Grammar and Incremental Processing of Dutch Word Order*

Glyn Morrill, Oriol Valentín and Mario Fadda

24th September 2007

1 Lambek calculus L

A *prosodic algebra* for **L** is a free semigroup $(L, +)$. The set $\mathcal{F}$ of *types* of **L** is defined on the basis of a set $\mathcal{A}$ of atomic types as follows:

(1) $\mathcal{F} ::= \mathcal{A} \mid \mathcal{F}\bullet\mathcal{F} \mid \mathcal{F}\backslash\mathcal{F} \mid \mathcal{F}/\mathcal{F}$

A *prosodic interpretation of* **L** is a function $[[\cdot]]$ mapping each type $A \in \mathcal{F}$ into a subset of L such that:

(2)
$$\begin{array}{lcl} [[A\bullet B]] & = & \{s_1{+}s_2 \mid s_1 \in [[A]]\ \&\ s_2 \in [[B]]\} \\ [[A\backslash C]] & = & \{s_2 \mid \forall s_1 \in [[A]], s_1{+}s_2 \in [[C]]\} \\ [[C/B]] & = & \{s_1 \mid \forall s_2 \in [[B]], s_1{+}s_2 \in [[C]]\} \end{array}$$

The set $\mathcal{O}$ of *configurations* of **L** is defined as follows:

(3) $\mathcal{O} ::= \mathcal{F} \mid \mathcal{F}, \mathcal{O}$

We extend the interpretation of types to include configurations as follows:

(4) $[[A, \Gamma]] \;=\; \{s_1{+}s_2 \mid s_1 \in [[A]]\ \&\ s_2 \in [[\Gamma]]\}$

A *sequent* $\Gamma \Rightarrow A$ comprises an *input* configuration Γ and an *output* type A; it is *valid* iff $[[\Gamma]] \subseteq [[A]]$ in every prosodic interpretation. The *sequent calculus* for **L** is as follows, where $\Delta(\Gamma)$ indicates a configuration Δ with a distinguished subconfiguration Γ:

(5)
$$\frac{}{A \Rightarrow A}\,id \qquad \frac{\Gamma \Rightarrow A \qquad \Delta(A) \Rightarrow B}{\Delta(\Gamma) \Rightarrow B}\,Cut$$

$$\frac{\Gamma \Rightarrow A \qquad \Delta(C) \Rightarrow D}{\Delta(\Gamma, A\backslash C) \Rightarrow D}\,\backslash L \qquad \frac{A, \Gamma \Rightarrow C}{\Gamma \Rightarrow A\backslash C}\,\backslash R$$

*Work partially funded by the DGICYT project TIN2005–08832–C03–03 (MOISES-BAR).
Email: morrill@lsi.upc.edu, oriol.valentin@upf.edu, and fadda@lsi.upc.edu. Http: //www-lsi.upc.edu/˜morrill/.

$$\frac{\Gamma \Rightarrow B \qquad \Delta(C) \Rightarrow D}{\Delta(C/B, \Gamma) \Rightarrow D}/L \qquad \frac{\Gamma, B \Rightarrow C}{\Gamma \Rightarrow C/B}/R$$

$$\frac{\Delta(A, B) \Rightarrow D}{\Delta(A{\bullet}B) \Rightarrow D}\bullet L \qquad \frac{\Gamma \Rightarrow A \qquad \Delta \Rightarrow B}{\Gamma, \Delta \Rightarrow A{\bullet}B}\bullet R$$

A *theorem* is a sequent which is derivable in this calculus.

(6) **Proposition** (*soundness of* **L**)

In **L**, every theorem is valid.

Proof. Straightforward induction on the length of proofs. □

(7) **Theorem** (*Cut-elimination for* **L**)

In **L**, every theorem has a Cut-free proof.

Proof. Lambek (1958)[3]. □

(8) **Corollary** (*subformula property for* **L**)

In **L**, every theorem has a proof containing only its subformulas.

Proof. Every rule except Cut has the property that all the types in the premises are either in the conclusion (side formulas) or are the immediate subtypes of the active formula, and Cut itself is eliminable. □

(9) **Corollary** (*decidability of* **L**)

In **L**, it is decidable whether a sequent is a theorem.

Proof. By backward-chaining in the finite Cut-free sequent search space. □

(10) **Theorem** (*completeness of* **L**)

In **L**, every valid sequent is a theorem.

Proof. By the sophisticated reasoning of Pentus (1993)[10], which goes via "quasi-models". □

1.1 Proof nets for L

A *polar type* A^p comprises a type A and a polarity $p = \bullet$ (input) or $\circ$ (output). The *complements* $\overline{\bullet} =_{df.} \circ$ and $\overline{\circ} =_{df.} \bullet$. The *logical links* are as shown in figure 1.

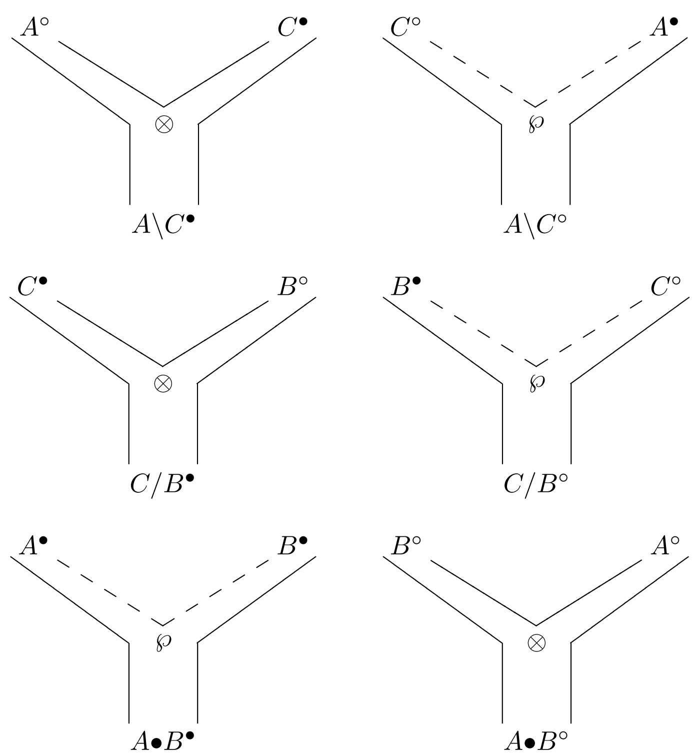
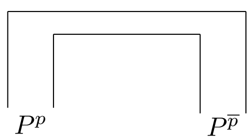


Figure 1: **L** logical links

We refer to edges as *parameter edges* and we refer to sequences of dashed parameter edges as $\forall$-*segments*. We refer to entire roadways seen as single edges as *predicate edges*.

A *polar type tree* is the result of unfolding a polar type up to atomic leaves according to the logical links. A *proof frame* for a sequent $A_0, \ldots, A_n \Rightarrow A$ is the multiset of polar type trees of $A^\circ, {A_1}^\bullet, \ldots, {A_n}^\bullet$. An *axiom link* is as follows, where P is atomic:

(11) 

A *proof structure* is a proof frame to which have been added axiom links connecting each leaf to exactly one other complementary leaf.

A *proof net* is a proof structure satisfying the following correctness criteria:

(12)
- (*Danos-Regnier acyclicity*) Every predicate edge cycle crosses both premise edges of some $\wp$-link.
- ($\forall$-*correctness*) If a parameter path does not form part of a cycle then it does not contain any $\forall$-segment, and every parameter edge cycle contains exactly one $\forall$-segment.

(13) **Theorem** (*soundness and completeness of proof nets*)

A sequent is a theorem iff a proof net can be built on its proof frame.

Proof. Morrill and Fadda (to appear)[7]. $\square$

2 1-Discontinuous Lambek calculus, 1-DLC

A *discontinuous prosodic algebra* is a free algebra $(L, +, 0, 1)$ where $(L, +, 0)$ is a monoid and 1 (the *separator*) is a prime (Morrill 2002)[6]; let $\sigma(s)$ be the number of separators in a prosodic object s. This induces the *1-discontinuous prosodic structure* $(L_0, L_1, +, \times, 0, 1)$ where

(14)
- $L_0 = \{s \in L |\ \sigma(s) = 0\}$
- $L_1 = \{s \in L |\ \sigma(s) = 1\} = L_0 1 L_0$
- $+ : L_i, L_j \to L_{i+j}, i + j \leq 1$
- $\times : L_1, L_j \to L_j, j \leq 1$ is such that $(s_1{+}1{+}s_3){\times}s_2 = s_1{+}s_2{+}s_3$

The sets $\mathcal{F}_0$ and $\mathcal{F}_1$ of *1-discontinuous types* of *sort zero* and *one* are defined on the basis of sets $\mathcal{A}_0$ and $\mathcal{A}_1$ of primitive 1-discontinuous types of sort zero and one as follows:[1]

[1]Sorting for discontinuity was introduced in Morrill and Merenciano (1996)[9].

$$
\begin{array}{rcll}
[[\triangleright A]] & = & \{1{+}s|\ s \in [[A]]\} & \text{right injection}\\
[[\triangleright^{-1} B]] & = & \{s|\ 1{+}s \in [[B]]\} & \text{right projection}\\
[[\triangleleft A]] & = & \{s{+}1|\ s \in [[A]]\} & \text{left injection}\\
[[\triangleleft^{-1} B]] & = & \{s|\ s{+}1 \in [[B]]\} & \text{left projection}\\
[[\hat{\ } A]] & = & \{s_1{+}s_2|\ s_1{+}1{+}s_2 \in [[A]]\} & \text{bridge}\\
[[\check{\ } B]] & = & \{s_1{+}1{+}s_2|\ s \in [[B]]\} & \text{split}\\
[[A{\bullet}B]] & = & \{s_1{+}s_2|\ s_1 \in [[A]]\ \&\ s_2 \in [[B]]\} & \text{(continuous) product}\\
[[A\backslash C]] & = & \{s_2|\ \forall s_1 \in [[A]], s_1{+}s_2 \in [[C]]\} & \text{under}\\
[[C/B]] & = & \{s_1|\ \forall s_2 \in [[B]], s_1{+}s_2 \in [[C]]\} & \text{over}\\
[[A{\odot}B]] & = & \{s_1{+}s_2{+}s_3|\ s_1{+}1{+}s_3 \in [[A]]\ \&\ s_2 \in [[B]]\} & \text{discontinuous product}\\
[[A{\downarrow}C]] & = & \{s_2|\ \forall s_1{+}1{+}s_3 \in [[A]], s_1{+}s_2{+}s_3 \in [[C]]\} & \text{infix}\\
[[C{\uparrow}B]] & = & \{s_1{+}1{+}s_3|\ \forall s_2 \in [[B]], s_1{+}s_2{+}s_3 \in [[C]]\} & \text{extract}
\end{array}
$$

Figure 2: Prosodic interpretation of **1-DLC** types

$$
\begin{array}{llll}
(15) & \mathcal{F}_0 & ::= & \mathcal{A}_0 \mid \triangleright^{-1}\mathcal{F}_1 \mid \triangleleft^{-1}\mathcal{F}_1 \mid \hat{\ }\mathcal{F}_1 \mid \mathcal{F}_0\backslash\mathcal{F}_0 \mid \mathcal{F}_1\backslash\mathcal{F}_1 \mid\\
 & & & \mathcal{F}_0/\mathcal{F}_0 \mid \mathcal{F}_1/\mathcal{F}_1 \mid \mathcal{F}_0{\bullet}\mathcal{F}_0 \mid \mathcal{F}_1{\downarrow}\mathcal{F}_0 \mid \mathcal{F}_1{\odot}\mathcal{F}_0\\
 & \mathcal{F}_1 & ::= & \mathcal{A}_1 \mid \triangleright\mathcal{F}_0 \mid \triangleleft\mathcal{F}_0 \mid \check{\ }\mathcal{F}_0 \mid \mathcal{F}_0\backslash\mathcal{F}_1 \mid \mathcal{F}_1/\mathcal{F}_0 \mid\\
 & & & \mathcal{F}_0{\bullet}\mathcal{F}_1 \mid \mathcal{F}_1{\bullet}\mathcal{F}_0 \mid \mathcal{F}_1{\downarrow}\mathcal{F}_1 \mid \mathcal{F}_0{\uparrow}\mathcal{F}_0 \mid \mathcal{F}_1{\odot}\mathcal{F}_1
\end{array}
$$

A *prosodic interpretation of 1-discontinuous types* is a function $[[\cdot]]$ mapping each type $\mathcal{A}_i \in \mathcal{F}_i$ into a subset of L_i as shown in figure 2.[2]

We give *hypersequent calculus* (not in the sense of A. Avron) for sorted discontinuity (Morrill 1997)[4]. The sets $\mathcal{Q}_0$ and $\mathcal{Q}_1$ of output *figures of sort zero and one of* **1-DLC** are defined as follows:

$$
\begin{array}{llll}
(16) & \mathcal{Q}_0 & ::= & A_0\\
 & \mathcal{Q}_1 & ::= & \sqrt[0]{A_1}, [\,], \sqrt[1]{A_1}
\end{array}
$$

The vectorial notation $\overrightarrow{A}$ refers to the figure of a type A. The sets $\mathcal{O}_0$ and $\mathcal{O}_1$ of input *configurations of sort zero and one of* **1-DLC** are defined as follows:

$$
\begin{array}{llll}
(17) & \mathcal{O}_0 & ::= & \Lambda \mid A_0, \mathcal{O}_0 \mid \sqrt[0]{A_1}, \mathcal{O}_0, \sqrt[1]{A_1}, \mathcal{O}_0\\
 & \mathcal{O}_1 & ::= & \mathcal{O}_0, [\,], \mathcal{O}_0 \mid \mathcal{O}_0, \sqrt[0]{A_1}, \mathcal{O}_1, \sqrt[1]{A_1}, \mathcal{O}_0
\end{array}
$$

Note that figures are "singular" configurations. We define the *components* of a configuration as its maximal substrings not containing $[\,]$. We extend the interpretation of types to include configurations as follows:

$$
\begin{array}{lrcl}
(18) & [[\Lambda]] & = & \{0\}\\
 & [[[\,], \Gamma]] & = & \{1{+}s|\ s \in [[\Gamma]]\}\\
 & [[A, \Gamma]] & = & \{s_1{+}s_2|\ s_1 \in [[A]]\ \&\ s_2 \in [[\Gamma]]\}\\
 & [[\sqrt[0]{A}, \Gamma, \sqrt[1]{A}, \Delta]] & = & \{s_1{+}s_2{+}s_3{+}s_4|\ s_1{+}1{+}s_3 \in [[A]]\ \&\ s_2 \in [[\Gamma]]\ \&\ s_4 \in [[\Delta]]\}
\end{array}
$$

[2]The first type-logical formulations of discontinuous product, infix and extract were made by M. Moortgat. Bridge and split were introduced in Morrill and Merenciano (1996)[9]. Injections and projections are new here.

$$\frac{}{\overrightarrow{A} \Rightarrow \overrightarrow{A}}\,id \qquad \frac{\Gamma \Rightarrow \overrightarrow{A} \qquad \Delta(\overrightarrow{A}) \Rightarrow X}{\Delta(\Gamma) \Rightarrow X}\,Cut$$

$$\frac{\Delta(\sqrt[0]{A}, \Gamma, \sqrt[1]{A}) \Rightarrow X}{\Delta(\Gamma, \triangleright^{-1}A) \Rightarrow X}\,\triangleright^{-1}L \qquad \frac{[\,], \Gamma \Rightarrow \sqrt[0]{A}, [\,], \sqrt[1]{A}}{\Gamma \Rightarrow \triangleright^{-1}A}\,\triangleright^{-1}R$$

$$\frac{\Delta(\Gamma, A) \Rightarrow X}{\Delta(\sqrt[0]{\triangleright A}, \Gamma, \sqrt[1]{\triangleright A}) \Rightarrow X}\,\triangleright L \qquad \frac{\Gamma \Rightarrow A}{[\,], \Gamma \Rightarrow \sqrt[0]{\triangleright A}, [\,], \sqrt[1]{\triangleright A}}\,\triangleright R$$

$$\frac{\Delta(\sqrt[0]{A}, \Gamma, \sqrt[1]{A}) \Rightarrow X}{\Delta(\triangleleft^{-1}A, \Gamma) \Rightarrow X}\,\triangleleft^{-1}L \qquad \frac{\Gamma, [\,] \Rightarrow \sqrt[0]{A}, [\,], \sqrt[1]{A}}{\Gamma \Rightarrow \triangleleft^{-1}A}\,\triangleleft^{-1}R$$

$$\frac{\Delta(A, \Gamma) \Rightarrow X}{\Delta(\sqrt[0]{\triangleleft A}, \Gamma, \sqrt[1]{\triangleleft A}) \Rightarrow X}\,\triangleleft L \qquad \frac{\Gamma \Rightarrow A}{\Gamma, [\,] \Rightarrow \sqrt[0]{\triangleleft A}, [\,], \sqrt[1]{\triangleleft A}}\,\triangleleft R$$

$$\frac{\Delta(B) \Rightarrow X}{\Delta(\sqrt[0]{\check{\ }B}, \sqrt[1]{\check{\ }B}) \Rightarrow X}\,\check{\ }L \qquad \frac{\Gamma(\Lambda) \Rightarrow B}{\Gamma([\,]) \Rightarrow \sqrt[0]{\check{\ }B}, [\,], \sqrt[1]{\check{\ }B}}\,\check{\ }R$$

$$\frac{\Delta(\sqrt[0]{A}, \sqrt[1]{A}) \Rightarrow X}{\Delta(\hat{\ }A) \Rightarrow X}\,\hat{\ }L \qquad \frac{\Gamma([\,]) \Rightarrow \sqrt[0]{A}, [\,], \sqrt[1]{A}}{\Gamma(\Lambda) \Rightarrow \hat{\ }A}\,\hat{\ }R$$

Figure 3: **1-DLC** hypersequent calculus, part I

A *hypersequent* $\Gamma \Rightarrow X$ of sort i comprises an input configuration Γ of sort i and an output figure X of sort i; it is *valid* iff $[[\Gamma]] \subseteq [[X]]$ in every prosodic interpretation. The *hypersequent calculus* for **1-DLC** is as shown in figures 3 and 4 where $\Delta(\Gamma)$ means a configuration Δ in which in some distinguished positions the components of Γ appear in order successively though not necessarily continuously.

(19) **Proposition** (*soundness of* **1-DLC**)

In **1-DLC**, every theorem is valid.

Proof. Easy induction on the length of proofs. □

(20) **Theorem** (*Cut-elimination for* **1-DLC**)

In **1-DLC**, every theorem has a Cut-free proof.

Proof. Essentially Valentín (2006)[12]. □

$$\frac{\Gamma \Rightarrow \overrightarrow{A} \qquad \Delta(\overrightarrow{C}) \Rightarrow X}{\Delta(\Gamma, \overrightarrow{A\backslash C}) \Rightarrow X}\backslash L \qquad \frac{\overrightarrow{A}, \Gamma \Rightarrow \overrightarrow{C}}{\Gamma \Rightarrow \overrightarrow{A\backslash C}}\backslash R$$

$$\frac{\Gamma \Rightarrow \overrightarrow{B} \qquad \Delta(\overrightarrow{C}) \Rightarrow X}{\Delta(\overrightarrow{C/B}, \Gamma) \Rightarrow X}/L \qquad \frac{\Gamma, \overrightarrow{B} \Rightarrow \overrightarrow{C}}{\Gamma \Rightarrow \overrightarrow{C/B}}/R$$

$$\frac{\Delta(\overrightarrow{A}, \overrightarrow{B}) \Rightarrow X}{\Delta(\overrightarrow{A\bullet B}) \Rightarrow X}\bullet L \qquad \frac{\Gamma_1 \Rightarrow \overrightarrow{A} \qquad \Gamma_2 \Rightarrow \overrightarrow{B}}{\Gamma_1, \Gamma_2 \Rightarrow \overrightarrow{A\bullet B}}\bullet R$$

$$\frac{\Gamma_1, [\,], \Gamma_2 \Rightarrow \sqrt[0]{A}, [\,], \sqrt[1]{A} \qquad \Delta(\overrightarrow{C}) \Rightarrow X}{\Delta(\Gamma_1, \overrightarrow{A{\downarrow}C}, \Gamma_2) \Rightarrow X}{\downarrow}L \qquad \frac{\sqrt[0]{A}, \Gamma, \sqrt[1]{A} \Rightarrow \overrightarrow{C}}{\Gamma \Rightarrow \overrightarrow{A{\downarrow}C}}{\downarrow}R$$

$$\frac{\Gamma \Rightarrow \overrightarrow{B} \qquad \Delta(\overrightarrow{C}) \Rightarrow X}{\Delta(\sqrt[0]{C{\uparrow}B}, \Gamma, \sqrt[1]{C{\uparrow}B}) \Rightarrow X}{\uparrow}L \qquad \frac{\Gamma_1, \overrightarrow{B}, \Gamma_2 \Rightarrow \overrightarrow{C}}{\Gamma_1, [\,], \Gamma_2 \Rightarrow \overrightarrow{C{\uparrow}B}}{\uparrow}R$$

$$\frac{\Delta(\sqrt[0]{A}, \overrightarrow{B}, \sqrt[1]{A}) \Rightarrow X}{\Delta(\overrightarrow{A\odot B}) \Rightarrow X}\odot L \qquad \frac{\Gamma_1, [\,], \Gamma_3 \Rightarrow \sqrt[0]{A}, [\,], \sqrt[1]{A} \qquad \Gamma_2 \Rightarrow \overrightarrow{B}}{\Gamma_1, \Gamma_2, \Gamma_3 \Rightarrow \overrightarrow{A\odot B}}\odot R$$

Figure 4: **1-DLC** hypersequent calculus, part II

(21) **Corollary** (*subformula property for* **1-DLC**)

In **1-DLC**, every theorem has a proof containing only its subformulas.

Proof. Every rule except Cut has the property that all the types in the premises are either in the conclusion (side formulas) or are the immediate subtypes of the active formula, and Cut itself is eliminable. □

(22) **Corollary** (*decidability of* **1-DLC**)

In **1-DLC**, it is decidable whether a sequent is a theorem.

Proof. By backward-chaining in the finite Cut-free sequent search space. □

(23) **Conjecture** (*completeness of* **1-DLC**)

In **1-DLC**, every valid sequent is a theorem.

Proof. We think the reasoning of Pentus (1993)[10] can be replicated. For some results see Valentín (2006)[12]. □

2.1 Prospects for proof nets for 1-DLC

Morrill and Fadda (to appear)[7] give proof nets for a subsystem of **1-DLC** called *basic discontinuous Lambek calculus*, **BDLC**. **BDLC** has only functionalities $+ : L_0, L_0 \to L_0$ and $\times : L_1, L_0 \to L_0$, and has no unary connectives. The logical links for the discontinuous connectives of that subsystem are exemplified in figure 5. For **BDLC** we need to augment the correctness criteria (12) of **L** with:

(24) (*Input-acyclicity*) No parameter edge cycle goes through both the start and the end points of any input type of sort zero.

It is an open question whether this characterisation remains adequate for the more polymorphic **1-DLC** binary connectives. It is an even more open problem how to formulate proof nets for the **1-DLC** unary connectives. The difficulty is that they are akin to units, for which it has been found difficult to give proof nets.

3 Dutch word order

Subordinate clauses are verb final:

(25)

(. . . dat)	Jan	boeken	las		
(. . . that)	J.	books	reads		
CP/S	N	N	N\(N\S)	⇒	CP

‘(. . . that) Jan reads books’

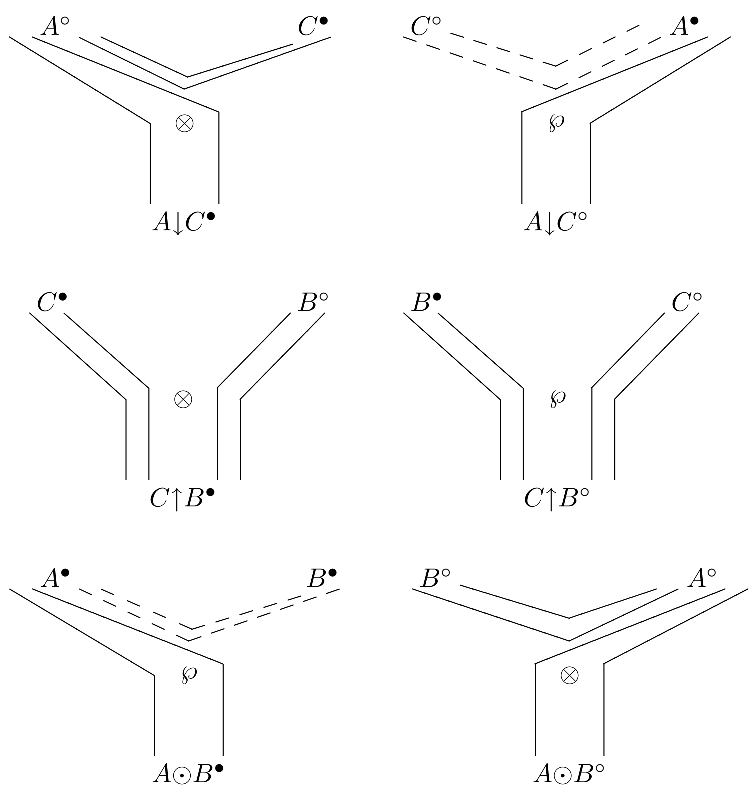


Figure 5: **BDLC** discontinuous logical links

Modals and control verbs, so-called verb raising triggers, appear in a verb final verb cluster with the English word order:

(26)	(… dat)	Jan	boeken	kan	lezen		
	(… that)	J.	books	is able	read		
	CP/S	N	N	$(N\backslash Si){\downarrow}(N\backslash S)$	$\triangleright^{-1}(N\backslash(N\backslash Si))$	$\Rightarrow$	CP
	'(… that) Jan is able to read books'						

(27)	(… dat)	Jan	boeken	wil	kunnen
	(… that)	J.	books	wants	be able
	CP/S	N	N	$(N\backslash Si){\downarrow}(N\backslash S)$	$\triangleright^{-1}((N\backslash Si){\downarrow}(N\backslash Si))$
	lezen				
	read				
	$\triangleright^{-1}(N\backslash(N\backslash Si))$ $\Rightarrow$ CP				
	'(… that) Jan wants to be able to read books'				

When the infinitival complement verbs also take objects, cross-serial dependencies are generated. Calcagno (1995)[1] provides an analysis of cross-serial dependencies which is a close precedent to ours, but in terms of categorial head-wrapping of headed strings, rather than wrapping of separated strings.

(28)	(… dat)	Jan	Cecilia$_1$	Henk$_2$	de	nijlpaarden$_3$
	(… that)	J.	C.	H.	the	hippos
	CP/S	N	N	N	N/CN	CN
	zag$_1$	helpen$_2$	voeren$_3$			
	saw	help	feed			
	$(N\backslash Si){\downarrow}(N\backslash(N\backslash S))$	$\triangleright^{-1}((N\backslash Si){\downarrow}(N\backslash(N\backslash Si)))$	$\triangleright^{-1}(N\backslash(N\backslash Si))$	$\Rightarrow$	CP	
	'(… that) Jan saw$_1$ Cecilia$_1$ help$_2$ Henk$_2$ feed$_3$ the hippos$_3$'					

(29) 'An increasing load in processing makes such multiple embeddings increasingly unacceptable.' [Steedman (1985)[11], fn. 29, p.546]

Main clause yes/no interrogative word order, V1, is derived from subordinate clause word order by fronting the finite verb. We therefore propose a lexical rule mapping (subordinate clause) finite verb types V to $Q/\hat{}(S{\uparrow}V)$, cf. Hepple (1990)[2].

(30)	Wil	Jan	boeken	lezen?		
	wants	J.	books	read		
	$Q/\hat{}(S{\uparrow}((N\backslash Si){\downarrow}(N\backslash S)))$	N	N	$\triangleright^{-1}(N\backslash(N\backslash Si))$	$\Rightarrow$	Q
	'Does Jan want to read books?'					

Main clause declarative word order, V2, is further derived from V1 by fronting a major constituent. We propose to achieve this by allowing complex distinguished types (cf. Morrill and Gavarró 1992)[8].

(31)	Jan	wil	boeken	lezen.		
	J.	wants	books	read		
	N	$Q/\hat{}(S{\uparrow}((N\backslash Si){\downarrow}(N\backslash S)))$	N	$\triangleright^{-1}(N\backslash(N\backslash Si))$	$\Rightarrow$	$N{\bullet}\hat{}(Q{\uparrow}N)$
	'Jan wants to read books.'					

$$\dfrac{N \Rightarrow N \qquad \dfrac{\dfrac{N \Rightarrow N \qquad \dfrac{\dfrac{\dfrac{N \Rightarrow N \qquad \dfrac{\sqrt[0]{VPi}, [\,], \sqrt[1]{VPi} \Rightarrow \sqrt[0]{VPi}, [\,], \sqrt[1]{VPi} \qquad N, VP \Rightarrow S}{N, \sqrt[0]{VPi}, VPi{\downarrow}VP, \sqrt[1]{VPi} \Rightarrow S}{\downarrow}L}{N, N, \sqrt[0]{N\backslash VPi}, VPi{\downarrow}VP, \sqrt[1]{N\backslash VPi} \Rightarrow S}\backslash L}{N, N, VPi{\downarrow}VP, \triangleright^{-1}(N\backslash VPi) \Rightarrow S}\triangleright^{-1}L}{\dfrac{N, N, [\,], \triangleright^{-1}(N\backslash VPi) \Rightarrow \sqrt[0]{S{\uparrow}(VPi{\downarrow}VP)}, [\,], \sqrt[1]{S{\uparrow}(VPi{\downarrow}VP)}}{N, N, \triangleright^{-1}(N\backslash VPi) \Rightarrow {}^\wedge(S{\uparrow}(VPi{\downarrow}VP))}{}^\wedge R}{\uparrow}R \qquad Q \Rightarrow Q}{\dfrac{\dfrac{Q/{}^\wedge(S{\uparrow}(VPi{\downarrow}VP)), N, N, \triangleright^{-1}(N\backslash VPi) \Rightarrow Q}{Q/{}^\wedge(S{\uparrow}(VPi{\downarrow}VP)), [\,], N, \triangleright^{-1}(N\backslash VPi) \Rightarrow \sqrt[0]{Q{\uparrow}N}, [\,], \sqrt[1]{Q{\uparrow}N}}{\uparrow}R}{Q/{}^\wedge(S{\uparrow}(VPi{\downarrow}VP)), N, \triangleright^{-1}(N\backslash VPi) \Rightarrow {}^\wedge(Q{\uparrow}N)}{}^\wedge R}/L}{N, Q/{}^\wedge(S{\uparrow}(VPi{\downarrow}VP)), N, \triangleright^{-1}(N\backslash VPi) \Rightarrow N{\bullet}{}^\wedge(Q{\uparrow}N)}{\bullet}R$$

Figure 6: Hypersequent derivation of *Jan wil boeken lezen*.

4 Analyses of Dutch

A hypersequent calculus derivation of *Jan wil boeken lezen* is given in figure 6, where here and henceforth VP abbreviates N\S and VPi abbreviates N\Si. The proof net syntactic structure for *Jan wil boeken lezen* is given in figure 7. The complexity profile (Morrill 2000)[5] is as follows:

(32)

3	a						a		
2			a		a				
1									
0									a
		Jan		wil		boeken		lezen	

A hypersequent calculus derivation of *Marie zegt dat Jan Cecilia Henk de nijlpaarden zag helpen voeren* ('Marie says that Jan saw Cecilia help Henk feed the hippos') is given in figure 8. The syntactic structure for *Marie zegt dat Jan Cecilia Henk de nijlpaarden zag helpen voeren* is given in figures 9 and 10. The complexity profile is as follows:

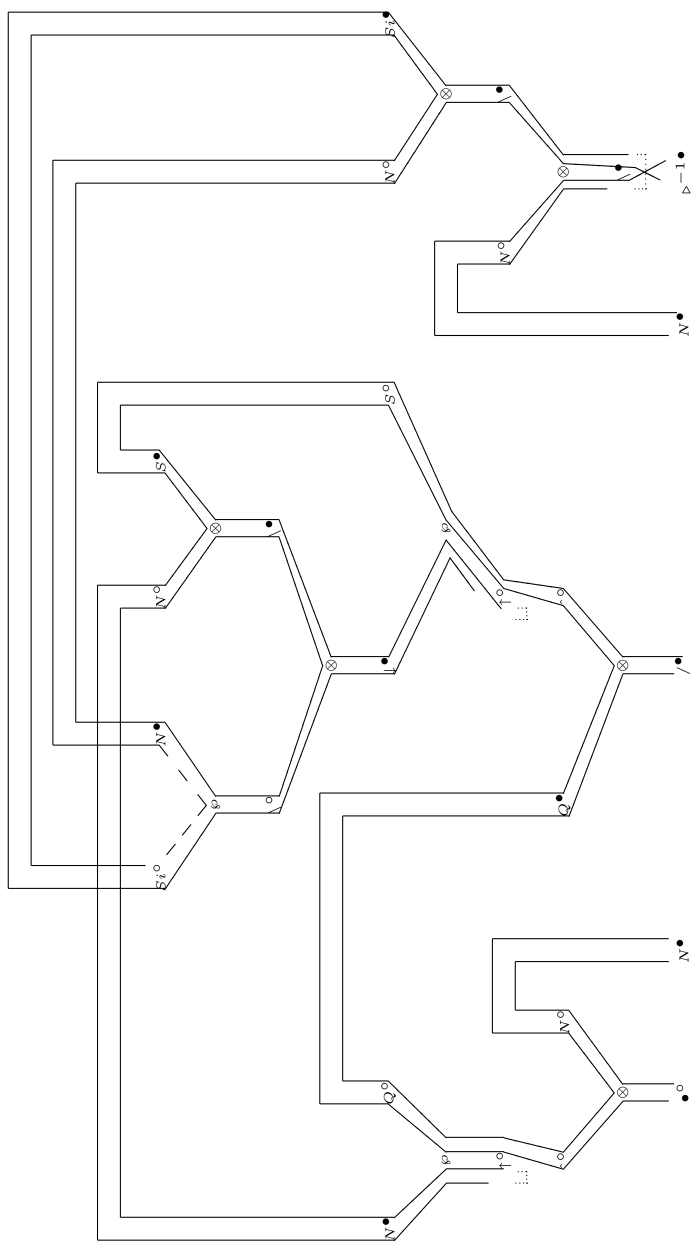


Figure 7: Proof net syntactic structure for *Jan wil boeken lezen*

(33)

6												
5												
4												
3	a										a	
2			a						a			
1					a		a					
0												
		Marie		zegt		dat		Jan		Cecilia		Henk

6			a								
5					a						
4	a						a				
3									a		
2											
1											
											a
		de		nijlpaarden		zag		helpen		voeren	

References

[1] Mike Calcagno. A Sign-Based Extension to the Lambek Calculus for Discontinuous Constituency. *Bulletin of the IGPL*, 3(4):555–578, 1995.

[2] Mark Hepple. *The Grammar and Processing of Order and Dependency.* PhD thesis, University of Edinburgh, 1990.

[3] Joachim Lambek. The mathematics of sentence structure. *American Mathematical Monthly*, 65:154–170, 1958. Reprinted in Buszkowski, W., W. Marciszewski, and J. van Benthem, editors, 1988, *Categorial Grammar*, Linguistic & Literary Studies in Eastern Europe volume 25, John Benjamins, Amsterdam, 153–172.

[4] Glyn Morrill. Proof Syntax of Discontinuity. In Paul Dekker, Martin Stokhof, and Yde Venema, editors, *Proceedings of the 11th Amsterdam Colloquium*, pages 235–240, Universiteit van Amsterdam, 1997. Institute for Logic, Language and Computation, ILLC.

[5] Glyn Morrill. Incremental Processing and Acceptability. *Computational Linguistics*, 26(3):319–338, 2000.

[6] Glyn Morrill. Towards Generalised Discontinuity. In Gerhard Jäger, Paula Monachesi, Gerald Penn, and Shuly Wintner, editors, *Proceedings of the 7th Conference on Formal Grammar*, pages 103–111, Trento, 2002. ESSLLI.

[7] Glyn Morrill and Mario Fadda. Proof Nets for Basic Discontinuous Lambek Calculus. *Logic and Computation*, to appear.

[8] Glyn Morrill and Anna Gavarró. Catalan Clitics. In Alain Lecomte, editor, *Word Order in Categorial Grammar / L'Ordre des mots dans les grammaires catégorielles*, pages 211–232. Édicions Adosa, Clermont-Ferrand, 1992.

[9] Glyn Morrill and Josep-Maria Merenciano. Generalising discontinuity. *traitement automatique des langues*, 37(2):119–143, 1996.

[10] M. Pentus. Lambek calculus is L-complete. *ILLC Report, University of Amsterdam*, 1993. Shortened version published as Language completeness of the Lambek calculus, *Proceedings of the Ninth Annual IEEE Symposium on Logic in Computer Science*, Paris, pages 487–496, 1994.

[11] Mark Steedman. Dependency and Coordination in the Grammar of Dutch and English. *Language*, 61:523–568, 1985.

[12] Oriol Valentín. 1-Discontinuous Lambek Calculus: Type Logical Grammar and discontinuity in natural language. *DEA dissertation, Universitat Autònoma de Barcelona*, 2006. http://seneca.uab.es/ggt/tesis.htm.

$$\frac{N \Rightarrow N \qquad N, VP/CP, CP/S, N, VP \Rightarrow S}{N, VP/CP, CP/S, N, N, N\backslash VP \Rightarrow S}\ \backslash L$$

$$\frac{\sqrt[0]{VPi}, [\,], \sqrt[1]{VPi} \Rightarrow \sqrt[0]{VPi}, [\,], \sqrt[1]{VPi} \qquad N, VP/CP, CP/S, N, N, N\backslash VP \Rightarrow S}{N, VP/CP, CP/S, N, N, \sqrt[0]{VPi}, VPi{\downarrow}(N\backslash VP), \sqrt[1]{VPi} \Rightarrow S}\ {\downarrow}L$$

$$\frac{N \Rightarrow N \qquad N, VP/CP, CP/S, N, N, \sqrt[0]{VPi}, VPi{\downarrow}(N\backslash VP), \sqrt[1]{VPi} \Rightarrow S}{N, VP/CP, CP/S, N, N, N, \sqrt[0]{N\backslash VPi}, VPi{\downarrow}(N\backslash VP), \sqrt[1]{N\backslash VPi} \Rightarrow S}\ \backslash L$$

$$\frac{\sqrt[0]{VPi}, [\,], \sqrt[1]{VPi} \Rightarrow \sqrt[0]{VPi}, [\,], \sqrt[1]{VPi} \qquad N, VP/CP, CP/S, N, N, N, \sqrt[0]{N\backslash VPi}, VPi{\downarrow}(N\backslash VP), \sqrt[1]{N\backslash VPi} \Rightarrow S}{N, VP/CP, CP/S, N, N, N, \sqrt[0]{VPi}, \sqrt[0]{VPi{\downarrow}(N\backslash VPi)}, VPi{\downarrow}(N\backslash VP), \sqrt[1]{VPi{\downarrow}(N\backslash VPi)}, \sqrt[1]{VPi} \Rightarrow S}\ {\downarrow}L$$

$$\frac{N, VP/CP, CP/S, N, N, N, \sqrt[0]{VPi}, \sqrt[0]{VPi{\downarrow}(N\backslash VPi)}, VPi{\downarrow}(N\backslash VP), \sqrt[1]{VPi{\downarrow}(N\backslash VPi)}, \sqrt[1]{VPi} \Rightarrow S}{N, VP/CP, CP/S, N, N, N, \sqrt[0]{VPi}, VPi{\downarrow}(N\backslash VP), \triangleright^{-1}(VPi{\downarrow}(N\backslash VPi)), \sqrt[1]{VPi} \Rightarrow S}\ \triangleright^{-1}L$$

$$\frac{N \Rightarrow N \qquad N, VP/CP, CP/S, N, N, N, \sqrt[0]{VPi}, VPi{\downarrow}(N\backslash VP), \triangleright^{-1}(VPi{\downarrow}(N\backslash VPi)), \sqrt[1]{VPi} \Rightarrow S}{N, VP/CP, CP/S, N, N, N, N, \sqrt[0]{N\backslash VPi}, VPi{\downarrow}(N\backslash VP), \triangleright^{-1}(VPi{\downarrow}(N\backslash VPi)), \sqrt[1]{N\backslash VPi} \Rightarrow S}\ \backslash L$$

$$\frac{N, VP/CP, CP/S, N, N, N, N, \sqrt[0]{N\backslash VPi}, VPi{\downarrow}(N\backslash VP), \triangleright^{-1}(VPi{\downarrow}(N\backslash VPi)), \sqrt[1]{N\backslash VPi} \Rightarrow S}{N, VP/CP, CP/S, N, N, N, N, VPi{\downarrow}(N\backslash VP), \triangleright^{-1}(VPi{\downarrow}(N\backslash VPi)), \triangleright^{-1}(N\backslash VPi) \Rightarrow S}\ \triangleright^{-1}L$$

$$\frac{N, VP/CP, CP/S, N, N, N, N, VPi{\downarrow}(N\backslash VP), \triangleright^{-1}(VPi{\downarrow}(N\backslash VPi)), \triangleright^{-1}(N\backslash VPi) \Rightarrow S}{N, [\,], CP/S, N, N, N, N, VPi{\downarrow}(N\backslash VP), \triangleright^{-1}(VPi{\downarrow}(N\backslash VPi)), \triangleright^{-1}(N\backslash VPi) \Rightarrow \sqrt[0]{S{\uparrow}(VP/CP)}, [\,], \sqrt[1]{S{\uparrow}(VP/CP)}}\ {\uparrow}R$$

$$\frac{N, [\,], CP/S, N, N, N, N, VPi{\downarrow}(N\backslash VP), \triangleright^{-1}(VPi{\downarrow}(N\backslash VPi)), \triangleright^{-1}(N\backslash VPi) \Rightarrow \sqrt[0]{S{\uparrow}(VP/CP)}, [\,], \sqrt[1]{S{\uparrow}(VP/CP)}}{N, CP/S, N, N, N, N, VPi{\downarrow}(N\backslash VP), \triangleright^{-1}(VPi{\downarrow}(N\backslash VPi)), \triangleright^{-1}(N\backslash VPi) \Rightarrow \hat{}(S{\uparrow}(VP/CP))}\ \hat{}R$$

$$\frac{N, CP/S, N, N, N, N, VPi{\downarrow}(N\backslash VP), \triangleright^{-1}(VPi{\downarrow}(N\backslash VPi)), \triangleright^{-1}(N\backslash VPi) \Rightarrow \hat{}(S{\uparrow}(VP/CP)) \qquad Q \Rightarrow Q}{Q/\hat{}(S{\uparrow}(VP/CP)), N, CP/S, N, N, N, N, VPi{\downarrow}(N\backslash VP), \triangleright^{-1}(VPi{\downarrow}(N\backslash VPi)), \triangleright^{-1}(N\backslash VPi) \Rightarrow Q}\ /L$$

$$\frac{Q/\hat{}(S{\uparrow}(VP/CP)), N, CP/S, N, N, N, N, VPi{\downarrow}(N\backslash VP), \triangleright^{-1}(VPi{\downarrow}(N\backslash VPi)), \triangleright^{-1}(N\backslash VPi) \Rightarrow Q}{Q/\hat{}(S{\uparrow}(VP/CP)), [\,], CP/S, N, N, N, N, VPi{\downarrow}(N\backslash VP), \triangleright^{-1}(VPi{\downarrow}(N\backslash VPi)), \triangleright^{-1}(N\backslash VPi) \Rightarrow \sqrt[0]{Q{\uparrow}N}, [\,], \sqrt[1]{Q{\uparrow}N}}\ {\uparrow}R$$

$$\frac{Q/\hat{}(S{\uparrow}(VP/CP)), [\,], CP/S, N, N, N, N, VPi{\downarrow}(N\backslash VP), \triangleright^{-1}(VPi{\downarrow}(N\backslash VPi)), \triangleright^{-1}(N\backslash VPi) \Rightarrow \sqrt[0]{Q{\uparrow}N}, [\,], \sqrt[1]{Q{\uparrow}N}}{Q/\hat{}(S{\uparrow}(VP/CP)), CP/S, N, N, N, N, VPi{\downarrow}(N\backslash VP), \triangleright^{-1}(VPi{\downarrow}(N\backslash VPi)), \triangleright^{-1}(N\backslash VPi) \Rightarrow \hat{}(Q{\uparrow}N)}\ \hat{}R$$

$$\frac{N \Rightarrow N \qquad Q/\hat{}(S{\uparrow}(VP/CP)), CP/S, N, N, N, N, VPi{\downarrow}(N\backslash VP), \triangleright^{-1}(VPi{\downarrow}(N\backslash VPi)), \triangleright^{-1}(N\backslash VPi) \Rightarrow \hat{}(Q{\uparrow}N)}{N, Q/\hat{}(S{\uparrow}(VP/CP)), CP/S, N, N, N, N, VPi{\downarrow}(N\backslash VP), \triangleright^{-1}(VPi{\downarrow}(N\backslash VPi)), \triangleright^{-1}(N\backslash VPi) \Rightarrow N\bullet\hat{}(Q{\uparrow}N)}\ \bullet R$$

Figure 8: Hypersequent derivation of *Marie zegt dat Jan Cecilia Henk de nijlpaarden zag helpen voeren*

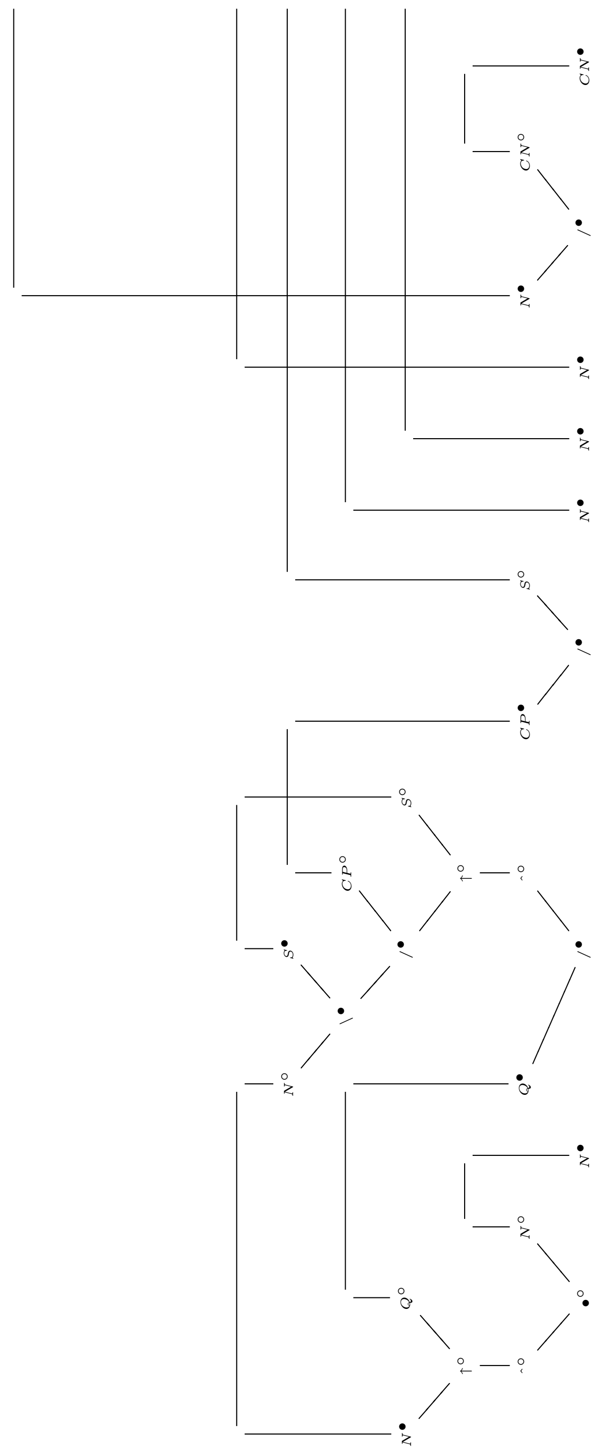


Figure 9: Syntactic structure for *Marie zegt dat Jan Cecilia Henk de nijlpaarden zag helpen voeren*, part I

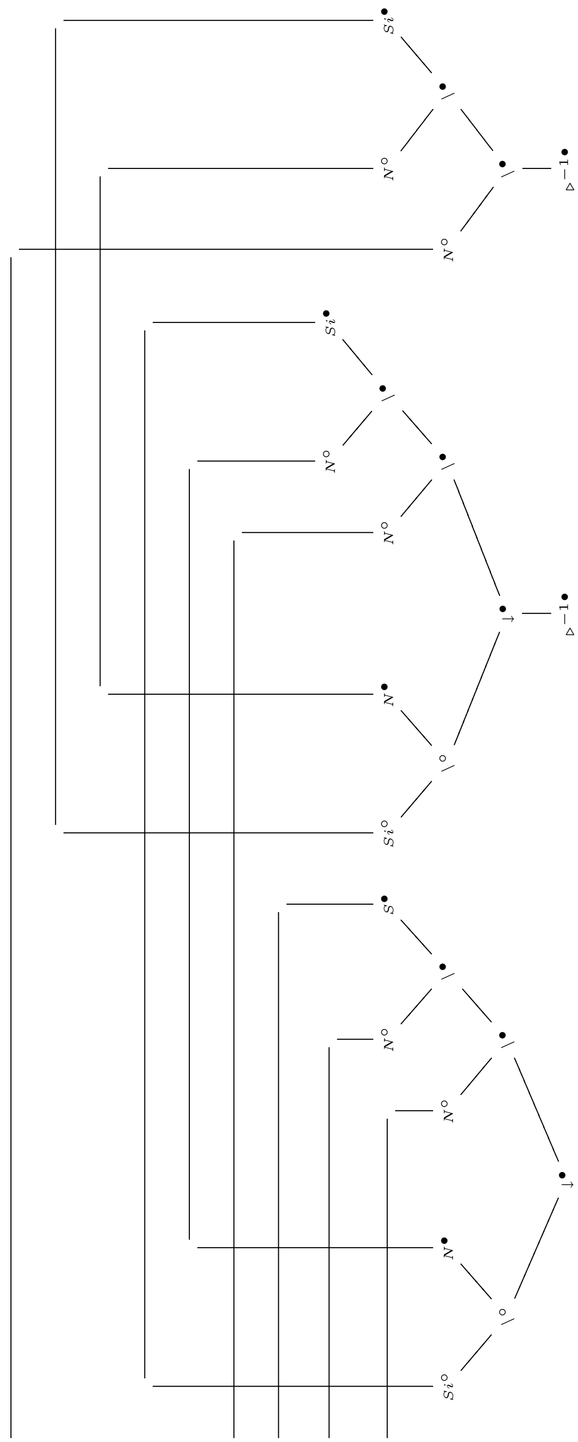


Figure 10: Syntactic structure for *Marie zegt dat Jan Cecilia Henk de nijlpaarden zag helpen voeren*, part II