On the Proof Theory of Non-Commutative Subexponentials

Max Kanovich[a], Stepan Kuznetsov[b], Vivek Nigam[c], and Andre Scedrov[d]

[a]University College London, London, UK
and National Research University Higher School of Economics, Moscow, Russia
max.kanovich@qmul.ac.uk
[b]Steklov Mathematical Institute, Moscow, Russia
sk@mi.ras.ru
[c]Federal University of Paraíba, João Pessoa, Brazil
vivek.nigam@gmail.com
[d]University of Pennsylvania, Philadelphia, USA
and National Research University Higher School of Economics, Moscow, Russia
scedrov@math.upenn.edu

Abstract

Linear logical frameworks with subexponentials have been used for the specification of among other systems, proof systems, concurrent programming languages and linear authorization logics. In these frameworks, subexponentials can be configured to allow or not for the application of the contraction and weakening rules while the exchange rule can always be applied. This means that formulae in such frameworks can only be organized as sets and multisets of formulae not being possible to organize formulae as lists of formulae. This paper investigates the proof theory of linear logic proof systems in the non-commutative variant. These systems can disallow the application of exchange rule on some subexponentials. We investigate conditions for when cut-elimination is admissible in the presence of non-commutative subexponentials, investigating the interaction of the exchange rule with local and non-local contraction rules. We also obtain some new undecidability results on non-commutative linear logic with subexponentials.

To Dale Miller's Festschrift and his Contributions to Logic in Computer Science. Dale's work has been an inspiration to us. He is a great researcher, colleague, advisor and friend.

1 Introduction

Logic and proof theory have played an important role in computer science. The introduction of linear logic by Girard is an example of how the beauty of logic can be applied to the principles of computer science. More than 20 years ago, Hodas and Miller [3, 4] proposed the intuitionistic linear logical framework, Lolli, which distinguishes between to kinds of formulae: *linear*, that cannot be contracted and weakened, and *unbounded*, that can be contracted and weakened.[1]. In contrast to existing intuitionistic/classical logical frameworks, Lolli allowed to express stateful computations using logical connectives. Some years later, Miller proposed the classical linear logical framework Forum [12, 13] demonstrating that linear logic can be used among other things to design concurrent systems.[2]

Forum sequents organizes formulae into two contexts $\Psi : \Delta$, where Ψ is a set of unbounded formulae, and Δ is a multiset of linear formulae. Formally, all formulae in Ψ are assumed to be marked with a bang !. Thus, Forum allows to organize formulae into exactly one set of unbounded formulae and exactly one multiset of linear formulae.

[1]The authors received the LICS Test of Time Award for this work
[2]For this work, Miller received yet another LICS Test of Time Award prize.

It has been known, however, since Girard's original linear logic paper [2], that the linear logic exponentials $!, ?$ are not canonical. Indeed, proof systems with non-equivalent exponentials [1] can be formulated. Nigam and Miller [18] called them subexponentials and proposed a more expressive linear logical framework called SELL which allows for the specification of any number of non-equivalent subexponentials $!^{\mathsf{s}}, ?^{\mathsf{s}}$. Each subexponential can be specified to behave as linear or as unbounded. This is reflected in the syntax. SELL sequents associate a different context to each subexponential. Thus formulae may be organized into a number of sets of unbounded formulae and a number of multisets of linear formulae. Nigam and Miller show that SELL is more expressive than Forum being capable of expressing algorithmic specifications in logic. In the recent years, it has been shown that SELL can also be used to specify linear authorization logics [16, 17], concurrent constraint programming languages [19, 21] and proof systems [20].

While these logical frameworks have been sucessfully used for a number of applications, they do not allow formulae to be organized as lists of formulae. This is because all the frameworks above assume that the exchange rule can be applied to any formula. This paper investigates the proof theory of subexponentials in non-commutative linear logic. Our contribution is two-fold:

1. We construct general non-commutative linear logic proof systems with subexponentials and investigate conditions for when these systems fail to admit cut-elimination.
2. For systems, in which at least one subexponential obeys the contraction rule in its non-local form, we prove undecidability results.

2 The Lambek Calculus with One Subexponential

We start with the version of the Lambek calculus allowing empty antecedents [9]. The Lambek calculus is historically the oldest version of non-commutative linear logic, and we take it as the base system. Formulae of the Lambek calculus are built from propositional variables using three connectives: $\cdot$ (product), $/$ (left division), and $\backslash$ (right division). From the logical perspective, product is conjunction, and implication in the non-commutative situation splits into two operations, called the left and right division operations. Sequents of the Lambek calculus are expressions of the form $\Pi \to A$, where A is a formula, and Π is an ordered sequence of types. In contrast to the commutative case, Π is not a set or a multiset: the order matters! The axioms and rules of the Lambek calculus are depicted in Figure 1.

$$\frac{}{A \to A}\ (\mathrm{id})$$

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

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

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

Figure 1: Lambek Calculus

The cut rule
$$\frac{\Pi \to A \quad \Gamma, A, \Delta \to C}{\Gamma, \Pi, \Delta \to C}\ (\mathrm{cut})$$
is admissible in this calculus [9].

We extend the Lambek calculus with a unary connective, !, with the following two introduction rules:

$$\frac{\Delta_1, A, \Delta_2 \to C}{\Delta_1, !A, \Delta_2 \to C}\ (! \to) \qquad \frac{!A_1, \ldots, !A_n \to B}{!A_1, \ldots, !A_n \to !B}\ (\to !)$$

and *some* of the following structural rules:

1. exchange:

$$\frac{\Delta_1, \Delta_2, !A, \Delta_3 \to C}{\Delta_1, !A, \Delta_2, \Delta_3 \to C}\ (\mathrm{ex}_1) \qquad \frac{\Delta_1, !A, \Delta_2, \Delta_3 \to C}{\Delta_1, \Delta_2, !A, \Delta_3 \to C}\ (\mathrm{ex}_2)$$

2. weakening:

$$\frac{\Delta_1, \Delta_2 \to C}{\Delta_1, !A, \Delta_2 \to C}\ (\mathrm{weak})$$

3. local contraction:

$$\frac{\Delta_1, !A, !A, \Delta_2 \to C}{\Delta_1, !A, \Delta_2 \to C}\ (\mathrm{contr})$$

4. non-local contraction:

$$\frac{\Delta_1, !A, \Delta_2, !A, \Delta_3 \to C}{\Delta_1, !A, \Delta_2, \Delta_3 \to C}\ (\mathrm{ncontr}_1) \qquad \frac{\Delta_1, !A, \Delta_2, !A, \Delta_3 \to C}{\Delta_1, \Delta_2, !A, \Delta_3 \to C}\ (\mathrm{ncontr}_2)$$

These rules are not independent. In particular, the following holds (both with and without (cut)):

Proposition 2.1.

(contr) *and* $(\mathrm{ex}_{1,2})$ *yield* $(\mathrm{ncontr}_{1,2})$*;*

$(\mathrm{ncontr}_{1,2})$ *and* (weak) *yield* $(\mathrm{ex}_{1,2})$*.*

3 Cut vs. Contraction

Notably enough, some combinations of the rules for ! lead to problems. Namely, if one takes only the local version of contraction rule (contr) for !, cut elimination **fails.**

Theorem 3.1. *The proof system obtained by adding the rules* $(! \to)$, $(\to !)$, (contr)*, and, optionally,* (weak) *to the Lambek calculus (Figure 1) does not admit* (cut)*.*

Proof. One can take the following sequent as a counter-example:

$$r \,/\, q, !p, !(p \setminus q), q \setminus s \to r \cdot s.$$

This sequent has a proof with (cut):

$$
\begin{prooftree}
\AxiomC{$p \to p$}
\AxiomC{$q \to q$}
\RightLabel{$(\backslash \to)$}
\BinaryInfC{$p, p \mathbin{\backslash} q \to q$}
\RightLabel{$(! \to)$ twice}
\UnaryInfC{$!p, !(p \mathbin{\backslash} q) \to q$}
\RightLabel{$(\to !)$}
\UnaryInfC{$!p, !(p \mathbin{\backslash} q) \to !q$}
\AxiomC{$q \to q$}
\AxiomC{$q \to q$}
\AxiomC{$r \to r$}
\AxiomC{$s \to s$}
\RightLabel{$(\to \cdot)$}
\BinaryInfC{$r, s \to r \cdot s$}
\RightLabel{$(\backslash \to)$}
\BinaryInfC{$r, q, q \mathbin{\backslash} s \to r \cdot s$}
\RightLabel{$(/ \to)$}
\BinaryInfC{$r \mathbin{/} q, q, q, q \mathbin{\backslash} s \to r \cdot s$}
\RightLabel{$(! \to)$ twice}
\UnaryInfC{$r \mathbin{/} q, !q, !q, q \mathbin{\backslash} s \to r \cdot s$}
\RightLabel{(contr)}
\UnaryInfC{$r \mathbin{/} q, !q, q \mathbin{\backslash} s \to r \cdot s$}
\RightLabel{(cut)}
\BinaryInfC{$r \mathbin{/} q, !p, !(p \mathbin{\backslash} q), q \mathbin{\backslash} s \to r \cdot s$}
\end{prooftree}
$$

but doesn't have a cut-free proof. In order to verify the latter, we notice that due to subformula and polarity properties, the only rules that can be applied are $(/ \to)$, $(\backslash \to)$, $(\to \cdot)$, $(! \to)$, and (contr). Moreover, since ! appears only on the top level, the rules operating ! can be moved to the very bottom of the proof (this is actually a small *focusing* instance here). These rules would be applied to a (pure Lambek) sequent of the form

$$r \mathbin{/} q, p, \ldots, p, (p \mathbin{\backslash} q), \ldots, (p \mathbin{\backslash} q), q \mathbin{\backslash} s \to r \cdot s$$

but an easy proof search attempt shows that none of these sequents is derivable in the Lambek calculus. □

This failure of cut elimination of the calculus with (contr) motivates its replacement by the non-local version ($\mathrm{ncontr}_{1,2}$) or alternatively by local contractions and exchange rules (in this case, by Proposition 2.1, ($\mathrm{ncontr}_{1,2}$) become admissible).

With non-local contraction, the sequent used in the proof of Theorem 3.1 obtains a cut-free proof:

$$
\begin{prooftree}
\AxiomC{$p \to p$}
\AxiomC{$p \to p$}
\AxiomC{$q \to q$}
\AxiomC{$q \to q$}
\AxiomC{$r \to r$}
\AxiomC{$s \to s$}
\RightLabel{$(\to \cdot)$}
\BinaryInfC{$r, s \to r \cdot s$}
\RightLabel{$(\backslash \to)$}
\BinaryInfC{$r, q, q \mathbin{\backslash} s \to r \cdot s$}
\RightLabel{$(/ \to)$}
\BinaryInfC{$r \mathbin{/} q, q, q, q \mathbin{\backslash} s \to r \cdot s$}
\RightLabel{$(\backslash \to)$ twice}
\TrinaryInfC{$r \mathbin{/} q, p, p \mathbin{\backslash} q, p, p \mathbin{\backslash} q, q \mathbin{\backslash} s \to r \cdot s$}
\RightLabel{$(! \to)$ 4 times}
\UnaryInfC{$r \mathbin{/} q, !p, !(p \mathbin{\backslash} q), !p, !(p \mathbin{\backslash} q), q \mathbin{\backslash} s \to r \cdot s$}
\RightLabel{(ncontr_2)}
\UnaryInfC{$r \mathbin{/} q, !p, !p, !(p \mathbin{\backslash} q), q \mathbin{\backslash} s \to r \cdot s$}
\RightLabel{(contr)}
\UnaryInfC{$r \mathbin{/} q, !p, !(p \mathbin{\backslash} q), q \mathbin{\backslash} s \to r \cdot s$}
\end{prooftree}
$$

Another motivation for the non-local contraction rule is given by the following linguistic example [14][6]. Let np stand for *noun phrase*, n for *noun*, s for *sentence*, gc for *gerund clause*.

$$\begin{array}{ccccc} np, & (np \mathbin{\backslash} s) \mathbin{/} np, & np \mathbin{/} n, & n & \to s \\ \text{John} & \text{signed} & \text{the} & \text{document} & \end{array}$$

$$\begin{array}{cccccccc} np, & (np \mathbin{\backslash} s) \mathbin{/} np, & np \mathbin{/} n, & n, & (np \mathbin{\backslash} s) \mathbin{/} gc, & gc \mathbin{/} np, & np & \to s \\ \text{John} & \text{signed} & \text{the} & \text{document} & \text{without} & \text{reading} & \text{it} & \end{array}$$

The next two constructions are called "extraction" and "parasitic extraction" respectively:

$$\begin{array}{ccccc} n, & (n \mathbin{\backslash} n) \mathbin{/} (s \mathbin{/} np), & np, & (np \mathbin{\backslash} s) \mathbin{/} np & \to n \\ \text{document} & \text{that} & \text{John} & \text{signed} & \end{array}$$

$$
\dfrac{\dfrac{\dfrac{\dfrac{np, (np \setminus s) / np, np, (np \setminus s) / gc, gc / np, np \to s}{np, (np \setminus s) / np, !np, (np \setminus s) / gc, gc / np, !np \to s}}{np, (np \setminus s) / np, (np \setminus s) / gc, gc / np, !np \to s}\ (\mathrm{ncontr}_2)}{np, (np \setminus s) / np, (np \setminus s) / gc, gc / np \to s \,/\, !np} \qquad n, n \setminus n \to n}{\begin{array}{ccccccc} n, & (n \setminus n) / (s / !np), & np, & (np \setminus s) / np, & (np \setminus s) / gc, & gc / np & \to n \\ \text{document} & \text{that} & \text{John} & \text{signed} & \text{without} & \text{reading} & \end{array}}
$$

Modelling parasitic extraction requires the contraction rule in its non-local form.

4 Undecidability of the Lambek calculus with Non-Local Contraction

In this section we prove the following that once we have a subexponential ! that allows nonlocal contraction $(\mathrm{ncontr}_{1,2})$, the derivability problem becomes undecidable:

Theorem 4.1. *The derivability problem for the Lambek calculus enriched with a subexponential ! that allows* $(\mathrm{ncontr}_{1,2})$ *(and, possibly, also* $(\mathrm{ex}_{1,2})$ *and/or* (weak)*) is r.e.-complete (and therefore undecidable).*

The proof is very close to the one presented in [7], using ideas from [10] and [5].

In our undecidability proof we encode word rewriting (semi-Thue) systems. A word rewriting system over alphabet Σ is a finite set P of pairs of words over Σ. Elements of P are called *rewriting rules* and are applied as follows: if $\langle \alpha, \beta \rangle \in P$, then $\eta\, \alpha\, \theta \Rightarrow \eta\, \beta\, \theta$ for arbitrary (possibly empty) words η and θ over Σ. The relation $\Rightarrow^*$ is the reflexive transitive closure of $\Rightarrow$.

The following classical result appears (independently) in works of Markov (Jr.) [11] and Post [23].

Theorem 4.2. *There exists a word rewriting system P such that the set $\{\langle \gamma, \delta \rangle \mid \gamma \Rightarrow^* \delta\}$ is r.e.-complete (and therefore undecidable).*

In our encoding we'll actually need the weakening rule. However, our subexponential doesn't necessarily enjoy it. To simulate weakening, we add the unit constant to our version of the Lambek calculus with !; in the end of this section we show how to eliminate it. For the unit constant, $\mathbf{1}$, we have one axiom $\to \mathbf{1}$ and one rule:

$$
\frac{\Gamma, \Delta \to C}{\Gamma, \mathbf{1}, \Delta \to C}\ (\mathbf{1} \to)
$$

(actually it is weakening, but for $\mathbf{1}$ rather than $!A$.

Let P be the word rewriting system from Theorem 4.2 and consider all elements of Σ as variables of the Lambek calculus. We convert rewriting rules of P into Lambek formulae in the following way:

$$
\mathcal{B} = \{(u_1 \cdot \dots \cdot u_k) / (v_1 \cdot \dots \cdot v_m) \mid \langle u_1 \dots u_k, v_1 \dots v_m \rangle \in P\}.
$$

If $\mathcal{B} = \{B_1, \dots, B_n\}$ (we can take any ordering of $\mathcal{B}$), let

$$
\Phi = !(\mathbf{1} / !B_1), !B_1, \dots, !(\mathbf{1} / !B_n), !B_n, \Gamma = !B_1, \dots, !B_n.
$$

Finally, we consider a *theory* (finite set of sequents) $\mathcal{T}$ associated with P:

$$\mathcal{T} = \{v_1, \dots, v_m \to u_1 \cdot \ldots \cdot u_k \mid \langle u_1 \dots u_k, v_1 \dots v_m \rangle \in P\}.$$

When talking about *derivability from theory* $\mathcal{T}$, we use the rules of the original Lambek calculus, including cut.

Lemma 4.3. *Let $\gamma = a_1 \dots a_l$ and $\delta = b_1 \dots b_k$ be arbitrary words over Σ. Then the following are equivalent:*

1. $\gamma \Rightarrow^* \delta$;
2. *the sequent $\Phi, b_1, \dots, b_k \to a_1 \cdot \ldots \cdot a_l$ is derivable in our extension of the Lambek calculus;*
3. *the sequent $\Gamma, b_1, \dots, b_k \to a_1 \cdot \ldots \cdot a_l$ is derivable in the extension of the Lambek calculus in which $!$ enjoys all structural rules, in particular,* (weak)*;*
4. *the sequent $b_1, \dots, b_k \to a_1 \cdot \ldots \cdot a_l$ is derivable from $\mathcal{T}$.*

Proof. $\boxed{1 \Rightarrow 2}$ Proceed by induction on $\Rightarrow^*$. The base case ($\gamma \Rightarrow^* \gamma$) is handled as follows:

$$\dfrac{\dfrac{B_1 \to B_1 \quad \dots \quad B_n \to B_n \qquad \dfrac{\dfrac{a_1 \to a_1 \quad \dots \quad a_m \to a_l}{a_1 \dots a_m \to a_1 \cdot \ldots \cdot a_l}\,(\to \cdot)\ (l-1) \text{ times}}{\mathbf{1}, \dots, \mathbf{1}, a_1, \dots, a_l \to a_1 \cdot \ldots \cdot a_l}\,(\mathbf{1} \to)\ n \text{ times}}{\mathbf{1} \,/\, !B_1, !B_1, \dots, \mathbf{1} \,/\, !B_n, !B_n, a_1, \dots, a_l \to a_1 \cdot \ldots \cdot a_l}\,(/ \to)\ n \text{ times}}{!(\mathbf{1} \,/\, !B_1), !B_1, \dots, !(\mathbf{1} \,/\, !B_n), !B_n, a_1, \dots, a_l \to a_1 \cdot \ldots \cdot a_l}\,(! \to)\ n \text{ times}$$

For the induction step, consider the last step of $\Rightarrow^*$:

$$\gamma \Rightarrow^* \eta\, u_1 \dots u_k\, \theta \Rightarrow \eta\, v_1 \dots v_m\, \theta.$$

Then, since $!((u_1 \cdot \dots \cdot u_k) \,/\, (v_1 \cdot \dots \cdot v_m))$ is in Φ, we enjoy the following derivation:

$$\dfrac{\dfrac{\dfrac{\dfrac{v_1 \to v_1 \quad \dots \quad v_m \to v_m}{v_1, \dots, v_m \to v_1 \cdot \ldots \cdot v_m}\,(\to \cdot)\ (m-1) \text{ times} \qquad \dfrac{\Phi, \eta, u_1, \dots, u_k, \theta \to a_1 \cdot \ldots \cdot a_l}{\Phi, \eta, u_1 \cdot \ldots \cdot u_k, \theta \to a_1 \cdot \ldots \cdot a_l}\,(\cdot \to)\ (k-1) \text{ times}}{\Phi, \eta, (u_1 \cdot \ldots \cdot u_k) \,/\, (v_1 \cdot \ldots \cdot v_m), v_1, \dots, v_m, \theta \to a_1 \cdot \ldots \cdot a_l}\,(/ \to)}{\Phi, \eta, !((u_1 \cdot \ldots \cdot u_k) \,/\, (v_1 \cdot \ldots \cdot v_m)), v_1, \dots, v_m, \theta \to a_1 \cdot \ldots \cdot a_l}\,(! \to)}{\Phi, \eta, v_1, \dots, v_m, \theta \to a_1 \cdot \ldots \cdot a_l}\,(\mathrm{ncontr}_1)$$

The sequent $\Phi, \eta, u_1, \dots, u_k, \theta \to a_1 \cdot \ldots \cdot a_l$ is derivable by induction hypothesis.

$\boxed{2 \Rightarrow 3}$ For each formula $!(\mathbf{1} \,/\, !B_i)$ from Φ the sequent $\to !(\mathbf{1} \,/\, !B_i)$ is derivable using the weakening rule:

$$\dfrac{\dfrac{\dfrac{\to \mathbf{1}}{!B_i \to \mathbf{1}}\,(\mathrm{weak})}{\to \mathbf{1} \,/\, !B_i}\,(\to /)}{\to !(\mathbf{1} \,/\, !B_i)}\,(\to !)$$

Then we notice that $\Phi, b_1, \dots, b_k \to a_1 \cdot \ldots \cdot a_l$ is still derivable in the calculus with more structural rules, and apply (cut) to remove formulae from Φ that do not belong to Γ. This yields $\Gamma, b_1, \dots, b_k \to a_1 \cdot \ldots \cdot a_l$.

$\boxed{3 \Rightarrow 4}$ Consider the cut-free derivation of $\Gamma, b_1, \dots, b_k \to a_1 \cdot \ldots \cdot a_l$. (The system where $!$ enjoys all structural rules is a fragment of non-commutative linear logic, so we can use the cut elimination theorem from [10, Appendix 1].) Remove all formulae of the form $!E$ from the left-hand sides of the sequents in this derivation. This transformation doesn't affect rules not operating with !, they remain valid. Applications of structural rules ((ncontr$_{1,2}$, (ex$_{1,2}$), (weak)) do not alter the sequent. The only non-trivial case is $(! \to)$. Since all formulae of the form $!E$ come from Γ (due to the subformula property of the cut-free derivation), the only possible case is the following one:

$$\frac{\Delta_1, (u_1 \cdot \ldots \cdot u_k) \,/\, (v_1 \cdot \ldots \cdot v_m), \Delta_2 \to C}{\Delta_1, \Delta_2 \to C}$$

$((u_1 \cdot \ldots \cdot u_k) \,/\, (v_1 \cdot \ldots \cdot v_m)$ transforms into an invisible $!((u_1 \cdot \ldots \cdot u_k) \,/\, (v_1 \cdot \ldots \cdot v_m)))$. This application is simulated using an extra axiom from the theory $\mathcal{T}$ that we're allowed to use:

$$\dfrac{\dfrac{\dfrac{v_1, \ldots, v_m \to u_1 \cdot \ldots \cdot u_k}{v_1 \cdot \ldots \cdot v_m \to u_1 \cdot \ldots \cdot u_k}\,(\cdot \to)\ (k-1)\text{ times}}{\to (u_1 \cdot \ldots \cdot u_k) \,/\, (v_1 \cdot \ldots \cdot v_m)}\,(\to /) \qquad \Delta_1, (u_1 \cdot \ldots \cdot u_k) \,/\, (v_1 \cdot \ldots \cdot v_m), \Delta_2 \to C}{\Delta_1, \Delta_2 \to C}\,(\text{cut})$$

The sequent $v_1, \ldots, v_m \to u_1 \cdot \ldots \cdot u_k$ belongs to $\mathcal{T}$.

$\boxed{4 \Rightarrow 1}$ Derivations from $\mathcal{T}$ essentially need the cut rule. However, if one tries to apply the standard cut elimination procedure, all the cuts move directly to new axioms from $\mathcal{T}$ (this procedure is called *cut normalization*). This yields a weak form of subformula property: any formula appearing in a normalized derivation is a subformula either of $\mathcal{T}$, or of the goal sequent. Since both $\mathcal{T}$ and $b_1, \ldots, b_k \to a_1 \cdot \ldots \cdot a_l$ include only variables and the product operation, $\cdot$, rules for other connectives are never applied in the normalized derivation. For simplicity we omit parentheses and the "$\cdot$" symbols, and the rules get formulated in the following way:

$$\frac{\beta_1 \to \alpha_1 \quad \beta_2 \to \alpha_2}{\beta_1\beta_2 \to \alpha_1\alpha_2}\,(\to \cdot) \qquad \frac{\alpha \to \beta \quad \eta\alpha\theta \to \gamma}{\eta\beta\theta \to \gamma}\,(\text{cut})$$

(the $(\cdot \to)$ rule becomes trivial), and the axioms are $\alpha \to \alpha$ and rewriting rules from P with the arrows inversed.

One can easily check the following:

- if $\alpha_1 \Rightarrow^* \beta_2$ and $\alpha_2 \Rightarrow^* \beta_2$, then $\alpha_1\alpha_2 \Rightarrow^* \beta_1\beta_2$;
- if $\alpha \Rightarrow^* \beta$ and $\gamma \Rightarrow^* \eta\alpha\theta$, then $\gamma \Rightarrow^* \eta\beta\theta$.

Then, by induction on the derivation, we get $a_1 \ldots a_l \Rightarrow^* b_1 \ldots b_k$, *i.e.*, $\gamma \Rightarrow^* \delta$. □

Finally, we get rid of the unit constant, using the technique from [8]:

Lemma 4.4. *Let q be a fresh variable and let $\widetilde{\Gamma} \to \widetilde{C}$ be the sequent $\Gamma \to C$ with $\mathbf{1}$ replaced with $q \,/\, q$ and every variable p_i replaced with $(q \,/\, q) \cdot p_i \cdot (q \,/\, q)$. Then $\Gamma \to C$ is derivable if and only if $\widetilde{\Gamma} \to \widetilde{C}$ is derivable.*

Proof. The $(\mathbf{1} \to)$ rule can be interchanged with any rule applied before. Thus one can place all applications of $(\mathbf{1} \to)$ directly after axioms. All other rules, except $(\mathbf{1} \to)$, remain valid after the replacements; axioms after applications of $(\mathbf{1} \to)$ are sequents of the form $\mathbf{1}, \ldots, \mathbf{1}, p_i, \mathbf{1}, \ldots, \mathbf{1} \to p_i$ or $\mathbf{1}, \ldots, \mathbf{1} \to \mathbf{1}$. After the replacements they become derivable sequents $q \,/\, q, \ldots, q \,/\, q, (q \,/\, q) \cdot p_i \cdot (q \,/\, q), q \,/\, q, \ldots, q \,/\, q \to q \,/\, q$ and $q \,/\, q, \ldots, q \,/\, q \to q \,/\, q$. This justifies the "only if" part.

For the "if" part, we start with $\widetilde{\Gamma} \to \widetilde{C}$ and substitute $\mathbf{1}$ for q (substitution of arbitrary formulae for variables is legal in the Lambek calculus). Since $(\mathbf{1} \,/\, \mathbf{1})$ is equivalent to $\mathbf{1}$ and $(\mathbf{1} \,/\, \mathbf{1}) \cdot p_i \cdot (\mathbf{1} \,/\, \mathbf{1})$ is equivalent to p_i, the result of this substitution is equivalent to $\Gamma \to C$, whence this sequent is derivable. □

Theorem 4.2 and Lemmas 4.3 and 4.4 immediately yield Theorem 4.1.

5 The Lambek Calculus with Several Subexponentials

In this section we extend our system allowing several mutually interacting subexponentials, thus getting a non-commutative variant of linear logic with subexponentials [18]. Every subexponential is governed by its own subset of structural rules: weakening, contraction, and exchange in various combinations. Due to the issues discussed in Section 3, we consider only the non-local version of contraction.

First we fix a *subexponential signature* of the form

$$\Sigma = \langle \mathcal{I}, \preceq, \mathcal{W}, \mathcal{C}, \mathcal{E} \rangle,$$

where $I = \{s_1, \ldots, s_n\}$ is a set of subexponential labels with a preorder $\preceq$, and $\mathcal{W}$, $\mathcal{C}$, and $\mathcal{E}$ are subsets of $\mathcal{I}$. The sets $\mathcal{W}$, $\mathcal{C}$, and $\mathcal{E}$ are required to be upwardly closed with respect to $\preceq$. That is, if $s_1 \in \mathcal{W}$ and $s_1 \preceq s_2$, then $s_2 \in \mathcal{W}$ and ditto for the sets $\mathcal{E}$ and $\mathcal{C}$.

The proof system SLC_Σ is defined by adding the following rules to the Lambek calculus (Figure 1).

- For each combination of subexponential labels $s_1, \ldots, s_m, s \in I$ we add the following rules:

$$\frac{!^{\mathrm{s}_1} F_1, \ldots, !^{\mathrm{s}_\mathrm{n}} F_n \to F}{!^{\mathrm{s}_1} F_1, \ldots, !^{\mathrm{s}_\mathrm{n}} F_n \to !^{\mathrm{s}} F}\ (\to !^{\mathrm{s}}) \qquad \frac{\Gamma, F, \Delta \to C}{\Gamma, !^{\mathrm{s}} F, \Delta \to C}\ (!^{\mathrm{s}} \to)$$

  provided $s \preceq s_i$ for $1 \le i \le n$.

- $\mathcal{W} \subseteq \mathcal{I}$ is the set of labels that allow for weakening. Thus if $s \in \mathcal{W}$, then we add the rule:

$$\frac{\Gamma, \Delta \to C}{\Gamma, !^{\mathrm{s}} F, \Delta \to C}\ (\mathrm{weak})$$

- $\mathcal{C} \subseteq \mathcal{I}$ is the set of labels that allow for non-local contraction. Thus if $s \in \mathcal{C}$, then we add the rules:

$$\frac{\Gamma, !^{\mathrm{s}} F, \Delta, !^{\mathrm{s}} F, \Phi \to C}{\Gamma, !^{\mathrm{s}} F, \Delta, \Phi \to C}\ (\mathrm{ncontr}_1) \qquad \frac{\Gamma, !^{\mathrm{s}} F, \Delta, !^{\mathrm{s}} F, \Phi \to C}{\Gamma, \Delta, !^{\mathrm{s}} F, \Phi \to C}\ (\mathrm{ncontr}_2)$$

- $\mathcal{E} \subseteq \mathcal{I}$ is the set of labels that allow for the application of exchange rules. Thus if $s \in \mathcal{C}$, then we add the rules:

$$\frac{\Gamma, !^{\mathrm{s}} F, G, \Delta \to C}{\Gamma, G, !^{\mathrm{s}} F, \Delta \to C}\ (\mathrm{ex}_1) \qquad \frac{\Gamma, G, !^{\mathrm{s}} F, \Delta \to C}{\Gamma, !^{\mathrm{s}} F, G, \Delta \to C}\ (\mathrm{ex}_2)$$

In the view of Proposition 2.1 we also require the sets of labels to satisfy the following condition:

$$\mathcal{C} \cap \mathcal{W} \subseteq \mathcal{E}.$$

Due to Theorem 4.1, the system becomes undecidable, once $\mathcal{C}$ is non-empty. For the case when $\mathcal{C}$ is empty, every sequent has a derivation of polynomial size (w.r.t. the size of the sequent), therefore the derivability problem belongs to the NP class. It is NP-complete, because so is the derivability problem for the original Lambek calculus [22].

References

[1] Vincent Danos, Jean-Baptiste Joinet, and Harold Schellinx. The structure of exponentials: Uncovering the dynamics of linear logic proofs. In Georg Gottlob, Alexander Leitsch, and Daniele Mundici, editors, *Kurt Gödel Colloquium*, volume 713 of *LNCS*, pages 159–171. Springer, 1993.

[2] Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.

[3] Joshua Hodas and Dale Miller. Logic programming in a fragment of intuitionistic linear logic: Extended abstract. In G. Kahn, editor, *Sixth Annual Symposium on Logic in Computer Science*, pages 32–42, Amsterdam, July 1991.

[4] Joshua Hodas and Dale Miller. Logic programming in a fragment of intuitionistic linear logic. *Information and Computation*, 110(2):327–365, 1994.

[5] Makoto Kanazawa. Lambek calculus: Recoginzing power and complexity. In J. Gerbrandy et al., editors, *JFAK. Essays dedicated to Johan van Benthem on the occasion of his 50th birthday.* Vossiuspers, Amsterdam Univ. Press, 1999.

[6] Max Kanovich, Stepan Kuznetsov, Andre Scedrov. Undecidability of the Lambek calculus with a relevant modality. *Proc. Formal Grammar 2015 and 2016* (LNCS vol. 9804), pp. 240–256. Springer, 2016.

[7] Max Kanovich, Stepan Kuznetsov, Andre Scedrov. Undecidability of the Lambek calculus with subexponentials and bracket modalities. arXiv preprint 1608.04020, 2016.

[8] Stepan Kuznetsov. Lambek grammars with the unit. *Formal Grammar 2010 and 2011* (LNCS vol. 7395), pp. 262–266. Springer, 2012.

[9] Jim Lambek. On the calculus of syntactic type. *Structure of Language and Its Mathematical Aspects* (Proc. Symposia Appl. Math., vol. 12), pp. 166–178. AMS, 1961.

[10] Patrick Lincoln, John Mitchell, Andre Scedrov, Natarajan Shankar. Decision problems for propositional linear logic. *APAL*, 56:239–311, 1992.

[11] Andrei Markov. On the impossibility of certain algorithms in the theory of associative systems. *Dokl. Akad. Nauk SSSR (N.S.)* 55:583-586, 1947.

[12] Dale Miller. A multiple-conclusion meta-logic. In S. Abramsky, editor, *Ninth Annual Symposium on Logic in Computer Science*, pages 272–281, Paris, July 1994. IEEE Computer Society Press.

[13] Dale Miller. Forum: A multiple-conclusion specification logic. *Theoretical Computer Science*, 165(1):201–232, September 1996.

[14] Glyn Morrill, Oriol Valentín. Computational coverage of TLG: Nonlinearity. *Proc. NLCS '15* (EPiC Series, vol. 32), pp. 51–63, 2015.

[15] Vivek Nigam. *Exploiting non-canonicity in the sequent calculus.* PhD thesis, Ecole Polytechnique, September 2009.

[16] Vivek Nigam. On the complexity of linear authorization logics. In *LICS*, pages 511–520. IEEE, 2012.

[17] Vivek Nigam. A framework for linear authorization logics. *Theoretical Computer Science*, 536(0):21 – 41, 2014.

[18] Vivek Nigam and Dale Miller. Algorithmic specifications in linear logic with subexponentials. In *ACM SIGPLAN Conference on Principles and Practice of Declarative Programming (PPDP)*, pages 129–140, 2009.

[19] Vivek Nigam, Carlos Olarte, and Elaine Pimentel. A general proof system for modalities in concurrent constraint programming. In Pedro R. D'Argenio and Hernán C. Melgratti, editors, *CONCUR*, volume 8052 of *LNCS*, pages 410–424. Springer, 2013.

[20] Vivek Nigam, Elaine Pimentel, and Giselle Reis. An extended framework for specifying and reasoning about proof systems. *J. Log. Comput.*, 26(2):539–576, 2016. Special issue in honor of Roy Dyckhoff.

[21] Carlos Olarte, Elaine Pimentel, and Vivek Nigam. Subexponential concurrent constraint program-

ming. *Theor. Comput. Sci.*, 606:98–120, 2015.

[22] Mati Pentus. Lambek calculus is NP-complete. *Theor. Comput. Sci.*, 357:186–201, 2006.

[23] Emil Leon Post. Recursive unsolvability of a problem of Thue. *J. Symb. Log.*, 12:1–11, 1947.