Proof Nets for the Multimodal Lambek Calculus

Quintijn Puite* Richard Moot†

March 29, 1999

*puite@math.uu.nl URL: http://www.math.uu.nl/people/puite/
†moot@let.uu.nl URL: http://www.let.uu.nl/~Richard.Moot/

Contents

1 Introduction

Since the introduction of proof nets as an elegant proof theory for the multiplicative fragment of linear logic in [Girard 87], a number of attempts have been made to adapt this proof theory to a variety of Lambek Calculi, as shown by work from e.g. [Roorda 91], [Morrill 96] and [Moortgat 97].

In this paper we will present a new way to look at proof nets for the multimodal Lambek Calculus. We will show how we can uniformly handle both the unary and the binary connectives and how we have a natural correctness criterion for the base logic $\mathbf{NL}\diamond$ together with a set $\mathcal{R}$ of structural rules subject to a linearity condition.

First, we introduce proof structures for our calculus. Following [Puite 98] we will allow proof structures to have hypotheses in addition to conclusions. Then we will look at slightly more abstract graphs, which we will call hypothesis structures, on which we will formulate a correctness criterion in the form of graph conversions. Proof nets will be those proof structures of which the hypothesis structure converts to a tree.

As our main result we will prove our proof net calculus is sound and complete with respect to the sequent calculus. In the following sections we will sketch a proof of cut elimination, show applications of our calculus to automated deduction and look at how to adapt our correctness criterion to the Lambek Calculus $\mathbf{L}$, a special case of $\mathbf{NL}\diamond_{\mathcal{R}}$ for which we formulate an alternative correctness criterion.

We will conclude with comparing our work to some related proposals and addressing a number of open problems.

2 The calculus $\mathbf{NL}\Diamond_{\mathcal{R}}$

Starting from a set of atoms $\{p_1, p_2, \ldots\}$, the *formulas* of the multimodal Lambek Calculus with $\Diamond$ ($\mathbf{NL}\Diamond$) are built up with the unary connectives $\Diamond_j$ and $\Box^{\downarrow}{}_j$ and with the binary connectives[1] $\bullet_i$, $\backslash_i$ and $/_i$, where j and i vary over given fixed finite sets of *modes* J respectively I.

Structure trees are built up from formulas with unary constructors $\langle - \rangle^j$ and binary constructors $- \circ_i -$, where again j and i vary over the modes. Derivable objects are *sequents* $\Gamma \vdash C$ in which the antecedent part is a structure tree and in which the succedent part is a formula.

An *n-ary structural rule*

$$\frac{\Delta[\,\Xi[\,\Gamma_1, \ldots, \Gamma_n\,]\,] \vdash C}{\Delta[\,\Xi'[\,\Gamma_{\pi_1}, \ldots, \Gamma_{\pi_n}\,]\,] \vdash C}\ \langle\Xi, \Xi', \pi\rangle$$

is defined by a pair of formal trees[2] Ξ, Ξ' of length n and a rearrangement of the variables $\pi \in \mathfrak{S}_n$, the symmetric group of degree n. Observe that every subtree Γ_k occurs exactly once in both the upper and lower sequent of the inference, whence any non-linear structural rule like

$$\frac{\Delta[\,\Gamma_1 \circ_i \Gamma_1\,] \vdash C}{\Delta[\,\Gamma_1\,] \vdash C}\ \textsc{Contraction}_i \qquad \frac{\Delta[\,\Gamma_1\,] \vdash C}{\Delta[\,\Gamma_2 \circ_i \Gamma_1\,] \vdash C}\ \textsc{LWeakening}_i \qquad \frac{\Delta[\,\Gamma_1\,] \vdash C}{\Delta[\,\Gamma_1 \circ_i \Gamma_2\,] \vdash C}\ \textsc{RWeakening}_i$$

does not conform to this definition. Neither does the following rule, though it is linear:

$$\frac{\Delta[\,\Gamma_1\,] \circ_i \Gamma_2 \vdash C}{\Delta[\,\Gamma_2\,] \circ_i \Gamma_1 \vdash C}$$

However, it may be admissible, depending on $\mathcal{R}$.

Given a set $\mathcal{R}$ of structural rules, we define the sequent calculus $\mathbf{NL}\Diamond_{\mathcal{R}}$ by the inference rules of figure 1. From each n-ary structural rule

$$\frac{\Delta[\,\Xi[\,\Gamma_1, \ldots, \Gamma_n\,]\,] \vdash C}{\Delta[\,\Xi'[\,\Gamma_{\pi_1}, \ldots, \Gamma_{\pi_n}\,]\,] \vdash C}\ \langle\Xi, \Xi', \pi\rangle$$

we can derive — for every n-tuple of formulas $A_1, \ldots, A_n$ — the sequent $\Xi'[\,A_{\pi_1}, \ldots, A_{\pi_n}\,]^\bullet \vdash \Xi[\,A_1, \ldots, A_n\,]^\bullet$, where by $\Gamma^\bullet$ we mean the formula obtained from Γ by replacing all $\langle\,\rangle^j$- and $\circ_i$ -occurrences by $\Diamond_j$- and $\bullet_i$-occurrences respectively. This means that the axiom rule

$$\frac{}{\Xi'[\,A_{\pi_1}, \ldots, A_{\pi_n}\,]^\bullet \vdash \Xi[\,A_1, \ldots, A_n\,]^\bullet}$$

is admissible. In fact, adding the structural rule to the calculus is equivalent to adding (all instances of) the corresponding axiom rule to the calculus.

Let $\mathcal{R}_{\max}$ be the following set of structural rules, where j, i and i' vary over the modes:

[1] The "working linear logician" may prefer the symbols $\otimes_i$, $\multimap_i$ and $\circ\!\!-_i$ for the respective binary connectives.
[2] Precise definitions may be found in the appendix.

Identity rules

$$\frac{}{A \vdash A}\ \textsc{Ax}$$

$$\frac{\Gamma \vdash A \qquad \Delta[A] \vdash C}{\Delta[\Gamma] \vdash C}\ \textsc{Cut}$$

Logical rules for the $\otimes$-like connectives

$$\frac{\Gamma[\langle A \rangle^j] \vdash C}{\Gamma[\Diamond_j A] \vdash C}\ \textsc{L}\Diamond_j \qquad \frac{\Gamma \vdash A}{\langle \Gamma \rangle^j \vdash \Diamond_j A}\ \textsc{R}\Diamond_j$$

$$\frac{\Gamma[A \circ_i B] \vdash C}{\Gamma[A \bullet_i B] \vdash C}\ \textsc{L}\bullet_i \qquad \frac{\Gamma \vdash A \qquad \Delta \vdash B}{\Gamma \circ_i \Delta \vdash A \bullet_i B}\ \textsc{R}\bullet_i$$

Logical rules for the $\parr$-like connectives

$$\frac{\Delta[B] \vdash C}{\Delta[\langle \Box^{\downarrow}_j B \rangle^j] \vdash C}\ \textsc{L}\Box^{\downarrow}_j \qquad \frac{\langle \Gamma \rangle^j \vdash B}{\Gamma \vdash \Box^{\downarrow}_j B}\ \textsc{R}\Box^{\downarrow}_j$$

$$\frac{\Gamma \vdash A \qquad \Delta[B] \vdash C}{\Delta[\Gamma \circ_i A \backslash_i B] \vdash C}\ \textsc{L}\backslash_i \qquad \frac{A \circ_i \Gamma \vdash B}{\Gamma \vdash A \backslash_i B}\ \textsc{R}\backslash_i$$

$$\frac{\Gamma \vdash A \qquad \Delta[B] \vdash C}{\Delta[B /_i A \circ_i \Gamma] \vdash C}\ \textsc{L}/_i \qquad \frac{\Gamma \circ_i A \vdash B}{\Gamma \vdash B /_i A}\ \textsc{R}/_i$$

Structural rules (for all $\langle \Xi, \Xi', \pi \rangle \in \mathcal{R}$)

$$\frac{\Delta[\,\Xi[\,\Gamma_1, \ldots, \Gamma_n\,]\,] \vdash C}{\Delta[\,\Xi'[\,\Gamma_{\pi_1}, \ldots, \Gamma_{\pi_n}\,]\,] \vdash C}\ \langle \Xi, \Xi', \pi \rangle$$

Figure 1: The sequent calculus $\mathbf{NL}\Diamond_{\mathcal{R}}$

$$\frac{\Delta[\,\Gamma_1\,] \vdash C}{\Delta[\,\langle\,\Gamma_1\,\rangle^j\,] \vdash C}\,[\mathrm{LTriv}_j] \qquad \frac{\Delta[\,\langle\,\Gamma_1\,\rangle^j\,] \vdash C}{\Delta[\,\Gamma_1\,] \vdash C}\,[\mathrm{RTriv}_j]$$

$$\frac{\Delta[\,\Gamma_1 \;\circ_i\; (\Gamma_2 \;\circ_i\; \Gamma_3)\,] \vdash C}{\Delta[\,(\Gamma_1 \;\circ_i\; \Gamma_2) \;\circ_i\; \Gamma_3\,] \vdash C}\,[\mathrm{LAss}_i] \qquad \frac{\Delta[\,(\Gamma_1 \;\circ_i\; \Gamma_2) \;\circ_i\; \Gamma_3\,] \vdash C}{\Delta[\,\Gamma_1 \;\circ_i\; (\Gamma_2 \;\circ_i\; \Gamma_3)\,] \vdash C}\,[\mathrm{RAss}_i]$$

$$\frac{\Delta[\,\Gamma_1 \;\circ_i\; \Gamma_2\,] \vdash C}{\Delta[\,\Gamma_2 \;\circ_i\; \Gamma_1\,] \vdash C}\,[\mathrm{Com}_i] \qquad \frac{\Delta[\,\Gamma_1 \;\circ_i\; \Gamma_2\,] \vdash C}{\Delta[\,\Gamma_1 \;\circ_{i'}\; \Gamma_2\,] \vdash C}\,[\mathrm{Eq}_{i,i'}]$$

Lemma 2.1 Any possible structural rule is admissible in $\mathbf{NL}\diamond_{\mathcal{R}_{\max}}$. ♦

Let $\lceil - \rceil$ be the following translation from $\mathbf{NL}\diamond$ formulas to formulas of intuitionistic multiplicative linear logic (**iMLL**), deleting the unary connectives and the mode indices and identifying both implications:

$$\begin{aligned} \lceil p_k \rceil &:= p_k & \lceil A \bullet_i B \rceil &:= \lceil A \rceil \otimes \lceil B \rceil \\ \lceil \diamond_j A \rceil &:= \lceil A \rceil & \lceil A \backslash_i B \rceil &:= \lceil A \rceil \multimap \lceil B \rceil \\ \lceil \Box^{\downarrow}{}_j A \rceil &:= \lceil A \rceil & \lceil B /_i A \rceil &:= \lceil A \rceil \multimap \lceil B \rceil \end{aligned}$$

For any structure tree Γ, let $\|\Gamma\|$ be the multiset of elements in Γ. We write $\lceil \|\Gamma\| \rceil$ for the multiset $\{\lceil A \rceil \mid A \in \|\Gamma\|\}$. Let $\mathbf{iMLL}^{>0}$ stand for the calculus **iMLL** restricted to the requirement that the antecedent multiset of all sequents in a derivation be non-empty.

Corollary 2.2 The following maps between collections of sequents:

$$\mathcal{SEQ}(\mathbf{NL}\diamond) \;=\; \mathcal{SEQ}(\mathbf{NL}\diamond_{\mathcal{R}}) \;=\; \mathcal{SEQ}(\mathbf{NL}\diamond_{\mathcal{R}_{\max}}) \;\overset{\lceil\|-\|\rceil}{\to}\; \mathcal{SEQ}(\mathbf{iMLL}^{>0}) \;=\; \mathcal{SEQ}(\mathbf{iMLL})$$

restrict to the collections of derivable sequents:

$$\mathcal{D}\mathcal{SEQ}(\mathbf{NL}\diamond) \hookrightarrow \mathcal{D}\mathcal{SEQ}(\mathbf{NL}\diamond_{\mathcal{R}}) \hookrightarrow \mathcal{D}\mathcal{SEQ}(\mathbf{NL}\diamond_{\mathcal{R}_{\max}}) \;\overset{\lceil\|-\|\rceil}{\twoheadrightarrow}\; \mathcal{D}\mathcal{SEQ}(\mathbf{iMLL}^{>0}) \hookrightarrow \mathcal{D}\mathcal{SEQ}(\mathbf{iMLL})$$

Moreover, $\mathcal{D}\mathcal{SEQ}(\mathbf{iMLL}^{>0})$ is the image of $\mathcal{D}\mathcal{SEQ}(\mathbf{NL}\diamond_{\mathcal{R}_{\max}})$ under the map $\lceil\|-\|\rceil$. ♦

From the previous corollary we conclude that adding structural rules to $\mathbf{NL}\diamond$ will never move us outside **MLL**, whence CONTRACTION$_i$ or L/RWEAKENING$_i$ are never admissible.

Lemma 2.3 The left rules for the $\otimes$-like connectives (L$\diamond_j$, L$\bullet_i$) and the right rules for the $⅋$-like connectives (R$\Box^{\downarrow}{}_j$, R$\backslash_i$, R$/_i$) are reversible. ♦

This is proved by means of their respective counterparts and CUT. The reversibility of L$\diamond_j$ and L$\bullet_i$ means that the role of $\langle\;\rangle^j$ and $\circ_i$ in the antecedent structure trees actually coincides with that of $\diamond_j$ respectively $\bullet_i$. However, this does not mean we can forget about the constructors, since the occurrence of a formula A as a leaf of a structure tree $\Gamma[A]$ guarantees A occurs positively (and not negatively) in the formula $\Gamma[A]^\bullet$, which is needed in order to have meaningful inference rules.

As an immediate consequence of the previous lemma we have

Lemma 2.4 This calculus satisfies the following adjunctions:

$$\begin{array}{rcll} A \bullet_i (-) & \dashv & A \backslash_i (-) & \text{(for all formulas } A) \\ (-) \bullet_i A & \dashv & (-) /_i A & \text{(for all formulas } A) \\ \Diamond_j (-) & \dashv & \Box^{\downarrow}{}_j (-) & \end{array}$$

i.e.

$$\dfrac{A \bullet_i B \vdash C}{B \vdash A \backslash_i C} \Updownarrow \qquad \dfrac{B \bullet_i A \vdash C}{B \vdash C /_i A} \Updownarrow \qquad \dfrac{\Diamond_j B \vdash C}{B \vdash \Box^{\downarrow}{}_j C} \Updownarrow$$

♦

We divide the logical rules in two parts:
• the *tensor rules* are the right rules for the ⊗-like connectives and the left rules for the ⅋-like connectives (R$\Diamond_j$, R$\bullet_i$, L$\Box^{\downarrow}{}_j$, L$\backslash_i$, L$/_i$);
• the *par rules* are the left rules for the ⊗-like connectives and the right rules for the ⅋-like connectives (L$\Diamond_j$, L$\bullet_i$, R$\Box^{\downarrow}{}_j$, R$\backslash_i$, R$/_i$).
In the derivations we will indicate par rules by dashed horizontal lines. Lemma 2.3 now can be reformulated as: all par rules are reversible.

In the sequel we will introduce square bracketed abbreviations like [Q] for structural rules.

Example 2.5 Let $\mathcal{R}$ consist of

$$\dfrac{\Delta[\,\Gamma_1 \circ_0 (\Gamma_2 \circ_0 \Gamma_3)\,] \vdash C}{\Delta[\,(\Gamma_1 \circ_0 \Gamma_2) \circ_0 \Gamma_3\,] \vdash C}\,[\mathrm{LAss}_0]$$

Then we can derive:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{A \vdash A} \qquad \dfrac{\dfrac{}{B \vdash B} \qquad \dfrac{}{C \vdash C}}{B \circ_0 C \vdash B \bullet_0 C}}{A \circ_0 (B \circ_0 C) \vdash A \bullet_0 (B \bullet_0 C)}}{(A \circ_0 B) \circ_0 C \vdash A \bullet_0 (B \bullet_0 C)}\,[\mathrm{LAss}_0]}{(A \bullet_0 B) \circ_0 C \vdash A \bullet_0 (B \bullet_0 C)}\,\mathrm{L}\bullet_0}{(A \bullet_0 B) \bullet_0 C \vdash A \bullet_0 (B \bullet_0 C)}\,\mathrm{L}\bullet_0$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{C \vdash C} \qquad \dfrac{\dfrac{}{A \vdash A} \qquad \dfrac{}{B \vdash B}}{A \circ_0 A \backslash_0 B \vdash B}}{A \circ_0 ((A \backslash_0 B) /_0 C \circ_0 C) \vdash B}}{(A \circ_0 (A \backslash_0 B) /_0 C) \circ_0 C \vdash B}\,[\mathrm{LAss}_0]}{A \circ_0 (A \backslash_0 B) /_0 C \vdash B /_0 C}\,\mathrm{R}/_0}{(A \backslash_0 B) /_0 C \vdash A \backslash_0 (B /_0 C)}\,\mathrm{R}\backslash_0$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\overline{B \vdash B} \qquad \dfrac{\overline{A \vdash A} \qquad \overline{C \vdash C}}{A \circ_0 A \backslash_0 C \vdash C}}{A \circ_0 (B \circ_0 B \backslash_0 (A \backslash_0 C)) \vdash C}}{(A \circ_0 B) \circ_0 B \backslash_0 (A \backslash_0 C) \vdash C}\,[\mathrm{LAss}_0]}{(A \bullet_0 B) \circ_0 B \backslash_0 (A \backslash_0 C) \vdash C}\,\mathrm{L}\bullet_0}{B \backslash_0 (A \backslash_0 C) \vdash (A \bullet_0 B) \backslash_0 C}\,\mathrm{R}\backslash_0$$

♦

Illustration: wh-extraction in English

To give an indication of how we can use the calculus described in the previous section to give an account of linguistic phenomena, we will look at what is often called *wh*-extraction.

We will, for the purpose of the current discussion look at only two *wh* words, 'which' and 'whom'. Both are noun modifiers which select a sentence from which a noun phrase is missing, the difference being that with 'whom' the missing noun phrase cannot occur in subject position, as indicated by the following examples. The * in example 4 denotes this sentence is ungrammatical.

(1) agent which [$[\,]_{np}$ read National Enquirer $]_s$

(2) agent which [Mulder liked $[\,]_{np}$ $]_s$

(3) agent which [Skinner considered $[\,]_{np}$ dangerous $]_s$

(4) *agent whom [$[\,]_{np}$ read National Enquirer $]_s$

(5) agent whom [Mulder liked $[\,]_{np}$ $]_s$

(6) agent whom [Skinner considered $[\,]_{np}$ dangerous $]_s$

To account for this different behaviour, we give a very simple grammar fragment with one binary mode 0 and two unary modes 0 and 1. An extracted np is marked as $\Diamond_0\Box_0^{\downarrow}np$ if subject extraction is allowed and as $\Diamond_1\Box_1^{\downarrow}np$ if it isn't. As $\Diamond_j\Box_j^{\downarrow}A \vdash A$ is a theorem of the base logic for all j and A, this allows these constituents to function as an np. What is crucial is that the $\mathrm{L}\Box_j^{\downarrow}$ rule, read from premiss to conclusion, introduces unary brackets, which makes the following structural rules available for $\langle - \rangle^0$.

$$\frac{\Delta[\,\Gamma_1 \circ_0 (\Gamma_2 \circ_0 \langle \Gamma_3 \rangle^0)\,] \vdash C}{\Delta[\,(\Gamma_1 \circ_0 \Gamma_2) \circ_0 \langle \Gamma_3 \rangle^0\,] \vdash C}\,[\mathrm{Ass}_{0,0}]$$

$$\frac{\Delta[\,(\Gamma_1 \circ_0 \langle \Gamma_2 \rangle^0) \circ_0 \Gamma_3\,] \vdash C}{\Delta[\,(\Gamma_1 \circ_0 \Gamma_3) \circ_0 \langle \Gamma_2 \rangle^0\,] \vdash C}\,[\mathrm{MxCom}_{0,0}]$$

$$\frac{\Delta[\,\langle \Gamma_1 \rangle^0 \circ_0 \Gamma_2\,] \vdash C}{\Delta[\,\Gamma_2 \circ_0 \langle \Gamma_1 \rangle^0\,] \vdash C}\,[\mathrm{Com}_{0,0}]$$

The $[\mathrm{Ass}_{0,0}]$ and $[\mathrm{MxCom}_{0,0}]$ rules allow us to move out an embedded $\langle \Gamma \rangle^0$ constituent, whereas $[\mathrm{Com}_{0,0}]$ moves a $\langle \Gamma \rangle^0$ constituent from a left branch to a right branch after which any of the two other structural rules can apply.

Formulas marked with $\Diamond_1$, however, can only move from a right branch of a structure to another right branch. As a subject would appear on a left branch, this prevents subject extraction as desired.

$$\dfrac{\Delta[\,\Gamma_1 \circ_0 (\Gamma_2 \circ_0 \langle \Gamma_3 \rangle^1)\,] \vdash C}{\Delta[\,(\Gamma_1 \circ_0 \Gamma_2) \circ_0 \langle \Gamma_3 \rangle^1\,] \vdash C}\,{\scriptstyle [\mathrm{Ass}_{0,1}]}$$

$$\dfrac{\Delta[\,(\Gamma_1 \circ_0 \langle \Gamma_2 \rangle^1) \circ_0 \Gamma_3\,] \vdash C}{\Delta[\,(\Gamma_1 \circ_0 \Gamma_3) \circ_0 \langle \Gamma_2 \rangle^1\,] \vdash C}\,{\scriptstyle [\mathrm{MxCom}_{0,1}]}$$

The lexicon, with which we can derive all well-formed example sentences given above is the following:

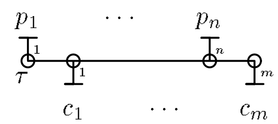
$$\begin{aligned}
&\mathrm{lex}(\mathit{agent}) &&= n\\
&\mathrm{lex}(\mathit{dangerous}) &&= n/_0 n\\
&\mathrm{lex}(\mathit{Mulder}) &&= np\\
&\mathrm{lex}(\mathit{Skully}) &&= np\\
&\mathrm{lex}(\mathit{Skinner}) &&= np\\
&\mathrm{lex}(\mathit{National\ Enquirer}) &&= np\\
&\mathrm{lex}(\mathit{liked}) &&= (np\backslash_0 s)/_0 np\\
&\mathrm{lex}(\mathit{read}) &&= (np\backslash_0 s)/_0 np\\
&\mathrm{lex}(\mathit{considered}) &&= ((np\backslash_0 s)/_0(n/_0 n))/_0 np\\
&\mathrm{lex}(\mathit{which}) &&= (n\backslash_0 n)/_0(s/_0\Diamond_0\Box_0^{\downarrow} np)\\
&\mathrm{lex}(\mathit{whom}) &&= (n\backslash_0 n)/_0(s/_0\Diamond_1\Box_1^{\downarrow} np)
\end{aligned}$$

We can, for example, derive sentence 2 as follows.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{np \vdash np}{\scriptstyle \mathrm{Ax}} \quad \dfrac{\dfrac{}{np \vdash np}{\scriptstyle \mathrm{Ax}} \quad \dfrac{}{s \vdash s}{\scriptstyle \mathrm{Ax}}}{np \circ_0 np\backslash_0 s \vdash s}{\scriptstyle \mathrm{L}\backslash_0}}{np \circ_0 ((np\backslash_0 s)/_0 np \circ_0 np) \vdash s}{\scriptstyle \mathrm{L}/_0}}{np \circ_0 ((np\backslash_0 s)/_0 np \circ_0 \langle \Box_0^{\downarrow} np \rangle^0) \vdash s}{\scriptstyle \mathrm{L}\Box_0^{\downarrow}}}{(np \circ_0 (np\backslash_0 s)/_0 np) \circ_0 \langle \Box_0^{\downarrow} np \rangle^0 \vdash s}{\scriptstyle [\mathrm{Ass}_{0,0}]}}{(np \circ_0 (np\backslash_0 s)/_0 np) \circ_0 \Diamond_0\Box_0^{\downarrow} np \vdash s}{\scriptstyle \mathrm{L}\Diamond_0}}{np \circ_0 (np\backslash_0 s)/_0 np \vdash s/_0\Diamond_0\Box_0^{\downarrow} np}{\scriptstyle \mathrm{R}/_0} \quad \dfrac{\dfrac{}{n \vdash n}{\scriptstyle \mathrm{Ax}} \quad \dfrac{}{n \vdash n}{\scriptstyle \mathrm{Ax}}}{n \circ_0 n\backslash_0 n \vdash n}{\scriptstyle \mathrm{L}\backslash_0}}{n \circ_0 ((n\backslash_0 n)/_0(s/_0\Diamond_0\Box_0^{\downarrow} np) \circ_0 (np \circ_0 (np\backslash_0 s)/_0 np)) \vdash n}{\scriptstyle \mathrm{L}/_0}$$

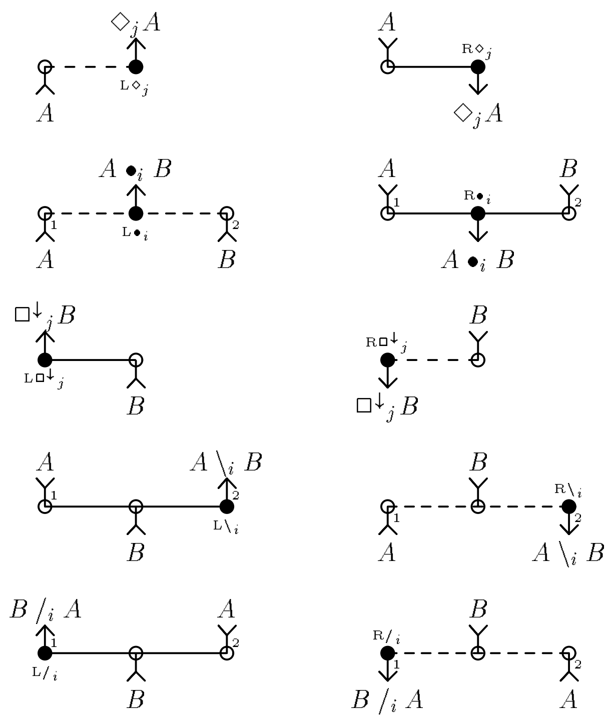
3 Proof structures

Given a set S, we generally define a *link* L in S to be an ordered pair $\langle P, C \rangle_\tau$ of sequences of elements of S (called the premisses and conclusions of L) labeled by a certain type τ. It will be represented by a horizontal bar, also labeled by τ, together with the elements of P above it and the elements of C below it:



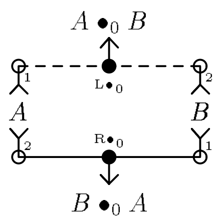
Let S be a multiset of formulas, i.e. a set of formula occurrences. We will restrict to links L where one of the formulas (called the *main* formula or the *output* formula of L) is obtained as a connective applied to the other formulas (called the *active* formulas or the *input* formulas of L). Depending on whether the main formula is a premiss or a conclusion, and moreover on which connective is applied, we distinguish $6\,|I| + 4\,|J|$ types (where I and J are the sets of modes):

Definition 3.1 A *proof structure* $\langle S, \mathcal{L} \rangle$ consists of a finite set S of formulas together with a set $\mathcal{L}$ of links in S of the following forms:

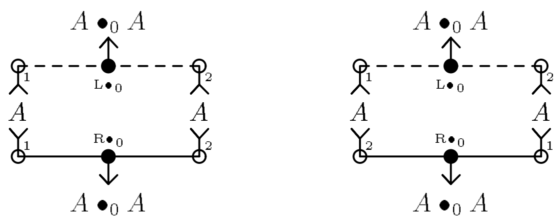


such that the following holds:
- every formula of S is at most once a conclusion of a link;
- every formula of S is at most once a premiss of a link. ♦

Here we have ordered the premisses/conclusions of the links in the picture from left to right. However, in general this is not always possible. E.g. in



the R$\bullet_0$ link has first premiss B and second premiss A. Observe that the following two proof structures are different as the first conclusion of the L$\bullet_0$ link is the first respectively second premiss of the R$\bullet_0$ link:

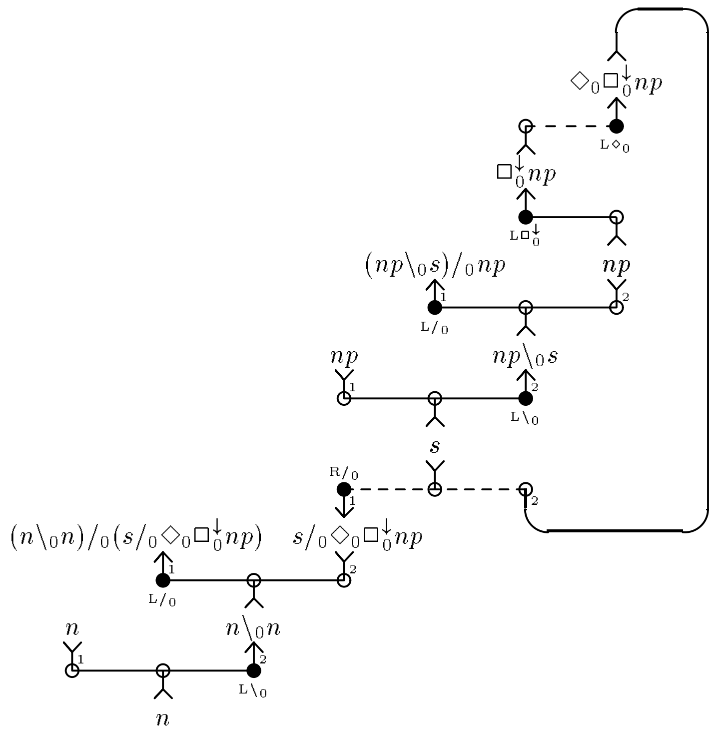


The formulas which are not the conclusion of a link are the *hypotheses* H_k of $\mathcal{S} = \langle S, \mathcal{L} \rangle$, while those that are not the premiss of a link are the *conclusions* Q_l of $\mathcal{S}$. This is also expressed by saying that $\mathcal{S}$ is a proof structure from $\{H_1, H_2, \ldots\}$ to $\{Q_1, Q_2, \ldots\}$.

We divide the links in two parts:
• the *tensor links* are the right links for the $\otimes$-like connectives and the left links for the $⅋$-like connectives (R$\Diamond_j$, R$\bullet_i$, L$\Box^{\downarrow}{}_j$, L$\backslash_i$, L$/_i$);
• the *par links* are the left links for the $\otimes$-like connectives and the right links for the $⅋$-like connectives (L$\Diamond_j$, L$\bullet_i$, R$\Box^{\downarrow}{}_j$, R$\backslash_i$, R$/_i$).
We graphically indicate tensor vs. par links by solid vs. dashed horizontal lines.

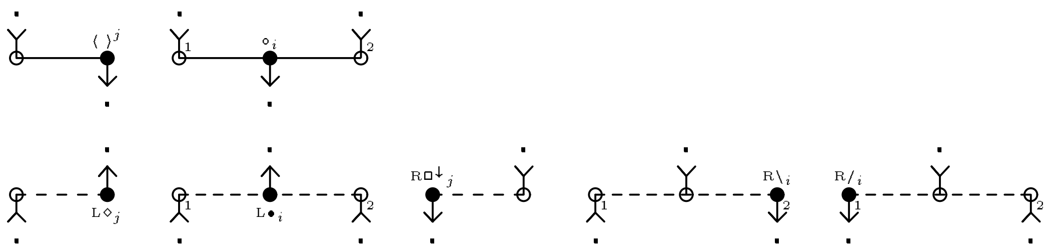
Note that there are no links corresponding to the identity rules. Instead, we will have axiomatic and cut formulas. An *axiomatic formula* is a formula which is not the main formula of any link, whereas a *cut formula* is a formula which is the main formula of two links.

Example 3.2 A proof structure corresponding to the sequent derivation on page 8 is shown below.



♦

Definition 3.3 A *correction structure* $\langle N, \mathcal{L}\rangle$ consists of a finite set N of nodes together with a set $\mathcal{L}$ of links in N of the following forms:



such that the following holds:
• every node of S is at most once a conclusion of a link;
• every node of S is at most once a premiss of a link. ♦

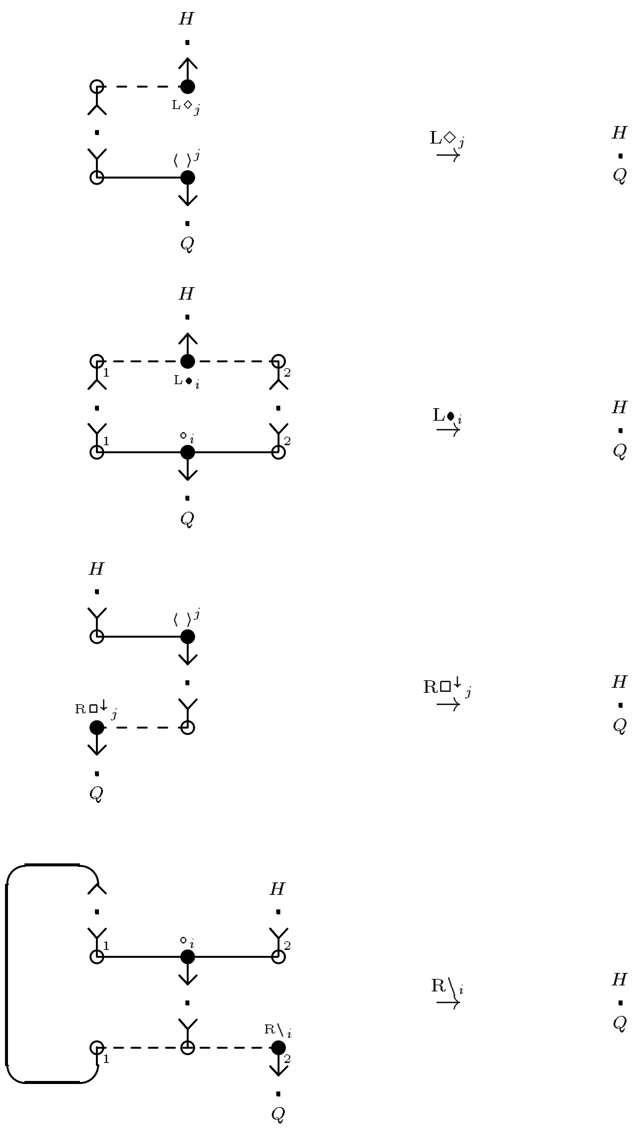
Definition 3.4 A *hypothesis structure* $\langle N, \mathcal{L}, \lambda\rangle$ consist of a correction structure $\langle N, \mathcal{L}\rangle$ and a labeling λ of its nodes: to every node there are assigned a (perhaps empty) *upper label* and a *lower label*

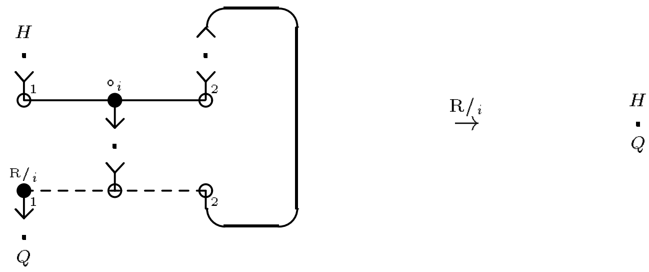
upper label
.
lower label

each one consisting of zero or one formulas. This labeling is such that exactly each hypothesis node h has a non-empty upper label $\{H_h\}$, and exactly each conclusion node q has a non-empty lower label $\{Q_q\}$, and we say that $\langle N, \mathcal{L}, \lambda\rangle$ is a hypothesis structure from $\{H_1, H_2, \ldots\}$ to $\{Q_1, Q_2, \ldots\}$. ♦

Next we will define *conversion steps* on hypothesis structures. One easily checks that these conversion steps preserve the labels: if $\mathcal{H} = \langle N, \mathcal{L}, \lambda\rangle$ is a hypothesis structure from $\{H_1, H_2, \ldots\}$ to $\{Q_1, Q_2, \ldots\}$, then so is $\mathcal{H}'$ which is obtained from $\mathcal{H}$ by applying a conversion step. There are two kinds of conversion steps: contractions and structural conversions. Every conversion step works on a number of links, constituting the so called *redex*, which is a correction structure itself. Below, all nodes of each depictured redex are distinct. Hence the redex of a contraction has one hypothesis node and one conclusion node, while the redex of an n-ary structural conversion has n hypothesis nodes and one conclusion node.

By a *contraction* we mean the replacement of one of the following pairs of links by a single node, which will be labeled as indicated (H and Q are labels, so each of them consists of zero or one formulas). The contraction will be named after the par link (L$\Diamond_j$, L$\bullet_i$, R$\Box^{\downarrow}{}_j$, R$\backslash_i$, R$/_i$).





By a *structural conversion* we mean the following: for an n-ary structural rule

$$\frac{\Delta[\ \Xi[\ \Gamma_1, \ldots, \Gamma_n\ ]\ ]\ \vdash\ C}{\Delta[\ \Xi'[\ \Gamma_{\pi_1}, \ldots, \Gamma_{\pi_n}\ ]\ ]\ \vdash\ C}\ \langle \Xi, \Xi', \pi \rangle$$

both formal trees Ξ and Ξ' may be represented by a correction structure with n hypotheses and one conclusion. Ordering the premisses of all our R $\circ_i$ links in the picture from left to right, yields an order on the hypotheses of both correction structures. Now, if Ξ is part of $\mathcal{H}$ — the hypothesis nodes being $x_1, \ldots, x_n$ (in this order) — the structural conversion consists of replacing Ξ by Ξ' and permuting the nodes to get them in the order $x_{\pi_1}, \ldots, x_{\pi_n}$. The conversion is denoted by

$$\Xi[\ x_1, \ldots, x_n\ ]\ \to\ \Xi'[\ x_{\pi_1}, \ldots, x_{\pi_n}\ ].$$

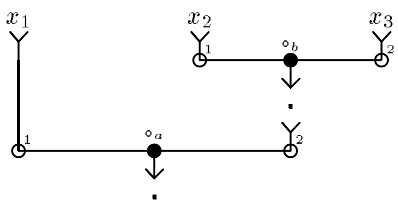
Example 3.5 Let us consider the structural rule [Q]:

$$\frac{\Delta[\ \Gamma_1\ \circ_a\ (\Gamma_2\ \circ_b\ \Gamma_3)\ ]\ \vdash\ C}{\Delta[\ (\Gamma_3\ \circ_d\ \langle\ \Gamma_1\ \rangle^e)\ \circ_c\ \Gamma_2\ ]\ \vdash\ C}\ [\mathrm{Q}]$$

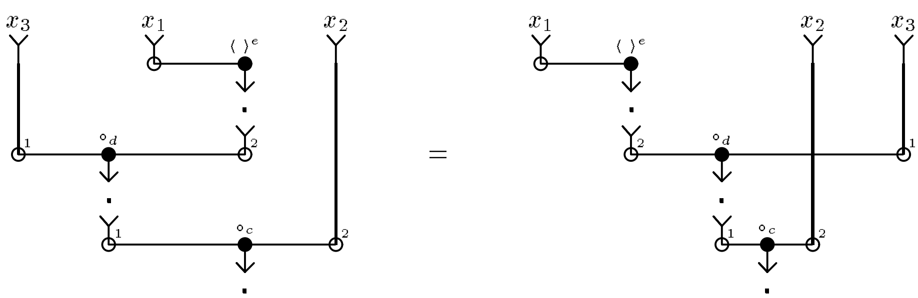
where a, b, c, d are binary modes and e is a unary mode. The corresponding structural conversion

$$x_1\ \circ_a\ (x_2\ \circ_b\ x_3)\ \to\ (x_3\ \circ_d\ \langle\ x_1\ \rangle^e)\ \circ_c\ x_2$$

consists in the replacement of



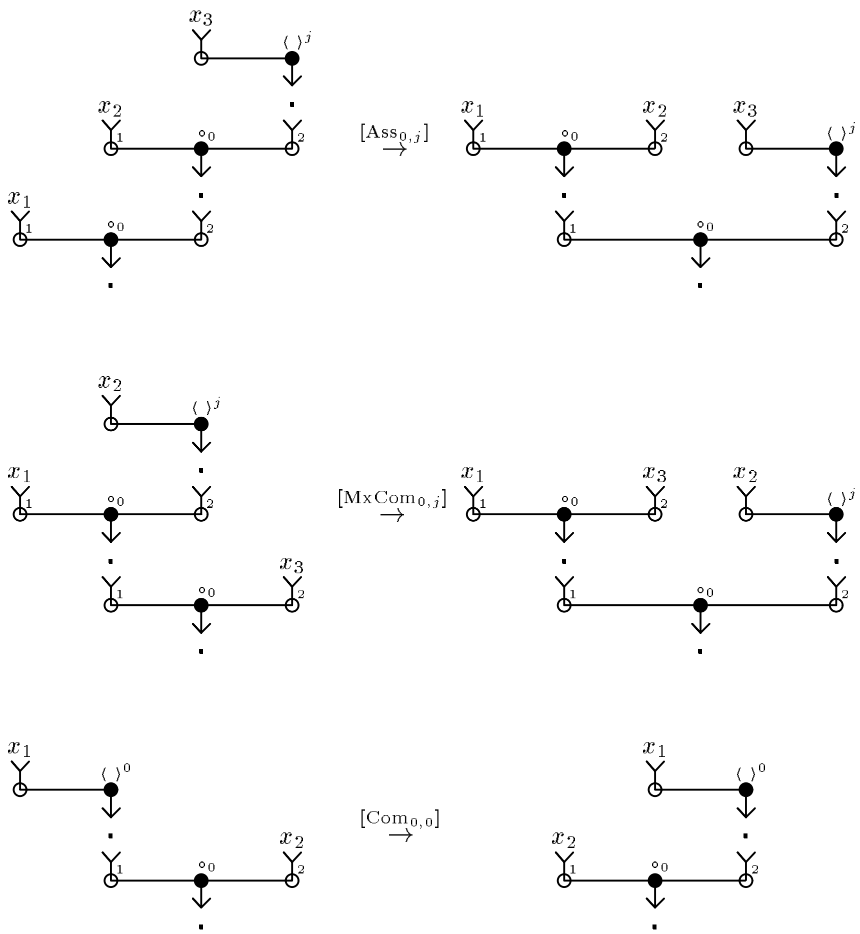
by



♦

Observe that a structural conversion is a local operation; it does not influence the other links of $\mathcal{L}$, nor the other nodes of N. The last example shows how to see this in our graphical representation: we need not permute the hypothesis nodes, if we represent Ξ' in the appropriate way.

Example 3.6 The structural rules for our English fragment correspond to the following structural conversions, where j ranges over $\{0,1\}$.

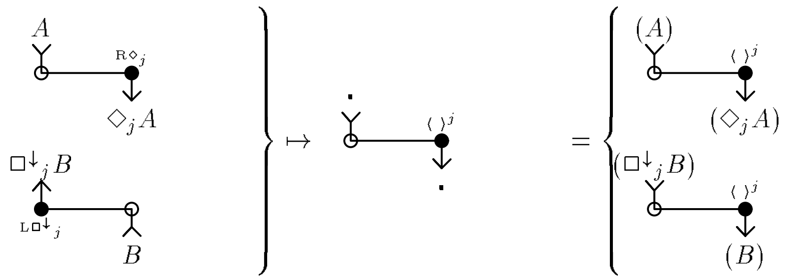


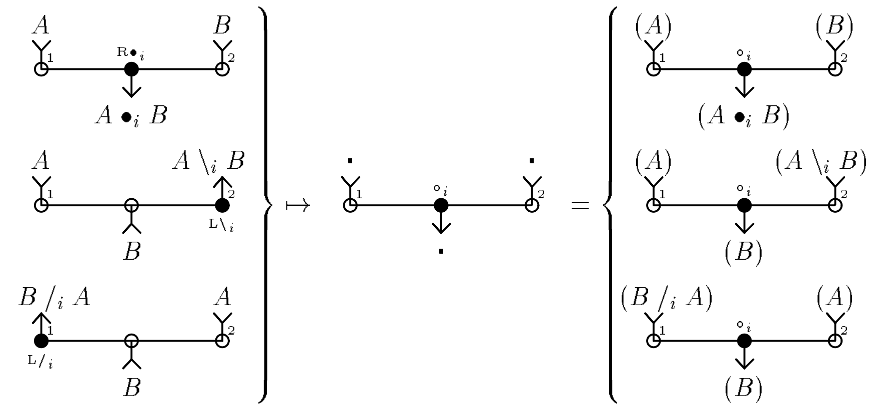
♦

To any proof structure $\mathcal{S}$ from $\{H_1, H_2, \ldots\}$ to $\{Q_1, Q_2, \ldots\}$ we assign a hypothesis structure $\widehat{\mathcal{S}}$ from $\{H_1, H_2, \ldots\}$ to $\{Q_1, Q_2, \ldots\}$ by a replacement of the link types R$\Diamond_j$ and L$\Box^{\downarrow}{}_j$ by the new link type $\langle\ \rangle^j$, and by a replacement of the link types R$\bullet_i$, L$\backslash_i$ and L$/_i$ by the new link type $\circ_i$. The formulas A become the nodes (A) of $\widehat{\mathcal{S}}$, and the label H (resp. Q) of a node

$$(A)\overset{H}{\underset{Q}{\cdot}}$$

is chosen A precisely if A is a hypothesis (resp. conclusion), and empty otherwise. $\widehat{\mathcal{S}}$ is called the *underlying* hypothesis structure of $\mathcal{S}$.





For any structure tree Γ and formula C, let $\|\Gamma\|$ be the multiset of elements in Γ; let $\langle\!\langle\Gamma\rangle\!\rangle$ be the sequence of elements in Γ obtained by left to right traversal of the tree; let Γ_C be the obvious hypothesis structure from $\|\Gamma\|$ to $\{C\}$ with conclusion node (lower) labeled by C. Any hypothesis structure of this form will be called a *hypothesis tree*. Let $\twoheadrightarrow_{\mathcal{R}}$ be the transitive, reflexive closure of $\rightarrow_{\mathcal{R}}$, by which we mean the contractions as well as the structural conversions belonging to $\mathcal{R}$.

In the next sections we will prove our main theorem:

Theorem 3.7
$\Gamma \vdash C$ is derivable in $\mathbf{NL}\Diamond_{\mathcal{R}}$ if and only if there is a proof structure $\mathcal{S}$ (from $\|\Gamma\|$ to $\{C\}$) such that $\widehat{\mathcal{S}} \twoheadrightarrow_{\mathcal{R}} \Gamma_C$. ♦

Note that any proof structure $\mathcal{S}$ such that $\widehat{\mathcal{S}} \twoheadrightarrow_{\mathcal{R}} \Gamma_C$ is automatically from $\|\Gamma\|$ to $\{C\}$.

Definition 3.8 Let $\mathcal{S}$ be a proof structure.

1. $(\mathcal{S}, \rho)$ is a *$\mathcal{R}$-conversion sequence of* $\Gamma \vdash C$ iff $\widehat{\mathcal{S}} \overset{\rho}{\twoheadrightarrow}_{\mathcal{R}} \Gamma_C$;
2. $\mathcal{S}$ is a *$\mathcal{R}$-proof net of* $\Gamma \vdash C$ iff $\widehat{\mathcal{S}} \twoheadrightarrow_{\mathcal{R}} \Gamma_C$, i.e. iff $\widehat{\mathcal{S}} \overset{\rho}{\twoheadrightarrow}_{\mathcal{R}} \Gamma_C$ for some ρ;
3. Let Σ be a multiset of formulas. We say $\mathcal{S}$ is a *$\mathcal{R}$-proof net from* Σ *to* $\{C\}$ iff $\widehat{\mathcal{S}} \twoheadrightarrow_{\mathcal{R}} \Gamma_C$ for some Γ such that $\|\Gamma\| = \Sigma$, or equivalently: iff $\mathcal{S}$ is a proof structure from Σ to $\{C\}$ on which all contractions can be applied (together with the necessary structural conversions) such that we end with a hypothesis tree.

♦

With this definition theorem 3.7 can be reformulated as:
There is a $\mathbf{NL}\diamond_{\mathcal{R}}$ derivation of $\Gamma \vdash C$ if and only if there is a $\mathcal{R}$-proof net of $\Gamma \vdash C$.

For many applications one is interested in derivability of $\Gamma \vdash C$ for some structure tree Γ subject to certain constraints. Instead of checking the RHS condition for each Γ, we get the witnessing Γ as a result of the conversion steps, as stated in the following corollary. This fact shows the computational strength of our condition (see section 7).

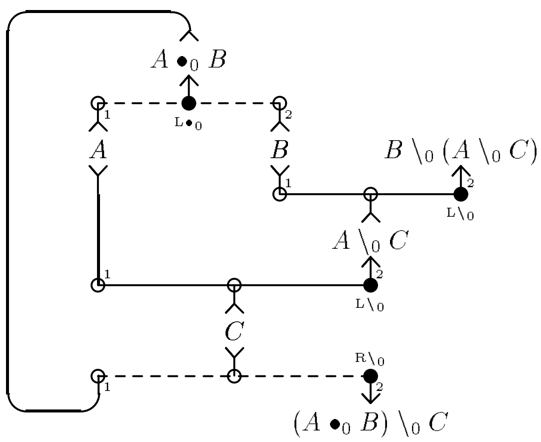
Corollary 3.9 (Abstraction of Γ)

1. Let us call a sequence Σ of formulas a C-*sequence* if, for some Γ such that $\langle\!\langle\Gamma\rangle\!\rangle = \Sigma$, the sequent $\Gamma \vdash C$ is derivable. Then Σ is a C-sequence iff there is a proof structure from $\|\Sigma\|$ to $\{C\}$, on which all contractions can be applied (together with the necessary structural conversions) such that we end with a hypothesis tree in which the order of the hypotheses equals Σ.

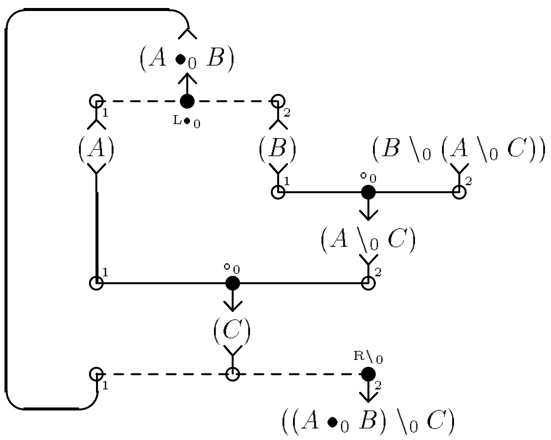
2. Let us call a multiset Σ of formulas a C-*multiset* if, for some Γ such that $\|\Gamma\| = \Sigma$, the sequent $\Gamma \vdash C$ is derivable. Then Σ is a C-multiset iff there is a proof structure from Σ to $\{C\}$, on which all contractions can be applied (together with the necessary structural conversions) such that we end with a hypothesis tree, i.e. iff there is a proof net from Σ to $\{C\}$.

♦

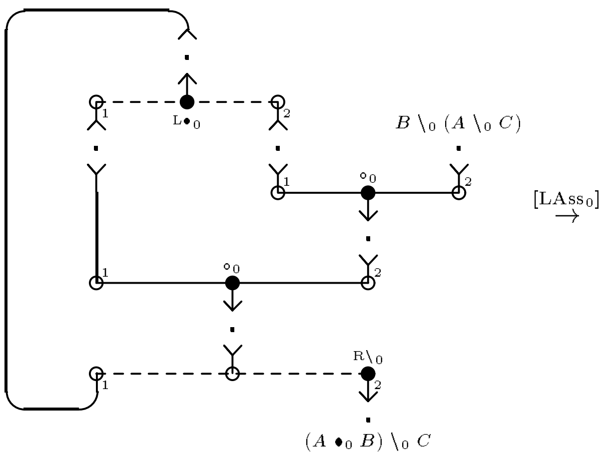
Example 3.10 The following proof structure

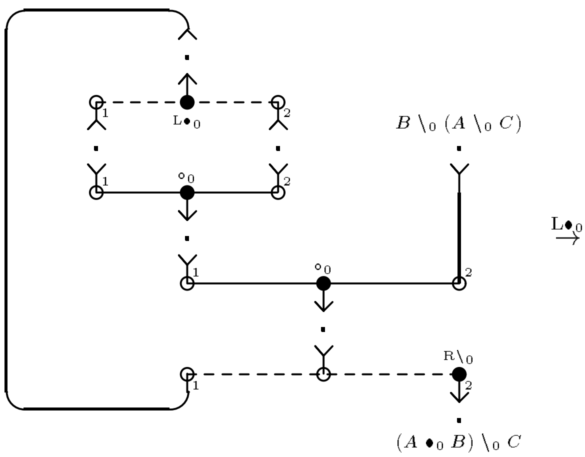


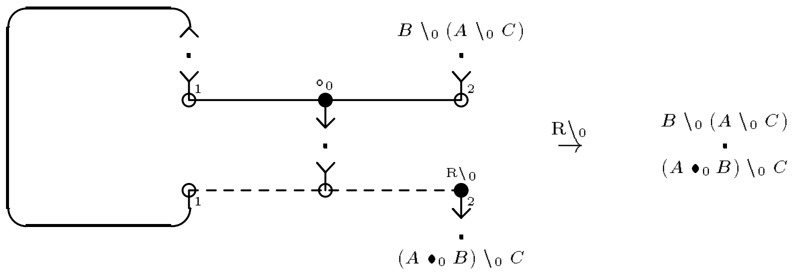
has hypothesis structure



which converts (under $\mathcal{R} = \{[\mathrm{LAss}_0]\}$) as follows:

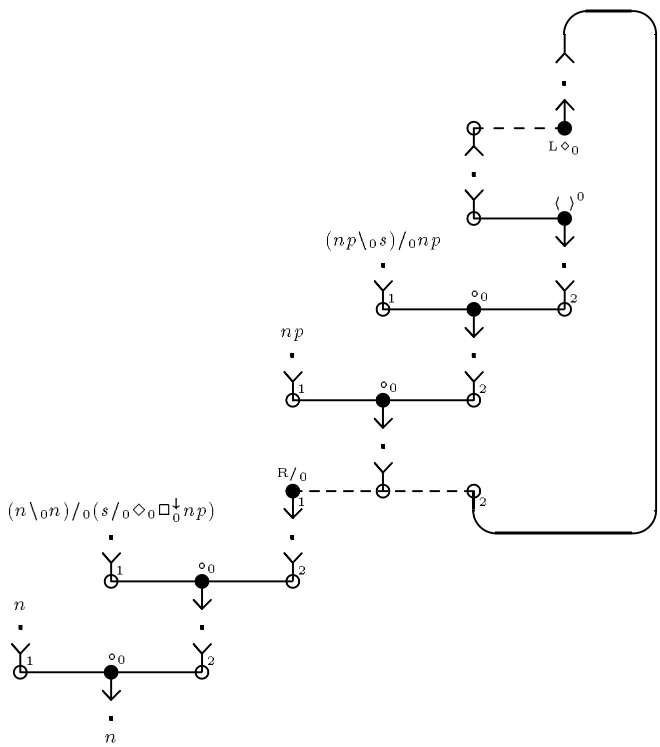




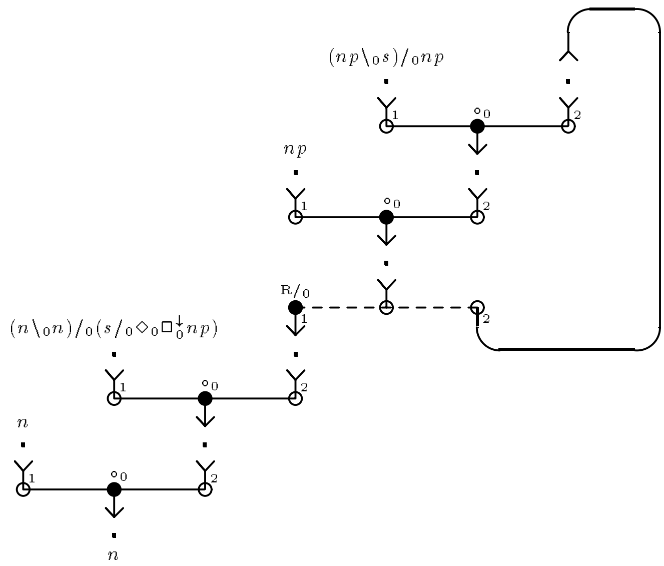


♦

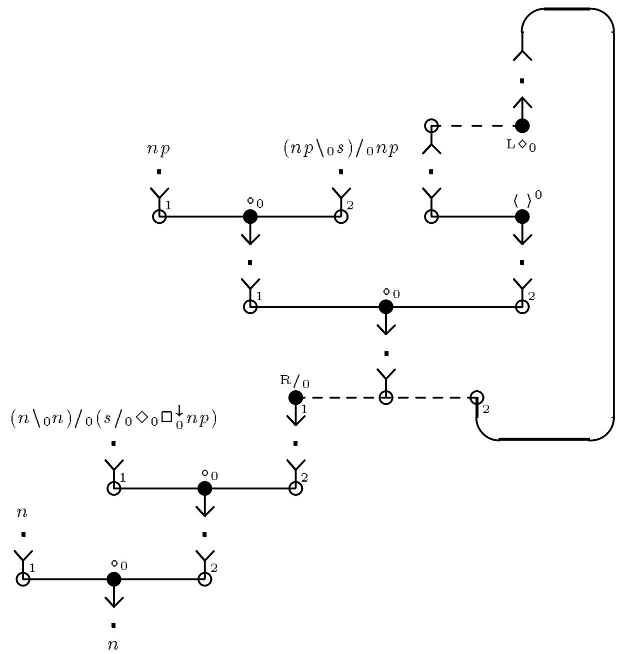
Example 3.11 The hypothesis structure corresponding to the proof structure of example 3.2 is the following:



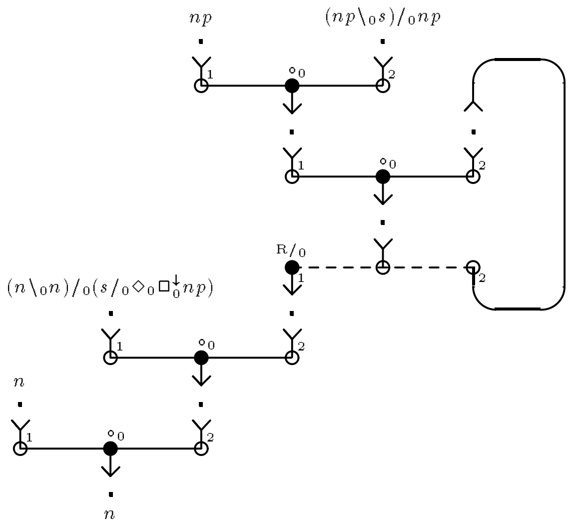
We can convert this hypothesis structure to a hypothesis tree using the following conversions. From the initial hypothesis structure, we have two choices: we can apply either the L$\Diamond_0$ contraction or the [Ass$_{0,0}$] conversion. The L$\Diamond_0$ contraction gives us the following hypothesis structure:



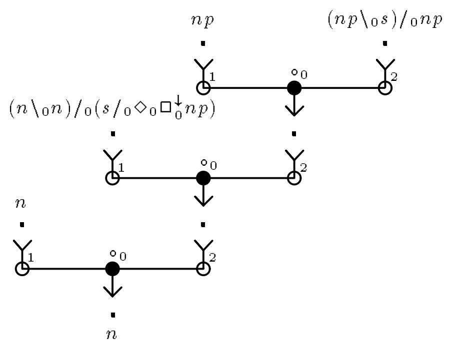
At this stage, none of the conversions apply and we still have to remove the $R/_0$ link to get a hypothesis tree. This does not mean our proof structure is not a proof net, however, as we can try the other possibility:



after which we apply the $L\Diamond_0$ contraction.



Finally, we can apply the R$/_0$ contraction to obtain the following hypothesis tree:



♦

4 Soundness

Theorem 4.1 ($\boxed{\Longrightarrow}$ part of theorem 3.7)
If $\Gamma \vdash C$ is derivable in $\mathbf{NL}\diamond_{\mathcal{R}}$, then there is a proof structure $\mathcal{S}$ (from $\|\Gamma\|$ to $\{C\}$) such that $\widehat{\mathcal{S}} \twoheadrightarrow_{\mathcal{R}} \Gamma_C$. ♦

Proof: We apply induction on the derivation of $\Gamma \vdash C$. By 'applying a conversion step to a proof structure' we will mean 'applying a conversion step to its underlying hypothesis structure'.

The identity rules

For an axiom

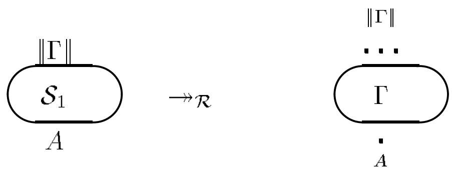
$$\frac{}{A \vdash A}\ \text{Ax}$$

take the trivial proof structure consisting of one formula A and no links. Its hypothesis structure $\overset{A}{\underset{A}{\cdot}}$ converts into A_A in zero steps.

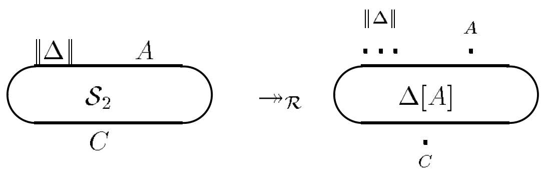
For a Cut inference

$$\frac{\Gamma \vdash A \qquad \Delta[A] \vdash C}{\Delta[\Gamma] \vdash C}\ \text{Cut}$$

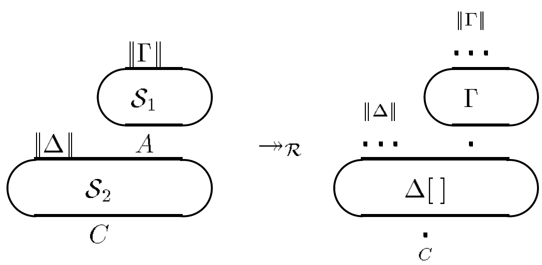
by induction hypothesis we know that there are proof structures $\mathcal{S}_1$ from $\|\Gamma\|$ to $\{A\}$ such that $\widehat{\mathcal{S}_1} \twoheadrightarrow_{\mathcal{R}} \Gamma_A$:



and $\mathcal{S}_2$ from $\|\Delta\| \cup \{A\}$ to $\{C\}$ such that $\widehat{\mathcal{S}_2} \twoheadrightarrow_{\mathcal{R}} \Delta[A]_C$:



Pasting $\mathcal{S}_1$ and $\mathcal{S}_2$ in A yields a proof structure from $\|\Gamma\| \cup \|\Delta\|$ to $\{C\}$ which converts into Γ_A pasted to $\Delta[A]_C$, i.e. it converts into $\Delta[\Gamma]_C$, as desired:

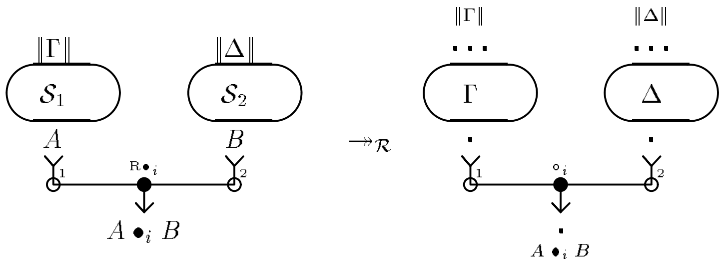


The tensor rules

For a R$\bullet_i$ rule

$$\frac{\Gamma \vdash A \qquad \Delta \vdash B}{\Gamma \circ_i \Delta \vdash A \bullet_i B}\ {\scriptstyle \mathrm{R}\bullet_i}$$

assuming the appropriate induction hypothesis, we find:



The unary version

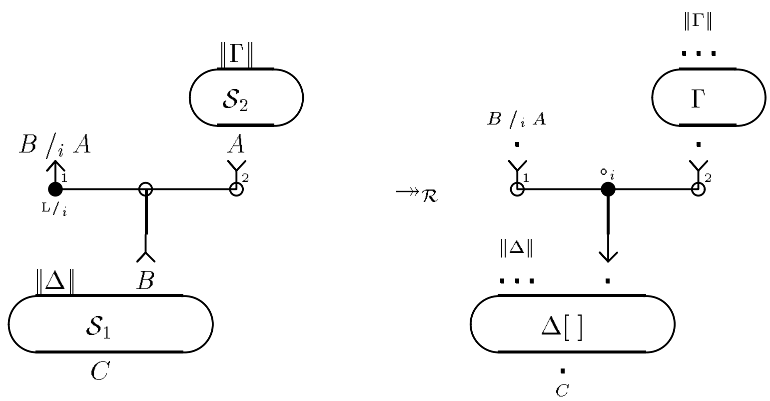
$$\frac{\Gamma \vdash A}{\langle \Gamma \rangle^j \vdash \Diamond_j A}\ {\scriptstyle \mathrm{R}\Diamond_j}$$

is proved similarly.

For a L$/_i$ rule

$$\frac{\Gamma \vdash A \qquad \Delta[B] \vdash C}{\Delta[B /_i A \circ_i \Gamma] \vdash C}\ {\scriptstyle \mathrm{L}/_i}$$

assuming the appropriate induction hypothesis, we find:



The L$\backslash_i$ case is the symmetric counterpart, while the unary version

$$\frac{\Delta[B] \vdash C}{\Delta[\langle \Box^{\downarrow}_j B \rangle^j] \vdash C}\ {\scriptstyle \mathrm{L}\Box^{\downarrow}_j}$$

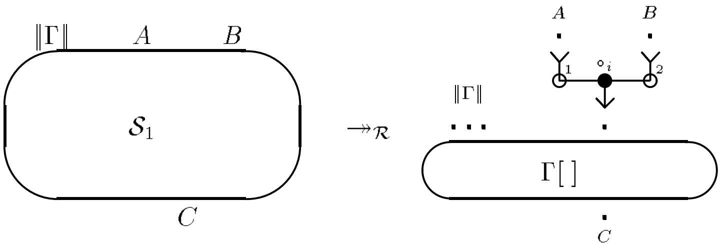
is proved analogously by deleting $\mathcal{S}_2$ in the diagram above.

The par rules

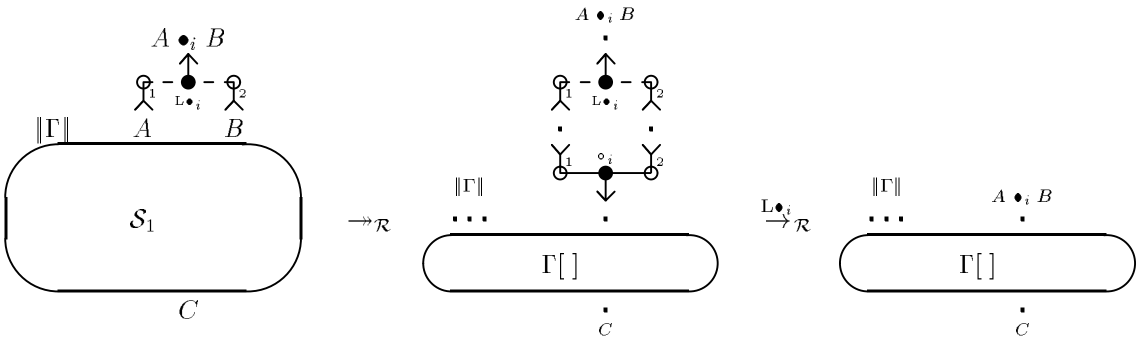
For a $\mathrm{L}\bullet_i$ rule

$$\frac{\Gamma[A \circ_i B] \vdash C}{\Gamma[A \bullet_i B] \vdash C} \ \mathrm{L}\bullet_i$$

we know by induction hypothesis that



whence



The unary version

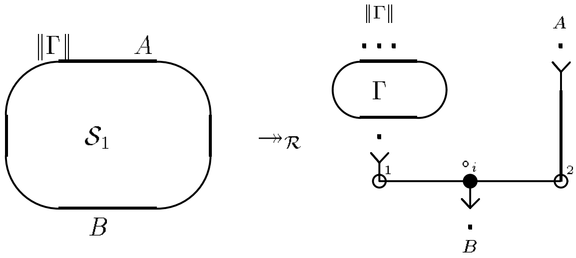
$$\frac{\Gamma[\langle A \rangle^j] \vdash C}{\Gamma[\Diamond_j A] \vdash C} \ \mathrm{L}\Diamond_j$$

is proved analogously, by an extension of the original conversion sequence by a $\mathrm{L}\Diamond_j$ contraction.

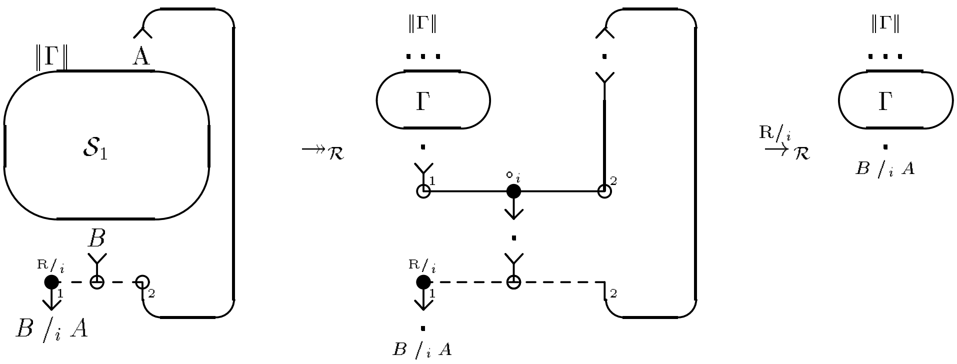
For a $\mathrm{R}/_i$ rule

$$\frac{\Gamma \circ_i A \vdash B}{\Gamma \vdash B /_i A} \ \mathrm{R}/_i$$

we know by induction hypothesis that



whence



The R$\backslash_i$ case is the symmetric counterpart, while the unary version

$$\frac{\langle \Gamma \rangle^j \vdash B}{\Gamma \vdash \Box^{\downarrow}{}_j B}\ \text{R}\Box^{\downarrow}{}_j$$

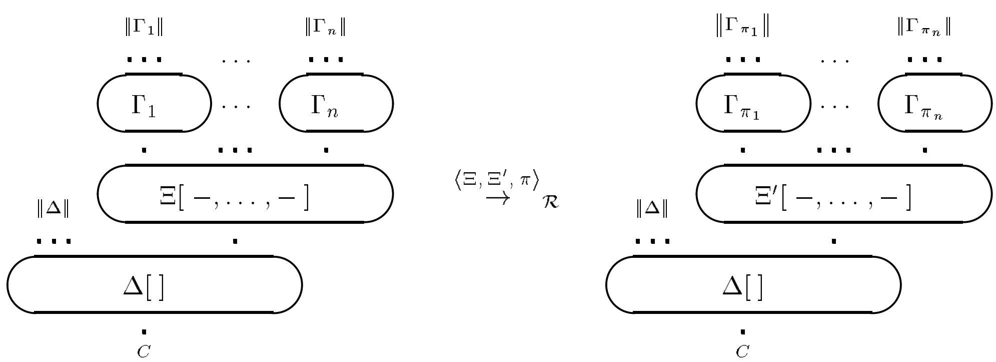
is proved analogously, by an extension of the original conversion sequence by a R$\Box^{\downarrow}{}_j$ contraction.

The structural rules

For a structural rule

$$\frac{\Delta[\ \Xi[\ \Gamma_1, \ldots, \Gamma_n\ ]\ ] \vdash C}{\Delta[\ \Xi'[\ \Gamma_{\pi_1}, \ldots, \Gamma_{\pi_n}\ ]\ ] \vdash C}\ \langle \Xi, \Xi', \pi \rangle$$

belonging to $\mathcal{R}$, assuming that $\widehat{\mathcal{S}_1} \twoheadrightarrow_{\mathcal{R}} \Delta[\ \Xi[\ \Gamma_1, \ldots, \Gamma_n\ ]\ ]_C$, we can start with the same proof structure $\mathcal{S}_1$ and extend this conversion sequence by



■

Given a derivation $\mathcal{D}$, the construction in this proof yields exactly one proof structure and at least one conversion sequence. Actually it yields a non-empty collection of conversion sequences, in the following way.

Observe that for every structural rule and for every par rule we have to extend an inductively obtained conversion sequence by the corresponding conversion, whereas no other inference rule induces a new conversion. Hence for every conversion sequence there is a bijective correspondence between the set of structural rules and par rules on the one hand, and the set of conversion steps on the other hand.

A priori there is no unique order of executing these conversions. For a binary tensor rule ($R\bullet_i$, $L\backslash_i$, $L/_i$) as well as for a Cut inference, the conversion steps in the components $\mathcal{S}_1$ and $\mathcal{S}_2$ may be executed in a parallel way; i.e. independently of each other. Now we define the collection of conversion sequences of $\mathcal{D}$ by all possible ways of interleaving a conversion sequence of $\mathcal{D}_1$ and a conversion sequence of $\mathcal{D}_2$. This yields

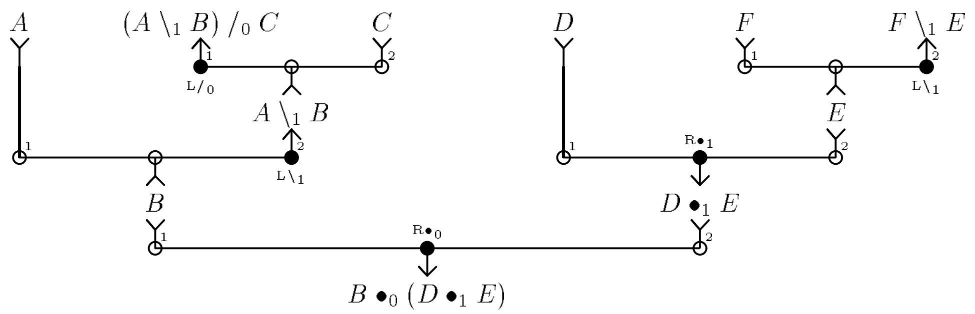
$$|\mathcal{D}| := \binom{k+l}{k} |\mathcal{D}_1|\, |\mathcal{D}_2|$$

conversion sequences, where k is the number of structural rules and par rules in $\mathcal{D}_1$, l is the number of structural rules and par rules in $\mathcal{D}_2$, and $|-|$ counts the number of conversion sequences in the inductively defined collection.

Example 4.2 1. Let $\mathcal{D}$ be the following derivation:

$$\dfrac{\dfrac{C \vdash C \qquad \dfrac{A \vdash A \qquad B \vdash B}{A \circ_1 A \backslash_1 B \vdash B}\,{\scriptstyle L\backslash_1}}{A \circ_1 ((A \backslash_1 B) /_0 C \circ_0 C) \vdash B}\,{\scriptstyle L/_0} \qquad \dfrac{F \vdash F \qquad \dfrac{D \vdash D \qquad E \vdash E}{D \circ_1 E \vdash D \bullet_1 E}\,{\scriptstyle R\bullet_1}}{D \circ_1 (F \circ_1 F \backslash_1 E) \vdash D \bullet_1 E}\,{\scriptstyle L\backslash_1}}{\Big(A \circ_1 ((A \backslash_1 B) /_0 C \circ_0 C)\Big) \circ_0 \Big(D \circ_1 (F \circ_1 F \backslash_1 E)\Big) \vdash B \bullet_0 (D \bullet_1 E)}\,{\scriptstyle R\bullet_0}$$

The proof structure of this derivation reads:



and we find only one conversion sequence: the empty one.

2. The last derivation in example 2.5 has the proof structure shown in example 3.10. The conversion sequence given there is again the only one our procedure yields. ♦

5 Sequentialisation

Lemma 5.1 If $\mathcal{S}$ is a non-trivial proof structure such that the underlying hypothesis structure $\widehat{\mathcal{S}}$ is actually a hypothesis tree Γ_C (for some structure tree Γ and formula C), then at least one of the leaves (conclusion and hypotheses) of $\mathcal{S}$ is the main formula of its link. ♦

Proof: As $\mathcal{S}$ is not trivial (i.e. a singleton), it is clear that every leaf is connected to exactly one link, so the formulation of the lemma is well-defined.

To prove: if every hypothesis is an active formula of its link, then the conclusion is the main formula of its link. We proceed by induction on Γ.

The trivial case $\Gamma = \overset{A}{\cdot}$ can not occur.

In case $\Gamma = \Gamma_1 \ \circ_i \ \Gamma_2$, assume every hypothesis is an active formula of its link. We write L for the final $\circ_i$ link, connecting Γ_1 and Γ_2. If Γ_1 is trivial, the assumption entails that the corresponding formula in $\mathcal{S}$ is an active formula of L. If Γ_1 is non-trivial, by induction hypothesis we know that its conclusion is the main formula of the link above, whence of the form $\Diamond_j A$ or $A \bullet_i B$. This implies that it is not the main formula of L, which would be $\Box^{\downarrow}{}_j B$, $A \backslash_i B$ or $B /_i A$. Hence it is an active formula of L. The same holds for the second premiss of L. As both premisses are active, the conclusion of L must be main, as desired.

The case $\Gamma = \langle \Gamma_1 \rangle^j$ is proved analogously. ■

Alternatively, we can obtain this result as a corollary of the following lemma.

Lemma 5.2 If $\mathcal{S}$ is a proof structure such that the underlying hypothesis structure $\widehat{\mathcal{S}}$ is actually a hypothesis tree, then $\lambda = \alpha$, where λ denotes the number of hypotheses of $\mathcal{S}$, and α the number of axiomatic formulas of $\mathcal{S}$. ♦

Proof: By induction on Γ.

If $\Gamma = \overset{A}{\cdot}$, then $\mathcal{S}$ is singleton A, which has one hypothesis ($\lambda = 1$) and contains one axiomatic formula ($\alpha = 1$).

In case $\Gamma = \Gamma_1 \ \circ_i \ \Gamma_2$ we have $\lambda = \lambda_1 + \lambda_2$. In order to calculate α we differentiate on the following subcases:

- $\mathcal{S} = \mathcal{S}_1$ R$\bullet_i$ $\mathcal{S}_2$: In this case $\alpha = \alpha_1 + \alpha_2 + 0$;
- $\mathcal{S} = \mathcal{S}_1$ L$\backslash_i$ $\mathcal{S}_2$: Now $\alpha = \alpha_1 + (\alpha_2 - 1) + 1$, since the second premiss of the final $\circ_i$ link L becomes non-axiomatic, whereas the conclusion of L is a new axiomatic formula;
- $\mathcal{S} = \mathcal{S}_1$ L$/_i$ $\mathcal{S}_2$: Similarly we find $\alpha = (\alpha_1 - 1) + \alpha_2 + 1$.

Hence in all subcases $\alpha = \alpha_1 + \alpha_2$. Using the induction hypothesis we find

$$\lambda = \lambda_1 + \lambda_2 = \alpha_1 + \alpha_2 = \alpha$$

In case $\Gamma = \langle \Gamma_1 \rangle^j$, we have $\lambda = \lambda_1$, while $\alpha = \alpha_1 + 0$ (in the subcase that $\mathcal{S} = \mathrm{R}\Diamond_j \mathcal{S}_1$) or $\alpha = (\alpha_1 - 1) + 1$ (in the subcase that $\mathcal{S} = \mathrm{L}\Box^{\downarrow}{}_j \mathcal{S}_1$). So by induction hypothesis $\lambda = \alpha$. ■

Now suppose $\mathcal{S}$ is as in lemma 5.1. Besides the hypotheses there is one other leaf: the conclusion. This yields

$$\text{the number of leaves} = \text{the number of axiomatic formulas} + 1.$$

Since a leaf is active exactly if it is an axiomatic formula of $\mathcal{S}$, substracting the number of active leaves at both sides of this equation gives:

$$\text{the number of main leaves} = \text{the number of internal axiomatic formulas} + 1.$$

This implies that there is at least one main leaf, reproving lemma 5.1.

Example 5.3 The proof structure in example 4.2.1 has $\lambda = 6$ hypotheses and $\alpha = 6$ axiomatic formulas. Neglecting the axiomatic leaves, we count 3 main leaves and 2 internal axiomatic formulas. ♦

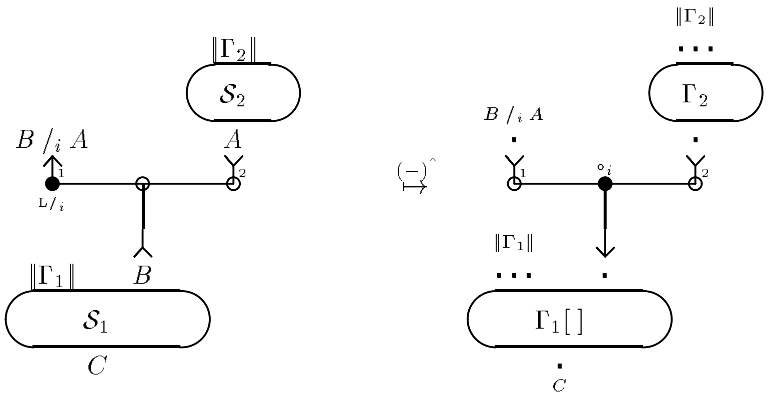
Theorem 5.4 ($\boxed{\Longleftarrow}$ part of theorem 3.7)
If there is a proof structure $\mathcal{S}$ (from $\|\Gamma\|$ to $\{C\}$) such that $\widehat{\mathcal{S}} \twoheadrightarrow_{\mathcal{R}} \Gamma_C$, then $\Gamma \vdash C$ is derivable in $\mathbf{NL}\Diamond_{\mathcal{R}}$. ♦

Proof: We apply induction on the length l of the conversion sequence $\widehat{\mathcal{S}} \twoheadrightarrow_{\mathcal{R}} \Gamma_C$.

In case $l = 0$ we have $\widehat{\mathcal{S}} = \Gamma_C$. We proceed by induction on the size of Γ.

If $\Gamma = \overset{A}{\cdot}$, then A equals C, since they originate from the same formula constituting the whole proof structure $\mathcal{S}$. Now $A \vdash A$ is derivable by Ax.

If Γ is not trivial, by lemma 5.1 we know $\mathcal{S}$ has at least one main leaf D. In the subcase that D is the main formula of a $\mathrm{L}/_i$ link, D is of the form $B /_i A$ and must be the first premisse of this link. Now $\mathcal{S}$ and Γ are of the form



By induction hypothesis there are derivations $\mathcal{D}_2$ of $\Gamma_2 \vdash A$ and $\mathcal{D}_1$ of $\Gamma_1[B] \vdash C$, which may be combined into

$$\dfrac{\begin{array}{c}\mathcal{D}_2\\ \Gamma_2 \vdash A\end{array} \qquad \begin{array}{c}\mathcal{D}_1\\ \Gamma_1[B] \vdash C\end{array}}{\Gamma_1[B /_i A \circ_i \Gamma_2] \vdash C}\ \mathrm{L}/_i$$

which is a derivation of $\Gamma \vdash C$.

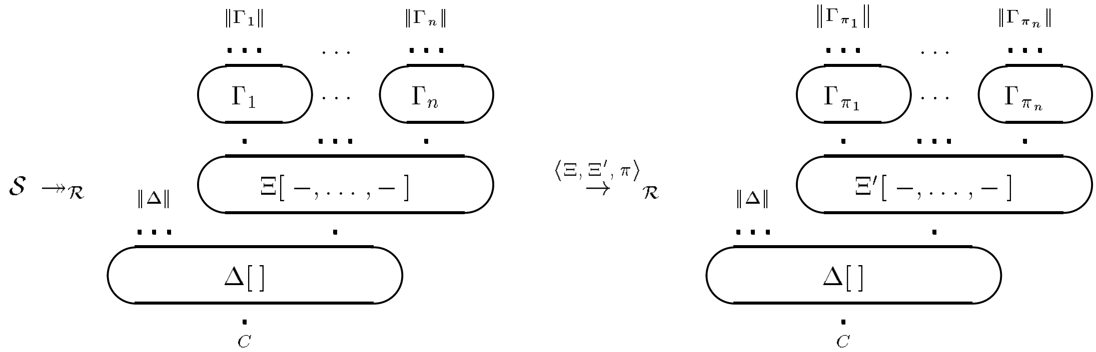
The remaining subcases that D is the main formula of a $\mathrm{R}\Diamond_j$, $\mathrm{R}\bullet_i$, $\mathrm{L}\Box^{\downarrow}{}_j$ or $\mathrm{L}\backslash_i$ link are proved similarly.

Now assume $l > 0$.

If the last conversion step is a structural conversion

$$\Xi[\, x_1, \ldots, x_n \,] \;\rightarrow\; \Xi'[\, x_{\pi_1}, \ldots, x_{\pi_n} \,].$$

our conversion sequence is

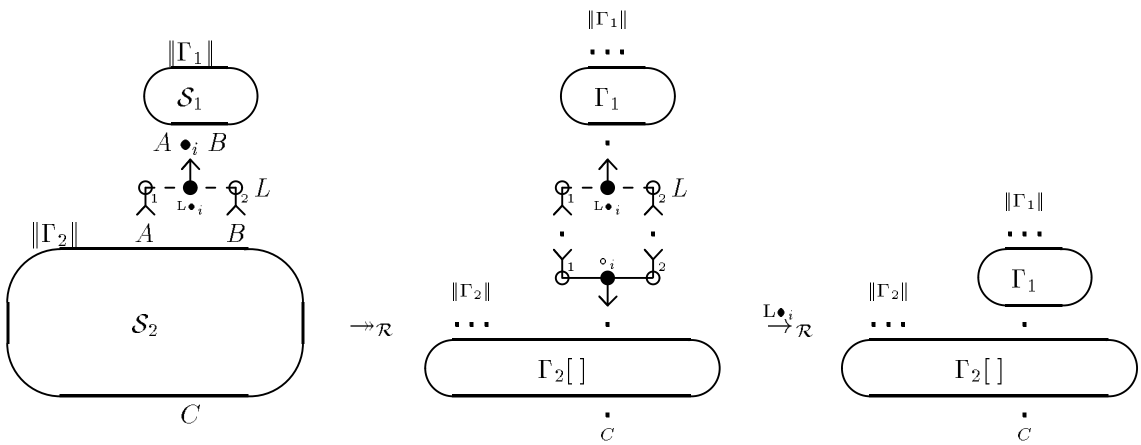


By induction hypothesis we know there is a derivation $\mathcal{D}_1$ of $\Delta[\,\Xi[\,\Gamma_1,\ldots,\Gamma_n\,]\,] \vdash C$, yielding

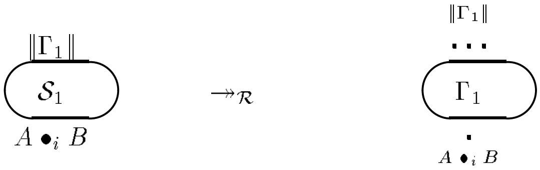
$$\cfrac{\begin{array}{c}\mathcal{D}_1\\ \Delta[\,\Xi[\,\Gamma_1,\ldots,\Gamma_n\,]\,] \vdash C\end{array}}{\Delta[\,\Xi'[\,\Gamma_{\pi_1},\ldots,\Gamma_{\pi_n}\,]\,] \vdash C}\,\langle \Xi, \Xi', \pi\rangle$$

which is a derivation of $\Gamma \vdash C$.

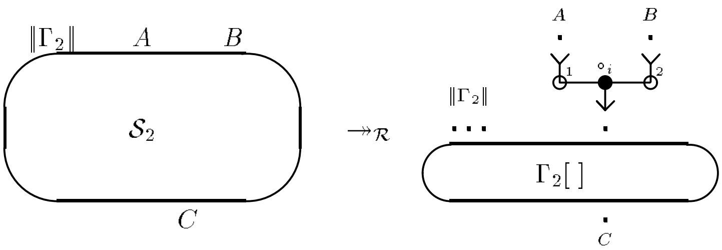
If the last conversion step is a $\mathrm{L}\bullet_i$ contraction, we can split Γ in the node by which the pair of links is replaced. This yields two trees Γ_1 and $\Gamma_2[\,]$, satisfying $\Gamma = \Gamma_2[\Gamma_1]$.



Reading this sequence from right to left, the $\mathrm{L}\bullet_i$ link (L) serves as a boundary: any structural rule that is applied is strictly above L or below L. Obviously also each point that is blown up is above or below L, whence our proof structure $\mathcal{S}$ splits as indicated. We can partition the conversion sequence into two conversion sequences for both of the substructures:



and



Now the total length of both sequences is $l-1$, whence each sequence is at most of length $l-1$, and applying the induction hypothesis we find derivations $\mathcal{D}_1$ of $\Gamma_1 \vdash A \bullet_i B$ and $\mathcal{D}_2$ of $\Gamma_2[A \circ_i B] \vdash C$. These may be combined into

$$\dfrac{\begin{array}{c}\mathcal{D}_1\\ \Gamma_1 \vdash A \bullet_i B\end{array} \qquad \begin{array}{c}\mathcal{D}_2\\ \Gamma_2[A \circ_i B] \vdash C \\ \hdashline \Gamma_2[A \bullet_i B] \vdash C\end{array}\ {\scriptstyle \mathrm{L}\bullet_i}}{\Gamma_2[\Gamma_1] \vdash C}\ {\scriptstyle \mathrm{CUT}}$$

which is a derivation of $\Gamma \vdash C$, as desired.

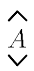
The other four contractions are treated similarly. ∎

Given $\mathcal{S}$ and $\rho : \widehat{\mathcal{S}} \twoheadrightarrow_{\mathcal{R}} \Gamma_C$, we might find different derivations of $\Gamma \vdash C$ by this proof. This non-uniqueness lies in the possible several main leaves in the case where $l = 0$ and Γ is non-trivial.

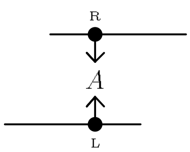
However, any such derivation $\mathcal{D}$ satisfies the property that there is a bijective correspondence between the links of $\mathcal{S}$ and the logical rules of $\mathcal{D}$. Observe that each par link of $\mathcal{S}$ also corresponds to a contraction in ρ. The remaining conversions, i.e. the structural conversions, correspond to structural rules of $\mathcal{D}$. What about the identity rules of $\mathcal{D}$?

We claim that is it possible to adapt the proof of theorem 5.4 in such a way that all derivations $\mathcal{D}$ of $\Gamma \vdash C$ we find satisfy the following in addition to the above property:

- there is a bijective correspondence between axiomatic formulas and AX rules, such that

  corresponds with an AX rule on A;
- there is a bijective correspondence between cut formulas and CUT rules, such that

  corresponds with a CUT rule on A.

For this purpose we first state the following lemma.

Lemma 5.5 (Substitution) Let $\mathcal{D}_1$ be a derivation of $\Gamma_1 \vdash C_1$ and $\mathcal{D}_2$ be a derivation of $\Gamma_2[C_1] \vdash C_2$.
1. If $C_1 \vdash C_1$ is an axiom of $\mathcal{D}_1$, the succedent formula of which coincides with the succedent formula of $\Gamma_1 \vdash C_1$, then we can substitute $\mathcal{D}_2$ into $\mathcal{D}_1$ in order to get a derivation $\mathcal{D}_1[\mathcal{D}_2]$ of $\Gamma_2[\Gamma_1] \vdash C_2$.

$$
\begin{array}{c}
\begin{array}{ccc}
\dfrac{}{C_1 \vdash \boldsymbol{C_1}} \quad \ldots & & \\
\mathcal{D}_1 \quad \ddots \;\; \vdots \;\; \iddots & & \mathcal{D}_2 \\
\Gamma_1 \vdash \boldsymbol{C_1} & & \Gamma_2[C_1] \vdash C_2
\end{array} \\
\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
\Gamma_2[\Gamma_1] \vdash C_2
\end{array}
\quad \text{becomes} \quad
\begin{array}{c}
\mathcal{D}_2 \\
\boldsymbol{\Gamma_2}[C_1] \vdash \boldsymbol{C_2} \quad \ldots \\
\mathcal{D}_1 \quad \ddots \;\; \vdots \;\; \iddots \\
\boldsymbol{\Gamma_2}[\Gamma_1] \vdash \boldsymbol{C_2}
\end{array}
$$

2. If $C_1 \vdash C_1$ is an axiom of $\mathcal{D}_2$, the antecedent formula of which coincides with the occurrence in $\Gamma_2[C_1] \vdash C_2$, then we can substitute $\mathcal{D}_1$ into $\mathcal{D}_2$ in order to get a derivation $\mathcal{D}_2[\mathcal{D}_1]$ of $\Gamma_2[\Gamma_1] \vdash C_2$.

$$
\begin{array}{c}
\begin{array}{ccc}
 & & \ldots \quad \dfrac{}{\boldsymbol{C_1} \vdash C_1} \\
\mathcal{D}_1 & & \ddots \;\; \vdots \;\; \iddots \quad \mathcal{D}_2 \\
\Gamma_1 \vdash C_1 & & \Gamma_2[\boldsymbol{C_1}] \vdash C_2
\end{array} \\
\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
\Gamma_2[\Gamma_1] \vdash C_2
\end{array}
\quad \text{becomes} \quad
\begin{array}{c}
\mathcal{D}_1 \\
\ldots \quad \boldsymbol{\Gamma_1} \vdash C_1 \\
\ddots \;\; \vdots \;\; \iddots \quad \mathcal{D}_2 \\
\Gamma_2[\boldsymbol{\Gamma_1}] \vdash C_2
\end{array}
$$

♦

Proof: In general every leaf of a tree determines a branch to the root. In particular every axiom rule of a derivation determines a branch of sequents from that axiom to the conclusion of the derivation. Let $\Gamma \vdash \Delta$ and $\Gamma' \vdash \Delta'$ be two successive sequents in a certain branch β, i.e. $\Gamma' \vdash \Delta'$ is the conclusion of an inference rule with $\Gamma \vdash \Delta$ among its hypotheses. For a binary inference rule R we say that β passes R via the left (right) hypothesis if $\Gamma \vdash \Delta$ is the first (second) hypothesis of R.

1. As the occurrence $\boldsymbol{C_1}$ is preserved in the branch β in $\mathcal{D}_1$ between $C_1 \vdash \boldsymbol{C_1}$ and $\Gamma_1 \vdash \boldsymbol{C_1}$, the possible inference rules β passes are Cut (via the right hypothesis), the left logical rules (if binary, then via the right hypothesis), or a structural rule. Each of these rules have the property that if

$$\frac{(\;\Gamma_0 \vdash C_0\;) \qquad \Gamma_1 \vdash \boldsymbol{C_1}}{\Gamma_3 \vdash \boldsymbol{C_1}}$$

is an instance, then so is

$$\frac{(\;\Gamma_0 \vdash C_0\;) \qquad \boldsymbol{\Gamma_2}[\Gamma_1] \vdash \boldsymbol{C_2}}{\boldsymbol{\Gamma_2}[\Gamma_3] \vdash \boldsymbol{C_2}}$$

2. As the occurrence $\boldsymbol{C_1}$ is preserved in the branch β in $\mathcal{D}_2$ between $\boldsymbol{C_1} \vdash C_1$ and $\Gamma_2[\boldsymbol{C_1}] \vdash C_2$, it will never be an active formula in any inference rule β passes. Hence if

$$\frac{(\;\Gamma_0 \vdash C_0\;) \qquad \Gamma_2[\boldsymbol{C_1}] \vdash C_2}{\Gamma_3[\boldsymbol{C_1}] \vdash C_3}$$

is an instance of a rule, then so is

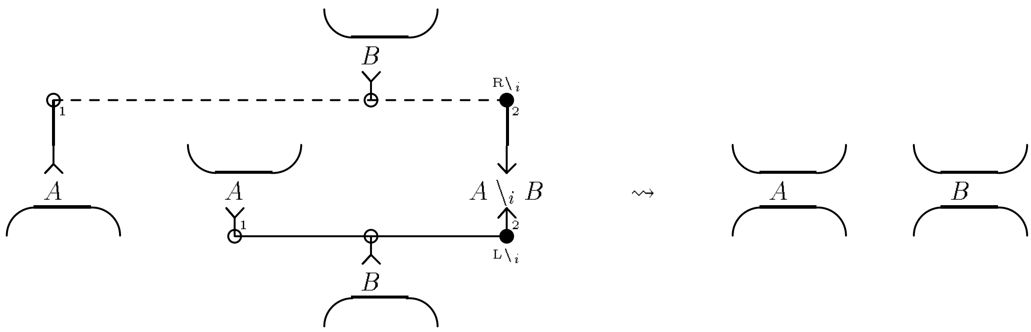
$$\frac{(\quad \Gamma_0 \vdash C_0 \quad) \qquad \Gamma_2[\mathbf{\Gamma_1}] \vdash C_2}{\Gamma_3[\mathbf{\Gamma_1}] \vdash C_3}$$

■

Now extend the proof of theorem 5.4 by simultaniously showing that every axiomatic formula corresponds to an Ax rule, and moreover that every axiomatic conclusion corresponds to an axiom as in lemma 5.5.1 and every axiomatic hypothesis corresponds to an axiom as in lemma 5.5.2. We adapt the proof in the case that $l > 0$ and the last conversion step is a contraction: If the main formula of L (say: D) is a cut formula of $\mathcal{S}$ we proceed as described earlier. However, if D is not a cut formula, then D is an axiomatic leaf of one of the two substructures, whence we can apply the par rule followed by the appropriate substitution.

6 Cut elimination

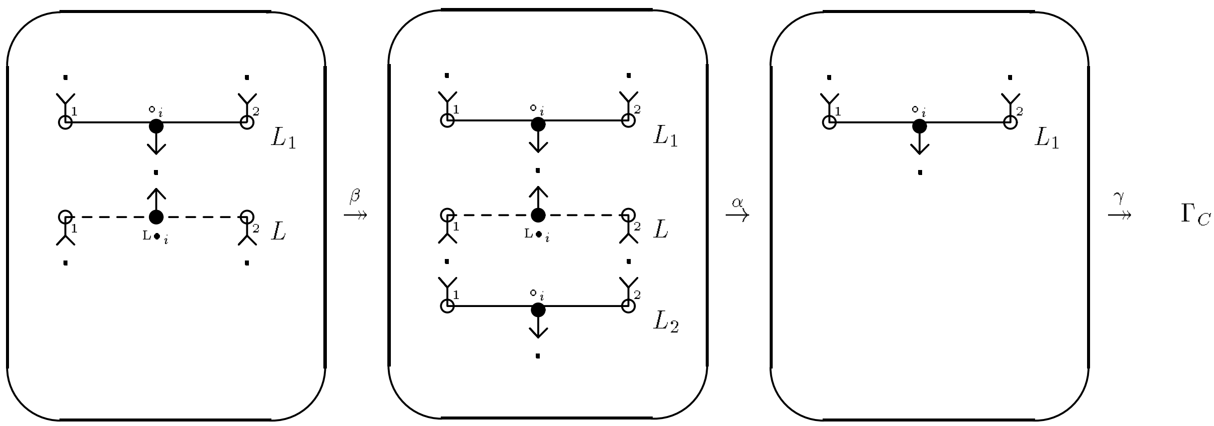
Remember that a cut formula is a formula which is the main formula of two dual links. A cut reduction step is defined by deleting these links and the cut formula, while pairwise identifying the active formulas in case they are different (as occurrence of the same formula), or deleting them if they are identical.



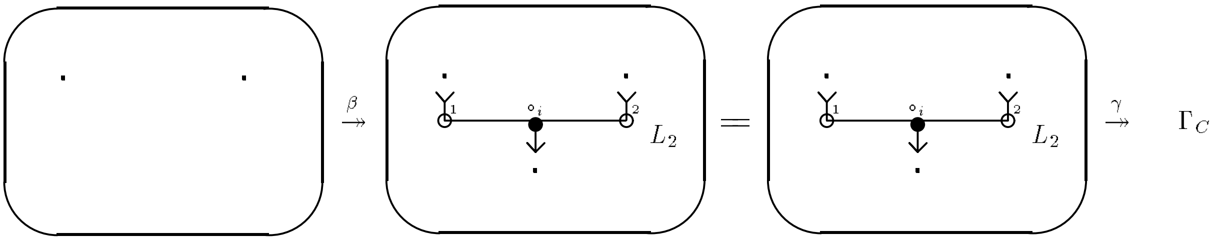
Let D be a cut formula and L the corresponding par link. We will show that if $(\mathcal{S}, \rho)$ is a conversion sequence of $\Gamma \vdash C$, then so is $(\mathcal{S}', \rho')$, where $\mathcal{S} \rightsquigarrow \mathcal{S}'$ and ρ' consist of the same set of conversion steps as ρ, except the contraction α of L, in a sense to be made precise shortly.

Theorem 6.1 (Cut elimination) If $\mathcal{S}$ is a proof net of $\Gamma \vdash C$, and $\mathcal{S} \rightsquigarrow \mathcal{S}'$ by a cut reduction step, then $\mathcal{S}'$ is a proof net of $\Gamma \vdash C$ as well. ♦

Example 6.2 1. In order to get an idea of the proof, let us consider the following conversion sequence in which we assume L_1 remains untouched during β.

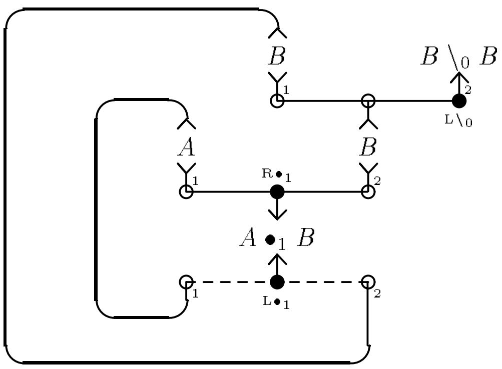


Executing a cut reduction step yields a proof structure to which we can apply the same conversion steps as before and in the same order, except the contraction α:

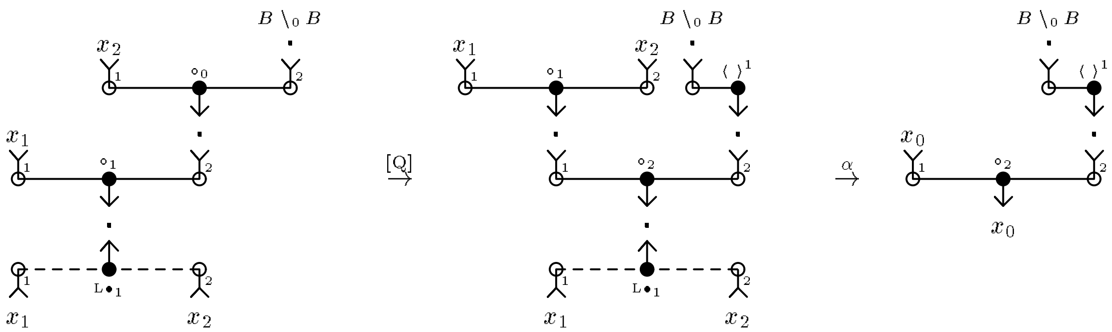


Observe that L_2 plays the role of L_1 in γ.

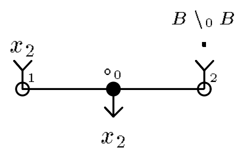
2. If L_1 does not remain untouched, we can not in general transform an arbitrary conversion sequence $\widehat{\mathcal{S}} \overset{\beta}{\twoheadrightarrow} \mathcal{H}_1 \overset{\alpha}{\to} \mathcal{H}$ into $\widehat{\mathcal{S}'} \overset{\beta'}{\twoheadrightarrow} \mathcal{H}$. This is shown by the following proof net in the calculus $\mathbf{NL}\diamond_{\{[\mathrm{Q}]\}}$, where [Q] may be read off from the conversion step below.



This proof structure may be converted as follows:



but after cut elimination we obtain

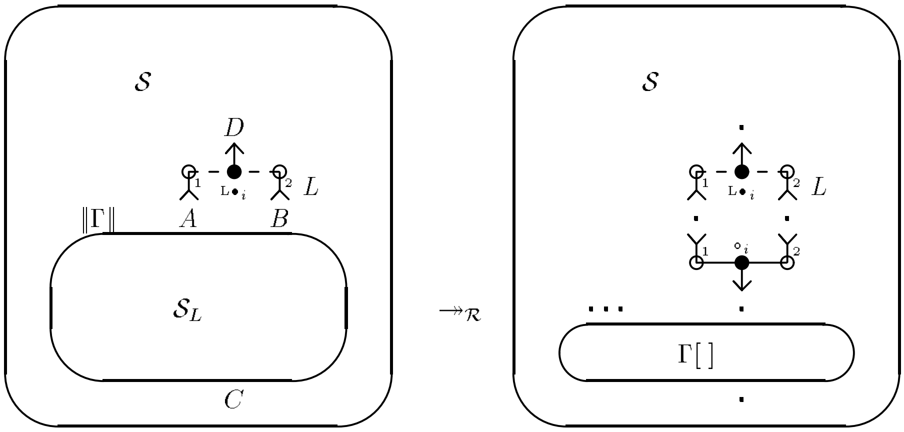


and no conversion step applies anymore, whence we cannot arrive at the same hypothesis structure as before. ♦

Let $(\mathcal{S}, \rho)$ be a conversion sequence of $\Gamma \vdash C$:

$$\widehat{\mathcal{S}} \overset{\rho}{\twoheadrightarrow}_{\mathcal{R}} \Gamma_C$$

For each par link L with main formula D and active formula(s) A (and B) we will define a substructure $\mathcal{S}_L$ of $\mathcal{S}$ (called the *block* of L in $\mathcal{S}$ w.r.t. ρ) and a subsequence ρ_L satisfying the following properties:



- $\mathcal{S}_L$ does not contain D. Consequently it does not contain L either;
- $\mathcal{S}_L$ has A (and B) among its leaves.
- The conversion sequence ρ_L acts completely within $\mathcal{S}_L$, and this restriction to $\mathcal{S}_L$ yields a *nice* hypothesis tree with respect to L; i.e. attaching L enables its contraction α.
- Our original conversion sequence may be replaced by

$$\widehat{\mathcal{S}} \overset{\rho_L}{\twoheadrightarrow}_{\mathcal{R}} \mathcal{H} \overset{\alpha}{\to} \mathcal{H}' \overset{\rho'_L}{\twoheadrightarrow}_{\mathcal{R}} \Gamma_C$$

We will only sketch the idea; the formal definition and proof may be given simultaniously by induction on the length l of ρ.

First of all, deleting all p par links (but not their nodes) yield $p+1$ hypothesis trees, called the *components* of $\mathcal{S}$. This even holds for all intermediate hypothesis structures between $\widehat{\mathcal{S}}$ and Γ_C: Reasoning backwards from the hypothesis tree, we start with one component ($p = 0$). After a number of structural conversions, a contraction $\mathcal{H} \overset{\alpha}{\to} \Gamma'$ splits this component Γ' into two parts and replaces one node by a redex. The par link L of this redex now serves as a boundary between the two new components Γ_1 (attached to A (and B)) and Γ_2 (attached to D), while (at this moment) Γ_1 is a *nice* hypothesis tree w.r.t. L. All next structural conversions take place completely within one of the two components, and the next contraction takes place in exactly one of the two components as well. In this way every par link replaces one component by two new components, yielding $p+1$ components in each hypothesis structure, and $2p+1$ distinct components in the whole conversion sequence so far (read from rigth to left).

The block of a par link L is Γ_1 at stage $\mathcal{H}$, and further on it grows with the remaining conversions in ρ; it is clear that every conversion is completely inside the block, or completely outside the block, which proves our properties.

This shows we can reorder our original conversion sequence ρ by ρ_1 in which L_1 remains untouched until α (cf. example 6.2.1). Executing a cut reduction step yields a proof structure to which ρ_1 applies until α, and further on ater α.

7 Automated deduction

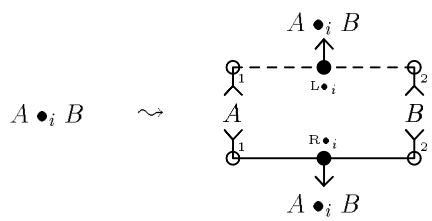
One of the attractive aspects of our formalism is that it lends itself well to automated proof search. First of all, in the previous section we saw that we could eliminate cut formulas from proof nets, making it unnecessary to consider cut formulas in our proof search. Secondly, we can restrict ourselves to proof nets where all our axiomatic formulas are atomic, as indicated by the following lemma:

Lemma 7.1 Given a proof structure $\mathcal{S}$ we can construct a proof structure $\mathcal{S}'$ with the same hypotheses and conclusions where all axiomatic formulas are atomic and where $\widehat{\mathcal{S}'} \twoheadrightarrow_{\mathcal{R}} \widehat{\mathcal{S}}$, for an arbitrary set of structural conversions $\mathcal{R}$. We will call such a proof structure *eta-expanded.* ♦

Proof: By induction on the total complexity of the axiomatic formulas.

If there are no complex axiomatic formulas in the proof structure, we take $\mathcal{S}' = \mathcal{S}$ and an empty conversion sequence.

If we have a proof structure $\mathcal{S}_0$ where the axiomatic formulas have $n+1$ total connectives, we can expand a complex axiomatic $A \bullet_i B$ formula in the following way (the other connectives are treated similarly):



The resulting proof structure $\mathcal{S}_1$ will have two new axiomatic formulas and the total number of connectives of axiomatic formulas will be n.

By induction hypothesis we know that $\widehat{\mathcal{S}}_1' \twoheadrightarrow_{\mathcal{R}} \widehat{\mathcal{S}}_1$, so we can suffix a $\mathrm{L}\bullet_i$ conversion producing $\widehat{\mathcal{S}}_1' \twoheadrightarrow_{\mathcal{R}} \widehat{\mathcal{S}}_1 \stackrel{\mathrm{L}\bullet_i}{\to} \widehat{\mathcal{S}}_0$. As we use only contractions, the theorem holds regardless of the structural rules. ∎

As an immediate consequence of cut elimination and lemma 7.1 we get the following corollary:

Corollary 7.2 For every $\mathcal{R}$-proof net $\mathcal{P}$ of $\Gamma \vdash C$ there exists a $\mathcal{R}$-proof net $\mathcal{P}'$, also of $\Gamma \vdash C$, which is cut-free and eta-expanded. ♦

So we can, without loss of generality, restrict ourselves to proof structures where all complex formulas are neither axiomatic nor cut formulas. A simple algorithm for the enumeration of cut-free, eta-expanded proof nets is the following.

Input
- sequence $w_1, \ldots, w_n$ of words
- lexicon l, which assigns formulas to words
- goal formula Q
- set $\mathcal{R}$ of structural rules

Output set of cut-free, eta-expanded proof nets from $\{l(w_1), \ldots, l(w_n)\}$ to $\{Q\}$

Usually, we want to restrict this set of proof nets to those satisfying some additional constraints, like that left to right traversal of the hypothesis tree yields the formulas in the order indicated by the input sequence.

1. For each of the words w_i in the input sequence, select one of the formulas assigned to this word from the lexicon.
2. Decompose the formulas according to the links of definition 3.1 until we reach the atomic subformulas. The disjoint union of these proof structures is itself a proof structure, though it will have several hypotheses in addition to those from the lexicon and several conclusions in addition to the goal formula.
3. Identify each atomic premiss with an atomic conclusion to produce a proof structure from $\{l(w_1), \ldots, l(w_n)\}$ to $\{Q\}$.
4. Convert the hypothesis structure corresponding to this proof structure to a hypothesis tree using only the structural conversions of $\mathcal{R}$ and the contractions.
5. Check if this hypothesis tree satisfies our constraints.

We assume computation is nondeterministic, i.e. the steps of our algorithm can produce a number of solutions: the lexicon can produce different formulas for each word, there can be many different ways of identifying the atomic formulas and many hypothesis trees to which we can convert our hypothesis structure. When one step in our algorithm fails to produce a solution, we backtrack to a previous step and try the next solution there until we have found all solutions.

As the connecting of step 3 and the conversions of step 4 are computationally expensive it is desirable to do some static tests on the set of proof structures we get from the lexical formulas after step 2 of the algorithm to make sure we at least have a chance of ultimately converting to a hypothesis tree. The following are two quick tests to reject proof structures which can never satisfy our correctness criterion.

First, by our definition of hypotheses and conclusions of proof structures, all atomic formulas other than lexical formulas or the conclusion Q must be both a premiss and a conclusion of some link in a proof structure from $\{l(w_1), \ldots, l(w_n)\}$ to $\{Q\}$. So we can count if each of these atomic formulas occurs as many times as a conclusion as it occurs as a premiss.

Secondly, the following lemma gives us a condition on the number of binary links occurring in a proof net.

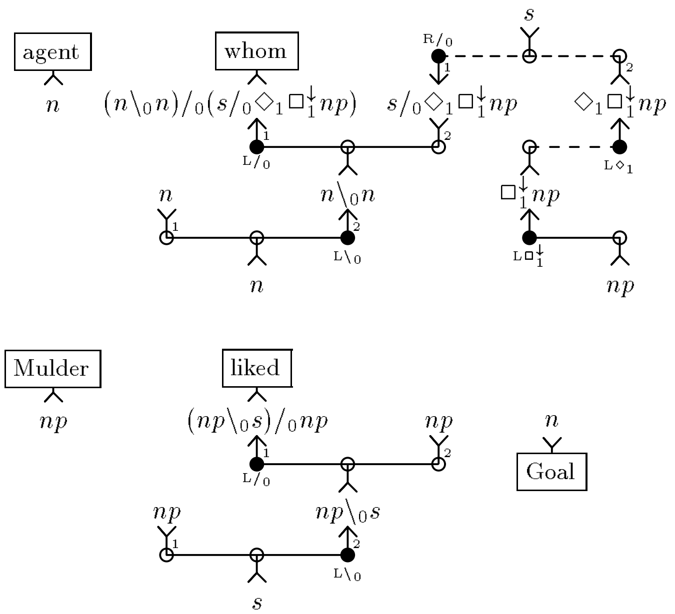
Lemma 7.3 Suppose we have a proof structure $\mathcal{S}$ with h hypotheses, t binary tensor links, p binary par links and a single conclusion. Then the following holds if $\mathcal{S}$ is a proof net:

$$t + 1 = p + h$$

♦

Proof: Reasoning backwards from the hypothesis tree to the initial hypothesis structure we see that it holds for the hypothesis tree (with $p = 0$), that the structural conversions and the unary contractions preserve t, p and h and that the contractions for the binary links increase t and p simultaneously. ■

Example 7.4 The proof structure we would get after lexical lookup and formula decomposition of 'agent whom Mulder liked' is the following:



We have marked the lexical hypotheses (resp. the conclusion) with an arrow leaving the formula from above (resp. below). This is just a reminder they should not be used as the conclusion (resp. premiss) of a link.

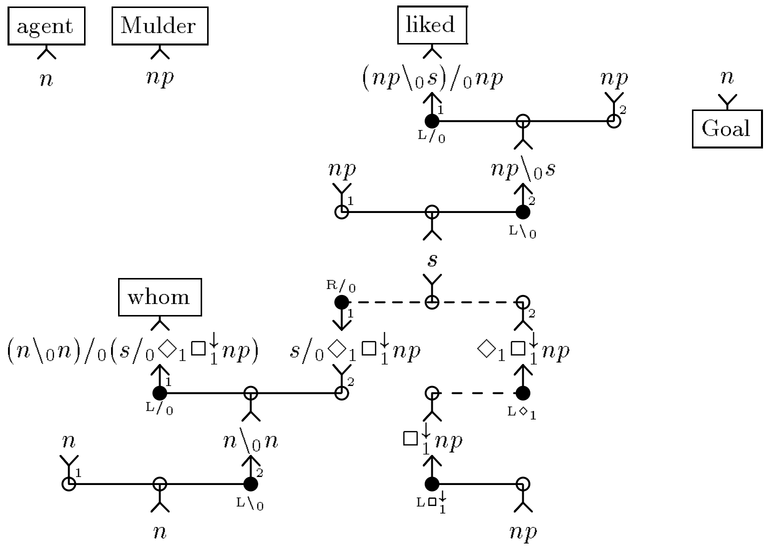
Now, we count the atomic premisses and conclusions which are available. We have two formulas n as premisses and two as conclusions, we have one formula s as a premiss and one as a conclusion and we have two formulas np as premisses and two as conclusions. So it should at least be possible to find a way to identify these and produce a proof structure from $\{n, (n\backslash_0 n)/_0(s/_0 np), np, (np\backslash_0 s)/_0 np\}$ to $\{n\}$.

Next we count 4 lexical hypotheses, 4 binary tensor links and 1 binary par link, satisfying our equation $t + 1 = p + h$ and proceed to the identification phase.

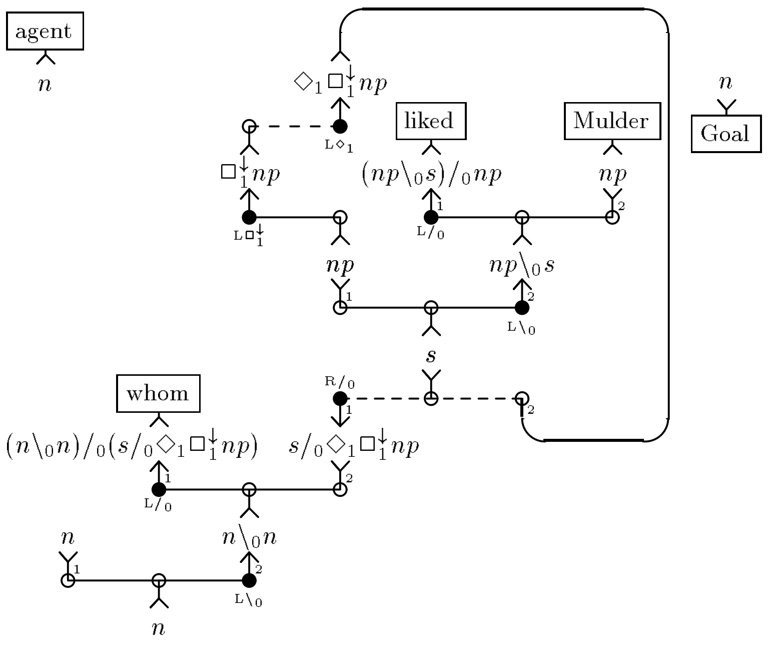
The proof structure for '*agent whom Mulder liked Skully' would fail both tests. There would be one more np occurrence as a conclusion and there would be one hypothesis too many. ♦

For the identification of the atomic formulas, we will generally have a number of possible choices. Given a proof structure where an atomic formula A occurs k times as a premiss (and k times as a conclusion, according to our count check), there will be $k!$ ways of performing these identifications. In the proof structure above we count 1 formula s, 2 formulas np and 2 formulas n, giving us a total of $1! \times 2! \times 2! = 4$ proof structures we have to consider. For larger examples, this will quickly lead to an unacceptably long computation time. However, we will see that a smart algorithm can in many cases perform better than the worst case complexity might suggest, by exploiting the different kinds of information present in the proof structure.

Example 7.5 As a first step, we identify the premiss s with the conclusion, resulting in the following proof structure:



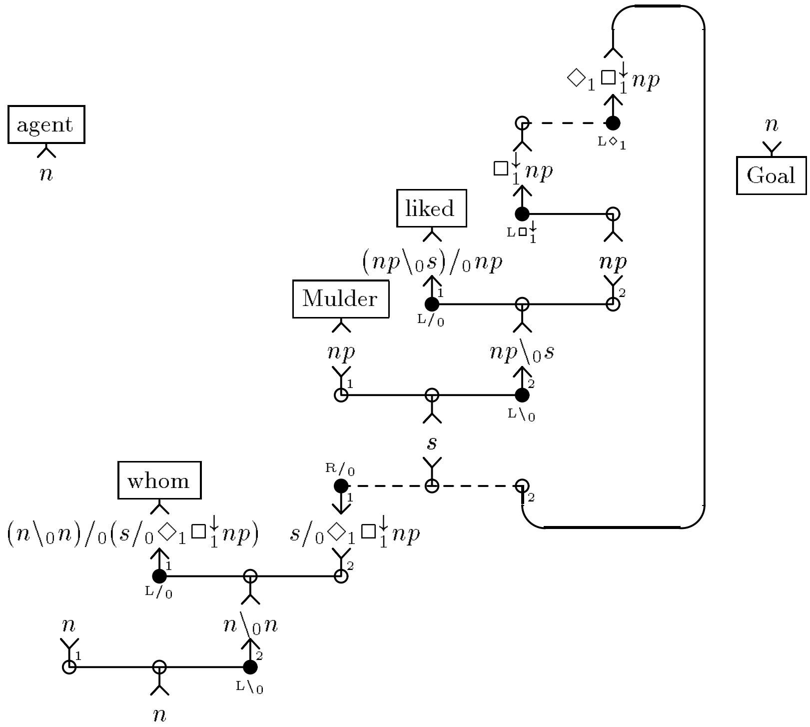
Next, we decide to identify the np formulas. Here we have two choices, either the lexical np formula is a premiss of the $L\backslash_0$ link or it is a premiss of the $L/_0$ link. We choose to explore the second possibility first, and get the following proof structure:



Now, when we look at the hypothesis structure of this proof structure, we see that no structural conversions apply to this hypothesis structure and that the only contraction we can apply is the one for $L\Diamond_1$, after which we will be unable to contract the $R/_0$ link or apply any structural conversion. Furthermore, this will apply to any hypothesis structure we get after identifying more formulas of the proof structure, because the $R/_0$ link remains splitting in the sense of our sequentialization theorem, regardless of which additional formulas we identify in this proof structure.

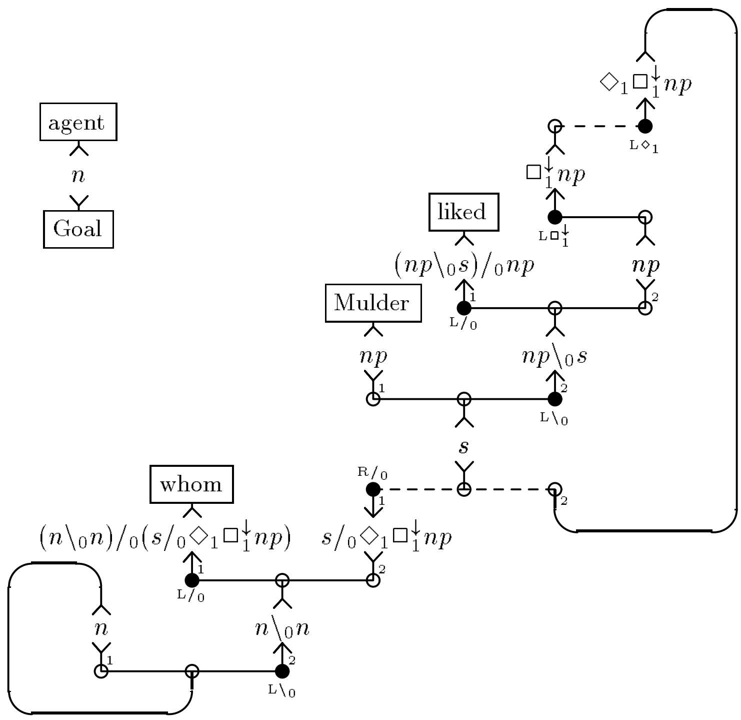
Observe also that if we would have used 'which' instead of 'whom' for this example, we could have applied the $[\mathrm{Com}_{0,0}]$ conversion at this point and continued to produce a proof net of 'agent which liked Mulder'.

We return to the other way of identifying the np formulas, resulting in the following proof structure:



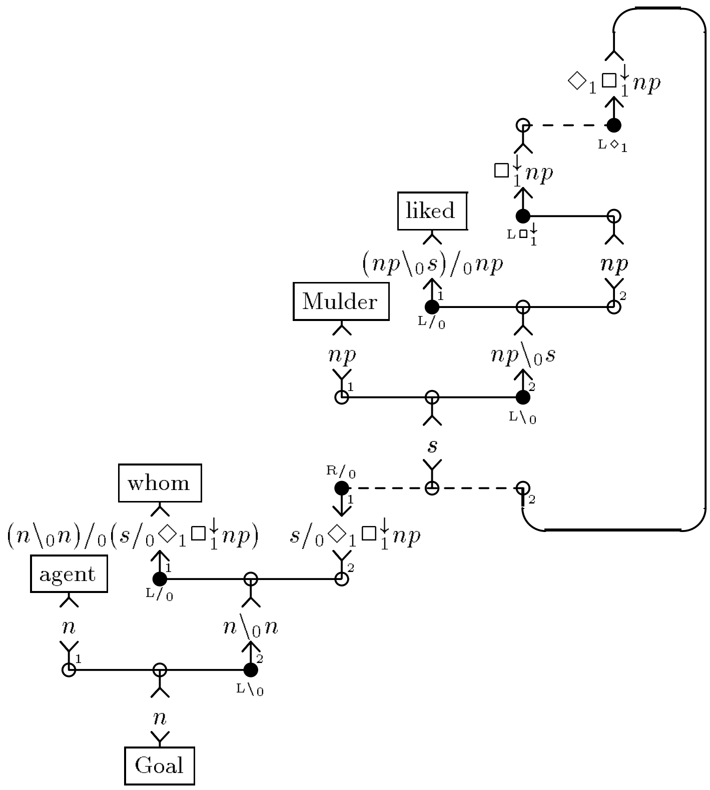
Applying the same reasoning as before, we see that at this point we *can* contract the $R/_0$ link after the $[Ass_{0,1}]$ structural conversion and the $L\Diamond_1$ contraction, so continuing is warranted.

For the n formulas, we again have two choices: we either identify the n hypothesis with the conclusion of the proof structure or with the conclusion of the $L\backslash_0$ link. Selecting the first option will give us the following proof structure:



Now, we can immediately see we will be unable to contract the hypothesis structure of this proof structure to a tree, as none of our conversions allow us to connect the two separate components of the hypothesis structure and therefore we can never convert to a tree.

Fortunately, the final proof structure of this sequent



can be converted to a hypothesis tree (almost) as shown in example 3.11. We can also see that this hypothesis tree has all formulas in the desired order.

Given our small example fragment, this completes the enumeration of all possible cut-free, eta-expanded proof nets from $\{n, (n\backslash_0 n)/_0(s/_0\Diamond_1\Box_1^{\downarrow}np), np, (np\backslash_0 s)/_0 np\}$ to $\{n\}$. ♦

8 Lambek Calculus

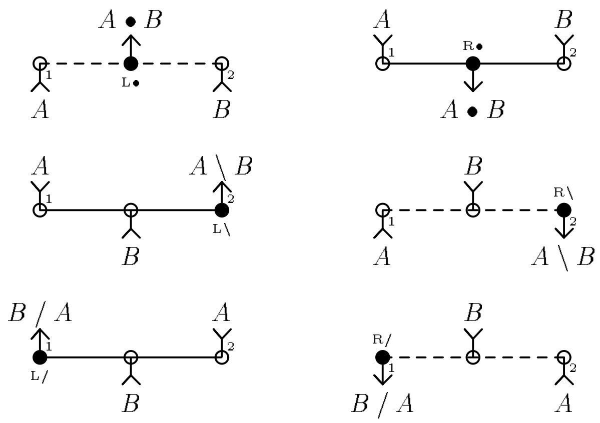
Lambek Calculus (**L**) as introduced in [Lambek 58] is defined by the inference rules of figure 2, where the antecedent part of each sequent is a sequence rather than a structure tree. It is related to the following special case of $\mathbf{NL}\Diamond_{\mathcal{R}}$;

- zero unary modes ($J = \emptyset$), implying no unary connectives;
- only one binary mode ($I = \{*\}$; we will henceforth delete the unique subscript);
- the structural rules[3] [LAss] and [RAss], mimicking the fact that each sequent has a sequence instead of a structure tree as antecedent part;
- no other structural rules.

This calculus is equivalent to $\mathbf{L}^{>0}$, by which we mean **L** restricted to the requirement that the antecedent sequence of all sequents in a derivation be non-empty. In this way theorem 3.7 provides us with a correctness criterium for $\mathbf{L}^{>0}$.

However, a much more attractive correctness criterion is obtained when we adapt our theory in such a way that the structural rules become part of the theory and are not present explicitly anymore. This is done by a generalization of the links in the definition of correction structure.

Definition 8.1 An **L**-*proof structure* $\langle S, \mathcal{L}\rangle$ consists of a finite set S of ($\Diamond$- and $\Box^{\downarrow}$-free and unimodal) formulas together with a set $\mathcal{L}$ of links in S of the following forms:



such that the following holds:
• every formula of S is at most once a conclusion of a link;
• every formula of S is at most once a premiss of a link. ♦

Definition 8.2 An **L**-*correction structure* $\langle N, \mathcal{L}\rangle$ consists of a finite set N of nodes together with a set $\mathcal{L}$ of links in N of the following forms (where n varies over $n = 0$ or $n \geq 2$):

[3] See page 5 for the definition of the mentioned structural rules.

Identity rules

$$\frac{}{A \vdash A}\ \text{Ax}$$

$$\frac{\Gamma \vdash A \qquad \Delta_1, A, \Delta_2 \vdash C}{\Delta_1, \Gamma, \Delta_2 \vdash C}\ \text{Cut}$$

Logical rules for the $\otimes$-like connectives

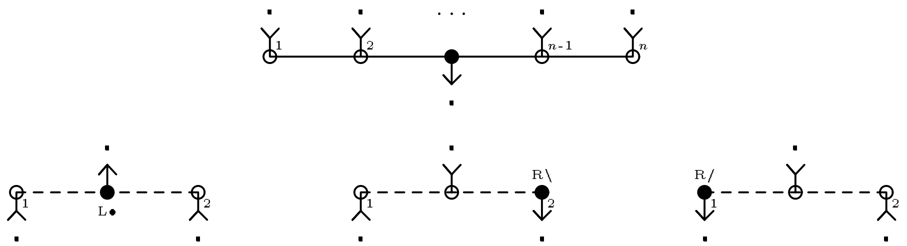
$$\frac{\Gamma_1, A, B, \Gamma_2 \vdash C}{\Gamma_1, A \bullet B, \Gamma_2 \vdash C}\ \text{L}\bullet \qquad \frac{\Gamma \vdash A \qquad \Delta \vdash B}{\Gamma, \Delta \vdash A \bullet B}\ \text{R}\bullet$$

Logical rules for the ⅋-like connectives

$$\frac{\Gamma \vdash A \qquad \Delta_1, B, \Delta_2 \vdash C}{\Delta_1, \Gamma, A \setminus B, \Delta_2 \vdash C}\ \text{L}\setminus \qquad \frac{A, \Gamma \vdash B}{\Gamma \vdash A \setminus B}\ \text{R}\setminus$$

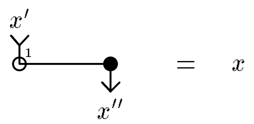
$$\frac{\Gamma \vdash A \qquad \Delta_1, B, \Delta_2 \vdash C}{\Delta_1, B / A, \Gamma, \Delta_2 \vdash C}\ \text{L}/ \qquad \frac{\Gamma, A \vdash B}{\Gamma \vdash B / A}\ \text{R}/$$

Figure 2: The sequent calculus **L**



such that the following holds:
• every node of S is at most once a conclusion of a link;
• every node of S is at most once a premiss of a link. ♦

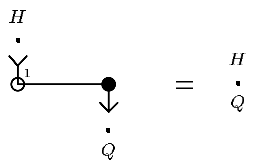
The generalized tensor link will be called an n-comb. Note that we admit $n = 0$. We define a 1-comb to be a single node:



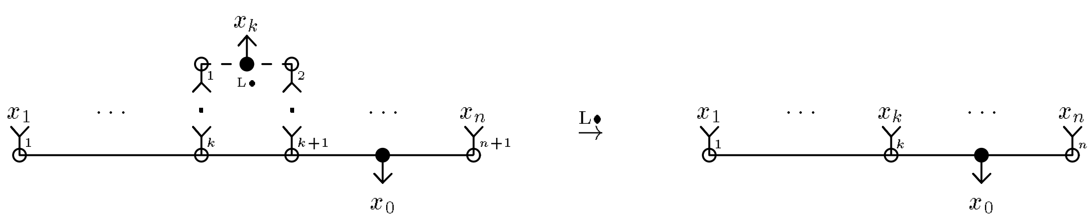
if x' and x'' are distinct nodes, and empty otherwise.

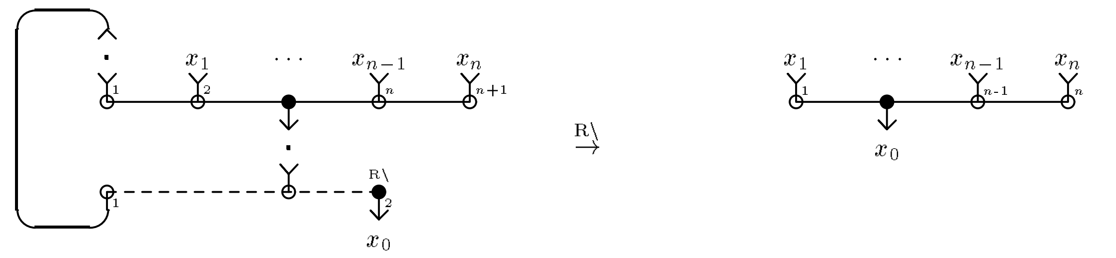
An **L**-hypothesis structure is the same as in definition 3.4, but now w.r.t. an **L**-correction structure. Henceforth we will omit the prefix **L**.

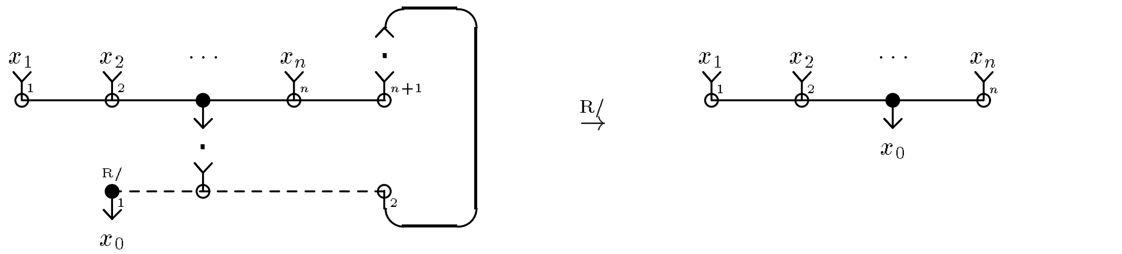
The redex of a *contraction* consists of a par link and an n+1-comb, as depictured below (where we require — as usual — all nodes to be distinct). Observe that in any case the par link is attached to two successive formulas of the n+1-comb, when we order them in a cyclic way. It is replaced by an n-comb, and all nodes keep their labels, which for $n = 1$ has to be interpreted as



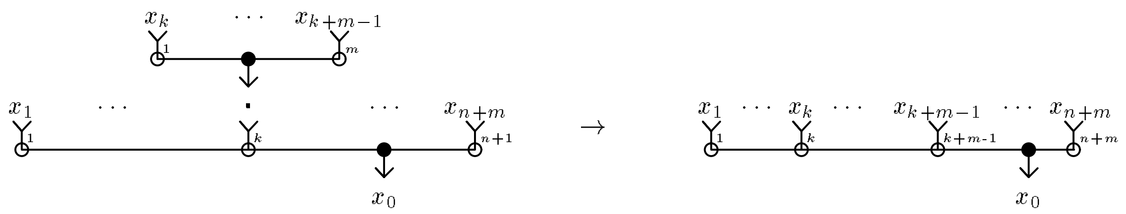
The contraction will be named after the par link (L•, R\, R/). A L• contraction only applies if $n + 1 \geq 2$.



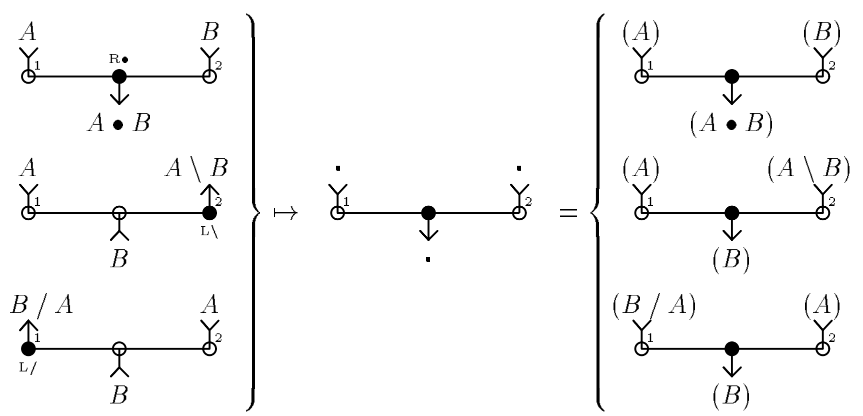




By a *structural conversion* we mean the following composition of combs:



Now, starting with a proof structure $\mathcal{S}$, we can form the underlying hypothesis structure $\widehat{\mathcal{S}}$ in the usual way (which — besides nodes — consists of par links and 2-combs only):



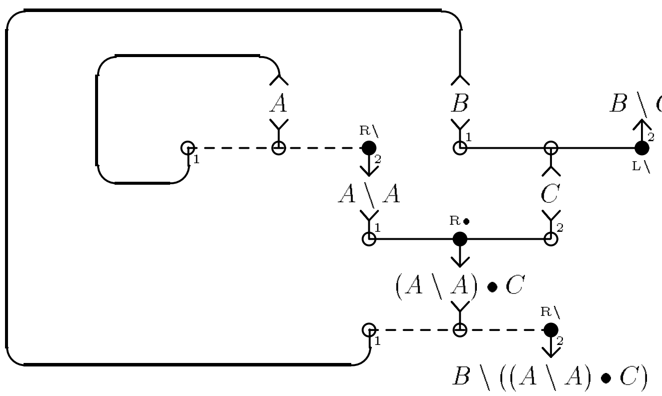
For any (possible empty) sequence Γ and formula C, let $\|\Gamma\|$ be the multiset of elements in Γ; let Γ_C be the obvious hypothesis structure (consisting of one n-comb) from $\|\Gamma\|$ to $\{C\}$ with conclusion node (lower) labeled by C. Any hypothesis structure of this form will be called a *hypothesis comb*. Let $\twoheadrightarrow$

be the transitive, reflexive closure of $\rightarrow$, by which we mean the contractions as well as the structural conversions.

Theorem 8.3
$\Gamma \vdash C$ is derivable in **L** if and only if there is a proof structure $\mathcal{S}$ (from $\|\Gamma\|$ to $\{C\}$) such that $\widehat{\mathcal{S}} \twoheadrightarrow \Gamma_C$.
♦

The proof is similar to that of theorem 3.7.

Example 8.4 The following proof structure converts to a point, showing that $B\backslash C \vdash B\backslash((A\backslash A)\bullet C)$ is derivable in **L**. This example illustrates the degenerate instances of n-combs, where n equals 0 or 1.



♦

9 Discussion

We have presented a proof net calculus for the multimodal Lambek Calculus which is new, elegant and very general.

The formalism we have presented here is related to a number of other proposals, notably to Danos' graph contractions [Danos 90], of which our contractions are a special case. As a result, acyclicity and connectedness are a consequence of our correctness criterion.

Our approach is also related to the labeled proof nets of [Moortgat 97]. Our hypothesis structures correspond closely to the labels Moortgat assigns to proof nets. Advantages of our formalism are that we have a very direct correspondence between proof structures and their hypothesis structures and that we can handle cyclic or disconnected proof structures unproblematically, whereas only acylic and connected proof structures can be assigned a meaningful label.

Some interesting questions remain open. It seems possible, for instance, to formulate a classical version of our proof net calculus, though it is unclear to us at the moment what kind of sequent calculus a (multimodal) classical Lambek Calculus would have.

We would also like to have some natural notion of equality on conversion sequences. As defined now, a number of uninteresting permutations are possible in conversion sequences, which is somewhat against the spirit of proof nets as 'sequent proofs modulo permutations of inferences'.

A Structure Trees

Given sets J and I of unary respectively binary modes, we define the set of *structure trees with holes over a set* S as follows:

$$\begin{aligned} \mathsf{trees}_{\mathsf{S}} ::= &\ \mathsf{S} \cup \{\, [\,] \,\} \\ &\mid\ \langle\, \mathsf{trees}_{\mathsf{S}} \,\rangle^{J} \\ &\mid\ \mathsf{trees}_{\mathsf{S}} \ \circ_I \ \mathsf{trees}_{\mathsf{S}} \end{aligned}$$

The *length* $\lambda(\Xi)$ and the *number of holes* $\kappa(\Xi)$ of such a tree Ξ is defined by

$$\begin{aligned} \lambda(A) &:= 1 \qquad (A \in \mathsf{S}) \\ \lambda([\,]) &:= 1 \\ \lambda(\langle\, \Xi_1 \,\rangle^j) &:= \lambda(\Xi_1) \\ \lambda(\Xi_1 \ \circ_i \ \Xi_2) &:= \lambda(\Xi_1) + \lambda(\Xi_2) \end{aligned} \qquad\qquad \begin{aligned} \kappa(A) &:= 0 \qquad (A \in \mathsf{S}) \\ \kappa([\,]) &:= 1 \\ \kappa(\langle\, \Xi_1 \,\rangle^j) &:= \kappa(\Xi_1) \\ \kappa(\Xi_1 \ \circ_i \ \Xi_2) &:= \kappa(\Xi_1) + \kappa(\Xi_2) \end{aligned}$$

Let $\mathsf{tree}_{\mathsf{S}}^{\lambda,\kappa}$ be the subset of $\mathsf{trees}_{\mathsf{S}}$ consisting of trees with length λ and number of holes κ. Observe that $\mathsf{S} \subset \mathsf{tree}_{\mathsf{S}}^{1,0}$.

For every $\Xi \in \mathsf{tree}_{\mathsf{S}}^{\lambda,\kappa}$ there is a multiset $\|\Xi\|$ of $\lambda - \kappa$ elements of S which equals Ξ modulo structural information and holes. Moreover, we define $\langle\!\langle \Xi \rangle\!\rangle$ to be the order in which the elements of $\|\Xi\|$ occur in Ξ.

$$\begin{aligned} \|A\| &:= \{A\} \qquad (A \in \mathsf{S}) \\ \|[\,]\| &:= \emptyset \\ \big\|\langle\, \Xi_1 \,\rangle^j\big\| &:= \|\Xi_1\| \\ \|\Xi_1 \ \circ_i \ \Xi_2\| &:= \|\Xi_1\| \cup \|\Xi_2\| \end{aligned} \qquad\qquad \begin{aligned} \langle\!\langle A \rangle\!\rangle &:= A \qquad (A \in \mathsf{S}) \\ \langle\!\langle [\,] \rangle\!\rangle &:= \emptyset \\ \big\langle\!\big\langle \langle\, \Xi_1 \,\rangle^j \big\rangle\!\big\rangle &:= \langle\!\langle \Xi_1 \rangle\!\rangle \\ \langle\!\langle \Xi_1 \ \circ_i \ \Xi_2 \rangle\!\rangle &:= \langle\!\langle \Xi_1 \rangle\!\rangle \,, \langle\!\langle \Xi_2 \rangle\!\rangle \end{aligned}$$

Example A.1

$$\begin{aligned} \langle\, A_1 \ \circ_1 \ \langle\, A_2 \,\rangle^1 \,\rangle^1 \ \circ_2 \ (A_1 \ \circ_1 \ [\,]) &\in \mathsf{tree}_{\mathsf{S}}^{4,1} \\ \big\|\langle\, A_1 \ \circ_1 \ \langle\, A_2 \,\rangle^1 \,\rangle^1 \ \circ_2 \ (A_1 \ \circ_1 \ [\,])\big\| &= \{A_1, A_1, A_2\} \\ \langle\, [\,] \,\rangle^1 \ \circ_1 \ (\langle\, [\,] \,\rangle^1 \ \circ_1 \ (\langle\, [\,] \,\rangle^1 \ \circ_2 \ [\,])) &\in \mathsf{tree}_{\mathsf{S}}^{4,4} \\ \left|\left\{ \Xi \in \mathsf{tree}_{\mathsf{S}}^{7,7} \,\middle|\, \Xi \text{ is } \langle\ \rangle^j\text{-free} \right\}\right| &= 132 \ |I|^6 \end{aligned}$$

♦

There is a substitution operation

$$\begin{aligned} \mathsf{tree}_{\mathsf{S}}^{\lambda,\kappa} \times \mathsf{tree}_{\mathsf{S}}^{l_1,k_1} \times \ldots \times \mathsf{tree}_{\mathsf{S}}^{l_\kappa,k_\kappa} &\to \mathsf{tree}_{\mathsf{S}}^{\lambda-\kappa+\sum_{j=1}^{\kappa} l_j \ ,\ \sum_{j=1}^{\kappa} k_j} \\ (\Xi, \Theta_1, \ldots, \Theta_\kappa) &\mapsto \Xi[\ \Theta_1, \ldots, \Theta_\kappa\ ] \end{aligned}$$

which can be defined by induction on Ξ:

- if $\Xi \in \mathsf{S}$, then $\lambda = 1; \kappa = 0$, and we define the image of Ξ to be Ξ itself;
- if $\Xi = [\,]$, then $\lambda = 1; \kappa = 1$, and we define the image of (Ξ, Θ_1) to be $\Xi[\ \Theta_1\ ] := \Theta_1$;

- if $\Xi = \langle \Xi_1 \rangle^j$, then $\lambda = \lambda_1$ and $\kappa = \kappa_1$. By induction hypothesis we know that $\Xi_1[\,\Theta_1, \ldots, \Theta_{\kappa_1}\,]$ is defined and in $\mathsf{tree}_{\mathsf{S}}^{\lambda_1-\kappa_1+\sum_{j=1}^{\kappa_1} l_j\,,\,\sum_{j=1}^{\kappa_1} k_j}$. Now we define the image of $(\Xi, \Theta_1, \ldots, \Theta_\kappa)$ to be

$$\Xi[\,\Theta_1, \ldots, \Theta_\kappa\,] := \langle \Xi_1[\,\Theta_1, \ldots, \Theta_{\kappa_1}\,] \rangle^j$$

which is in

$$\mathsf{tree}_{\mathsf{S}}^{\lambda-\kappa+\sum_{j=1}^{\kappa} l_j\,,\,\sum_{j=1}^{\kappa} k_j},$$

as desired.

- if $\Xi = \Xi_1 \circ_i \Xi_2$, then $\lambda = \lambda_1 + \lambda_2$ and $\kappa = \kappa_1 + \kappa_2$. By induction hypothesis we know that $\Xi_1[\,\Theta_1, \ldots, \Theta_{\kappa_1}\,]$ is defined and in $\mathsf{tree}_{\mathsf{S}}^{\lambda_1-\kappa_1+\sum_{j=1}^{\kappa_1} l_j\,,\,\sum_{j=1}^{\kappa_1} k_j}$ while $\Xi_2[\,\Theta_{\kappa_1+1}, \ldots, \Theta_{\kappa_1+\kappa_2}\,]$ is defined and in $\mathsf{tree}_{\mathsf{S}}^{\lambda_2-\kappa_2+\sum_{j=\kappa_1+1}^{\kappa_1+\kappa_2} l_j\,,\,\sum_{j=\kappa_1+1}^{\kappa_1+\kappa_2} k_j}$. Now we define the image of $(\Xi, \Theta_1, \ldots, \Theta_\kappa)$ to be

$$\Xi[\,\Theta_1, \ldots, \Theta_\kappa\,] := \Xi_1[\,\Theta_1, \ldots, \Theta_{\kappa_1}\,] \circ_i \Xi_2[\,\Theta_{\kappa_1+1}, \ldots, \Theta_{\kappa_1+\kappa_2}\,]$$

which is in

$$\mathsf{tree}_{\mathsf{S}}^{\lambda_1-\kappa_1+\sum_{j=1}^{\kappa_1} l_j+\lambda_2-\kappa_2+\sum_{j=\kappa_1+1}^{\kappa_1+\kappa_2} l_j\,,\,\sum_{j=1}^{\kappa_1} k_j+\sum_{j=\kappa_1+1}^{\kappa_1+\kappa_2} k_j} = \mathsf{tree}_{\mathsf{S}}^{\lambda-\kappa+\sum_{j=1}^{\kappa} l_j\,,\,\sum_{j=1}^{\kappa} k_j},$$

as desired.

Example A.2 • Restricting our attention to the case where $\lambda = \kappa$ *(formal trees)*, this map is

$$\mathsf{tree}^{\lambda,\lambda} \times \mathsf{tree}^{l_1,l_1} \times \ldots \times \mathsf{tree}^{l_\lambda,l_\lambda} \to \mathsf{tree}^{\sum_{j=1}^{\lambda} l_j\,,\,\sum_{j=1}^{\lambda} l_j}$$

where we have deleted the subscript S since these sets do not depend on it.

- Restricting our attention to substitution of elements of S, this map is

$$\mathsf{tree}_{\mathsf{S}}^{\lambda,\kappa} \times \mathsf{S}^{\kappa} \to \mathsf{tree}_{\mathsf{S}}^{\lambda,0}$$

- Restricting our attention to the case where $\kappa = 0$ *(*S*-trees)*, the substitution map is the identity $\mathsf{tree}_{\mathsf{S}}^{\lambda,0} \to \mathsf{tree}_{\mathsf{S}}^{\lambda,0}$ (proof by induction); there is nothing to substitute.

♦

Lemma A.3 Given $(\Xi, \Theta_1, \ldots, \Theta_\kappa) \in \mathsf{tree}_{\mathsf{S}}^{\lambda,\kappa} \times \mathsf{tree}_{\mathsf{S}}^{l_1,k_1} \times \ldots \times \mathsf{tree}_{\mathsf{S}}^{l_\kappa,k_\kappa}$, the following holds:

$$\|\Xi[\,\Theta_1, \ldots, \Theta_\kappa\,]\| = \|\Xi\| \cup \bigcup_{j=1}^{\kappa} \|\Theta_j\|$$

♦

Proof: By induction on Ξ.

We study sequents in which the antecedent part is a structure tree of formulas rather than a sequence or a multiset of formulas.

$$\mathsf{seq} := \left\{ \Gamma \vdash C \mid n \geq 1; \Gamma \in \mathsf{tree}_{\mathsf{form}}^{n,0}; C \in \mathsf{form} \right\}$$

Atoms will be denoted by $p, q, \ldots$; modes by $i, j, \ldots$; formulas by $A, B, C, \ldots$; and form-trees by $\Gamma, \Delta, \ldots$. Observe that all sequents have non-empty antecedent part, since there are no empty trees.

Convention Writing down a sequent like $\Delta[\,\Gamma_1, \Gamma_2\,] \vdash C$ implies that $\Delta \in \mathsf{tree}^{n,2}_{\mathsf{form}}$; $\Gamma_1 \in \mathsf{tree}^{n_1,0}_{\mathsf{form}}$ and $\Gamma_2 \in \mathsf{tree}^{n_2,0}_{\mathsf{form}}$, yielding $\Delta[\,\Gamma_1, \Gamma_2\,] \in \mathsf{tree}^{n-2+n_1+n_2,0}_{\mathsf{form}}$

There is a map $\mathsf{tree}^{n,0}_{\mathsf{form}} \to \mathsf{form}$, which replaces all $\langle\ \rangle^j$- and $\circ_i$ -occurrences by $\Diamond_j$- and $\bullet_i$-occurrences. The image of Γ under this map will be denoted by $\Gamma^\bullet$.

$$
\begin{aligned}
(A)^\bullet &:= A \qquad (A \in \mathsf{form}) \\
(\langle\,\Xi_1\,\rangle^j)^\bullet &:= \Diamond_j \Xi_1^\bullet \\
(\Xi_1 \ \circ_i \ \Xi_2)^\bullet &:= \Xi_1^\bullet \bullet_i \Xi_2^\bullet
\end{aligned}
$$

References

[Danos 90] Danos, V. *La Logique Linéaire Appliquée à l'étude de Divers Processus de Normalisation (Principalement du Lambda-Calcul).* Thèse de Doctorat, Université de Paris VII, 1990.

[Girard 87] Girard, J.Y. *Linear Logic.* Theoretical Computer Science **50**, 1987, pp. 1-102.

[Lambek 58] Lambek, J. *The Mathematics of Sentence Structure* American Mathematical Monthly 65, 1958, pp. 154-170.

[Moortgat 97] Moortgat, M. *Categorial Type Logics.* Chapter 2 of Benthem, J. van, and A. ter Meulen (eds.) *Handbook of Logic and Language.* Elsevier, 1997.

[Morrill 96] Morrill, G. *Memoisation of Categorial Proof Nets: Parallelism in Categorial Processing.* Report de Recerca LSI-96-24-R, Departament de Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, 1996.

[Puite 98] Puite, Q. *Proof Nets with Explicit Negation for Multiplicative Linear Logic.* Preprint 1079, Department of Mathematics, Utrecht University, 1998.

[Roorda 91] Roorda, D. *Resource Logics: A Proof-theoretical Study.* PhD Thesis, University of Amsterdam, 1991.