Undecidability of a Newly Proposed Calculus for CatLog3*

Max Kanovich, Stepan Kuznetsov, and Andre Scedrov

[1] University College London (London, UK)
m.kanovich@ucl.ac.uk
[2] Steklov Mathematical Institute of the RAS (Moscow, Russia)
sk@mi-ras.ru
[3] University of Pennsylvania (Philadelphia, USA)
scedrov@math.upenn.edu
[4] National Research University Higher School of Economics (Moscow, Russia)
{mkanovich,slkuznetsov,ascedrov}@hse.ru

Abstract. In his recent papers "Parsing/theorem-proving for logical grammar CatLog3" and "A note on movement in logical grammar," Glyn Morrill proposes a new substructural calculus to be used as the basis for the categorial grammar parser CatLog3. In this paper we prove that the derivability problem for a fragment of this calculus is algorithmically undecidable.

Keywords: categorial grammar · contraction rule · undecidability.

1 Introduction

In his recent paper [32, 31], Glyn Morrill proposes a new substructural calculus, to be used as the basis for the categorial grammar parser CatLog3. As the first step on the road of investigating algorithmic properties of the new Morrill's system, in this paper we shall prove that the derivability problem for a fragment of this calculus is algorithmically undecidable.

The source of undecidability is the contraction rule. In Morrill's systems, however, contraction appears in a very non-standard form. Moreover, the contraction rule presented in Morrill's new papers significantly differs from other ones, therefore, earlier undecidability proofs [23, 12, 14] do not work for this new version of contraction rule. Thus, a new technique should be invented, and we do that in the present paper.

The idea of categorial grammars goes back to Ajdukiewicz [2] and Bar-Hillel [3]. The version of categorial grammars used by Morrill is an extension of Lambek categorial grammars [22]. In a categorial grammar, each word (lexeme) of the language is given one or several syntactic categories (types), which

* Presented at Formal Grammar 2019 and published in its lecture notes (Springer LNCS vol. 11668). The final authenticated version is available online at https://doi.org/10.1007/978-3-662-59648-7_5

are formulae of a specific logical system, an extension of the Lambek calculus. Parsing with categorial grammars, that is, checking whether a sentence is considered correct according to the grammar, reduces to checking derivability in the logical system involved. Namely, a sequence of words $a_1 \dots a_n$ is accepted by the grammar if and only if there exist such formulae $A_1, \dots, A_n$ that, for each i, A_i is one of the syntactic categories for a_i, and the *sequent* $A_1, \dots, A_n \Rightarrow S$ is derivable. Here S is a designated syntactic category for grammatically correct sentences.

The Lambek calculus, which is used as the basis for categorial grammars, is a substructural logic and is closely related to Girard's linear logic [7, 1]. In linear logic, formulae are treated as *resources,* thus, each formula should be used exactly once. This motivates the absence of the structural rules of contraction and weakening. Moreover, the Lambek calculus is also non-commutative, *i.e.,* does not include the rule of permutation (word order matters).

Sometimes, however, structural rules are allowed to be restored, in a restricted and controlled way, in order to treat subtle syntactic phenomena. One of such phenomena is *parasitic extraction,* which happens in phrases like *"the paper that John signed without reading."* Here the dependent clause has two *gaps,* which we denote by []: *"John signed* [] *without reading* []," which should both be filled by the same *"the paper"* in order to obtain a complete sentence. In the logic, this is handled by the contraction rule in its non-local form:

$$\frac{\Gamma_1, !A, \Gamma_2, !A, \Gamma_3 \Rightarrow C}{\Gamma_1, !A, \Gamma_2, \Gamma_3 \Rightarrow C}$$

Here ! is the (sub)exponential modality, and the contraction rule is allowed to be applied to formulae of the form $!A$, and only to them.

Extension of the Lambek calculus with a subexponential modality which allows the non-local contraction rule formulated as presented above are undecidable [16]. In Morrill's systems, however, the contraction rule is presented in a rather non-standard form. The reason is in the usage of brackets which introduce controlled non-associativity. Brackets prevent the calculus from overgeneration, that is, from justifying grammatically incorrect sentences as correct ones. The contraction rule, as shown below, also essentially interacts with brackets. This makes standard undecidability proofs unapplicable to Morrill's systems, so new undecidability proofs are needed.

In order to make our examples more formal, we assign syntactic type N to noun phrases, like *"John"* or *"the paper,"* and S to grammatically correct sentences. Our dependent clause *"John signed without reading"* receives type $S \,/\, !N$, meaning a syntactic object which lacks a noun phrase in order to become a complete sentence (*"John signed the paper without reading the paper"*). The *subexponential modality* ! applied to N means that our noun phrase should be commutative (in order to find its place inside the sentence) and allow contraction (in order to fill both gaps).

Overgeneration is exhibited by the following example: * *"the paper that John signed and Pete ate a pie"* (the asterisk marks the sentence as ungrammatical).

On one hand, this phrase is clearly ungrammatical, because of the irrelevant fragment *"Pete ate a pie"* in the dependent clause. On the other hand, being a sentence with one gap, *"John signed [] and Pete ate a pie"* receives the same type $S\,/\,!N$, which makes it equivalent to correct dependent clauses like *"John signed [] yesterday."* In order to address this issue, Morrill [26] and Moortgat [25] introduce brackets which embrace so-called islands, which, within our setting, cannot be penetrated by $!N$. In particular, *and*-coordination of sentences makes the result a bracketed island. Since phrases like * *"the paper that John left the office without reading"* are also ungrammatical, a *without*-clause also forms an island and should be embraced in brackets. This leads to Morrill's idea of handling parasitic extraction [27, Sect. 5.5]: in the dependent clause, there is one *principal,* or *host* gap, which should not be inside an island, and *parasitic* gaps, which reside in islands. Moreover, a parasitic gap can also be a host for its own "second-order" parasitic gaps.

Thus, the contraction rule should take one $!A$ from a bracket-embraced island and remove it, in the presence of another $!A$ outside the island. However, after that the bracketing should be somehow changed, in order to avoid another usage of the same island for parasitic gapping. This general idea, however, has different realisations in a number of works of Morrill and his co-authors [27–29, 31, 32]. In the next section we show the most recent approach [32, 31], which essentially resembles the original construction from Morrill's 2011 book [27].

2 The Calculus

Morrill's calculus for CatLog3 [32] is quite involved, including up to 45 connectives. The metasyntax of sequents in this calculus is also rather non-standard, involving brackets and meta-operations for discontinuity. In this paper we consider its simpler fragment, involving only the multiplicative Lambek operations: left and right divisions ($\backslash$, $/$), multiplication ($\bullet$), and the unit ($\mathbf{I}$), brackets and bracket modalities ($\langle\rangle$, $[]^{-1}$), and the subexponential modality, and already for this fragment we show undecidability.

Notice that Morrill's system also includes Kleene star, axiomatised by means of an ω-rule. In Morrill's system, it is called "existential exponential" and denoted by "?". In the presence of the Kleene star the Lambek calculus is known to be at least Π_1^0-hard [5, 20] and thus undecidable. Moreover, in the view of Kozen's [18] results on complexity of Horn theories of Kleene algebras, the complexity of the system including *both* Kleene star and the subexponential could potentially rise even higher, up to Π_1^1-completeness. Morrill, however, emphasizes the fact that in formulae used in categorial grammars designed for real languages the Kleene star never occurs with positive polarity. Thus, the ω-rule is never used, and the Kleene star does not incur problems with decidability. Thus, the only possible source of undecidability is the specific contraction rule for the subexponential. We consider a fragment of Morrill's system with this rule, which is sufficient to show undecidability.

Let us define the syntax of our fragment. Formulae will be built from variables (primitive types) $p, q, \ldots$ and the multiplicative unit constant $\mathbf{I}$ using three binary operations: $\backslash$ (left division), $/$ (right division), $\bullet$ (product), and three unary operation: $\langle\rangle$ and $[]^{-1}$ (bracket modalities) and $!$ (subexponential). Sequents (in Morrill's terminology, *h-sequents)* of $\mathbf{!^s_{b1/2}L^*b}$ are expressions of the form $\Xi \Rightarrow A$, where A is a formula and Ξ is a complex metasyntactic structure which we call *meta-formula* (Morrill calls them *zones).* Meta-formulae are built from formulae using comma and brackets; also formulae which are intended to be marked by the subexponential !, which allows permutation, are placed into special commutative areas called *stoups* (cf. [8, 9, 17]). Following Morrill [32], we define the notion of meta-formula along with two auxiliary notions, stoup and *tree term*, simultaneously.

- A stoup is a multiset of formulae: $\zeta = \{A_1, \ldots, A_n\}$. A stoup could be empty, the empty stoup is denoted by $\varnothing$.
- A tree term is either a formula or a bracketed expression of the form $[\Xi]$, where Ξ is a meta-formula.
- A meta-formula is an expression of the form $\zeta; \Gamma$, where ζ is a stoup and Γ is a linearly ordered sequence of tree terms. Here Γ could also be empty; the empty sequence is denoted by Λ.

We use comma both for concatenation of tree term sequences and for multiset union of stoups (Morrill uses $\uplus$ for the latter). Moreover, for adding one formula into a stoup we write ζ, A instead of $\zeta, \{A\}$.

Axioms and rules of $\mathbf{!^s_{b1/2}L^*b}$ are as follows.

$$\frac{}{\varnothing; A \Rightarrow A}\ id$$

$$\frac{\zeta_1; \Gamma \Rightarrow B \quad \Xi(\zeta_2; \Delta_1, C, \Delta_2) \Rightarrow D}{\Xi(\zeta_1, \zeta_2; \Delta_1, C \,/\, B, \Gamma, \Delta_2) \Rightarrow D}\ /L \qquad \frac{\zeta; \Gamma, B \Rightarrow C}{\zeta; \Gamma \Rightarrow C \,/\, B}\ /R$$

$$\frac{\zeta_1; \Gamma \Rightarrow A \quad \Xi(\zeta_2; \Delta_1, C, \Delta_2) \Rightarrow D}{\Xi(\zeta_1, \zeta_2; \Delta_1, \Gamma, A \setminus C, \Delta_2) \Rightarrow D}\ \backslash L \qquad \frac{\zeta; A, \Gamma \Rightarrow C}{\zeta; \Gamma \Rightarrow A \setminus C}\ \backslash R$$

$$\frac{\Xi(\zeta; \Delta_1, A, B, \Delta_2) \Rightarrow D}{\Xi(\zeta; \Delta_1, A \bullet B, \Delta_2) \Rightarrow D}\ \bullet L \qquad \frac{\zeta_1; \Delta \Rightarrow A \quad \zeta_2; \Gamma \Rightarrow B}{\zeta_1, \zeta_2; \Delta, \Gamma \Rightarrow A \bullet B}\ \bullet R$$

$$\frac{\Xi(\zeta; \Delta_1, \Delta_2) \Rightarrow A}{\Xi(\zeta; \Delta_1, \mathbf{I}, \Delta_2) \Rightarrow A}\ \mathbf{I}L \qquad \frac{}{\varnothing; \Lambda \Rightarrow \mathbf{I}}\ \mathbf{I}R$$

$$\frac{\Xi(\zeta; \Delta_1, A, \Delta_2) \Rightarrow B}{\Xi(\zeta; \Delta_1, [\varnothing; []^{-1}A], \Delta_2) \Rightarrow B}\ []^{-1}L \qquad \frac{\varnothing; [\Xi] \Rightarrow A}{\Xi \Rightarrow []^{-1}A}\ []^{-1}R$$

$$\frac{\Xi(\zeta; \Delta_1, [\varnothing; A], \Delta_2) \Rightarrow B}{\Xi(\zeta; \Delta_1, \langle\rangle A, \Delta_2) \Rightarrow B}\ \langle\rangle L \qquad \frac{\Xi \Rightarrow A}{\varnothing; [\Xi] \Rightarrow \langle\rangle A}\ \langle\rangle R$$

$$\frac{\Xi(\zeta, A; \Gamma_1, \Gamma_2) \Rightarrow B}{\Xi(\zeta; \Gamma_1, !A, \Gamma_2) \Rightarrow B}\ !L \qquad \frac{\varnothing; !A \Rightarrow B}{\varnothing; !A \Rightarrow !B}\ !R$$

$$\frac{\Xi(\zeta; \Gamma_1, A, \Gamma_2) \Rightarrow B}{\Xi(\zeta, A; \Gamma_1, \Gamma_2) \Rightarrow B}\ !P \qquad \frac{\Xi(\zeta, A; \Gamma_1, [A; \Gamma_2], \Gamma_3) \Rightarrow B}{\Xi(\zeta, A; \Gamma_1, [\varnothing; [\varnothing; \Gamma_2]], \Gamma_3) \Rightarrow B}\ !C$$

Morrill [32, 31] does not give any particular name to its calculus. In this paper, we denote our fragment by $\mathbf{!^s_{b1/2}L^*b}$. Here "**b**" stands for "bracketed," and the decorations of ! mean the following. The superscript "**s**" means that the right rule for ! is in the style of soft and light linear logic [10, 21, 15], allowing, in particular, only one $!A$ in the left-hand side. The subscript "**b1/2**" means that contraction operates brackets, using single bracketing in the premise and double bracketing in the conclusion.

In his older paper [29], Morrill uses another form of contraction rule, which in our notation looks like

$$\frac{\Xi(\zeta, A; \Gamma_1, [A; \Gamma_2], \Gamma_3) \Rightarrow B}{\Xi(\zeta, A; \Gamma_1, \Gamma_2, \Gamma_3) \Rightarrow B}$$

Thus, this system could be called $\mathbf{!^s_{b1/0}L^*b}$, in our notations. For the system with this sort of contraction, undecidability was established in [14]. The new contraction rule of Morrill [32, 31], however, significantly differs from the old contraction rule, and the undecidability proof from [14] does not work for Morrill's new system. Thus, undecidability becomes a separate issue and we address it in this paper.

For convenience, we use the following derivable *dereliction* rule

$$\frac{\Xi(\zeta; \Gamma_1, A, \Gamma_2) \Rightarrow B}{\Xi(\zeta; \Gamma_1, !A, \Gamma_2) \Rightarrow B}\ !D$$

which is actually consecutive application of $!P$ and $!L$:

$$\cfrac{\cfrac{\Xi(\zeta; \Gamma_1, A, \Gamma_2) \Rightarrow B}{\Xi(\zeta, A; \Gamma_1, \Gamma_2) \Rightarrow B}\ !P}{\Xi(\zeta; \Gamma_1, !A, \Gamma_2) \Rightarrow B}\ !L$$

Notice that in Morrill's calculus [32, 31] there is no cut rule. Thus, the question of cut-elimination is transformed into the question of admissibility of cut, proving which is marked in [32] as an ongoing work by O. Valentín. Since the calculus considered in [32, 31] does not include cut, our fragment, which uses only a restricted set of connectives and consists of the corresponding inference rules, is a conservative fragment of the complete system [32]. Namely, for sequents in the restricted language, derivability in the fragment is equivalent to derivability in the big system. Therefore, undecidability for $\mathbf{!^s_{b1/2}L^*b}$ (Theorem 3 below) yields undecidability for the whole system also.

Using $\mathbf{!^{s}_{b1/2}L^{*}b}$, one can analyze our example *"the paper that John signed without reading"* in the following way, simplyfing Morrill's analysis [32]. Assign the following syntactic types to words:

$$\begin{array}{rlrl}
\textit{the} \rhd & N \,/\, CN & \textit{likes, signed} \rhd & (\langle\rangle N \setminus S) \,/\, N \\
\textit{man, paper} \rhd & CN & \textit{without} \rhd & ([]^{-1}((\langle\rangle N \setminus S) \setminus (\langle\rangle N \setminus S))) \,/\, (\langle\rangle N \setminus S) \\
\textit{reading} \rhd & (\langle\rangle N \setminus S) \,/\, N & \textit{who, that} \rhd & ([]^{-1}[]^{-1}(CN \setminus CN)) \,/\, (S \,/\, !N) \\
\textit{John} \rhd & \langle\rangle N & &
\end{array}$$

Here N stands for "noun phrase," CN states for "common noun" (without an article), and S stands for "sentence."

In order to parse this sentence in this grammar, one first needs to impose the bracketing structure on it. This is done in the following way:

$$\textit{the paper } [[\textit{that } [\textit{John}] \textit{ signed } [[\textit{without reading}]]\ ]].$$

Indeed, in Morrill's CatLog categorial grammar the subject group and the *without*-clause form islands, and the *that*-clause forms a strong island, embraced by double brackets. Moreover, we also have to double-bracket our *without*-clause (make it a "strong island"), since it will be used for parasitic extraction. Each pair of brackets has its own stoup, which is originally empty. Unfortunately, in CatLog the bracketed structure is required as an input from the user (while it is of course not part of the original sentence). Morrill et al. [30], however, provide an algorithm for automated induction (guessing) of the bracketed structure, for a small fragment of the CatLog grammar (in particular, without subexponential).

With the bracketing shown above, the corresponding sequent is derived in $\mathbf{!^{s}_{b1/2}L^{*}b}$ as shown in Fig. 1.

At the request of one of the referees, we discuss the following example, which is used by Morrill [31] to motivate the changes made in the contraction rule from $\mathbf{b1/0}$ to $\mathbf{b1/2}$ (see above). This example features an incorrect noun phrase, * *"the man who likes,"* analysed with two gaps in the dependent clause: * *"the man who* [] *likes* []*."* (Asterisks denote ungrammaticality.) The intended semantics (and the correct version of the phrase) here is *"the man who likes himself."* In $\mathbf{!^{s}_{b1/0}L^{*}b}$, however, * *"the man who* [] *likes* []*,"* with brackets imposed as *"the man* [[*who likes*]]*,"* is parsed as follows. First one derives the sequent $\varnothing; (\langle\rangle N \setminus S) \,/\, N \Rightarrow S \,/\, !N$, which (ungrammatically) treats *"likes"* as a dependent clause with two

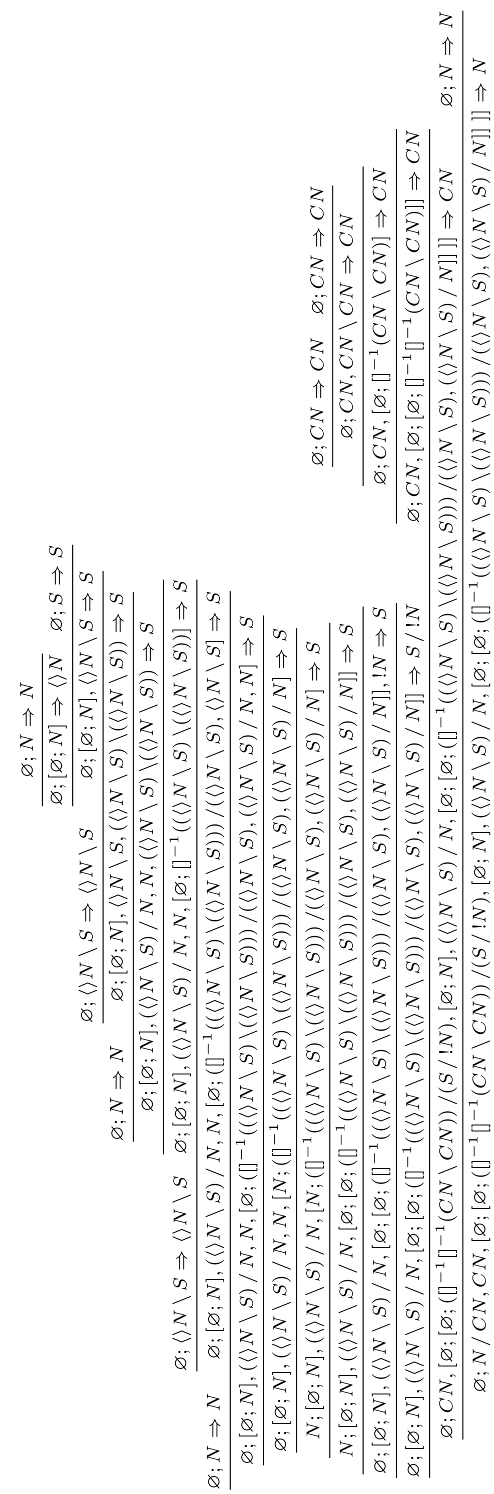

Fig. 1. Derivation for *"the paper that John signed without reading"* (cf. [32, Fig. 24])

gaps, a host one for the object and a parasitic one for the subject:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\varnothing; N \Rightarrow N \qquad \dfrac{\dfrac{\varnothing; N \Rightarrow N}{[\varnothing; N] \Rightarrow \langle\rangle N} \quad \varnothing; S \Rightarrow S}{\varnothing; [\varnothing; N], \langle\rangle N \setminus S \Rightarrow S}}{\varnothing; [\varnothing; N], (\langle\rangle N \setminus S) \,/\, N, N \Rightarrow S}}{\varnothing; [N; \Lambda], (\langle\rangle N \setminus S) \,/\, N, N \Rightarrow S}}{N; [N; \Lambda], (\langle\rangle N \setminus S) \,/\, N \Rightarrow S}}{N; (\langle\rangle N \setminus S) \,/\, N \Rightarrow S}}{\varnothing; (\langle\rangle N \setminus S) \,/\, N, !N \Rightarrow S}}{\varnothing; (\langle\rangle N \setminus S) \,/\, N \Rightarrow S \,/\, !N}$$

Here the whole subject island is introduced by $!C$ (in its **b1/0** version, with $\Gamma_2 = \Lambda$) as a parasitic extraction site. Next, one finishes the derivation as it is done in Fig. 1 and obtains

$$\varnothing; N \,/\, CN, CN, [\varnothing; [\varnothing; ([]^{-1}[]^{-1}(CN \setminus CN) \,/ (S \,/\, !N), (\langle\rangle N \setminus S) \,/\, N]] \Rightarrow N.$$

With the new, **b1/2** contraction rule, this derivation of * "*the man* [[*who likes*]]" becomes impossible. However, there still exists a way to derive * "*the man who likes,*" if the user imposes the following weird bracketing: "*the man* [[*who* [[ ]] *likes*]]." This bracketing explicitly creates an empty strong, double-bracketed island as the subject of the dependent clause, and the $!C$ rule transforms it into a single-bracketed one. (In other parts, the derivation is similar to the one presented above.) In one of the reviews, the referee asks whether one can consider a system where empty brackets are explicitly disallowed, and whether our undecidability proof is still valid for this system. This constraint, however, is tightly connected with the Lambek's antecedent non-emptiness restriction. It appears that reconciling this constraint with (sub)exponential modalities raises certain issues with keeping good proof-theoretic properties of the system, such as cut elimination and substitution [11, 13]. We accurately formulate these questions in the "Future Work" section and leave them as open problems for future research.

3 The Bracket-Free System and the π Projection

In this section we define $\mathbf{!L^*}$, a system without brackets and with a full-power exponential modality. This is a more well-known system, and it is simpler from the logical point of view. We shall need $\mathbf{!L^*}$ inside our undecidability proof in Section 4. In this section we define a projection that maps derivability in $\mathbf{!^s_{b1/2}L^*b}$ to derivability in $\mathbf{!L^*}$. This projection is similar to the bracket-forgetting projection in [14].

Formulae of $\mathbf{!L^*}$ are defined similary to the ones of $\mathbf{!^s_{b1/2}L^*b}$, but without bracket modalities ($\langle\rangle$ and $[]^{-1}$). Sequents of $\mathbf{!L^*}$ have a simpler structure, and are expressions of the form $\Gamma \Rightarrow A$, where A is a formula and Γ is a linearly ordered sequence of formulae. Axioms and inference rules of $\mathbf{!L^*}$ are as follows.

$$\overline{A \Rightarrow A}\ id$$

$$\frac{\Gamma \Rightarrow B \quad \Delta_1, C, \Delta_2 \Rightarrow D}{\Delta_1, C \,/\, B, \Gamma, \Delta_2 \Rightarrow D}\ /L \qquad \frac{\Gamma, B \Rightarrow C}{\Gamma \Rightarrow C \setminus B}\ /R$$

$$\frac{\Gamma \Rightarrow A \quad \Delta_1, C, \Delta_2 \Rightarrow D}{\Delta_1, \Gamma, A \setminus B, \Delta_2 \Rightarrow D}\ \setminus L \qquad \frac{A, \Gamma \Rightarrow C}{\Gamma \Rightarrow A \setminus C}\ /R$$

$$\frac{\Delta_1, A, B, \Delta_2 \Rightarrow D}{\Delta_1, A \bullet B, \Delta_2 \Rightarrow D}\ \bullet L \qquad \frac{\Delta \Rightarrow A \quad \Gamma \Rightarrow B}{\Delta, \Gamma \Rightarrow A \bullet B}\ \bullet R$$

$$\frac{\Delta_1, \Delta_2 \Rightarrow A}{\Delta_1, \mathbf{I}, \Delta_2 \Rightarrow A}\ \mathbf{I}L \qquad \frac{}{\Lambda \Rightarrow \mathbf{I}}\ \mathbf{I}R$$

$$\frac{\Gamma_1, A, \Gamma_2 \Rightarrow B}{\Gamma_1, !A, \Gamma_2 \Rightarrow B}\ !L \qquad \frac{!A_1, \ldots, !A_n \Rightarrow B}{!A_1, \ldots, !A_n \Rightarrow !B}\ !R \qquad \frac{\Gamma_1, \Gamma_2 \Rightarrow B}{\Gamma_1, !A, \Gamma_2 \Rightarrow B}\ !W$$

$$\frac{\Gamma_1, !A, \Gamma_2, !A, \Gamma_3 \Rightarrow B}{\Gamma_1, !A, \Gamma_2, \Gamma_3 \Rightarrow B}\ !C_1 \qquad \frac{\Gamma_1, !A, \Gamma_2, !A, \Gamma_3 \Rightarrow B}{\Gamma_1, \Gamma_2, !A, \Gamma_3 \Rightarrow B}\ !C_2$$

$$\frac{\Gamma \Rightarrow A \quad \Delta_1, A, \Delta_2 \Rightarrow D}{\Delta_1, \Gamma, \Delta_2 \Rightarrow D}\ \mathit{cut}$$

Notice that, unlike $\mathbf{!^s_{b1/2}L^*b}$, here cut is included as an official rule of the system. However, here the cut rule is eliminable by a standard technique by using the mix rule.

Proposition 1 *Any sequent derivable in* $\mathbf{!L^*}$ *is derivable without using the cut rule.*

This proof of cut elimination is explained, for example, in [16], where $\mathbf{!L^*}$ acts as a specific case of $\mathbf{SMALC}_\Sigma$, an extension of the multiplicative-additive Lambek calculus with a family of subexponential modalities.

In our version of $\mathbf{!L^*}$, contraction rules ($!C_1$ and $!C_2$) are non-local (cf. [16]), and permutation rules of the following form

$$\frac{\Gamma_1, \Gamma_2, !A, \Gamma_3 \Rightarrow B}{\Gamma_1, !A, \Gamma_2, \Gamma_3 \Rightarrow B}\ !P_1 \qquad \frac{\Gamma_1, !A, \Gamma_2, \Gamma_3 \Rightarrow B}{\Gamma_1, \Gamma_2, !A, \Gamma_3 \Rightarrow B}\ !P_2$$

are derivable using non-local contraction and weakening:

$$\cfrac{\cfrac{\Gamma_1, \Gamma_2, !A, \Gamma_3 \Rightarrow B}{\Gamma_1, !A, \Gamma_2, !A, \Gamma_3 \Rightarrow B}\ !W}{\Gamma_1, !A, \Gamma_2, \Gamma_3 \Rightarrow B}\ !C_1 \qquad \cfrac{\cfrac{\Gamma_1, !A, \Gamma_2, \Gamma_3 \Rightarrow B}{\Gamma_1, !A, \Gamma_2, !A, \Gamma_3 \Rightarrow B}\ !W}{\Gamma_1, \Gamma_2, !A, \Gamma_3 \Rightarrow B}\ !C_2$$

Next, we define a translation from $\mathbf{!^s_{b1/2}L^*b}$ to $\mathbf{!L^*}$, which is actually a forgetting projection, erasing all brackets and bracket modalities, and also translating stoups into plain sequences of !-formulae. We denote this projection by π and define it in the following recursive way.

– For a formula A, its projection $\pi(A)$ is defined as follows:

$$
\begin{array}{ll}
\pi(p) = p \text{ for any variable } p; & \pi(\mathbf{I}) = \mathbf{I}; \\
\pi(A \setminus B) = \pi(A) \setminus \pi(B); & \pi(B \mathbin{/} A) = \pi(B) \mathbin{/} \pi(A); \\
\pi(A \bullet B) = \pi(A) \bullet \pi(B); & \pi(!A) = !\pi(A); \\
\pi(\langle\rangle A) = \pi([]^{-1} A) = \pi(A). &
\end{array}
$$

– For a stoup $\zeta = \{A_1, \dots, A_n\}$, its π-projection is the sequence of formulae $!\pi(A_1), \dots, !\pi(A_n)$. Since in $\mathbf{!L^*}$ we have permutation rules for !-formulae, the order does not matter. The π-projection of an empty stoup is the empty sequence Λ.
– For a tree term there are two cases. If it is a formula, A, then its π-projection is $\pi(A)$. If the tree term is of the form $[\Xi]$, where Ξ is a meta-formula then its π-projection is $\pi(\Xi)$ (as defined below).
– For a meta-formula of the form $\zeta; \Upsilon_1, \dots, \Upsilon_k$, where Υ_i are tree terms, its π-projection is $\pi(\zeta), \pi(\Upsilon_1), \dots, \pi(\Upsilon_k)$.

Proposition 2 *If $\Xi \Rightarrow A$ is derivable in $\mathbf{!^s_{b1/2}L^*b}$, then $\pi(\Xi) \Rightarrow \pi(A)$ is derivable in $\mathbf{!L^*}$.*

Proof. Proceed by induction on derivation; recall that it is cut-free by definition.

Axioms id and $\mathbf{I}R$ and rules $/\, R$, $\bullet\, L$, and $\mathbf{I}L$ of $\mathbf{!^s_{b1/2}L^*b}$, translate exactly to the corresponding rules of $\mathbf{!L^*}$. The rules for bracket modalities ($\langle\rangle L$, $\langle\rangle R$, $[]^{-1} L$, $[]^{-1} R$) become trivial: after applying the π-projection, the conclusion of such a rule coincides with its premise. For the rules $\setminus R$, $\bullet\, R$, $/\, L$, and $\setminus L$ are translated to the corresponding rules in $\mathbf{!L^*}$, together with necessary permutations ($!P_{1,2}$) for !-formulae coming from the stoups. Finally, the !-rules of $\mathbf{!^s_{b1/2}L^*b}$ translate to the corresponding rules of $\mathbf{!L^*}$: $!L$ becomes $!P_2$, $!R$ maps to $!R$, $!P$ becomes $!L$ together with $!P_1$, and $!C$ maps to $!C_1$.

Notice that the reverse implication does not hold, which can be shown by analysis of our examples for brackets, like * "*the paper that John signed and Pete ate a pie.*"

4 Undecidability Proof

In this section we prove undecidability of the derivability problem in $\mathbf{!^s_{b1/2}L^*b}$.

Theorem 3. *The derivability problem in $\mathbf{!^s_{b1/2}L^*b}$ is undecidable, more precisely, Σ^0_1-complete.*

The general outline of our proof is rather standard, following the ideas of Lincoln et al. [23]: encoding of semi-Thue systems in $\mathbf{!^s_{b1/2}L^*b}$. Maintaining the correct bracket structure, however, makes the encoding more involved and requires some technical tricks.

A *semi-Thue system* [34] over an alphabet Σ is a set of pairs of words over Σ, called *rewriting rules* and written as $x_1 \dots x_m \to y_1 \dots y_k$ ($k, m \geq 0$, $x_i, y_i \in \Sigma$). A rewriting sequence in a semi-Thue system $\mathcal{S}$ is a sequence of words $w_1, w_2, \dots, w_N$, in which each word w_ℓ, starting from the second one, is obtained from the previous word $w_{\ell-1}$ by applying a rewriting rule as follows:

$$w_{\ell-1} = u x_1 \dots x_m v \to u y_1 \dots y_k v = w_\ell,$$

where $x_1 \dots x_m \to y_1 \dots y_k$ is a rewriting rule of $\mathcal{S}$ and u, v are arbitrary words. If there exists a rewriting sequence $w_1 \to w_2 \to \dots \to w_N$ in $\mathcal{S}$, we say that w_N is derivable from w_1 in $\mathcal{S}$.

A famous result by Markov [24] and Post [33] shows that the derivability problem for semi-Thue systems is undecidable; more precisely, it is Σ^0_1-complete (that is, the membership problem for any recursively enumerable language can be reduced to the derivability problem in semi-Thue systems). Moreover, the problem of derivability of a word w from a one-letter word s, like in Chomsky's [6] type-0 grammars, is also undecidable. This can be shown by the following reduction: for arbitrary words w_1 and w_N, checking derivability of w_N from w in a semi-Thue system $\mathcal{S}$ is equivalent to checking derivability of w_N from the one-letter word s in the semi-Thue system $\mathcal{S}$ extended by a new symbol s and a new rewriting rule $s \to w_1$.

Let us proceed with our encoding of semi-Thue systems in $\mathbf{!^s_{b1/2}L^*b}$. Let

$$\mathcal{A}_\mathcal{S} = \{(x_1 \bullet \dots \bullet x_m) \,/\, (y_1 \bullet \dots \bullet y_k) \mid \\ x_1 \dots x_m \to y_1 \dots y_k \text{ is a rewriting rule of } \mathcal{S}\}.$$

If the word $x_1 \dots x_m$ is empty, then $x_1 \bullet \dots \bullet x_m$ is replaced by $\mathbf{I}$; the same for $y_1 \dots y_k$.

Let $\mathcal{A}_\mathcal{S} = \{A_1, \dots, A_n\}$ (the order does not matter). For each A_i let

$$Z_i = []^{-1}(!A_i \bullet \langle\rangle\langle\rangle\mathbf{I})$$

and define the following two sets of formulae (further they will be considered as multisets and used in the stoup):

$$\mathcal{Z}_\mathcal{S} = \{!Z_1, \dots, !Z_n\};$$
$$\mathcal{X}_\mathcal{S} = \{\mathbf{I} \,/\, !Z_1, \dots, \mathbf{I} \,/\, !Z_n, \mathbf{I} \,/\, (\langle\rangle\langle\rangle\mathbf{I})\}.$$

Finally, consider the following linearly ordered sequence of formulae:

$$\Gamma_\mathcal{S} = !A_1, \dots, !A_n.$$

The intuition behind Z_i is as follows and is best understood when reading simultaneously with the formal proof of the $1 \Rightarrow 2$ implication in Theorem 4 below. In the sequent, we keep a special empty tree-term with double bracketing, $[\varnothing; [\varnothing; \Lambda]]$, which is used as the "landing zone" for Z_i. Double brackets, with empty stoups, allow the usage of Morrill's contraction rule, $!C$. Applying this rule (we trace the derivation tree from bottom to top) destroys one pair of brackets and puts $!Z_i$, taken from the stoup, inside. Dereliction removes the !, and the bracket modality inside Z_i destroys the second pair of brackets around it. Now we have $!A_i$ and $\langle\rangle\langle\rangle\mathbf{I}$. The former, by $!L$ and $!P$, is put to an arbitrary place of the antecedent, allowing application of a rewriting rule of the semi-Thue system $\mathcal{S}$. The latter restores the landing zone which was destroyed by contraction, and leaves a configuration which is ready for the next reduction step. Finally, formulae from $\mathcal{X}_{\mathcal{S}}$ are used for garbage collection on the top of the derivation.

This gives a translation of semi-Thue derivations to $\mathbf{!^s_{b1/2}L^*b}$ ones. The backwards translation, from $\mathbf{!^s_{b1/2}L^*b}$ derivations back to $\mathcal{S}$, is performed via the π-projection. This projection trivialises everything connected to brackets, and the resulting sequent, derivable in $\mathbf{!L^*}$ by Proposition 2, happens to be $\mathbf{!L^*}$-equivalent to the standard encoding as in [23]. Thus, the fact that its derivability yields the corresponding derivability in $\mathcal{S}$ is proved by the good old argument. Notice that in our reasoning we never use the cut rule: semi-Thue derivations are encoded by cut-free derivations in $\mathbf{!^s_{b1/2}L^*b}$, the π-projection maps them onto cut-free derivations in $\mathbf{!L^*}$, and they are mapped back onto semi-Thue derivations.

The idea described above is formalised by the following theorem, which serves as the principal technical lemma for Theorem 3.

Theorem 4. *The following three statements are equivalent:*

1. *the word $a_1 \ldots a_n$ is derivable from s in the semi-Thue system $\mathcal{S}$;*
2. *the sequent $\mathcal{X}_{\mathcal{S}}, \mathcal{Z}_{\mathcal{S}}; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots, a_n \Rightarrow s$ is derivable in $\mathbf{!^s_{b1/2}L^*b}$;*
3. *the sequent $\Gamma_{\mathcal{S}}, a_1, \ldots, a_n \Rightarrow s$ is derivable in $\mathbf{!L^*}$.*

Proof. We establish the equivalence by proving round-robin implications: $1 \Rightarrow 2 \Rightarrow 3 \Rightarrow 1$.

$\boxed{1 \Rightarrow 2}$ This part of the proof formalises the idea we explained just before formulating Theorem 4. Proceed by induction on the length of the rewriting sequence. Induction base is $n = 1$, $a_1 = s$, and the necessary sequent, $\mathcal{X}_{\mathcal{S}}, \mathcal{Z}_{\mathcal{S}}; [\varnothing; [\varnothing; \Lambda]], s \Rightarrow s$, is derived as follows:

$$\dfrac{\dfrac{\dfrac{\dfrac{\varnothing; \Lambda \Rightarrow \mathbf{I}}{\varnothing; [\varnothing; \Lambda] \Rightarrow \langle\rangle\mathbf{I}}\ \langle\rangle R}{\varnothing; [\varnothing; [\varnothing; \Lambda]] \Rightarrow \langle\rangle\langle\rangle\mathbf{I}}\ \langle\rangle R \qquad \dfrac{\dfrac{\dfrac{\varnothing; !Z_1 \Rightarrow !Z_1 \quad \ldots \quad \varnothing; !Z_n \Rightarrow !Z_n \quad \dfrac{\varnothing; s \Rightarrow s}{\varnothing; \mathbf{I}, \ldots, \mathbf{I}, s \Rightarrow s}\ \mathbf{I}L\ (n \text{ times})}{\varnothing; \mathbf{I} \,/\, !Z_1, !Z_1, \ldots, \mathbf{I} \,/\, !Z_n, !Z_n, s \Rightarrow s}\ /L\ (n \text{ times})}{\mathbf{I} \,/\, !Z_1, \ldots, \mathbf{I} \,/\, !Z_n, !Z_1, \ldots, !Z_n; s \Rightarrow s}\ !P\ (2n \text{ times})}{\mathbf{I} \,/\, !Z_1, \ldots, \mathbf{I} \,/\, !Z_n, !Z_1, \ldots, !Z_n; \mathbf{I}, s \Rightarrow s}\ \mathbf{I}L}{\mathbf{I} \,/\, !Z_1, \ldots, \mathbf{I} \,/\, !Z_n, !Z_1, \ldots, !Z_n; \mathbf{I} \,/\, (\langle\rangle\langle\rangle\mathbf{I}), [\varnothing; [\varnothing; \Lambda]], s \Rightarrow s}\ /L}{\mathbf{I} \,/\, !Z_1, \ldots, \mathbf{I} \,/\, !Z_n, \mathbf{I} \,/\, (\langle\rangle\langle\rangle\mathbf{I}), !Z_1, \ldots, !Z_n; [\varnothing; [\varnothing; \Lambda]], s \Rightarrow s}\ !P$$

For the induction step, we first establish derivability of the following "landing" rule:

$$\dfrac{\mathcal{X}_{\mathcal{S}}, \mathcal{Z}_{\mathcal{S}}; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots, a_i, A_j, a_{i+1}, \ldots, a_n \Rightarrow s}{\mathcal{X}_{\mathcal{S}}, \mathcal{Z}_{\mathcal{S}}; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots, a_i, a_{i+1}, \ldots, a_n \Rightarrow s}\ \mathit{land}$$

for any $A_j \in \mathcal{A}_\mathcal{S}$. This rule is derived as follows:

$$\begin{array}{c l}
\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots a_i, A_j, a_{i+1}, \ldots, a_n \Rightarrow s & \\ \hline
\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}, A_j; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots a_i, a_{i+1}, \ldots, a_n \Rightarrow s & !P \\ \hline
\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; !A_j, [\varnothing; [\varnothing; \Lambda]], a_1, \ldots a_i, a_{i+1}, \ldots, a_n \Rightarrow s & !L \\ \hline
\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; !A_j, [\varnothing; [\varnothing; \mathbf{I}]], a_1, \ldots a_i, a_{i+1}, \ldots, a_n \Rightarrow s & \mathbf{I}L \\ \hline
\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; !A_j, [\varnothing; \langle\rangle\mathbf{I}], a_1, \ldots a_i, a_{i+1}, \ldots, a_n \Rightarrow s & \langle\rangle L \\ \hline
\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; !A_j, \langle\rangle\langle\rangle\mathbf{I}, a_1, \ldots a_i, a_{i+1}, \ldots, a_n \Rightarrow s & \langle\rangle L \\ \hline
\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; !A_j \bullet \langle\rangle\langle\rangle\mathbf{I}, a_1, \ldots a_i, a_{i+1}, \ldots, a_n \Rightarrow s & \bullet L \\ \hline
\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; [\varnothing; []^{-1}(!A_j \bullet \langle\rangle\langle\rangle\mathbf{I})], a_1, \ldots, a_i, a_{i+1}, \ldots, a_n \Rightarrow s & []^{-1}L \\ \hline
\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; [\varnothing; ![]^{-1}(!A_j \bullet \langle\rangle\langle\rangle\mathbf{I})], a_1, \ldots, a_i, a_{i+1}, \ldots, a_n \Rightarrow s & !D \\ \hline
\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; [![]^{-1}(!A_j \bullet \langle\rangle\langle\rangle\mathbf{I}); \Lambda], a_1, \ldots, a_i, a_{i+1}, \ldots, a_n \Rightarrow s & !P \\ \hline
\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots, a_i, a_{i+1}, \ldots, a_n \Rightarrow s & !C\ (![]^{-1}(!A_j \bullet \langle\rangle\langle\rangle\mathbf{I}) = !Z_i \in \mathcal{Z}_\mathcal{S})
\end{array}$$

Using the *land* rule, the last rewriting step, from $a_1 \ldots a_i x_1 \ldots x_m a_r \ldots a_n$ to $a_1 \ldots a_i y_1 \ldots y_k a_r \ldots a_n$ is simulated as follows. Since $x_1 \ldots x_m \to y_1 \ldots y_k$ is a rewriting rule of $\mathcal{S}$, the formula $A_j = (x_1 \bullet \ldots \bullet x_m) / (y_1 \bullet \ldots \bullet y_k)$ belongs to $\mathcal{A}_\mathcal{S}$. Thus, the *land* rule is applicable.

$$\dfrac{\dfrac{\dfrac{\varnothing; y_1 \Rightarrow y_1 \quad \ldots \quad \varnothing; y_k \Rightarrow y_k}{\varnothing; y_1, \ldots, y_k \Rightarrow y_1 \bullet \ldots \bullet y_k}\,\bullet R \qquad \dfrac{\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots, a_i, x_1, \ldots, x_m, a_r, \ldots, a_n \Rightarrow s}{\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots, a_i, x_1 \bullet \ldots \bullet x_m, a_r, \ldots, a_n \Rightarrow s}\,\bullet L}{\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots, a_i, (x_1 \bullet \ldots \bullet x_m) / (y_1 \bullet \ldots \bullet y_k), y_1, \ldots, y_k, a_r, \ldots, a_n \Rightarrow s}\,/L}{\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots, a_i, y_1, \ldots, y_k, a_r, \ldots, a_n \Rightarrow s}\,\mathit{land}$$

For the case of empty $x_1 \ldots x_m$ or $y_1 \ldots y_m$ the derivations are a bit different:

$$\dfrac{\dfrac{\dfrac{\varnothing; y_1 \Rightarrow y_1 \quad \ldots \quad \varnothing; y_k \Rightarrow y_k}{\varnothing; y_1, \ldots, y_k \Rightarrow y_1 \bullet \ldots \bullet y_k}\,\bullet R \qquad \dfrac{\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots, a_i, a_r, \ldots, a_n \Rightarrow s}{\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots, a_i, \mathbf{I}, a_r, \ldots, a_n \Rightarrow s}\,\mathbf{I}L}{\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots, a_i, \mathbf{I} / (y_1 \bullet \ldots \bullet y_k), y_1, \ldots, y_k, a_r, \ldots, a_n \Rightarrow s}\,/L}{\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots, a_i, y_1, \ldots, y_k, a_r, \ldots, a_n \Rightarrow s}\,\mathit{land}$$

$$\dfrac{\dfrac{\dfrac{}{\varnothing; \Lambda \Rightarrow \mathbf{I}}\,\mathbf{I}R \qquad \dfrac{\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots, a_i, x_1, \ldots, x_m, a_r, \ldots, a_n \Rightarrow s}{\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots, a_i, x_1 \bullet \ldots \bullet x_m, a_r, \ldots, a_n \Rightarrow s}\,\bullet L}{\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots, a_i, (x_1 \bullet \ldots \bullet x_m) / \mathbf{I}, a_r, \ldots, a_n \Rightarrow s}\,/L}{\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots, a_i, a_r, \ldots, a_n \Rightarrow s}\,\mathit{land}$$

$\boxed{2 \Rightarrow 3}$ By Proposition 2, since $\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots, a_n \Rightarrow s$ is derivable in $\mathbf{!^s_{b1/2}L^*b}$, $\pi(\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots, a_n) \Rightarrow s$ is derivable in $\mathbf{!L^*}$. The π-projection of Z_i is $!A_i \bullet \mathbf{I}$ and the π-projection of $\mathbf{I} / (\langle\rangle\langle\rangle\mathbf{I})$ is $\mathbf{I} / \mathbf{I}$. Thus, by definition of π,

$$\pi(\mathcal{X}_\mathcal{S}, \mathcal{Z}_\mathcal{S}; [\varnothing; [\varnothing; \Lambda]], a_1, \ldots, a_n) =$$
$$!(\mathbf{I} / !(!A_1 \bullet \mathbf{I})), \ldots, !(\mathbf{I} / !(!A_n \bullet \mathbf{I})), !(\mathbf{I} / \mathbf{I}), !!(!A_1 \bullet \mathbf{I}), \ldots, !!(!A_n \bullet \mathbf{I}), a_1, \ldots, a_n,$$

and the sequent

$$!(\mathbf{I} / !(!A_1 \bullet \mathbf{I})), \ldots, !(\mathbf{I} / !(!A_n \bullet \mathbf{I})), !(\mathbf{I} / \mathbf{I}), !!(!A_1 \bullet \mathbf{I}), \ldots, !!(!A_n \bullet \mathbf{I}), a_1, \ldots, a_n \Rightarrow s$$

is derivable in $\mathbf{!L^*}$. Next, $\Lambda \Rightarrow !(\mathbf{I} \,/\, !(!A_i \bullet \mathbf{I}))$, $\Lambda \Rightarrow !(\mathbf{I} \,/\, \mathbf{I})$, and $!A_i \Rightarrow !!(!A_i \bullet \mathbf{I})$ are derivable in $\mathbf{!L^*}$:

$$\dfrac{\dfrac{\dfrac{\Lambda \Rightarrow \mathbf{I}}{!(!A_i \bullet \mathbf{I}) \Rightarrow \mathbf{I}}\,!W}{\Lambda \Rightarrow \mathbf{I} \,/\, !(!A_i \bullet \mathbf{I})}\,/R}{\Lambda \Rightarrow !(\mathbf{I} \,/\, !(!A_i \bullet \mathbf{I}))}\,!R \qquad \dfrac{\dfrac{\mathbf{I} \Rightarrow \mathbf{I}}{\Lambda \Rightarrow \mathbf{I} \,/\, \mathbf{I}}\,/R}{\Lambda \Rightarrow !(\mathbf{I} \,/\, \mathbf{I})}\,!R \qquad \dfrac{\dfrac{\dfrac{!A_i \Rightarrow !A_i \quad \Lambda \Rightarrow \mathbf{I}}{!A_i \Rightarrow !A_i \bullet \mathbf{I}}\,\bullet R}{!A_i \Rightarrow !(!A_i \bullet \mathbf{I})}\,!R}{!A_i \Rightarrow !!(!A_i \bullet \mathbf{I})}\,!R$$

Using cut in $\mathbf{!L^*}$, we obtain

$$!A_1, \ldots, !A_n, a_1, \ldots, a_n \Rightarrow s,$$

which is exactly the necessary $\Gamma_{\mathcal{S}}, a_1, \ldots, a_n \Rightarrow s$.

Next, we can eliminate applications of cut in the $\mathbf{!L^*}$ derivation by Proposition 1.

$\boxed{3 \Rightarrow 1}$ This part comes directly from the standard undecidability proof for $\mathbf{!L^*}$, see [16]. Consider the derivation of $\Gamma_{\mathcal{S}}, a_1, \ldots, a_n \Rightarrow s$ in $\mathbf{!L^*}$. Recall that the cut rule can be eliminated by Proposition 1, so we can suppose that this derivation is cut-free. All formulae in this derivation are subformulae of the goal sequent, and the only applicable rules are $\bullet L$, $\bullet R$, $/L$, and rules operating ! in the antecedent: $!L$, $!C_{1,2}$, $!W$.

Now let us hide all the formulae which include /. Since all formulae with ! in our sequent included /, this trivialises all !-operating rules. Next, let us replace all $\bullet$'s in the antecedents with commas, and remove unnecessary $\mathbf{I}$'s there. This, in its turn, trivialises $\bullet L$ and $\mathbf{I}L$. All sequents in our derivation are now of the form $b_1, \ldots, b_s \Rightarrow C$, where $s \geq 0$ and $C = c_1 \bullet \ldots \bullet c_r$ ($r \geq 1$) or $C = \mathbf{I}$. For the sake of uniformity, we also write $C = \mathbf{I}$ as $C = c_1 \bullet \ldots \bullet c_r$ with $r = 0$. Inference rules reduce to

$$\frac{b_{i+1}, \ldots, b_j \Rightarrow y_1 \bullet \ldots \bullet y_k \quad b_1, \ldots, b_i, x_1, \ldots, x_m, b_{j+1}, \ldots, b_s \Rightarrow C}{b_1, \ldots, b_i, b_{i+1}, \ldots, b_j, b_{j+1}, \ldots, b_s \Rightarrow C}$$

where $x_1, \ldots, x_m \to y_1, \ldots y_k$ is a rewriting rule of $\mathcal{S}$;

$$\frac{b_1, \ldots, b_i \Rightarrow c_1 \bullet \ldots \bullet c_j \quad b_{i+1}, \ldots, b_s \Rightarrow c_{j+1} \bullet \ldots \bullet c_r}{b_1, \ldots, b_i, b_{i+1}, \ldots, b_s \Rightarrow c_1 \bullet \ldots \bullet c_j \bullet c_{j+1} \bullet \ldots \bullet c_r}$$

and, finally, we have axioms of the form $a \Rightarrow a$ and $\Lambda \Rightarrow \mathbf{I}$.

Now straightforward induction on derivation establishes the following fact: if $b_1, \ldots, b_s \Rightarrow c_1 \bullet \ldots \bullet c_r$ is derivable in the simplified calculus presented above, then $b_1 \ldots b_s$ is derivable from $c_1 \ldots c_r$ in the semi-Thue system $\mathcal{S}$. This finishes our proof.

5 Conclusion

In this paper, we have discussed a new version of interaction between brackets and exponential, recently proposed by Glyn Morrill [32, 31]. This system is intended to be the basis for the categorial grammar parser CatLog3. For a fragment of this system, we have proved undecidability of the derivability problem. Undecidability for the corresponding fragment of a previous version [28] of Morrill's system was shown in [14]. The new contraction rule introduced by Morrill, however, significantly differs from the earlier ones, and, unfortunately, existing undecidability proofs [23, 12, 14] do not directly extend to the new version. The necessary new technique for proving undecidability with the new form of the contraction rule [32, 31] was developed in the present paper.

Future Work

One of the referees pointed out the following interesting question. The calculus $\mathbf{!^s_{b1/2}L^*b}$, considered in this paper, can generate ungrammatical sentences (see end of Section 2), since it allows the user to put brackets on empty substrings of the sentence being parsed. The question is whether the undecidability proof presented in this paper is still valid for the variant of $\mathbf{!^s_{b1/2}L^*b}$ where such bracketing is disallowed. Furthermore, for the sake of cut-elimination, this nonemptiness restriction should possibly be propagated to all bracketed expressions and generally all meta-formulae inside the derivation. In particular, this condition would require excluding the product unit, **I**. The product unit is essentially used in our undecidability proof, but potentially could be replaced by a unit-free formula (cf. [19]). We leave this problem open for future research. There are also issues with reconciling non-emptiness restrictions, cut-elimination, the substitution property, and the full-power exponential modality [11, 13]. Settling these issues for $\mathbf{!^s_{b1/2}L^*b}$, the calculus with brackets and non-standard rules for !, requires further investigation.

There are several other problems which are still open. One open problem is whether syntactic condition could be imposed on the formulae under ! (like the so-called bracket non-negative condition [28, 14]), under which the system becomes decidable. There is also an issue of extending the bracket-inducing algorithm from [30] to the system with the subexponential discussed in the present paper. Finally, it is interesting whether our result could be strengthened to the undecidability of the one-division fragment of $\mathbf{!^s_{b1/2}L^*b}$, as it was done in [12] using Buszkowski's technique [4] of encoding semi-Thue derivations in the one-divison Lambek calculus.

Acknowledgments

The authors are thankful to the anonymous referees for helpful comments and interesting questions. They would also like to thank Glyn Morrill for fruitful discussions of linguistic motivations for the calculi Morrill introduced in [27–29, 31, 32], which are considered in this paper.

Financial Support

The work of Max Kanovich and Andre Scedrov was supported by the Russian Science Foundation under grant 17-11-01294 and performed at National Research

University Higher School of Economics, Moscow, Russia. The work of Stepan Kuznetsov was supported by the Young Russian Mathematics award, by the grant MK-430.2019.1 of the President of Russia, and by the Russian Foundation for Basic Research grant 18-01-00822. Section 3 was contributed by Kanovich and Scedrov. Section 4 was contributed by Kuznetsov. Sections 1, 2, and 5 were contributed jointly and equally by all co-authors.

References

1. Abrusci, V. M.: A comparison between Lambek syntactic calculus and intuitionistic linear logic. Zeitschr. math. Logik Grundl. Math. **36**, 11–15, (1990). https://doi.org/10.1002/malq.19900360103
2. Ajdukiewicz, K.: Die syntaktische Konnexität. Studia Philosophica **1**, 1–27 (1935)
3. Bar-Hillel, Y.: A quasi-arithmetical notation for syntactic description. Language **29**, 47–58 (1953)
4. Buszkowski, W.: Some decision problems in the theory of syntactic categories. Zeitschr. math. Logik Grundl. Math. **28**, 539–548 (1982). https://doi.org/10.1002/malq.19820283308
5. Buszkowski, W., Palka, E.: Infinitary action logic: complexity, models and grammars. Studia Logica **89**, 1–18 (2008). https://doi.org/10.1007/s11225-008-9116-7
6. Chomsky, N.: Three models for the description of language. IRE Trans. Inform. Theory **I T-2** (3), 113–124 (1956)
7. Girard, J.-Y.: Linear logic. Theor. Comput. Sci. **50**(1), 1–102 (1987). https://doi.org/10.1016/0304-3975(87)90045-4
8. Girard, J.-Y.: A new constructive logic: classical logic. Math. Struct. Comput. Sci. **1**(3), 255–296 (1991) https://doi.org/10.1017/S0960129500001328
9. Girard, J.-Y.: On the unity of logic. Annals Pure Appl. Logic **59**(3), 201–217 (1993). https://doi.org/10.1016/0168-0072(93)90093-S
10. Girard, J.-Y.: Light linear logic. Inform. Comput. **143**(2), 175–204 (1998). https://doi.org/10.1006/inco.1998.2700
11. Kanovich, M., Kuznetsov, S., Scedrov, A.: On Lambek's restriction in the presence of exponential modalities. In: Artemov, S., Nerode, A. (eds.) LFCS 2016, LNCS, vol. 9537, pp. 146–158 (2016). https://doi.org/10.1007/978-3-319-27683-0_11
12. Kanovich, M., Kuznetsov, S., Scedrov, A.: Undecidability of the Lambek calculus with a relevant modality. In: Foret, A. et al (eds.) Formal Grammar 2015/2016, LNCS, vol. 9804, pp. 240–256 (2016). https://doi.org/10.1007/978-3-662-53042-9_14
13. Kanovich, M., Kuznetsov, S., Scedrov, A.: Reconciling Lambek's restriction, cut-elimination, and substitution in the presence of exponential modalities. Annals Pure Appl. Logic, accepted for publication. arXiv preprint 1608.02254 (2016).
14. Kanovich, M., Kuznetsov, S., Scedrov, A.: Undecidability of the Lambek calculus extended with subexponential and bracket modalities. In: Klasing, R., Zeitoun, M. (eds.) FCT 2017, LNCS, vol. 10472, pp. 326–340. Springer, Berlin, Heidelberg (2017). https://doi.org/10.1007/978-3-662-55751-8_26
15. Kanovich, M., Kuznetsov, S., Scedrov, A.: Lambek calculus enriched with multiplexing (abstract). In Internat. Conf. Mal'tsev Meeting 2018, Collection of Abstracts. Sobolev Institute of Mathematics & Novosibirsk State University, Novosibirsk (2018). http://www.math.nsc.ru/conference/malmeet/18/maltsev18.pdf

16. Kanovich, M., Kuznetsov, S., Nigam, V., Scedrov, A.: Subexponentials in non-commutative linear logic. Math. Struct. Comput. Sci. (2018). https://doi.org/10.1017/S0960129518000117 (FirstView: published online May 2, 2018)
17. Kanovich, M., Kuznetsov, S., Nigam, V., Scedrov, A.: A logical framework with commutative and non-commutative subexponentials. In: Galmiche, D., Schulz, S., Sebastiani, R. (eds.) IJCAR 2018, LNCS/LNAI, vol. 10900, pp. 228–245. Springer (2018). https://doi.org/10.1007/978-3-319-94205-6_16
18. Kozen, D.: On the complexity of reasoning in Kleene algebra. Inform. Comput. **179**, 152–162 (2002). https://doi.org/10.1006/inco.2001.2960
19. Kuznetsov, S.L.: On the Lambek calculus with a unit and one division. Moscow Univ. Math. Bull. **66**(4), 173–175 (2011). https://doi.org/10.3103/S0027132211040085
20. Kuznetsov, S.: The Lambek calculus with iteration: two variants. In: Kennedy, J., de Queiroz, R. (eds.), WoLLIC 2017, LNCS, vol. 10388, pp. 182–198. Springer (2017). https://doi.org/10.1007/978-3-662-55386-2_13
21. Lafont, Y.: Soft linear logic and polynomial time. Theor. Comput. Sci. **318**(12), 163–180 (2004). https://doi.org/10.1016/j.tcs.2003.10.018
22. Lambek, J.: The mathematics of sentence structure. Amer. Math. Monthly **65**, 154–170 (1958). https://doi.org/10.2307/2310058
23. Lincoln, P., Mitchell, J., Scedrov, A., Shankar, N.: Decision problems for propositional linear logic. Annals Pure Appl. Log. **56**(1–3), 239–311 (1992). https://doi.org/10.1016/0168-0072(92)90075-B
24. Markov, A.: On the impossibility of certain algorithms in the theory of associative systems. Doklady Acad. Sci. USSR (N. S.) **55**, 583–586 (1947)
25. Moortgat, M.: Multimodal linguistic inference. J. Log. Lang. Inform. **5**(3/4), 349–385 (1996). https://doi.org/10.1007/BF00159344
26. Morrill, G.: Categorial formalisation of relativisation: pied piping, islands, and extraction sites. Technical Report LSI-92-23-R, Universitat Politècnica de Catalunya (1992)
27. Morrill, G. V.: Categorial grammar: logical syntax, semantics, and processing. Oxford University Press (2011)
28. Morrill, G., Valentín, O.: Computational coverage of TLG: nonlinearity. In Proc. NLCS 2015, EPiC Series, vol. 32, pp. 51–63 (2015)
29. Morrill, G.: Grammar logicised: relativisation. Linguist. Philos. **40**(2), 119–163 (2017). https://doi.org/10.1007/s10988-016-9197-0
30. Morrill, G., Kuznetsov, S., Kanovich, M., Scedrov, A.: Bracket induction for Lambek calculus with bracket modalities. In Foret, A., Kobele, G., Pogodalla, S., Formal Grammar 2018, LNCS, vol. 10950, pp. 84–101. Springer (2018). https://doi.org/10.1007/978-3-662-57784-4_5
31. Morrill, G.: A note on movement in logical grammar. J. Lang. Model. **6**(2), 353–363 (2018). https://doi.org/10.15398/jlm.v6i2.233
32. Morrill, G.: Parsing/theorem-proving for logical grammar CatLog3. J. Log. Lang. Inform. (2019). https://doi.org/10.1007/s10849-018-09277-w (published online Jan 18, 2019)
33. Post, E.L.: Recursive unsolvability of a problem of Thue. J. Symb. Log. **12**, 1–11 (1947)
34. Thue, A.: Probleme über Veränderungen von Zeichenreihen nach gegebener Regeln. Kra. Vidensk. Selsk. Skrifter. **10** (1914)