Lambek calculus is NP-complete

Mati Pentus*

May 12, 2003

Abstract

We prove that for both the Lambek calculus L and the Lambek calculus allowing empty premises L* the derivability problem is NP-complete. It follows that also for the multiplicative fragments of cyclic linear logic and noncommutative linear logic the derivability problem is NP-complete.

Introduction

The Lambek syntactic calculus L (introduced in [12]) is one of the logical calculi used in the paradigm of categorial grammar for deriving reduction laws of syntactic types (also called "categories") in natural and formal languages. In categorial grammars based on the Lambek calculus (or its variants) an expression is assigned to category $B \,/\, A$ (resp. $A \setminus B$) if and only if the expression produces an expression of category B whenever it is followed (resp. preceded) by an expression of category A. An expression is assigned to category $A \cdot B$ if and only if the expression can be obtained by concatenation of an expression of category A and an expression of category B. The reduction laws derivable in this calculus are of the form $A \to B$ (meaning "every expression of category A is also assigned to category B"). A survey of proof-theoretical properties of Lambek calculus can be found in [4].

There is a natural modification of the original Lambek calculus, which we call *the Lambek calculus allowing empty premises* and denote L* (see [22, p. 44]). Intuitively, the modified calculus allows the empty expression to be assigned to some categories. This calculus is in fact a fragment of noncommutative linear logic (introduced by V. M. Abrusci in [3]). Essentially the same logic has been called BL2 by J. Lambek [13] (it has also been studied by several other authors). Also the cyclic linear logic (defined by D. N. Yetter in [23]) is conservative over L*. In the propositional multiplicative fragments of all these logics cut-free proofs are of polynomial size. Thus the derivability problem for these fragments is in NP.

It is known that the derivability problem for the multiplicative commutative linear logic is NP-complete (see [10, 11, 14]). The same question for L, L*, and multiplicative noncommutative linear logics was an open problem (see e. g. [5, 15, 16, 17, 18]).

We show that the classical satisfiability problem SAT is polynomial time reducible to the L-derivability problem and thus L is NP-complete. This yields NP-completeness of the

*This research was partially supported by the Russian Foundation for Basic Research and by the program "Grants for Leading Scientific Schools".

following parsing problem: given a string and a Lambek categorial grammar, to decide whether the string is accepted by the grammar (even in the case where each terminal symbol is assigned to only one category).

The same reduction from *SAT* works also for the calculus L^* and consequently for the multiplicative fragment of noncommutative linear logic (and for the multiplicative fragment of cyclic linear logic).

This paper is organized as follows. The first section contains definitions of the calculi L and L^*. In Section 2 we give the main construction that reduces *SAT* to the L-derivability problem (and also to the L^*-derivability problem). Correctness of this construction is proved in Section 3, but the proof of one lemma is deferred until Section 6, where we use the technique of proof nets provided by Sections 4 and 5 (these proof nets are slightly different from those introduced by D. Roorda [21] and those studied by Ph. de Groote [7, 8]).

1 Lambek calculus

First we define *the Lambek calculus allowing empty premises* (denoted by L^*).

Assume that an enumerable set of *variables* Var is given. The *types* of L^* are built of variables (also called *primitive types* in the context of the Lambek calculus) and three binary connectives $\cdot$, $/$, and $\backslash$. The set of all types is denoted by Tp. The letters p, q, … range over the set Var, capital letters A, B, … range over types, and capital Greek letters range over finite (possibly empty) sequences of types. For notational convenience, we assume that $\cdot$ associates to the left.

The *sequents* of L^* are of the form $\Gamma \to A$ (Γ can be the empty sequence). The calculus L^* has the following axioms and rules of inference:

$$A \to A, \qquad \frac{\Phi \to B \quad \Gamma B \Delta \to A}{\Gamma \Phi \Delta \to A}\ (\mathrm{CUT}),$$

$$\frac{\Pi A \to B}{\Pi \to B \,/\, A}\ (\to/), \qquad \frac{\Phi \to A \quad \Gamma B \Delta \to C}{\Gamma (B \,/\, A) \Phi \Delta \to C}\ (/\to),$$

$$\frac{A \Pi \to B}{\Pi \to A \setminus B}\ (\to\backslash), \qquad \frac{\Phi \to A \quad \Gamma B \Delta \to C}{\Gamma \Phi (A \setminus B) \Delta \to C}\ (\backslash\to),$$

$$\frac{\Gamma \to A \quad \Delta \to B}{\Gamma \Delta \to A \cdot B}\ (\to\cdot), \qquad \frac{\Gamma A B \Delta \to C}{\Gamma (A \cdot B) \Delta \to C}\ (\cdot\to).$$

As usual, we shall write $\mathrm{L}^* \vdash \Gamma \to A$ to indicate that the sequent $\Gamma \to A$ is derivable in L^*.

The calculus L has the same axioms and rules with the only exception that in the rules $(\to\backslash)$ and $(\to/)$ we require Π to be nonempty. The calculus L is the original syntactic calculus introduced in [12]. Evidently, if $\mathrm{L} \vdash \Gamma \to A$, then $\mathrm{L}^* \vdash \Gamma \to A$.

It is known that the cut-elimination theorem holds for both L and L^*.

The rules $(\to\backslash)$, $(\to/)$ and $(\cdot\to)$ are reversible in both L and L^* (the converse rules are easy to derive with the help of the cut rule).

For every $p \in \mathrm{Var}$ we define a function $\#_p$ that maps types to integers as follows:

$$\#_p(q) \rightleftharpoons \begin{cases} 1 & \text{if } p = q, \\ 0 & \text{if } q \in \mathrm{Var} \text{ and } p \neq q, \end{cases}$$
$$\#_p(A \cdot B) \rightleftharpoons \#_p(A) + \#_p(B),$$
$$\#_p(A \setminus B) \rightleftharpoons \#_p(B) - \#_p(A),$$
$$\#_p(B \,/\, A) \rightleftharpoons \#_p(B) - \#_p(A).$$

This definition is extended to sequences of types as follows:

$$\#_p(A_1 \ldots A_n) \rightleftharpoons \#_p(A_1) + \ldots + \#_p(A_n).$$

Straightforward induction on derivations shows that if $\mathrm{L}^* \vdash \Gamma \to A$, then $\#_p(\Gamma) = \#_p(A)$ for every $p \in \mathrm{Var}$.

2 Reduction from *SAT*

Let $c_1 \wedge \ldots \wedge c_m$ be a Boolean formula in conjunctive normal form with clauses $c_1, \ldots, c_m$ and variables $x_1, \ldots, x_n$. The reduction maps the formula to a sequent, which is derivable in L^* (and in L) if and only if the formula $c_1 \wedge \ldots \wedge c_m$ is satisfiable.

For any Boolean variable x_i let $\neg_0 x_i$ stand for the literal $\neg x_i$ and $\neg_1 x_i$ stand for the literal x_i. Note that $\langle t_1, \ldots, t_n \rangle \in \{0,1\}^n$ is a satisfying assignment for the Boolean formula $c_1 \wedge \ldots \wedge c_m$ if and only if for every index $j \leq m$ there exists an index $i \leq n$ such that the literal $\neg_{t_i} x_i$ appears in the clause c_j (as usual, 1 stands for “true” and 0 stands for “false”).

Let p_i^j $(0 \leq i \leq n,\ 0 \leq j \leq m)$ be distinct primitive types from Var.

We define three families of types:

$$\begin{aligned}
G_i^0 &\rightleftharpoons p_0^0 \setminus p_i^0 \quad \text{if } 1 \leq i \leq n, \\
G_i^j &\rightleftharpoons (p_0^j \setminus G_i^{j-1}) \cdot p_i^j \quad \text{if } 1 \leq i \leq n \text{ and } 1 \leq j \leq m, \\
H_i^0 &\rightleftharpoons p_{i-1}^0 \setminus p_i^0 \quad \text{if } 1 \leq i \leq n, \\
H_i^j &\rightleftharpoons p_{i-1}^j \setminus (H_i^{j-1} \cdot p_i^j) \quad \text{if } 1 \leq i \leq n \text{ and } 1 \leq j \leq m, \\
E_i^0(t) &\rightleftharpoons p_{i-1}^0 \setminus p_i^0 \quad \text{if } 1 \leq i \leq n \text{ and } t \in \{0,1\}, \\
E_i^j(t) &\rightleftharpoons \begin{cases} (p_{i-1}^j \setminus E_i^{j-1}(t)) \cdot p_i^j & \text{if the literal } \neg_t x_i \text{ appears in the clause } c_j, \\ p_{i-1}^j \setminus (E_i^{j-1}(t) \cdot p_i^j) & \text{otherwise} \end{cases} \\
&\qquad \text{if } 1 \leq i \leq n,\ 1 \leq j \leq m,\ t \in \{0,1\}.
\end{aligned}$$

For convenience we introduce the following abbreviations:

$$\begin{aligned}
G &\rightleftharpoons G_n^m, \\
H_i &\rightleftharpoons H_i^m \quad \text{if } 1 \leq i \leq n, \\
F_i &\rightleftharpoons (E_i^m(1) \,/\, H_i) \cdot H_i \cdot (H_i \setminus E_i^m(0)) \quad \text{if } 1 \leq i \leq n.
\end{aligned}$$

The aim of the rest of this paper is to demonstrate that $\mathrm{L}^* \vdash F_1 \ldots F_n \to G$ if and only if $\mathrm{L} \vdash F_1 \ldots F_n \to G$ if and only if the formula $c_1 \wedge \ldots \wedge c_m$ is satisfiable.

Example. Consider the Boolean formula $(x_1 \vee x_2) \wedge (\neg x_2) \wedge (\neg x_1 \vee x_2)$. Here $c_1 = x_1 \vee x_2$, $c_2 = \neg x_2$, $c_3 = \neg x_1 \vee x_2$. By construction

$$\begin{aligned}
G &= ((p_0^3 \setminus ((p_0^2 \setminus ((p_0^1 \setminus (p_0^0 \setminus p_2^0)) \cdot p_2^1)) \cdot p_2^2)) \cdot p_2^3),\\
H_1 &= (p_0^3 \setminus ((p_0^2 \setminus ((p_0^1 \setminus ((p_0^0 \setminus p_1^0) \cdot p_1^1)) \cdot p_1^2)) \cdot p_1^3)),\\
H_2 &= (p_1^3 \setminus ((p_1^2 \setminus ((p_1^1 \setminus ((p_1^0 \setminus p_2^0) \cdot p_2^1)) \cdot p_2^2)) \cdot p_2^3)),\\
E_1^3(1) &= (p_0^3 \setminus ((p_0^2 \setminus (((p_0^1 \setminus (p_0^0 \setminus p_1^0)) \cdot p_1^1) \cdot p_1^2)) \cdot p_1^3)),\\
E_1^3(0) &= ((p_0^3 \setminus (p_0^2 \setminus ((p_0^1 \setminus ((p_0^0 \setminus p_1^0) \cdot p_1^1)) \cdot p_1^2))) \cdot p_1^3),\\
E_2^3(1) &= ((p_1^3 \setminus (p_1^2 \setminus (((p_1^1 \setminus (p_1^0 \setminus p_2^0)) \cdot p_2^1) \cdot p_2^2))) \cdot p_2^3),\\
E_2^3(0) &= (p_1^3 \setminus (((p_1^2 \setminus (p_1^1 \setminus ((p_1^0 \setminus p_2^0) \cdot p_2^1))) \cdot p_2^2) \cdot p_2^3)).
\end{aligned}$$

The sequent

$$((E_1^3(1) \,/\, H_1) \cdot H_1 \cdot (H_1 \setminus E_1^3(0)))\,((E_2^3(1) \,/\, H_2) \cdot H_2 \cdot (H_2 \setminus E_2^3(0))) \to G$$

is not derivable in L^*.

3 Correctness of the reduction

First we prove the easy part: $\mathrm{L} \vdash F_1 \ldots F_n \to G$ whenever the formula $c_1 \wedge \ldots \wedge c_m$ is satisfiable.

Lemma 3.1. *If* $\mathrm{L} \vdash B_1 \to B_2$, *then* $\mathrm{L} \vdash (A \setminus B_1) \cdot C \to A \setminus (B_2 \cdot C)$.

Proof.

$$\cfrac{\cfrac{\cfrac{A \to A \qquad \cfrac{B_1 \to B_2 \qquad C \to C}{B_1\, C \to B_2 \cdot C}\,(\to\cdot)}{A\,(A \setminus B_1)\, C \to B_2 \cdot C}\,(\setminus\to)}{(A \setminus B_1)\, C \to A \setminus (B_2 \cdot C)}\,(\to\setminus)}{(A \setminus B_1) \cdot C \to A \setminus (B_2 \cdot C)}\,(\cdot\to)$$

□

Lemma 3.2. *If* $1 \le i \le n$, $0 \le j \le m$, *and* $t \in \{0,1\}$, *then* $\mathrm{L} \vdash E_i^j(t) \to H_i^j$.

Proof. Induction on j. The induction step follows from Lemma 3.1 and from the observation that $\mathrm{L} \vdash A \setminus (B_1 \cdot C) \to A \setminus (B_2 \cdot C)$ whenever $\mathrm{L} \vdash B_1 \to B_2$. □

Lemma 3.3. *If* $1 \le i \le n$ *and* $t \in \{0,1\}$, *then* $\mathrm{L} \vdash F_i \to E_i^m(t)$.

Proof. In view of Lemma 3.2, $\mathrm{L} \vdash E_i^m(1) \to H_i$. From this we derive $\mathrm{L} \vdash F_i \to E_i^m(0)$ as follows.

$$\cfrac{\cfrac{\cfrac{H_i \to H_i \qquad \cfrac{E_i^m(1) \to H_i \qquad E_i^m(0) \to E_i^m(0)}{E_i^m(1)\,(H_i \setminus E_i^m(0)) \to E_i^m(0)}\,(\setminus\to)}{(E_i^m(1) \,/\, H_i)\, H_i\, (H_i \setminus E_i^m(0)) \to E_i^m(0)}\,(/\to)}{((E_i^m(1) \,/\, H_i) \cdot H_i)\,(H_i \setminus E_i^m(0)) \to E_i^m(0)}\,(\cdot\to)}{((E_i^m(1) \,/\, H_i) \cdot H_i \cdot (H_i \setminus E_i^m(0))) \to E_i^m(0)}\,(\cdot\to)$$

Similarly $\mathrm{L} \vdash F_i \to E_i^m(1)$ follows from $\mathrm{L} \vdash E_i^m(0) \to H_i$. □

Lemma 3.4. $\mathrm{L} \vdash (p_0^0 \setminus p_1^0) \ldots (p_{n-1}^0 \setminus p_n^0) \to p_0^0 \setminus p_n^0$.

Proof. By induction on i one can prove that $\mathrm{L} \vdash (p_0^0 \setminus p_1^0) \ldots (p_{i-1}^0 \setminus p_i^0) \to p_0^0 \setminus p_i^0$ whenever $1 \leq i \leq n$. The induction step involves the cut rule. $\square$

Lemma 3.5. *If* $1 \leq i \leq n$, $1 \leq j \leq m$, $t \in \{0, 1\}$, *and* $\mathrm{L} \vdash \Gamma\, E_i^{j-1}(t)\, p_i^j\, \Delta \to C$, *then* $\mathrm{L} \vdash \Gamma\, p_{i-1}^j\, E_i^j(t)\, \Delta \to C$.

Proof. According to Lemma 3.1, $\mathrm{L} \vdash E_i^j(t) \to p_{i-1}^j \setminus (E_i^{j-1}(t) \cdot p_i^j)$ regardless of whether the literal $\neg_t x_i$ appears in the clause c_j. Thus we can use the following derivation.

$$\dfrac{E_i^j(t) \to p_{i-1}^j \setminus (E_i^{j-1}(t) \cdot p_i^j) \qquad \dfrac{p_{i-1}^j \to p_{i-1}^j \qquad \dfrac{\Gamma\, E_i^{j-1}(t)\, p_i^j\, \Delta \to C}{\Gamma\, (E_i^{j-1}(t) \cdot p_i^j)\, \Delta \to C}\,(\cdot\to)}{\Gamma\, p_{i-1}^j\, (p_{i-1}^j \setminus (E_i^{j-1}(t) \cdot p_i^j))\, \Delta \to C}\,(\setminus\to)}{\Gamma\, p_{i-1}^j\, E_i^j(t)\, \Delta \to C}\,(\mathrm{CUT})$$

$\square$

Lemma 3.6. *Let* $0 \leq j \leq m$. *Suppose* $\langle t_1, \ldots, t_n \rangle$ *is a satisfying assignment for the Boolean formula* $c_1 \wedge \ldots \wedge c_j$. *Then* $\mathrm{L} \vdash E_1^j(t_1) \ldots E_n^j(t_n) \to G_n^j$.

Proof. Induction on j. The induction base is provided by Lemma 3.4. To prove the induction step, assume that $j \geq 1$ and $\mathrm{L} \vdash E_1^{j-1}(t_1) \ldots E_n^{j-1}(t_n) \to G_n^{j-1}$. Since $\langle t_1, \ldots, t_n \rangle$ is a satisfying assignment for the clause c_j, there exists an index k such that the literal $\neg_{t_k} x_k$ appears in the clause c_j. Thus $E_k^j(t) = (p_{k-1}^j \setminus E_k^{j-1}(t)) \cdot p_k^j$. The induction hypothesis yields

$$\mathrm{L} \vdash E_1^{j-1}(t_1) \ldots E_{k-1}^{j-1}(t_{k-1})\, p_{k-1}^j\, (p_{k-1}^j \setminus E_k^{j-1}(t_k))\, E_{k+1}^{j-1}(t_{k+1}) \ldots E_n^{j-1}(t_n) \to G_n^{j-1}.$$

Applying Lemma 3.5 $k-1$ times we obtain

$$\mathrm{L} \vdash p_0^j\, E_1^j(t_1) \ldots E_{k-1}^j(t_{k-1})\, (p_{k-1}^j \setminus E_k^{j-1}(t_k))\, E_{k+1}^{j-1}(t_{k+1}) \ldots E_n^{j-1}(t_n) \to G_n^{j-1}.$$

Application of the rules $(\to\setminus)$ and $(\to\cdot)$ yields

$$\mathrm{L} \vdash E_1^j(t_1) \ldots E_{k-1}^j(t_{k-1})\, (p_{k-1}^j \setminus E_k^{j-1}(t_k))\, E_{k+1}^{j-1}(t_{k+1}) \ldots E_n^{j-1}(t_n)\, p_n^j \to G_n^j.$$

Applying Lemma 3.5 $n-k$ times we obtain

$$\mathrm{L} \vdash E_1^j(t_1) \ldots E_{k-1}^j(t_{k-1})\, (p_{k-1}^j \setminus E_k^{j-1}(t_k))\, p_k^j\, E_{k+1}^j(t_{k+1}) \ldots E_n^j(t_n) \to G_n^j.$$

Application of the rule $(\cdot\to)$ yields $\mathrm{L} \vdash E_1^j(t_1) \ldots E_n^j(t_n) \to G_n^j$. $\square$

Lemma 3.7. *If the formula* $c_1 \wedge \ldots \wedge c_m$ *is satisfiable, then* $\mathrm{L} \vdash F_1 \ldots F_n \to G$.

Proof. Suppose $\langle t_1, \ldots, t_n \rangle$ is a satisfying assignment for the formula $c_1 \wedge \ldots \wedge c_m$. According to Lemma 3.6 $\mathrm{L} \vdash E_1^m(t_1) \ldots E_n^m(t_n) \to G$. It remains to apply Lemma 3.3 and the cut rule n times. $\square$

Now our aim is to prove that if $\mathrm{L}^* \vdash F_1 \ldots F_n \to G$, then the formula $c_1 \wedge \ldots \wedge c_m$ is satisfiable.

Lemma 3.8. *If $1 \le i \le n$, $1 \le j \le m$, $t \in \{0,1\}$, and $\mathrm{L}^* \vdash \Gamma\, E_i^j(t)\, \Delta \to C$, then $\mathrm{L}^* \vdash \Gamma\, (p_{i-1}^j \setminus E_i^{j-1}(t))\, p_i^j\, \Delta \to C$.*

Proof. Following the derivation in the proof of Lemma 3.1 (dropping the last step) one can easily verify that $\mathrm{L} \vdash (p_{i-1}^j \setminus E_i^{j-1}(t))\, p_i^j \to E_i^j(t)$ regardless of whether the literal $\neg_t x_i$ appears in the clause c_j. It remains to apply the cut rule. □

Lemma 3.9. *If $p \in \mathrm{Var}$, $\mathrm{L}^* \vdash \Upsilon\, p \to p$, and p does not occur in Υ, then Υ is empty.*

Proof. We take a cut-free derivation of $\Upsilon\, p \to p$ and proceed by induction on derivation length. The induction step involves three simple cases, which correspond to the rules $(/\to)$, $(\setminus\to)$, and $(\cdot\to)$.

We consider the case $(/\to)$. Let

$$\frac{\Phi \to A \quad \Gamma B \Delta \to p}{\Gamma(B \,/\, A)\Phi\Delta \to p}\ (/\to),$$

where $\Gamma(B \,/\, A)\Phi\Delta = \Upsilon p$. From $\#_p(A) = 0$ we conclude that $\#_p(\Phi) = 0$. Therefore $\Delta = \Delta' p$ for some Δ', whence we can apply the induction hypothesis for the sequent $\Gamma B \Delta \to p$ and obtain contradiction.

The other two cases are straightforward. □

Lemma 3.10. *If $p \in \mathrm{Var}$, $\mathrm{L}^* \vdash \Upsilon\, p \to D \cdot p$, and p does not occur in the sequent $\Upsilon \to D$, then $\mathrm{L}^* \vdash \Upsilon \to D$.*

Proof. Induction on the length of a cut-free derivation of $\Upsilon\, p \to D \cdot p$. In the induction step we consider the last rule of the derivation.

Case 1: $(\to\cdot)$.

Then $\Upsilon = \Gamma\Delta'$ and

$$\frac{\Gamma \to D \quad \Delta' p \to p}{\Gamma\Delta' p \to D \cdot p}\ (\to\cdot).$$

In view of (i), Δ' is empty. Thus $\Upsilon = \Gamma$ and $\mathrm{L}^* \vdash \Upsilon \to D$.

Case 2: $(/\to)$.

Let

$$\frac{\Phi \to A \quad \Gamma B \Delta \to D \cdot p}{\Gamma(B \,/\, A)\Phi\Delta \to D \cdot p}\ (/\to),$$

where $\Upsilon p = \Gamma(B \,/\, A)\Phi\Delta$. From $\#_p(A) = 0$ we conclude that $\#_p(\Phi) = 0$. Therefore $\Delta = \Delta' p$ and $\Upsilon = \Gamma(B \,/\, A)\Phi\Delta'$. Applying the induction hypothesis for the sequent $\Gamma B \Delta' p \to D \cdot p$ we obtain $\Gamma B \Delta' \to D$. It remains to derive in L^*:

$$\frac{\Phi \to A \quad \Gamma B \Delta' \to D}{\Gamma(B \,/\, A)\Phi\Delta' \to D}\ (/\to),$$

The cases $(\cdot\to)$ and $(\setminus\to)$ are similar. □

Lemma 3.11. *If $p \in \mathrm{Var}$, $\mathrm{L}^* \vdash \Upsilon\, p\, (p \setminus D)\, \Psi \to E$, and p does not occur in the sequent $\Upsilon\, D\, \Psi \to E$, then $\mathrm{L}^* \vdash \Upsilon\, D\, \Psi \to E$.*

Proof. Induction on the length of a cut-free derivation of $\Upsilon\, p\,(p \setminus D)\, \Psi \to E$. Again we consider the last rule of the derivation. We shall investigate in detail only the rule $(\to\bullet)$ (other rules can be treated similarly).

Let

$$\frac{\Gamma \to A \quad \Delta \to B}{\Gamma\Delta \to A \bullet B}\ (\to\bullet),$$

where $\Gamma\Delta = \Upsilon\, p\,(p \setminus D)\, \Psi$. We consider three cases.
CASE 1: $\Gamma = \Upsilon\, p$ and $\Delta = (p \setminus D)\, \Psi$.
This is impossible, since $\#_p(\Upsilon\, p) = 1$ and $\#_p(A) = 0$.
CASE 2: $\Upsilon = \Upsilon'\, \Upsilon''$, $\Gamma = \Upsilon'$, and $\Delta = \Upsilon''\, p\,(p \setminus D)\, \Psi$.
The induction hypothesis for $\Upsilon''\, p\,(p \setminus D)\, \Psi \to B$ gives $\Upsilon''\, D\, \Psi \to B$. It remains to derive in L^*:

$$\frac{\Upsilon' \to A \quad \Upsilon''\, D\, \Psi \to B}{\Upsilon'\, \Upsilon''\, D\, \Psi \to A \bullet B}\ (\to\bullet).$$

CASE 3: $\Psi = \Psi'\, \Psi''$, $\Gamma = \Upsilon\, p\,(p \setminus D)\, \Psi'$, and $\Delta = \Psi''$.
The induction hypothesis for $\Upsilon\, p\,(p \setminus D)\, \Psi' \to A$ gives $\Upsilon\, D\, \Psi' \to A$. It remains to derive in L^*:

$$\frac{\Upsilon\, D\, \Psi' \to A \quad \Psi'' \to B}{\Upsilon\, D\, \Psi'\, \Psi'' \to A \bullet B}\ (\to\bullet).$$

□

Lemma 3.12. *Let $1 \le j \le m$ and $\langle t_1, \dots, t_n \rangle \in \{0,1\}^n$. If $\mathrm{L}^* \vdash E_1^j(t_1) \dots E_n^j(t_n) \to G_n^j$, then $\mathrm{L}^* \vdash E_1^{j-1}(t_1) \dots E_n^{j-1}(t_n) \to G_n^{j-1}$.*

Proof. First we apply Lemma 3.8 n times. Next we apply Lemma 3.10 and the converse of the rule $(\to\setminus)$. Finally we apply Lemma 3.11 n times. □

Lemma 3.13. *Let $1 \le j \le m$ and $\langle t_1, \dots, t_n \rangle \in \{0,1\}^n$. If $\mathrm{L}^* \vdash E_1^j(t_1) \dots E_n^j(t_n) \to G_n^j$, then $\langle t_1, \dots, t_n \rangle$ is a satisfying assignment for the clause c_j.*

Proof. Assume for the contrary that $\langle t_1, \dots, t_n \rangle$ is not a satisfying assignment for the clause c_j. This means that none of the literals $\neg_{t_i} x_i$ appears in the clause c_j. Thus the sequent $E_1^j(t_1) \dots E_n^j(t_n) \to G_n^j$ is of the form

$$(p_0^j \setminus (A_1 \bullet p_1^j)) \dots (p_{n-1}^j \setminus (A_n \bullet p_n^j)) \to (p_0^j \setminus B) \bullet p_n^j$$

for some types A_1, …, A_n, and B that contain none of the variables p_0^j, …, p_n^j. It is easy to see that the last rule in a derivation of such a sequent can only be the cut rule (a variable p_i^j can not occur in a derivable sequent exactly once). Thus the sequent has no cut-free derivation. Hence it is not derivable in L^*. □

Lemma 3.14. *If $\langle t_1, \dots, t_n \rangle \in \{0,1\}^n$ and $\mathrm{L}^* \vdash E_1^m(t_1) \dots E_n^m(t_n) \to G$, then $\langle t_1, \dots, t_n \rangle$ is a satisfying assignment for the Boolean formula $c_1 \wedge \dots \wedge c_m$.*

Proof. Immediate from Lemma 3.12 and Lemma 3.13.

In fact this lemma can also be proved by means of an argument concerning proof nets, which are defined in Section 5 (then Lemmas 3.8–3.13 are not needed). □

The following key lemma provides a “switch”, which guarantees that the value of a Boolean variable x_k (which is modelled in L^* by the type F_k) can only be changed in all clauses simultaneously.

Lemma 3.15. *Let $k \geq 1$, $B \in \mathrm{Tp}$, and $\mathrm{L}^* \vdash F_1 \dots F_{k-1} F_k \to B$. Let the variables p_0^m and p_k^m occur only once in B and none of the variables p_1^m, ..., p_{k-1}^m occur in B. Then $\mathrm{L}^* \vdash F_1 \dots F_{k-1} E_k^m(t) \to B$ for some $t \in \{0,1\}$.*

Lemma 3.15 will be proved in Section 6.

Lemma 3.16. *If $\mathrm{L}^* \vdash F_1 \dots F_n \to G$, then $\mathrm{L}^* \vdash E_1^m(t_1) \dots E_n^m(t_n) \to G$ for some $\langle t_1, \dots, t_n \rangle \in \{0,1\}^n$.*

Proof. By induction on $n - k$ we show that for every $k \leq n$ there is an assignment $\langle t_{k+1}, \dots, t_n \rangle \in \{0,1\}^{n-k}$ such that $\mathrm{L}^* \vdash F_1 \dots F_k E_{k+1}^m(t_{k+1}) \dots E_n^m(t_n) \to G$.

To prove the induction step, assume that

$$\mathrm{L}^* \vdash F_1 \dots F_k E_{k+1}^m(t_{k+1}) \dots E_n^m(t_n) \to G.$$

We apply Lemma 3.15 to the derivable sequent

$$F_1 \dots F_k \to (\dots (G \,/\, E_n^m(t_n)) \dots /\, E_{k+1}^m(t_{k+1})).$$

Now the converse of the rule $(\to /)$ can be applied $n - k$ times. We obtain

$$\mathrm{L}^* \vdash F_1 \dots F_{k-1} E_k^m(t_k) \dots E_n^m(t_n) \to G$$

for some $t_k \in \{0,1\}$. □

Lemma 3.17. *If $\mathrm{L}^* \vdash F_1 \dots F_n \to G$, then the formula $c_1 \wedge \dots \wedge c_m$ is satisfiable.*

Proof. Immediate from Lemma 3.16 and Lemma 3.14. □

Theorem 1. *The* L*-derivability problem is* NP*-complete.*

Proof. Due to cut-elimination the L-derivability problem is in NP (the size of a cut-free derivation in L can not exceed the square of the length of the final sequent).

According to Lemma 3.7 and Lemma 3.17 the construction in Section 2 provides a mapping reduction from the classical satisfiability problem *SAT* to the L-derivability problem. The problem *SAT* is NP-hard (see [6]). Thus the L-derivability problem is NP-hard. □

Theorem 2. *The* L^**-derivability problem is* NP*-complete.*

Proof. Like the previous theorem, also this one follows immediately from Lemma 3.7 and Lemma 3.17. □

4 Noncommutative linear logic

Several equivalent to each other sequent calculi for the pure noncommutative classical linear propositional logic were introduced in [3]. Here we consider only the minimal multiplicative fragment $\mathrm{SPNCL}'_{\parr\otimes}$ of that logic (without the constants $\perp$ and $\mathbf{1}$). For shortness we shall denote that fragment by NCL in this paper. The calculus NCL may also be considered as a fragment of Lambek's bilinear logic BL2 from [13] (but we use $\parr$ instead of $\oplus$).

In the definition of formulas of NCL we shall employ the same enumerable set Var that was used in the definition of Lambek calculus types. First, we introduce the set At of formal symbols called *atoms*

$$\mathrm{At} \rightleftharpoons \{p^{\perp n} \mid p \in \mathrm{Var},\ n \in \mathbf{Z}\}$$

(as usual, $\mathbf{Z}$ stands for the set of all integers). Intuitively, if $n \geq 0$, then $p^{\perp n}$ means "p with n right negations" and $p^{\perp -n}$ means "p with n left negations".

The set of *normal formulas* (or just *formulas* for shortness) is defined to be the smallest set NF satisfying the following conditions:

- $\mathrm{At} \subset \mathrm{NF}$,
- if $A \in \mathrm{NF}$ and $B \in \mathrm{NF}$, then $(A \otimes B) \in \mathrm{NF}$ and $(A \mathbin{⅋} B) \in \mathrm{NF}$.

Here $\otimes$ is the multiplicative conjunction, called *tensor*, and $⅋$ is the multiplicative disjunction, called *par*. For notational convenience, it is assumed that $\otimes$ and $⅋$ associate to the left. The set of all finite sequences of formulas is denoted by NF^*.

The *right negation* ($A^\perp$) and the *left negation* (${}^\perp A$) of a formula A are defined as follows:

$$\begin{array}{rlrl}
(p^{\perp n})^\perp & \rightleftharpoons p^{\perp(n+1)}, & {}^\perp(p^{\perp n}) & \rightleftharpoons p^{\perp(n-1)}, \\
(A \otimes B)^\perp & \rightleftharpoons (B^\perp) \mathbin{⅋} (A^\perp), & {}^\perp(A \otimes B) & \rightleftharpoons ({}^\perp B) \mathbin{⅋} ({}^\perp A), \\
(A \mathbin{⅋} B)^\perp & \rightleftharpoons (B^\perp) \otimes (A^\perp), & {}^\perp(A \mathbin{⅋} B) & \rightleftharpoons ({}^\perp B) \otimes ({}^\perp A).
\end{array}$$

Example. If

$$A = ((p_0^2)^{\perp 1} \mathbin{⅋} (((p_0^1)^{\perp 1} \mathbin{⅋} (((p_0^0)^{\perp 1} \mathbin{⅋} (p_1^0)^{\perp 0}) \otimes (p_1^1)^{\perp 0})) \otimes (p_1^2)^{\perp 0})),$$

then

$$A^\perp = (((p_1^2)^{\perp 1} \mathbin{⅋} (((p_1^1)^{\perp 1} \mathbin{⅋} ((p_1^0)^{\perp 1} \otimes (p_0^0)^{\perp 2})) \otimes (p_0^1)^{\perp 2})) \otimes (p_0^2)^{\perp 2}).$$

The *sequents* of NCL are of the form $\to\Gamma$, where $\Gamma \in \mathrm{NF}^*$.

The calculus NCL has the following axioms and rules of inference:

$$\to p^{\perp(n+1)}\, p^{\perp n},$$

$$\frac{\to\Gamma A B \Delta}{\to\Gamma(A \mathbin{⅋} B)\Delta}, \qquad \frac{\to\Gamma A \quad \to B\Delta}{\to\Gamma(A \otimes B)\Delta},$$

$$\frac{\to A\Gamma}{\to\Gamma({}^\perp({}^\perp A))}, \qquad \frac{\to\Gamma A}{\to((A^\perp)^\perp)\Gamma}.$$

Here capital letters A, B, ... stand for formulas, capital Greek letters denote finite (possibly empty) sequences of formulas, p ranges over Var, and n ranges over $\mathbf{Z}$. As usual, $\mathrm{NCL} \vdash \to\Gamma$ means that the sequent $\to\Gamma$ is derivable in NCL.

The set of all *subformulas* of a formula is defined as follows:

$$\begin{aligned}
\mathrm{SubNF}(p^{\perp n}) &\rightleftharpoons \{p^{\perp n}\}, \\
\mathrm{SubNF}(A \mathbin{⅋} B) &\rightleftharpoons \{A \mathbin{⅋} B\} \cup \mathrm{SubNF}(A) \cup \mathrm{SubNF}(B), \\
\mathrm{SubNF}(A \otimes B) &\rightleftharpoons \{A \otimes B\} \cup \mathrm{SubNF}(A) \cup \mathrm{SubNF}(B).
\end{aligned}$$

To embed L^* into NCL, we shall map each type $A \in \mathrm{Tp}$ to a formula $\widehat{A} \in \mathrm{NF}$:

$$\begin{aligned}
\widehat{p} &\rightleftharpoons p, \\
\widehat{A / B} &\rightleftharpoons \widehat{A} \parr ({}^{\perp}\widehat{B}), \\
\widehat{A \setminus B} &\rightleftharpoons (\widehat{A}^{\perp}) \parr \widehat{B}, \\
\widehat{A \cdot B} &\rightleftharpoons \widehat{A} \otimes \widehat{B}.
\end{aligned}$$

Example. Consider the type

$$G_5^2 = ((p_0^2 \setminus ((p_0^1 \setminus (p_0^0 \setminus p_5^0)) \cdot p_5^1)) \cdot p_5^2).$$

Then
$$\widehat{G_5^2} = (((p_0^2)^{\perp 1} \parr (((p_0^1)^{\perp 1} \parr ((p_0^0)^{\perp 1} \parr (p_5^0)^{\perp 0})) \otimes (p_5^1)^{\perp 0})) \otimes (p_5^2)^{\perp 0}).$$

The following lemma is proved in [20].

Lemma 4.1. *Let $A_1, \ldots, A_n, B \in \mathrm{Tp}$. The sequent $A_1 \ldots A_n \to B$ is derivable in* L^* *if and only if the sequent* $\to(\widehat{A_n}^{\perp}) \ldots (\widehat{A_1}^{\perp})\widehat{B}$ *is derivable in* NCL.

5 Proof nets

We shall repeat the definition of proof net from [20] (but without the multiplicative constants $\perp$ and $\mathbf{1}$).

Definition. For the purposes of proof nets it is convenient to measure the *length* of a formula using the following function:

$$\begin{aligned}
\|p^{\perp n}\| &\rightleftharpoons 2, \\
\|A \otimes B\| &\rightleftharpoons \|A\| + \|B\|, \\
\|A \parr B\| &\rightleftharpoons \|A\| + \|B\|.
\end{aligned}$$

The notion of length is extended to finite sequences of formulas in the natural way: $\|A_1 \ldots A_n\| \rightleftharpoons \|A_1\| + \ldots + \|A_n\|$, the length of the empty sequence is 0.

Evidently $\|A^{\perp}\| = \|A\|$ for every $A \in \mathrm{NF}$.

Definition. To formalize the notion of *occurrences* of subformulas, we introduce the set $\mathrm{Occ} \rightleftharpoons \mathrm{NF} \times \mathbf{Z}$. Let c be the map from NF to $\mathbf{Z}$ defined by

$$\begin{aligned}
c(p^{\perp n}) &\rightleftharpoons 1, \\
c(A \otimes B) &\rightleftharpoons \|A\|, \\
c(A \parr B) &\rightleftharpoons \|A\|.
\end{aligned}$$

The binary relation $\prec$ on the set Occ is defined as the least transitive binary relation satisfying $\langle A, k - \|A\| + c(A)\rangle \prec \langle (A \mathbin{\lambda} B), k\rangle$ and $\langle B, k + c(B)\rangle \prec \langle (A \mathbin{\lambda} B), k\rangle$ for every $\lambda \in \{\otimes, \parr\}$, $A \in \mathrm{NF}$, $B \in \mathrm{NF}$, and $k \in \mathbf{Z}$. The symbol $\preceq$ is introduced in the usual manner.

Given a formula A, one can associate occurrences of its subformulas with elements of Occ. Each subformula occurrence B corresponds to a pair $\langle B, k\rangle \in \mathrm{Occ}$ such that $\langle B, k\rangle \preceq \langle A, c(A)\rangle$ and k is the "$\|\cdot\|$-distance" of the main connective of B from the left end of A.

Example. Let $p \in \mathrm{Var}$, $q \in \mathrm{Var}$, and $A = (p^{\perp 2} \parr (p^{\perp 2} \otimes q^{\perp 1})) \otimes q^{\perp 1}$. Then $\|A\| = 8$ and $c(A) = 6$. There are seven elements $\alpha \in \mathrm{Occ}$ such that $\alpha \preceq \langle A, 6\rangle$. These elements are

$$\begin{aligned}
\alpha_1 &= \langle p^{\perp 2}, 1\rangle,\\
\alpha_2 &= \langle p^{\perp 2} \parr (p^{\perp 2} \otimes q^{\perp 1}), 2\rangle,\\
\alpha_3 &= \langle p^{\perp 2}, 3\rangle,\\
\alpha_4 &= \langle p^{\perp 2} \otimes q^{\perp 1}, 4\rangle,\\
\alpha_5 &= \langle q^{\perp 1}, 5\rangle,\\
\alpha_6 &= \langle (p^{\perp 2} \parr (p^{\perp 2} \otimes q^{\perp 1})) \otimes q^{\perp 1}, 6\rangle,\\
\alpha_7 &= \langle q^{\perp 1}, 7\rangle.
\end{aligned}$$

Definition. For any sequence of formulas $\Gamma = A_1 \dots A_n$ we construct a relational structure $\mathbf{\Omega}_\Gamma = \langle \Omega_\Gamma, \prec_\Gamma, <_\Gamma\rangle$ as follows. By definition, put

$$\begin{aligned}
\Omega_\Gamma \rightleftharpoons \{\langle B, k + \|A_1 \dots A_{i-1}\|\rangle \mid 1 \le i \le n \text{ and } \langle B, k\rangle \preceq \langle A_i, c(A_i)\rangle\}&\\
\cup \{\langle \diamond, \|A_1 \dots A_{i-1}\|\rangle \mid 1 \le i \le n\},&
\end{aligned}$$

where $\diamond$ is a new symbol that does not belong to NF. The set Ω_Γ can be considered as consisting of four disjoint parts

$$\begin{aligned}
\Omega_\Gamma^\diamond &\rightleftharpoons \{\langle C, k\rangle \in \Omega_\Gamma \mid C = \diamond\},\\
\Omega_\Gamma^{\mathrm{At}} &\rightleftharpoons \{\langle C, k\rangle \in \Omega_\Gamma \mid C \in \mathrm{At}\},\\
\Omega_\Gamma^{\otimes} &\rightleftharpoons \{\langle C, k\rangle \in \Omega_\Gamma \mid C = A \otimes B \text{ for some } A \text{ and } B\},\\
\Omega_\Gamma^{\parr} &\rightleftharpoons \{\langle C, k\rangle \in \Omega_\Gamma \mid C = A \parr B \text{ for some } A \text{ and } B\}.
\end{aligned}$$

We shall abbreviate $\Omega_\Gamma^{\parr} \cup \Omega_\Gamma^{\diamond}$ as $\Omega_\Gamma^{\parr\diamond}$. The relation $\prec_\Gamma$ is the irreflexive partial order on Ω_Γ such that $\alpha \prec_\Gamma \beta$ iff $\alpha \notin \Omega_\Gamma^\diamond$, $\beta \notin \Omega_\Gamma^\diamond$, and $\alpha \prec \beta$. The relation $<_\Gamma$ is the irreflexive linear order on Ω_Γ such that $\langle A, k\rangle <_\Gamma \langle B, l\rangle$ iff $k < l$. The symbols $\preceq_\Gamma$ and $\le_\Gamma$ are introduced in the usual manner.

Let us remark that the binary relation $\prec_\Gamma$ specifies a forest of ordered binary trees, where vertices are the elements of $\Omega_\Gamma - \Omega_\Gamma^\diamond$ and $<_\Gamma$ corresponds to the infix order.

Example. Let $A_1 = p^{\perp 1}$, $A_2 = p^{\perp 4}$, $A_3 = p^{\perp 3} \otimes p^{\perp 0}$, and $\Gamma = A_1 A_2 A_3$. Then $\Omega_\Gamma^{\diamond} = \{\alpha_0, \alpha_2, \alpha_4\}$, $\Omega_\Gamma^{\mathrm{At}} = \{\alpha_1, \alpha_3, \alpha_5, \alpha_7\}$, $\Omega_\Gamma^{\otimes} = \{\alpha_6\}$, and $\Omega_\Gamma^{\invamp} = \emptyset$, where

$$\begin{aligned}
\alpha_0 &= \langle \diamond, 0 \rangle, \\
\alpha_1 &= \langle p^{\perp 1}, 1 \rangle, \\
\alpha_2 &= \langle \diamond, 2 \rangle, \\
\alpha_3 &= \langle p^{\perp 4}, 3 \rangle, \\
\alpha_4 &= \langle \diamond, 4 \rangle, \\
\alpha_5 &= \langle p^{\perp 3}, 5 \rangle, \\
\alpha_6 &= \langle p^{\perp 3} \otimes p^{\perp 0}, 6 \rangle, \\
\alpha_7 &= \langle p^{\perp 0}, 7 \rangle.
\end{aligned}$$

Obviously, $\alpha_0 <_\Gamma \alpha_1 <_\Gamma \ldots <_\Gamma \alpha_7$, $\alpha_5 \prec_\Gamma \alpha_6$, and $\alpha_7 \prec_\Gamma \alpha_6$.

Definition. For every $\Theta \subseteq \Omega_\Gamma$ we put $\flat(\Theta) \rightleftharpoons |\Omega_\Gamma^{\invamp\diamond} \cap \Theta| - |\Omega_\Gamma^{\otimes} \cap \Theta|$.

Definition. For any $\alpha \in \Omega_\Gamma$ and $\beta \in \Omega_\Gamma$ we denote by $\mathrm{Bt}(\alpha, \beta)$ the set

$$\{\gamma \in \Omega_\Gamma \mid \alpha <_\Gamma \gamma <_\Gamma \beta \text{ or } \beta <_\Gamma \gamma <_\Gamma \alpha\}.$$

Example. Let $A_1 = p^{\perp 1} \otimes ((p^{\perp 2} \otimes (p^{\perp 3} \invamp p^{\perp 2})) \invamp p^{\perp 1})$, $A_2 = p^{\perp 0}$, and $\Gamma = A_1 A_2$. Consider the elements $\alpha = \langle (p^{\perp 2} \otimes (p^{\perp 3} \invamp p^{\perp 2})) \invamp p^{\perp 1}, 8 \rangle$ and $\beta = \langle p^{\perp 2} \otimes (p^{\perp 3} \invamp p^{\perp 2}), 4 \rangle$. Then $\mathrm{Bt}(\alpha, \beta) = \{\langle p^{\perp 3}, 5 \rangle, \langle p^{\perp 3} \invamp p^{\perp 2}, 6 \rangle, \langle p^{\perp 2}, 7 \rangle\}$.

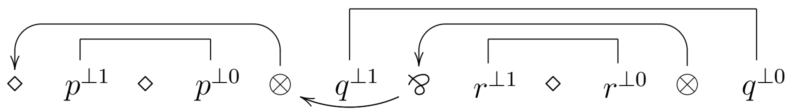
Definition. Let $\mathcal{C} \subseteq \Omega_\Gamma \times \Omega_\Gamma$. We say that the directed graph $\langle \Omega_\Gamma, \mathcal{C} \rangle$ is *$<_\Gamma$-planar* if for every edge $\langle \alpha, \beta \rangle \in \mathcal{C}$ and every edge $\langle \gamma, \delta \rangle \in \mathcal{C}$ the statements $\gamma \in \mathrm{Bt}(\alpha, \beta)$ and $\delta \in \mathrm{Bt}(\alpha, \beta)$ are either both true or both false, provided that $\{\alpha, \beta\} \cap \{\gamma, \delta\} = \emptyset$.

In intuitive language, a graph is $<_\Gamma$-planar if and only if its edges can be drawn without intersections on a semiplane while the vertices of the graph are ordered according to $<_\Gamma$ on the border of the semiplane.

Definition. Let $\Gamma \in \mathrm{NF}^*$. A *proof net for* Γ is a relational structure $\langle \mathbf{\Omega}_\Gamma, \mathcal{A}, \mathcal{E} \rangle$, where

- $\flat(\Omega_\Gamma) = 2$,
- $\mathcal{A}$ is the graph of a function from $\Omega_\Gamma^{\otimes}$ to $\Omega_\Gamma^{\invamp\diamond}$,
- $\mathcal{E}$ is the graph of a function from $\Omega_\Gamma^{\mathrm{At}}$ to $\Omega_\Gamma^{\mathrm{At}}$,
- if $\langle \alpha, \beta \rangle \in \mathcal{E}$, then $\langle \beta, \alpha \rangle \in \mathcal{E}$,
- if $\langle \alpha, \beta \rangle \in \mathcal{E}$ and $\alpha \leq_\Gamma \beta$, then there are $p \in \mathrm{Var}$ and $n, i, j \in \mathbf{Z}$ such that $\alpha = \langle p^{\perp(n+1)}, i \rangle$ and $\beta = \langle p^{\perp n}, j \rangle$,
- the graph $\langle \Omega_\Gamma, \mathcal{A} \cup \mathcal{E} \rangle$ is $<_\Gamma$-planar, and
- the graph $\langle \Omega_\Gamma, \prec_\Gamma \cup \mathcal{A} \rangle$ is acyclic (i. e., the transitive closure of the binary relation $\prec_\Gamma \cup \mathcal{A}$ is irreflexive).

Example. Let $A_1 = p^{\perp 1}$, $A_2 = p^{\perp 0} \otimes (q^{\perp 1} \parr r^{\perp 1})$, $A_3 = r^{\perp 0} \otimes q^{\perp 0}$, and $\Gamma = A_1 A_2 A_3$. There is a proof net for Γ. We illustrate it with the following picture, where the elements of $\Omega_\Gamma^{\otimes}$ and $\Omega_\Gamma^{\parr}$ are depicted by $\otimes$ and $\parr$ respectively, the linear order $<_\Gamma$ goes from left to right, the relation $\prec_\Gamma$ is shown by dotted arrows, and the relations $\mathcal{A}$ and $\mathcal{E}$ are drawn on the upper semiplane.



Lemma 5.1. *A sequent $\to\Gamma$ is derivable in* NCL *if and only if there exists a proof net for Γ.*

Proof. This lemma is simply the constant-free case of Theorem 7.12 from [20]. $\square$

Intuitively, if $\langle \alpha, \beta \rangle \in \mathcal{E}$, then α and β come from the same axiom. If $\langle \langle A \otimes B, k \rangle, \beta \rangle \in \mathcal{A}$, then β designates the point where a sequent should be divided into two premises when $A \otimes B$ is introduced by an instance of the $\otimes$-introduction rule. Evidently, if $\beta = \langle C \parr D, l \rangle$, then the rule that introduces $C \parr D$ must be lower than the rule that introduces $A \otimes B$. Similarly, if $\langle E, k \rangle \prec_\Gamma \langle F, l \rangle$, then the rule that introduces F must be lower than the rule that introduces E. This explains the acyclicity condition in the definition of proof net.

Example. Consider the proof net from the previous example. It corresponds to the derivation

$$\dfrac{\to p^{\perp 1}\,p^{\perp 0} \qquad \dfrac{\dfrac{\to r^{\perp 1}\,r^{\perp 0} \quad \to q^{\perp 1}\,q^{\perp 0}}{\to q^{\perp 1}\,r^{\perp 1}\,(r^{\perp 0} \otimes q^{\perp 0})}}{\to (q^{\perp 1} \parr r^{\perp 1})\,(r^{\perp 0} \otimes q^{\perp 0})}}{\to p^{\perp 1}\,(p^{\perp 0} \otimes (q^{\perp 1} \parr r^{\perp 1}))\,(r^{\perp 0} \otimes q^{\perp 0})}\,.$$

In this derivation we use one of the following two generalized $\otimes$-introduction rules

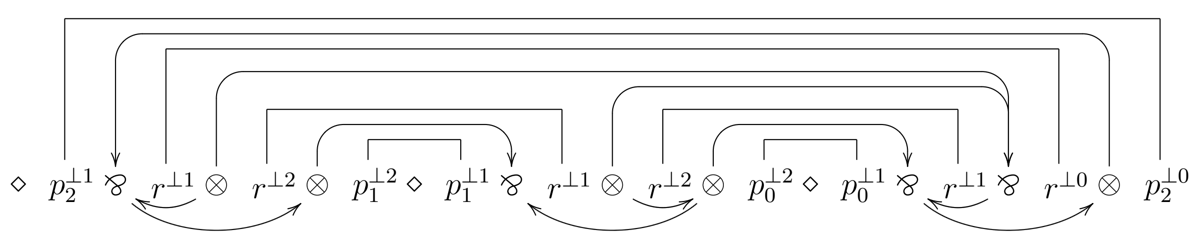
$$\dfrac{\to \Gamma\, A \quad \to \Phi\, B\, \Delta}{\to \Phi\, \Gamma\, (A \otimes B)\, \Delta}\,, \qquad\qquad \dfrac{\to \Gamma\, A\, \Pi \quad \to B\, \Delta}{\to \Gamma\, (A \otimes B)\, \Delta\, \Pi}\,.$$

These rules are admissible in NCL. If we include them in the calculus, then the two rules concerning cyclic permutation with double negation are no longer needed.

Example. Let

$$\Gamma = ((p_2^{\perp 1} \parr (r^{\perp 1} \otimes r^{\perp 2})) \otimes p_1^{\perp 2})\,(p_1^{\perp 1} \parr ((r^{\perp 1} \otimes r^{\perp 2}) \otimes p_0^{\perp 2}))\,((p_0^{\perp 1} \parr (r^{\perp 1} \parr r^{\perp 0})) \otimes p_2^{\perp 0}).$$

The following figure shows a proof net for Γ.



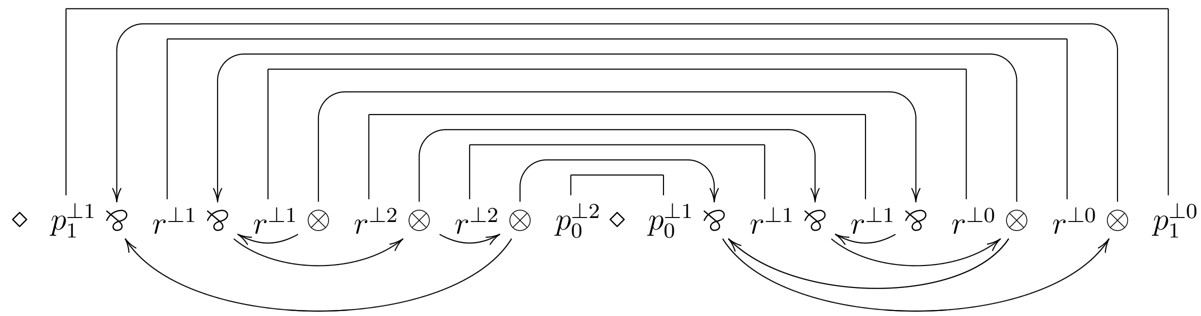
According to Lemma 5.1, $\mathrm{NCL} \vdash \to\Gamma$. In view of Lemma 4.1,

$$\mathrm{L}^* \vdash ((p_0 \setminus (r \setminus r)) \cdot p_1)\,(p_1 \setminus ((r \setminus r) \cdot p_2)) \to ((p_0 \setminus (r \setminus r)) \cdot p_2).$$

Example. Let

$\Gamma = (p_1^{\perp 1} \parr (((r^{\perp 1} \parr (r^{\perp 1} \otimes r^{\perp 2})) \otimes r^{\perp 2}) \otimes p_0^{\perp 2}))\, ((p_0^{\perp 1} \parr ((r^{\perp 1} \parr (r^{\perp 1} \parr r^{\perp 0})) \otimes r^{\perp 0})) \otimes p_1^{\perp 0})$.

The following figure illustrates the only way to construct a relational structure $\langle \mathbf{\Omega}_\Gamma, \mathcal{A}, \mathcal{E} \rangle$ that satisfies the first six conditions from the definition of proof net.



The graph $\langle \Omega_\Gamma, \prec_\Gamma \cup \mathcal{A} \rangle$ contains a cycle. In view of Lemma 5.1 and Lemma 4.1,

$$\mathrm{L}^* \nvdash ((p_0 \setminus (r \setminus ((r \setminus r) \cdot r))) \cdot p_1) \to ((p_0 \setminus ((r \setminus (r \setminus r)) \cdot r)) \cdot p_1).$$

6 Proof of the key lemma

In this section we prove Lemma 3.15. Recall that m is the number of clauses, used as a parameter in construction of the types H_i and F_i.

Let $k \geq 1$, $B \in \mathrm{Tp}$, and $\mathrm{L}^* \vdash F_1 \ldots F_{k-1} F_k \to B$. Let p_0^m and p_k^m occur only once in B and none of the variables $p_1^m, \ldots, p_{k-1}^m$ occur in B. We must prove that $\mathrm{L}^* \vdash F_1 \ldots F_{k-1} E_k^m(0) \to B$ or $\mathrm{L}^* \vdash F_1 \ldots F_{k-1} E_k^m(1) \to B$.

In view of Lemma 4.1 and Lemma 5.1, there exists a proof net $\langle \mathbf{\Omega}_\Gamma, \mathcal{A}, \mathcal{E} \rangle$ for $\Gamma = (\widehat{F_k}^\perp)(\widehat{F_{k-1}}^\perp) \ldots (\widehat{F_1}^\perp)\widehat{B}$. By definitions,

$$\widehat{F_i}^\perp = (\widehat{E_i^m(0)}^\perp \otimes \widehat{H_i}^{\perp\perp}) \parr (\widehat{H_i}^\perp \parr (\widehat{H_i} \otimes \widehat{E_i^m(1)}^\perp))$$

for every i. Note that $\|\widehat{H_i}\| = \|\widehat{E_i^m(0)}\| = \|\widehat{E_i^m(1)}\| = 4(m+1)$ and $\|\widehat{F_i}\| = 20(m+1)$ for every i.

Evidently $\Omega^{\mathrm{At}}_{\widehat{F_k}^\perp} \subset \Omega^{\mathrm{At}}_\Gamma$. According to definitions,

$$\begin{aligned}
\Omega^{\mathrm{At}}_{\widehat{F_k}^\perp} = {} & \{\langle (p_k^j)^{\perp 1}, 2(m-j)+1 \rangle \mid 0 \leq j \leq m\} \\
& \cup \{\langle (p_{k-1}^j)^{\perp 2}, 4(m+1) - (2(m-j)+1) \rangle \mid 0 \leq j \leq m\} \\
& \cup \{\langle (p_{k-1}^j)^{\perp 3}, 4(m+1) + (2(m-j)+1) \rangle \mid 0 \leq j \leq m\} \\
& \cup \{\langle (p_k^j)^{\perp 2}, 8(m+1) - (2(m-j)+1) \rangle \mid 0 \leq j \leq m\} \\
& \cup \{\langle (p_k^j)^{\perp 1}, 8(m+1) + (2(m-j)+1) \rangle \mid 0 \leq j \leq m\} \\
& \cup \{\langle (p_{k-1}^j)^{\perp 2}, 12(m+1) - (2(m-j)+1) \rangle \mid 0 \leq j \leq m\} \\
& \cup \{\langle (p_{k-1}^j)^{\perp 1}, 12(m+1) + (2(m-j)+1) \rangle \mid 0 \leq j \leq m\} \\
& \cup \{\langle (p_k^j)^{\perp 0}, 16(m+1) - (2(m-j)+1) \rangle \mid 0 \leq j \leq m\} \\
& \cup \{\langle (p_k^j)^{\perp 1}, 16(m+1) + (2(m-j)+1) \rangle \mid 0 \leq j \leq m\} \\
& \cup \{\langle (p_{k-1}^j)^{\perp 2}, 20(m+1) - (2(m-j)+1) \rangle \mid 0 \leq j \leq m\}.
\end{aligned}$$

We shall use the following abbreviations

$$\begin{aligned}
\alpha_1 &= \langle (p_k^m)^{\perp 1}, 1\rangle,\\
\beta_2 &= \langle (p_{k-1}^m)^{\perp 2}, 4(m+1)-1\rangle,\\
\beta_3 &= \langle (p_{k-1}^m)^{\perp 3}, 4(m+1)+1\rangle,\\
\alpha_4 &= \langle (p_k^m)^{\perp 2}, 8(m+1)-1\rangle,\\
\alpha_5 &= \langle (p_k^m)^{\perp 1}, 8(m+1)+1\rangle,\\
\beta_6 &= \langle (p_{k-1}^m)^{\perp 2}, 12(m+1)-1\rangle,\\
\beta_7 &= \langle (p_{k-1}^m)^{\perp 1}, 12(m+1)+1\rangle,\\
\alpha_8 &= \langle (p_k^m)^{\perp 0}, 16(m+1)-1\rangle,\\
\alpha_9 &= \langle (p_k^m)^{\perp 1}, 16(m+1)+1\rangle,\\
\beta_{10} &= \langle (p_{k-1}^m)^{\perp 2}, 20(m+1)-1\rangle.
\end{aligned}$$

Our nearest task is to prove that either

$$\langle \alpha_1, \alpha_8\rangle \in \mathcal{E} \tag{1}$$

or

$$\langle \beta_3, \beta_{10}\rangle \in \mathcal{E}. \tag{2}$$

Note that $\#_{p_0^m}(B) = \#_{p_0^m}(F_1 \ldots F_{k-1}F_k) = -1$ and $\#_{p_k^m}(B) = \#_{p_k^m}(F_1 \ldots F_{k-1}F_k) = 1$. Thus the only occurrence of p_k^m in B is a "positive occurrence" and the corresponding element of Ω_Γ is of the form $\langle (p_k^m)^{\perp n'}, l'\rangle$, where n' is even, and consequently

$$\langle \alpha_8, \langle (p_k^m)^{\perp n'}, l'\rangle\rangle \notin \mathcal{E}.$$

According to the definition of proof net, there are only two possible values for the $\mathcal{E}$-image of α_8. If $\langle \alpha_8, \alpha_1\rangle \in \mathcal{E}$, then we have (1). Let $\langle \alpha_8, \alpha_5\rangle \in \mathcal{E}$. Then we consider the $\mathcal{E}$-image of β_3. It can not come from the type B (if $k \geq 2$, then there are no occurrences of p_{k-1}^m in B; if $k = 1$, then the only occurrence of p_0^m in B is a "negative occurrence" and the corresponding element of Ω_Γ is of the form $\langle (p_0^m)^{\perp n''}, l''\rangle$, where n'' is odd).

Evidently, if $k \geq 2$, then the occurrences of p_{k-1}^m and p_{k-2}^m in F_{k-1} contribute the following elements to $\Omega_\Gamma^{\mathrm{At}}$:

$$\begin{aligned}
\beta_{11} &= \langle (p_{k-1}^m)^{\perp 1}, 20(m+1)+1\rangle,\\
\gamma_{12} &= \langle (p_{k-2}^m)^{\perp 2}, 24(m+1)-1\rangle,\\
\gamma_{13} &= \langle (p_{k-2}^m)^{\perp 3}, 24(m+1)+1\rangle,\\
\beta_{14} &= \langle (p_{k-1}^m)^{\perp 2}, 28(m+1)-1\rangle,\\
\beta_{15} &= \langle (p_{k-1}^m)^{\perp 1}, 28(m+1)+1\rangle,\\
\gamma_{16} &= \langle (p_{k-2}^m)^{\perp 2}, 32(m+1)-1\rangle,\\
\gamma_{17} &= \langle (p_{k-2}^m)^{\perp 1}, 32(m+1)+1\rangle,\\
\beta_{18} &= \langle (p_{k-1}^m)^{\perp 0}, 36(m+1)-1\rangle,\\
\beta_{19} &= \langle (p_{k-1}^m)^{\perp 1}, 36(m+1)+1\rangle,\\
\gamma_{20} &= \langle (p_{k-2}^m)^{\perp 2}, 40(m+1)-1\rangle.
\end{aligned}$$

We consider three cases depending on the value of the $\mathcal{E}$-image of β_3.
Case 1: $k \geq 1$ and $\langle \beta_3, \beta_6 \rangle \in \mathcal{E}$.
Together with $\langle \alpha_8, \alpha_5 \rangle \in \mathcal{E}$ this contradicts $<_\Gamma$-planarity of $\mathcal{E}$.
Case 2: $k \geq 1$ and $\langle \beta_3, \beta_{10} \rangle \in \mathcal{E}$.
In this case we have (2).
Case 3: $k \geq 2$ and $\langle \beta_3, \beta_{14} \rangle \in \mathcal{E}$.
But then γ_{12} can have no $\mathcal{E}$-image without contradicting $<_\Gamma$-planarity of $\mathcal{E}$.

Thus we have proved that either (1) or (2) holds.

Suppose (1) holds. We denote $\Gamma' = (\widehat{E_k^m(1)}^\perp)(\widehat{F_{k-1}}^\perp)\ldots(\widehat{F_1}^\perp)\widehat{B}$. Consider the function $g\colon \Omega_{\Gamma'} \to \Omega_\Gamma$ such that for every $\langle C, l \rangle \in \Omega_{\Gamma'}$

$$g(\langle C, l \rangle) = \begin{cases} \langle C, l \rangle & \text{if } l = 0, \\ \langle C, l + 16(m+1) \rangle & \text{if } l > 0. \end{cases}$$

This function is defined correctly (note that $\|\widehat{F_k}^\perp\| - \|\widehat{E_k^m(1)}^\perp\| = 16(m+1)$). Evidently g maps $\Omega_{\Gamma'}^{\mathrm{At}}$ to $\Omega_\Gamma^{\mathrm{At}}$, $\Omega_{\Gamma'}^{\otimes}$ to $\Omega_\Gamma^{\otimes}$, and $\Omega_{\Gamma'}^{\wp\diamond}$ to $\Omega_\Gamma^{\wp\diamond}$. We put $\mathcal{A}' = \{\langle \eta, \theta \rangle \mid \langle g(\eta), g(\theta) \rangle \in \mathcal{A}\}$ and $\mathcal{E}' = \{\langle \eta, \theta \rangle \mid \langle g(\eta), g(\theta) \rangle \in \mathcal{E}\}$. Note that if $\langle \mu, \nu \rangle \in \mathcal{A} \cup \mathcal{E}$ and μ is in the range of g, then ν is in the range of g (since $\langle \alpha_1, \alpha_8 \rangle \in \mathcal{E}$ and $\mathcal{A} \cup \mathcal{E}$ is $<_\Gamma$-planar). Now it is easy to verify that $\langle \mathbf{\Omega}_{\Gamma'}, \mathcal{A}', \mathcal{E}' \rangle$ is a proof net. According to Lemma 5.1 and Lemma 4.1, $\mathrm{L}^* \vdash F_1 \ldots F_{k-1} E_k^m(1) \to B$.

Now suppose (2) holds. In this case we put $\Gamma' = (\widehat{E_k^m(0)}^\perp)(\widehat{F_{k-1}}^\perp)\ldots(\widehat{F_1}^\perp)\widehat{B}$ and define the function $g\colon \Omega_{\Gamma'} \to \Omega_\Gamma$ as follows:

$$g(\langle C, l \rangle) = \begin{cases} \langle C, l \rangle & \text{if } l < 4(m+1), \\ \langle C, l + 16(m+1) \rangle & \text{if } l \geq 4(m+1). \end{cases}$$

We put $\mathcal{A}' = \{\langle \eta, \theta \rangle \mid \langle g(\eta), g(\theta) \rangle \in \mathcal{A}\}$ and $\mathcal{E}' = \{\langle \eta, \theta \rangle \mid \langle g(\eta), g(\theta) \rangle \in \mathcal{E}\}$. Again it can be verified that $\langle \mathbf{\Omega}_{\Gamma'}, \mathcal{A}', \mathcal{E}' \rangle$ is a proof net. According to Lemma 5.1 and Lemma 4.1, $\mathrm{L}^* \vdash F_1 \ldots F_{k-1} E_k^m(0) \to B$.

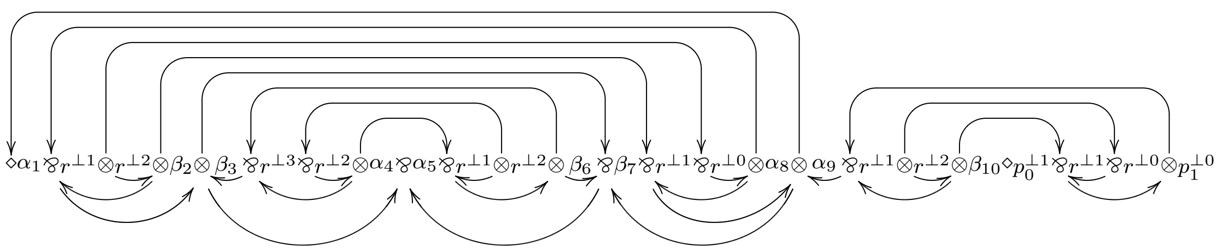
This completes the proof of Lemma 3.15.

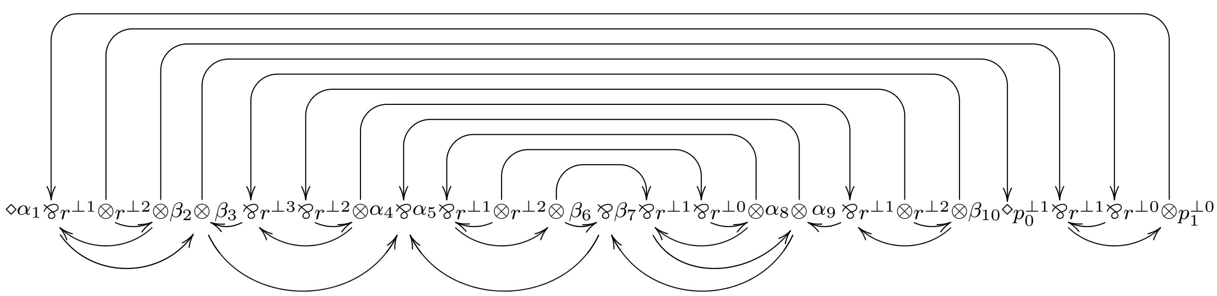
In the construction of Section 2 all the variables p_i^j for $j < m$ may be replaced by one variable r (then the construction involves only $n+2$ different variables). It can be verified that then Lemma 3.14 and Lemma 3.15 remain provable (in this case it is more convenient to use proof nets for both lemmas). Thus also such simplified reduction is correct.

Example. Consider the Boolean formula $x_1 \vee \neg x_1$. If we write p_i instead of p_i^m, then the simplified construction yields $H_1 = (p_0 \setminus ((r \setminus r) \cdot p_1))$, $E_1^1(1) = E_1^1(0) = G = ((p_0 \setminus (r \setminus r)) \cdot p_1)$, and $F_1 = ((E_1^1(1) \,/\, H_1) \cdot H_1) \cdot (H_1 \setminus E_1^1(0))$. Evidently,

$$\widehat{F_1}^{\perp} = \Big(\Big((p_1^{\perp 1} \parr ((r^{\perp 1} \otimes r^{\perp 2}) \otimes p_0^{\perp 2})) \otimes (p_0^{\perp 3} \parr ((r^{\perp 3} \parr r^{\perp 2}) \otimes p_1^{\perp 2}))\Big)$$
$$\parr \Big(((p_1^{\perp 1} \parr (r^{\perp 1} \otimes r^{\perp 2})) \otimes p_0^{\perp 2}) \parr \Big((p_0^{\perp 1} \parr ((r^{\perp 1} \parr r^{\perp 0}) \otimes p_1^{\perp 0})) \otimes (p_1^{\perp 1} \parr ((r^{\perp 1} \otimes r^{\perp 2}) \otimes p_0^{\perp 2}))\Big)\Big)\Big),$$
$$\widehat{G} = ((p_0^{\perp 1} \parr (r^{\perp 1} \parr r^{\perp 0})) \otimes p_1^{\perp 0}).$$

There are two proof nets for $(\widehat{F_1}^{\perp})\widehat{G}$.





Here we have omitted the binary relation $\mathcal{E}$. The symbols $\alpha_1, \ldots, \beta_{10}$ are taken from the proof of Lemma 3.15. The first proof net corresponds to the satisfying assignment $x_1 = 1$, the second one corresponds to $x_1 = 0$.

Fragments

F. Métayer [15] has proved that the decision problems for propositional multiplicative cyclic linear logic, its single-variable fragment, and its constant-only fragment are polynomially equivalent. Thus both these fragments are NP-complete. F. Métayer's method applies also to Abrusci's noncommutative linear logic (a simple translation between multiplicative noncommutative linear logic and multiplicative cyclic linear logic can be found in [19]). Moreover, the construction from [15] can be easily adapted to prove NP-completeness of the single-variable fragments of L and L* (variables q_i, where $0 < i < k$, are replaced by the types $\underbrace{r \setminus \ldots \setminus r \setminus}_{i \text{ times}} r \underbrace{/ \, r \, / \ldots / \, r}_{k-i \text{ times}}$, where $r \in \mathrm{Var}$).

Some natural fragments of Lambek calculus are known to be decidable in polynomial time (see e. g. [1]).

It should be mentioned that another closely related system, the non-associative variant of Lambek calculus, is decidable in polynomial time (see [2, 9]).

Conclusion

We have proved that the decision problem for the calculus L (and for L^*) is NP-complete. It is well-known that the multiplicative fragment of noncommutative linear logic is in NP and that it is conservative over L^*. Thus also the multiplicative fragment of noncommutative linear logic is NP-complete. The same holds for the multiplicative fragment of cyclic linear logic.

Acknowledgements

I would like to thank Prof. S. Artemov for his support during many years of research in this area.

References

[1] E. Aarts, *Proving theorems of the second order Lambek calculus in polynomial time*, Studia Logica **53** (1994), pp. 373–387.

[2] E. Aarts and K. Trautwein, *Non-associative Lambek categorial grammar in polynomial time*, Mathematical Logic Quarterly **41** (1995), pp. 476–484.

[3] V. M. Abrusci, *Phase semantics and sequent calculus for pure noncommutative classical linear propositional logic*, The Journal of Symbolic Logic **56** (1991), no. 4, pp. 1403–1451.

[4] W. Buszkowski, *Mathematical linguistics and proof theory*, In: Handbook of Logic and Language (J. van Benthem and A. ter Meulen, eds.), Elsevier/MIT Press, 1997, Chapter 12, pp. 683–736.

[5] B. Carpenter, *Type-Logical Semantics*, MIT Press, Cambridge, Massachusetts, 1997.

[6] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman and Company, San-Francisco, 1979.

[7] Ph. de Groote, *An algebraic correctness criterion for intuitionistic proof-nets*, 4th International Symposium on Logical Foundations of Computer Science, LFCS'97 (S. Adian and A. Nerode, eds.), Springer, Berlin, 1997, pp. 130–140.

[8] Ph. de Groote, *An algebraic correctness criterion for intuitionistic proof-nets*, Theoretical Computer Science **224** (1999), no. 1–2, pp. 115–134.

[9] Ph. de Groote, *The non-associative Lambek calculus with product in polynomial time*, In: Automated Reasoning with Analytic Tableaux and Related Methods (N. V. Murray, ed.), Springer, 1999, pp. 128–139.

[10] M. I. Kanovich, *The multiplicative fragment of linear logic is* NP*-complete*, ITLI Prepublication Series X–91–13, University of Amsterdam, 1991.

[11] M. I. Kanovich, *Horn programming in linear logic is* NP*-complete*, Proceedings of the 7th Annual IEEE Symposium on Logic in Computer Science, IEEE Computer Society Press, 1992, pp. 200–210.

[12] J. Lambek, *The mathematics of sentence structure*, American Mathematical Monthly **65** (1958), no. 3, pp. 154–170.

[13] J. Lambek, *From categorial grammar to bilinear logic*, Substructural Logics (K. Došen and P. Schroeder-Heister, eds.), Oxford University Press, 1993, pp. 207–237.

[14] P. Lincoln and T. Winkler, *Constant-only multiplicative linear logic is* NP*-complete*, Theoretical Computer Science **135** (1994), no. 1, pp. 155–169.

[15] F. Métayer, *Polynomial equivalence among systems* LLNC*,* LLNC_a *and* LLNC_0, Theoretical Computer Science **227** (1999), pp. 221–229.

[16] M. Moortgat, *Categorial Type Logics*, In: Handbook of Logic and Language (J. van Benthem and A. ter Meulen, eds.), North-Holland, Amsterdam, 1997, Chapter 2.

[17] G. Morrill, *Memoisation of categorial proof nets: parallelism in categorial processing*, In: Proofs and Linguistic Categories, Proceedings 1996 Roma Workshop, (V. M. Abrusci and C. Casadio, eds.), 1996, pp. 157–169.

[18] G. Penn, *A graph-theoretic approach to sequent derivability in the Lambek calculus*, Electronic Notes in Theoretical Computer Science **53** (2002), 22 pp.

[19] M. Pentus, *Equivalence of multiplicative fragments of cyclic linear logic and noncommutative linear logic*, 4th International Symposium on Logical Foundations of Computer Science, LFCS'97 (S. Adian and A. Nerode, eds.), Springer, Berlin, 1997, pp. 306–311.

[20] M. Pentus, *Free monoid completeness of the Lambek calculus allowing empty premises*, Logic Colloquium '96: Proceedings of the colloquium held in San Sebastian, Spain, July 9–15, 1996 (J. M. Larrazabal, D. Lascar, and G. Mints, eds.), Springer, Berlin, 1998, pp. 171–209.

[21] D. Roorda, *Resource Logics: Proof-theoretical Investigations*, Ph.D. dissertation, Universiteit van Amsterdam, 1991.

[22] J. van Benthem, *Language in Action: Categories, Lambdas and Dynamic Logic*, North-Holland, Amsterdam, 1991.

[23] D. N. Yetter, *Quantales and noncommutative linear logic*, Journal of Symbolic Logic **55** (1990), no. 1, pp. 41–64.

Author address

Department of Mathematical Logic and Theory of Algorithms
Faculty of Mechanics and Mathematics
Moscow State University
Moscow, Russia, 119992
E-mail address: pentus@lpcs.math.msu.ru
URL: http://markov.math.msu.ru/~pentus/