A dynamic programming approach to categorial deduction

Philippe de Groote

LORIA UMR n^o 7503 – INRIA, Campus Scientifique, B.P. 239
54506 Vandœuvre lès Nancy Cedex – France
e-mail: degroote@loria.fr

Abstract. We reduce the provability problem of any formula of the Lambek calculus to some context-free parsing problem. This reduction, which is based on non-commutative proof-net theory, allows us to derive an automatic categorial deduction algorithm akin to the well-known Cocke-Kasami-Younger parsing algorithm.

1 Introduction

Modern categorial grammars [8], which are intended to give a deductive account of grammatical composition, are based on substructural logics whose paradigm is the Lambek calculus [7]. Any parsing problem for a given categorial grammar gives rise to some decision problem in the substructural logic on which the grammar is based. Consequently, to design an efficient parsing algorithm for a categorial grammar amounts to design an efficient decision procedure for some logic akin to the Lambek calculus.

On the other hand, the Lambek calculus, which has been introduced four decades ago, appears *a posteriori* to be a non-commutative fragment of linear logic [4]. Consequently, it is possible to take advantage of Girard's proof-net theory in order to design automatic deduction procedures for the Lambek calculus.

In this paper, we introduce an original correctness criterion for non-commutative proof-nets, from which we derive a dynamic programming algorithm for deciding the provability of a Lambek sequent.

The paper is organised as follows. Section 2 is a short presentation of the Lambek calculus. In Section 3 we introduce an original notion of non-commutative proof-net, and we prove that this notion of proof-net corresponds to an actual notion of proof. In Section 4, we associate to each Lambek sequent a context-free grammar that allows to see the provability problem, for this sequent, as a rewriting problem. This gives rise to the algorithm described in Section 5. Finally, we conclude in Section 6.

It is to be noted that the theory of non-commutative proof-nets appeals to the theory of planar graphs. The formalisation of this theory relies on advanced geometrical concepts whose exposition is out of the scope of this paper. On the other hand, the elementary concepts of planar graph theory that we used (essentially the notion of *face*) are rather intuitive. Consequently, we have decided to illustrate the different concepts by several examples in order not to lose the reader into the formal details.

2 The Lambek Calculus

The Lambek calculus [7], which has been introduced as a logical basis for categorial grammars [8], corresponds exactly to the non-commutative intuitionistic multiplicative fragment of linear logic [4].

Given an alphabet of atomic formulas $\mathcal{A}$, the syntax of the formulas obeys the following grammar:

$$\mathcal{F} \ ::= \ \mathcal{A} \mid \mathcal{F} \bullet \mathcal{F} \mid \mathcal{F} \setminus \mathcal{F} \mid \mathcal{F} / \mathcal{F}$$

where formulas of the form $A \bullet B$ correspond to conjunctions (or products), formulas of the form $A \setminus B$ correspond to direct implications (i.e., A *implies* B), and formulas of the form A/B to retro-implication (i.e., A *is implied by* B).

Then, the deduction relation is specified by means of the following sequent calculus.

$$A \vdash A \quad \text{(ident)} \qquad \frac{\Gamma \vdash A \quad \Delta_1, A, \Delta_2 \vdash B}{\Delta_1, \Gamma, \Delta_2 \vdash B} \quad \text{(cut)}$$

$$\frac{\Gamma, A, B, \Delta \vdash C}{\Gamma, A \bullet B, \Delta \vdash C} \quad (\bullet \text{ left}) \qquad \frac{\Gamma \vdash A \quad \Delta \vdash B}{\Gamma, \Delta \vdash A \bullet B} \quad (\bullet \text{ right})$$

$$\frac{\Gamma \vdash A \quad \Delta_1, B, \Delta_2 \vdash C}{\Delta_1, \Gamma, A \setminus B, \Delta_2 \vdash C} \quad (\setminus \text{ left}) \qquad \frac{A, \Gamma \vdash B}{\Gamma \vdash A \setminus B} \quad (\setminus \text{ right})$$

$$\frac{\Gamma \vdash A \quad \Delta_1, B, \Delta_2 \vdash C}{\Delta_1, B/A, \Gamma, \Delta_2 \vdash C} \quad (/ \text{ left}) \qquad \frac{\Gamma, A \vdash B}{\Gamma \vdash B/A} \quad (/ \text{ right})$$

It is to be noted that the above system does not include any structural rule. In particular, the absence of an exchange rule is responsible for the non-commutativity of the connectives. This, in turn, explains the presence of two different implications.

Example 1. As an illustration, consider the sequent

$$(a/b) \bullet b,\ b \setminus (b \bullet (a \setminus a)) \vdash a,$$

which may be derived as follows:

$$\dfrac{\dfrac{b \vdash b \qquad \dfrac{\dfrac{b \vdash b \qquad \dfrac{a \vdash a \quad a \vdash a}{a,\ a \setminus a \vdash a}}{a/b,\ b,\ a \setminus a \vdash a}}{a/b,\ b \bullet (a \setminus a) \vdash a}}{a/b,\ b,\ b \setminus (b \bullet (a \setminus a)) \vdash a}}{(a/b) \bullet b,\ b \setminus (b \bullet (a \setminus a)) \vdash a}$$

3 Proof-nets for the Lambek calculus

Proof-nets are concrete structures that allow the proofs of linear logic to be represented geometrically [4]. As we mentioned, the Lambek calculus is a non-commutative fragment of linear logic. It is consequently possible to adapt the notion of proof net to the case of the Lambek calculus [6, 10]. To this end, we first give a translation of the Lambek calculus into (non-commutative) multiplicative linear logic.

The formulas of multiplicative linear logic are built upon a set of atomic propositions $\mathcal{A}$ according to the following grammar:

$$\mathcal{MF} \;::=\; \mathcal{A} \mid \mathcal{A}^{\perp} \mid \mathcal{MF} \otimes \mathcal{MF} \mid \mathcal{MF} \parr \mathcal{MF}$$

where the unary connective $^{\perp}$ denotes the linear negation, and the binary connectives $\otimes$ (*tensor*) and $\parr$ (*par*) correspond to multiplicative conjunction and disjunction respectively .

Definition 1. *The* translation $\mathcal{T}[\![\Gamma \vdash A]\!]$ *of a* Lambek sequent $\Gamma \vdash A$ *into a* sequence *of multiplicative formulas is defined as follows:*

$$\mathcal{T}[\![\Gamma \vdash A]\!] = \mathcal{T}^{-}[\![\Gamma]\!]^{-}, \mathcal{T}^{+}[\![A]\!]^{+}$$

where:

1. $\mathcal{T}^{+}[\![a]\!]^{+} = a$
2. $\mathcal{T}^{+}[\![A \bullet B]\!]^{+} = \mathcal{T}^{+}[\![B]\!]^{+} \otimes \mathcal{T}^{+}[\![A]\!]^{+}$
3. $\mathcal{T}^{+}[\![A \setminus B]\!]^{+} = \mathcal{T}^{+}[\![B]\!]^{+} \parr \mathcal{T}^{-}[\![A]\!]^{-}$
4. $\mathcal{T}^{+}[\![A / B]\!]^{+} = \mathcal{T}^{-}[\![B]\!]^{-} \parr \mathcal{T}^{+}[\![A]\!]^{+}$
5. $\mathcal{T}^{-}[\![a]\!]^{-} = a^{\perp}$
6. $\mathcal{T}^{-}[\![A \bullet B]\!]^{-} = \mathcal{T}^{-}[\![A]\!]^{-} \parr \mathcal{T}^{-}[\![B]\!]^{-}$
7. $\mathcal{T}^{-}[\![A \setminus B]\!]^{-} = \mathcal{T}^{+}[\![A]\!]^{+} \otimes \mathcal{T}^{-}[\![B]\!]^{-}$
8. $\mathcal{T}^{-}[\![A / B]\!]^{-} = \mathcal{T}^{-}[\![A]\!]^{-} \otimes \mathcal{T}^{+}[\![B]\!]^{+}$
9. $$\mathcal{T}^{-}[\![\Gamma, A]\!]^{-} = \mathcal{T}^{-}[\![\Gamma]\!]^{-}, \mathcal{T}^{-}[\![A]\!]^{-}$$

It is important to note that the translation $\mathcal{T}[\![\Gamma \vdash A]\!]$ does not define a set but a sequence of formulas. This is due to to fact that the order between the formulas of a Lambek sequent is relevant.

Example 2. Consider the sequent of Example 1. Its translation is the following:

$$\mathcal{T}[\![(a/b) \bullet b,\, b \setminus (b \bullet (a \setminus a)) \vdash a]\!] = (a^{\perp} \otimes b) \parr b^{\perp}, b \otimes (b^{\perp} \parr (a \otimes a^{\perp})), a$$

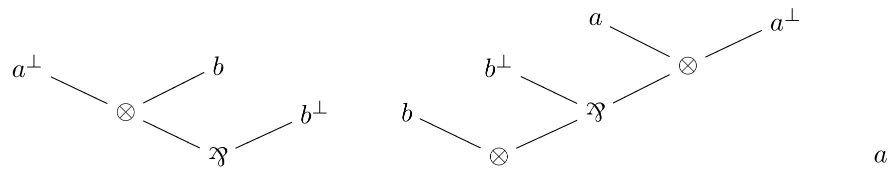
The definition of a multiplicative proof-net procceds as follows. The notion of proof-structure, which corresponds to a class of graphs decorated with formulas, is first defined. These proof-structures are intended to represents proofs. It is not the case, however, that they all correspond to actual proofs. It is therefore necessary to give some further criterion in order to distinguish the *correct* proof-structures, which are called proof-nets, from the other ones.

The notion of multiplicative proof-net may be adapted to the Lambek calculus by stating an additional condition that ensures non-commutativity. We do not follow this approach. The definition that we give is based on a correctness criterion that is intrinsically non-commutative. This new criterion, which has been especially devised in order to prove the correctness of the algorithm of Section 5, is a refinement of a similar criterion due to Fleury [3].

Proof-nets and proof-structure being a sort of graph, we use freely elementary graph-theoretic concepts that can be found in any textbook. In particular, the terminology we adopt is taken from [2]. We also take for granted the notion of parse tree of a formula. The leaves of such a parse tree are decorated with literals (i.e., atomic propositions, or negations of atomic propositions) and its internal nodes are decorated either with the connective $\otimes$ or the connective $⅋$. These internal nodes will be called $\otimes$- and $⅋$-nodes, respectively.

Definition 2. *Let $\Gamma \vdash A$ be a Lambek sequent. The* proof-frame *of $\Gamma \vdash A$ consists of (the sequence of) the parse-trees of the formulas in $\mathcal{T}[\![\Gamma \vdash A]\!]$. The roots of these parse trees are called the* conclusions *(or the conclusive nodes) of the proof-frame.*

Example 3. The proof-frame of the sequent of Example 1 is the following:



Let us associate two *polarised atoms* a^+ and a^- to each atomic formula a, and let Σ_1 be the set of these polarised atoms. The next step in defining a notion of proof-structure for the Lambek calculus consists in introducing the concept of *well-bracketed matching* on a word of Σ_1^*.

Definition 3. *Consider a word $\omega = \omega_1\omega_2 \ldots \omega_n$ of polarised atoms. We define a well-bracketed matching on ω to be a permutation p on the set of ω_i's such that:*

1. *$(\forall i, j \leq n)\ p(\omega_i) = \omega_j$ implies ($\omega_i = a^+$ and $\omega_j = a^-$) or ($\omega_i = a^-$ and $\omega_j = a^+$), for some atomic formula a.*
2. *$(\forall i, j \leq n)\ p(\omega_i) = \omega_j$ implies $p(\omega_j) = \omega_i$,*
3. *$(\forall i, j, k, l \in n)\ p(\omega_i) = \omega_j$, $p(\omega_k) = \omega_l$, and $i < k$ imply $l < j$.*

Conditions 1 and 2, in this definition, formalise the idea that a matching consists in grouping by pairs atoms of opposite polarities. Condition 3 corresponds to a notion of well-bracketed structures by forbidding configurations such as $\cdots \underset{i}{[} \cdots \underset{k}{(} \cdots \underset{j}{]} \cdots \underset{l}{)} \cdots$

Let $\Gamma \vdash A$ be a Lambek sequent. We write (A/Γ) to denote the sequent (consisting of one formula) obtained by applying Rule (/ right) to $\Gamma \vdash A$ as many times as possible. It is well-known that $\Gamma \vdash A$ is provable if and only if (A/Γ) is. We now associate, to each Lambek sequent, a word of polarised atoms as follows.

Definition 4. *The* word of polarised atoms $\mathcal{V}_1[\![\Gamma \vdash A]\!]$ *associated to a* Lambek sequent $\Gamma \vdash A$ *is defined as* $\mathcal{V}_1[\![\Gamma \vdash A]\!] = \mathcal{W}[\![\mathcal{T}[\![A/\Gamma]\!]]\!]$*, where:*

1. $\mathcal{W}[\![a]\!] = a^+$
2. $\mathcal{W}[\![a^\perp]\!] = a^-$
3. $\mathcal{W}[\![\alpha \mathbin{⅋} \beta]\!] = \mathcal{W}[\![\alpha]\!]\, \mathcal{W}[\![\beta]\!]$
4. $\mathcal{W}[\![\alpha \otimes \beta]\!] = \mathcal{W}[\![\alpha]\!]\, \mathcal{W}[\![\beta]\!]$

Let us interpret a^+ and a^- respectively as a and $a^\perp$, which is consistent with Equations 1 and 2 in the above definition. Then the word associated to a sequent corresponds exactly to the sequence of the leaves of the proof-frame of this sequent.

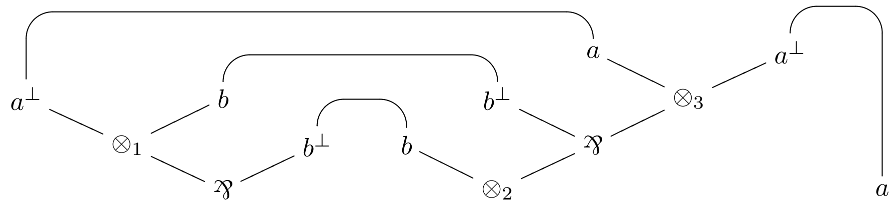
We are now in a position of defining a notion of proof-structure for the Lambek calculus.

Definition 5. *Let $\Gamma \vdash A$ be a Lambek sequent. A proof-structure of $\Gamma \vdash A$ (if any) is a simple decorated graph made of:*

1. *the proof-frame of $\Gamma \vdash A$;*
2. *a perfect matching on the leaves of this proof-frame that corresponds to a well-bracketed matching on $\mathcal{V}_1[\![\Gamma \vdash A]\!]$.*

The conclusions of the proof-structure are defined to be the conclusions of the proof-frame. The edges defining the perfect matching on the leaves of the proof-frame are called the axiom links of the proof-structure.

Example 4. The following graph is a proof-structure for the sequent of Example 1.



In order to state the definition of a proof-net, we need some elementary concepts of planar graph-theory. A graph is said to be planar if it may be represented on a plane in such a way that no two edges cross one another. Such a representation is called a *topological planar graph.* A *face* of a topological planar graph is defined to be a region of the plane bounded by edges in such a way that any two points of this region may be connected by a continuous curve that does not cross any edge.

It is is easy to see that the proof-structures of Definition 5 are planar graphs (this is due to the well-bracketing condition). Consider, for instance, the proof-structure of Example 4. It has three faces: two *bounded faces* that are contained within elementary cycles — respectively, (a, $a^\perp$, $\otimes_1$, b, $b^\perp$, ⅋, $\otimes_3$) and ($b^\perp$, b, $\otimes_1$, ⅋, $b^\perp$, b, $\otimes_2$, ⅋) —, and one *unbounded face* that corresponds to the region of the plane that is "outside of the proof-structure".

While the notion of face is proper to the notion of planar graph, the notion of bounded and unbounded face depends of the particular topological planar representation under consideration. Consequently, from now on, when speaking of a proof-net we mean its *natural topological representation*[1] as illustrated by example 4. This convention allows us to define the *faces of a proof-structure* as the bounded faces of the corresponding topological planar graph.

Let P be a proof-structure, F be a face of P and n be a $\parr$- or a $\otimes$-node of P. We say that F *contains* n if and only if n and its two daughter nodes belong to the boundary of F. Remark that some nodes may belong to the boundaries of two different faces. However, according to the present definition, any node *is contained* in at most one face. Finally, let t and u be two $\otimes$-nodes of P. We say that t is dominated by u (and we write $t \prec u$) if and only if t is the ancestor of a $\parr$-node that is contained in the same face as u.

We now define our notion of proof-net.

Definition 6. *Let $\Gamma \vdash A$ be a Lambek sequent. A proof-net of $\Gamma \vdash A$ (if any) is a proof-structure P of $\Gamma \vdash A$ such that*

1. *each face of P contains exactly one $\parr$-node;*
2. *each $\parr$-node is contained in a face of P;*
3. *the relation of dominance $\prec$ between the $\otimes$-nodes of P is acyclic, i.e., its transitive closure $\prec^+$ is irreflexive.*

Example 5. The proof-structure of example 4 is a proof-net. Indeed, each face contains exactly one $\parr$-node. There are no other $\parr$-nodes. The dominance relation consists simply of $\otimes_2 \prec \otimes_1$, whose transitive closure is clearly acyclic.

In constructing a proof-net, the difficulty consists in guessing an appropriate set of axiom links. A possible solution to this problem is to generate all the possible sets of axiom links and to check for each corresponding proof structure whether the proof net correctness criterion is satisfied or not. A more clever way of proceeding is to construct the set of axiom links incrementally so that the correctness criterion is ensured by construction. This is precisely what our algorithm is doing. Consequently, in order to prove its correctness, we will need a notion of partial proof-net.

Let us define a *module* to be a simple decorated graph made of a proof-frame and a partial perfect matching that obeys the conditions of Definition 5. In other words a module is a proof-structure from which some axiom links have been erased. Now, given a module M, we define an *possibly open face* to be a set of nodes that belong to the boundary of the same face in any proof-structure that contains M as a subgraph. If the boundary of a possibly open face is cyclic, we call it an *actual face.* If it is acyclic, we call it an *open face.*

The notion of possibly open face allows the relation of dominance to be defined on modules. This, in turn, allows us to define a *correct module.* to be a

[1] That is the representation obtained by drawing trees with their root at the bottom, by drawing left and right daughter-nodes respectively on the upper left and upper right of their mother, and by drawing sequences of trees from left to right.

module whose actual faces contain exactly one $\parr$-node, whose open faces contain at most one $\parr$-node, and for which the transitive closure of the dominance relation is acyclic.

We end this section by proving that the proof-nets of definition 6 correspond actually to a notion of proof for the Lambek calculus. In other words, we prove that any Lambek sequent $\Gamma \vdash A$ is provable if and only if there exists a proof-net for it.

Proposition 1. *Let $\Gamma \vdash A$ be a provable Lambek sequent. Then there exists a proof-net whose conclusions are $\mathcal{T}[\![\Gamma \vdash A]\!]$.*

Proof. The proof consists in a straightforward induction on the sequent calculus derivation. □

To prove the converse of this proposition, which corresponds to Girard's sequentialisation theorem [4, 5], we establish the following key lemma.

Lemma 1. *Let P be a proof-net that does not contain any conclusive $\parr$-node. If P contains at least one $\otimes$-node then it contains a conclusive $\otimes$-node whose removal splits P into two disconnected proof-nets.*

Proof. Consider some conclusive $\otimes$-node t_1 of P (there is at least one). If the removal of t_1 disconnects P, we are done. Otherwise, t_1 must be contained in a face of P. Consider the $\parr$-node p that is contained in this face. p cannot be a conclusion of P. Hence it must be the descendant of some conclusive $\otimes$-node t_2, and we have $t_2 \prec t_1$. By iterating this process, which terminates because of the acyclicity of the transitive closure of $\prec$, we eventually find a splitting $\otimes$. □

Proposition 2. *Let P be a proof-net whose conclusions are $\mathcal{T}[\![\Gamma \vdash A]\!]$. Then the Lambek sequent $\Gamma \vdash A$ is provable.*

Proof. The proof is done by induction on the number of nodes in P. If P is made of a single axiom link then it corresponds to an axiom $a \vdash a$. If P has at least one conclusive $\parr$-node, the induction is straightforward. Finally, if P does not consist of a single axiom link and does not contain any conclusive $\parr$-node, we apply Lemma 1. □

Proposition 3. *Any Lambek sequent is provable if and only if there exist a proof-net of it.* □

4 Grammars of axiom links

As we already stressed, to construct a proof-net consists essentially in finding an appropriate set of axiom links. This set must obey three conditions:

A. it must induce a well-bracketing on the leaves of the given proof-frame,
B. it must give rise to a proof-structure whose faces contain exactly one $\parr$,
C. it must induce a dominance relation σ whose transitive closure σ^+ is acyclic.

Given a Lambek sequent $\Gamma \vdash A$, it is easy to show that there exist a set of axiom links satisfying Condition A if and only if

$$\mathcal{V}_1[\![\Gamma \vdash A]\!] \quad \to^* \quad S \qquad\qquad (*)$$

according to System R_1 that is defined by the following rules:

$$\begin{array}{rclr@{\qquad}rclr}
T & \to & S & (1) & a^- \, a^+ & \to & T & (4)\\
T \, S & \to & S & (2) & a^+ \, S \, a^- & \to & T & (5)\\
a^+ \, a^- & \to & T & (3) & a^- \, S \, a^+ & \to & T & (6)
\end{array}$$

Indeed, the above system corresponds to a context-free grammar whose terminal alphabet is Σ_1 and whose non-terminal alphabet is the set $\{S, T\}$. Then, any rewriting sequence such as $(*)$ induces the well-bracketed matching that matches together two atoms of opposite polarities if and only if they are rewritten at the same time by an instance of Rule (3), (4), (5) or (6).

We now transform, step by step, the above grammar in order to take Conditions B and C into account.

Let us define a new terminal alphabet Σ_2 by associating to each $a^+ \in \Sigma_1$ (respectively, $a^- \in \Sigma_1$) two different symbols, namely, $a^+_{\parr}$, $a^+_{\otimes}$ (respectively, $a^-_{\parr}$, and $a^-_{\otimes}$). Similarly, we consider the non-terminal alphabet $\{S_{\parr}, S_{\otimes}, T_{\parr}, T_{\otimes}\}$. Then, we associate to each Lambek sequent $\Gamma \vdash A$ a word of Σ_2^* as follows:

$$\mathcal{V}_2[\![\Gamma \vdash A]\!] = \mathcal{W}[\![\mathcal{T}[\![A/\Gamma]\!]]\!]_{\parr}$$

where the transformation $\mathcal{W} : \mathcal{MF} \times \{\parr, \otimes\} \to \Sigma_2^*$ obeys the following equations:

1. $\mathcal{W}[\![a]\!]\mathbf{c} = a^+_{\mathbf{c}}$
2. $\mathcal{W}[\![a^\perp]\!]\mathbf{c} = a^-_{\mathbf{c}}$
3. $\mathcal{W}[\![\alpha \parr \beta]\!]\mathbf{c} = \mathcal{W}[\![\alpha]\!]_{\parr} \, \mathcal{W}[\![\beta]\!]\mathbf{c}$
4. $\mathcal{W}[\![\alpha \otimes \beta]\!]\mathbf{c} = \mathcal{W}[\![\alpha]\!]_{\otimes} \, \mathcal{W}[\![\beta]\!]\mathbf{c}$

with $\mathbf{c}$ ranging over $\{\parr, \otimes\}$. The intended meaning of this definition is the following: $\mathcal{V}_2[\![\Gamma \vdash A]\!]$ corresponds to the sequence of the leaves of the proof-frame of $\Gamma \vdash A$ with, for each leave, a subscript $\parr$ or $\otimes$ indicating respectively whether the open face immediately on the right of the leave contains or does not contain a $\parr$-node.

Let $\Gamma \vdash A$ be a Lambek sequent. We claim that there exists a proof-structure of $\Gamma \vdash A$ whose each face contains exactly one $\parr$-node if and only if

$$\mathcal{V}_2[\![\Gamma \vdash A]\!] \quad \to^* \quad S_{\parr} \qquad\qquad (**)$$

according System R_2 that is defined by the following rules:

$$\begin{array}{rclr@{\qquad}rclr}
T_{\mathbf{c}} & \to & S_{\mathbf{c}} & (1) & a^-_{\parr} \, a^+_{\mathbf{c}} & \to & T_{\mathbf{c}} & (6)\\
T_{\otimes} \, S_{\parr} & \to & S_{\parr} & (2) & a^+_{\parr} \, S_{\otimes} \, a^-_{\mathbf{c}} & \to & T_{\mathbf{c}} & (7)\\
T_{\parr} \, S_{\otimes} & \to & S_{\parr} & (3) & a^-_{\parr} \, S_{\otimes} \, a^+_{\mathbf{c}} & \to & T_{\mathbf{c}} & (8)\\
T_{\otimes} \, S_{\otimes} & \to & S_{\otimes} & (4) & a^+_{\otimes} \, S_{\parr} \, a^-_{\mathbf{c}} & \to & T_{\mathbf{c}} & (9)\\
a^+_{\parr} \, a^-_{\mathbf{c}} & \to & T_{\mathbf{c}} & (5) & a^-_{\otimes} \, S_{\parr} \, a^+_{\mathbf{c}} & \to & T_{\mathbf{c}} & (10)
\end{array}$$

where $\mathbf{c}$ ranges over $\{\parr, \otimes\}$.

The only-if-part of our claim is easy to show. In order to prove the if-part, we must show that the proof-frame of $\Gamma \vdash A$ admits a set of axiom links satisfying both Conditions A and B. The proof that (∗∗) induces a set of axiom links satisfying Condition A is similar to the case of System R_1. The proof that this set of axioms satisfies Condition B is based on the following facts:

- for each (occurrence of a) non-terminal symbol N involved in (∗∗) there exist $\omega_1, \omega_2, \omega_3 \in \Sigma_2^*$ such that: (1) $\mathcal{V}_2[\![\Gamma \vdash A]\!] = \omega_1\omega_2\omega_3$; (2) $\omega_2 \rightarrow^* N$; (3) $\omega_1 N \omega_2 \rightarrow^* S_{\parr}$. consequently, to each (occurrence of a) non-terminal symbol N, one may associate the portion of the proof-frame that corresponds to ω_2, the set of axiom links that is induced by $\omega_2 \rightarrow^* N$, and the module that is made of the proof-frame together with this set of axiom links;
- to each (occurrence of a) non-terminal symbol N involved in (∗∗) corresponds (a portion of) an open face of the associated module; this open face corresponds to the portion associated to N together with the leaf immediately on the right of this portion;
- a non-terminal symbol has $\parr$ as a subscript if and only if the corresponding open face contains a $\parr$-node (this property is preserved by all the rules of R_2);
- Rules (5) to (10) ensure that each open face that is turned into an actual face by adding an axiom link contains a $\parr$-node.

We will illustrate the above facts by examples, but first we transform System R_2 in order to take Condition C into account.

From now on, we distinguish the different occurrences of $\otimes$ within a multiplicative formula by means of indices. For instance, we write $(a \otimes_1 (b \otimes_2 c))$ instead of $(a \otimes (b \otimes c))$. Then we consider an infinite set of constants $\mathcal{K} = \{k_1, k_2, k_3, ...\}$, and we construct the alphabet Σ_3 by associating a new symbol $s[\Gamma]$ to any $s \in \Sigma_2$, $\Gamma \subset \mathcal{K}$. Similarly, we associate four different non-terminal symbols $S_{\parr}$, $S_{\otimes}$, $T_{\parr}$, $T_{\otimes}\}$ to any $\Gamma \cup \mathcal{K}$, $\sigma \subset \mathcal{K} \times \mathcal{K}$, such that the transitive closure σ^+ of the relation σ is acyclic. Then, we associate to each Lambek sequent $\Gamma \vdash A$ a word of Σ_3^* defined as follows:

$$\mathcal{V}_3[\![\Gamma \vdash A]\!] = \mathcal{W}[\![\mathcal{T}[\![A/\Gamma]\!]]\!]_{\parr}\, \emptyset\, \emptyset$$

where the transformation $\mathcal{W} : \mathcal{MF} \times \{\parr, \otimes\} \times 2^{\mathcal{K}} \times 2^{\mathcal{K}\times\mathcal{K}} \rightarrow \Sigma_3^*$ obeys the following equations:

1. $\mathcal{W}[\![a]\!]\mathbf{c}\, \Gamma\, \Delta = a^+_{\mathbf{c}}[\Gamma]$
2. $\mathcal{W}[\![a^\perp]\!]\mathbf{c}\, \Gamma\, \Delta = a^-_{\mathbf{c}}[\Gamma]$
3. $\mathcal{W}[\![\alpha \parr \beta]\!]\mathbf{c}\, \Gamma\, \Delta = \mathcal{W}[\![\alpha]\!]_{\parr}\, \Delta\, \Delta\;\; \mathcal{W}[\![\beta]\!]\mathbf{c}\, \Gamma\, \Delta$
4. $\mathcal{W}[\![\alpha \otimes_i \beta]\!]\mathbf{c}\, \Gamma\, \Delta = \mathcal{W}[\![\alpha]\!]_{\otimes}\, \{\mathrm{k}_i\}\, (\Delta \cup \{\mathrm{k}_i\})\;\; \mathcal{W}[\![\beta]\!]\mathbf{c}\, \Gamma\, (\Delta \cup \{\mathrm{k}_i\})$

with $\mathbf{c}$ ranging over $\{\parr, \otimes\}$. The interpretation of this definition is the following. The different constants $k_1, k_2, \ldots$ occurring in $\mathcal{V}_3[\![\Gamma \vdash A]\!]$ correspond to the different occurrences of $\otimes$ in $\Gamma \vdash A$. Then a symbol $a^+_{\parr}[\Gamma]$ or $a^-_{\parr}[\Gamma]$ occurring in $\mathcal{V}_3[\![\Gamma \vdash A]\!]$ corresponds to a leaf whose right open face contains one $\parr$-node, and Γ corresponds to the set of $\otimes$-nodes that are ancestors of this $\parr$-node. On the other hand, a symbol $a^+_\otimes[\Gamma]$ or $a^-_\otimes[\Gamma]$ corresponds to a leave whose right open face contains one $\otimes$-node (and, consequently, does not contain a $⅋$-node), then Γ is a singleton corresponding to this $\otimes$-node.

Definition 7. *The rewriting system R_3 is defined as follows:*

$$
\begin{array}{rcll}
T_{\mathbf{c}}[\Gamma,\sigma] & \rightarrow & S_{\mathbf{c}}[\Gamma,\sigma] & (1)\\
T_\otimes[\Gamma_1,\sigma_1]\,S_⅋[\Gamma_2,\sigma_2] & \rightarrow & S_⅋[\Gamma_2,\sigma_1\cup\sigma_2\cup(\Gamma_2\times\Gamma_1)] & (2)\\
& & \textit{provided that } (\sigma_1\cup\sigma_2\cup(\Gamma_2\times\Gamma_1))^+ \textit{ is acyclic.} & \\
T_⅋[\Gamma_1,\sigma_1]\,S_\otimes[\Gamma_2,\sigma_2] & \rightarrow & S_⅋[\Gamma_1,\sigma_1\cup\sigma_2\cup(\Gamma_1\times\Gamma_2)] & (3)\\
& & \textit{provided that } (\sigma_1\cup\sigma_2\cup(\Gamma_1\times\Gamma_2))^+ \textit{ is acyclic.} & \\
T_\otimes[\Gamma_1,\sigma_1]\,S_\otimes[\Gamma_2,\sigma_2] & \rightarrow & S_\otimes[\Gamma_1\cup\Gamma_2,\sigma_1\cup\sigma_2] & (4)\\
& & \textit{provided that } (\sigma_1\cup\sigma_2)^+ \textit{ is acyclic.} & \\
a^+_⅋[\Gamma_1]\,a^-_{\mathbf{c}}[\Gamma_2] & \rightarrow & T_{\mathbf{c}}[\Gamma_2,\emptyset] & (5)\\
a^-_⅋[\Gamma_1]\,a^+_{\mathbf{c}}[\Gamma_2] & \rightarrow & T_{\mathbf{c}}[\Gamma_2,\emptyset] & (6)\\
a^+_⅋[\Gamma_1]\,S_\otimes[\Gamma_2,\sigma_2]\,a^-_{\mathbf{c}}[\Gamma_3] & \rightarrow & T_{\mathbf{c}}[\Gamma_3,\sigma_2\cup(\Gamma_1\times\Gamma_2)] & (7)\\
& & \textit{provided that } (\sigma_2\cup(\Gamma_1\times\Gamma_2))^+ \textit{ is acyclic.} & \\
a^-_⅋[\Gamma_1]\,S_\otimes[\Gamma_2,\sigma_2]\,a^+_{\mathbf{c}}[\Gamma_3] & \rightarrow & T_{\mathbf{c}}[\Gamma_3,\sigma_2\cup(\Gamma_1\times\Gamma_2)] & (8)\\
& & \textit{provided that } (\sigma_2\cup(\Gamma_1\times\Gamma_2))^+ \textit{ is acyclic.} & \\
a^+_\otimes[\Gamma_1]\,S_⅋[\Gamma_2,\sigma_2]\,a^-_{\mathbf{c}}[\Gamma_3] & \rightarrow & T_{\mathbf{c}}[\Gamma_3,\sigma_2\cup(\Gamma_2\times\Gamma_1)] & (9)\\
& & \textit{provided that } (\sigma_2\cup(\Gamma_2\times\Gamma_1))^+ \textit{ is acyclic.} & \\
a^-_\otimes[\Gamma_1]\,S_⅋[\Gamma_2,\sigma_2]\,a^+_{\mathbf{c}}[\Gamma_3] & \rightarrow & T_{\mathbf{c}}[\Gamma_3,\sigma_2\cup(\Gamma_2\times\Gamma_1)] & (10)\\
& & \textit{provided that } (\sigma_2\cup(\Gamma_2\times\Gamma_1))^+ \textit{ is acyclic.} &
\end{array}
$$

Proposition 4. *Let $\Gamma \vdash A$ be a Lambek sequent. Then $\Gamma \vdash A$ is provable if and only if*

$$\mathcal{V}_3[\![\Gamma \vdash A]\!] \quad \rightarrow^* \quad S_⅋[\emptyset,\sigma] \qquad (***)$$

according to system R_3, for some $\sigma \subset \mathcal{K}\times\mathcal{K}$ whose transitive closure is acyclic.

Proof. (sketch). By Proposition 3, we have that $\Gamma \vdash A$ is provable if and only if there exists a proof-net of it. Given such a proof-net, it easy to construct a rewriting such as $(***)$ by taking σ to be the dominance relation of the proof-net. This proves the only-if part of the proposition.

Conversely, we show that the set of axiom links induced by $(***)$ satisfies Conditions A, B, and C. To show that Conditions A and B are satisfied, one proceeds similarly to cases of Systems R_1 and R_2. To show that Condition C is satisfied, one establishes the following invariants:

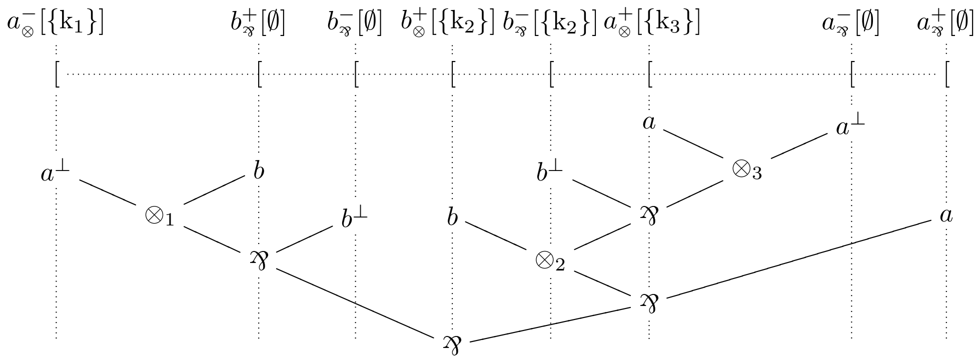
- *the set Γ appearing in a non-terminal symbol $S_⅋[\Gamma,\sigma]$ or $T_⅋[\Gamma,\sigma]$ corresponds the $\otimes$-nodes that are ancestors of the $⅋$-nodes contained in the open face associated to this non-terminal symbol;*
- *the set Γ appearing in a non-terminal symbol $S_\otimes[\Gamma,\sigma]$ or $T_\otimes[\Gamma,\sigma]$ corresponds the $\otimes$-nodes that are contained in the open face associated to this non-terminal symbol;*

– *the set σ appearing in a non-terminal symbol $S_{\mathbf{c}}[\Gamma, \sigma]$ or $T_{\mathbf{c}}[\Gamma, \sigma]$ corresponds the dominance relation of the module associated to this non-terminal symbol.* □

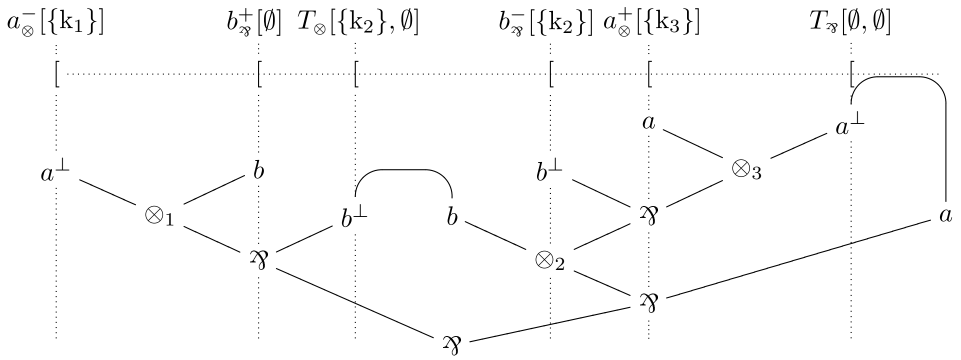
Example 6. Consider again the sequent of Example 1. Its provability may be established by the following rewriting (where applications of Rule (1) have been left implicit):

$$
\begin{array}{l}
a^-_{\otimes}[\{k_1\}]\ \ b^+_{\parr}[\emptyset]\ \ \underbrace{b^-_{\parr}[\emptyset]\ \ b^+_{\otimes}[\{k_2\}]}\ \ b^-_{\parr}[\{k_2\}]\ \ a^+_{\otimes}[\{k_3\}]\ \ \underbrace{a^-_{\parr}[\emptyset]\ \ a^+_{\parr}[\emptyset]} \\
a^-_{\otimes}[\{k_1\}]\ \ \underbrace{b^+_{\parr}[\emptyset]\ \ T_{\otimes}[\{k_2\},\emptyset]\ \ b^-_{\parr}[\{k_2\}]}\ \ a^+_{\otimes}[\{k_3\}]\ \ T_{\parr}[\emptyset,\emptyset] \\
\underbrace{a^-_{\otimes}[\{k_1\}]\ \ T_{\parr}[\{k_2\},\emptyset]\ \ a^+_{\otimes}[\{k_3\}]}\ \ T_{\parr}[\emptyset,\emptyset] \\
\underbrace{T_{\otimes}[\{k_3\},\{(k_2,k_1)\}]\ \ T_{\parr}[\emptyset,\emptyset]} \\
S_{\parr}[\emptyset,\{(k_2,k_1)\}]
\end{array}
$$

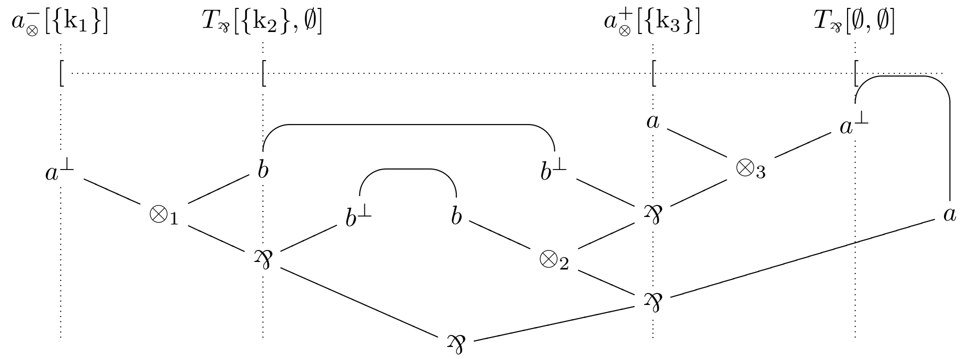
Now, the different rewriting steps of the above derivation may be interpreted as incremental steps in the construction of a proof-net. We starts with the proof-frame:



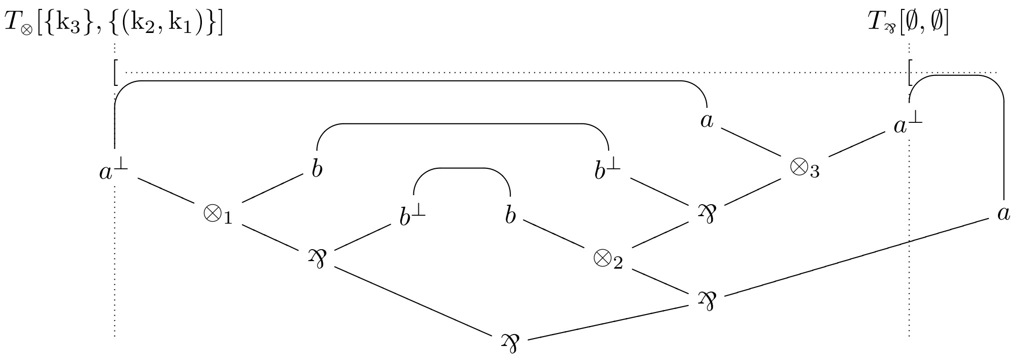
The first rewriting steps, $b^-_{\parr}[\emptyset]\, b^+_{\otimes}[\{k_2\}] \to T_{\otimes}[\{k_2\},\emptyset]$ and $a^-_{\parr}[\emptyset]\, a^+_{\parr}[\emptyset] \to T_{\parr}[\emptyset,\emptyset]$, yield the following module:



Then, the interpretation of $b^{+}_{\parr}[\emptyset]\, S_{\otimes}[\{\mathrm{k}_2\},\emptyset]\, b^{-}_{\parr}[\{\mathrm{k}_2\}] \to T_{\parr}[\{\mathrm{k}_2\},\emptyset]$ gives rise to a supplementary axiom link:



The next step, $a^{-}_{\otimes}[\{\mathrm{k}_1\}]\, S_{\parr}[\{\mathrm{k}_2\},\emptyset]\, a^{+}_{\otimes}[\{\mathrm{k}_3\} \to T_{\otimes}[\{\mathrm{k}_3\},\{(\mathrm{k}_2,\mathrm{k}_1)\}]$, is interpreted similarly:



Finally, the last step, $T_{\otimes}[\{\mathrm{k}_3\},\{(\mathrm{k}_2,\mathrm{k}_1)\}]\, S_{\parr}[\emptyset,\emptyset] \to S_{\parr}[\emptyset,\{(\mathrm{k}_2,\mathrm{k}_1)\}]$, establishes that the above module is indeed a proof-net.

5 The proof-net construction algorithm

The fact that we have reduced the decision problem of the Lambek calculus to a context-free parsing problem allows us to take advantage of the dynamic programming techniques that are used to recognize the context-free languages. In particular, we present an algorithm derived from the well-known Cocke-Kasami-Younger procedure.

This algorithm, which is based on the Chomsky normal form of System R_3 (See Appendix A), constructs an upper-triangular recognition matrix A whose entries are sets of non-terminal symbols. The initial conditions are the following: Given a sequent $\Gamma \vdash A$, $\omega = \omega_1\omega_2\ldots\omega_n$ contains $\mathcal{V}_3[\![\Gamma \vdash A]\!]$, and all the entries of A are empty.

```
  for i := 1 to n − 1 do
(* Rules 4 and 5 *)
    if      ω_i = a^+_⅋[Γ_1] and ω_{i+1} = a^-_c[Γ_2] then A_{i,i} := {T_c[Γ_2, ∅], S_c[Γ_2, ∅]}
(* Rules 6 and 7 *)
    else if ω_i = a^-_⅋[Γ_1] and ω_{i+1} = a^+_c[Γ_2] then A_{i,i} := {T_c[Γ_2, ∅], S_c[Γ_2, ∅]}
  od;
  for d := 2 to n − 1 do
    if d is even
    then for i := 1 to n − d do
           j := (d + i) − 1;
(* Rule 8 *)
           if      ω_i = a^+_⅋[Γ_1] then for each S_⊗[Γ_2, σ_2] ∈ A_{i+1,j} do
                                           σ := σ_2 ∪ (Γ_1 × Γ_2);
                                           if σ^+ is acyclic
                                             then A_{i,j} := A_{i,j} ∪ {U[a, +, σ]}
                                         od
(* Rule 9 *)
           else if ω_i = a^-_⅋[Γ_1] then for each S_⊗[Γ_2, σ_2] ∈ A_{i+1,j} do
                                           σ := σ_2 ∪ (Γ_1 × Γ_2);
                                           if σ^+ is acyclic
                                             then A_{i,j} := A_{i,j} ∪ {U[a, −, σ]}
                                         od
(* Rule 10 *)
           else if ω_i = a^+_⊗[Γ_1] then for each S_⅋[Γ_2, σ_2] ∈ A_{i+1,j} do
                                           σ := σ_2 ∪ (Γ_2 × Γ_1);
                                           if σ^+ is acyclic
                                             then A_{i,j} := A_{i,j} ∪ {U[a, +, σ]}
                                         od
(* Rule 11 *)
           else if ω_i = a^-_⊗[Γ_1] then for each S_⅋[Γ_2, σ_2] ∈ A_{i+1,j} do
                                           σ := σ_2 ∪ (Γ_2 × Γ_1);
                                           if σ^+ is acyclic
                                             then A_{i,j} := A_{i,j} ∪ {U[a, −, σ]}
                                         od
         od
    else (* d is odd *)
         for i := 1 to n − d do
           j := (d + i) − 1;
(* Rules 12 and 13 *)
           for each U[a, +, σ] ∈ A_{i,j−1} do
             if ω_{j+1} = a^-_c[Γ] then A_{i,j} := A_{i,j} ∪ {T_c[Γ, σ], S_c[Γ, σ]}
           od;
(* Rules 14 and 15 *)
           for each U[a, −, σ] ∈ A_{i,j−1} do
             if ω_{j+1} = a^+_c[Γ] then A_{i,j} := A_{i,j} ∪ {T_c[Γ, σ], S_c[Γ, σ]}
```

```
          od;
          k := i;
          while k < j do
(∗ Rule 2 ∗)
            for each T⅋[Γ1, σ1] ∈ A_{i,k} do
              for each S⊗[Γ2, σ2] ∈ A_{k+2,j} do
                σ := σ1 ∪ σ2 ∪ (Γ1 × Γ2);
                if σ+ is acyclic then A_{i,j} := A_{i,j} ∪ {S⅋[Γ1, σ]}
              od
            od;
(∗ Rules 1 and 3 ∗)
            for each T⊗[Γ1, σ1] ∈ A_{i,k} do
              for each S⅋[Γ2, σ2] ∈ A_{k+2,j} do
                σ := σ1 ∪ σ2 ∪ (Γ2 × Γ1);
                if σ+ is acyclic then A_{i,j} := A_{i,j} ∪ {S⅋[Γ2, σ]}
              od
              for each S⊗[Γ2, σ2] ∈ A_{k+2,j} do
                σ := σ1 ∪ σ2;
                if σ+ is acyclic then A_{i,j} := A_{i,j} ∪ {S⊗[Γ1 ∪ Γ2, σ]}
              od
            od;
            k := k + 2
          od
        od
  od
```

6 Conclusions and future work

Our proof-net construction algorithm does not take any advantage of the intuitionistic nature of the Lambek calculus. Consequently it is easily adaptable to classical calculus such as Yetter's [11] and Abrusci's [1]

Our work indirectly addresses the problem of the complexity of the Lambek calculus decision problem. Whether this problem is NP-hard or not is still open. The theoretical complexity of our algorithm is exponential because of the number of possible non-terminal symbols. It is possible to reduce this number because it is not necessary to record the complete dominance relation but only the constraints that are "still active". However, this optimisation, which we will describe in an extended version of this paper, does not give rise to a polynomial algorithm. Nonetheless, it allows to define interesting fragments of the Lambek calculus for which the decision problem is polynomial.

Acknowledgements I wish to thank Glyn Morrill, François Lamarche, and Yves Lafont for helpful discussions and comments.

References

1. M. Abrusci. Phase semantics and sequent calculus for pure non-commutative classical linear logic. *Journal of Symbolic Logic*, 56(4):1403–1451, 1991.
2. C. Berge. *Graphs.* North-Holland, second revised edition edition, 1985.
3. A Fleury. *La règle d'échange : logique lineaire multiplicative tressée.* Thès de Doctorat, spécialité Mathématiques, Université Paris 7, 1996.
4. J.-Y. Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.
5. J.-Y. Girard. Quantifiers in linear logic II. Technical Report 19, Equipe de Logique Mathématique, Université de Paris VII, 1991.
6. F. Lamarche and C. Retoré. Proof nets for the Lambek calculus. In M. Abrusci and C. Casadio, editors, *Proofs and Linguistic Categories, Proceedings 1996 Roma Workshop.* Cooperativa Libraria Universitaria Editrice Bologna, 1996.
7. J. Lambek. The mathematics of sentence structure. *Amer. Math. Monthly*, 65:154–170, 1958.
8. M. Moortgat. Categorial type logic. In J. van Benthem and A. ter Meulen, editors, *Handbook of Logic and Language*, chapter 2. Elsevier, 1997.
9. G. Morrill. Memoisation of categorial proof nets: parallelism in categorial processing. In V. M. Abrusci and C. Casadio, editors, *Proofs and Linguistic Categories, Proceedings 1996 Roma Workshop.* Cooperativa Libraria Universitaria Editrice Bologna, 1996.
10. D. Roorda. *Resource Logics: proof-theoretical investigations.* PhD thesis, University of Amsterdam, 1991.
11. D. N. Yetter. Quantales and (non-commutative) linear logic. *Journal of Symbolic Logic*, 55:41–64, 1990.

A Chomsky normal form of System $\boldsymbol{R_3}$

$$T_{\otimes}[\Gamma_1,\sigma_1]\,S_{\wp}[\Gamma_2,\sigma_2] \;\to\; S_{\wp}[\Gamma_2,\sigma_1\cup\sigma_2\cup(\Gamma_2\times\Gamma_1)] \tag{1}$$
$$T_{\wp}[\Gamma_1,\sigma_1]\,S_{\otimes}[\Gamma_2,\sigma_2] \;\to\; S_{\wp}[\Gamma_1,\sigma_1\cup\sigma_2\cup(\Gamma_1\times\Gamma_2)] \tag{2}$$
$$T_{\otimes}[\Gamma_1,\sigma_1]\,S_{\otimes}[\Gamma_2,\sigma_2] \;\to\; S_{\otimes}[\Gamma_1\cup\Gamma_2,\sigma_1\cup\sigma_2] \tag{3}$$
$$a^{+}_{\wp}[\Gamma_1]\,a^{-}_{\mathbf{c}}[\Gamma_2] \;\to\; T_{\mathbf{c}}[\Gamma_2,\emptyset] \tag{4}$$
$$a^{+}_{\wp}[\Gamma_1]\,a^{-}_{\mathbf{c}}[\Gamma_2] \;\to\; S_{\mathbf{c}}[\Gamma_2,\emptyset] \tag{5}$$
$$a^{-}_{\wp}[\Gamma_1]\,a^{+}_{\mathbf{c}}[\Gamma_2] \;\to\; T_{\mathbf{c}}[\Gamma_2,\emptyset] \tag{6}$$
$$a^{-}_{\wp}[\Gamma_1]\,a^{+}_{\mathbf{c}}[\Gamma_2] \;\to\; S_{\mathbf{c}}[\Gamma_2,\emptyset] \tag{7}$$
$$a^{+}_{\wp}[\Gamma_1]\,S_{\otimes}[\Gamma_2,\sigma_2] \;\to\; U[a,+,\sigma_2\cup(\Gamma_1\times\Gamma_2)] \tag{8}$$
$$a^{-}_{\wp}[\Gamma_1]\,S_{\otimes}[\Gamma_2,\sigma_2] \;\to\; U[a,-,\sigma_2\cup(\Gamma_1\times\Gamma_2)] \tag{9}$$
$$a^{+}_{\otimes}[\Gamma_1]\,S_{\wp}[\Gamma_2,\sigma_2] \;\to\; U[a,+,\sigma_2\cup(\Gamma_2\times\Gamma_1)] \tag{10}$$
$$a^{-}_{\otimes}[\Gamma_1]\,S_{\wp}[\Gamma_2,\sigma_2] \;\to\; U[a,-,\sigma_2\cup(\Gamma_2\times\Gamma_1)] \tag{11}$$
$$U[a,+,\sigma]\,a^{-}_{\mathbf{c}}[\Gamma] \;\to\; T_{\mathbf{c}}[\Gamma,\sigma] \tag{12}$$
$$U[a,+,\sigma]\,a^{-}_{\mathbf{c}}[\Gamma] \;\to\; S_{\mathbf{c}}[\Gamma,\sigma] \tag{13}$$
$$U[a,-,\sigma]\,a^{+}_{\mathbf{c}}[\Gamma] \;\to\; T_{\mathbf{c}}[\Gamma,\sigma] \tag{14}$$
$$U[a,-,\sigma]\,a^{+}_{\mathbf{c}}[\Gamma] \;\to\; S_{\mathbf{c}}[\Gamma,\sigma] \tag{15}$$