On the generative capacity of multi-modal Categorial Grammars

Gerhard Jäger
Institute for Research in Cognitive Science
University of Pennsylvania
3401 Walnut Street, Suite 400 A
Pennsylvania, PA 19143, USA
gjaeger@linc.cis.upenn.edu

Abstract

In Moortgat 1996 the Lambek Calculus **L** (Lambek 1958) is extended by a pair of residuation modalities $\diamond$ and $\Box^{\downarrow}$. Categorial Grammars based on the resulting logic $\mathbf{L}\diamond$ are attractive for linguistic purposes since they offer a compromise between the strict constituent structures imposed by context free grammars and related formalisms on the one hand, and the complete absence of hierarchical information in Lambek grammars on the other hand. The paper contains some results on the generative capcity of Categorial Grammars based on $\mathbf{L}\diamond$. First it is shown that adding residuation modalities does not extend the weak generative capacity. This is proved by extending the proof for the context freenes of **L**-grammars from Pentus 1993 to $\mathbf{L}\diamond$. Second the strong generative capacity of $\mathbf{L}\diamond$-grammars is compared to context free grammars. The results are mainly negative; The set of tree languages generated by $\mathbf{L}\diamond$-grammars neither contains nor is contained in the class of context free tree languages.

1 Introduction

Lambek style Categorial Grammar is characterized by the complete absence of hierarchical information in syntactic structures. All syntactic information resides in the linear order of the lexical entries. This is an advantage in all cases where the traditional notion of constituency proves to be too rigid, notably in connection with non-constituent co-ordination. Doing away with hierarchical structure altogether seems to be too radical, however. For one thing, constituents are a *factum brutum* that cannot be discussed away in phonology. But also in the realm of syntax proper and its connection to semantics, certain generalizations cannot be formulated in purely linear terms. Island constraints and quantifier scope are obvious examples. These and similar considerations motivate the extension of **L** with unary modalities in many recent works on type logical grammar. Pentus 1993 makes the inadequacy of **L**-based grammars precise in proving that they are weakly equivalent to context-free grammars.

Multimodality extends the generative power. In a sense Categorial Grammar now repeats the history of transformational grammar, since the attempt to leave context-freeness leads to a grammar format that is Turing complete. A proof of this claim can be found in Carpenter 1995. To develop a realistic framework for NL grammar, the use of multimodality therefore has to be restricted appropriately. After introducing some basic prerequisites in sections 2 and 3, we prove in sections 4 and 5 that multimodal Lambek grammar is still context free as long as the inference rules governing the behavior of the unary modalities are restricted to the residuation laws. This has two notable consequences. First, it shows that the use of multimodality is "harmless" and we can use it to formalize syntactic island constraints and prosodic phrasing without suffering from a generative overkill. Second, it shows that there is a tight connection between interaction postulates and generative capacity. This may eventually lead to a taxonomy of languages and grammars that is much more fine-grained than the traditional Chomsky hierarchy.

Since $\mathbf{L}\diamond$-grammars—unlike **L**-grammars—generate trees rather than strings, the issue of strong generative capacity can be addressed. This is done in section 6. It will turn out that context free grammars and $\mathbf{L}\diamond$-grammars are in a sense complementary: CFGs restrict the breadth of local trees, i.e. the number of daughter nodes that a mother node can have, while trees can be arbitrarily high. $\mathbf{L}\diamond$-grammars impose the opposite constraint: while local trees can be arbitrily broad, the height of trees is bounded.

2 Two notions of recognition

A Lambek Grammar consists of a lexicon $\mathcal{L}$, i.e. a finite relation between lexical items and categories, and a finite set $\mathcal{D}$ of designated categories. A string $l_1 \ldots l_n$ of lexical assignments is recognized by the grammar iff there is a string $A_1 \ldots A_n$ of categories such that $\langle l_i, A_i \rangle \in \mathcal{L}$ for all $1 \leq i \leq n$ and there is a category $S \in \mathcal{D}$ such that $\vdash_{\mathbf{L}} A_1 \ldots A_n \Rightarrow S$.
Now let us consider the simplest multimodal extension of $\mathbf{L}$, $\mathbf{L}\Diamond$ from Moortgat 1996. Here the inventory of type-forming connectives is extended by the unary operators $\Diamond$ and $\Box^{\downarrow}$. Premises of sequents are bracketed strings of types, i.e. sequences of trees. As additional inference rules we have the rules of use and rules of proof for both connectives (cf. Moortgat 1997):

Definition 1 (Sequent Calculus for $\mathbf{L}\Diamond$)

$$\frac{X, (A), Y \Rightarrow B}{X, \Diamond A, Y \Rightarrow B} \Diamond L \qquad \frac{X \Rightarrow A}{(X) \Rightarrow \Diamond A} \Diamond R$$

$$\frac{X, A, Y \Rightarrow B}{X, (\Box^{\downarrow} A), Y \Rightarrow B} \Box^{\downarrow} L \qquad \frac{(X) \Rightarrow A}{X \Rightarrow \Box^{\downarrow} A} \Box^{\downarrow} R$$

This offers two ways to define string recognition in a $\mathbf{L}\Diamond$ grammar (where a grammar is still a lexical assignment plus a set of designated categories):

Definition 2 (S-Recognition) $l_1 \ldots l_n$ is s-recognized iff there are types $A_1 \ldots A_n, S$ such that for all i ($1 \leq i \leq n$): $\langle l_i, A_1 \rangle \in \mathcal{L}$, $S \in \mathcal{D}$, and $\vdash_{\mathbf{L}\Diamond} A_1, \ldots, A_n \Rightarrow S$.

Definition 3 (T-Recognition) $l_1 \ldots l_n$ is t-recognized if there are types $A_1 \ldots A_n, S$ such that for all i ($1 \leq i \leq n$): $\langle l_i, A_1 \rangle \in \mathcal{L}$, $S \in \mathcal{D}$, and there is a sequence of trees X with $A_1, \ldots, A_n$ as its yield such that $\vdash_{\mathbf{L}\Diamond} X \Rightarrow S$.

In the next section we will prove that $\mathbf{L}\Diamond$-based grammars both s-recognize and t-recognize exactly the context-free languages. To this end we make use of the translation from $\mathbf{L}\Diamond$ to $\mathbf{L}$ given in Versmissen 1996. This translation allows us to reduce the proof in both cases to Pentus' proof.

3 Translation

Definition 4 Let $\mathcal{A}$ be the (finite) set of atoms of $\mathbf{L}\Diamond$ and $\mathcal{A}'$ be a disjoint set of atoms of the same cardinality and t_0, t_1 two atoms with $t_0, t_1 \notin \mathcal{A} \cup \mathcal{A}'$. Let f be a bijective function from $\mathcal{A}$ to $\mathcal{A}'$. The translation is given by the clauses

$$\begin{array}{rcll} p^{\#} & = & f(p) & \qquad\qquad (p \text{ atomic}) \\ (A \bullet B)^{\#} & = & A^{\#} \bullet B^{\#} & \\ (A \setminus B)^{\#} & = & A^{\#} \setminus B^{\#} & \\ (A/B)^{\#} & = & A^{\#}/B^{\#} & \\ (\Diamond A)^{\#} & = & t_0 \bullet A^{\#} \bullet t_1 & \\ (\Box^{\downarrow} A)^{\#} & = & t_0 \setminus A^{\#}/t_1 & \\ (A, X)^{\#} & = & A^{\#}, X^{\#} & \\ ((X))^{\#} & = & t_0, X^{\#}, t_1 & \end{array}$$

Lemma 1

$$\vdash_{\mathbf{L}\Diamond} X \Rightarrow A \iff \vdash_{\mathbf{L}} X^{\#} \Rightarrow A^{\#}$$

Proof: see Versmissen 1996

4 The context-freeness of s-recognition

Moortgat 1996 proves that $\mathbf{L}\Diamond$ enjoys the sub-formula property and cut elimination. This leads to the following lemma:

Lemma 2 If a sequent Σ contains no modal operators, then Σ is derivable in $\mathbf{L}\Diamond$ iff it is derivable in $\mathbf{L}$.

Proof: An inspection of the inference rules of $\mathbf{L}\Diamond$ shows that the rules involving modalities only come into play if the conclusion contains modal operators. By the sub-formula property, no modalities occur in a cut free derivation of Σ. Hence any cut free derivation of Σ in $\mathbf{L}\Diamond$ is also a derivation of Σ in $\mathbf{L}$. Since every inference rule of $\mathbf{L}$ is also a rule of $\mathbf{L}\Diamond$, the other direction holds as well.
$\dashv$

Theorem 1 **L**$\Diamond$-grammars s-recognize exactly the context-free languages.

Proof: Let an **L**$\Diamond$-grammar $\langle \mathcal{L}, \mathcal{D} \rangle$ be given. We transform it into and **L**-grammar $\langle \mathcal{L}', \mathcal{D}' \rangle$ by assuming $\mathcal{L}' = \{\langle v, [A] \rangle | \langle v, A \rangle \in \mathcal{L}\}$ and $\mathcal{D}' = \{[S] | S \in \mathcal{D}\}$. By lemma 1, these grammars (s-)recognize the same language. Pentus' proof shows that this language is context-free, since it is recognized by an **L**-grammar. So every language s-recognized by a **L**$\Diamond$-grammar is context-free. As for the other direction, assume that a given language L is context-free. Cohen 1967 proves that there is an **L**-grammar which recognizes L. Since neither the lexical nor the designated categories contain modal operators, lemma 2 entails that this grammar s-recognizes the very same language L if conceived as an **L**$\Diamond$-grammar.
$\dashv$

5 T-recognition

In this section we will demonstrate that **L**$\Diamond$-grammars t-recognize exactly the context free languages. The proof idea is adopted from Pentus' proof. First we extend Roorda's 1991 proof of the Interpolation Theorem for **L** to **L**$\Diamond$. Then we show that all derivable **L**$\Diamond$-sequences can be derived from a finite set of axioms by using only cut if we impose an upper bound to the length of formulas. This finite axiom set can be used as the core of a context free grammar that is equivalent to a given **L**$\Diamond$-grammar.
Let X be a sequence of **L**-formulas. By $m(X)$ we refer to the multiset of atomic types occurring in X.

Theorem 2 (Interpolation Theorem) Let X and Y be bracketed sequences of **L**$\Diamond$-formulas such that $X = ZYW$[1], and A be an **L**$\Diamond$-formula such that $\mathbf{L}\Diamond \vdash X \Rightarrow A$. Then there is a formula B such that $\mathbf{L}\Diamond \vdash Y \Rightarrow B, \mathbf{L}\Diamond \vdash ZBW \Rightarrow A$, and $m(B^\#) \subseteq m(Y^\#) \cap (m((ZW)^\#) \cup m(A^\#))$.

Proof: By induction over cut-free derivations. We refer to the interpolant of a sequence U in the premise of a sequent rule with "$i(U)$"; for the interpolant in the conclusion we use "$j(U)$".[2]

[1] Z and W need not be well-bracketed
[2] The **L**-part of the proof is essentially due to Roorda 1991.

1. $$\frac{}{C \Rightarrow C}\, id$$

   $X = Y = B = C$

2. $$\frac{U_1 U_2 \Rightarrow C \qquad V_1 V_2 D W_1 W_2 \Rightarrow E}{V_1 V_2 U_1 U_2\ C \setminus D\ W_1 W_2 \Rightarrow E}\, \setminus L$$

   (a) $j(V_2 U_1) = i(V_2) \bullet i(U_1)$ (V_2, U_1 non-empty)
   (b) $j(U_2\ C \setminus D\ W_1) = i(U_1) \setminus i(DW_1)$ (U_1 non-empty)
   (c) $j(U\ C \setminus D\ W_1) = i(DW_1)$
   (d) $j(V_2 U_1 U_2\ C \setminus D\ W_1) = i(V_2 D W_1)$
   (e) else $j(Y) = i(Y)$

3. $$\frac{C, U \Rightarrow D}{U \Rightarrow C \setminus D}\, \setminus R$$

   (a) $j(Y) = i(Y)$

   The induction steps for $/L$ and $/R$ are analogous.

4. $$\frac{U_1 U_2 C D V_1 V_2 \Rightarrow E}{U_1 U_2 C \bullet D V_1 V_2 \Rightarrow E}\, \bullet L$$

   (a) $j(U_2 C \bullet D V_1) = i(U_2 C D V_1)$
   (b) else $j(Y) = i(Y)$

5. $$\frac{U_1 U_2 \Rightarrow C \qquad V_1 V_2 \Rightarrow D}{U_1 U_2 V_1 V_2 \Rightarrow C \bullet D}\, \bullet R$$

   (a) $j(U_2 V_1) = i(U_2) \bullet i(V_1)$ (U_2, V_1 non-empty)
   (b) else $j(Y) = i(Y)$

6. $$\frac{U_1 U_2 (C) V_1 V_2 \Rightarrow D}{U_1 U_2 \Diamond C V_1 V_2 \Rightarrow D} \Diamond L$$

   (a) $j(U_2 \Diamond C V_1) = i(U_2 (C) V_1)$ (U_2, V_1 non-empty)
   (b) else $j(Y) = i(Y)$

7. $$\frac{U \Rightarrow C}{(U) \Rightarrow \Diamond C} \Diamond R$$

   (a) $j((U)) = \Diamond(i(U))$
   (b) else $j(Y) = i(Y)$

8. $$\frac{U_1 U_2 C V_1 V_2 \Rightarrow D}{U_1 U_2 (\Box^{\downarrow} C) V_1 V_2 \Rightarrow D} \Box^{\downarrow} L$$

   (a) $j(\Box^{\downarrow} C) = \Box^{\downarrow} i(C)$
   (b) $j(U_2 (\Box^{\downarrow} C) V_1) = i(U_2 C V_1)$
   (c) else $j(Y) = i(Y)$

9. $$\frac{(U) \Rightarrow C}{U \Rightarrow \Box^{\downarrow} C} \Box^{\downarrow} R$$

   (a) $j(Y) = i(Y)$

$\dashv$

Definition 5 (Length of a type)

1. $\rho(p) = 1$
2. $\rho(A \bullet B) = \rho(A \setminus B) = \rho(A/B) = \rho(A) + \rho(B)$
3. $\rho(\Diamond A) = \rho(\Box^{\downarrow} A) = \rho(A) + 2$

In the sequel we distinguish between *exocentric* bracket pairs that are introduced by $\Diamond R$ and *endocentric* bracket pairs that are introduced by $\Box^{\downarrow}L$.

Lemma 3 Let $X[(Y)] \Rightarrow A$ be a derivable $\mathbf{L}\Diamond$-sequent such that the longest formula occurring in it has length n.

1. If the bracket pair around Y is exocentric, the interpolant of Y is some formula B and the interpolant of (Y) is $\Diamond B$ with $\rho(\Diamond B) \leq n$.

2. If the bracket pair around Y is endocentric, the interpolant of (Y) is some formula B and the interpolant of Y is $\Box^{\downarrow}B$ with $\rho(\Box^{\downarrow}B) \leq n$.

Proof: By induction over cut-free derivations, following the proof of the interpolation theorem. The induction base for exocentric brackets is $\Diamond R$. Here the interpolant[3] of (U) is $\Diamond i(U)$, and the interpolant of U is $i(U)$. From the very notion of interpolation it follows that $\rho(i(U)) \leq \rho(C)$, and thus $\rho(\Diamond i(U)) \leq \rho(\Diamond C)$. Due to the subformula property, $\Diamond C$ must occur as a subformula in $X \Rightarrow A$. Hence $\rho(\Diamond C) \leq n$. Furthermore, note that the interpolant of a certain subsequence in the premise of a sequent rule is always identical to the interpolant of the corresponding subsequence in the conclusion. Thus the interpolant of of Y is $i(U)$, and the interpolant of (Y) is $\Diamond i(U)$.
As for endocentric brackets, the induction base is $\Box^{\downarrow}L$. Here the interpolant of (Y) is $i(C)$ and the interpolant of Y is $\Box^{\downarrow}i(C)$. $\rho(i(C)) \leq \rho(C)$, thus $\rho(\Box^{\downarrow}i(C)) \leq \rho(\Box^{\downarrow}C) \leq n$. $\dashv$

A deductive system is a set of sequents $X \Rightarrow A$ which is closed under Cut. A deductive system is finitely axiomatizable iff it is the closure of a finite set of sequents under Cut. For any natural number n, the deductive system P_n is is the closure of the following set of axioms under Cut: $\{A \Rightarrow B | \mathbf{L} \vdash A \Rightarrow B \text{ and } \rho(A), \rho(B) \leq n\} \cup \{A, B \Rightarrow C | \mathbf{L} \vdash A, B \Rightarrow C \text{ and } \rho(A), \rho(B), \rho(C) \leq n\} \cup \{A \Rightarrow A^{\#} | \rho(A) \leq n\} \cup \{A^{\#} \Rightarrow A | \rho(A) \leq n\}$. Obviously, P_n is finitely axiomatizable.

Lemma 4 Let $A_1, \ldots, A_i \Rightarrow B$ be an $\mathbf{L}\Diamond$-sequent. $A_1, \ldots, A_i \Rightarrow B \in P_n$ iff $\mathbf{L}\Diamond \vdash A_1, \ldots, A_i \Rightarrow B$ and $\rho(A_j), \rho(B) \leq n$.

[3] There may be more than one interpolant for a given sequence. In the sequel, by "the interpolant" we refer to the unique interpolant that is determined by the algorithm given in the proof of the interpolation theorem.

Proof: Pentus proves that the set $\{A \Rightarrow B | \mathbf{L} \vdash A \Rightarrow B$ and $\rho(A), \rho(B) \leq n\} \cup \{A, B \Rightarrow C | \mathbf{L} \vdash A, B \Rightarrow C$ and $\rho(A), \rho(B), \rho(C) \leq n\}$ axiomatizes the set of valid **L**-sequents where the maximal length of a formula does not exceed n. P_n is a proper extension of this deductive system. Now suppose $\mathbf{L}\diamond \vdash A_1, \ldots, A_i \Rightarrow B$ and no formula involved exceeds length n. According to lemma 1, $\mathbf{L} \vdash A_1^\#, \ldots, A_i^\# \Rightarrow B^\#$. Since the translation preserves the length of a formula, $A_1^\#, \ldots, A_i^\# \Rightarrow B^\# \in P_n$. Since the translation relation is derivable in both directions in P_n, by repeated application of Cut we derive $A_1, \ldots, A_n \Rightarrow B \in P_n$.
Now suppose $A_1, \ldots, A_i \Rightarrow B \in P_n$ with $A_1 \ldots A_i, B$ being $\mathbf{L}\diamond$-formulas. We take some sound and complete semantics for $\mathbf{L}$[4] and stipulate for $\mathbf{L}\diamond$-formulas that $\|A\| = \|A^\#\|$. Clearly P_n is sound and complete for this semantics if we restrict ourselves to formulas with a length $\leq n$. Thus $A_1, \ldots, A_i \Rightarrow B$ is valid. Since every $\mathbf{L}\diamond$-formula has the same interpretation as its translation, $A_1^\#, \ldots, A_i^\# \Rightarrow B^\#$ is valid too. By completeness we know that this sequent is **L**-derivable, and by completeness of the translation, we infer that the original sequent is $\mathbf{L}\diamond$-derivable.$\dashv$
In a next step, we extend P_n by adding the axioms $(A) \Rightarrow \diamond A$ and $(\Box^{\downarrow} A) \Rightarrow A$ for all $\mathbf{L}\diamond$-formulas A with $\rho(A) \leq n-2$ and closing under Cut. The new system is dubbed P'_n. P'_n is also finitely axiomatizable.

Lemma 5 Let Σ be an $\mathbf{L}\diamond$-sequent which only involves formulas with a length $\leq n$. Then $\mathbf{L}\diamond \vdash \Sigma$ iff $\Sigma \in P'_n$.

Proof: By induction over the number b of bracket pairs in Σ. For $b = 0$, this is just lemma 4. Now suppose the claim holds for $b = k$, and let $X[(Y)] \Rightarrow A$ be a sequent involving $k+1$ bracket pairs. We have to distinguish two cases. Suppose the bracket pair around Y is exocentric and $\mathbf{L}\diamond \vdash X[(Y)] \Rightarrow A$. Then, according to lemma 3, the interpolant of Y is some formula B and the interpolant of (Y) is $\diamond B$ with $\diamond B \leq n$. Therefore $Y \Rightarrow B, X[\diamond B] \Rightarrow A$ are derivable. Since both sequents contain at most k bracket pairs, they are in P'_n by induction hypothesis. Furthermore, $(B) \Rightarrow \diamond B \in P'_n$. By applying Cut twice, we derive that $X[(Y)] \Rightarrow A \in P'_n$.
Now suppose the bracket pair around Y is endocentric and $\mathbf{L}\diamond \vdash X[(Y)] \Rightarrow A$. According to lemma 3, the interpolant of (Y) is some formula B, the interpolant of Y is $\Box^{\downarrow} B$, and $\rho(\Box^{\downarrow} B) \leq n$. Hence $Y \Rightarrow \Box^{\downarrow} B$ and $X[B] \Rightarrow A$ are valid. Since both sequents involve less than k bracket pairs, both are in

[4] For instance interpretation in ternary frames, cf. Došen 1992

P'_n. Furthermore, $(\Box^{\downarrow}B) \Rightarrow B \in P'_n$. By applying Cut twice, we derive that $X[(Y)] \Rightarrow A \in P'_n$.
As for the other direction, suppose $X[(Y)] \Rightarrow A \in P'_n$. Let us extend the semantics for P_n given above by the stipulation that $\|(X)\| = \|t_0, (X)^{\#}, t_1\|$. It is straightforward to see that P'_n is sound with respect to this interpretation. Therefore $X \Rightarrow A$ is valid under this interpretation. This entails that $X^{\#} \Rightarrow A^{\#}$ is valid too. By completeness of the chosen interpretation for **L**, $\mathbf{L} \vdash X^{\#} \Rightarrow A^{\#}$. By completeness of translation, $\mathbf{L}\Diamond \vdash X \Rightarrow A$.
$\dashv$

Theorem 3 ***L***$\Diamond$*-grammars t-recognize exactly the context-free languages.*

Proof: Suppose a language L is recognized by an $\mathbf{L}\Diamond$-grammar $G_1 = \langle \mathcal{L}, \mathcal{D} \rangle$. Let the maximal length of formulas occurring either in $\mathcal{L}$ or in $\mathcal{D}$ be n. Fix a CFG $G_2 = \langle T, NT, P, S \rangle$ (S not being a formula of **L** or $\mathbf{L}\Diamond$) such that T is the set of lexical items in G_1, $NT = \{A|\rho(A) \leq n\} \cup \{A^{\#}|\rho(A) \leq n\} \cup \{(,), S\}$, $P = \{A \to X | X \Rightarrow A \in P'_n\} \cup \{S \to A | A \in \mathcal{D}\} \cup \{A \to v | \langle v, A \rangle \in \mathcal{L}\} \cup \{(\to \varepsilon,) \to \varepsilon\}$. Suppose a string $l_1 \dots l_i$ is recognized by G_1. Then there is a bracketed sequence X of $\mathbf{L}\Diamond$-formulas such that the yield of X is $A_1, \dots, A_i$, $\langle l_j, A_j \rangle \in \mathcal{L}$, and a formula B with $B \in \mathcal{D}$ and $\mathbf{L}\Diamond \vdash X \Rightarrow B$. By lemma 5, there is a derivation from B to X in P'_n and thus also in G_2. By the construction of G_2, we can prefix this derivation with S, and replace every A_i and every bracket by the empty string. Hence $l_1 \dots l_i$ is recognized by G_2. Now suppose $l_1 \dots l_i$ is recognized by G_2. Then, by the construction of G_2, there must be a bracketed sequence of formulas X with $A_1, \dots, A_i$ as its yield and $\langle l_j, A_j \rangle \in \mathcal{L}$, and a formula $B \in \mathcal{D}$ such that $X \Rightarrow B \in P'_n$. By lemma 5, $\mathbf{L}\Diamond \vdash X \Rightarrow B$. Thus $l_1 \dots l_n$ is recognized by G_1. In other words, every language that is t-recognized by an $\mathbf{L}\Diamond$-grammar is context-free. The proof of the reversed inclusion is identical as in the case of s-recognition. $\dashv$

6 Strong generative capacity

Unlike **L**-grammars, $\mathbf{L}\Diamond$-grammars (t-)recognize trees rather than strings. So it makes sense to ask how they relate to CFG's wrt. their strong capacity. Here an interesting new perspective comes into view. Unlike context free grammars, $\mathbf{L}\Diamond$-grammars do not impose an upper bound on the number of daughter nodes a node in a tree might have. So the set of tree languages generated by $\mathbf{L}\Diamond$ is not contained in the context free tree languages. The inclusion in the other direction doesn't hold either. The proof of the latter fact is based on the insight that every $\mathbf{L}\Diamond$-grammar imposes an upper limit to the height of the trees in its tree language (where the height of a tree is the length of the shortest path from the root to a leaf). No such constraint exists for context free grammars.

We will first prove that the height of the trees in an $\mathbf{L}\Diamond$-language is bounded. Based on this fact, we show that the context free tree languages are not contained in the class of $\mathbf{L}\Diamond$-languages. After this, we show that the other inclusion doesn't hold either.

Definition 6 (Tree recognition) A tree (X) with the yield $l_1 \dots l_n$ is recognized by an $\mathbf{L}\Diamond$-grammar $\langle \mathcal{L}, \mathcal{D} \rangle$ iff there are $\mathbf{L}\Diamond$-formulas $A_1 \dots A_n$ with $\langle l_i, A_i \rangle \in \mathcal{L}$ and there is a formula $S \in \mathcal{D}$ such that the result of replacing every l_i in X by A_i yields a sequence of trees Y with $\mathbf{L}\Diamond \vdash Y \Rightarrow S$.

Note that the outermost bracket pair (i.e. the root node) of the tree to be recognized is ignored in the definition since the premises of $\mathbf{L}\Diamond$-sequents are sequences of trees.

The height of a sequence of trees over some vocabulary V is defined recursively as

1. $h(v) = 0$ if $v \in V$
2. $h(XY) = min(h(X), h(Y))$
3. $h((X)) = h(X) + 1$

Obviously, if a tree of height n is recognized by an $\mathbf{L}\Diamond$-grammar, the corresponding sequence of trees over categories has height $n-1$.

Next we introduce the notion of the modal embedding depth of an occurrence of a formula within a superformula. Following common usage, $A(B)$ ranges over formulas that contain an occurrence of the atom p.

Definition 7

$$\begin{aligned}
d(p,p) &= 0 \\
d(p, A(p) \diamond B) &= d(A, B \diamond A(p)) = d(p, A(p)) \\
&\quad \text{where } \diamond \text{ ranges over } /, \bullet, \backslash \\
d(p, \Diamond A(p)) &= d(p, A(p)) + 1 \\
d(p, \Box^{\downarrow} A(p)) &= d(p, A(p)) - 1
\end{aligned}$$

The modal embedding depth of an occurrence of a formula A in a sequence of trees $X[A]$ ($d(A, X)$) is defined as the number of bracket pairs enclosing A in X. These two notions can be extended to the modal embedding depth of an occurrence of an atom p in a sequent $X \Rightarrow A$. If p occurs in the conclusion, this parameter is $d(p, A)$. If p occurs in some formula B in the premise, it is $d(p, B) + d(B, X)$.

A sequent is called *special* iff every atom that occurs in it occurs exactly twice.[5] **L**$\diamond$ and **L** share the property of *independence of branches.* This means that we can uniformly substitute the formulas in a axiom and all its occurrences down in the proof tree by some other formula, and the resulting proof tree remains valid. Furthermore, we can restrict the identity axiom to atomic formulas. Therefore we can transform any derivable sequent into a special one by renaming the atoms in the axiom leafs of its proof tree. Clearly, the height of the premise of this special sequent is identical to the height of the original premise. Likewise, the modal depth of an atom in a formula or of a formula in the premise in the original sequent are preserved during this transformation. Thus we can restrict out attention in what follows to special sequents.

Two formulas in a special sequent are called *directly connected* iff they share one atom. Two formulas are *connected* iff they are in the transitive closure of the previous relation. A sequent is *connected* iff every formula in it is connected with every other formula.

Lemma 6 Every derivable special sequent in **L**$\diamond$ is connected.

Proof: By induction over sequent derivations. $\dashv$

Lemma 7 Let $X \Rightarrow A$ be a derivable special sequent and p be an atom occurring in it. Then the modal embedding depths of the two occurrences of p in $X \Rightarrow A$ are equal.

Proof: By induction over sequent derivations. $\dashv$

Lemma 8 For any tree language L generated by an **L**$\diamond$-grammar, there is an upper bound for the height of trees in L.

[5] This is the terminology of Buszkowski 1997. Pentus 1993 calls these sequents “thin”.

Proof: Let $G = \langle \mathcal{L}, \mathcal{D} \rangle$ be an $\mathbf{L}\diamond$-grammar that generates L, and let X be a tree in L. Then there is a sequence of trees Y which is isomorphic to the sequence of immediate constituents of X such that the leafs of Y are lexical categories from $\mathcal{L}$, and there is a designated category S from $\mathcal{D}$ such that $\mathbf{L}\diamond \vdash Y \Rightarrow S$. Let $Y' \Rightarrow S'$ be the result of transforming $Y \Rightarrow S$ into a special sequent. Since $Y' \Rightarrow S'$ is connected, there is an atom p which occurs both in S' and in Y'. Let us call the formula from Y which contains p A. Let n be the maximal modal embedding depth of an atom in a formula in $\mathcal{D}$. Then, according to lemma 7, $d(p, A) + d(A, Y') = d(p, S') \leq n$. Let m be the minimal modal embedding depth of an atom in any lexcal category from $\mathcal{L}$. Then $d(p, A) \geq m$. Thus $d(A, Y') \leq n - m$. Consequently, $h(Y') \leq n - m$. Since Y' is isomorphic to Y, $h(Y) \leq n - m$, and therefore $h(X) \leq n - m + 1$. Thus $n - m + 1$ is an upper bound for the height of trees in L. $\dashv$

Theorem 4 There are context free grammars that are not strongly equivalent to any $\mathbf{L}\diamond$-grammar.

Proof: Take the CFG $G = \langle \{a\}, \{S\}, \{S \to SS, S \to aa, S \to aS, S \to Sa\}, S \rangle$. It generates all binary trees where all terminal nodes are labeled with a and all other nodes with S. Clearly there is no upper bound for the height of such trees. It follows from lemma 8 that there can't be any $\mathbf{L}\diamond$-grammar that is strongly equivalent to G. $\dashv$

To prove that the inclusion does not hold in the other direction is even simpler.

Theorem 5 There are $\mathbf{L}\diamond$-grammars that are not strongly equivalent to any context free grammar.

Proof: Take $G = \langle \{\langle a, S \rangle, \langle a, S/S \rangle\}, \{S\} \rangle$. It generates the set of trees (a^n) for $n \geq 1$. So there is no upper bound for the number of daughter nodes of the top node, and therefore there cannot be a strongly equivalent CFG.$\dashv$

7 Conclusion and further research

This article contains two main results. It shows that adding residuation modalities to the Lambek calculus does not affect weak generative capacity, and it demonstrates that the class of tree languages defined by $\mathbf{L}\Diamond$is a genuinely new class that is independent from the corresponding class generated by context free grammars. These two results together entail that the $\mathbf{L}\Diamond$-grammars do not coincide with other well-studied grammar formats either in strong generative capacity. For Basic Categorial Grammars (Bar-Hillel 1953) for instance, this follows from the fact that the corresponding class of tree languages is properly contained in the class of context free languages. Tree Adjoing Grammars (Joshi 1985) extend the weak generative capacity of context free grammars and thus cannot be equivalent to $\mathbf{L}\Diamond$-grammars either.

These two results suggest two different agendas for further inquiries. On the level of weak capacity, we may study the impact of interaction postulates for the modalities. It is obvious that already fairly innocent postulates lead us beyond context freeness. Emms 1994 for example proves that the permutation postulate $\Diamond A \bullet B \leftrightarrow B \bullet \Diamond A$ has this effect, and it is easy to show that the distributivity postulates $\Diamond A \bullet B \to \Diamond(A \bullet B)$ and $A \bullet \Diamond B \to \Diamond(A \bullet B)$ together are sufficient to come up with a grammar for $a^n b^m c^n d^m (n, m \geq 1)$.[6] It is an interesting question whether there are natural sets of postulates that lead exactly to the tree adjoining languages or to the context sensitive languages. As for strong generative capacity, one might ask whether there is an independent characterization of the class of $\mathbf{L}\Diamond$-grammars, analogously to the result of Thatcher 1967 for context free languages. Furthermore it is open whether the Basic Categorial Languages are included in the intersection of $\mathbf{L}\Diamond$-languages and context free languages, and if yes, whether or not this inclusion is proper or not. Last but not least one would like to know which interaction postulates are necessary to overcome the boundedness of the height of trees.

References

Bar-Hillel, Y.: 1953, A quasi-arithmetical notation for syntactic description, *Language* 29, 47–58

Buszkowski, W.: 1997, Mathematical linguistics and proof theory, in J. van Benthem and A. ter Meulen (eds.), *Handbook of Logic and Language*, Chapt. 12, pp 683–736, Elesvier, MIT Press

[6]Take $G = \langle\{\langle a, S/\Diamond S\rangle, \langle b, S/D/T\rangle, \langle b, S/D/S\rangle, \langle c, \Diamond T\rangle, \langle c, \Diamond T/T\rangle, \langle d, D\rangle\}.\{S\}\rangle$ and chose s-recognition.

Carpenter, B.: 1995, *The Turing-completeness of multimodal categorial grammars*, unpublished ms.

Cohen, J. M.: 1967, The equivalence of two concepts of categorial grammar, *Information and Control* 10, 475–484

Došen, K.: 1992, A brief survey of frames for the Lambek Calculus, *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik* 38, 179–187

Emms, M.: 1994, Extraction covering extensions of Lambek calculus are not cf, in P. Dekker and M. Stokhof (eds.), *Proceedings of the Ninth Amsterdam Colloquium*, pp 269–286, University of Amsterdam

Joshi, A.: 1985, How much context-sensitivity is necessary for characterizing structural descriptions — tree adjoining grammars, in D. Dowty, L. Karttunen, and A. Zwicky (eds.), *Natural Language Processing. Theoretical, Computational and Psychological Perspectives*, Cambridge University Press, Cambridge, MA

Lambek, J.: 1958, The mathematics of sentence structure, *American Mathematical Monthly* 65, 154–170

Moortgat, M.: 1996, Multimodal linguistic inference, *Journal of Logic, Language and Information* 5(3/4), 349–385

Moortgat, M.: 1997, Categorial type logics, in J. van Benthem and A. ter Meulen (eds.), *Handbook of Logic and Language*, Chapt. 2, pp 93–178, Elesvier, MIT Press

Pentus, M.: 1993, Lambek grammars are context-free, in *Proceedings of the 8th Annual IEEE Symposium on Logic in Computer Science*, Montreal

Roorda, D.: 1991, *Resource logics: Proof-theoretical investigations*, Ph.D. thesis, University of Amsterdam

Thatcher, J. W.: 1967, Characterizing derivation trees of context-free grammars through a generalization of finite automata theory, *Journal of Computer Science* 1, 317–322

Versmissen, J. A. G.: 1996, *Grammatical Composition: Modes, Models, Modalities*, Ph.D. thesis, University of Utrecht