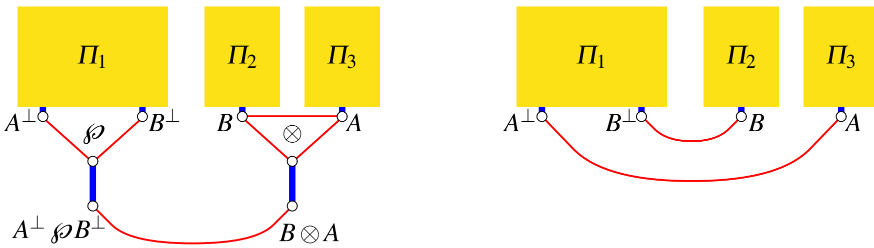
| Name | Graph | Premises | Conclusions |
|---|---|---|---|
| Cut |  | $A$ and $A^{\perp}$ | none |
| $\text{Cut}_{\otimes}$ |  | $A$ and $A^{\perp}$ | the special constant "Cut" |

Cut elimination for proof nets is very simple — at least in the commutative case, we will discuss why cut elimination is more diffcult for non-commutative calculi in the next section. The base case occurs when the cut link is connected to an axiom link. In this case we are in the following situation.




We know that $\Pi_1$ and $\Pi_2$ must be disjoint, since the vertices of $\Pi_1$ and $\Pi_2$ are connected by the path passing through the cut and the axiom link and we know that the complete proof net does not have an alternate elementary cycle. We can eliminate the cut as shown in the figure above on the right: we remove both the cut and the axiom link and the resulting structure satisfies ØÆ and SAT because the unreduced structure did.

In case the cut link is connected to a complex formula, we must be in the situation shown below on the left.




That is, given that we have a proof net on the left hand side of the figure, $\Pi_2$ and $\Pi_3$ are connected by the tensor link which is shown in the figure and therefore not connected elsewhere and $\Pi_1$ is connected to $\Pi_2$ and $\Pi_3$ by means of the paths shown in the figure and therefore not by any other paths.

We can replace the cut link by two cut links on the subformulas as shown in the figure above on the right. It is easy to see that the resulting structure is again a proof net. The path from $A$ (and the formulas of $\Pi_3$) to $B$ (and the formulas of $\Pi_2$) which used to be connected directly through the *Times* link now goes through $\Pi_1$, but all other paths have been shortened.

It is also easy to see that cut elimination is confluent.

**Lemma 6.43.** *Let $\Pi$ be a proof net with cuts with possible cuts $K_i$, $K_i^\perp$ such that all conclusions are polarized and with one output conclusion, then all $K_i$, $K_i^\perp$ are polarized.*

*Proof.* This lemma is an easy corollary of Proposition 6.30 when we treat cut links as tensor links $Cut_\otimes$. We assume, without loss of generality, that $\Pi$ has only atomic axiom links. Since $\Pi$ is a proof net, we sequentialise as before.

If $\Pi$ contains conclusions which are par links, then we can remove the par link and the result will be a proof net, it is easy to verify that this new proof net is still polarized since all polarized par links reduce the number of negative formulas, but keep the number of positive formulas constant. In case there are no terminal par links, by Proposition 6.30 there is a hereditary splitting *Times* link, possibly a cut. In case it is not a cut, removing the $n > 0$ hereditary splitting tensors will produce $n+1$ disjoint proof nets $\Pi_1, \ldots, \Pi_n$. We only need to verify that all $\Pi_i$ are polarized. If the conclusion of the hereditary splitting link is a postive *Times* link with conclusion $A \otimes B$, then all $n$ *Times* links are positive and each $\Pi_i$ will have a positive conclusion after removal of all the *Times* links. If it is a negative *Times* link, then extactly one of the $\Pi_i$, say $\Pi_k$, has a positive conclusion which is already a conclusion of $\Pi$ and is therefore connected to the hereditary splitting *Times* by an input conclusion of $\Pi_k$. Given the form of the polarized *Times* links, this means that all $\Pi_i$ for $i \neq k$ have a single positive conclusion.

The interesting case is when there hereditary splitting *Times* link is a cut link between $A$ and $A^\perp$ where $A$ and $A^\perp$ are not necessarily polarized. However, we know by induction hypothesis that all conclusions of $\Pi$ are polarized. In case $A$ is an atomic formula, this means that $A$ is connected by an axiom link to a formula $A^{\perp}$ which is a conclusion of $\Pi$ and therefore both $A$ and $A^{\perp}$ are polarized. Now suppose $A = B \wp C$ and $A^{\perp} = C^{\perp} \otimes B^{\perp}$: this is the only combination which is not polarized and we show it leads to a contradiction. One step of cut elimination connects $B$ to $B^{\perp}$ and $C$ to $C^{\perp}$. We also know that $\Pi_1$, the subnet with conclusion $B^{\perp}$, and $\Pi_2$, the subnet with conclusion $C^{\perp}$, are both a proof nets and the the proof net $\Pi$ is polarized, that is $\Pi$ has a single polarized output conclusion and all other conclusions of $\Pi$ a polarized input conclusions. Since $B^{\perp}$ and $C^{\perp}$ are both input conclusions of their respective proof nets $\Pi_1$ and $\Pi_2$ and all other conclusions of both proof nets were conclusions of $\Pi$ this means that one of $\Pi_1$ and $\Pi_2$ does not have a positive conclusion and therefore is not a polarized proof net.

### 6.4.7 Cuts and Non-commutative Proof Nets

There are a variety of multiplicative proof nets criteria in the usual commutative case that are fully satisfying. But the non-commutative case is rather tricky when there are cuts. To the best of our knowledge, only the criterion by Paul-André Melliès is fully satisfactory (Melliès, 2004). What do we mean by "satisfactory" for a correctness criterion?

1. every sequent calculus proof should be mapped to a correct proof net, rules corresponding to links (in particular cut-free proofs should be mapped to cut-free proof nets)
2. every correct proof net should correspond to a sequent calculus proof links corresponding to rules (in particular cut-free proofs should be mapped to cut-free proof nets)
3. sequent calculus proofs that only differ up to rule permutations should be mapped to the same proof net
4. the criterion should be preserved under cut elimination in proof nets (not as obvious as it may seem)

The reason why criteria are trickier in the non commutative case mainly comes from the difference between cut links and times links. In a planar representation, a cut link is allowed to be included in an internal face, while a times link which is a conclusion of the proof net is not. Figure 6.3 (after Melliès, 2004, p. 294) shows an example.

As the reader can easily verify (Exercises 6.5 asks you to verify a number of properties of this proof structure), it is a proof structure of

$$\vdash (b^{\perp} \wp b) \otimes (a^{\perp} \wp a)$$

which, though this sequent *is* derivable, the proof structure shown in Figure 6.3 is not sequentialisable in a non-commutative calculus. However, it is planar and satisfies all other conditions for proof nets (at least for proof nets which allow empty antecedent derivations).

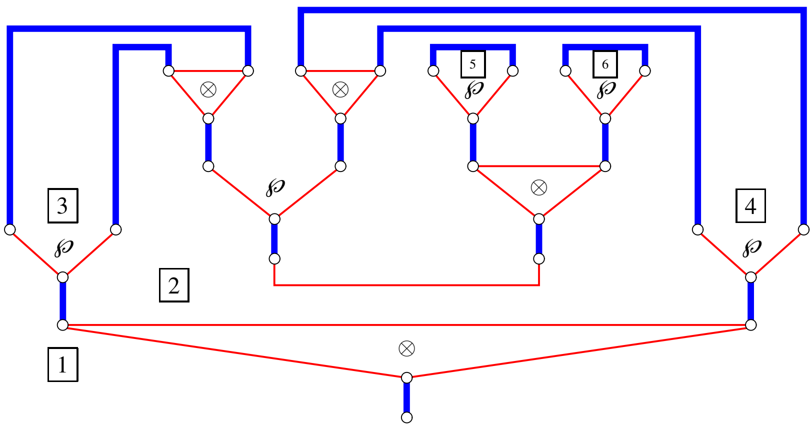



**Fig. 6.3.** Proof structure which does not correspond to a non-commutative sequent proof

To see that this proof net is not sequentialisable, the only conclusions of the proof net are the tensor link and the cut link (which, as before, we treat as a tensor link as well for the purpose of sequentialisation). Only the cut link splits the proof structure into two disconnected proof structures: one proof structure with conclusion $\vdash (a^{\perp} \wp a) \otimes (b^{\perp} \wp b)$ — which is both derivable and a substructure which is a proof net — but the second proof structure is a proof structure of

$$\vdash (b^{\perp} \otimes b) \wp (a^{\perp} \otimes a), (b^{\perp} \wp b) \otimes (a^{\perp} \wp a)$$

which is *not* derivable in a non-commutative logic.

The correctness condition of Melliès (2004) (though the terminology we use is closer to de Groote, 1999) formalizes this restriction on the conclusions. This condition is, to the best of our knowledge, the only correctness condition which works correctly for non-commutative proof nets with cut links.

**Definition 6.44.** *Let $P$ be a planar drawing of a proof structure. A* face *$f$ of $P$ is a connected area enclosed by the edges of the proof structure such that the border of $f$ contains at least one* B *edge.*

The faces of Figure 6.3 are shown as $\boxed{n}$. The R triangle of a *Times* link is not counted as a face, though face $\boxed{5}$ and $\boxed{6}$ show valid three-edge faces.

**Definition 6.45.** *A face $f$ of a proof structure is an* internal *face iff it contains both* R *edges of at least one* Par *link. A face which is not internal is called* external.

**Definition 6.46.** *A proof structure $\Pi$ is a proof net iff it satisfies* ØÆ, SAT *and all conclusions of $\Pi$ are on the unique external face of $\Pi$.*

We can see that the external face of the proof structure in Figure 6.3 is face $\boxed{2}$ (and not face $\boxed{1}$ whose frontier contains both R edges of the par link connected to the cut link). As a conquence, the proof structure is not a proof net, since its conclusion is on the internal face $\boxed{1}$.

### 6.4.8 Basic Properties of Graphs and Proof Nets

This section covers some basic properties of graphs and proof nets. Notably, it shows that, under certain conditions we can replace the acyclicity and connectedness condition of Danos and Regnier (1989) by either an acyclicity condition or a connectedness condition, using some basic properties of acyclic and connected graphs (Bondy and Murty, 1976; Diestel, 2010). Though people have been aware of many of these properties for a long time (eg. J. van de Wiele (1991, p.c.)), it is actually rather hard to find in print, though Guerrini (2011) gives a clear presentation of many of the results of this section and Morrill and Fadda (2008) independently prove that acyclicity implies connectedness (part of Corollary 6.56 in this section).

In this section, we will often say that a *prenet* $\Pi$ is acyclic or connected, in the sense of Danos and Regnier (Definition 6.26) to indicate that all correction graphs of $\Pi$ are acyclic or connected.

**Definition 6.47.** *Let $G$ be a graph. We will use $v$ to denote the number of its vertices and $e$ to denote the number of its edges.*

**Proposition 6.48.** *If $G$ is acyclic and connected, then $e = v - 1$.*

*Proof.* By induction on $v$. If $v = 1$ then $e = 0$, which is the only acyclic connected graph with a single vertex.

Suppose $v > 1$, then $G$ contains at least one edge $m$ and since $G$ is acyclic $G - m$ has two components $G_1$ and $G_2$ which are acyclic and connected and which have less than $v$ vertices. By induction hypothesis $e_1 = v_1 - 1$ and $e_2 = v_2 - 1$ and therefore, since $e = e_1 + e_2 + 1$, we have $e = (v_1 - 1) + (v_2 - 1) + 1 = v_1 + v_2 - 1$ and since $v = v_1 + v_2$, we have $e = v - 1$ as required. □

**Proposition 6.49.** *If $G$ is a graph such that $e = v - 1$ then $G$ is acyclic iff $G$ is connected.*

*Proof.* If $G$ is acyclic, then it consists of a number of connected components $n$, each of which, being acyclic and connected, satisfies $e_i = v_i - 1$ for $1 \leq i \leq n$, according to Proposition 6.48. Therefore, for the complete graph we have $e = v - n$, where $n$ is the number of components. Since $e = v - 1$ by assumption, there is only a single connected component and therefore $G$ is connected.

If $G$ is connected then let $G'$ be an acyclic, connected subgraph of $G$ which contains all vertices of $G$ (that is, a spanning tree: to obtain $G'$, we delete the necessary number of edges from $G$ to obtain an acyclic, connected graph). Since $G'$ is acyclic and connected, according to Proposition 6.48 it has $v - 1$ edges. But since $G$ has $v - 1$ edges by assumption, $G$ is equal to $G'$ and therefore acyclic. □

**Definition 6.50.** *Let $\Pi$ be a (cut-free) prenet. We will use $c$ to denote its number of conclusions, $p$ to denote its number of par links, $t$ to denote its number of tensor links and $a$ to denote its number of axiom links.*

The following proposition, relating the number of tensor links, par links and conclusions of a proof net is also rather easy to show (Exercise 7.4 asks you to prove this proposition yourself). It was first noticed by in the early nineties and appears in Fleury (1996).

**Proposition 6.51.** *If $\Pi$ is a proof net, then $c+p=t+2$.*

**Proposition 6.52.** *If a prenet $\Pi$ is polarized and has a single positive conclusion, then $c+p=t+2=a+1$.*

*Proof.* When we look at the construction of a prenet from the axioms down, we see that a prenet with $a$ axioms, without any tensor and par links, has $a$ positive/output conclusions and $a$ negative/input conclusions. If $\Pi$ has a single positive conclusion, then it has $c-1$ negative conclusions.

- For each par link we add, we take two (possibly disconnected) conclusions of the prenet as its premises and introduce a new conclusion, increasing the number of par links by one, keeping the number of positive conclusions constant, but reducing the total number of negative conclusions by one — a polarized par link has either two negative premises and a negative conclusion or a positive and a negative premise and a positive conclusion.
- For each tensor link we add to the structure, on the other hand, we keep the number of negative conclusions constant, but reduce the total number of positive conclusions — a polarized tensor link has either two positive premises and a positive conclusion or a positive and a negative premise and a negative conclusion.

Therefore, the only way to obtain a prenet with $c-1$ negative conclusions is for the prenet to have $a-(c-1)$ par links and the only way to obtain a prenet with one positive conclusion is for the prenet to have $a-1$ tensor links. Therefore, every polarized prenet with a single output conclusion has $t=a-1$, $p=a-(c-1)$, which gives $c+p=t+2=a+1$. □

**Proposition 6.53.** *If $\Pi$ is a prenet, then every correction graph $G$ of $\Pi$ has $2a+p+t$ vertices and $a+p+2t$ edges.*

*Proof.* Since each vertex is the conclusion of exactly one link, we can simply count the conclusions of the links in the proof nets: two for each axiom link and one for each tensor and par link. For the edges, each correction graph replaces a tensor link by two edges and a par link by a single edge (the axiom links stay single edges). □

Propositions 6.54 and 6.55 follow the simple and elegant proofs of Guerrini (2011).

In Proposition 6.52, we have seen that $a=t+1$ held for polarized prenets with a unique positive conclusion. The next proposition shows that $a=t+1$ holds in general for proof *nets*.

**Proposition 6.54.** *If $\Pi$ is a proof net (not necessarily polarized) then $a = t + 1$.*

*Proof.* Since $\Pi$ is a proof net, all its correction graphs are acyclic and connected. Therefore, according to Proposition 6.48, $e = v - 1$. By Proposition 6.53 all correction graphs of $\Pi$ have $v = 2a + p + t$ and $e = a + p + 2t$ giving $a + p + 2t = 2a + p + t - 1$, which simplifies to $a = t + 1$. □

We can now show that for any prenet $\Pi$ such that $a = t + 1$, it suffices to check either acyclicity or connectedness to determine whether or not $\Pi$ is a proof net: in other words a prenet with a cycle (and satisfying $a = t + 1$) will necessarily be disconnected and a disconnected prenet satisfying $a = t + 1$ will necessarily contain a cycle.

**Proposition 6.55.** *Let $\Pi$ be a prenet with $a = t + 1$, $\Pi$ is a proof net iff $\Pi$ is acyclic and $\Pi$ is a proof net iff $\Pi$ is connected.*

*Proof.* If $\Pi$ is a proof net, then $a = t + 1$ by Proposition 6.54 and all correction graphs are both acyclic and connected by Definition 6.26.

For the other direction, suppose $\Pi$ is a prenet such that $a = t + 1$, then, by the same reasoning as used for the proof of Proposition 6.54, $e = v - 1$, which, according to Proposition 6.49 means that acyclicity implies connectedness and vice versa. □

By the preceding propositions, we can do even better in the intuitionistic case, where all formulas are polarized and there is a conclusion of output polarity. In this case $a = t + 1$ is satisfied by Proposition 6.52.

**Corollary 6.56.** *If $\Pi$ is polarized prenet with a single output conclusion then $\Pi$ is a proof net iff $\Pi$ is acyclic and $\Pi$ is a proof net iff $\Pi$ is connected.*

*Proof.* Immediate from Proposition 6.52 and Proposition 6.55 □

## 6.5 Parsing as Proof Net Construction

Assume we want to analyze the noun phrase '*un exemple très simple*', according to the lexicon provided in Example 6.4. We need a proof in L of

$$np\,/\,n, n, (n\setminus n)\,/\,(n\setminus n), n\setminus n \vdash np$$

Because of Proposition 6.42 this amounts to construct a Lambek proof net without empty antecedent with conclusions:

$$\rhd\, n^{\perp}\otimes n; (n^{\perp}\wp n)\otimes(n^{\perp}\otimes n); n^{\perp}; n\otimes np^{\perp}; np\,\rhd$$

— these "linear types" are automatically computed as we did in Example 6.1, and the order is inverted (see Proposition 6.9). So the lexicon automatically provides the R&B subformula trees of the proof net shown in Figure 6.4.

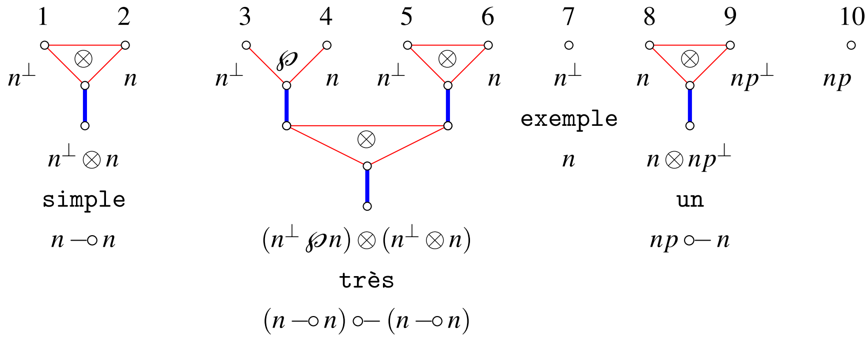



**Fig. 6.4.** Subformula trees of "un exemple très simple"

What is missing to obtain a proof net is $\sigma_\Pi$, the axiom links between the occurrences

$$n_1^\perp, n_2, n_3^\perp, n_4, n_5^\perp, n_6 n_7^\perp, n_8, np_9^\perp, np_{10}$$

They should be placed in such a way that the conditions ØÆ, SAT, INTUI, NC, $\varepsilon$-FREE are met. Of course, INTUI is automatically satisfied since all conclusions belong to $(\mathsf{Lp})^\perp$ and one (S) is in $\mathsf{Lp}$

Because axioms link dual formulae there must be an axiom $(np_9^\perp, np_{10})$. One should then link the $n$ and the $n^\perp$, and this makes 24 possibilities. However, thanks to the constraints expressed by ØÆ, SAT, NC and $\varepsilon$-FREE we almost have no choice:

$(n_4, n_5^\perp) \notin \sigma_\Pi$ — ØÆ, æ cycle with the *Times* link $(n_3^\perp \wp n_4) \otimes (n_5^\perp \otimes n_6)$.
$(n_5^\perp, n_6) \notin \sigma_\Pi$ — ØÆ, æ cycle with the *Times* link between these two atoms.
$(n_3^\perp, n_4) \notin \sigma_\Pi$ — $\varepsilon$-FREE, sub-prenet with a single conclusion.
$(n_4, n_7^\perp) \notin \sigma_\Pi$ — NC this would force $(n_5^\perp, n^6)$, which was shown to be impossible.
$(n_1^\perp, n_4) \in \sigma_\Pi$ — only possible choice for $n_4$.
$(n_2, n_3^\perp) \in \sigma_\Pi$ — NC, because of the previous line.
$(n_7^\perp, n_8) \notin \sigma_\Pi$ — SAT, yields a disconnected prenet, since we already have $(np_9^\perp, np_{10}) \in \sigma_\Pi$.
$(n_5^\perp, n_8), (n_6, n_7^\perp) \in \sigma_\Pi$ — only possible choice for these atoms, according to the above decisions.

Hence the only possible solution is the 2-permutation $\sigma_\Pi$ given in the example 6.31: $(n_1^\perp, n_4), (n_2, n_3^\perp), (n_5^\perp, n_8), (n_6, n_7^\perp), (np_9^\perp, np_{10})$. It corresponds to the prenet 6.22.

Remark that though we have shown in Corollary 6.56 that it suffices to check either connectedness or acyclicity, enforcing the two conditions together allows us to disqualify more invalid axiom links directly (see (Moot, 2007) for discussion).

Next, one has to check that the result is a Lambek proof net, without empty antecedent, and this is straightforward and quick. It corresponds to the sequent calculus proofs given in examples 6.5 and 6.6. The identification of various sequent calculus proofs into a single proof net leads to less possibilities when constructing the proof.

A natural question is the algorithmic complexity of this parsing algorithm. For the less constrained calculus MLL (only satisfying ØÆ and SAT) it is known to be NP complete (Lincoln et al, 1992), but the notion of splitting *Times* leads to efficient heuristics using the fact that there never can be any axiom link between the two sides of a *Times* link (de Groote, 1995). This considerably reduces the search space. The intuitionistic restriction does not lead to any improvement.

For the non commutative calculi, and in particular for the Lambek calculus, the order constraint NC is so restrictive that one might be tempted to think that the complexity is polynomial. However, a recent paper of Mati Pentus (Pentus, 2006) shows that the Lambek calculus with product is NP complete as well, with the help of a variation of the proof nets studied in this chapter. In addition, Yuri Savateev (Savateev, 2009) has shown that NP completeness holds even for the Lambek calculus without product.

Interesting work has been done on using dynamic programming techniques for finding proof nets for the Lambek calculus. De Groote (1999) — who improves the tabulation techniques introduced in (Morrill, 1996) — uses dynamic programming for the placement of axiom links, defining them by a context-free grammar. Given the results by Pentus and Savateev cited above, these strategies evidently do not give polynomial algorithms, but they may be extended to find interesting polynomial fragments of the Lambek calculus.

We'll have more to say about parsing using proof nets in Section 7.2 in the next chapter, where we talk about parsing categorial grammars using *multimodal* proof nets.

## 6.6 Proof Nets and Human Processing

Starting with a study by Johnson (Johnson, 1998) for center embedded relatives and then improved and extended by Morrill (Morrill, 2000, 2011), proof nets happen to be interesting parse structure not only from a mathematical viewpoint, but also from a linguistic viewpoint. Indeed they are able to address various performance questions like garden paths, center embedding unacceptability, preference for lower attachment, and heavy noun phrase shift, that can be observed when we use proof net construction as a way to parse sentences.

We follow Morrill (Morrill, 2000) and consider the following examples:

*Garden path sentences*

*1(a) The horse raced past the barn.*
*1(b) ?The horse raced past the barn fell.*

*2(a) The boat floated down the river.*
*2(b) ?The boat floated down the river sank.*

*3(a) The dog that knew the cat disappeared.*
*3(b) ?The dog that knew the cat disappeared was rescued.*

The (b) sentences are correct but seem incorrect. Indeed there is a natural tendency to interpret the first part of the (b) sentences as their (a) counterparts. Hence the correct, alternative analysis, which is a paraphrase of "The horse *which was* raced past the barn fell" is difficult to obtain.

*Quantifier-scope ambiguity*

Here are some examples of quantifier-scope ambiguity, with the preferred reading:

*I(a) Someone loves everyone.* $\exists\forall$
*I(b) Everyone is loved by someone.* $\forall\exists$

*II(a) Everyone loves someone.* $\forall\exists$
*II(b) Someone is loved by everyone.* $\exists\forall$

So in fact the preference goes for the first quantifier having the wider scope.

*Embedded relative clauses.*

*III(a) The dog that chased the rat barked.*
*III(b) The dog that chased the cat that saw the rat barked.*
*III(c) The dog that chased the cat that saw the rat that ate the cheese barked.*

*IV(a) The cheese that the rat ate stank.*
*IV(b) ? The cheese that the rat that the cat saw ate stank.*
*IV(c) ?? The cheese that the rat that the cat that the dog chased saw ate stank.*

*V(a) That two plus two equals four surprised Jack.*
*V(b) ?That that two plus two equals four surprised Jack astonished Ingrid.*
*V(c) ??That that that two plus two equals four surprised Jack astonished Ingrid bothered Frank.*

*VI(a) Jack was surprised that two plus two equals four.*
*VI(b) Ingrid was astonished that Jack was surprised that two plus two equals four.*
*VI(c) Frank was bothered that Ingrid was astonished that Jack was surprised that two plus two equals four.*

In his paper (Morrill, 2000) Morrill provides an account of our processing preferences, based on our preference for a lower complexity profile. Given an analysis in Lambek calculus of a sentence depicted by a proof net, we have conclusions corresponding to the syntactic types of the words, and a single conclusion corresponding to $S$. All these conclusions are cyclically ordered. This cyclic order is easily turned into a linear order by choosing a conclusion and a rotation sense. Let us take the output conclusion $S$ as the first conclusion, and let us choose the clockwise rotation with respect to the proof nets of the previous sections. According to the way proof nets are drawn we thus are moving from right to left, and we successively meet $S$, the type of the first word, the type of the second word, etc.

Now let us define *the complexity of a place in between two words $w_n$ and $w_{n+1}$* ($w_0$ being a fake word corresponding to $S$) as the number of axioms $a - a^\perp$ which pass over this place, and such that the $a$ belongs to a conclusion which is, in the linear order, before the conclusion containing $a^\perp$.

Observe that this measure relies on the fact that Lambek calculus is an intuitionistic or polarized calculus in which $a$ and $a^\perp$ are of a different nature: indeed waiting for a category is not the same as providing a category. This measure also depends on the fact that we chose the output $S$ to be the first conclusion: this corresponds to the fact that when someone starts speaking we are expecting a sentence (it could be another category as well, but we still expect some well-formed utterance).

Now we can associate to a sentence with $n$ words a sequence of $n$ integers (since $S$ has been added there are $n$ places) called its *complexity profile*.

In all examples above, the preferred reading always has the lower profile (that is a profile which is always lower, or at least does not go as high) and sentences that are difficult to parse have a high profile.

Here we only present one example, in Figures 6.5 and 6.6, as the others provide excellent exercises (and drawing proof nets on the computer is painful).

| word | type $u$ | $u^\perp$ for constructing the proof net |
|---|---|---|
| someone (subject) | $S\,/\,(np\,\backslash\,S)$ | $(np^\perp \wp S)\otimes S^\perp$ |
| (object) | $(S\,/\,np)\,\backslash\,S$ | $S^\perp \otimes (S \wp np^\perp)$ |
| everyone (subject) | $S\,/\,(np\,\backslash\,S)$ | $(np^\perp \wp S)\otimes S^\perp$ |
| (object) | $(S\,/\,np)\,\backslash\,S$ | $S^\perp \otimes (S \wp np^\perp)$ |
| loves: | $(np\,\backslash\,S)\,/\,np$ | $np \otimes (S^\perp \otimes np)$ |

To complete the example, one should compute the semantics according to the algorithm given in Chapter 3.

## 6.7 Semantic Uses of Proof Nets

Once one is convinced of the relevance of proof nets for parsing, it is worth looking at what else can be achieved with proof nets, in order to avoid translating from one formalism into another, which can be unpleasant and algorithmically costly. A major advantage of categorial grammars is their relation to Montague semantics, and this link has been explored by many authors (Chapter 3 and the references therein provide an introduction to the subject).

As intuitionistic logic can be embedded into linear logic (Girard, 1987) the algorithm for computing semantic readings can be performed within linear logic. Indeed $\lambda$-terms can be depicted as proof nets, and $\beta$-reduction (or cut-elimination) for proof nets is extremely efficient. In particular the translation can limit the use of replication to its strict minimum. This has been explored by de Groote and Retoré (1996).

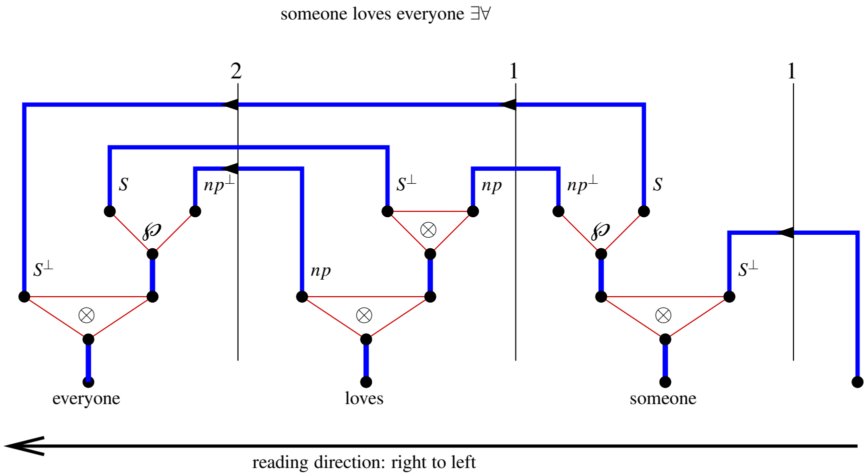



**Fig. 6.5.** "Someone loves everyone" with wide scope for someone. The complexity profile — read from right to left — is $1-1-2$.

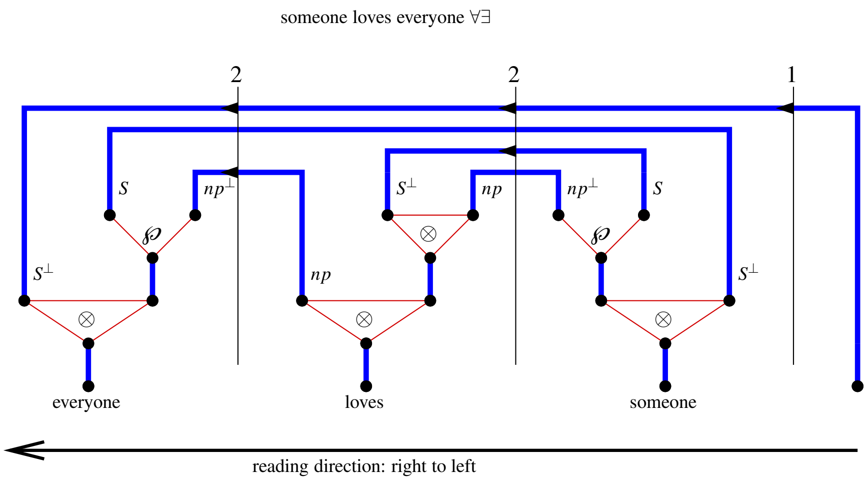



**Fig. 6.6.** "Someone loves everyone" with wide scope for everyone. The complexity profile — read from right to left — is $1-2-2$.

The correspondence between syntax and semantics with proof nets has been used for generation, firstly by Merenciano and Morrill (Merenciano and Morrill, 1996). Assuming that the semantics of a sentence is known, as well as the semantics of the words, the problem is to reconstruct a syntactic analysis out of this information. This mainly consists of reversing the process involved in the previous paragraph, which is essentially cut elimination. Using a representation of cut elimination by matrix computations (graphs can be viewed as matrices) Pogodalla has thus defined an efficient method for generation (Pogodalla, 2000c,a,b).

## 6.8 Concluding Remarks

This chapter has given a detailed treatment of proof nets for the associative Lambek calculus that have been discussed in Chapter 2. A central thesis has been that proof nets are not only interesting from a formal point of view, but also from the point of view of the nature of a parse. Indeed, proof nets identify proofs representing the same analysis thus avoiding the so-called spurious ambiguity problem. Therefore proof nets can be said to implement the very idea of *parsing-as-deduction*.

We have also touched upon several other aspects of proof nets: its connection to semantics and some suggestive evidence about proof net construction as a model for human sentence processing.

The proof nets presented here naturally suggest a further radicalisation: a formula can be depicted as a set of Red edges between its atoms, and the Blue edges are the atoms and a simple correctness criterion, ”every alternate elementary cycle contains a chord”, recognises exactly the proofs (Retoré, 2003). This way, the algebraic properties of the connectives, like associativity are interpreted by equality of the formulae, hence identifying even more proofs than usual proofnets with links. This was firstly done for commutative multiplicative linear logic, but it also works for non commutative logic like the Lambek calculus (Pogodalla and Retoré, 2004). In this setting as well, the cuts are a bit tricky to handle, and this is ongoing work.

## Exercises for Chapter 6

**Exercise 6.1.** Using proof nets, show whether or not the following sequents are derivable in multiplicative linear logic.

$\vdash (a \otimes b) \otimes c, (c^{\perp} \wp b^{\perp}) \wp a^{\perp}$
$\vdash (a \wp b) \otimes c, (c^{\perp} \wp b^{\perp}) \otimes a^{\perp}$
$\vdash c^{\perp}, ((a \wp a^{\perp}) \otimes b) \otimes d^{\perp}, b^{\perp} \wp (c \otimes d)$

**Exercise 6.2.** Section 6.1.3 defines a translation of Lambek calculus formulae into linear logic formulae using polarities. For all of the following formulae $F$, give both the translation $+F$ and $-F$.

$(np \setminus S) / np$
$(n \setminus n) / (S / np)$
$S / (np \setminus S)$
$(S / np) \setminus S$
$((np \setminus S) / np) \setminus (np \setminus S)$

**Exercise 6.3.** Proposition 6.2 on page 196 defines the sets of formulae $\mathsf{Li}^{\circ}$ and $\mathsf{Li}^{\bullet}$. For each of the following formulae, show if they are members $\mathsf{Li}^{\circ}$, members of $\mathsf{Li}^{\bullet}$ or if they are not a member of either set of formulae.

$(a \otimes b) \wp c^{\perp}$
$(a^{\perp} \wp b) \wp c^{\perp}$
$a^{\perp} \wp (b^{\perp} \wp c)$
$a \otimes (b^{\perp} \wp c)$
$a \otimes (b^{\perp} \otimes c)$

**Exercise 6.4.** Using Lambek calculus proof nets (refer to Definition 6.41 on page 220), show which of the following sequents are derivable.

$np \vdash S / (np \setminus S)$
$S / (np \setminus S) \vdash np$
$np, (np \setminus S) / np, np \vdash S$
$np, (np \setminus S) / np, S / (np \setminus S) \vdash S$
$np, (np \setminus S) / np, (S / np) \setminus S \vdash S$
$S / (np \setminus S), (np \setminus S) / np, ((np \setminus S) / np) \setminus (np \setminus S) \vdash S$

That is, translate each formula in the corresponding linear logic formula, compute the possible axiom linking $\sigma_{\Pi}$ and verify that all conditions of Definition 6.41, including $\varepsilon$-FREE, are satisfied.

**Exercise 6.5.** Look back to Figure 6.3 on page 224.

1. Verify it is a proof structure of $\vdash (b^{\perp} \wp b) \otimes (a^{\perp} \wp a)$ by assigning formulas to each node in the proof structure.
2. Remove the cut link from the figure and compute the sequents corresponding to the two substructures. Are both of these structures derivable?

3. Perform cut elimination on the proof structure of Figure 6.3. Can you remark anything special about the result of cut elimination? If so, what does this mean?
4. Draw the proof structure in such a way the the external face is on the outside of the proof structure. What — if anything — is different about this proof structure?
5. Give a correct (planar) proof structure for $\vdash (b^{\perp} \wp b) \otimes (a^{\perp} \wp a)$ and verify it satisfies all constraints.

**Exercise 6.6.** Following Johnson and Morrill (Johnson, 1998; Morrill, 2000, 2011), Section 6.6 states that the acceptability of sentences is related the "nesting" of axiom links to the (Figures 6.5 and 6.6, page 232 gives an example comparison). Compute a similar complexity profile for each of the other phenomena discussed at the beginning of Section 6.6, by assigning each of them appropriate lexical formulas and constructing the proof nets.

## References

Abrusci, V.M.: Phase semantics and sequent calculus for pure non-commutative classical linear logic. Journal of Symbolic Logic 56(4), 1403–1451 (1991)

Abrusci, V.M.: Non-commutative proof nets. In: Girard, et al., pp. 271–296 (1995)

Asperti, A.: A linguistic approach to dead-lock. Tech. Rep. LIENS 91-15, Dép. Maths et Info, Ecole Normale Supérieure, Paris (1991)

Asperti, A., Dore, G.: Yet Another Correctness Criterion for Multiplicative Linear Logic with Mix. In: Nerode, A., Matiyasevich, Y. (eds.) LFCS 1994. LNCS, vol. 813, pp. 34–46. Springer, Heidelberg (1994)

Bellin, G., Scott, P.J.: On the $\pi$-calculus and linear logic. Theoretical Computer Science 135, 11–65 (1994)

van Benthem, J.: Language in Action: Categories, Lambdas and Dynamic Logic. Studies in logic and the foundation of mathematics, vol. 130. North-Holland, Amsterdam (1991)

Bondy, J.A., Murty, U.S.R.: Graph Theory and Applications. Macmillan Press (1976)

Dalrymple, M., Lamping, J., Pereira, F., Saraswat, V.: Linear logic for meaning assembly. In: Morrill, G., Oehrle, R. (eds.) Formal Grammar, pp. 75–93. FoLLI, Barcelona (1995)

Danos, V.: La logique linéaire appliquée à l'étude de divers processus de normalisation et principalement du $\lambda$-calcul. Thèse de Doctorat, spécialité Mathématiques, Université Paris 7 (1990)

Danos, V., Regnier, L.: The structure of multiplicatives. Archive for Mathematical Logic 28, 181–203 (1989)

Diestel, R.: Graph Theory, 4th edn. Springer (2010)

Fleury, A.: La règle d'échange: logique linéaire multiplicative tréssée. Thèse de Doctorat, spécialité Mathématiques, Université Paris 7 (1996)

Fleury, A., Retoré, C.: The mix rule. Mathematical Structures in Computer Science 4(2), 273–285 (1994)

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

Girard, J.Y.: Linear logic: its syntax and semantics. In: Girard, et al., pp. 1–42 (1995)

Girard, J.Y., Lafont, Y., Regnier, L. (eds.): Advances in Linear Logic. London Mathematical Society Lecture Notes, vol. 222. Cambridge University Press (1995)

de Groote, P.: Linear Logic with Isabelle: Pruning the Proof Search Tree. In: Baumgartner, P., Posegga, J., Hähnle, R. (eds.) TABLEAUX 1995. LNCS (LNAI), vol. 918, pp. 263–277. Springer, Heidelberg (1995)

de Groote, P.: A Dynamic Programming Approach to Categorial Deduction. In: Ganzinger, H. (ed.) CADE 1999. LNCS (LNAI), vol. 1632, pp. 1–15. Springer, Heidelberg (1999)

de Groote, P., Lamarche, F.: Classical non-associative Lambek calculus. Studia Logica 71(3), 355–388 (2002)

de Groote, P., Retoré, C.: Semantic readings of proof nets. In: Kruijff, G.J., Morrill, G., Oehrle, D. (eds.) Formal Grammar, pp. 57–70. FoLLI, Prague (1996)

Guerrini, S.: Correctness of multiplicative proof nets is linear. In: 14th Symposium on Logic in Computer Science (LICS 1999), pp. 454–463. IEEE (1999)

Guerrini, S.: A linear algorithm for MLL proof net correctness and sequentialization. Theoretical Computer Science 412(20), 1958–1978 (2011); Girard's Festschrift

Johnson, M.E.: Proof nets and the complexity of processing center-embedded constructions. Journal of Logic Language and Information Special Issue on Recent Advances in Logical and Algebraic Approaches to Grammar 7(4), 433–447 (1998)

Lamarche, F.: Proof nets for intuitionistic linear logic: Essential nets. 35 page technical report available by FTP from the Imperial College archives (1994)

Lambek, J.: From categorial grammar to bilinear logic. In: Došen, K., Schröder-Heister, P. (eds.) Substructural Logics, pp. 207–237. Oxford University Press, Oxford (1993)

Lincoln, P., Mitchell, J., Scedrov, A., Shankar, N.: Decision problems for propositional linear logic. Annals of Pure and Applied Logic 56(1-3), 239–311 (1992)

Melliès, P.A.: 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 Mathematical Society Lecture Note, vol. 316, ch. 8, pp. 283–321. Cambridge University Press (2004)

Merenciano, J.M., Morrill, G.: Generation as Deduction on Labelled Proof Nets. In: Retoré, C. (ed.) LACL 1996. LNCS (LNAI), vol. 1328, pp. 310–328. Springer, Heidelberg (1997)

Métayer, F.: Homology of proof-nets. Prépublication 39, Equipe de Logique, Université Paris 7 (1993)

Moot, R.: Filtering axiom links for proof nets. In: Kallmeyer, L., Monachesi, P., Penn, G., Satta, G. (eds.) Proceedings of the 12th Conference on Formal Grammar (FG 2007). CSLI Publications, Dublin (2007) (to appear) ISSN 1935-1569

Morrill, G.: Memoisation of categorial proof nets: parallelism in categorial processing. In: Abrusci, V.M., Casadio, C. (eds.) Third Roma Workshop: Proofs and Linguistics Categories – Applications of Logic to the Analysis and Implementation of Natural Language. CLUEB, Bologna (1996)

Morrill, G.: Incremental processing and acceptability. Computational Linguistics 26(3), 319–338 (2000); preliminary version: UPC Report de Recerca LSI-98-46-R (1998)

Morrill, G.: Categorial Grammar: Logical Syntax, Semantics, and Processing. Oxford University Press (2011)

Morrill, G., Fadda, M.: Proof nets for basic discontinuous Lambek calculus. Journal of Logic and Computation 18(2), 239–256 (2008)

Murawski, A., Ong, C.H.: Dominator trees and fast verification of proof nets. In: 15th Symposium on Logic in Computer Science (LICS 2000), pp. 181–191. IEEE (2000)

Pentus, M.: Lambek calculus is NP-complete. Theoretical Computer Science 357(1), 186–201 (2006)

Pogodalla, S.: Generation in the Lambek calculus framework: an approach with semantic proof nets. In: Proceedings of NAACL 2000 (2000a)

Pogodalla, S.: Generation, Lambek calculus, montague's semantics and semantic proof nets. In: Proceedings of Coling 2000 (2000b)

Pogodalla, S.: Generation with semantic proof nets. Research Report 3878, INRIA (2000c), `http://hal.inria.fr/docs/00/07/27/75/PDF/RR-3878.pdf`

Pogodalla, S., Retoré, C.: Handsome non-commutative proof-nets: perfect matchings, series-parallel orders and hamiltonian circuits. Tech. Rep. RR-5409, INRIA, presented at Categorial Grammars (2004); to appear in the Journal of Applied Logic

Regnier, L.: Lambda calcul et réseaux. Thèse de doctorat, spécialité mathématiques, Université Paris 7 (1992)

Retoré, C.: Réseaux et séquents ordonnés. Thèse de Doctorat, spécialité Mathématiques, Université Paris 7 (1993)

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

Retoré, C.: Perfect matchings and series-parallel graphs: multiplicative proof nets as R&B-graphs. In: Girard, J.Y., Okada, M., Scedrov, A. (eds.) Linear 1996. Electronic Notes in Theoretical Science, vol. 3. Elsevier (1996)

Retoré, C.: A semantic characterisation of the correctness of a proof net. Mathematical Structures in Computer Science 7(5), 445–452 (1997)

Retoré, C.: Handsome proof-nets: perfect matchings and cographs. Theoretical Computer Science 294(3), 473–488 (2003), complete version RR-3652, http://hal.inria.fr/docs/00/07/30/20/PDF/RR-3652.pdf

Savateev, Y.: Product-Free Lambek Calculus Is NP-Complete. In: Artemov, S., Nerode, A. (eds.) LFCS 2009. LNCS, vol. 5407, pp. 380–394. Springer, Heidelberg (2008)

Yetter, D.N.: Quantales and (non-commutative) linear logic. Journal of Symbolic Logic 55, 41–64 (1990)