Some Observations on the Proof Theory of Second Order Propositional Multiplicative Linear Logic

Lutz Straßburger

INRIA Saclay – Île-de-France — Équipe-projet Parsifal

École Polytechnique — LIX — Rue de Saclay — 91128 Palaiseau Cedex — France
http://www.lix.polytechnique.fr/Labo/Lutz.Strassburger/

Abstract. We investigate the question of what constitutes a proof when quantifiers and multiplicative units are both present. On the technical level this paper provides two new aspects of the proof theory of MLL2 with units. First, we give a novel proof system in the framework of the calculus of structures. The main feature of the new system is the consequent use of deep inference, which allows us to observe a decomposition which is a version of Herbrand's theorem that is not visible in the sequent calculus. Second, we show a new notion of proof nets which is independent from any deductive system. We have "sequentialisation" into the calculus of structures as well as into the sequent calculus. Since cut elimination is terminating and confluent, we have a category of MLL2 proof nets. The treatment of the units is such that this category is star-autonomous.

1 Introduction

The question of when two proofs are the same is important for proof theory and its applications. It comes down to the question of which information contained in a proof is essential, and which information is purely bureaucratic, due to the chosen deductive system. One of the first results in that direction is Herbrand's theorem which allows a separation between the quantifiers and the propositional fragment of first order classical predicate logic. The work on expansion trees by Miller [1] shows how Herbrand's result can be generalized to higher order. In this paper we present a similar result for linear logic. Our work is motivated by the desire to find eventually a general treatment for the quantifiers, independent from the propositional fragment of the logic (see the related work by McKinley [2]).

The first contribution of this paper is a presentation of MLL2 in the calculus of structures, which is a new deductive formalism using *deep inference*. That means that inferences are allowed anywhere deep inside a formula, very similar to what happens in term rewriting. As a consequence of this freedom we can show a decomposition theorem, which is not possible in the sequent calculus, and which can be seen as a version of Herbrand's Theorem for MLL2. Secondly, we give a combinatorial presentation of MLL2 proofs that we call here *proof nets* (following the tradition) and that quotient away irrelevant rule permutations in the deductive systems (sequent calculus and calculus of structures). The identifications made by these proof nets are consistent with ones for MLL (with units) made by star-autonomous categories [3–5]. The main motivation for these proof nets is to exhibit the precise relation between deep inference and

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

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

Fig. 1. Sequent calculus system for MLL2

the existing presentations of MLL2-proofs: sequent calculus, Girard's proof nets with boxes [6], and Girard's proof nets with jumps [7]. In particular, there is a close connection between the decomposition theorem in deep inference, and the sequentialization of proof nets. Furthermore, our proof nets are the first to accomodate the quantifiers and the multiplicative units together without boxes. The proof nets proposed here are independent from the deductive system, i.e., we do not have the strong connection between links in the proof net and rule applications in the sequent calculus. However, we have "sequentialization" into the sequent calculus as well as into the calculus of structures. As expected, there is a confluent and terminating cut elimination procedure, and thus, the two conclusion proof nets form a category.

2 MLL2 in the sequent calculus

Let us recall how MLL2 is presented in the sequent calculus. Let $\mathscr{A} = \{a, b, c, \ldots\}$ be a countable set of *propositional variables*. Then the set $\mathscr{F}$ of *formulas* is generated by

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

Formulas are denoted by capital Latin letters ($A, B, C, \ldots$). Linear negation $(-)^{\perp}$ is defined for all formulas by the De Morgan laws. *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 $\parr$ and $(\ldots)$ for $\otimes$.
(ii) We omit parentheses if they are superfluous under the assumption that $\parr$ and $\otimes$ associate to the left, e.g., $[A \parr B \parr C \parr D]$ abbreviates $[[[A \parr B] \parr C] \parr D]$.
(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 $[\forall a.(a \otimes b) \parr \exists c.c \parr a]$, 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 $\mathsf{MLL2}_{\mathsf{Seq}}$. As shown in [6], it has the cut elimination property:

2.1 Theorem *The cut rule* $\mathsf{cut}\,\dfrac{\vdash \Gamma, A \quad \vdash A^{\perp}, \Delta}{\vdash \Gamma, \Delta}$ *is admissible for* $\mathsf{MLL2}_{\mathsf{Seq}}$.

$$
\begin{array}{cccc}
\mathsf{ai}\!\downarrow \dfrac{S\{1\}}{S[a^{\perp} \parr a]} &
\bot\!\downarrow \dfrac{S\{A\}}{S[\bot \parr A]} &
1\!\downarrow \dfrac{S\{A\}}{S(1 \otimes A)} &
\mathsf{e}\!\downarrow \dfrac{S\{1\}}{S\{\forall a.1\}} \\[2ex]
\alpha\!\downarrow \dfrac{S[[A \parr B] \parr C]}{S[A \parr [B \parr C]]} &
\sigma\!\downarrow \dfrac{S[A \parr B]}{S[B \parr A]} &
\mathsf{ls}\, \dfrac{S([A \parr B] \otimes C)}{S[A \parr (B \otimes C)]} &
\mathsf{rs}\, \dfrac{S(A \otimes [B \parr C])}{S[(A \otimes B) \parr C]} \\[2ex]
\mathsf{u}\!\downarrow \dfrac{S\{\forall a.[A \parr B]\}}{S[\forall a.A \parr \exists a.B]} &
\mathsf{n}\!\downarrow \dfrac{S\{A\langle a \backslash B \rangle\}}{S\{\exists a.A\}} &
\mathsf{f}\!\downarrow \dfrac{S\{\exists a.A\}}{S\{A\}}\ \text{a not free in A.} &
\end{array}
$$

Fig. 2. Deep inference system for MLL2

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 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., $S[A \parr B]$ abbreviates $S\{[A \parr B]\}$. The system in Figure 2 is called $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$. 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 $\mathsf{MLL2}_{\mathsf{Seq}}$ is in $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$ decomposed into three pieces, namely, $\mathsf{e}\!\downarrow$, $\mathsf{u}\!\downarrow$, and $\mathsf{f}\!\downarrow$. We also need an explicit rule for associativity which is "built in" the sequent calculus. The relation between the $\otimes$-rule and the rules ls and rs (called *left switch* and *right switch*) has already in detail been investigated by several authors [13–15, 9]. The following theorem ensures that $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$ is indeed a deductive system for MLL2.

3.1 Theorem *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.*

As for $\mathsf{MLL2}_{\mathsf{Seq}}$, we also have for $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$ the cut elimination property, which can be stated as follows:

3.2 Theorem *The cut rule* $\mathsf{i}\!\uparrow \dfrac{S(A \otimes A^{\perp})}{S\{\bot\}}$ *is admissible for* $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$.

---

[1] In the literature on deep inference, e.g., [8, 9], the formula $(a \otimes [b \parr (a^{\perp} \otimes c)])$ would be written as $(a, [b, (a^{\perp}, c)])$, while without our convention it would be written as $a \otimes (b \parr (a^{\perp} \otimes c))$. Our 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 $\parr$ on the formula level and the comma on the sequent level, e.g., $[\forall a.(a \otimes b) \parr \exists c.c \parr a]$ 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., [10, 11, 8, 12] for details.

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

$$\mathsf{1f}\!\downarrow\frac{S\{\exists a.1\}}{S\{1\}} \qquad \bot\mathsf{f}\!\downarrow\frac{S\{\exists a.\bot\}}{S\{\bot\}} \qquad \mathsf{af}\!\downarrow\frac{S\{\exists a.b\}}{S\{b\}} \qquad \hat{\mathsf{a}}\mathsf{f}\!\downarrow\frac{S\{\exists a.b^\perp\}}{S\{b^\perp\}} \quad \text{in } \mathsf{af}\!\downarrow \text{ and } \hat{\mathsf{a}}\mathsf{f}\!\downarrow, \ a \text{ is different from } b$$

Fig. 3. Towards a local system for MLL22

We write $\mathsf{MLL2_{DI\downarrow}}\ \begin{array}{c}A\\ \Big\| \mathscr{D}\\ B\end{array}$ for denoting a derivation $\mathscr{D}$ in $\mathsf{MLL2_{DI\downarrow}}$ with premise A and conclusion B. The following decomposition theorem for $\mathsf{MLL2_{DI\downarrow}}$ can be seen as a version of Herbrand's theorem for MLL2 and has no counterpart in the sequent calculus.

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

This decomposition is obtained by permuting all instances of $\mathsf{ai}\!\downarrow,\bot\!\downarrow,1\!\downarrow,\mathsf{e}\!\downarrow$ up and permuting all instances of $\mathsf{n}\!\downarrow,\mathsf{f}\!\downarrow$ down. There are two versions of the "switch" in $\mathsf{MLL2_{DI\downarrow}}$, the *left switch* ls, and the *right switch* rs. For Thm. 3.1, the ls-rule would be sufficient, but for obtaining the decomposition in Thm. 3.3 we also need the rs-rule.

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* [16, 17] 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 with its flow-graph. (We left some some applications of $\alpha\!\downarrow$ and $\sigma\!\downarrow$ implicit.)

$$\mathsf{u}\!\downarrow\dfrac{\mathsf{u}\!\downarrow\dfrac{\mathsf{u}\!\downarrow\dfrac{\mathsf{rs}\dfrac{\mathsf{ls}\dfrac{\forall a.\forall c.([a^\perp\parr a]\otimes[c^\perp\parr c])}{\forall a.\forall c.[a^\perp\parr(a\otimes[c^\perp\parr c])]}}{\forall a.\forall c.[a^\perp\parr[(a\otimes c^\perp)\parr c]]}}{\forall a.[\exists c.a^\perp\parr\forall c.[(a\otimes c^\perp)\parr c]]}}{\forall a.[\exists c.a^\perp\parr[\exists c.(a\otimes c^\perp)\parr\forall c.c]]}}{[\forall a.\exists c.a^\perp\parr\exists a.[\exists c.(a\otimes c^\perp)\parr\forall c.c]]} \tag{1}$$

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 Γ, namely, that the variable a does not appear freely in it. This is the reason for the boxes in [6] and the jumps in [7]. 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 MLL2. Now we have the following:

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

4 Proof nets for MLL2

For defining proof nets for MLL2, we follow the ideas presented in [18, 5] 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 $\parr$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. Now, the set $\mathscr{L}$ of *linking formulas* is generated by the grammar

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

In [18, 5] a proof net consists of the sequent forest and the linking formula. The presence of the quantifiers, in particular, the presence of instantiation and substitution, makes it necessary to expand the structure of the sequent in the proof net. The set $\mathscr{E}$ of *expanded formulas*[3] is generated by

$$\mathscr{E} ::= \bot \mid 1 \mid \mathscr{A} \mid \mathscr{A}^{\bot} \mid [\mathscr{E}\parr\mathscr{E}] \mid (\mathscr{E}\otimes\mathscr{E}) \mid \forall\mathscr{A}.\mathscr{E} \mid \exists\mathscr{A}.\mathscr{E} \mid \rotatebox{0}{$\exists\!\!\!\exists$}\mathscr{A}.\mathscr{E} \mid \boldsymbol{\exists}\mathscr{A}.\mathscr{E}$$

There are only two additional syntactic primitives: the $\exists\!\!\!\exists$, 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 (Γ, Δ, ...). For disambiguation, the formulas/sequents introduced in Section 2 (i.e., those without $\exists\!\!\!\exists$ 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 $\mathscr{A}\cup\mathscr{A}^{\bot}\cup\{1,\bot\}$. We can think of the inner nodes as decorated either with the connectives/quantifiers $\otimes$, $\parr$, $\forall a$, $\exists a$, $\boldsymbol{\exists} a$, $\exists\!\!\!\exists a$, or with the whole subformula rooted at that node. For this reason we will use capital Latin letters (A, B, C, ...) 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 Γ (resp. formula A).

[3] This is almost the same structure as Miller's *expansion trees* [1]. The idea is to code a formula and its "expansion" together in the same syntactic object. But our case is simpler than in [1] because we do not have to deal with duplication.

4.1 Definition A *stretching σ for a sequent Γ* consists of two binary relations $\overset{\sigma}{\underset{+}{\rightarrow}}$ and $\overset{\sigma}{\underset{-}{\rightarrow}}$ on the set of nodes of Γ (i.e., its subformula occurrences) such that $\overset{\sigma}{\underset{+}{\rightarrow}}$ and $\overset{\sigma}{\underset{-}{\rightarrow}}$ are disjoint.

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 expanded sequent Γ with a stretching σ, denoted by $\Gamma \blacktriangleleft \sigma$, we will draw these edges either inside Γ when it is written as a tree, or below Γ when it is written as string. The positive edges are dotted and the negative ones are dashed. Examples are shown in Figures 6, 4 and 5 below. If A is a node in Γ, we write σ_A to denote the restriction of σ to A.

The virtue of second order MLL is the possibility of substitution and instantiation, which is the *raison d'être* of the expansion via $\boldsymbol{\exists}$ and $\exists\!\!\exists$.

4.2 Definition For an expanded formula E and a stretching σ, we define the *ceiling* and the *floor*[4], denoted by $\lceil E \blacktriangleleft \sigma \rceil$ and $\lfloor E \blacktriangleleft \sigma \rfloor$, respectively, to be simple formulas, which are inductively defined as follows:

$$
\begin{array}{rll}
\lceil 1 \blacktriangleleft \emptyset \rceil = 1 & \lceil A \parr B \blacktriangleleft \sigma \rceil = \lceil A \blacktriangleleft \sigma_A \rceil \parr \lceil B \blacktriangleleft \sigma_B \rceil & \\
\lceil \bot \blacktriangleleft \emptyset \rceil = \bot & \lceil A \otimes B \blacktriangleleft \sigma \rceil = \lceil A \blacktriangleleft \sigma_A \rceil \otimes \lceil B \blacktriangleleft \sigma_B \rceil & \\
\lceil a \blacktriangleleft \emptyset \rceil = a & \lceil \forall a.A \blacktriangleleft \sigma \rceil = \forall a.\lceil A \blacktriangleleft \sigma \rceil & \lceil \exists\!\!\exists a.A \blacktriangleleft \sigma \rceil = \exists a.\lceil A \blacktriangleleft \sigma \rceil \\
\lceil a^\bot \blacktriangleleft \emptyset \rceil = a^\bot & \lceil \exists a.A \blacktriangleleft \sigma \rceil = \exists a.\lceil A \blacktriangleleft \sigma \rceil & \lceil \boldsymbol{\exists} a.A \blacktriangleleft \sigma \rceil = \lceil A \blacktriangleleft \sigma_A \rceil
\end{array}
$$

$$
\begin{array}{rll}
\lfloor 1 \blacktriangleleft \emptyset \rfloor = 1 & \lfloor A \parr B \blacktriangleleft \sigma \rfloor = \lfloor A \blacktriangleleft \sigma_A \rfloor \parr \lfloor B \blacktriangleleft \sigma_B \rfloor & \\
\lfloor \bot \blacktriangleleft \emptyset \rfloor = \bot & \lfloor A \otimes B \blacktriangleleft \sigma \rfloor = \lfloor A \blacktriangleleft \sigma_A \rfloor \otimes \lfloor B \blacktriangleleft \sigma_B \rfloor & \\
\lfloor a \blacktriangleleft \emptyset \rfloor = a & \lfloor \forall a.A \blacktriangleleft \sigma \rfloor = \forall a.\lfloor A \blacktriangleleft \sigma \rfloor & \lfloor \exists\!\!\exists a.A \blacktriangleleft \sigma \rfloor = \lfloor A \blacktriangleleft \sigma \rfloor \\
\lfloor a^\bot \blacktriangleleft \emptyset \rfloor = a^\bot & \lfloor \exists a.A \blacktriangleleft \sigma \rfloor = \exists a.\lfloor A \blacktriangleleft \sigma \rfloor & \lfloor \boldsymbol{\exists} a.A \blacktriangleleft \sigma \rfloor = \exists a.\lfloor \tilde{A} \blacktriangleleft \sigma_{\tilde{A}} \rfloor
\end{array}
$$

The expanded formula $\tilde{A}$ in the last line is obtained from A as follows: For every node B with $A \preccurlyeq B$ and $\boldsymbol{\exists} a.A \overset{\sigma}{\underset{+}{\rightarrow}} B$ remove the whole subtree B and replace it by a, and for every B with $\boldsymbol{\exists} a.A \overset{\sigma}{\underset{-}{\rightarrow}} B$ replace B by $a^\bot$.

Note that ceiling and floor of an expanded sequent Γ differ from Γ only on $\boldsymbol{\exists}$ and $\exists\!\!\exists$. In the ceiling, the $\exists\!\!\exists$ is treated as ordinary $\exists$, and the $\boldsymbol{\exists}$ is completely ignored. In the floor, the $\exists\!\!\exists$ 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 4.3 below.

Given $\Gamma \blacktriangleleft \sigma$ and nodes A, B in Γ, then we write $A \rightsquigarrow B$ if A is a $\boldsymbol{\exists}$-node and there is a stretching edge from A to B, or A is an ordinary quantifier node and $A \preccurlyeq B$ and B is the variable (or its negation) that is bound by A in $\lfloor A \blacktriangleleft \sigma_A \rfloor$.

4.3 Definition A pair $\Gamma \blacktriangleleft \sigma$ is *appropriate*, if the following three conditions hold:

1. If $A \overset{\sigma}{\underset{+}{\rightarrow}} B$ and $A \overset{\sigma}{\underset{+}{\rightarrow}} C$, then $\lfloor B \blacktriangleleft \sigma_B \rfloor = \lfloor C \blacktriangleleft \sigma_C \rfloor$,
   if $A \overset{\sigma}{\underset{-}{\rightarrow}} B$ and $A \overset{\sigma}{\underset{-}{\rightarrow}} C$, then $\lfloor B \blacktriangleleft \sigma_B \rfloor = \lfloor C \blacktriangleleft \sigma_C \rfloor$,
   if $A \overset{\sigma}{\underset{+}{\rightarrow}} B$ and $A \overset{\sigma}{\underset{-}{\rightarrow}} C$, then $\lfloor B \blacktriangleleft \sigma_B \rfloor = \lfloor C \blacktriangleleft \sigma_C \rfloor^\bot$.

[4] Note the close correspondece to Miller's expansion trees [1], where these two functions are called *Deep* and *Shallow*, respectively.

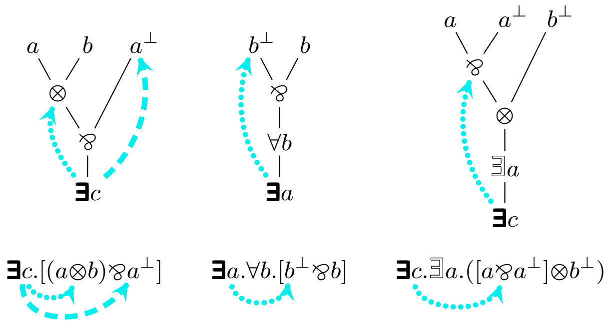

Fig. 4. Examples of expanded sequents with stretchings that are not appropriate

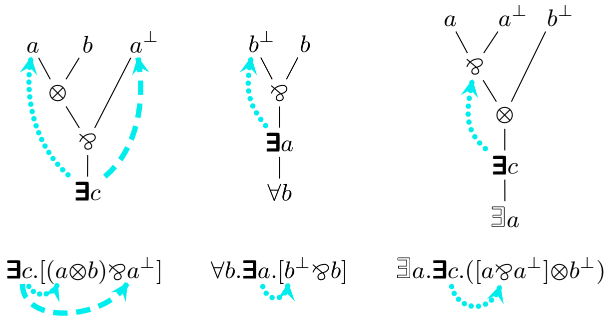

Fig. 5. Appropriate examples of expanded sequents with stretchings

2. If $A_1 \rightsquigarrow B_1$ and $A_2 \rightsquigarrow B_2$ and $A_1 \preccurlyeq A_2$ and $B_1 \preccurlyeq B_2$, then $B_1 \preccurlyeq A_2$.
3. For all $\exists a.A$, the variable a must not occur free in the formula $\lfloor A \blacktriangleleft \sigma_A \rfloor$.

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. 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.

In [6] and [7], the first two conditions of Definition 4.3 appear only implicitly without being mentioned in the treatment of the $\exists$-rule. But for capturing the essence of a proof independently of a deductive system, we have to make everything explicit.

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

For $B \in \varnothing\Gamma$ we write B^ν for its image under ν in $\varnothing P$. When we draw a pre-proof graph $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$, then we write P above Γ (as trees or as strings) and the leaves are connected by edges according to ν. Figure 6 shows an example written in both ways.

4.5 Definition A *switching* s of a pre-proof graph $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$ is the graph that is obtained from the whole of $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$ by removing all stretching edges and by removing

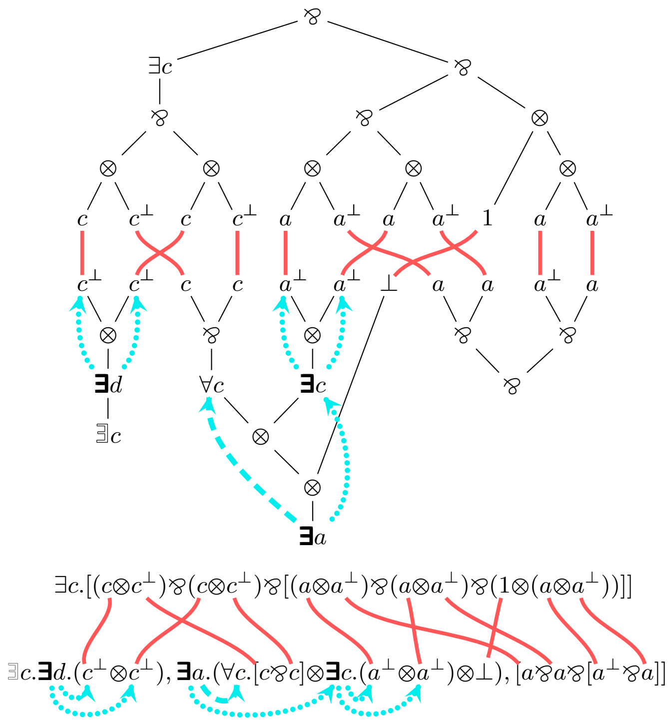

Fig. 6. Two ways of writing a proof graph

for each $\parr$-node one of the two edges connecting it to its children. A pre-proof graph $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$ is *multiplicatively correct* if all its switchings are acyclic and connected [19].

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 Γ. We write $A \smile\!\!\textcircled{s}\!\!\smile B$ if there is a path in s from A to B, starting from A by going down to its parent and coming into B from below. Similarly, one can define the notations $A \frown\!\!\textcircled{s}\!\!\frown B$ and $A \frown\!\!\textcircled{s}\!\!\smile B$ and $A \smile\!\!\textcircled{s}\!\!\frown B$.

Let A and B be nodes in Γ with $A \preccurlyeq B$. The *quantifier depth* of B in A, denoted by $\bigtriangledown_A B$, is the number of quantifier nodes on the path from A to B (including A if it happens to be an $\forall$ or an $\exists$, but not including B). Similarly we define $\bigtriangledown_\Gamma B$. For quantifier nodes A' in P and A in Γ, we say A and A' are *partners*, denoted by $A' \overset{P\ \Gamma}{\leftarrow\!\blacksquare\!\rightarrow} A$, if there is a leaf $B \in \varnothing\Gamma$ with $A \preccurlyeq B$ in Γ, and $A' \preccurlyeq B^\nu$ in P, and $\bigtriangledown_A B = \bigtriangledown_{A'} B^\nu$.

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

1. For every $B \in \varnothing\Gamma$, we have $\bigtriangledown_\Gamma B = \bigtriangledown_P B^\nu$.
2. If $A' \overset{P\ \Gamma}{\leftarrow\!\blacksquare\!\rightarrow} A$, then A' and A quantify the same variable.
3. For every quantifier node A in Γ there is exactly one $\exists$-node A' in P with $A' \overset{P\ \Gamma}{\leftarrow\!\blacksquare\!\rightarrow} A$.

$$
\begin{array}{ll}
(1) & \begin{array}{c}\exists a.\exists c.[(a \otimes a^\perp) \parr (c \otimes c^\perp)] \\ \exists c.a^\perp, \forall a.[\exists c.(a \otimes c^\perp) \parr \forall c.c]\end{array}
\qquad
(2) \; \begin{array}{c}\exists a.\exists c.[(a \otimes a^\perp) \parr (c \otimes c^\perp)] \\ \forall a.\exists b.a^\perp, \exists a.[\exists d.(a \otimes c^\perp) \parr \forall c.c]\end{array} \\[2ex]
(3) & \begin{array}{c}\exists a.[\exists c.(a \otimes a^\perp) \parr \exists c.(c \otimes c^\perp)] \\ \forall a.\exists c.a^\perp, \exists a.[\exists c.(a \otimes c^\perp) \parr \forall c.c]\end{array}
\qquad
(4) \; \begin{array}{c}\exists a.\exists c.[(a \otimes a^\perp) \parr (c \otimes c^\perp)] \\ \exists a.\forall c.a^\perp, \exists a.[\exists c.(a \otimes c^\perp) \parr \forall c.c]\end{array} \\[2ex]
(5) & \begin{array}{c}\exists a.\exists c.[(a \otimes a^\perp) \parr (c \otimes c^\perp)] \\ \forall a.\exists c.a^\perp, \exists a.[(\exists c.a \otimes \exists c.c^\perp) \parr \forall c.c]\end{array}
\qquad
(6) \; \begin{array}{c}\exists a.\exists c.[(a \otimes a^\perp) \parr (c \otimes c^\perp)] \\ \forall a.\exists c.a^\perp, \exists a.[\exists c.(a \otimes c^\perp) \parr \forall c.c]\end{array}
\end{array}
$$

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

4. For every $\exists$-node A' in P there is exactly one $\forall$-node A in Γ with $A' \xleftrightarrow{P \blacksquare \Gamma} A$.
5. If $A' \xleftrightarrow{P \blacksquare \Gamma} A_1$ and $A' \xleftrightarrow{P \blacksquare \Gamma} A_2$, then there is no switching s with $A_1 \overset{s}{\smile} A_2$.

Every quantifier node in P must be an $\exists$, and every quantifier node in Γ has exactly one of them as partner. On the other hand, an $\exists$ in P can have many partners in Γ, but exactly one of them has to be an $\forall$. Following Girard [6], we can call an $\exists$ in P together with its partners in Γ 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. 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.

4.7 Definition We say that a pre-proof graph $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$ is *correct* if the pair $\Gamma \blacktriangleleft \sigma$ is appropriate and the simple pre-proof graph $P \overset{\nu}{\triangleright} \lceil \Gamma \blacktriangleleft \sigma \rceil$ is well-nested and multiplicatively correct. 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 we have that $\lceil \Gamma \blacktriangleleft \sigma \rceil$ is the simple sequent $\vdash \exists c.(c^\perp \otimes c^\perp), (\forall c.[c \parr c] \otimes (a^\perp \otimes a^\perp) \otimes \bot), [a \parr a \parr [a^\perp \parr a]]$ and the conclusion $\lfloor \Gamma \blacktriangleleft \sigma \rfloor$ is $\vdash \exists d.(d \otimes d), \exists a.(a^\perp \otimes a \otimes \bot), [a \parr a \parr [a^\perp \parr a]]$.

As said before, due to the presence of the multiplicative units (see [18, 5]), we need to enforce an equivalence relation on proof graphs.

4.8 Definition Let $\sim$ be the smallest equivalence on proof graphs satisfying

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

$$
\mathsf{id}\,\frac{}{a\otimes a^{\perp} \overset{\nu_0}{\triangleright} a^{\perp}, a \blacktriangleleft \emptyset}
\qquad
\exists\,\frac{P \overset{\nu}{\triangleright} \Gamma, A\langle a\backslash B\rangle \blacktriangleleft \sigma}{P \overset{\nu}{\triangleright} \Gamma, \boldsymbol{\exists} a.A\langle a\backslash B\rangle \blacktriangleleft \sigma'}
\qquad
\forall\,\frac{P \overset{\nu}{\triangleright} A, B_1, \ldots, B_n \blacktriangleleft \sigma}{\exists a.P \overset{\nu}{\triangleright} \forall a.A, \exists\!\!\exists a.B_1, \ldots, \exists\!\!\exists a.B_n \blacktriangleleft \sigma}
$$

$$
\perp\,\frac{P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma}{(1\otimes P) \overset{\nu_{\perp}}{\triangleright} \Gamma, \perp \blacktriangleleft \sigma}
\qquad
\parr\,\frac{P \overset{\nu}{\triangleright} A, B, \Gamma \blacktriangleleft \sigma}{P \overset{\nu}{\triangleright} [A \parr B], \Gamma \blacktriangleleft \sigma}
\qquad
\otimes\,\frac{P \overset{\nu}{\triangleright} \Gamma, A \blacktriangleleft \sigma \qquad Q \overset{\nu'}{\triangleright} B, \Delta \blacktriangleleft \tau}{[P \parr Q] \overset{\nu\cup\nu'}{\triangleright} \Gamma, (A\otimes B), \Delta \blacktriangleleft \sigma\cup\tau}
$$

$$
1\,\frac{}{\perp \overset{\nu_1}{\triangleright} 1 \blacktriangleleft \emptyset}
\qquad
\mathsf{exch}\,\frac{P \overset{\nu}{\triangleright} \Gamma, A, B, \Delta \blacktriangleleft \sigma}{P \overset{\nu}{\triangleright} \Gamma, B, A, \Delta \blacktriangleleft \sigma}
\qquad
\mathsf{cut}\,\frac{P \overset{\nu}{\triangleright} \Gamma, A \blacktriangleleft \sigma \qquad Q \overset{\nu'}{\triangleright} A^{\perp}, \Delta \blacktriangleleft \tau}{[P \parr Q] \overset{\nu\cup\nu'}{\triangleright} \Gamma, A \obar A^{\perp}, \Delta \blacktriangleleft \sigma\cup\tau}
$$

Fig. 8. Translating sequent calculus proofs into proof nets

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

The first two equations in Definition 4.8 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.[5] 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.

5 Sequentialisation

In this section we will discuss how we can translate proofs in the sequent calculus and the calculus of structures into proof nets and back.

Let us begin with the sequent calculus. 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 (ν_0 and ν_1 are uniquely determined and the stretching is empty). In the rule $\perp$, the $\nu_\perp$ is obtained from ν by adding an edge between the new 1 and $\perp$. The exch and $\parr$-rules are also rather trivial (P, ν, and σ 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 $\obar$. 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 $\exists\!\!\exists$ can be interpreted as simulating the border of the box. The $\exists$-rule is the only one where the stretching σ is changed. As shown in Figure 1, in the conclusion of that rule, the

[5] In [18, 5] the relation $\sim$ is defined on pre-proof graphs, and therefore a side condition had to be given to that equation (see also [20]).

subformula B of A is replaced by the 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 σ'.

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.

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

For the other direction, i.e, for going from proof graphs to $\mathsf{MLL2}_{\mathsf{Seq}}$ proofs we need to consider two linking formulas P_1 and P_2 to be equivalent modulo associativity and commutativity of $\otimes$. We write this as $P_1 \overset{\otimes}{\sim} P_2$. Then, we have to remove all $\exists$-nodes from Γ in order to get a sequentialization theorem because the translation shown in Figure 8 never introduces an $\exists$-node in Γ. For this we replace in Γ every $\exists a.A$ with $\exists a.\boldsymbol{\exists} a.A$ 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 $\Gamma \blacktriangleleft \sigma$.

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

The proof works in the usual way by induction on the size of $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$. It is a combination of the sequentialization proofs in [5] and [6], and it makes crucial use of the "splitting tensor lemma" which in our case also needs well-nestedness.

Let us now discuss the translation between proof nets and derivations in the calculus of structures. This can be done in a more modular way than for the sequent calculus.

5.3 Proposition *An* $\mathsf{MLL2}$ *formula P is a linking formula if and only if there is a derivation*

$$\{\mathsf{ai}\!\downarrow, \bot\!\downarrow, 1\!\downarrow, \mathsf{e}\!\downarrow\} \left\| \begin{matrix} 1 \\ \mathscr{D} \\ P^\perp \end{matrix} \right. \quad . \tag{2}$$

5.4 Lemma *Let P_1 and P_2 be two linkings. Then there is a derivation*

$$\{\alpha\!\downarrow, \sigma\!\downarrow, \mathsf{rs}\} \left\| \begin{matrix} P_1 \\ \mathscr{D} \\ P_2 \end{matrix} \right.$$

if and only if the simple pre-proof graph $P_2 \triangleright P_1^\perp$ is correct.

If P_1 and P_2 have this property, we say that P_1 *is weaker than* P_2, and denote it as $P_1 \lesssim P_2$. We can now characterize simple proof graphs in terms of deep inference:

5.5 Proposition *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*

$$\{\alpha\!\downarrow, \sigma\!\downarrow, \mathsf{ls}, \mathsf{rs}, \mathsf{u}\!\downarrow\} \left\| \begin{matrix} P'^\perp \\ \mathscr{D} \\ \Gamma \end{matrix} \right. \quad , \tag{3}$$

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

As an example, consider the derivation in (1) which corresponds to (6) in Figure 7. Finally, we characterize appropriate pairs $\Gamma \blacktriangleleft \sigma$ in terms of deep inference.

5.6 Proposition *For every derivation*

$$\{\mathsf{n}\downarrow, \mathsf{f}\downarrow\} \, \begin{array}{c} D \\ \Big\| \, \mathscr{D} \\ C \end{array} \tag{4}$$

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)*.*

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

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

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

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

We can now easily translate a $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$ proof into a pre-proof graph by first decomposing it via Theorem 3.3 and then applying Propositions 5.3, 5.5, and 5.6. Let us call a pre-proof graph *DI-sequentializable* if is obtained in this way from a $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$ proof.

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

By the method presented in [21], 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 into $\mathsf{MLL2}_{\mathsf{DI}\downarrow}$ proofs (i.e., "sequentialization" into the calculus of structures). Propositions 5.3, 5.5, and 5.6 give us the following:

5.8 Theorem *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.*

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 Proposition 5.5 which deals with the purely multiplicative part. Then comes Proposition 5.6 which only deals with instantiation and substitution, i.e, the second-order aspect. Finally, Proposition 5.3 takes care of the linking, whose task is to describe the role of the units in the proof. Therefore the equivalence in 4.8, which is due to the mobility if the $\perp$, only deals with the linkings. This modularity in the sequentialization is possible because of the decomposition in Theorem 3.3. Because of this modularity we treated the units via the linking formulas [18, 5] instead of a linking function as done by Hughes in [22, 20].

6 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 [7] the units are not considered, and in [6] 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 4.8 and identify proof graphs and proof nets.

The translation from our proof nets to Girard's boxed proof nets of [6] is immediate: If $P \overset{\nu}{\triangleright} \Gamma \blacktriangleleft \sigma$ is a given proof net, then (1) for each $\exists$ in P draw a box around the sub-proof net which has as doors this $\exists$ and its partners in Γ; (2) replace in Γ every node A that is not a $\reflectbox{$\mathbb{E}$}$ by its floor $\lfloor A \blacktriangleleft \sigma \rfloor$, and remove all stretching edges and all $\reflectbox{$\mathbb{E}$}$-nodes, and finally (3) remove all $\exists$- 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. It is clear that in both directions correctness is preserved, i.e., the two criteria 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 the $\exists$-link is attached to it 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.

The proof nets of [7] 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 $\reflectbox{$\mathbb{E}$}$-nodes in Γ. Hence, this new data structure contains less information. This raises the question whether the other two representations contain reduntant 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 [7] 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:

$$\begin{array}{c} \exists a.[(a^{\perp} \otimes a) \parr (c^{\perp} \otimes c)] \\ \forall a.a, \reflectbox{$\mathbb{E}$}a.(\boldsymbol{\exists} b.a^{\perp} \otimes [c \parr c^{\perp}]) \end{array} \quad \text{and} \quad \begin{array}{c} [\exists a.(a^{\perp} \otimes a) \parr (c^{\perp} \otimes c)] \\ \forall a.a, (\reflectbox{$\mathbb{E}$}a.\boldsymbol{\exists} b.a^{\perp} \otimes [c \parr c^{\perp}]) \end{array} \tag{6}$$

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 [7], Girard avoids this by variable renaming. The reason why this gives unique representants is the stability and uniqueness of empires in MLL^{-} proof nets. However, as already noted in [5], under the presence of the units, empires are no longer stable, i.e., due to the mobility of the $\perp$ 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 [7] 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 MLL^--proof nets, our presentation is closely related to Miller's expansion trees [1] and the recent development by McKinley [2]. Thus, we can hope for a unified treatment of quantifiers in classical and linear logic.

7 Concluding Remarks

We have investigated the relation between deep inference and proof nets and the sequent calculus for $\mathsf{MLL2}$, and we have shown that this relation is much closer than one might expect. We did not go into the details of cut elimination because from the previous sections it should be clear that everything works as laid out in [6, 7] and [5, 18]. There are no technical surprises, and we have a confluent and terminating cut elimination procedure for our proof nets. An important consequence 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 $\vdash A^\perp, B$, where the composition of maps is defined by cut elimination. A detailed investigation of this category (which is *-autonomous [5]) has to be postponed to future research. The proof identifications made in this paper are motivated by the interplay between proof nets, calculus of structures, and sequent calculus. They should not be considered to be the final word. For example the proof nets by Girard [7] make more identifications, and the ones by Hughes [22] make less identifications. However, there are some observations about the units to be made here. The units can be expressed with the second-order quantifiers via $1 \equiv \forall a.[a^\perp \parr a]$ and $\perp \equiv \exists a.(a \otimes a^\perp)$. 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 [6] they are, but in our category of proof nets they are not: The two canonical maps $\forall a.[a^\perp \parr a] \to 1$ and $1 \to \forall a.[a^\perp \parr a]$ are given by:

$$\begin{array}{c}[\perp \parr (1 \otimes \perp)]\\ \boldsymbol{\exists} a.(1 \otimes \perp)\ ,\ 1\end{array} \qquad \text{and} \qquad \begin{array}{c}(1 \otimes \exists a.(a \otimes a^\perp))\\ \perp\ ,\ \forall a.[a^\perp \parr a]\end{array} \tag{7}$$

respectively. Composing them means performing this cut eliminating:

$$\begin{array}{c}[\perp \parr (1 \otimes \perp) \parr (1 \otimes \exists a.(a \otimes a^\perp))]\\ \boldsymbol{\exists} a.(1 \otimes \perp)\ ,\ 1 \oplus \perp\ ,\ \forall a.[a^\perp \parr a]\end{array} \quad \to \quad \begin{array}{c}[\perp \parr (1 \otimes \exists a.(a \otimes a^\perp))]\\ \boldsymbol{\exists} a.(1 \otimes \perp)\ ,\ \forall a.[a^\perp \parr a]\end{array} \tag{8}$$

If the two maps in (7) where isos, the result of (8) 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

$$\begin{array}{c}\exists a.[(a^\perp \otimes a) \parr (a \otimes a^\perp)]\\ \exists a.(a \otimes a^\perp)\ ,\ \forall a.[a^\perp \parr a]\end{array} \tag{9}$$

This is obviously not the case (even if we replaced $\exists a$ by $\mathbb{\exists} a.\boldsymbol{\exists} a$ as for Theorem 5.2). A similar situation occurs with the additive units, for which we have $0 \equiv \forall a.a$ and $\top \equiv \exists a.a$. 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 $\forall a.a \rightarrow A$ for every formula A, but it is *not* necessarily unique. The correct treatment of additive units in proof nets is still an open problem for future research.

References

1. Miller, D.: A compact representation of proofs. Studia Logica **46**(4) (1987) 347–370
2. McKinley, R.: On Herbrand's theorem and cut elimination (extended abstract). preprint (2008)
3. Barr, M.: *-Autonomous Categories. Volume 752 of Lecture Notes in Mathematics. Springer (1979)
4. Lafont, Y.: Logique, Catégories et Machines. PhD thesis, Université Paris 7 (1988)
5. Lamarche, F., Straßburger, L.: From proof nets to the free *-autonomous category. Logical Methods in Computer Science **2**(4:3) (2006) 1–44
6. Girard, J.Y.: Linear logic. Theoretical Computer Science **50** (1987) 1–102
7. Girard, J.Y.: Quantifiers in linear logic II. Preépublication de l'Equipe de Logique, Université Paris VII, Nr. 19 (1990)
8. Brünnler, K., Tiu, A.F.: A local system for classical logic. In Nieuwenhuis, R., Voronkov, A., eds.: LPAR 2001. Volume 2250 of LNAI., Springer (2001) 347–361
9. Guglielmi, A.: A system of interaction and structure. ACM Transactions on Computational Logic **8**(1) (2007)
10. Straßburger, L.: Linear Logic and Noncommutativity in the Calculus of Structures. PhD thesis, Technische Universität Dresden (2003)
11. Brünnler, K.: Deep Inference and Symmetry for Classical Proofs. PhD thesis, Technische Universität Dresden (2003)
12. Guglielmi, A., Straßburger, L.: Non-commutativity and MELL in the calculus of structures. In Fribourg, L., ed.: Computer Science Logic, CSL 2001. Volume 2142 of LNCS., Springer-Verlag (2001) 54–68
13. Retoré, C.: Réseaux et Séquents Ordonnés. Thèse de Doctorat, spécialité mathématiques, Université Paris VII (February 1993)
14. Blute, R., Cockett, R., Seely, R., Trimble, T.: Natural deduction and coherence for weakly distributive categories. Journal of Pure and Applied Algebra **113** (1996) 229–296
15. Devarajan, H., Hughes, D., Plotkin, G., Pratt, V.R.: Full completeness of the multiplicative linear logic of Chu spaces. In: 14th IEEE Symposium on Logic in Computer Science (LICS 1999). (1999)
16. Buss, S.R.: The undecidability of k-provability. Annals of Pure and Applied Logic **53** (1991) 72–102
17. Guglielmi, A., Gundersen, T.: Normalisation control in deep inference via atomic flows. Logical Methods in Computer Science **4**(1:9) (2008) 1–36
18. Straßburger, L., Lamarche, F.: On proof nets for multiplicative linear logic with units. In Marcinkowski, J., Tarlecki, A., eds.: Computer Science Logic, CSL 2004. Volume 3210 of LNCS., Springer-Verlag (2004) 145–159
19. Danos, V., Regnier, L.: The structure of multiplicatives. Annals of Mathematical Logic **28** (1989) 181–203
20. Hughes, D.: Simple free star-autonomous categories and full coherence. Preprint (2005)
21. Straßburger, L.: From deep inference to proof nets. In: Structures and Deduction — The Quest for the Essence of Proofs (Satellite Workshop of ICALP 2005). (2005)
22. Hughes, D.: Simple multiplicative proof nets with units. Preprint (2005)