# Deep Inference, Expansion Trees, and Proof Graphs for Second Order Propositional Multiplicative Linear Logic

Lutz Straßburger

# Deep Inference, Expansion Trees, and Proof Graphs for Second Order Propositional Multiplicative Linear Logic

Lutz Straßburger

RESEARCH REPORT
N° 9071
May 2017
Project-Team Parsifal

ISSN 0249-6399    ISRN INRIA/RR--9071--FR+ENG

# Deep Inference, Expansion Trees, and Proof Graphs for Second Order Propositional Multiplicative Linear Logic

Lutz Straßburger

Project-Team Parsifal

Research Report n° 9071 — May 2017 — 35 pages

**Abstract:** In this paper we introduce the notion of expansion tree for linear logic. As in Miller's original work, we have a shallow reading of an expansion tree that corresponds to the conclusion of the proof, and a deep reading which is a formula that can be proved by propositional rules. We focus our attention to MLL2, and we also present a deep inference system for that logic. This allows us to give a syntactic proof to a version of Herbrand's theorem.

**Key-words:** Deep inference, expansion trees, proof nets, cut elimination, linear logic

RESEARCH CENTRE
SACLAY – ÎLE-DE-FRANCE

1 rue Honoré d'Estienne d'Orves
Bâtiment Alan Turing
Campus de l'École Polytechnique
91120 Palaiseau

# Inférence profonde, arbres d'expansion et graphes de preuves pour la logique linéaire multiplicative propositionelle du second ordre

**Résumé :** Dans cet article, nous introduisons la notion d'arbre d'expansion pour la logique linéaire. Comme dans le travail original de Miller, nous définissons la formule extérieure d'un arbre d'expansion qui correspond à la conclusion de la preuve et la formule intérieure qui peut être prouvée par des règles propositionnelles. Nous concentrons notre attention sur MLL2 et nous présentons également un système d'inférence profonde pour cette logique. Cela nous permet de donner une preuve syntaxique d'une version du théorème de Herbrand.

**Mots-clés :** Inférence profonde, arbres d'expansion, réseaux de preuves, élimination des coupures, logique linéaire

$$
\mathsf{id}\,\frac{}{\vdash a^{\perp}, a}
\qquad
\bot\,\frac{\vdash \Gamma}{\vdash \bot, \Gamma}
\qquad
1\,\frac{}{\vdash 1}
\qquad
\mathsf{exch}\,\frac{\vdash \Gamma, A, B, \Delta}{\vdash \Gamma, B, A, \Delta}
$$

$$
\parr\,\frac{\vdash A, B, \Gamma}{\vdash [A \parr B], \Gamma}
\qquad
\otimes\,\frac{\vdash \Gamma, A \quad \vdash B, \Delta}{\vdash \Gamma, (A \otimes B), \Delta}
\qquad
\exists\,\frac{\vdash A\langle a \backslash B\rangle, \Gamma}{\vdash \exists a.A, \Gamma}
\qquad
\forall\,\frac{\vdash A, \Gamma}{\vdash \forall a.A, \Gamma}\ \ \text{$a$ not free in $\Gamma$}
$$

Figure 1: Sequent calculus system for MLL2

# 1 Introduction

Expansion trees [Mil87] have been introduced by Miller to generalize Herbrand's theorem to higher order logic. In principle, an expansion tree is a data structure for proofs that carries the information of two formulas. The *shallow* formula is the conclusion of the proof, and the *deep* formula is a propositional tautology for which the information about the proof has to be provided by other means.

This possible separation of the "quantifier part" and the "propositional part" in a proof is a unique property of classical logic. For intuitionistic logic, for example, only a limited form of Herbrand's theorem can be obtained [LK06]. The question we would like to address in this paper is whether some form of Herbrand's theorem can be achieved for linear logic.

For simplicity, we concentrate in this paper on second order multiplicative linear logic (MLL2) because the notion of proof in its propositional fragment (MLL) is thoroughly understood: on the deductive level via rule permutations in the sequent calculus [Laf95] and the calculus of structures [Str03], on the combinatoric level via proof nets [Gir87], and on the algebraic level via star-autonomous categories [Laf88, LS06], and the first order version has a rather simple proof theory [BvdW95].

There are two main contributions in this paper:

1. First, we will present a data structure for linear logic proofs that carries the information of two formulas: a *shallow* formula that is the conclusion of the proof, and the *deep* formula for which another proof data structure will be provided that is essentially an ordinary MLL proof net. Due to the similarities to Miller's work [Mil87], we will call our data structure *expansion tree*. Since we will also consider the multiplicative units, we follow the work in [SL04, LS06] to present a notion of proof graph, that can be plugged on top of our expansion tree and that will cover the propositional part of an MLL2 proof. In order to make cut elimination work, we need to impose an equivalence relation on these proof graphs. This is a consequence of the PSPACE-completeness of proof equivalence in MLL [HH14].

2. Our second contribution will be a deductive proof system for MLL2 in the calculus of structures [GS01, Gug07], making extensive use of deep inference features. This allows us to achieve the same decomposition of a proof into a "quantifier part" and a "propositional part", as it happens with the expansion trees and the proof graphs. This relation will be made precise via a correspondence theorem.

The paper is organized as follows: We will first recall the presentation of MLL2 in the sequent calculus (Section 2) and then give its presentation in the calculus of structures (Section 3), followed by a discussion on locality in the calculus of structures (Section 4). We also show the relation between the sequent calculus system, that we call $\mathsf{MLL2}_{\mathsf{Seq}}$ and the deep inference system that we call $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$. Then, in Section 5, we introduce our expansion trees and also show their relation to the deep inference system. This is followed by the introduction of proof graphs in Section 6. In Sections 7, 8 and 9 we explore the relation between proof graphs and the two deductive systems, i.e., we show "sequentialization" into the calculus of structures and into the sequent calculus. Finally, in Section 10 we show cut elimination for our proof graphs with expansion trees.

Some of the results of this paper have already been published at the TLCA 2009 conference [Str09]. The main additions here are (1) full proofs of all results, (2) the presentation of cut elimination, and (3) an improved presentation that clearly separates the expansion trees from the propositional part.

# 2 MLL2 in the sequent calculus

Let us first recall the logic MLL2 by giving its presentation in the sequent calculus, by providing a grammar for well-formed formulas and sequents, together with a set of (sequent style) inference rules. Then the theorems of the logic are

$$
\mathsf{ai}\!\downarrow\frac{S\{1\}}{S[a^\perp \bindnasrepma a]}\qquad
\bot\!\downarrow\frac{S\{A\}}{S[\bot \bindnasrepma A]}\qquad
1\!\downarrow\frac{S\{A\}}{S(1\otimes A)}\qquad
\mathsf{e}\!\downarrow\frac{S\{1\}}{S\{\forall a.1\}}
$$

$$
\alpha\!\downarrow\frac{S[[A \bindnasrepma B]\bindnasrepma C]}{S[A \bindnasrepma [B \bindnasrepma C]]}\qquad
\sigma\!\downarrow\frac{S[A \bindnasrepma B]}{S[B \bindnasrepma A]}\qquad
\mathsf{ls}\frac{S([A \bindnasrepma B]\otimes C)}{S[A \bindnasrepma (B\otimes C)]}\qquad
\mathsf{rs}\frac{S(A\otimes[B \bindnasrepma C])}{S[(A\otimes B)\bindnasrepma C]}
$$

$$
\mathsf{u}\!\downarrow\frac{S\{\forall a.[A \bindnasrepma B]\}}{S[\forall a.A \bindnasrepma \exists a.B]}\qquad
\mathsf{n}\!\downarrow\frac{S\{A\langle a\backslash B\rangle\}}{S\{\exists a.A\}}\qquad
\mathsf{f}\!\downarrow\frac{S\{\exists a.A\}}{S\{A\}}\ \ a \text{ not free in } A.
$$

Figure 2: Deep inference system for MLL2

defined to be those formulas that are derivable via the rules. For MLL2 the set $\mathscr{F}$ of *formulas* is generated by the grammar

$$\mathscr{F} \quad ::= \quad \bot \mid 1 \mid \mathscr{A} \mid \mathscr{A}^\perp \mid [\mathscr{F} \bindnasrepma \mathscr{F}] \mid (\mathscr{F}\otimes\mathscr{F}) \mid \forall\mathscr{A}.\mathscr{F} \mid \exists\mathscr{A}.\mathscr{F}$$

where $\mathscr{A} = \{a, b, c, \ldots\}$ is a countable set of *propositional variables*. Formulas are denoted by capital Latin letters $(A, B, C, \ldots)$. Linear negation $(-)^\perp$ is defined for all formulas by the usual De Morgan laws:

$$
\begin{array}{cccc}
\bot^\perp = 1 & a^\perp = a^\perp & [A \bindnasrepma B]^\perp = (A^\perp \otimes B^\perp) & (\exists a.A)^\perp = \forall a.A^\perp \\
1^\perp = \bot & a^{\perp\perp} = a & (A\otimes B)^\perp = [A^\perp \bindnasrepma B^\perp] & (\forall a.A)^\perp = \exists a.A^\perp
\end{array}
$$

An *atom* is a propositional variable $a$ or its dual $a^\perp$. *Sequents* are finite lists of formulas, separated by comma, and are denoted by capital Greek letters $(\Gamma, \Delta, \ldots)$. The notions of *free* and *bound variable* are defined in the usual way, and we can always rename bound variables. In view of the later parts of the paper, and in order to avoid changing syntax all the time, we use the following syntactic conventions:

(i) We always put parentheses around binary connectives. For better readability we use $[\ldots]$ for $\bindnasrepma$ and $(\ldots)$ for $\otimes$.

(ii) We omit parentheses if they are superfluous under the assumption that $\bindnasrepma$ and $\otimes$ associate to the left, e.g., we write $[A \bindnasrepma B \bindnasrepma C \bindnasrepma D]$ to abbreviate $[[[A \bindnasrepma B] \bindnasrepma C] \bindnasrepma D]$.

(iii) The scope of a quantifier ends at the earliest possible place (and not at the latest possible place as usual). This helps saving unnecessary parentheses. For example, in $[\forall a.(a\otimes b) \bindnasrepma \exists c.c \bindnasrepma a]$, the scope of $\forall a$ is $(a\otimes b)$, and the scope of $\exists c$ is just $c$. In particular, the $a$ at the end is free.

The inference rules for MLL2 are shown in Figure 1. In the following, we will call this system $\mathsf{MLL2_{Seq}}$. As shown in [Gir87], it has the cut elimination property:

**Theorem 2.1.** *The cut rule* $\mathsf{cut}\dfrac{\vdash \Gamma, A \quad \vdash A^\perp, \Delta}{\vdash \Gamma, \Delta}$ *is admissible for* $\mathsf{MLL2_{Seq}}$.

## 3 MLL2 in the calculus of structures

We now present a deductive system for MLL2 based on deep inference. We use the calculus of structures, in which the distinction between formulas and sequents disappears. This is the reason for the syntactic conventions introduced above.[1]

The inference rules now work directly (as rewriting rules) on the formulas. The system for MLL2 is shown in Figure 2. There, $S\{\ \}$ stands for an arbitrary (positive) formula context. We omit the braces if the structural parentheses fill the hole. E.g., $S[A \bindnasrepma B]$ abbreviates $S\{[A \bindnasrepma B]\}$. The system in Figure 2 is called $\mathsf{MLL2_{DI\downarrow}}$. We use the down-arrow in the name to emphasize that we consider here only the so-called *down fragment* of the system, which corresponds to the cut-free system in the sequent calculus.[2] Note that the $\forall$-rule of $\mathsf{MLL2_{Seq}}$ is in $\mathsf{MLL2_{DI\downarrow}}$ decomposed into three pieces, namely, $\mathsf{e}\!\downarrow$, $\mathsf{u}\!\downarrow$, and $\mathsf{f}\!\downarrow$. In $\mathsf{MLL2_{DI\downarrow}}$, we also need an explicit rule for associativity which is in the sequent calculus "built in". The

[1] In the literature on deep inference, e.g., [BT01, Gug07], the formula $(a\otimes[b \bindnasrepma (a^\perp\otimes c)])$ would be written as $(a, [b, (a^\perp, c)])$, while without our convention mentioned in the previous section, it would be written as $a\otimes(b \bindnasrepma (a^\perp\otimes c))$. Our syntactic convention can therefore be seen as an attempt to please both communities. In particular, note that the motivation for the syntactic convention (iii) above is the collapse of the $\bindnasrepma$ on the formula level and the comma on the sequent level, e.g., $[\forall a.(a\otimes b) \bindnasrepma \exists c.c \bindnasrepma a]$ is the same as $[\forall a.(a, b), \exists c.c, a]$.

[2] The *up fragment* (which corresponds to the cut in the sequent calculus) is obtained by dualizing the rules in the down fragment, i.e., by negating and exchanging premise and conclusion. See, e.g., [Str03, Brü03, BT01, GS01, CGS11] for details.

other rules are almost the same as in the sequent calculus. In particular, the relation between the $\otimes$-rule and the rules $\mathsf{ls}$ and $\mathsf{rs}$ (called *switch*) has already in detail been investigated by several authors [Ret93, BCST96, DHPP99, Gug07]. A *derivation* $\mathscr{D}$ in the system $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$ is denoted by

$$\mathsf{MLL2}_{\mathsf{DI}\downarrow} \left\| \begin{matrix} A \\ \mathscr{D} \\ B \end{matrix} \right.$$

and is simply a rewriting path from $A$ to $B$ using the inference rules in $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$. We say $A$ is the *premise* and $B$ the *conclusion* of $\mathscr{D}$. A *proof* in $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$ is a derivation whose premise is $1$. The following theorem ensures that $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$ is indeed a deductive system for $\mathsf{MLL2}$.

**Theorem 3.1.** *Let $A_1, \ldots, A_n$ be arbitrary* $\mathsf{MLL2}$ *formulas. For every proof of $\vdash A_1, \ldots, A_n$ in* $\mathsf{MLL2}_{\mathsf{Seq}}$*, there is a proof of $[A_1 \parr \cdots \parr A_n]$ in* $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$*, and vice versa.*

*Proof.* We proceed by structural induction on the sequent proof to construct the deep inference proof. The only non-trivial cases are the rules for $\otimes$ and $\forall$. If the last rule application in the sequent proof is a $\otimes$, then we have by induction hypothesis two proofs

$$\mathsf{MLL2}_{\mathsf{DI}\downarrow} \left\| \begin{matrix} 1 \\ \mathscr{D}_1 \\ [\Gamma \parr A] \end{matrix} \right. \qquad \text{and} \qquad \mathsf{MLL2}_{\mathsf{DI}\downarrow} \left\| \begin{matrix} 1 \\ \mathscr{D}_2 \\ [B \parr \Delta] \end{matrix} \right.$$

From these we can built

$$\begin{array}{c} 1 \\ \mathsf{MLL2}_{\mathsf{DI}\downarrow} \| \mathscr{D}_2 \\ 1\!\downarrow \dfrac{[B \parr \Delta]}{[(1 \otimes B) \parr \Delta]} \\ \mathsf{MLL2}_{\mathsf{DI}\downarrow} \| \mathscr{D}_1 \\ \mathsf{ls}\, \dfrac{[([\Gamma \parr A] \otimes B) \parr \Delta]}{[\Gamma \parr (A \otimes B) \parr \Delta]} \end{array}$$

In case of the $\forall$-rule, we have by induction hypothesis a proof

$$\mathsf{MLL2}_{\mathsf{DI}\downarrow} \left\| \begin{matrix} 1 \\ \mathscr{D} \\ [A \parr \Gamma] \end{matrix} \right.$$

from which we get

$$\begin{array}{c} \mathsf{e}\!\downarrow \dfrac{1}{\forall a.1} \\ \mathsf{MLL2}_{\mathsf{DI}\downarrow} \| \mathscr{D} \\ \mathsf{u}\!\downarrow \dfrac{\forall a.[A \parr \Gamma]}{[\forall a.A \parr \exists a.\Gamma]} \\ \mathsf{f}\!\downarrow \dfrac{}{[\forall a.A \parr \Gamma]} \end{array}$$

Conversely, for translating a $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$ proof $\mathscr{D}$ into the sequent calculus, we proceed by induction on the length of $\mathscr{D}$. We then translate

$$\begin{array}{c} 1 \\ \mathsf{MLL2}_{\mathsf{DI}\downarrow} \| \mathscr{D}' \\ \rho\, \dfrac{A}{B} \end{array}$$

into

$$\dfrac{\begin{array}{c}\mathscr{D}_1\\ \vdash A\end{array} \qquad \begin{array}{c}\mathscr{D}_2\\ \vdash A^\perp, B\end{array}}{\vdash B}\ \mathsf{cut} \tag{1}$$

where $\mathscr{D}_1$ exists by induction hypothesis and $\mathscr{D}_2$ exists because every rule $\rho$ of $\mathsf{MLL2_{DI\downarrow}}$ is a valid implication of $\mathsf{MLL2}$. Finally, we apply cut elimination (Theorem 2.1). □

**Remark 3.2.** Later in this paper we will introduce methods that will allow us to translate cut-free proofs from deep inference to the sequent calculus without introducing cuts.

As for $\mathsf{MLL2_{Seq}}$, we also have for $\mathsf{MLL2_{DI\downarrow}}$ the cut elimination property, which can be stated as follows:

**Theorem 3.3.** *The cut rule* $\mathsf{i}{\uparrow}\,\dfrac{S(A \otimes A^\perp)}{S\{\bot\}}$ *is admissible for* $\mathsf{MLL2_{DI\downarrow}}$.

*Proof.* Given a proof in $\mathsf{MLL2_{DI\downarrow}} \cup \{\mathsf{i}{\uparrow}\}$, we translate it into $\mathsf{MLL2_{Seq}}$ as done in the proof of Theorem 3.1, eliminate the cut (Theorem 2.1), and translate the result back into $\mathsf{MLL2_{DI\downarrow}}$. When translating a sequent calculus proof with cuts into the calculus of structures as described in the proof of Theorem 3.1, then the sequent cut rule $\mathsf{cut}\,\dfrac{\vdash \Gamma, A \quad \vdash A^\perp, \Delta}{\vdash \Gamma, \Delta}$ is simulated exactly by the rule $\mathsf{i}{\uparrow}\,\dfrac{S(A \otimes A^\perp)}{S\{\bot\}}$ . □

We could also give a direct proof of Theorem 3.3, inside the calculus of structures, without referring to the sequent calculus, by using a combination of the techniques of *decomposition* and *splitting* [Str03, Brü03, Gug07, SG11, GS11, Tub16]. However, presenting all the details would go beyond the scope of this paper. We show here only the "one-sided" version of the *decomposition theorem* for $\mathsf{MLL2_{DI\downarrow}}$, which can be seen as a version of Herbrand's theorem for $\mathsf{MLL2}$, and which has no counterpart in the sequent calculus.

**Theorem 3.4.**

*Every derivation* $\begin{array}{c}1\\ \mathsf{MLL2_{DI\downarrow}}\,\Big\Vert\,\mathscr{D}\\ C\end{array}$ *can be transformed into* $\begin{array}{c}1\\ \{\mathsf{ai}{\downarrow}, \bot{\downarrow}, 1{\downarrow}, \mathsf{e}{\downarrow}\}\,\Big\Vert\,\mathscr{D}_1\\ A\\ \{\alpha{\downarrow}, \sigma{\downarrow}, \mathsf{ls}, \mathsf{rs}, \mathsf{u}{\downarrow}\}\,\Big\Vert\,\mathscr{D}_2\\ B\\ \{\mathsf{n}{\downarrow}, \mathsf{f}{\downarrow}\}\,\Big\Vert\,\mathscr{D}_3\\ C\end{array}$ .

*Proof.* The construction is done in two phases. First, we permute all instances of $\mathsf{ai}{\downarrow}, \bot{\downarrow}, 1{\downarrow}, \mathsf{e}{\downarrow}$ to the top of the derivation. For $\mathsf{ai}{\downarrow}$ and $\mathsf{e}{\downarrow}$ this is trivial, because all steps are similar to the following:

$$\mathsf{e}{\downarrow}\,\dfrac{\sigma{\downarrow}\,\dfrac{S[A \parr B\{1\}]}{S[B\{1\} \parr A]}}{S[B\{\forall a.1\} \parr A]} \quad \to \quad \sigma{\downarrow}\,\dfrac{\mathsf{e}{\downarrow}\,\dfrac{S[A \parr B\{1\}]}{S[A \parr B\{\forall a.1\}]}}{S[B\{\forall a.1\} \parr A]}$$

For $\bot{\downarrow}$ and $1{\downarrow}$ there are some more cases to inspect. We show here only one because all others are similar:

$$1{\downarrow}\,\dfrac{\mathsf{u}{\downarrow}\,\dfrac{S\{\forall a.[A \parr B]\}}{S[\forall a.A \parr \exists a.B]}}{S[(1 \otimes \forall a.A) \parr \exists a.B]} \quad \to \quad \mathsf{rs}\,\dfrac{\mathsf{u}{\downarrow}\,\dfrac{1{\downarrow}\,\dfrac{S\{\forall a.[A \parr B]\}}{S(1 \otimes \forall a.[A \parr B])}}{S(1 \otimes [\forall a.A \parr \exists a.B])}}{S[(1 \otimes \forall a.A) \parr \exists a.B]}$$

Here, in order to permute the $1{\downarrow}$ above the $\mathsf{u}{\downarrow}$, we need an additional instance of $\mathsf{rs}$ (and possibly two instances of $\sigma{\downarrow}$). The situation is analogous if we permute the $1{\downarrow}$ over $\mathsf{ls}$, $\mathsf{rs}$, or $\alpha{\downarrow}$ (or $\mathsf{ai}{\downarrow}$ or $\bot{\downarrow}$, but this is not needed for this theorem). When permuting $\bot{\downarrow}$ up (instead of $1{\downarrow}$), then we need $\alpha{\downarrow}$ (and $\sigma{\downarrow}$) instead of $\mathsf{rs}$. For a detailed analysis of this kind of permutation arguments, the reader is referred to [Str03].

In the second phase of the decomposition, all instances of $\mathsf{n}{\downarrow}$ and $\mathsf{f}{\downarrow}$ are permuted down to the bottom of the derivation. For the rule $\mathsf{n}{\downarrow}$ this is trivial since no rule can interfere (except for $\mathsf{f}{\downarrow}$, which is also permuted down). For permuting down the rule $\mathsf{f}{\downarrow}$, the problematic cases are as before caused by the rules $\mathsf{u}{\downarrow}$, $\mathsf{ls}$, $\mathsf{rs}$, and $\alpha{\downarrow}$. To get our result, we need an additional inference rule:

$$\mathsf{v}{\downarrow}\,\frac{S\{\exists a.[A \parr B]\}}{S[\exists a.A \parr \exists a.B]} \tag{2}$$

Now we can do the following replacement

$$\mathsf{ls}\,\frac{\mathsf{f}{\downarrow}\,\dfrac{S(\exists a.[A \parr B] \otimes C)}{S([A \parr B] \otimes C)}}{S[A \parr (B \otimes C)]} \quad\to\quad \mathsf{f}{\downarrow}\,\frac{\mathsf{f}{\downarrow}\,\dfrac{\mathsf{ls}\,\dfrac{\mathsf{v}{\downarrow}\,\dfrac{S(\exists a.[A \parr B] \otimes C)}{S([\exists a.A \parr \exists a.B] \otimes C)}}{S[\exists a.A \parr (\exists a.B \otimes C)]}}{S[A \parr (\exists a.B \otimes C)]}}{S[A \parr (B \otimes C)]}$$

and continue permuting the two new $\mathsf{f}{\downarrow}$ further down. Finally, we eliminate all instances of $\mathsf{v}{\downarrow}$ by permuting them up. This is trivial since no rule has an $\exists$ in its conclusion, except for $\mathsf{u}{\downarrow}$ and $\mathsf{n}{\downarrow}$. In the case of $\mathsf{u}{\downarrow}$ we can replace

$$\mathsf{v}{\downarrow}\,\frac{\mathsf{u}{\downarrow}\,\dfrac{S\{\forall a.[A \parr [B \parr C]]\}}{S[\forall a.A \parr \exists a.[B \parr C]]}}{S[\forall a.A \parr [\exists a.B \parr \exists a.C]]} \qquad \text{by} \qquad \alpha{\downarrow}\,\frac{\mathsf{u}{\downarrow}\,\dfrac{\mathsf{u}{\downarrow}\,\dfrac{\alpha{\downarrow},\sigma{\downarrow}\,\dfrac{S\{\forall a.[A \parr [B \parr C]]\}}{S\{\forall a.[[A \parr B] \parr C]\}}}{S[\forall a.[A \parr B] \parr \exists a.C]}}{S[[\forall a.A \parr \exists a.B] \parr \exists a.C]}}{S[\forall a.A \parr [\exists a.B \parr \exists a.C]]}$$

and in the case of $\mathsf{n}{\downarrow}$, we can replace

$$\mathsf{v}{\downarrow}\,\frac{\mathsf{n}{\downarrow}\,\dfrac{S\{[A_1 \parr A_2]\langle a \backslash B\rangle\}}{S\{\exists a.[A_1 \parr A_2]\}}}{S[\exists a.A_1 \parr \exists a.A_2]} \qquad \text{by} \qquad \mathsf{n}{\downarrow}\,\frac{\mathsf{n}{\downarrow}\,\dfrac{=\,\dfrac{S\{[A_1 \parr A_2]\langle a \backslash B\rangle\}}{S[A_1\langle a \backslash B\rangle \parr A_2\langle a \backslash B\rangle]}}{S[A_1\langle a \backslash B\rangle \parr \exists a.A_2]}}{S[\exists a.A_1 \parr \exists a.A_2]}$$

Because we start from a proof, i.e., the premise of the derivation is 1, all $\mathsf{v}{\downarrow}$ must eventually disappear. □

**Observation 3.5.** The attentive reader might wonder why there are two versions of the "switch" in $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$, the *left switch* $\mathsf{ls}$, and the *right switch* $\mathsf{rs}$. For completeness (Theorem 3.1), the $\mathsf{ls}$-rule would be sufficient, but for obtaining the decomposition in Theorem 3.4 we need the $\mathsf{rs}$-rule as well.

If a derivation $\mathscr{D}$ uses only the rules $\alpha{\downarrow}, \sigma{\downarrow}, \mathsf{ls}, \mathsf{rs}, \mathsf{u}{\downarrow}$, then premise and conclusion of $\mathscr{D}$ (and every formula in between the two) must contain the same atom occurrences. Hence, the *atomic flow-graph* [Bus91, GG08] of the derivation $\mathscr{D}$ defines a bijection between the atom occurrences of premise and conclusion of $\mathscr{D}$. Here is an example of a derivation together with its flow-graph.

$$\mathsf{u}{\downarrow}\,\frac{\mathsf{u}{\downarrow}\,\dfrac{\mathsf{u}{\downarrow}\,\dfrac{\mathsf{rs}\,\dfrac{\mathsf{ls}\,\dfrac{\forall a.\forall c.([a^\perp \parr a] \otimes [c^\perp \parr c])}{\forall a.\forall c.[a^\perp \parr (a \otimes [c^\perp \parr c])]}}{\forall a.\forall c.[a^\perp \parr [(a \otimes c^\perp) \parr c]]}}{\forall a.[\exists c.a^\perp \parr \forall c.[(a \otimes c^\perp) \parr c]]}}{\forall a.[\exists c.a^\perp \parr [\exists c.(a \otimes c^\perp) \parr \forall c.c]]}}{[\forall a.\exists c.a^\perp \parr \exists a.[\exists c.(a \otimes c^\perp) \parr \forall c.c]]} \tag{3}$$

To avoid crossings in the flow-graph, we left some applications of $\alpha{\downarrow}$ and $\sigma{\downarrow}$ implicit.

# 4 Some observations on locality

In the sequent calculus the $\forall$-rule has a non-local behavior, in the sense that for applying the rule we need some global knowledge about the context $\Gamma$, namely, that the variable $a$ does not appear freely in it. This is the reason for the boxes in [Gir87] and the jumps in [Gir90]. In the calculus of structures this "checking" whether a variable appears freely is done in the rule $\mathsf{f}{\downarrow}$, which is as non-local as the $\forall$-rule in the sequent calculus. However, with deep inference, this rule can be made local, i.e., reduced to an atomic version (in the same sense as the identity axiom can be reduced to an atomic version). For this, we need an additional set of rules which is shown in Figure 3 (again, we show only the down fragment), and which is called $\mathsf{Lf}{\downarrow}$. Clearly, all rules are sound, i.e., proper implications of $\mathsf{MLL2}$. Now we have the following:

$$\mathsf{x}\,\frac{S\{\exists a.\forall b.A\}}{S\{\forall b.\exists a.A\}} \qquad \mathsf{y}{\downarrow}\,\frac{S\{\exists a.\exists b.A\}}{S\{\exists b.\exists a.A\}} \qquad \mathsf{v}{\downarrow}\,\frac{S\{\exists a.[A \mathbin{⅋} B]\}}{S[\exists a.A \mathbin{⅋} \exists a.B]} \qquad \mathsf{w}{\downarrow}\,\frac{S\{\exists a.(A\otimes B)\}}{S(\exists a.A \otimes \exists a.B)}$$

$$\mathsf{1f}{\downarrow}\,\frac{S\{\exists a.1\}}{S\{1\}} \qquad \bot\mathsf{f}{\downarrow}\,\frac{S\{\exists a.\bot\}}{S\{\bot\}} \qquad \mathsf{af}{\downarrow}\,\frac{S\{\exists a.b\}}{S\{b\}} \qquad \hat{\mathsf{a}}\mathsf{f}{\downarrow}\,\frac{S\{\exists a.b^\bot\}}{S\{b^\bot\}}$$

in $\mathsf{af}{\downarrow}$ and $\hat{\mathsf{a}}\mathsf{f}{\downarrow}$, $a$ is different from $b$

Figure 3: Towards a local system for MLL2

**Theorem 4.1.** *Every derivation* $\begin{array}{c}B\\ \{\mathsf{n}{\downarrow},\mathsf{f}{\downarrow}\}\Big\Vert\mathscr{D}\\ C\end{array}$ *can be transformed into* $\begin{array}{c}B\\ \{\mathsf{n}{\downarrow}\}\cup\mathsf{Lf}{\downarrow}\Big\Vert\mathscr{D}'\\ C\end{array}$ *, and vice versa.*

*Proof.* For transforming $\mathscr{D}$ into $\mathscr{D}'$, we replace every instance of $\mathsf{f}{\downarrow}$ by a derivation using only the rules in Figure 3. For this, we proceed by structural induction on the formula $A$ in the $\mathsf{f}{\downarrow}$. We show here only one case, the others are similar: If $A = (A' \otimes A'')$ then replace

$$\mathsf{f}{\downarrow}\,\frac{S\{\exists a.(A\otimes A'')\}}{S(A\otimes A'')} \quad \text{by} \quad \mathsf{f}{\downarrow}\,\frac{\mathsf{f}{\downarrow}\,\frac{\mathsf{w}{\downarrow}\,\frac{S\{\exists a.(A\otimes A'')\}}{S(\exists a.A\otimes \exists a.A'')}}{S(A\otimes \exists a.A'')}}{S(A\otimes A'')}$$

Conversely, for transforming $\mathscr{D}'$ into a derivation using only $\mathsf{n}{\downarrow}$ and $\mathsf{f}{\downarrow}$, note that $\mathsf{1f}{\downarrow}$, $\bot\mathsf{f}{\downarrow}$, $\mathsf{af}{\downarrow}$, and $\hat{\mathsf{a}}\mathsf{f}{\downarrow}$ are already instances of $\mathsf{f}{\downarrow}$. The rules $\mathsf{x}$, $\mathsf{y}{\downarrow}$, $\mathsf{v}{\downarrow}$, and $\mathsf{w}{\downarrow}$ can be replaced as follows:

$$\mathsf{v}{\downarrow}\,\frac{S\{\exists a.[A \mathbin{⅋} B]\}}{S[\exists a.A \mathbin{⅋} \exists a.B]} \quad\to\quad \mathsf{f}{\downarrow}\,\frac{\mathsf{n}{\downarrow}\,\frac{\mathsf{n}{\downarrow}\,\frac{S\{\exists a.[A \mathbin{⅋} B]\}}{S\{\exists a.[\exists a.A \mathbin{⅋} B]\}}}{S\{\exists a.[\exists a.A \mathbin{⅋} \exists a.B]\}}}{S[\exists a.A \mathbin{⅋} \exists a.B]}$$

where in the two $\mathsf{n}{\downarrow}$, the variable $a$ is substituted by itself. The other rules are handled similarly. □

There are seemingly two other sources of non-locality in $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$. The first is the renaming of bound variables ($\alpha$-conversion) that might be necessary for applying $\mathsf{u}{\downarrow}$ or $\mathsf{f}{\downarrow}$ (or $\mathsf{af}{\downarrow}$). But if we see bound variables as pointers to the quantifier which binds them, as for example in DeBruijn presentation [dB72], then $\alpha$-conversion is no issue anymore.[3] What remains is the substitution of an arbitrary formula for a variable in the instantiation of an existential quantifier (the $\mathsf{n}{\downarrow}$-rule), in which we have the unbounded number of occurrences of the quantified variable and the unbounded size of the formula to be substituted. It is a question of future research whether some sort of explicit substitution [ACCL91] can be coded directly into the inference rules in order to obtain a completely local system, in a similar way as it has been done in [CG14], and by this closing the gap between the formal deductive system and the implementation.

# 5 Expansion trees for MLL2

In their essence, expansion trees [Mil87] are enriched formula trees that encode two formulas, called the *deep formula* and the *shallow formula*, at the same time. The shallow formula is the conclusion of the proof, and the deep formula is a propositional tautology. Miller's original work [Mil87] makes indirect use of the properties of classical logic, and it is an interesting question whether we can achieve a similar data structure for linear logic. In one sense, the situation is more difficult because there is no simple Boolean semantics, but on the other hand, the situation is simpler because we do not have to deal with contraction.

We start with a set $\mathscr{E}$ of *expanded formulas* that are generated by

$$\mathscr{E} \quad ::= \quad \bot \mid 1 \mid \mathscr{A} \mid \mathscr{A}^\bot \mid [\mathscr{E} \mathbin{⅋} \mathscr{E}] \mid (\mathscr{E}\otimes\mathscr{E}) \mid \forall\mathscr{A}.\mathscr{E} \mid \exists\mathscr{A}.\mathscr{E} \mid \exists\mathscr{A}.\mathscr{E} \mid \boldsymbol{\exists}\mathscr{A}.\mathscr{E}$$

[3] The only reason for not using DeBruijn presentation here is its unreadability for human beings.

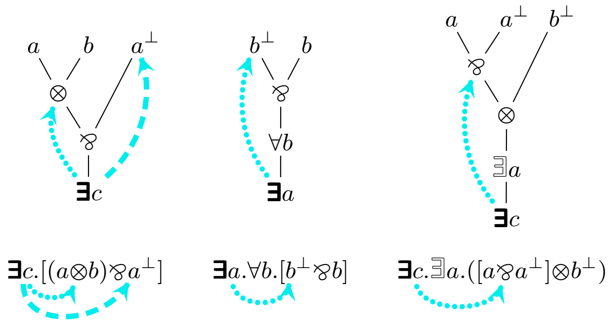


Figure 4: Examples of expanded sequents with stretchings that are not appropriate

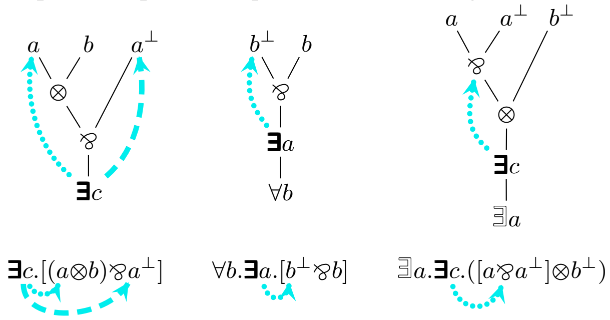


Figure 5: Appropriate examples of expanded sequents with stretchings

There are only two additional syntactic primitives: the $\reflectbox{$\mathbb{E}$}$, called *virtual existential quantifier*, and the $\boldsymbol{\exists}$, called *bold existential quantifier*. An *expanded sequent* is a finite list of expanded formulas, separated by comma. We denote expanded sequents by capital Greek letters ($\Gamma$, $\Delta$, $\ldots$). For disambiguation, the formulas/sequents introduced in Section 2 (i.e., those without $\reflectbox{$\mathbb{E}$}$ and $\boldsymbol{\exists}$) will also be called *simple formulas/sequents*.

In the following we will identify formulas with their syntax trees, where the leaves are decorated by elements of $\mathscr{A} \cup \mathscr{A}^\perp \cup \{1, \perp\}$. We can think of the inner nodes as decorated either with the connectives/quantifiers $\otimes$, $\invamp$, $\forall a$, $\exists a$, $\boldsymbol{\exists} a$, $\reflectbox{$\mathbb{E}$} a$, or with the whole subformula rooted at that node. For this reason we will use capital Latin letters ($A$, $B$, $C$, $\ldots$) to denote nodes in a formula tree. We write $A \preccurlyeq B$ if $A$ is a (not necessarily proper) ancestor of $B$, i.e., $B$ is a subformula occurrence in $A$. We write $\mathscr{L}\Gamma$ (resp. $\mathscr{L}A$) for denoting the set of leaves of a sequent $\Gamma$ (resp. formula $A$).

**Definition 5.1.** A *stretching $\sigma$ for a sequent* $\Gamma$ consists of two binary relations $\overset{\sigma}{\underset{+}{\curvearrowright}}$ and $\overset{\sigma}{\underset{-}{\curvearrowright}}$ on the set of nodes of $\Gamma$ (i.e., its subformula occurrences) such that $\overset{\sigma}{\underset{+}{\curvearrowright}}$ and $\overset{\sigma}{\underset{-}{\curvearrowright}}$ are disjoint, and whenever $A \overset{\sigma}{\underset{+}{\curvearrowright}} B$ or $A \overset{\sigma}{\underset{-}{\curvearrowright}} B$ then $A = \boldsymbol{\exists} a.A'$ with $A' \preccurlyeq B$ in $\Gamma$. An *expansion tree* is an expanded formula $E$ or sequent $\Gamma$ with a stretching, denoted by $E \blacktriangleleft \sigma$ or $\Gamma \blacktriangleleft \sigma$, respectively.

A stretching consists of edges connecting $\boldsymbol{\exists}$-nodes with some of its subformulas, and these edges can be positive or negative. Their purpose is to mark the places of the substitution of the atoms quantified by the $\boldsymbol{\exists}$. When writing an expansion tree $\Gamma \blacktriangleleft \sigma$, we will draw the stretching edges either inside $\Gamma$ when it is written as a tree, or below $\Gamma$ when it is written as string. The positive edges are dotted and the negative ones are dashed. Examples are shown in Figures 4 and 5.

The next step is to define the deep and the shallow formula of an expansion tree.

**Definition 5.2.** For an expansion tree $E \blacktriangleleft \sigma$, we define the *deep formula*, denoted by $\lceil E \blacktriangleleft \sigma \rceil$, and the *shallow formula*, denoted by $\lfloor E \blacktriangleleft \sigma \rfloor$, (which are both simple formulas) inductively as follows:

$$
\begin{array}{rlrlrl}
\lceil 1 \blacktriangleleft \emptyset \rceil &= 1 & \lceil A \invamp B \blacktriangleleft \sigma \rceil &= \lceil A \blacktriangleleft \sigma' \rceil \invamp \lceil B \blacktriangleleft \sigma'' \rceil & & \\
\lceil \perp \blacktriangleleft \emptyset \rceil &= \perp & \lceil A \otimes B \blacktriangleleft \sigma \rceil &= \lceil A \blacktriangleleft \sigma' \rceil \otimes \lceil B \blacktriangleleft \sigma'' \rceil & & \\
\lceil a \blacktriangleleft \emptyset \rceil &= a & \lceil \forall a.A \blacktriangleleft \sigma \rceil &= \forall a.\lceil A \blacktriangleleft \sigma \rceil & \lceil \reflectbox{$\mathbb{E}$} a.A \blacktriangleleft \sigma \rceil &= \exists a.\lceil A \blacktriangleleft \sigma \rceil \\
\lceil a^\perp \blacktriangleleft \emptyset \rceil &= a^\perp & \lceil \exists a.A \blacktriangleleft \sigma \rceil &= \exists a.\lceil A \blacktriangleleft \sigma \rceil & \lceil \boldsymbol{\exists} a.A \blacktriangleleft \sigma \rceil &= \lceil A \blacktriangleleft \sigma' \rceil \\
\\
\lfloor 1 \blacktriangleleft \emptyset \rfloor &= 1 & \lfloor A \invamp B \blacktriangleleft \sigma \rfloor &= \lfloor A \blacktriangleleft \sigma' \rfloor \invamp \lfloor B \blacktriangleleft \sigma'' \rfloor & & \\
\lfloor \perp \blacktriangleleft \emptyset \rfloor &= \perp & \lfloor A \otimes B \blacktriangleleft \sigma \rfloor &= \lfloor A \blacktriangleleft \sigma' \rfloor \otimes \lfloor B \blacktriangleleft \sigma'' \rfloor & & \\
\lfloor a \blacktriangleleft \emptyset \rfloor &= a & \lfloor \forall a.A \blacktriangleleft \sigma \rfloor &= \forall a.\lfloor A \blacktriangleleft \sigma \rfloor & \lfloor \reflectbox{$\mathbb{E}$} a.A \blacktriangleleft \sigma \rfloor &= \lfloor A \blacktriangleleft \sigma \rfloor \\
\lfloor a^\perp \blacktriangleleft \emptyset \rfloor &= a^\perp & \lfloor \exists a.A \blacktriangleleft \sigma \rfloor &= \exists a.\lfloor A \blacktriangleleft \sigma \rfloor & \lfloor \boldsymbol{\exists} a.A \blacktriangleleft \sigma \rfloor &= \exists a.\lfloor \tilde{A} \blacktriangleleft \tilde{\sigma} \rfloor
\end{array}
$$

where $\sigma'$ is the restriction of $\sigma$ to $A$, and $\sigma''$ is the restriction of $\sigma$ to $B$. The expanded formula $\tilde{A}$ in the last line is obtained from $A$ as follows: For every node $B$ with $A \preccurlyeq B$ and $\boldsymbol{\exists} a.A \overset{\sigma}{\underset{+}{\curvearrowright}} B$ remove the whole subtree $B$ and replace it by $a$, and for every $B$ with $\boldsymbol{\exists} a.A \overset{\sigma}{\underset{-}{\curvearrowright}} B$ replace $B$ by $a^\perp$. The stretching $\tilde{\sigma}$ is the restriction of $\sigma$ to $\tilde{A}$. For an expanded sequent $\Gamma$, we proceed analogously.

Note that the deep and and the shallow formula an expansion tree differ only on $\boldsymbol{\exists}$ and $\mathbb{\exists}$. In the deep formula, the $\mathbb{\exists}$ is treated as ordinary $\exists$, and the $\boldsymbol{\exists}$ is completely ignored. In the shallow formula, the $\mathbb{\exists}$ is ignored, and the $\boldsymbol{\exists}$ uses the information of the stretching to "undo the substitution". To provide this information on the location is the purpose of the stretching. To ensure that we really only "undo the substitution" instead of doing something weird, we need some further constraints, which are given by Definition 5.3 below.

Before, we need some additional notation. Let $\Gamma \blacktriangleleft \sigma$ be given, and let $A$ and $B$ be nodes in $\Gamma$ with $A$ being a quantifier node and $A \preccurlyeq B$. Then we write $A \rightharpoonup B$ if $A$ is a $\boldsymbol{\exists}$-node and there is a stretching edge between $A$ and $B$, or $A$ is an ordinary quantifier node and $B$ is the variable (or its negation) that is bound in $A$.

**Definition 5.3.** An expansion tree $\Gamma \blacktriangleleft \sigma$ is *appropriate*, if the following three conditions hold:

1. *Same-formula-condition:* For all nodes $A$, $B_1$, $B_2$,
   if $A \overset{\sigma}{\underset{+}{\rightharpoonup}} B_1$ and $A \overset{\sigma}{\underset{+}{\rightharpoonup}} B_2$, then $\lfloor B_1 \blacktriangleleft \sigma_1 \rfloor = \lfloor B_2 \blacktriangleleft \sigma_2 \rfloor$,
   if $A \overset{\sigma}{\underset{-}{\rightharpoonup}} B_1$ and $A \overset{\sigma}{\underset{-}{\rightharpoonup}} B_2$, then $\lfloor B_1 \blacktriangleleft \sigma_1 \rfloor = \lfloor B_2 \blacktriangleleft \sigma_2 \rfloor$,
   if $A \overset{\sigma}{\underset{+}{\rightharpoonup}} B_1$ and $A \overset{\sigma}{\underset{-}{\rightharpoonup}} B_2$, then $\lfloor B_1 \blacktriangleleft \sigma_1 \rfloor = \lfloor B_2 \blacktriangleleft \sigma_2 \rfloor^{\perp}$,
   where $\sigma_1$ and $\sigma_2$ are the restrictions of $\sigma$ to $B_1$ and $B_2$, respectively.
2. *No-capture-condition:* For all nodes $A_1$, $A_2$, $B_1$, $B_2$, where $A_1$ is a $\boldsymbol{\exists}$-node,
   if $A_1 \rightharpoonup B_1$ and $A_2 \rightharpoonup B_2$ and $A_1 \preccurlyeq A_2$ and $B_1 \preccurlyeq B_2$, then $B_1 \preccurlyeq A_2$.
3. *Not-free-condition:* For all subformulas $\mathbb{\exists} a.A$, the formula $\lfloor A \blacktriangleleft \sigma' \rfloor$ does not contain a free occurrence of $a$,
   where $\sigma'$ is the restriction of $\sigma$ to $A$.

The first condition above says that in a substitution a variable is instantiated everywhere by the same formula $B$. The second condition ensures that there is no variable capturing in such a substitution step. The third condition is exactly the side condition of the rule $\mathsf{f}{\downarrow}$ in Figure 2. For better explaining the three conditions above, we show in Figure 4 three examples of pairs $\Gamma \blacktriangleleft \sigma$ that are not appropriate: the first fails Condition 1, the second fails Condition 2, and the third fails Condition 3. In Figure 5 all three examples are appropriate.

We can characterize expansion trees $\Gamma \blacktriangleleft \sigma$ that are appropriate very naturally in terms of deep inference.

**Lemma 5.4.** *For every derivation*

$$\{\mathsf{n}{\downarrow}, \mathsf{f}{\downarrow}\} \left\| \begin{matrix} D \\ \\ C \end{matrix} \right. \mathscr{D} \tag{4}$$

*there is an appropriate pair* $\Gamma \blacktriangleleft \sigma$ *with*

$$D = \lceil \Gamma \blacktriangleleft \sigma \rceil \quad \textit{and} \quad C = \lfloor \Gamma \blacktriangleleft \sigma \rfloor \quad . \tag{5}$$

*Conversely, if* $\Gamma \blacktriangleleft \sigma$ *is appropriate, then there is a derivation* (4) *with* (5)*.*

*Proof.* We begin by extracting $\Gamma \blacktriangleleft \sigma$ from $\mathscr{D}$. For this, we proceed by induction on the length of $\mathscr{D}$. In the base case, let $\Gamma = D = C$ and $\sigma$ be empty. In the inductive case let $\mathscr{D}$ be

$$\begin{array}{c} D \\ \{\mathsf{n}{\downarrow}, \mathsf{f}{\downarrow}\} \Big\| \mathscr{D}' \\ \rho \dfrac{C'}{C} \end{array}$$

where $\rho$ is either

$$\mathsf{f}{\downarrow} \frac{S\{\exists a.A\}}{S\{A\}} \qquad \text{or} \qquad \mathsf{n}{\downarrow} \frac{S\{A\langle a \backslash B \rangle\}}{S\{\exists a.A\}}$$

and let $\Gamma' \blacktriangleleft \sigma'$ be obtained by induction hypothesis from $\mathscr{D}'$. In particular, $C' = \lfloor \Gamma' \blacktriangleleft \sigma' \rfloor$.

- If $\rho$ is $\mathsf{f}{\downarrow}$, then we construct $\Gamma$ from $\Gamma'$ as follows: If the $\exists$ to which $\mathsf{f}{\downarrow}$ is applied appears in $\Gamma'$ as ordinary $\exists$, then replace it by a $\mathbb{\exists}$-node, and let $\sigma = \sigma'$. If the $\exists$ is in fact a $\boldsymbol{\exists}$, then completely remove it, and let $\sigma$ be obtained from $\sigma'$ by removing all edges adjacent to that $\boldsymbol{\exists}$. In both cases the same-formula-condition and the no-capture-condition (5.3-1 and 5.3-2) are satisfied for $\Gamma \blacktriangleleft \sigma$ by induction hypothesis (because $\Gamma' \blacktriangleleft \sigma'$ is appropriate). The not-free-condition (5.3-3) holds because otherwise the $\mathsf{f}{\downarrow}$ would not be a valid rule application.

- If $\rho$ is $\mathsf{n}{\downarrow}$, we insert an $\boldsymbol{\exists}$-node at the position where the $\mathsf{n}{\downarrow}$-rule is applied and let $\sigma$ be obtained from $\sigma'$ by adding a positive (resp. negative) edge from this new $\boldsymbol{\exists}$ to every occurrence of $B$ in $C'$ which is replaced by $a$ (resp. $a^{\perp}$) in $C$. Then clearly the same-formula-condition is satisfied since it is everywhere the same $B$ which is substituted. Let us now assume by way of contradiction, that the no-capture-condition is violated. This means we have $A_1, A_2, B_1, B_2$ such that $A_1 \rightharpoonup B_1$ and $A_2 \rightharpoonup B_2$ and $A_1 \leqslant A_2$ and $B_1 \leqslant B_2$ and $B_1 \nleqslant A_2$. Note that by the definition of stretching we have that $A_1, A_2, B_1, B_2$ all sit on the same branch in $\Gamma$. Therefore we must have that $A_2' \leqslant B_1$, where $A_2'$ is child of $A_2$. Since the no-capture-condition is satisfied for $\Gamma' \blacktriangleleft \sigma'$ we have that either $A_1$ or $A_2$ is the newly introduced $\boldsymbol{\exists}$. Note that it cannot be $A_2$ because then $B_1$ would not be visible in $\lfloor \Gamma' \blacktriangleleft \sigma' \rfloor$ because it has been replaced by the variable $a$ bound in $A_1$. Since $B_2$ is inside $B_1$ it would also be invisible in $\lfloor \Gamma' \blacktriangleleft \sigma' \rfloor$. Hence the new $\boldsymbol{\exists}$ must be $A_1$. Without loss of generality, let $A_1 = \boldsymbol{\exists} a.A_1'$. Then our $\mathsf{n}{\downarrow}$-instance must look like

$$\mathsf{n}{\downarrow}\,\frac{S\{A_1'\{\mathsf{Q}b.A_2'\{B_1\{b\}\}\}\}}{S\{\exists a.\tilde{A}_1'\{\mathsf{Q}b.\tilde{A}_2'\{a\}\}\}} \tag{6}$$

where $a$ is substituted by $B_1\{b\}$ everywhere inside $\tilde{A}_1'\{\mathsf{Q}b.\tilde{A}_2'\{a\}\}$ and $\mathsf{Q}$ is either $\forall$ or $\exists$. Clearly, the variable $b$ is captured. Therefore (6) is not a valid rule application. Hence, the no-capture-condition must be satisfied. Finally, the not-free-condition could only be violated in a situation as above where $A_2$ is a $\reflectbox{$\mathbb{E}$}$-node. But since (6) is not valid, the not-free-condition does also hold.

Conversely, for constructing $\mathscr{D}$ from $\Gamma \blacktriangleleft \sigma$, we proceed by induction on the number of $\reflectbox{$\mathbb{E}$}$ and $\boldsymbol{\exists}$ in $\Gamma$. The base case is trivial. Now pick in $\Gamma$ an $\reflectbox{$\mathbb{E}$}$ or $\boldsymbol{\exists}$ which is minimal wrt. $\leqslant$, i.e., has no other $\reflectbox{$\mathbb{E}$}$ or $\boldsymbol{\exists}$ as ancestor.

- If we pick an $\reflectbox{$\mathbb{E}$}$, say $\Gamma = S\{\reflectbox{$\mathbb{E}$} a.A\}$, then let $\Gamma' = S\{\exists a.A\}$. By the not-free-condition, $a$ does not appear free in $\lfloor A \blacktriangleleft \sigma \rfloor$. Hence

$$\mathsf{f}{\downarrow}\,\frac{\lfloor \Gamma' \blacktriangleleft \sigma \rfloor}{\lfloor \Gamma \blacktriangleleft \sigma \rfloor}$$

  is a proper application of $\mathsf{f}{\downarrow}$.

- If we pick an $\boldsymbol{\exists}$, say $\Gamma = S\{\boldsymbol{\exists} a.A\}$, then let $\Gamma' = S\{A\}$ and let $\sigma'$ be obtained from $\sigma$ by removing all edges coming out of the selected $\boldsymbol{\exists} a$. We now have to check that

$$\mathsf{n}{\downarrow}\,\frac{\lfloor \Gamma' \blacktriangleleft \sigma' \rfloor}{\lfloor \Gamma \blacktriangleleft \sigma \rfloor}$$

  is a proper application of $\mathsf{n}{\downarrow}$. Indeed, by the same-formula-condition, every occurrence of $a$ bound by $\boldsymbol{\exists} a$ in $\lfloor \Gamma \blacktriangleleft \sigma \rfloor$ is substituted by the same formula in $\lfloor \Gamma' \blacktriangleleft \sigma' \rfloor$. The no-capture-condition ensures that no other variable is captured by this.

In both cases we have that $\lceil \Gamma' \blacktriangleleft \sigma' \rceil = \lceil \Gamma \blacktriangleleft \sigma \rceil$. Therefore we can proceed by induction hypothesis. □

We can explain the idea of the previous lemma by considering again the examples in Figures 4 and 5. To the non-appropriate examples in Figure 4 would correspond the following **incorrect** derivations:

$$\mathsf{n}{\downarrow}\,\frac{[(a \otimes b) \parr a^{\perp}]}{\exists c.[c \parr c^{\perp}]} \qquad \mathsf{n}{\downarrow}\,\frac{\forall b.[b^{\perp} \parr b]}{\exists a.\forall b.[a \parr b]} \qquad \begin{array}{l} \mathsf{f}{\downarrow}\,\dfrac{\exists a.([a \parr a^{\perp}] \otimes b^{\perp})}{([a \parr a^{\perp}] \otimes b^{\perp})} \\ \mathsf{n}{\downarrow}\,\dfrac{}{\exists c.(c \otimes b^{\perp})} \end{array}$$

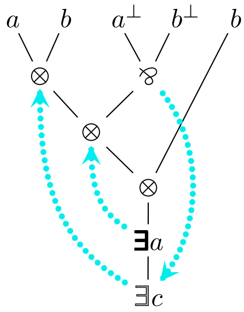
And to the appropriate examples in Figure 5 correspond the following **correct** derivations:

$$\mathsf{n}{\downarrow}\,\frac{[(a \otimes b) \parr a^{\perp}]}{\exists c.[(c \otimes b) \parr c^{\perp}]} \qquad \mathsf{n}{\downarrow}\,\frac{\forall b.[b^{\perp} \parr b]}{\forall b.\exists a.[a \parr b]} \qquad \begin{array}{l} \mathsf{n}{\downarrow}\,\dfrac{\exists a.([a \parr a^{\perp}] \otimes b^{\perp})}{\exists a.\exists c.(c \otimes b^{\perp})} \\ \mathsf{f}{\downarrow}\,\dfrac{}{\exists c.(c \otimes b^{\perp})} \end{array}$$

**Observation 5.5.** When we translate derivations (4) into appropriate pairs $\Gamma \blacktriangleleft \sigma$ we can lose more information than than just the information about rule permutation Consider the following example:

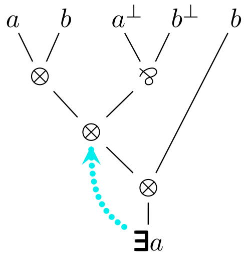
$$\begin{array}{l} \mathsf{n}{\downarrow}\,\dfrac{(a \otimes b \otimes [a^{\perp} \parr b^{\perp}] \otimes b)}{\exists c.(c \otimes c^{\perp} \otimes b)} \\ \mathsf{n}{\downarrow}\,\dfrac{}{\exists c.\exists a.(a \otimes b)} \\ \mathsf{f}{\downarrow}\,\dfrac{}{\exists a.(a \otimes b)} \end{array} \tag{7}$$

Translating it naively into an expansion tree would yield something like



(8)

where the ∃ must be at the same time a bold and virtual existential. In our translation these are eliminated:

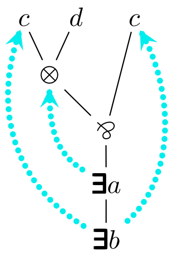


(9)

Translating this back into a derivation gives us

$$\mathsf{n}{\downarrow}\,\frac{(a\otimes b\otimes[a^{\perp}\parr b^{\perp}]\otimes b)}{\exists a.(a\otimes b)} \tag{10}$$

So far, there is no strong argumentation in favor or against the identification of the derivations (7) and (10). It is a straightforward exercise to extend the definitions in this paper to allow ∃ and avoid the identification of (7) and (10). We chose not to do so in order to simplify the theory. A similar situation occurs in the translation from expansion trees to derivations. Consider



(11)

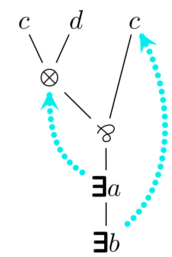
which comes from the derivation

$$\mathsf{n}{\downarrow}\,\frac{\mathsf{n}{\downarrow}\,\dfrac{[(c\otimes d)\parr c]}{\exists b.[(b\otimes d)\parr b]}}{\exists b.\exists a.[a\parr b]} \tag{12}$$

However, if we translate (11) back into a derivation, following the construction of the proof above, then we get

$$\mathsf{n}{\downarrow}\,\frac{\mathsf{n}{\downarrow}\,\dfrac{[(c\otimes d)\parr c]}{\exists a.[a\parr c]}}{\exists b.\exists a.[a\parr b]} \tag{13}$$

which would, when translated back into an expansion tree, yield



(14)

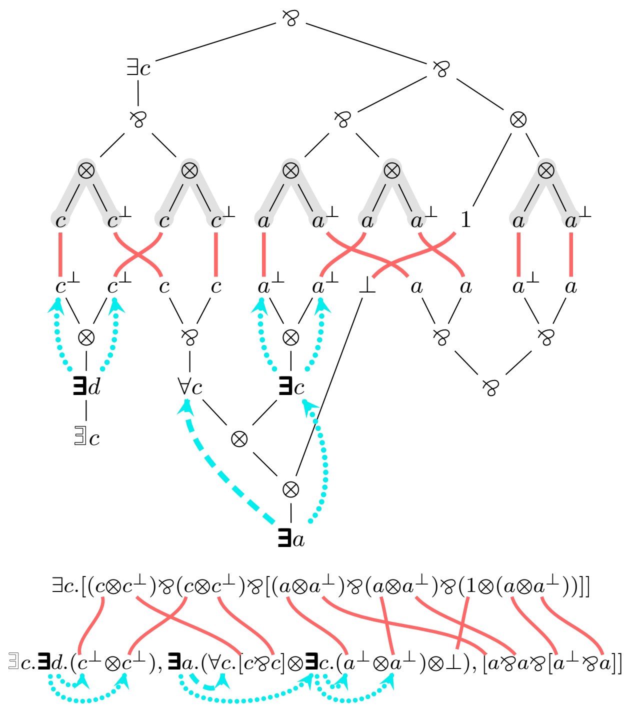


Figure 6: Two ways of writing a proof graph

As before, it is straightforward to modify the definitions such that either the identification of (12) and (13) is avoided or such that, for example, (11) would be disallowed. However, I could not find any strong argumentation in favor of one or the other. It remains a task of future work to find these arguments, for example via algebraic considerations, in the same sense as the axioms *-autonomous categories gave justification for identifications made by proof nets for MLL.

# 6 Proof graphs for MLL2

After studying the "quantifier part" of MLL2, we will now look into the "propositional part". Since the units are present, we have to have more structure than ordinary proof nets. Furthermore, in linear logic, we cannot fully separate the quantifier part from the propositional part, as it is the case with classical logic.

We follow the ideas presented in [SL04, LS06] where the axiom linking of multiplicative proof nets has been replaced by a *linking formula* to accommodate the units $1$ and $\bot$. In such a linking formula, the ordinary axiom links are replaced by $\otimes$-nodes, which are then connected by $⅋$s. A unit can then be attached to a sublinking by another $\otimes$, and so on. Here we extend the syntax for the linking formula by an additional construct to accommodate the quantifiers. The set $\mathscr{L}$ of *linking formulas* is generated by the grammar

$$\mathscr{L} ::= \bot \mid (\mathscr{A} \otimes \mathscr{A}^\bot) \mid (1 \otimes \mathscr{L}) \mid [\mathscr{L} ⅋ \mathscr{L}] \mid \exists \mathscr{A}.\mathscr{L}$$

The basic idea of our proof graphs is to attach a linking formula to an expansion tree. This is similar to Miller's idea of attaching a *mating* [And76] to an expansion tree in classical logic [Mil87].

**Definition 6.1.** A *pre-proof graph* is a quadruple, denoted by $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$, where $P$ a linking formula, $\Gamma$ is an expanded sequent, $\sigma$ is a stretching for $\Gamma$, and $\nu$ is a bijection $\mathscr{L}\Gamma \xrightarrow{\nu} \mathscr{L}P$ from the leaves of $\Gamma$ to the leaves of $P$, such that only dual atoms/units are paired up. If $\Gamma$ is simple, we say that the pre-proof graph is *simple*. In this case $\sigma$ is empty, and we can simply write $P \overset{\nu}{\triangleright} \Gamma$.

The "pre-" means that we do not yet know whether it really comes from an actual proof. Due to the mobility of $\bot$, we need to introduce the notion of proof graph. A proof net will then be an equivalence class of proof graphs (see also [SL04, LS06] for details). If there are no units present, then the notions of (pre-)proof graph and (pre-)proof net coincide.

For $B \in \mathscr{L}\Gamma$ we write $B^\nu$ for its image under $\nu$ in $\mathscr{L}P$. When we draw a pre-proof graph $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$, then we write $P$ above $\Gamma$ (as trees or as strings) and the leaves are connected by edges according to $\nu$. Figure 6 shows an example written in both ways. To help the reader we marked the traditional "axiom links" in the tree-like version.

Let us now turn our attention towards correctness. For this we concentrate first on simple pre-proof graphs and begin with pure multiplicative correctness, using the standard Danos-Regnier criterion [DR89].

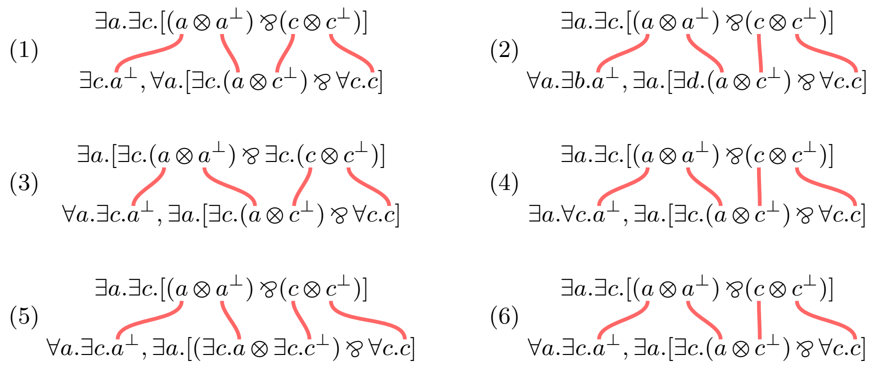



Figure 7: Examples (1)–(5) are not well-nested, only (6) is well-nested

**Definition 6.2.** A *switching* $s$ of a simple pre-proof graph $P \overset{\nu}{\triangleright} \Gamma$ is the graph that is obtained from $P \overset{\nu}{\triangleright} \Gamma$ by removing for each $\parr$-node one of the two edges connecting it to its children. A simple pre-proof graph $P \overset{\nu}{\triangleright} \Gamma$ is *multiplicatively correct* if all its switchings are acyclic and connected [DR89]. For a pre-proof graph $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$ we define *multiplicative correctness* accordingly, but we ignore the edges of the stretching when checking acyclicity and connectedness.

For multiplicative correctness the quantifiers are treated as unary connectives and are therefore completely irrelevant. The example in Figure 6 is multiplicatively correct. For involving the quantifiers into a correctness criterion, we need some more conditions.

Let $s$ be a switching for $P \overset{\nu}{\triangleright} \Gamma$, and let $A$ and $B$ be two nodes in $\Gamma$. We write $A \frown\!(s)\!\frown B$ for saying that there is a path in $s$ from $A$ to $B$ which starts from $A$ going up to one of its children and which comes into $B$ down from one of its children, and we write $A \frown\!(s)\!\smile B$ if the path comes into $B$ from its parent below. Similarly, we write $A \smile\!(s)\!\frown B$ (resp. $A \smile\!(s)\!\smile B$) if the path starts from $A$ going down to its parent and comes into $B$ from above (resp. from below).

Let $\Gamma$ be a simple sequent, and let $A$ and $B$ be nodes in $\Gamma$ with $A \preccurlyeq B$. Then the *quantifier depth* of $B$ in $A$, denoted by $\nabla_A B$, is defined to be the number of quantifier nodes on the path from $A$ to $B$ (including $A$ if it happens to be an $\forall a$ or an $\exists a$, but not including $B$). Similarly we define $\nabla_\Gamma B$. Now assume we have a simple pre-proof graph $P \overset{\nu}{\triangleright} \Gamma$ and quantifier nodes $A'$ in $P$ and $A$ in $\Gamma$. We say $A$ and $A'$ are *partners* if there is a leaf $B \in \mathcal{L}\Gamma$ with $A \preccurlyeq B$ in $\Gamma$, and $A' \preccurlyeq B^\nu$ in $P$, and $\nabla_A B = \nabla_{A'} B^\nu$. We denote this by $A' \overset{P\,\Gamma}{\leftrightarrow} A$.

**Definition 6.3.** We say a simple pre-proof graph $P \overset{\nu}{\triangleright} \Gamma$ is *well-nested* if the following five conditions are satisfied:

1. *Same-depth-condition:* For every $B \in \mathcal{L}\Gamma$, we have $\nabla_\Gamma B = \nabla_P B^\nu$.
2. *Same-variable-condition:* whenever $A' \overset{P\,\Gamma}{\leftrightarrow} A$, then $A'$ and $A$ quantify the same variable.
3. *One-$\exists$-condition:* For every quantifier node $A$ in $\Gamma$ there is exactly one $\exists$-node $A'$ in $P$ with $A' \overset{P\,\Gamma}{\leftrightarrow} A$.
4. *One-$\forall$-condition:* For every $\exists$-node $A'$ in $P$ there is exactly one $\forall$-node $A$ in $\Gamma$ with $A' \overset{P\,\Gamma}{\leftrightarrow} A$.
5. *No-down-path-condition:* If $A' \overset{P\,\Gamma}{\leftrightarrow} A_1$ and $A' \overset{P\,\Gamma}{\leftrightarrow} A_2$ for some $A'$ in $P$ and $A_1, A_2$ in $\Gamma$, then there is no switching $s$ with $A_1 \smile\!(s)\!\smile A_2$.

Every quantifier node in $P$ must be an $\exists$, and every quantifier node in $\Gamma$ has exactly one of them as partner. On the other hand, an $\exists$ in $P$ can have many partners in $\Gamma$, but exactly one of them has to be an $\forall$. Following Girard [Gir87], we can call an $\exists$ in $P$ together with its partners in $\Gamma$ the *doors of an* $\forall$*-box* and the sub-graph induced by the nodes that have such a door as ancestor is called the $\forall$*-box* associated to the unique $\forall$-door. Even if the boxes are not really present, we can use the terminology to relate our work to Girard's. There should be no surprise here: If we claim at some point that our proof graphs capture the essential information of a proof, we must be able to recover a sequent calculus proof, which necessarily contains the Girard-boxes. Furthermore, all the properties of these boxes that are postulated in [Gir87], e.g., that every box is correct in itself, follow from the global multiplicative correctness and the five conditions above. In order to help the reader to understand these five conditions, we show in Figure 7 six simple pre-proof graphs, where the first fails Condition 1, the second one fails Condition 2, and so on; only the sixth one is well-nested.

**Definition 6.4.** A simple pre-proof graph $P \overset{\nu}{\triangleright} \Gamma$ is *correct* if it is well-nested and multiplicatively correct. In this case we will also speak of a *simple proof graph*.

**Definition 6.5.** We say that a pre-proof graph $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$ is *correct* if the simple pre-proof graph $P \overset{\nu}{\triangleright} \lceil \Gamma \blacktriangleleft \sigma \rceil$ is correct and the expansion tree $\Gamma \blacktriangleleft \sigma$ is appropriate. In this case we say that $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$ is a *proof graph* and $\lfloor \Gamma \blacktriangleleft \sigma \rfloor$ is its *conclusion*.

The example in Figure 6 is correct. There $\lceil \Gamma \blacktriangleleft \sigma \rceil$ is

$$\vdash \exists c.(c^\perp \otimes c^\perp), (\forall c.[c \parr c] \otimes (a^\perp \otimes a^\perp) \otimes \perp), [a \parr a \parr [a^\perp \parr a]]$$

and the conclusion $\lfloor \Gamma \blacktriangleleft \sigma \rfloor$ is

$$\vdash \exists d.(d \otimes d), \exists a.(a^\perp \otimes a \otimes \perp), [a \parr a \parr [a^\perp \parr a]] \quad .$$

## 7 The relation between simple proof graphs and deep inference

With Lemma 5.4 we already gave a deep inference characterization of expansion trees. In this section we do something similar for simple proof graphs.

Let us begin with a characterization of linking formulas.

**Lemma 7.1.** *An* MLL2 *formula $P$ is a linking formula if and only if there is a derivation*

$$\{\mathsf{ai}\downarrow, \perp\downarrow, 1\downarrow, \mathsf{e}\downarrow\} \begin{array}{c} 1 \\ \Big\| \mathscr{D} \\ P^\perp \end{array} \quad . \tag{15}$$

*Proof.* We can proceed by structural induction on $P$ to construct $\mathscr{D}$. The base case is trivial. Here are the four inductive cases:

$$\mathsf{ai}\downarrow \frac{\{1\}}{[a^\perp \parr a]} \qquad \perp\downarrow \frac{\begin{array}{c} 1 \\ \Big\| \mathscr{D}' \\ A \end{array}}{[\perp \parr A]} \qquad \begin{array}{c} 1 \\ \Big\| \mathscr{D}' \\ 1\downarrow \dfrac{B}{(1 \otimes B)} \\ \Big\| \mathscr{D}'' \\ (A \otimes B) \end{array} \qquad \begin{array}{c} \mathsf{e}\downarrow \dfrac{1}{\forall a.1} \\ \Big\| \mathscr{D}' \\ \forall a.A \end{array}$$

where $\mathscr{D}'$ and $\mathscr{D}''$ always exist by induction hypothesis. Conversely, we proceed by induction on the length of $\mathscr{D}$ to show that $P$ is a linking formula. We show only the case where the bottommost rule in $\mathscr{D}$ is a $\mathsf{ai}\downarrow$, i.e., $\mathscr{D}$ is

$$\perp\downarrow \frac{\begin{array}{c} 1 \\ \Big\| \mathscr{D}' \\ S\{1\} \end{array}}{S[a^\perp \parr a]} \quad .$$

By induction hypothesis $S\{1\}^\perp = P\{\perp\}$ is a linking for some context $P\{\ \}$. Hence $S[a^\perp \parr a]^\perp = P(a \otimes a^\perp)$ is also a linking. The other cases are similar. □

**Definition 7.2.** If a linking has the shape $S_1(1 \otimes S_2(a \otimes a^\perp))$ for some contexts $S_1\{\ \}$ and $S_2\{\ \}$, then we say that the 1 *governs* the pair $(a \otimes a^\perp)$. Now let $P_1$ and $P_2$ be two linkings. We say that $P_1$ *is weaker than* $P_2$, denoted by $P_1 \lesssim P_2$, if

- $\varnothing P_1 = \varnothing P_2$,
- $P_1$ and $P_2$ contain the same set of $\exists$-nodes, and for every $\exists$-node, its set of leaves is the same in $P_1$ and $P_2$, and
- whenever a 1 governs a pair $(a \otimes a^\perp)$ in $P_2$, then it also governs this pair in $P_1$.

This $\lesssim$ relation can also be characterized by deep inference derivations. For this, we also use the following inference rules:

$$\alpha\uparrow \frac{S(A \otimes (B \otimes C))}{S((A \otimes B) \otimes C)} \qquad \text{and} \qquad \sigma\uparrow \frac{S(A \otimes B)}{S(B \otimes A)} \tag{16}$$

which are the duals for $\alpha\downarrow$ and $\sigma\downarrow$, respectively.

**Lemma 7.3.** *Let $P_1$ and $P_2$ be two linkings. Then the following are equivalent*

1. $P_1 \lesssim P_2$.
2. *There is a derivation* $\{\alpha\downarrow, \sigma\downarrow, \mathsf{rs}\} \left\| \begin{matrix} P_1 \\ \mathscr{D} \\ P_2 \end{matrix} \right.$.
3. *Dually, there is a derivation* $\{\alpha\uparrow, \sigma\uparrow, \mathsf{ls}\} \left\| \begin{matrix} P_2^\perp \\ \mathscr{D}' \\ P_1^\perp \end{matrix} \right.$.
4. *The simple pre-proof graph $P_2 \triangleright P_1^\perp$ is correct.*

*Proof.* $1 \Rightarrow 2$: The only way in which $P_1$ and $P_2$ can differ from each other are the $\parr$-trees above the pairs $(a \otimes a^\perp)$ and where in these trees the 1-occurrences are attached. Therefore, the rules for associativity and commutativity of $\parr$ and the rule

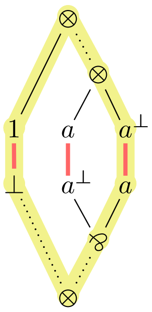
$$\mathsf{rs}\,\frac{S(1 \otimes [B \parr C])}{S[(1 \otimes B) \parr C]}$$

are sufficient to transform $P_1$ into $P_2$.

$2 \Rightarrow 3$: The derivation $\mathscr{D}'$ is the dual of $\mathscr{D}$.

$3 \Rightarrow 4$: We proceed by induction on the length of $\mathscr{D}'$. Clearly $P_2 \triangleright P_2^\perp$ is correct. Furthermore, all three inference rules $\alpha\uparrow$, $\sigma\uparrow$, and $\mathsf{ls}$ preserve correctness.

$4 \Rightarrow 1$: We have $\varnothing P_1 = \varnothing P_2$ because $P_2 \triangleright P_1^\perp$ is a simple proof graph. The second condition in Definition 7.2 follows immediately from the well-nestedness of $P_2 \triangleright P_1^\perp$ and the fact that $P_1$ and $P_2$ are both linkings, i.e., do not contain $\forall$-nodes. Therefore, we only have to check the last condition. Assume, by way of contradiction, that there is a 1-occurrence which governs a pair $(a \otimes a^\perp)$ in $P_2$ but not in $P_1$, i.e., $P_2 = S_1(1 \otimes S_2(a \otimes a^\perp))$ for some contexts $S_1\{\ \}$ and $S_2\{\ \}$, and $P_1 = S_3[S_4\{1\} \parr S_5(a \otimes a^\perp)]$ for some contexts $S_3\{\ \}$, $S_4\{\ \}$, and $S_5\{\ \}$. This means we have the following situation in $P_2 \triangleright P_1^\perp$



which clearly fails the acyclicity condition. □

The next step is to characterize correctness via deep inference.

**Lemma 7.4.** *A simple pre-proof graph $P \overset{\nu}{\triangleright} \Gamma$ is correct if and only if there is a linking $P'$ with $P' \lesssim P$ and a derivation*

$$\{\alpha\downarrow, \sigma\downarrow, \mathsf{ls}, \mathsf{rs}, \mathsf{u}\downarrow\} \left\| \begin{matrix} P'^\perp \\ \mathscr{D} \\ \Gamma \end{matrix} \right. , \tag{17}$$

*such that $\nu$ coincides with the bijection induced by the flow graph of $\mathscr{D}$.*

As an example, consider the derivation in (3) which corresponds to example (6) in Figure 7.

Before we can give the proof of this lemma, we need a series of additional lemmas that we have to show first. We also use the following notation: Let $A$ and $B$ be nodes in $\Gamma$ with $A \not\ast B$ and $B \not\ast A$. Then we write $A \overset{\Gamma}{\underset{\otimes}{\frown}} B$ if the first common ancestor of $A$ and $B$ is a $\otimes$, and we write $A \overset{\Gamma}{\underset{\parr}{\frown}} B$ if it is a $\parr$, or if $A$ and $B$ appear in different formulas of $\Gamma$. We will also sometimes identify a sequent $\vdash A_1, \ldots, A_n$ with the formula $[A_1 \parr \cdots \parr A_n]$.

**Lemma 7.5.** *Let*

$$P(a \otimes a^\perp) \overset{\nu}{\triangleright} S[\forall b. A'\{a^\perp\} \parr (B'\{a\} \otimes B'')] \tag{18}$$

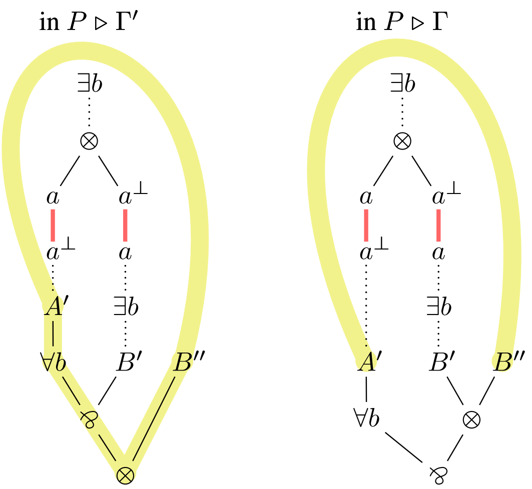
*be a simple proof graph, where $S\{\ \}$, $A'\{\ \}$, and $B'\{\ \}$ are arbitrary contexts, $P\{\ \}$ is a linking formula context, and $\nu$ pairs up the shown occurrences of $a$ and $a^{\perp}$. Then*

$$P(a \otimes a^{\perp}) \overset{\nu}{\triangleright} S([\forall b.A'\{a^{\perp}\} \parr B'\{a\}] \otimes B'') \tag{19}$$

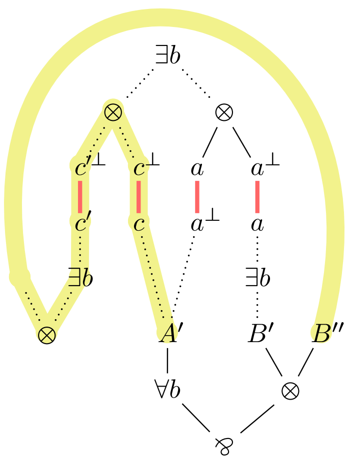
*is also correct.*

*Proof.* Let us abbreviate (18) by $P \overset{\nu}{\triangleright} \Gamma$ and (19) by $P \overset{\nu}{\triangleright} \Gamma'$. By way of contradiction, assume that $P \overset{\nu}{\triangleright} \Gamma'$ is not correct.

If it is not multiplicatively correct then there is a switching $s$ which is either disconnected or cyclic. If it is disconnected, then we get from $s$ immediately a disconnected switching for $P \overset{\nu}{\triangleright} \Gamma$. So, let us assume $s$ is cyclic. The only modification from $\Gamma$ to $\Gamma'$ that could produce such a cycle is the change from $A'\{a^{\perp}\} \underset{\parr}{\overset{\Gamma}{\smile}} B''$ to $A'\{a^{\perp}\} \underset{\otimes}{\overset{\Gamma'}{\smile}} B''$. Hence, we must have a path $A'\{a^{\perp}\} \frown\textcircled{s}\frown B''$, which is also present in $P \overset{\nu}{\triangleright} \Gamma$. Note that this path cannot pass through $a^{\perp}$ and $a$ because otherwise we could use $(B'\{a\} \otimes B'')$ to get a cyclic switching for $P \overset{\nu}{\triangleright} \Gamma$. Furthermore, because $P \overset{\nu}{\triangleright} \Gamma$ is well-nested, there is an $\exists b$-node inside $B'\{a\}$ below $a$. We can draw the following pictures to visualize the situation:

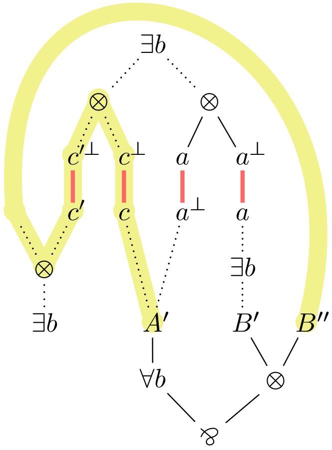



Now, let $c$ be the leaf at which our path leaves $A'\{a^{\perp}\}$ and goes into $P$, and let $c'$ be the leaf at which it leaves $P$ and comes back into $\Gamma$. by well-nestedness of $P \overset{\nu}{\triangleright} \Gamma$, there must be some $\exists b$-node somewhere in $\Gamma$ below $c'$. We also know that our path, coming into $\Gamma$ at $c'$, goes first down, and at some point goes up again. This turning point must be some $\otimes$-node below $c'$. Since the $\exists b$-node and the $\otimes$-node are both on the path from $c'$ to the root of the formula, one must be an ancestor of the other. Let us first assume the $\otimes$ is below the $\exists b$. Then our path is of the shape

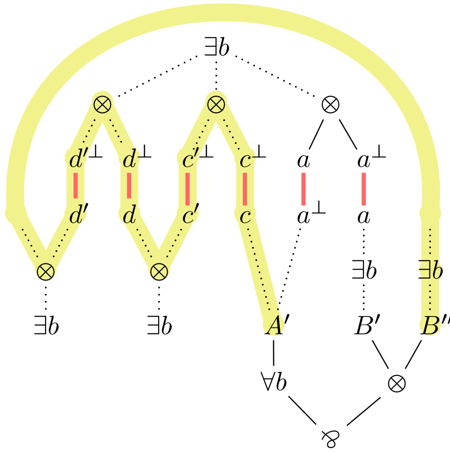


(20)

This, however, is a contradiction to the well-nestedness of $P \overset{\nu}{\triangleright} \Gamma$ because it violates the no-down-path-condition (6.3-5) because there is a path between the $\exists b$ below the $c'$ and the $\exists b$ below the $a$. Therefore the $\otimes$ must be above the $\exists b$. The situation is now as follows:

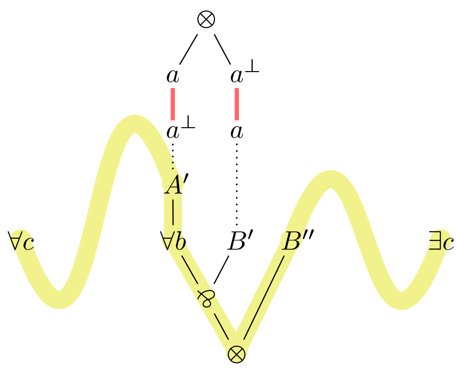



From the $\otimes$, the path must go up again. Without loss of generality, assume it leaves $\Gamma$ at $d$ and reenters $\Gamma$ at $d'$. For the same reasons as above, there must be an $\exists b$ and a $\otimes$ below $d'$. And so on. There are two possibilities: either at some point the $\otimes$ is below the $\exists b$, which gives us a violation of the no-down-path-condition as in (20), or we reach eventually $B''$:



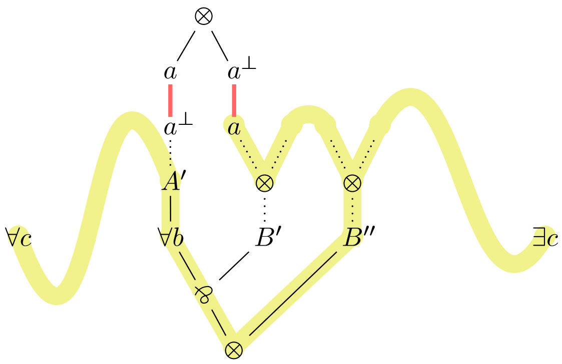

For the same reasons as above, there must be an $\exists b$ inside $B''$, and we get immediately a violation of the no-down-path-condition because of the short path between the two $\exists b$ above $B'$ and $B''$. Consequently, $P \overset{\nu}{\triangleright} \Gamma'$ must be multiplicatively correct.

Let us therefore assume $P \overset{\nu}{\triangleright} \Gamma'$ is not well-nested. The same-depth-condition and the same-variable-condition (6.3-1 and 6.3-2) must hold in $P \overset{\nu}{\triangleright} \Gamma'$ because they hold in $P \overset{\nu}{\triangleright} \Gamma$ and the quantifier structure is identical in $\Gamma$ and $\Gamma'$. For the same reasons also the one-$\exists$-condition and the one-$\forall$-condition (6.3-3 and 6.3-4) must hold in $P \overset{\nu}{\triangleright} \Gamma'$. Therefore, it must be the no-down-path-condition which is violated. This means we must have in $\Gamma'$ two quantifier nodes, say $\forall c$ and $\exists c$, connected by a path $\forall c \overset{s}{\smile} \exists c$ in some switching $s$. Because this path is not present in $P \overset{\nu}{\triangleright} \Gamma$ it must pass through the new $\otimes$ between $\forall b.A'\{a^\perp\}$ and $B''$, as follows:



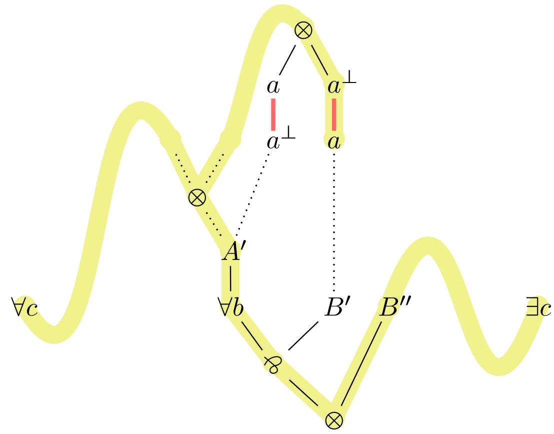

(21)

Since $P \overset{\nu}{\triangleright} \Gamma'$ is multiplicatively correct, the switching $s$ must be connected. Therefore there is in $s$ a path from the $\forall b$-node to the $a$ inside $B'$. This new path must follow the path between $\forall c$ and $\exists c$ for some steps in one direction. Hence, we either have



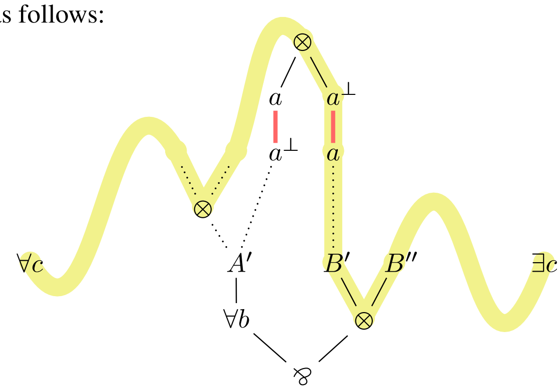

(22)

or




(23)

Clearly, (22) violates the acyclicity condition for $P \overset{\nu}{\triangleright} \Gamma'$ as well as for $P \overset{\nu}{\triangleright} \Gamma$. And from (23), we can obtain a switching for $P \overset{\nu}{\triangleright} \Gamma$ with a path $\forall c \smile\!\textcircled{\scriptsize s}\!\smile \exists c$ as follows:




(24)

Contradiction. (Note that although in (23) and (24) the path does not go through the $a^\perp$ inside $A'$, this case is not excluded by the argument.) □

**Lemma 7.6.** *Let*

$$P(a \otimes a^\perp) \overset{\nu}{\triangleright} S[(A'' \otimes A'\{a^\perp\}) \parr (B'\{a\} \otimes B'')] \tag{25}$$

*be a simple proof graph. Then at least one of*

$$P(a \otimes a^\perp) \overset{\nu}{\triangleright} S([(A'' \otimes A'\{a^\perp\}) \parr B'\{a\}] \otimes B'') \tag{26}$$

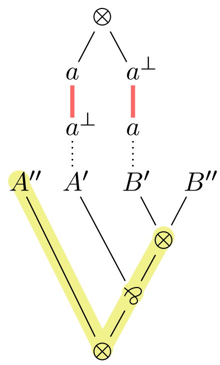
*and*

$$P(a \otimes a^\perp) \overset{\nu}{\triangleright} S(A'' \otimes [A'\{a^\perp\} \parr (B'\{a\} \otimes B'')]) \tag{27}$$

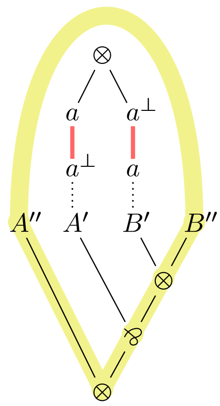
*is also correct.*

*Proof.* We will abbreviate (25) by $P \triangleright \Gamma$, (26) by $P \triangleright \Gamma'$, and (27) by $P \triangleright \Gamma''$.

We start by showing that one of $P \triangleright \Gamma'$ and $P \triangleright \Gamma''$ has to be multiplicatively correct. We consider here only the acyclicity condition and leave connectedness to the reader. First, assume that there is a switching $s'$ for $P \triangleright \Gamma'$ that is cyclic. Then the cycle must pass through $A''$, the root $\otimes$ and the $\parr$ as follows:

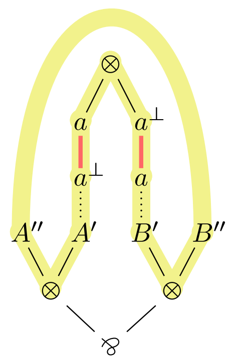



Otherwise we could construct a switching with the same cycle in $\pi$. If our cycle continues through $B''$, i.e.,

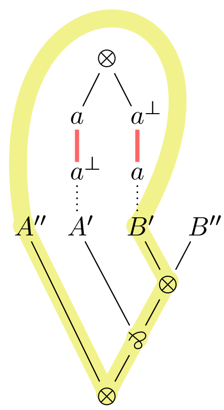



(28)

then we can use the path from $A''$ to $B''$ (which cannot go through $A'$ or $B'$) to construct a cyclic switching $s$ in $P \triangleright \Gamma$ as follows:



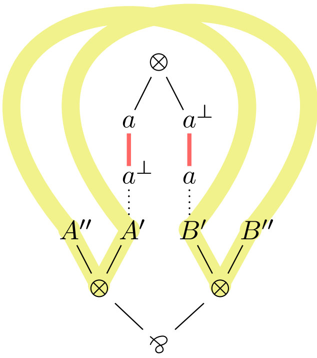

Hence, the cycle in $s'$ goes through $B'$, giving us a path from $A''$ to $B'$ (not passing through $A'$):




(29)

By the same argumentation we get a switching $s''$ in $P \triangleright \Gamma''$ with a path from $A'$ to $B''$, not going through $B'$. From $s'$

and $s''$, we can now construct a switching $s$ for $P \triangleright \Gamma$ with a cycle as follows:




which contradicts the correctness of $P \triangleright \Gamma$.

We now have to show that $P \triangleright \Gamma'$ and $P \triangleright \Gamma''$ are both well-nested. This can be done in almost literally the same way as in the proof of Lemma 7.5. □

**Lemma 7.7.** *Let*

$$P(a \otimes a^\perp) \overset{\nu}{\triangleright} S[\forall b.A'\{a^\perp\} \parr \exists b.B'\{a\}] \tag{30}$$

*be a simple proof graph. Then*

$$P(a \otimes a^\perp) \overset{\nu}{\triangleright} S\{\forall b.[A'\{a^\perp\} \parr B'\{a\}]\} \tag{31}$$

*is also correct.*

*Proof.* Multiplicative correctness of (31) follows immediately, because the $\otimes$-$\parr$-structure is the same as in (30). Furthermore, all five conditions in Definition 6.3 are obviously preserved by going from (30) to (31). Hence (31) is correct. □

**Lemma 7.8.** *Let*

$$P(a \otimes a^\perp) \overset{\nu}{\triangleright} S[a^\perp \parr (B'\{a\} \otimes B'')] \tag{32}$$

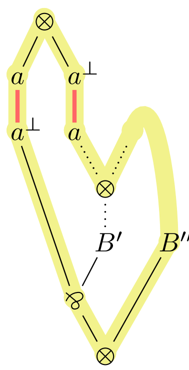
*be a simple proof graph. Then*

$$P(a \otimes a^\perp) \overset{\nu}{\triangleright} S([a^\perp \parr B'\{a\}] \otimes B'') \tag{33}$$

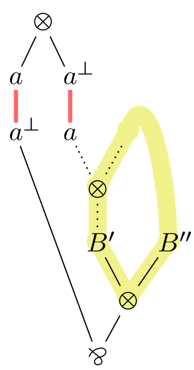
*is also correct.*

*Proof.* As before, we abbreviate (32) by $P \triangleright \Gamma$ and (33) by $P \triangleright \Gamma'$. Well-nestedness of $P \triangleright \Gamma'$ follows trivially from the well-nestedness of $P \triangleright \Gamma$. By way of contradiction, assume $P \triangleright \Gamma'$ is not multiplicatively correct. Since connectedness is trivial, assume there is a cyclic switching $s$. If the cycle does not involve the $\otimes$ between $a^\perp$ and $B''$, then we immediately have a cyclic switching for $P \triangleright \Gamma$. Since the cycle involves $a^\perp$, it must also involve $a$. Therefore it must leave $B'\{a\}$ at some other leaf, and finally enter $B''$ from above, as shown below on the left.

in $P \triangleright \Gamma'$




in $P \triangleright \Gamma$




This allows us to construct a cyclic switching for $P \triangleright \Gamma$, as shown on the right above. Contradiction. □

We can now complete our proof.

*Proof of Lemma 7.4.* Let a simple pre-proof graph $P \overset{\nu}{\triangleright} \Gamma$ be given, and assume we have a linking $P' \lesssim P$ and derivation $\mathscr{D}$ as in (17) whose flow-graph determines $\nu$. By Lemma 7.3 we have a derivation $\mathscr{D}_1$ such that

$$\begin{array}{c} P^{\perp} \\ \{\alpha\uparrow, \sigma\uparrow, \mathsf{ls}\} \left\Vert \vphantom{X} \right. \mathscr{D}_1 \\ P'^{\perp} \\ \{\alpha\downarrow, \sigma\downarrow, \mathsf{ls}, \mathsf{rs}, \mathsf{u}\downarrow\} \left\Vert \vphantom{X} \right. \mathscr{D} \\ \Gamma \end{array} \quad . \tag{34}$$

Now we proceed by induction on the length of $\mathscr{D}_1$ and $\mathscr{D}$ to show that $P \overset{\nu}{\triangleright} \Gamma$ is multiplicatively correct and well-nested. In the base case it is easy to see that $P \triangleright P^{\perp}$ has the desired properties. Now it remains to show that all rules $\alpha\downarrow, \sigma\downarrow, \alpha\uparrow, \sigma\uparrow, \mathsf{ls}, \mathsf{rs}, \mathsf{u}\downarrow$ preserve multiplicative correctness and well-nestedness. For multiplicative correctness it is easy: for $\mathsf{u}\downarrow$ it is trivial because it does not change the $\parr$-$\otimes$-structure of the graph, and for the other rules it is well-known. That well-nestedness is preserved is also easy to see: rules $\alpha\downarrow, \sigma\downarrow, \alpha\uparrow, \sigma\uparrow, \mathsf{ls}, \mathsf{rs}$ do not modify the $\forall$-$\exists$-structure of the graph, and therefore trivially preserve Conditions 1–4 in Definition 6.3. For the no-down-path condition it suffices to observe that it cannot happen that a $\underset{\parr}{\frown}$ is changed into $\underset{\otimes}{\frown}$ while going down in a derivation. Finally, it is easy to see that $\mathsf{u}\downarrow$ preserves all five conditions in Definition 6.3.

Conversely, assume $P \overset{\nu}{\triangleright} \Gamma$ is well-nested and multiplicatively correct. For constructing $\mathscr{D}$, we will again need the rule $\mathsf{v}\downarrow$ that has already been used in the proof of Theorem 3.4.

We proceed by induction on the distance between $P^{\perp}$ and $\Gamma$. For defining this formally, let $A$ be a simple formula and define $\#_{\parr} A$ to be the number of pairs $\langle a, b\rangle$ with $a, b \in \varnothing A$ and $a \overset{A}{\underset{\parr}{\frown}} b$, and define $\#_{\exists} A$ to be the number of $\exists$-nodes in $A$. Now observe that $P^{\perp}$ and $\Gamma$ have the same set of leaves. We can therefore define

$$\begin{array}{rcl} \delta_{\parr}\langle P^{\perp}, \Gamma\rangle & = & \#_{\parr}\Gamma - \#_{\parr} P^{\perp} \\ \delta_{\exists}\langle P^{\perp}, \Gamma\rangle & = & \#_{\exists}\Gamma - \#_{\exists} P^{\perp} \end{array}$$

Note that because of acyclicity it can never happen that for some $a, b \in \varnothing\Gamma$ we have $a \overset{P^{\perp}}{\underset{\parr}{\frown}} b$ and $a \overset{\Gamma}{\underset{\otimes}{\frown}} b$. Therefore $\delta_{\parr}\langle P^{\perp}, \Gamma\rangle$ is the number of pairs $a, b \in \varnothing\Gamma$ with $a \overset{P^{\perp}}{\underset{\otimes}{\frown}} b$ and $a \overset{\Gamma}{\underset{\parr}{\frown}} b$. Furthermore, observe that by definition there cannot be any $\exists$-node in $P^{\perp}$. Hence $\delta_{\exists}\langle P^{\perp}, \Gamma\rangle = \#_{\exists}\Gamma$. Now define the *distance between* $P^{\perp}$ *and* $\Gamma$ to be the pair

$$\delta\langle P^{\perp}, \Gamma\rangle \;=\; \langle \delta_{\parr}\langle P^{\perp}, \Gamma\rangle, \delta_{\exists}\langle P^{\perp}, \Gamma\rangle\rangle$$

where we assume the lexicographic ordering.

Let us now pick in $\Gamma$ a pair of dual atoms, say $a^{\perp}$ and $a$, which appear in the same "axiom link" in $P$, i.e., $P$ is $P(a \otimes a^{\perp})$. We now make a case analysis on the relative position of $a^{\perp}$ and $a$ to each other in $\Gamma$. Because of acyclicity we must have $a^{\perp} \overset{\Gamma}{\underset{\parr}{\frown}} a$. This means $\Gamma = S[A\{a^{\perp}\} \parr B\{a\}]$ for some contexts $S\{\ \}$, $A\{\ \}$, and $B\{\ \}$. Without loss of generality, we assume that neither $A$ nor $B$ has a $\parr$ as root (otherwise apply $\alpha\downarrow$ and $\sigma\downarrow$). There are the following cases:

1. $A\{\ \}$ and $B\{\ \}$ have both a quantifier as root. Then both must quantify the same variable (because of the same-depth-condition and the same-variable-condition), and at least one of them must be an $\exists$ (because of the one-$\exists$-condition and the one-$\forall$-condition). Assume, without loss of generality, that $A\{a^{\perp}\} = \forall b. A'\{a^{\perp}\}$ and $B\{a\} = \exists b. B'\{a\}$. Then by Lemma 7.7 we have that $P \overset{\nu}{\triangleright} \Gamma'$ with $\Gamma' = S\{\forall b.[A'\{a^{\perp}\} \parr B'\{a\}]\}$ is also correct. We can therefore apply the $\mathsf{u}\downarrow$-rule and proceed by induction hypothesis because $\delta\langle P^{\perp}, \Gamma'\rangle$ is strictly smaller than $\delta\langle P^{\perp}, \Gamma\rangle$. If $A$ and $B$ have both an $\exists$ as root, the situation is the same, except that we apply $\mathsf{v}\downarrow$-rule instead of $\mathsf{u}\downarrow$.

2. One of $A\{\ \}$ and $B\{\ \}$ has a quantifier as root and the other has a $\otimes$ as root. Without loss of generality, let $A\{\ \} = \forall b. A'\{\ \}$ and $B\{\ \} = (B'\{\ \} \otimes B'')$, i.e., $\Gamma = S[\forall b. A'\{a^{\perp}\} \parr (B'\{a\} \otimes B'')]$. Then by Lemma 7.5 we have that $P \overset{\nu}{\triangleright} \Gamma'$ with $\Gamma' = S([\forall b. A'\{a^{\perp}\} \parr B'\{a\}] \otimes B'')$ is also correct. We can therefore apply the $\mathsf{ls}$-rule and proceed by induction hypothesis because $\delta\langle P^{\perp}, \Gamma'\rangle$ is strictly smaller than $\delta\langle P^{\perp}, \Gamma\rangle$.

3. One of $A\{\ \}$ and $B\{\ \}$ has a quantifier as root and the other is just $\{\ \}$. This is impossible because it is a violation of the same-depth-condition.

4. $A\{\ \}$ and $B\{\ \}$ have both a $\otimes$ as root. Without loss of generality, assume that $\Gamma = S[(A'' \otimes A'\{a^{\perp}\}) \parr (B'\{a\} \otimes B'')]$. Then, we have by Lemma 7.6 that $P \overset{\nu}{\triangleright} \Gamma'$ is correct, with either $\Gamma' = S([(A'' \otimes A'\{a^{\perp}\}) \parr B'\{a\}] \otimes B'')$ or $\Gamma' = S(A'' \otimes [A'\{a^{\perp}\} \parr (B'\{a\} \otimes B'')])$. In one case we apply the $\mathsf{rs}$-rule, and in the other the $\mathsf{ls}$-rule. In both cases we have that $\delta\langle P^{\perp}, \Gamma'\rangle$ is strictly smaller than $\delta\langle P^{\perp}, \Gamma\rangle$. Therefore we can proceed by induction hypothesis.

5. One of $A\{\ \}$ and $B\{\ \}$ has a $\otimes$ as root and the other is just $\{\ \}$. Without loss of generality, $\Gamma = S[a^\perp \parr (B'\{a\} \otimes B'')]$. Then, by Lemma 7.8, we have that $P \overset{\nu}{\triangleright} \Gamma'$ with $\Gamma' = S([a^\perp \parr B'\{a\}] \otimes B'')$, is also correct. We can therefore apply the $\mathsf{ls}$-rule and proceed by induction hypothesis (as before $\delta\langle P^\perp, \Gamma'\rangle$ is strictly smaller than $\delta\langle P^\perp, \Gamma\rangle$).

6. If $A\{\ \}$ and $B\{\ \}$ are both just $\{\ \}$, i.e., $\Gamma = S[a^\perp \parr a]$, then do nothing and pick another pair of dual atoms.

We continue until we cannot proceed any further by applying these cases. This means, all pairs of dual atoms in $\mathscr{O}\Gamma$ are in a situation as in case 6 above. Now observe that a formula is the negation of a linking formula iff it is generated by the grammar

$$\mathscr{N} ::= 1 \mid [\mathscr{A}^\perp \parr \mathscr{A}] \mid [\perp \parr \mathscr{N}] \mid (\mathscr{N} \otimes \mathscr{N}) \mid \forall \mathscr{A}. \mathscr{N}$$

Consequently, the only thing that remains to do is to bring the all $\perp$ to the left-hand side of a $\parr$. This can be done in a similar fashion as we brought pairs $[a^\perp \parr a]$ together, by applying $\alpha\!\downarrow, \sigma\!\downarrow, \mathsf{ls}, \mathsf{rs}, \mathsf{u}\!\downarrow$. This makes $\Gamma$ the negation of a linking. (Because of well-nestedness, there can be no $\exists$-nodes left.) Let us call this linking formula $P'$. Now we have a proof graph $P \triangleright P'^\perp$. By Lemma 7.3 we have $P' \lesssim P$.

It remains to remove all instances of $\mathsf{v}\!\downarrow$, which is done exactly as in the proof of Theorem 3.4. ☐

# 8 Sequentialization

With the results in the previous sections, we can now directly translate between deep inference proofs and proof graphs. More precisely, we can translate a $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$ proof into a pre-proof graph by first decomposing it via Theorem 3.4 and then applying Lemmas 7.1, 7.4, and 5.4. Let us call a pre-proof graph *DI-sequentializable* if is obtained in this way from a $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$ proof.

**Theorem 8.1.** *Every DI-sequentializable pre-proof graph is a proof graph.*

*Proof.* Apply Theorem 3.4, and then Lemmas 7.1, 7.4, and 5.4. We get a pre-proof graph $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$ with $P^\perp = A$ and $\lceil \Gamma \blacktriangleleft \sigma \rceil = B$ and $\lfloor \Gamma \blacktriangleleft \sigma \rfloor = C$. ☐

By the method presented in [Str11], it is also possible to translate a $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$ directly into a proof graph without prior decomposition. However, the decomposition is the key for the translation from proof graphs back into $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$ proofs (i.e., "sequentialization" into the calculus of structures):

**Theorem 8.2.** *If $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$ is correct, then there is a $P' \lesssim P$, such that $P' \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$ is DI-sequentializable.*

*Proof.* Lemmas 7.1, 7.4, and 5.4 give us for a $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$ the derivation on the left below:

$$\begin{array}{c} 1 \\ \{\mathsf{ai}\!\downarrow, \perp\!\downarrow, 1\!\downarrow, \mathsf{e}\!\downarrow\} \Big\| \mathscr{D}_1' \\ P'^\perp \\ \{\alpha\!\downarrow, \sigma\!\downarrow, \mathsf{ls}, \mathsf{rs}, \mathsf{u}\!\downarrow\} \Big\| \mathscr{D}_2' \\ \lceil \Gamma \blacktriangleleft s \rceil \\ \{\mathsf{n}\!\downarrow, \mathsf{f}\!\downarrow\} \Big\| \mathscr{D}_3 \\ \lfloor \Gamma \blacktriangleleft s \rfloor \end{array} \qquad \begin{array}{c} 1 \\ \{\mathsf{ai}\!\downarrow, \perp\!\downarrow, 1\!\downarrow, \mathsf{e}\!\downarrow\} \Big\| \mathscr{D}_1 \\ P^\perp \\ \{\alpha\!\uparrow, \sigma\!\uparrow, \alpha\!\downarrow, \sigma\!\downarrow, \mathsf{ls}, \mathsf{rs}, \mathsf{u}\!\downarrow\} \Big\| \mathscr{D}_2 \\ \lceil \Gamma \blacktriangleleft s \rceil \\ \{\mathsf{n}\!\downarrow, \mathsf{f}\!\downarrow\} \Big\| \mathscr{D}_3 \\ \lfloor \Gamma \blacktriangleleft s \rfloor \end{array}$$

where $P' \lesssim P$. Note that together with Lemma 7.3, we also have derivation on the right above. ☐

Let us now look at the relation between sequent calculus proofs and proof graphs. The translation from $\mathsf{MLL2}_{\mathsf{Seq}}$ proofs into proof graphs is done inductively on the structure of the sequent proof as shown in Figure 8. For the rules id and 1, this is trivial ($\nu_0$ and $\nu_1$ are uniquely determined and the stretching is empty). In the rule $\perp$, the $\nu_\perp$ is obtained from $\nu$ by adding an edge between the new 1 and $\perp$. The exch and $\parr$-rules are also rather trivial ($P$, $\nu$, and $\sigma$ remain unchanged). For the $\otimes$ rule, the two linkings are connected by a new $\parr$-node, and the two principal formulas are connected by a $\otimes$ in the sequent forest. The same is done for the cut rule, where we use a special cut connective $\oplus$. The two interesting rules are the ones for $\forall$ and $\exists$. In the $\forall$-rule, to every root node of the proof graph for the premise a quantifier node is attached. This is what ensures the well-nestedness condition. It can be compared to Girard's putting a box around a proof net. The purpose of the $\exists$ can be interpreted as simulating the border of the box. The $\exists$-rule is the only one where the stretching $\sigma$ is changed. As shown in Figure 1, in the conclusion of that rule, the subformula $B$ of $A$ is replaced by the

$$\mathsf{id}\ \frac{}{a \otimes a^{\perp} \overset{\nu_0}{\triangleright} a^{\perp}, a \blacktriangleleft \emptyset} \qquad \perp\ \frac{P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma}{(1 \otimes P) \overset{\nu_\perp}{\triangleright} \Gamma, \perp \blacktriangleleft \sigma} \qquad 1\ \frac{}{\perp \overset{\nu_1}{\triangleright} 1 \blacktriangleleft \emptyset}$$

$$\mathsf{exch}\ \frac{P \overset{\nu}{\triangleright} \Gamma, A, B, \Delta \blacktriangleleft \sigma}{P \overset{\nu}{\triangleright} \Gamma, B, A, \Delta \blacktriangleleft \sigma} \qquad \parr\ \frac{P \overset{\nu}{\triangleright} A, B, \Gamma \blacktriangleleft \sigma}{P \overset{\nu}{\triangleright} [A \parr B], \Gamma \blacktriangleleft \sigma} \qquad \otimes\ \frac{P \overset{\nu}{\triangleright} \Gamma, A \blacktriangleleft \sigma \qquad Q \overset{\nu'}{\triangleright} B, \Delta \blacktriangleleft \tau}{[P \parr Q] \overset{\nu \cup \nu'}{\triangleright} \Gamma, (A \otimes B), \Delta \blacktriangleleft \sigma \cup \tau}$$

$$\forall\ \frac{P \overset{\nu}{\triangleright} A, B_1, \ldots, B_n \blacktriangleleft \sigma}{\exists a.P \overset{\nu}{\triangleright} \forall a.A, \exists a.B_1, \ldots, \exists a.B_n \blacktriangleleft \sigma} \qquad \exists\ \frac{P \overset{\nu}{\triangleright} \Gamma, A\langle a \backslash B \rangle \blacktriangleleft \sigma}{P \overset{\nu}{\triangleright} \Gamma, \boldsymbol{\exists} a.A\langle a \backslash B \rangle \blacktriangleleft \sigma'} \qquad \mathsf{cut}\ \frac{P \overset{\nu}{\triangleright} \Gamma, A \blacktriangleleft \sigma \qquad Q \overset{\nu'}{\triangleright} A^{\perp}, \Delta \blacktriangleleft \tau}{[P \parr Q] \overset{\nu \cup \nu'}{\triangleright} \Gamma, (A \oplus A^{\perp}), \Delta \blacktriangleleft \sigma \cup \tau}$$

Figure 8: Translating sequent calculus proofs into proof nets

quantified variable $a$. When translating this rule into proof graphs, we keep the $B$, but to every place where it has to be substituted we add a positive stretching edge from the new $\boldsymbol{\exists} a$. Similarly, whenever a $B^{\perp}$ should be replaced by $a^{\perp}$, we add a negative stretching edge. The new stretching is $\sigma'$.

We say that a pre-proof graph is *SC-sequentializable* if it can be obtained from a sequent proof as described above. If a pre-proof graph $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$ is obtained this way then the simple sequent $\lfloor \Gamma \blacktriangleleft \sigma \rfloor$ is exactly the conclusion of the sequent proof we started from.

**Theorem 8.3.** *Every SC-sequentializable pre-proof graph is a proof graph.*

*Proof.* The pre-proof graphs obtained from the rules id and 1 are obviously correct. Then it is an easy exercise to check that all other rules preserve correctness. □

The other direction is a bit more tricky, but there is no surprise. In fact, everything can already be found in [Gir87] and [LS06]. As explained in [LS06], we need, for reading back a sequent calculus proof from a proof graph, to consider linking formulas equivalent modulo associativity and commutativity of $\parr$. We write this as $P_1 \overset{\parr}{\sim} P_2$. Then, we have to remove all $\exists$-nodes from $\Gamma$ in order to get a sequentialization theorem because the translation shown in Figure 8 never introduces an $\exists$-node in $\Gamma$. For this we replace in $\Gamma$ every $\exists a.A$ with $\mathbb{E} a.\boldsymbol{\exists} a.A$ and by adding a stretching edge between the new $\boldsymbol{\exists} a$ and every $a$ and $a^{\perp}$ that was previously bound by $\exists a$ (i.e, is free in $A$). Let us write $\widehat{\Gamma \blacktriangleleft \sigma}$ for the result of this modification applied to $\Gamma \blacktriangleleft \sigma$.

**Theorem 8.4.** *If $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$ is correct, then there is a $P' \overset{\parr}{\sim} P$, such that $P' \overset{\nu}{\triangleright} \widehat{\Gamma \blacktriangleleft \sigma}$ is SC-sequentializable.*

*Proof.* We proceed by induction on the size of $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$, i.e., the number of nodes in the graph. In the base case where our proof graph is just $\perp \triangleright 1$ we have an instance of the 1-rule and we are done.

If there are any $\parr$-roots in $P$ or $\Gamma$, we simply remove them. If we remove a $\parr$-root in $\Gamma$, we have to apply the $\parr$-rule and can proceed by induction hypothesis. Note that by removing a $\parr$-root from $P$, the linking formula becomes a "linking sequent". This is the reason for the "modulo associativity and commutativity" in the statement of the theorem. If there is a $\boldsymbol{\exists}$-root in $\Gamma$ then we can simply remove this node, which corresponds to applying the $\exists$-rule because its conclusion is $\lfloor \widehat{\Gamma \blacktriangleleft \sigma} \rfloor$, and we proceed by induction hypothesis.

We are now in a situation where all roots of our proof graph are either $\forall$-, $\exists$-, $\mathbb{E}$-, or $\otimes$-nodes. (By our transformation above, all $\exists$-nodes are inside the linking.) Let us first consider the case in which there are no $\otimes$-roots. By well-nestedness and connectedness, all of them quantify the same variable, the linking $P$ consists of exactly one formula rooted by an $\exists$-node, and $\Gamma$ contains exactly one $\forall$-root, all other roots being $\mathbb{E}$-nodes. Therefore, we can apply the $\forall$-rule, remove all root-nodes, and proceed by induction hypothesis.

Let us now consider the case where $\otimes$-roots are present (but no $\parr$- nor $\boldsymbol{\exists}$-roots). By the splitting tensor lemma (Lemma 9.8, to be proved in the next section), we know that one of them must be splitting. This splitting tensor can either be inside the sequent $\Gamma$ or inside the linking $P$. If it is inside $\Gamma$, we can immediately apply the $\otimes$-rule and proceed by induction hypothesis. If the splitting tensor is inside $P$, then there are two possibilities: either both children are dual atoms, or one child is a 1. Both cases handled exactly as in [LS06]. □

An interesting side effect of this theorem is that we can now translate cut-free $\mathsf{MLL2}_{\mathsf{DI}}$ proofs into cut-free $\mathsf{MLL2}_{\mathsf{Seq}}$ proofs without using the detour of cut elimination, as done in the proof of Theorem 3.1.

In this section we have shown the "sequentialization" of proof graphs into the calculus of structures and into the sequent calculus. There is an important difference between the two sequentializations. While for the sequent calculus we have a monolithic procedure reducing the proof graph node by node, we have for the calculus of structures a modular procedure that treats the different parts of the proof graph (which correspond to the three different aspects of the logic) separately. The core is Lemma 7.4 which deals with the purely multiplicative part. Then comes Lemma 5.4 which only deals with instantiation and substitution, i.e, the second-order aspect. Finally, Lemma 7.1 takes care of the linking, whose task is to describe the role of the units in the proof. This modularity in the sequentialization is possible because of the decomposition in Theorem 3.4, which can be seen as a version of Herbrand's theorem. Because of this modularity we treated the units via the linking formulas [SL04, LS06] instead of a linking function as done by Hughes in [Hug05b, Hug05a].

# 9 The splitting tensor lemma

For proving our sequentialization (into the sequent calculus) we need the so-called "splitting tensor lemma", which is a well-known fact for the purely multiplicative case [Gir87]. Unfortunately, due to the presence of the quantifiers and the units, the proof of the splitting tensor lemma is slightly more complicated than in the purely multiplicative case. This means, for the sake of completeness, we have to prove it here again. We follow closely the presentation in [BvdW95]. We need the concept of a *weak* (pre-)proof graph $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$ which is a (pre-)proof graph in which the linking $P$ does not have to be a formula but can be a sequent, i.e., some of the root-$\parr$s are removed.

**Definition 9.1.** Let $\pi_1$ and $\pi_2$ be weak pre-proof graphs. We say $\pi_1$ is a *subpregraph* of $\pi_2$, written as $\pi_1 \subseteq \pi_2$ if all nodes appearing in $\pi_1$ are also present in $\pi_2$. We say $\pi_1$ is a *subgraph* of $\pi_2$ if $\pi_1 \subseteq \pi_2$, and $\pi_1$ and $\pi_2$ are both multiplicatively correct (i.e, for the time being we ignore well-nestedness and appropriateness). A *door* of $\pi_1$ is any root node (in $P$ or in $\Gamma$) of $\pi_1$.

**Lemma 9.2.** *Let $\pi'$ and $\pi''$ be subgraphs of some weak proof graph $\pi$.*

(i) *The subpregraph $\pi' \cup \pi''$ is a subgraph of $\pi$ if and only if $\pi' \cap \pi'' \neq \emptyset$.*

(ii) *If $\pi' \cap \pi'' \neq \emptyset$ then $\pi' \cap \pi''$ is a subgraph of $\pi$.*

*Proof.* Intersection and union in the statement of that lemma have to be understood in the canonical sense: An edge/node/link appears in in $\pi' \cap \pi''$ (resp. $\pi' \cup \pi''$) if it appears in both, $\pi'$ and $\pi''$ (resp. in at least one of $\pi'$ or $\pi''$). For proving the lemma, let us first note that because in $\pi$ every switching is acyclic, also in every subpregraph of $\pi$ every switching is acyclic, in particular also in $\pi' \cup \pi''$ and $\pi' \cap \pi''$. Therefore, we need only to consider the connectedness condition.

(i) If $\pi' \cap \pi'' = \emptyset$ then every switching of $\pi' \cup \pi''$ must be disconnected. Conversely, if $\pi' \cap \pi'' \neq \emptyset$, then every switching of $\pi' \cup \pi''$ must be connected (in every switching of $\pi' \cup \pi''$ every node in $\pi' \cap \pi''$ must be connected to every node in $\pi'$ and to every node in $\pi''$, because $\pi'$ and $\pi''$ are both multiplicatively correct).

(ii) Let $\pi' \cap \pi'' \neq \emptyset$ and let $s$ be a switching for $\pi' \cup \pi''$. Then $s$ is connected and acyclic by (i). Let $s'_\pi$, $s''_\pi$, and $s_{\pi' \cap \pi''}$, be the restrictions of $s$ to $\pi'$, $\pi''$, and $\pi' \cap \pi''$, respectively. Now let $A$ and $B$ be two nodes in $\pi' \cap \pi''$. Then $A$ and $B$ are connected by a path in $s'_\pi$ because $\pi'$ is correct, and by a path in $s''_\pi$ because $\pi''$ is correct. Since $s$ is acyclic, the two paths must be the same and therefore be contained in $s_{\pi' \cap \pi''}$.

□

**Lemma 9.3.** *Let $\pi$ be a weak proof graph, and let $A$ be a node appearing in $\pi$. Then there is a subgraph $\pi'$ of $\pi$, that has $A$ as a door.*

*Proof.* For proving this lemma, we need the following notation. Let $\pi$ be a proof graph, let $A$ be some node in $\pi$, and let $s$ be a switching for $\pi$. Then we write $s(\pi, A)$ for the graph obtained as follows:

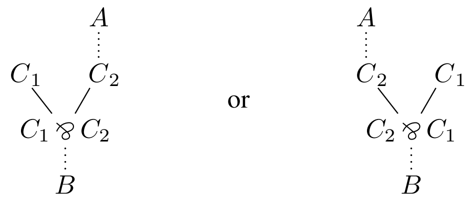
- If $A$ is a child of a binary node $B$ in $\pi$, and there is an edge from $B$ to $A$ in $s$, then remove that edge and let $s(\pi, A)$ be the connected component of (the remainder of) $s$ that contains $A$.
- Otherwise let $s(\pi, A)$ be just $s$.

Now let

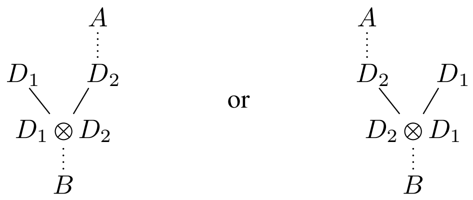
$$\pi' = \bigcap_s s(\pi, A)$$

where $s$ ranges over all possible switchings of $\pi$. (Note that it could happen that formally $\pi'$ is not a subpregraph because some edges in the formula trees might be missing. We graciously add these missing edges to $\pi'$ such that it becomes a subpregraph.) Clearly, $A$ is in $\pi'$.

We are now going to show that $A$ is a door of $\pi'$. By way of contradiction, assume it is not. This means there is ancestor $B$ of $A$ that is in $\bigcap_s s(\pi, A)$. Now choose a switching $\hat{s}$ such that whenever there is a $\parr$ node between $A$ and $B$, i.e.,



then $\hat{s}$ chooses $C_2$ (i.e., removes the edge between $C_1$ and its parent).[4] Then there must be a $\otimes$ between $A$ and $B$:



Otherwise $B$ would not be in $\pi'$ (because we remove every edge from $A$ to its parent). Now suppose we have chosen the uppermost such $\otimes$. Then the path connecting $A$ and $D_1$ in $\hat{s}(\pi, A)$ cannot pass through $D_2$ (by the construction of $\hat{s}(\pi, A)$). But this means that in $\hat{s}$ (where the edge between $A$ and its parent is not removed) there are two distinct paths connecting $A$ and $D_1$, which contradicts the acyclicity of $\hat{s}$.

Now we have to show that $\pi'$ is a subgraph. Let $s$ be a switching for $\pi'$. Since $\pi'$ is a subpregraph of $\pi$, we have that $s$ is acyclic. Now let $\tilde{s}$ be an extension of $s$ to $\pi$. Then $s$ is the restriction of $\tilde{s}(\pi, A)$ to $\pi'$, and hence connected. $\square$

**Definition 9.4.** Let $\pi$ be a weak proof graph, and let $A$ be a node in $\pi$. The *kingdom of $A$ in $\pi$*, denoted by $kA$, is the smallest subgraph of $\pi$, that has $A$ as a door. Similarly, the *empire of $A$ in $\pi$*, denoted by $eA$, is the largest subgraph of $\pi$, that has $A$ as a door. We define $A \ll B$ iff $A \in kB$, where $A$ and $B$ can be any nodes in $\pi$.

An immediate consequence of Lemmas 9.2 and 9.3 is that kingdom and empire always exist.

**Remark 9.5.** The subgraph $\pi'$ constructed in the proof of Lemma 9.3 is in fact the empire of $A$. But we will not need this fact later and will not prove it here.

**Lemma 9.6.** *Let $\pi$ be a weak proof graph, and let $A$, $A'$, $B$, and $B'$ be nodes in $\pi$, such that $A$ and $B$ are distinct, $A'$ is a child of $A$, and $B'$ is a child of $B$. Now suppose that $B' \in eA'$. Then we have that $B \notin eA'$ if and only if $A \in kB$.*

*Proof.* We have $B' \in eA' \cap kB$. Hence, $\pi' = eA' \cap kB$ and $\pi'' = eA' \cup kB$ are subnets of $\pi$. By way of contradiction, let $B \notin eA'$ and $A \notin kB$. Then $\pi''$ has $A'$ as door and is larger than $eA'$ because it contains $B$. This contradicts the definition of $eA'$. On the other hand, if $B \in eA'$ and $A \in kB$ then $\pi'$ has $B$ as door and is smaller than $kB$ because it does not contain $A$. This contradicts the definition of $kB$. $\square$

**Lemma 9.7.** *Let $A$ and $B$ be nodes in a weak proof graph $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$. If $A \ll B$ and $B \ll A$, then either $A$ and $B$ are the same node or they are dual leaf-nodes connected via an edge in $\nu$.*

*Proof.* If $a$ and $a^\perp$ are two dual leaf-nodes connected via $\nu$, then clearly $ka = ka^\perp$. Now let $A$ and $B$ be two distinct non-leaf nodes with $A \in kB$ and $B \in kA$. Then $kA \cap kB$ is a subgraph and hence $kA = kA \cap kB = kB$. We have three cases:

- If $A$ is a quantifier node, then the result of removing $A$ from $kB$ is still a subgraph, contradicting the minimality of $kB$.
- If $A = A' \parr A''$ then the result of removing $A$ from $kB$ is still a subgraph, contradicting the minimality of $kB$.
- If $A = A' \otimes A''$ then $kA = kA' \cup kA'' \cup \{A' \otimes A''\}$. Hence $B \in kA'$ or $B \in kA''$. This contradicts Lemma 9.6, which says that $B \notin eA'$ and $B \notin eA''$.

[4] Note that there is a mistake in [BvdW95].

□

From Lemma 9.7 it immediately follows that $\ll$ is a partial order on the nodes of a weak proof graph $\pi$. We make crucial use of this fact in in the proof of the splitting tensor lemma:

**Lemma 9.8.** *Let $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$ be a weak proof graph in which no root (in $P$ or $\Gamma$) is an $\parr$- or $\boldsymbol{\exists}$-node. If there are $\otimes$-roots in $P$ or $\Gamma$, then at least one of them is splitting, i.e., by removing that $\otimes$, the graph becomes disconnected.*

*Proof.* Choose among the $\otimes$-roots of $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$ one which is maximal w.r.t. $\ll$. Without loss of generality, assume it is $A_i = A'_i \otimes A''_i$. We will now show that it is splitting, i.e., $\pi = \{A'_i \otimes A''_i\} \cup eA'_i \cup eA''_i$. By way of contradiction, assume $A'_i \otimes A''_i$ is not splitting. This means we have somewhere in $\pi$ a node $B$ with two children $B'$ and $B''$ such that $B' \in eA'_i$ and $B'' \in eA''_i$, and therefore $B \notin eA'_i$ and $B \notin eA''_i$. We also know that $A_j \preccurlyeq B$ for some other root node $A_j$. We have now two cases to consider

- If $A_j$ is a $\otimes$-node, say $A_j = A'_j \otimes A''_j$, then $B \in kA_j$ and therefore $kB \subseteq kA_j$. But by Lemma 9.6 we have $A_i \in kB$ and therefore $A_i \in kA_j$, which contradicts the maximality of $A_i$ w.r.t. $\ll$.
- Otherwise $A_j$ is a $\forall$-, $\exists$-, or $\boldsymbol{\exists}$-node. Then $B$ is inside a box which has $A_j$ as a door. Since $eA'_i$ and $eA''_i$ are both multiplicatively correct, we have a switching $s$ with two paths, $A'_i \overset{s}{\frown} B'$ and $A''_i \overset{s}{\frown} B''$. Both paths must enter the box at some point. This can happen only through a door. And because of the acyclicity condition the two paths must come in through two different doors. At most one of them can be in the linking $P$, because otherwise the one-$\exists$-condition (6.3-3) would be violated. But if one of the doors is in $P$ and the other in $\Gamma$, we have immediately a violation of the acyclicity condition. (For every box we can always construct a switching with a direct path from the $\exists$-door in $P$ to any chosen door in $\Gamma$. Hence both doors must be inside $\Gamma$. But this violates the no-down-path condition (6.3-5), because there is a down path between the two doors going through $A'_i \otimes A''_i$. Contradiction.

□

# 10 Cut elimination

In proof graphs, the cut is represented by a special connective $\oplus$, such that whenever we have $A \oplus B$ in $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$, then we must have $\lfloor A \blacktriangleleft \sigma \rfloor = \lfloor B \blacktriangleleft \sigma \rfloor^{\perp}$.[5] Morally, a $\oplus$ may occur only at the root of a formula in $\Gamma$. However, due to well-nestedness we must allow cuts to have $\boldsymbol{\exists}$-nodes as ancestors. Then the $\oplus$ is treated in the correctness criterion in exactly the same way as the $\otimes$, and sequentialization does also hold for proof graphs with cut.

As already discussed in [SL04, LS06], we need to work with an equivalence relation on proof graphs, because of the presence of the multiplicative units. This is a consequence of the PSPACE-completeness of proof equivalence in MLL [HH14].

**Definition 10.1.** Let $\sim$ be the smallest equivalence relation on the set of proof graphs satisfying

$$\begin{aligned}
P[Q \parr R] \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma &\sim P[R \parr Q] \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma \\
P[[Q \parr R] \parr S] \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma &\sim P[Q \parr [R \parr S]] \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma \\
P(1 \otimes (1 \otimes Q)) \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma &\sim P(1 \otimes (1 \otimes Q)) \overset{\nu'}{\triangleright} \Gamma \blacktriangleleft \sigma \\
P(1 \otimes [Q \parr R]) \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma &\sim P[(1 \otimes Q) \parr R] \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma \\
P(1 \otimes \exists a.Q) \overset{\nu}{\triangleright} \Gamma\{\perp\} \blacktriangleleft \sigma &\sim P\{\exists a.(1 \otimes Q)\} \overset{\nu}{\triangleright} \Gamma\{\boldsymbol{\exists} a.\perp\} \blacktriangleleft \sigma
\end{aligned}$$

where in the third line $\nu'$ is obtained from $\nu$ by exchanging the pre-images of the two 1s. In all other equations the bijection $\nu$ does not change. In the last line $\nu$ must match the 1 and $\perp$. A *proof net* is an equivalence class of $\sim$.

The first two equations in Definition 10.1 are simply associativity and commutativity of $\parr$ inside the linking. The third is a version of associativity of $\otimes$. The fourth equation could destroy multiplicative correctness, but since we defined $\sim$ only on proof graphs we do not need to worry about that.[6] The last equation says that a $\perp$ can freely tunnel through the borders of a box. Let us emphasize that this quotienting via an equivalence is due to the multiplicative units. If one wishes to use a system without units, one could completely dispose the equivalence by using $n$-ary $\parr$s in the linking.

The cut reduction relation $\rightsquigarrow$ is defined on pre-proof graphs as shown in Figure 9. The reductions not involving quantifiers are exactly as shown in [LS06]. If we encounter a cut between two binary connectives, then we replace

[5] Note that it does not mean $A = B^{\perp}$, because $\Gamma$ is expanded.

[6] In [SL04, LS06] the relation $\sim$ is defined on pre-proof graphs, and therefore a side condition had to be given to that equation (see also [Hug05a]).

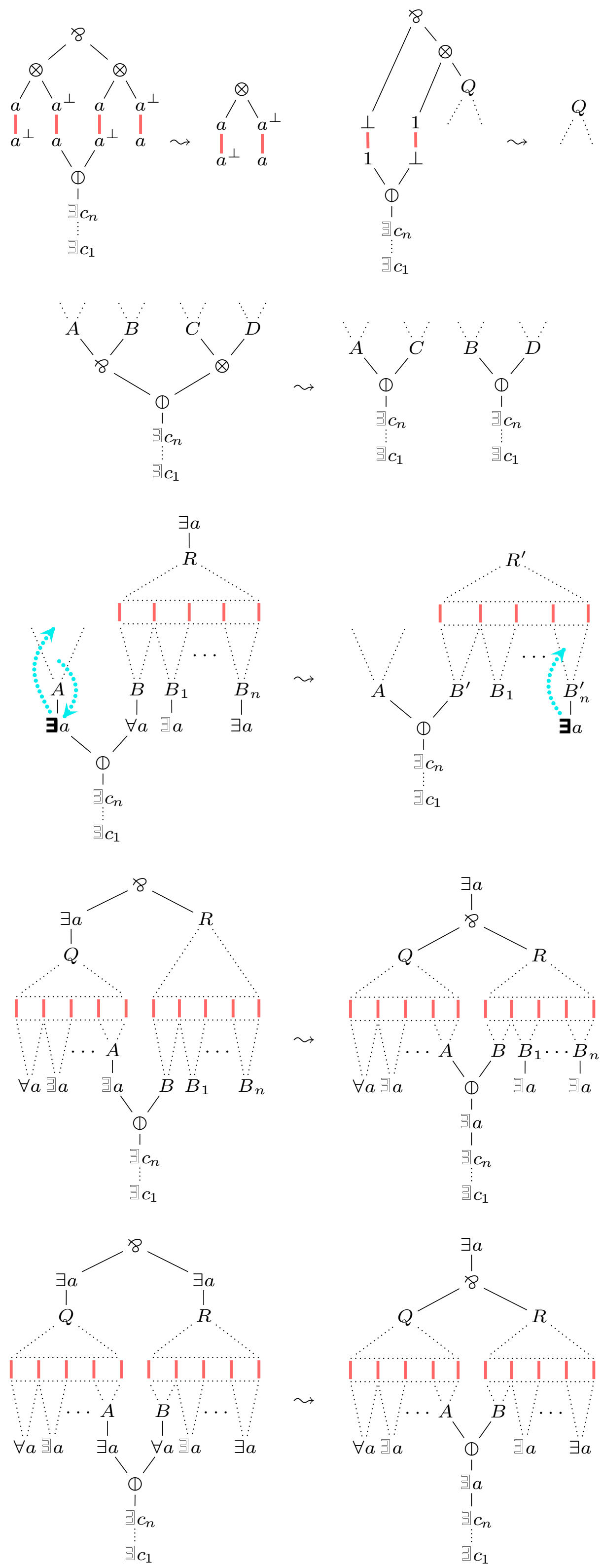


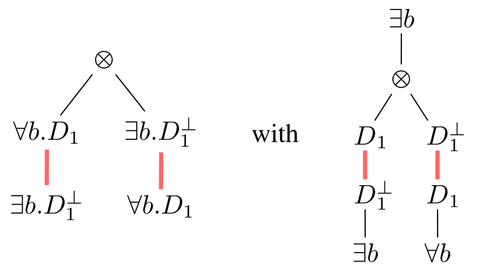
Figure 9: Cut reduction for MLL2 proof nets

$[A \parr B] \obar (C \otimes D)$ by two smaller cuts $A \obar C$ and $B \obar D$. Note that if $\lfloor [A \parr B] \blacktriangleleft \sigma \rfloor = \lfloor (C \otimes D) \blacktriangleleft \sigma \rfloor^{\perp}$ then $\lfloor A \blacktriangleleft \sigma \rfloor = \lfloor C \blacktriangleleft \sigma \rfloor^{\perp}$ and $\lfloor B \blacktriangleleft \sigma \rfloor = \lfloor D \blacktriangleleft \sigma \rfloor^{\perp}$. If we have an atomic cut $a^{\perp} \obar a$, then we must have in $P$ two "axiom links" $(a^{\perp} \otimes a)$, which are by the leaf mapping $\nu$ attached to the two atoms in the cut. It was shown in [LS06] that the two pairs $(a^{\perp} \otimes a)$

can, under the equivalence relation in Definition 10.1, be brought next to each other such that $P$ has $[(a \otimes a^\perp) \parr (a \otimes a^\perp)]$ as subformula. We can replace this by a single $(a^\perp \otimes a)$ and remove the cut. If we encounter a cut $1 \obar \perp$ on the units, we must have in the linking a corresponding $\perp$ and a subformula $(1 \otimes Q)$, which can (for the same reasons as for the atomic cut) be brought together, such that we have in $P$ a subformula $[\perp \parr (1 \otimes Q)]$. We replace this by $Q$ and remove the cut.

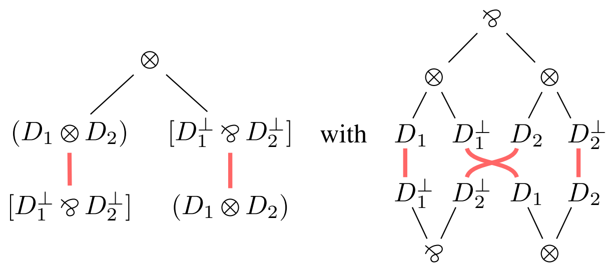
Let us now consider the cuts that involve the quantifiers. There are three cases, one for each of $\boldsymbol{\exists}$, $\exists\!\!\!\exists$, and $\exists$. The first two correspond to the ones in [Gir87]. The third one does not appear in [Gir87] because there is never a $\exists$-node created when a sequent calculus proof is translated into a proof net.

If one of the cut formulas is an $\boldsymbol{\exists}$-node, then the other must be an $\forall$, which quantifies the same variable, say we have $\boldsymbol{\exists} a.A \obar \forall a.B$. Then we pick a stretching edge starting from $\boldsymbol{\exists} a.A$. Let $C$ be the node where it ends and let $D = \lfloor C \blacktriangleleft \sigma \rfloor$. Note that by Condition 5.3-1, $D$ is independent from the choice of the edge in case there are many of them. (If there are only negative edges, then let $D = \lfloor C \blacktriangleleft \sigma \rfloor^\perp$. If there are no stretching edges at all, then let $D = a$. Now we can inside the box of $\forall a.B$ substitute $a$ everywhere by $D$. Then we remove all the doors of the $\forall a.B$-box and replace the cut by $A \obar B$. There are two subtleties involved in this case. First, "removing a door" means for a $\exists\!\!\!\exists$ that the node is removed, but for and $\exists$, it means that the node is replaced by an $\boldsymbol{\exists}$ and a stretching edge is added for every $a$ and $a^\perp$ bound by the $\exists$-node to be removed. Second, by substituting $a$ with $D$ we get "axiom links" which are not atomic anymore, but it is straightforward to make them atomic again: one proceeds by structural induction on $D$. If $D = \forall b.D_1$, then replace



(35)

and if $D = (D_1 \otimes D_2)$ then replace



(36)

The cases for $\exists a.D_1$ and $[D_1 \parr D_2]$ are similar.

If one of the two cut formulas is a $\exists\!\!\!\exists$-node, then the other one can be anything. Say, we have $\exists\!\!\!\exists a.A \obar B$. Let $eB$ be the *empire* of $B$, i.e, largest sub-proof graph of $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$ that has $B$ as a conclusion. Let $B_1, \ldots, B_n$ be the other doors of $eB$ inside $\Gamma$, and let $R$ be the door of $eB$ in $P$. If $eB$ has more than one root-node inside the linking $P$, then we can rearrange the $\parr$-nodes in $P$ via the equivalence in 10.1 such that $eB$ has a single $\parr$-root in $P$. Furthermore, as in the case of the atomic cut we can use the equivalence in 10.1 to get in $P$ a subformula $[\exists a.Q \parr R]$ where $\exists a.Q$ is the partner of $\exists\!\!\!\exists a.A$. Now we replace on $P$ the formula $[\exists a.Q \parr R]$ by $\exists a.[Q \parr R]$ and in $\Gamma$ the formulas $B_1, \ldots, B_n$ by $\exists\!\!\!\exists a.B_1, \ldots, \exists\!\!\!\exists a.B_n$. Put in plain words, we have pulled the whole empire of $B$ inside the box of $\exists\!\!\!\exists a.A$. But now we have a little problem: Morally, we should replace the cut $\exists\!\!\!\exists a.A \obar B$ by $A \obar B$; the cut is also pulled inside the box. But by this we would break our correctness criterion, namely, the same-depth-condition 6.3-1. To solve this problem, we allow cut-nodes to have $\exists\!\!\!\exists$-nodes as ancestors, and we replace the cut $\exists\!\!\!\exists a.A \obar B$ by $\exists\!\!\!\exists a.(A \obar B)$. Note that this does not cause problems for the other cut reduction steps because we can just keep all $\exists\!\!\!\exists$-ancestors when we replace a cut by a smaller one.

Finally, there is the cut between an ordinary $\exists$-node and a $\forall$-node, say $\exists a.A \obar \forall a.B$. Then we do not pull the whole empire of $\forall a.B$ inside the box of $\exists a.A$ but only the $\forall a.B$-box. This is the same as merging the two boxes into one. Formally, let $\exists a.Q$ and $\exists a.R$ be the partners of $\exists a.A$ and $\forall a.B$, respectively. Again, for the same reasons as in the case of the atomic cut, we can assume that we have the configuration $[\exists a.Q \parr \exists a.R]$ in $P$, which we replace by $\exists a.[Q \parr R]$. The cut is replaced by $\exists\!\!\!\exists a.(A \obar B)$.

This cut reduction relation is defined *a priori* only on pre-proof graphs. For a pre-proof graph $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$ and a cut $A \obar B$ in $\Gamma$, we say the cut is *ready*, if the cut can immediately be reduced without further modification of $P$. We now can show the following:

**Theorem 10.2.** *The cut reduction relation preserves correctness and is well-defined on proof nets.*

*Proof.* That correctness is preserved follows immediately from inspecting the six cases. To show that cut reduction is well-defined on proof nets we need to verify the following two facts:

1. Whenever the same cut is reduced in two different representations of the same proof net, then the two results also represent the same proof net.

2. Whenever there is a cut in a proof net, then this cut can be reduced, i.e., there is a representation to which the corresponding reduction step in Figure 9 can be applied.

For the first statement, it suffices to observe that whenever one of the basic equivalence steps in Definition 10.1 can be performed in the non-reduced net, then the same step can be performed in the reduced net or is vacuous in the reduced net. For the second statement we have to make a case analysis on the type of cut: If the cut is $[A \parr B] \obar (C \otimes D)$ or $\boldsymbol{\exists} a.A \obar \forall a.B$, then it is trivial because these cuts are always ready. Let us now consider a cut $\exists a.A \obar \forall a.B$. Clearly, the two boxes of which $\exists a$ and $\forall a$ are doors each have a single door $\exists a$ in $P$, and their first common ancestor is a $\parr$ (because of the acyclicity condition). Therefore, the linking is of the shape $P[S_1\{\exists a.Q\} \parr S_2\{\exists a.R\}]$ for some contexts $S_1\{\ \}$ and $S_2\{\ \}$. Now we proceed by induction on the size of $S_1\{\ \}$ and $S_2\{\ \}$ and make a case analysis on their root-nodes:[7]

- Both contexts are empty. In this case the linking has the desired shape, and we are done.
- One of them has a $\parr$-root. In this case we apply associativity of the $\parr$ and proceed by induction hypothesis.
- One of them has an $\exists$-node as root. This is impossible because it would violate the well-nested condition.
- One of them has a $\otimes$-root, and the other context is empty. Without loss of generality, the linking is of the shape $P[(1 \otimes S_1'\{\exists a.Q\}) \parr \exists a.R]$. We claim, that in this case the correctness is preserved if we replace the linking by $P(1 \otimes [S_1'\{\exists a.Q\} \parr \exists a.R])$. We leave the proof of this claim to the reader because it is very similar to the proof of Lemma 7.8. Hence, we can proceed by induction hypothesis.
- Both contexts have a $\otimes$-root. Then the linking is of the shape

  $$P[(1 \otimes S_1'\{\exists a.Q\}) \parr (1 \otimes S_2'\{\exists a.R\})] \quad .$$

  Now we claim that we can replace this linking with one of

  $$P(1 \otimes [S_1'\{\exists a.Q\} \parr (1 \otimes S_2'\{\exists a.R\})])$$

  and

  $$P(1 \otimes [(1 \otimes S_1'\{\exists a.Q\}) \parr S_2'\{\exists a.R\}])$$

  without destroying correctness. Again, we leave the proof to the reader because it is almost the same as the proof of Lemma 7.6. As before, we can proceed by induction hypothesis.

For a cut $\exists\!\!\!\exists a.A \obar B$ we proceed similarly. The only difference is that we first have to apply associativity and commutativity of $\parr$ to bring the proof graph in a form where the empire $eB$ has a single root $R$ in the linking. For cuts $a \obar a^{\perp}$ and $1 \obar \perp$ we can also proceed similarly. □

The main results of this section is now:

**Theorem 10.3.** *The cut reduction relation $\rightsquigarrow$ is terminating and confluent.*

*Proof.* Termination has already been shown in [Gir87], and we will not repeat it here. For showing confluence it suffices to show local confluence. We will do this first for proof graphs. Suppose we have two cuts which are ready in a given proof graph. We claim that the result of reducing them is independent from the order of the reduction. There is only one critical pair, since the only possibility for overlapping redexes is when one cut is $\boldsymbol{\exists} a.A \obar \forall a.B$ and the other is $\exists a.C \obar \forall a.D$ and the formulas $\forall a.B$ and $\exists a.C$ are doors of the same box. If we reduce first the cut $\boldsymbol{\exists} a.A \obar \forall a.B$, then we do first the substitution in the $\forall a.B$-box, remove its border, change the second cut to $\boldsymbol{\exists} a.C' \obar \forall a.D$, and then do the same substitution in the $\forall a.D$-box and remove its border. If we reduce first the cut $\exists a.C \obar \forall a.D$, then we merge the two boxes into one, and then do the substitution and remove the border of the box. Clearly, the result is the same in both cases. Hence, we have local confluence for the cut reduction on proof graphs. In the case of proof nets, it can happen that the two cuts are ready in two different representatives. With the method shown in the previous proof we can try to construct a representatives in which both cuts are ready. There are only two cases in which this fails. The first is when we have two atomic cuts using the same "axiom link". But then the result of reducing the two is a single axiom link, independent from the order. The second case is when we have two cuts $\exists a.A \obar \forall a.B$ and $\exists a.C \obar \forall a.D$ where $\forall a.B$ and $\exists a.C$ are doors of the same box. Here the result of reducing the two will be a big box which is the merge of all three boxes, independent of the order in which the two cuts are reduced. □

[7] Note the similarity to the proof of Lemma 7.4.

## 11 Some observations on the units

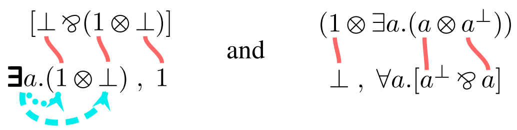
An important consequence of the last theorem is that we have a category of proof nets: the objects are (simple) formulas and a map $A \to B$ is a proof net with conclusion $\vdash A^\perp, B$. The composition of maps is defined by cut elimination. Unfortunately, we do not know much about this category, apart from the fact that it is *-autonomous [LS06]. But there are some observations that we can make about the units, which can be expressed with the second-order quantifiers:

$$1 \equiv \forall a.[a^\perp \parr a] \qquad \text{and} \qquad \perp \equiv \exists a.(a \otimes a^\perp) \tag{37}$$

An interesting question to ask is whether these logical equivalences should be isomorphisms in the categorification of the logic. In the category of coherent spaces [Gir87] they are, but in our category of proof nets they are not. This can be shown as follows. The two canonical maps $\forall a.[a^\perp \parr a] \to 1$ and $1 \to \forall a.[a^\perp \parr a]$ are given in the sequent calculus by:

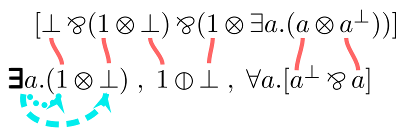
$$\exists\,\dfrac{\otimes\,\dfrac{1\,\dfrac{}{\vdash 1}\qquad \perp\,\dfrac{1\,\dfrac{}{\vdash 1}}{\vdash \perp, 1}}{\vdash (1 \otimes \perp), 1}}{\vdash \exists a.(a \otimes a^\perp), 1} \qquad \text{and} \qquad \mathsf{weak}\,\dfrac{\forall\,\dfrac{\parr\,\dfrac{\mathsf{id}\,\dfrac{}{\vdash a^\perp, a}}{\vdash [a^\perp \parr a]}}{\vdash \forall a.[a^\perp \parr a]}}{\vdash \perp, \forall a.[a^\perp \parr a]} \tag{38}$$

As proof nets they are given as follows:



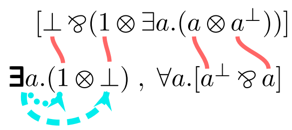
(39)

respectively. Composing them means eliminating the cut from



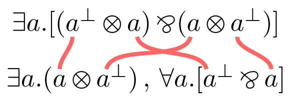
(40)

This yields



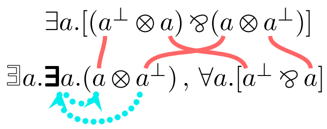
(41)

If the two maps in (39) where isos, the result (41) must be the same as the identity map $\forall a.[a^\perp \parr a] \to \forall a.[a^\perp \parr a]$ which is represented by the proof net



(42)

This is obviously not the case. Even if we replaced $\exists a$ by $\exists a.\boldsymbol{\exists} a$ as for Theorem 8.4:



(43)

we would not have an equality. Translating this back into the sequent calculus, we would get

$$\forall\,\dfrac{\exists\,\dfrac{\otimes\,\dfrac{1\,\dfrac{}{\vdash 1}\qquad \perp\,\dfrac{\parr\,\dfrac{\mathsf{id}\,\dfrac{}{\vdash a^\perp, a}}{\vdash [a^\perp \parr a]}}{\vdash \perp, [a^\perp \parr a]}}{\vdash (1 \otimes \perp), [a^\perp \parr a]}}{\vdash \exists a.(a \otimes a^\perp), [a^\perp \parr a]}}{\vdash \exists a.(a \otimes a^\perp), \forall a.[a^\perp \parr a]} \tag{44}$$

for the proof net in (41) and

$$\forall\,\dfrac{\exists\,\dfrac{\parr\,\dfrac{\otimes\,\dfrac{\mathsf{id}\,\dfrac{}{\vdash a, a^{\perp}} \qquad \mathsf{id}\,\dfrac{}{\vdash a^{\perp}, a}}{\vdash (a \otimes a^{\perp}), a^{\perp}, a}}{\vdash (a \otimes a^{\perp}), [a^{\perp} \parr a]}}{\vdash \exists a.(a \otimes a^{\perp}), [a^{\perp} \parr a]}}{\vdash \exists a.(a \otimes a^{\perp}), \forall a.[a^{\perp} \parr a]} \tag{45}$$

for the proof nets in (42) and (43).

A similar situation occurs with the additive units $0$ and $\top$. They can be expressed with second-order quantifiers as follows:

$$0 \equiv \forall a.a \qquad \text{and} \qquad \top \equiv \exists a.a \tag{46}$$

Since we do not have $0$ and $\top$ in the language, we cannot check whether we have these isos in our category. However, since $0$ and $\top$ are commonly understood as initial and terminal objects of the category of proofs, we could ask whether $\forall a.a$ and $\exists a.a$ have this property: We clearly have a canonical proof for $\forall a.a \to A$ for every formula $A$ (simply instantiate $a$ with $A$), but it is not unique for all $A$. for example, we could prove the sequent $\vdash \exists a.a^{\perp}, (c \otimes [b \parr b^{\perp}])$ by substituting $a$ with $c$. Nonetheless, one could imagine an isomorphism $0 \cong \forall a.a$ in a version of our proof nets which is extended with additives and exponentials. However, in this case $0$ would not be initial.

## 12 Comparison to Girard's proof nets for MLL2

Such a comparison can only make sense for $\mathsf{MLL2}^{-}$, i.e., the logic without the units $1$ and $\perp$. In [Gir90] the units are not considered, and in [Gir87] the units are treated in a way that is completely different from the one suggested here. Consequently, in this section we consider only proof nets without any occurrences of $1$ and $\perp$. For simplicity, we will allow $n$-ary $\parr$s in the linkings, so that we can discard the equivalence relation of Definition 10.1 and identify proof graphs and proof nets.

Now the translation from our proof nets to Girard's boxed proof nets of [Gir87] is immediate: Let $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$ be a given proof net. The translation to a boxed proof net is done in three steps:

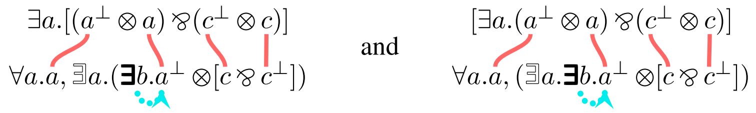
1. For each $\exists$ in $P$ draw a box around the sub-proof net which has as doors this $\exists$ and its partners in $\Gamma$.
2. Replace in $\Gamma$ every node $A$ that is not a *gexal* by its floor $\lfloor A \blacktriangleleft \sigma \rfloor$, and remove all stretching edges and all $\exists a.$-nodes.
3. Finally, remove all $\exists a.$- and all $\parr$-nodes in $P$, and replace the $\otimes$-nodes in $P$ by axiom links.

For the converse translation we proceed in the opposite order:

1. Replace each axiom link in the boxed proof net by a $\otimes$-linking-node.
2. Replace each other link in the boxed proof net by the root connective of its conclusion formula. If it is an $\exists$-link, replace it by a $\boldsymbol{\exists}$-node and add the stretching edges as discussed in section 8.
3. Finally, for each $\forall$-box add an $\exists a.$-node in the linking $P$ and an $\exists a.$-node at each door of the box in $\Gamma$ (except the $\forall$-door), where $a$ is the variable quantified by the $\forall$-door.

It is clear that in both directions correctness is preserved, i.e., the two criteria (the one of [Gir87] and ours) are equivalent. Both data structures contain the same information. However, Girard's boxed proof nets depend on the deductive structure of the sequent calculus. A box stands for the global view that the $\forall$-rule has in the sequent calculus, and an $\exists$-link is attached to its full premise and conclusion that are subject to the same side conditions as in the sequent calculus. The new proof nets presented in this paper make these side conditions explicit in the data structure, which is the reason why our definitions are a bit longer than Girard's. However, our proof nets are entirely independent from any deductive system.

The proof nets of [Gir90] are obtained from the box proof nets by simply removing the boxes. In our setting this is equivalent to removing all $\exists$-nodes in $P$ and all $\exists$-nodes in $\Gamma$. Hence, this new data structure contains less information. This raises the question whether the other two representations contain redundant data or whether Girard's box-free proof nets make more identifications, and whether the missing data can be recovered. The answer is that the proof nets of [Gir90] make indeed more proof identifications. For example the following proofs of $\vdash \forall a.a, (\exists b.b \otimes [c \parr c^{\perp}])$ would be identified:



(47)

When translating back to box-nets, we must for each $\forall$-link introduce a box around its whole empire. This can be done because a proof net does not lose its correctness if a $\forall$-box is extended to a larger (correct) subnet, provided the bound variable does not occur freely in the new scope. In [Gir90], Girard avoids this by variable renaming. The reason why this gives unique representatives is the stability and uniqueness of empires in $\mathsf{MLL}^-$ proof nets. However, as already noted in [LS06], under the presence of the units, empires are no longer stable, i.e., due to the mobility of the $\bot$ (the fourth equivalence in 10.1) the empire of an $\forall$-node might be different in different proof graphs, representing the same proof net.

Another reason for not using the solution of [Gir90] is the desire to find a treatment for the quantifiers that is independent from the underlying propositional structure, i.e., that is also applicable to classical logic. While Girard's nets are tightly connected to the structure of $\mathsf{MLL}^-$-proof nets, our presentation is closely related to Miller's expansion trees [Mil87] and the recent development by McKinley [McK08]. Thus, we can hope for a unified treatment of quantifiers in classical and linear logic.

# 13 Conclusions

In this paper we have investigated the relation between three different ways of presenting proofs in MLL2. First, in the sequent calculus, second, in the calculus of structures, and third, via proof graphs and expansion trees, and we have shown how these three presentations can be translated into each other. The main open question is now whether the identifications on proofs made by proof nets (i.e., equivalence classes of proof graphs) is the "right one". The observations in Section 11 show that the last word on this issue is not yet spoken. It would be important, to find independent (category theoretical) axiomatizations for the proof identity in MLL2, based on purely algebraic grounds. Then one could compare this algebraic notion of proof identity for MLL2 with the syntactic one based on proof nets.

Another direction for future research is the question how our method scales to larger fragments of linear logic. This concerns not only the exponentials and the additives [HvG03, HH15] but also higher-order linear logic.

# References

[ACCL91] Martín Abadi, Luca Cardelli, Pierre-Louis Curien, and Jean-Jacques Lévy. Explicit substitutions. *J. of Functional Programming*, 1(4):375–416, 1991.

[And76] Peter B. Andrews. Refutations by matings. *IEEE Transactions on Computers*, C-25:801–807, 1976.

[BCST96] Richard Blute, Robin Cockett, Robert Seely, and Todd Trimble. Natural deduction and coherence for weakly distributive categories. *J. of Pure and Applied Algebra*, 113:229–296, 1996.

[Brü03] Kai Brünnler. *Deep Inference and Symmetry for Classical Proofs*. PhD thesis, Technische Universität Dresden, 2003.

[BT01] Kai Brünnler and Alwen Fernanto Tiu. A local system for classical logic. In R. Nieuwenhuis and A. Voronkov, editors, *LPAR 2001*, volume 2250 of *LNAI*, pages 347–361. Springer, 2001.

[Bus91] Samuel R. Buss. The undecidability of $k$-provability. *Annals of Pure and Applied Logic*, 53:72–102, 1991.

[BvdW95] Gianluigi Bellin and Jacques van de Wiele. Subnets of proof-nets in $\mathsf{MLL}^-$. In J.-Y. Girard, Y. Lafont, and L. Regnier, editors, *Advances in Linear Logic*, volume 222 of *London Mathematical Society Lecture Notes*, pages 249–270. Cambridge University Press, 1995.

[CG14] Kaustuv Chaudhuri and Nicolas Guenot. Equality and fixpoints in the calculus of structures. In Tom Henzinger and Dale Miller, editors, *Proceedings of the Twenty-Third EACSL Annual Conference on Computer Science Logic (CSL) and the Twenty-Ninth Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, 2014.

[CGS11] Kaustuv Chaudhuri, Nicolas Guenot, and Lutz Straßburger. The focused calculus of structures. In Marc Bezem, editor, *CSL'11*, volume 12 of *LIPIcs*, pages 159–173. Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2011.

[dB72] Nicolaas Govert de Bruijn. Lambda-calculus notation with nameless dummies, a tool for automatic formula manipulation. *Indag. Math.*, 34:381–392, 1972.

[DHPP99] H. Devarajan, Dominic Hughes, Gordon Plotkin, and Vaughan R. Pratt. Full completeness of the multiplicative linear logic of Chu spaces. In *14th IEEE Symposium on Logic in Computer Science (LICS 1999)*, 1999.

[DR89] Vincent Danos and Laurent Regnier. The structure of multiplicatives. *Annals of Mathematical Logic*, 28:181–203, 1989.

[GG08] Alessio Guglielmi and Tom Gundersen. Normalisation control in deep inference via atomic flows. *Logical Methods in Computer Science*, 4(1:9):1–36, 2008.

[Gir87] Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.

[Gir90] Jean-Yves Girard. Quantifiers in linear logic II. Preépublication de l'Equipe de Logique, Université Paris VII, Nr. 19, 1990.

[GS01] Alessio Guglielmi and Lutz Straßburger. Non-commutativity and MELL in the calculus of structures. In Laurent Fribourg, editor, *Computer Science Logic, CSL 2001*, volume 2142 of *LNCS*, pages 54–68. Springer-Verlag, 2001.

[GS11] Alessio Guglielmi and Lutz Straßburger. A system of interaction and structure V: the exponentials and splitting. *Mathematical Structures in Computer Science*, 21(3):563–584, 2011.

[Gug07] Alessio Guglielmi. A system of interaction and structure. *ACM Transactions on Computational Logic*, 8(1):1–64, 2007.

[HH14] Willem Heijltjes and Robin Houston. No proof nets for MLL with units: proof equivalence in MLL is pspace-complete. In Thomas A. Henzinger and Dale Miller, editors, *Joint Meeting of the Twenty-Third EACSL Annual Conference on Computer Science Logic (CSL) and the Twenty-Ninth Annual ACM/IEEE Symposium on Logic in Computer Science (LICS), CSL-LICS '14, Vienna, Austria, July 14 - 18, 2014*, pages 50:1–50:10. ACM, 2014.

[HH15] Willem Heijltjes and Dominic J. D. Hughes. Complexity bounds for sum-product logic via additive proof nets and petri nets. In *30th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2015*, pages 80–91. IEEE Computer Society, 2015.

[Hug05a] Dominic Hughes. Simple free star-autonomous categories and full coherence. Preprint, 2005.

[Hug05b] Dominic Hughes. Simple multiplicative proof nets with units. Preprint, 2005.

[HvG03] Dominic Hughes and Rob van Glabbeek. Proof nets for unit-free multiplicative-additive linear logic. In *18th IEEE Symposium on Logic in Computer Science (LICS 2003)*, pages 1–10, 2003.

[Laf88] Yves Lafont. *Logique, Catégories et Machines*. PhD thesis, Université Paris 7, 1988.

[Laf95] Yves Lafont. From proof nets to interaction nets. In J.-Y. Girard, Y. Lafont, and L. Regnier, editors, *Advances in Linear Logic*, volume 222 of *London Mathematical Society Lecture Notes*, pages 225–247. Cambridge University Press, 1995.

[LK06] Alexander Lyaletski and Boris Konev. On herbrand's theorem for intuitionistic logic. In Michael Fisher, Wiebe van der Hoek, Boris Konev, and Alexei Lisitsa, editors, *Logics in Artificial Intelligence: 10th European Conference, JELIA 2006 Liverpool, UK*, pages 293–305. Springer Berlin Heidelberg, 2006.

[LS06] François Lamarche and Lutz Straßburger. From proof nets to the free *-autonomous category. *Logical Methods in Computer Science*, 2(4:3):1–44, 2006.

[McK08] Richard McKinley. On Herbrand's theorem and cut elimination (extended abstract). preprint, 2008.

[Mil87] Dale Miller. A compact representation of proofs. *Studia Logica*, 46(4):347–370, 1987.

[Ret93] Christian Retoré. *Réseaux et Séquents Ordonnés*. PhD thesis, Université Paris VII, 1993.

[SG11] Lutz Straßburger and Alessio Guglielmi. A system of interaction and structure IV: The exponentials and decomposition. *ACM Trans. Comput. Log.*, 12(4):23, 2011.

[SL04] Lutz Straßburger and François Lamarche. On proof nets for multiplicative linear logic with units. In Jerzy Marcinkowski and Andrzej Tarlecki, editors, *Computer Science Logic, CSL 2004*, volume 3210 of *LNCS*, pages 145–159. Springer-Verlag, 2004.

[Str03] Lutz Straßburger. *Linear Logic and Noncommutativity in the Calculus of Structures*. PhD thesis, Technische Universität Dresden, 2003.

[Str09] Lutz Straßburger. Some observations on the proof theory of second order propositional multiplicative linear logic. In Pierre-Louis Curien, editor, *Typed Lambda Calculi and Applications, TLCA'09*, volume 5608 of *Lecture Notes in Computer Science*, pages 309–324. Springer, 2009.

[Str11] Lutz Straßburger. From deep inference to proof nets via cut elimination. *Journal of Logic and Computation*, 21(4):589–624, 2011.

[Tub16] Andrea Aler Tubella. *A study of normalisation through subatomic logic*. PhD thesis, University of Bath, 2016.

**RESEARCH CENTRE**
**SACLAY – ÎLE-DE-FRANCE**

1 rue Honoré d'Estienne d'Orves
Bâtiment Alan Turing
Campus de l'École Polytechnique
91120 Palaiseau

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399