

Combinatorial Proofs and Decomposition Theorems for First-order Logic

Abstract—We uncover a close relationship between combinatorial and syntactic proofs for first-order logic. Whereas syntactic proofs are formalized in a deductive proof system based on inference rules, a combinatorial proof is a syntax-free presentation of a proof that is independent from any set of inference rules. We show that the two proof representations are related via a deep inference decomposition theorem that establishes a new kind of normal form for syntactic proofs. This yields (a) a simple proof of soundness and completeness for first-order combinatorial proofs, and (b) a full completeness theorem: every combinatorial proof is the image of a syntactic proof.

I. INTRODUCTION

First-order predicate logic is a cornerstone of modern logic. Since its formalisation by Frege [1] it has seen a growing usage in many fields of mathematics and computer science. Upon the development of proof theory by Hilbert [2], *proofs* became first-class citizens as mathematical objects that could be studied on their own. Since Gentzen’s *sequent calculus* [3], [4], many other proof systems have been developed that allow the implementation of efficient proof search, for example *analytic tableaux* [5] or *resolution* [6]. Despite the immense progress made in proof theory in general and in the area of automated and interactive theorem provers in particular, we still have no satisfactory notion of proof identity for first-order logic. In this respect, proof theory is quite different from any other mathematical field. For example in group theory, two groups are *the same* iff they are isomorphic; in topology, two spaces are *the same* iff they are homeomorphic; etc. In proof theory, we have no such notion telling us when two proofs are *the same*, even though Hilbert was considering this problem as a possible 24th problem for his famous lecture [7] in 1900 [8], before proof theory existed as a mathematical field.

The main reason for this problem is that formal proofs, as they are usually studied in logic, are inextricably tied to the syntactic (inference rule based) proof system in which they are carried out. And it is difficult to compare two proofs that are produced within two different syntactic proof systems, based on different sets of inference rules. Just consider the derivations in Figure 1, showing two proofs of the formula $((\bar{p} \vee q) \wedge \bar{p}) \vee p$ and two proofs of the formula $\exists x.(\bar{p}x \vee (\forall y.py))$, one in the sequent calculus (top) and one in a deep inference system (bottom). It is, *a priori*, not clear how to compare them.

This is where *combinatorial proofs* come in. They were introduced by Hughes [9] for classical propositional logic as a syntax-free notion of proof, and as a potential solution to Hilbert’s 24th problem [10] (see also [11]). The basic idea is to abstract away from the syntax of the inference rules used in inductively-generated proofs and consider the proof as a combinatorial object, more precisely as a special kind of graph homomorphism. For example, a propositional combinatorial

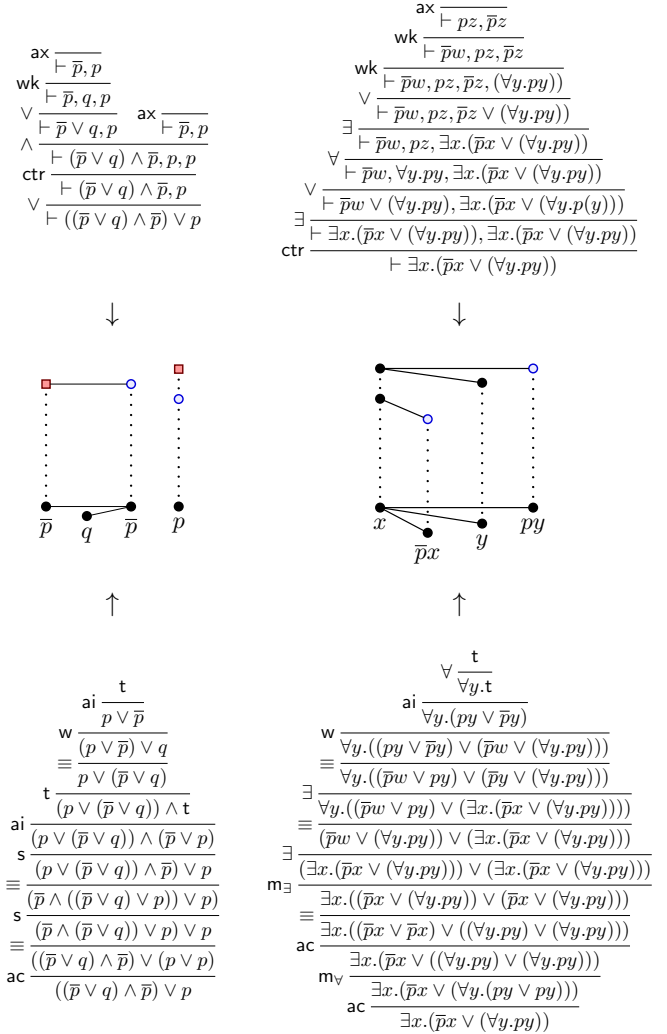


Fig. 1. Left: syntactic proofs in sequent calculus (above) and the calculus of structures (below) which translate to the same propositional combinatorial proof (centre). Right: syntactic proofs in sequent calculus (above) and the new calculus KS1 introduced in this paper (below), which translate to the same first-order combinatorial proof (centre).

proof of Peirce’s law $((p \Rightarrow q) \Rightarrow p) \Rightarrow p = ((\bar{p} \vee q) \wedge \bar{p}) \vee p$ is shown mid-left in Fig. 1, a homomorphism from a coloured graph to a graph labelled with propositional variables.

Several authors have illustrated how syntactic proofs in various proof systems can be translated to propositional combinatorial proofs: for sequent proofs in [10], for deep inference proofs in [12], for Frege systems in [13], and for tableaux systems and resolution in [14]. This enables a natural definition of proof identity for propositional logic: two proofs are *the same*, if they are mapped to the same combinatorial proof. For example, the left side of Fig. 1 translates syntactic proofs from sequent calculus and the calculus of structures

41
42
43
44
45
46
47
48
49
50
51
52

into the same combinatorial proofs, witnessing that the two syntactic proofs, from different systems, are *the same*.

Recently, Acclavio and Straßburger extended this notion to relevant logics [15] and to modal logics [16], and Heijlties, Hughes and Straßburger have provided combinatorial proofs for intuitionistic propositional logic [17].

In this paper we advance the idea that combinatorial proofs can provide a notion of proof identity for first-order logic. *First-order combinatorial proofs* were introduced by Hughes in [18]. For example, a first-order combinatorial proof of Smullyan’s “drinker paradox” $\exists x(px \Rightarrow \forall y py) = \exists x.(\bar{p}x \vee (\forall y.py))$ is shown on the right of Fig. 1, a homomorphism from a partially coloured graph to a labelled graph. However, even though Hughes proves soundness and completeness, the proof is highly unsatisfactory: (1) the soundness argument is extremely long, intricate and cumbersome, and (2) the completeness proof does not allow a syntactic proof to be read back from a combinatorial proof, i.e., completeness is not *sequentializable* [19] nor *full* [20]. A fundamental problem is that not all combinatorial proofs can be obtained as translations of sequent calculus proofs.

We solve these issues by moving to a deep inference system. More precisely, we introduce a new proof system, called KS1, for first-order logic, that (a) reflects every combinatorial proof, i.e., there is a surjection from KS1 proofs to combinatorial proofs, (b) yields far simpler proofs of soundness and completeness for combinatorial proofs, and (c) admits new decomposition theorems establishing a precise correspondence between certain syntactic inference rules and certain combinatorial notions. The right side of Fig. 1 illustrates the surjection in (a), and since the syntactic proofs of the two systems both translate the same combinatorial proof, they can be considered *the same*.

In general, a *decomposition theorem* provides normal forms of proofs, separating subsets of inference rules of a proof system. A prominent example of a decomposition theorem is Herbrand’s theorem [21], which allows a separation between the propositional part and the quantifier part in a first-order proof [4], [22]. Through the advent of deep inference, new kinds of proof decompositions became possible, most notably the separation between the linear part of a proof and the resource management of a proof. It has been shown by Straßburger [23] that a proof in classical propositional logic can be decomposed into a proof of multiplicative linear logic, followed by a proof consisting only of contractions and weakenings. In this paper we show that the same is possible for first-order logic.

Combinatorial proofs and deep inference can be seen as opposite ends of a spectrum: whereas deep inference allows for a very fine granularity of inference rules—one inference rule in a standard formalism, like sequent calculus or semantic tableaux, is usually simulated by a sequence of different deep inference rules—combinatorial proofs have completely abolished the concept of inference rule. And yet, there is a close relationship between the two, realized through a decomposition theorem, as we establish in this paper.

A. Terms and Formulas

Fix pairwise disjoint countable sets $\text{VAR} = \{x, y, z, \dots\}$ of variables, $\text{FUN} = \{f, g, \dots\}$ of function symbols, and $\text{PRED} = \{p, q, \dots\}$ of predicate symbols. Each function symbol and each predicate symbol has a finite arity. The grammars below generate the set TERM of *terms*, denoted by s, t, u, \dots , the set ATOM of *atoms*, denoted by a, b, c, \dots , and the set FORM of *formulas*, denoted by A, B, C, \dots :

$$\begin{aligned} t &::= x \mid f(t_1, \dots, t_n) \\ a &::= p(t_1, \dots, t_n) \mid \bar{p}(t_1, \dots, t_n) \\ A &::= a \mid \mathbf{t} \mid \mathbf{f} \mid A \wedge A \mid A \vee A \mid \exists x.A \mid \forall x.A \end{aligned}$$

where the arity of f and p is n . For better readability of often omit parentheses and write simply $ft_1 \dots t_n$ or $pt_1 \dots t_n$. We consider the truth constants \mathbf{t} (*true*) and \mathbf{f} (*false*) as additional atoms, and we consider all formulas in negation normal form, where *negation* ($\bar{\cdot}$) is defined on atoms and formulas via De Morgan’s laws:

$$\begin{aligned} \bar{\bar{a}} &= a & \bar{\mathbf{t}} &= \mathbf{f} & \overline{p(t_1, \dots, t_n)} &= \bar{p}(t_1, \dots, t_n) \\ \bar{\mathbf{f}} &= \mathbf{t} & \overline{\bar{p}(t_1, \dots, t_n)} &= p(t_1, \dots, t_n) \\ \overline{\exists x.A} &= \forall x.\bar{A} & \overline{A \wedge B} &= \bar{A} \vee \bar{B} \\ \overline{\forall x.A} &= \exists x.\bar{A} & \overline{A \vee B} &= \bar{A} \wedge \bar{B} \end{aligned}$$

Then we write $A \Rightarrow B$ as abbreviation for $\bar{A} \vee B$.

A formula is *rectified* if all bound variables are distinct from one another and from all free variables. Every formula can be transformed into a logically equivalent rectified form, by simply renaming the bound variables. If we consider formulas equivalent modulo α -conversion (renaming of bound variables), then the rectified form of a formula A is uniquely defined, and we denote it by \hat{A} .

A *substitution* is a function $\sigma: \text{VAR} \rightarrow \text{TERM}$ that is the identity almost everywhere. We denote substitutions as $\sigma = [x_1/t_1, \dots, x_n/t_n]$, where $\sigma(x_i) = t_i$ for $i = 1..n$ and $\sigma(x) = x$ for all $x \notin \{x_1, \dots, x_n\}$. Write $A\sigma$ for the formula obtained from A by applying σ , i.e., by simultaneously replacing all occurrences of x_i by t_i . A *variable renaming* is a substitution ρ with $\rho(x) \in \text{VAR}$ for all variables x .

B. Sequent Calculus LK1

Sequents, denoted by Γ, Δ, \dots , are finite multisets of formulas, written as lists, separated by comma. The *corresponding formula* of a (non-empty) sequent $\Gamma = A_1, A_2, \dots, A_n$ is the disjunction of its formulas: $\bigvee(\Gamma) = A_1 \vee A_2 \vee \dots \vee A_n$. A sequent is *rectified* iff its corresponding formula is.

In this paper we use the sequent calculus LK1, shown in Figure 2, which is a one-sided variant of Gentzen’s original calculus [3] for first-order logic. To simplify some technicalities later in this paper, we include the mix-rule.

Theorem 1. LK1 is sound and complete for first-order logic.

For a proof, see any standard textbook, e.g. [24].

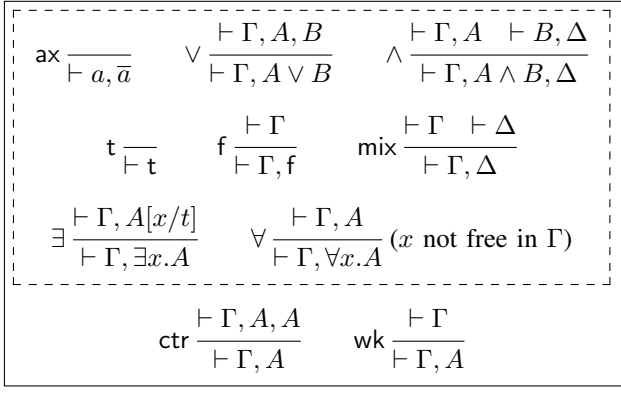


Fig. 2. Sequent calculi LK1 (all rules) and MLL1^X (rules in the dashed box)

The linear fragment of LK1, i.e., the fragment without the rules *ctr* (contraction) and *wk* (weakening) defines first-order multiplicative linear logic [19], [25] with *mix* [26], [27] (MLL1+*mix*). We denote that system here with MLL1^X (shown in Figure 2 in the dashed box).

We will use the cut elimination theorem. The *cut* rule is

$$\text{cut} \frac{\vdash \Gamma, A \quad \vdash \bar{A}, \Delta}{\vdash \Gamma, \Delta} \quad (1)$$

Theorem 2. *If a sequent $\vdash \Gamma$ is provable in LK1+cut then it is also provable in LK1. Furthermore, if $\vdash \Gamma$ is provable in MLL1^X+cut then it is also provable in MLL1^X.*

As before, this is standard, see e.g. [24] for a proof.

III. PRELIMINARIES: FIRST-ORDER GRAPHS

A. Graphs

A **graph** $\mathcal{G} = \langle V_{\mathcal{G}}, E_{\mathcal{G}} \rangle$ is a pair where $V_{\mathcal{G}}$ is a finite set of **vertices** and $E_{\mathcal{G}}$ is a finite set of **edges**, which are two-element subsets of $V_{\mathcal{G}}$. We write vw for an edge $\{v, w\}$.

The **complement** of a graph $\mathcal{G} = \langle V_{\mathcal{G}}, E_{\mathcal{G}} \rangle$ is the graph $\mathcal{G}^c = \langle V_{\mathcal{G}}, E_{\mathcal{G}}^c \rangle$ where $vw \in E_{\mathcal{G}}^c$ iff $vw \notin E_{\mathcal{G}}$.

Let $\mathcal{G} = \langle V_{\mathcal{G}}, E_{\mathcal{G}} \rangle$ and $\mathcal{H} = \langle V_{\mathcal{H}}, E_{\mathcal{H}} \rangle$ be graphs such that $V_{\mathcal{G}} \cap V_{\mathcal{H}} = \emptyset$. A **homomorphism** $\varphi: \mathcal{G} \rightarrow \mathcal{H}$ is a function $\varphi: V_{\mathcal{G}} \rightarrow V_{\mathcal{H}}$ such that if $vw \in E_{\mathcal{G}}$ then $\varphi(v)\varphi(w) \in E_{\mathcal{H}}$. The **union** $\mathcal{G} + \mathcal{H}$ is the graph $\langle V_{\mathcal{G}} \cup V_{\mathcal{H}}, E_{\mathcal{G}} \cup E_{\mathcal{H}} \rangle$ and the **join** $\mathcal{G} \times \mathcal{H}$ is the graph $\langle V_{\mathcal{G}} \cup V_{\mathcal{H}}, E_{\mathcal{G}} \cup E_{\mathcal{H}} \cup \{vw \mid v \in V_{\mathcal{G}}, w \in V_{\mathcal{H}}\} \rangle$. A graph \mathcal{G} is **disconnected** if $\mathcal{G} = \mathcal{G}_1 + \mathcal{G}_2$ for two non-empty graphs $\mathcal{G}_1, \mathcal{G}_2$, otherwise it is **connected**. It is **coconnected** if its complement is connected.

A graph \mathcal{G} is **labelled** in a set L if each vertex $v \in V_{\mathcal{G}}$ has an element $\ell(v) \in L$ associated with it, its **label**. A graph \mathcal{G} is (partially) **coloured** if it carries a partial equivalence relation $\sim_{\mathcal{G}}$ on $V_{\mathcal{G}}$; each equivalence class is a **colour**. A **vertex renaming** of $\mathcal{G} = \langle V_{\mathcal{G}}, E_{\mathcal{G}} \rangle$ along a bijection $(\cdot): V_{\mathcal{G}} \rightarrow \hat{V}_{\mathcal{G}}$ is the graph $\hat{\mathcal{G}} = \langle \hat{V}_{\mathcal{G}}, \{\hat{v}\hat{w} \mid vw \in E_{\mathcal{G}}\} \rangle$, with colouring and/or labelling inherited (i.e., $\hat{v} \sim \hat{w}$ if $v \sim w$, and $\ell(\hat{v}) = \ell(v)$). Following standard graph theory, we identify graphs modulo vertex renaming.

A **directed graph** $\mathcal{G} = \langle V_{\mathcal{G}}, E_{\mathcal{G}} \rangle$ is a set $V_{\mathcal{G}}$ of **vertices** and a set $E_{\mathcal{G}} \subseteq V_{\mathcal{G}} \times V_{\mathcal{G}}$ of **direct edges**. A **directed graph homomorphism** $\varphi: \langle V_{\mathcal{G}}, E_{\mathcal{G}} \rangle \rightarrow \langle V_{\mathcal{H}}, E_{\mathcal{H}} \rangle$ is a function $\varphi: V_{\mathcal{G}} \rightarrow V_{\mathcal{H}}$ such that if $(v, w) \in E_{\mathcal{G}}$ then $(\varphi(v), \varphi(w)) \in E_{\mathcal{H}}$.

B. Cographs

A graph $\mathcal{H} = \langle V_{\mathcal{H}}, E_{\mathcal{H}} \rangle$ is a **subgraph** of a graph $\mathcal{G} = \langle V_{\mathcal{G}}, E_{\mathcal{G}} \rangle$ if $V_{\mathcal{H}} \subseteq V_{\mathcal{G}}$ and $E_{\mathcal{H}} \subseteq E_{\mathcal{G}}$. It is **induced** if $v, w \in V_{\mathcal{H}}$ and $vw \in E_{\mathcal{G}}$ implies $vw \in E_{\mathcal{H}}$. An induced subgraph of $\mathcal{G} = \langle V_{\mathcal{G}}, E_{\mathcal{G}} \rangle$ is uniquely determined by its set of vertices V and we denote it by $\mathcal{G}[V]$. A graph is **\mathcal{H} -free** if it does not contain \mathcal{H} as an induced subgraph. The graph \mathbf{P}_4 is the (undirected) graph $\langle \{v_1, v_2, v_3, v_4\}, \{v_1v_2, v_2v_3, v_3v_4\} \rangle$. A **cograph** is a \mathbf{P}_4 -free undirected graph. The interest in cographs for our paper comes from the following well-known fact.

Theorem 3 ([28]). *A graph is a cograph iff it can be constructed from the singletons via the operations $+$ and \times .*

In a graph \mathcal{G} , the **neighbourhood** $N(v)$ of a vertex $v \in V_{\mathcal{G}}$ is defined as the set $\{w \mid vw \in E_{\mathcal{G}}\}$. A **module** is a set $M \subseteq V_{\mathcal{G}}$ such that $N(v) \setminus M = N(w) \setminus M$ for all $v, w \in M$. A module M is **strong** if for every module M' , we have $M' \subseteq M$, $M \subseteq M'$ or $M \cap M' = \emptyset$. A module is **proper** if it has two or more vertices.

C. Fographs

A cograph is **logical** if every vertex is labelled by either an atom or variable, and it has at least one atom-labelled vertex. An atom-labelled vertex is called a **literal** and a variable-labelled vertex is called a **binder**. A binder labelled with x is called an **x -binder**. The **scope** of a binder b is the smallest proper strong module containing b . An **x -literal** is a literal whose atom contains the variable x . An x -binder **binds** every x -literal in its scope. In a logical cograph \mathcal{G} , a binder b is **existential** (resp. **universal**) if, for every other vertex v in its scope, we have $bv \in E_{\mathcal{G}}$ (resp. $bv \notin E_{\mathcal{G}}$). An x -binder is **legal** if its scope contains no other x -binder and at least one literal.

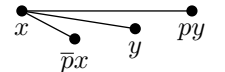
Definition 4. A **first-order graph** or **fograph** is a logical cograph with legal binders. The **binding graph** of a fograph \mathcal{G} is the directed graph $\hat{\mathcal{G}} = \langle V_{\mathcal{G}}, \{(b, l) \mid b \text{ binds } l\} \rangle$.

We define a mapping $\llbracket \cdot \rrbracket$ from formulas to (labelled) graphs, inductively as follows:

$$\begin{aligned} \llbracket t \rrbracket &= \bullet t & \llbracket f \rrbracket &= \bullet f & \llbracket a \rrbracket &= \bullet a \quad (\text{for any atom } a) \\ \llbracket A \vee B \rrbracket &= \llbracket A \rrbracket + \llbracket B \rrbracket & \llbracket \exists x. A \rrbracket &= \bullet x \times \llbracket A \rrbracket \\ \llbracket A \wedge B \rrbracket &= \llbracket A \rrbracket \times \llbracket B \rrbracket & \llbracket \forall x. A \rrbracket &= \bullet x + \llbracket A \rrbracket \end{aligned}$$

where we write $\bullet \alpha$ for a single-vertex labelled by α .

Example 5. Here is the fograph of the drinker formula $\exists x(p_x \Rightarrow \forall y py) = \exists x.(\bar{p}_x \vee (\forall y. py))$:



Lemma 6. *If A is a rectified formula then $\llbracket A \rrbracket$ is a fograph.*

212 *Proof.* That $\llbracket A \rrbracket$ is a logical cograph follows immediately from
 213 the definition and Theorem 3. The fact that every binder of
 214 $\llbracket A \rrbracket$ is legal can be proved by structural induction on A . \square

215 **Remark 7.** Note that $\llbracket A \rrbracket$ is not necessarily a fograph if A
 216 is not rectified. If $A = (\forall x.p(x)) \vee (\forall x.q(x))$, then $\llbracket A \rrbracket =$
 217 $\bullet x \bullet p(x) \bullet x \bullet q(x)$, the scope of each x -binder contains
 218 all the vertices, in particular, the other x -binder. On the other
 219 hand, there are non-rectified formulas which are translated to
 220 fographs by $\llbracket \cdot \rrbracket$. For example, in the graph of $(\exists x.p(x)) \vee$
 221 $(\exists x.q(x))$, both x -binders are legal, as they are not in each
 222 other's scope: $x \bullet \bullet px \quad x \bullet \bullet qx$.

We define a congruence relation \equiv on formulas by the follow-
 ing equations:

$$\begin{aligned} A \wedge B &\equiv B \wedge A & (A \wedge B) \wedge C &\equiv A \wedge (B \wedge C) \\ A \vee B &\equiv B \vee A & (A \vee B) \vee C &\equiv A \vee (B \vee C) \\ \forall x.\forall y.A &\equiv \forall y.\forall x.A & \forall x.(A \vee B) &\equiv (\forall x.A) \vee B \\ \exists x.\exists y.A &\equiv \exists y.\exists x.A & \exists x.(A \wedge B) &\equiv (\exists x.A) \wedge B \end{aligned} \quad (2)$$

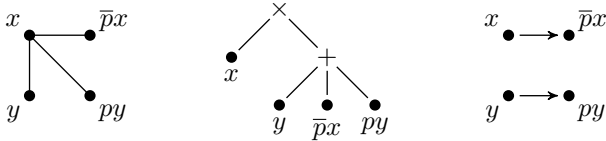
223 where x must not be free in B in the last two equations. Two
 224 formulas A and B are **equivalent** if $A \equiv B$. The following
 225 theorem shows that the set of fographs can be seen as the
 226 quotient FORM/\equiv .

Theorem 8. Let A, B be rectified formulas. Then

$$A \equiv B \iff \llbracket A \rrbracket = \llbracket B \rrbracket$$

227 *Proof.* By a straightforward induction on A . \square

Example 9. Both $\exists x.(\bar{p}x \vee (\forall y.py))$ and $\exists x \forall y(py \vee \bar{p}x)$, which
 are equivalent modulo \equiv , have the same (rectified) fograph \mathcal{D} ,
 shown below-left.



228 Above-middle we show the *cotree* of the underlying cograph
 229 (illustrating the idea behind Theorem 3) and above-right is its
 230 binding graph $\vec{\mathcal{D}}$.

IV. FIRST-ORDER COMBINATORIAL PROOFS

A. Fonets

233 Two atoms are **pre-dual** if their predicate symbols are dual
 234 (e.g. $p(x, y)$ and $\bar{p}(y, z)$) and two literals are **pre-dual** if their
 235 labels (atoms) are pre-dual. A **linked fograph** $\langle \mathcal{C}, \sim_{\mathcal{C}} \rangle$ is a
 236 coloured fograph \mathcal{C} such that every colour (i.e., equivalence
 237 class of $\sim_{\mathcal{C}}$), called a **link**, consists of two pre-dual literals,
 238 and every literal is either t-labelled or in a link.

239 Let \mathcal{C} be a linked fograph. The set of links can be seen as
 240 a unification problem by identifying dual predicate symbols.
 241 A **dualizer** of \mathcal{C} is a substitution δ unifying all the links of \mathcal{C} .
 242 Since a first-order unification problem is either unsolvable
 243 or has a most general unifier, we can define the notion of
 244 **most general dualizer**. A **dependency** is a pair $\{\bullet x, \bullet y\}$ of an
 245 existential binder $\bullet x$ and a universal binder $\bullet y$ such that the

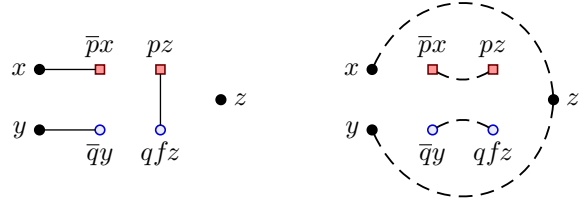


Fig. 3. A fonet (left) with dualizer $[x/z, y/fz]$ and its leap graph (right).

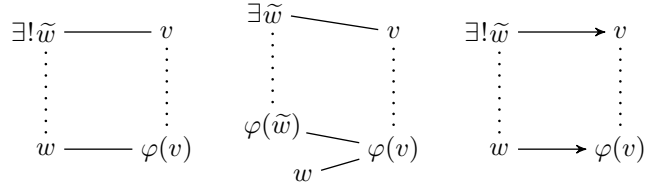
most general dualizer assigns to x a term containing y . A **leap**
 is either a link or a dependency. The **leap graph** \mathcal{C}^L of \mathcal{C} is
 the undirected graph $\langle V_{\mathcal{C}}, L_{\mathcal{C}} \rangle$ where $L_{\mathcal{C}}$ is the set of leaps of
 \mathcal{C} . A vertex set $W \subseteq V_{\mathcal{C}}$ induces a **matching** in \mathcal{C} if $W \neq \emptyset$
 and for all $w \in W$, $N(w) \cap W$ is a singleton. We say that W
 induces a **bimatching** in \mathcal{C} if it induces a matching in \mathcal{C} and
 a matching in \mathcal{C}^L .

Definition 10. A **first-order net** or **fonet** is a linked fograph
 which has a dualizer but no induced bimatching.

Figure 3 shows a fonet with a unique dualizer, and its leap
 graph.

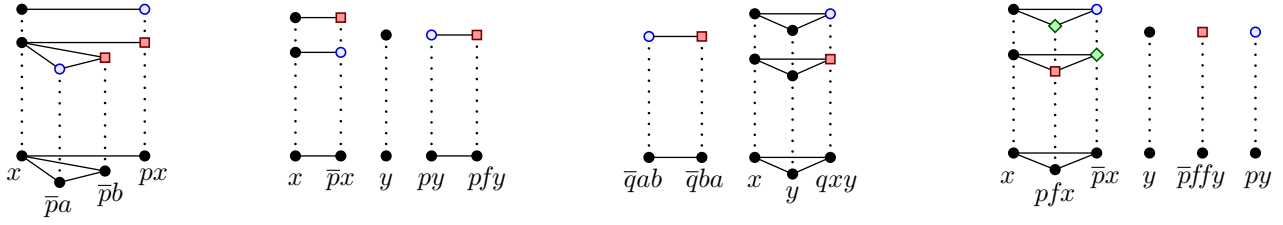
B. Skew Bifibrations

A graph homomorphism $\varphi: \langle V_{\mathcal{G}}, E_{\mathcal{G}} \rangle \rightarrow \langle V_{\mathcal{H}}, E_{\mathcal{H}} \rangle$ is a
fibration if for all $v \in V_{\mathcal{G}}$ and $w\varphi(v) \in E_{\mathcal{H}}$, there exists
 a unique $\tilde{w} \in V_{\mathcal{G}}$ such that $\tilde{w}v \in E_{\mathcal{G}}$ and $\varphi(\tilde{w}) = w$
 (indicated below-left), and is a **skew fibration** if for all
 $v \in V_{\mathcal{G}}$ and $w\varphi(v) \in E_{\mathcal{H}}$ there exists $\tilde{w} \in V_{\mathcal{G}}$ such that
 $\tilde{w}v \in E_{\mathcal{G}}$ and $\varphi(\tilde{w})w \notin E_{\mathcal{H}}$ (indicated below-centre). A
 directed graph homomorphism is a **fibration** if for all $v \in V_{\mathcal{G}}$
 and $(w, \varphi(v)) \in E_{\mathcal{H}}$, there exists a unique $\tilde{w} \in V_{\mathcal{G}}$ such that
 $(\tilde{w}, v) \in E_{\mathcal{G}}$ and $\varphi(\tilde{w}) = w$ (indicated below-right).



A **fograph homomorphism** $\varphi = \langle \varphi, \rho_{\varphi} \rangle$ is a pair where
 $\varphi: \mathcal{G} \rightarrow \mathcal{H}$ is a graph homomorphism between the underlying
 graphs, and ρ_{φ} , also called the **substitution induced by** φ
 is a variable renaming such that for all $v \in V_{\mathcal{G}}$ we have
 $\ell(\varphi(v)) = \rho_{\varphi}(\ell(v))$, and ρ_{φ} is the identity on variables not
 in \mathcal{G} . Note that φ necessarily maps binders to binders and
 literals to literals. Since ρ_{φ} is fully determined by φ alone, we
 often leave ρ_{φ} implicit. A fograph homomorphism $\varphi: \mathcal{G} \rightarrow \mathcal{H}$
preserves existentials if for all existential binders b in \mathcal{G} , the
 binder $\varphi(b)$ is an existential in \mathcal{H} .

Definition 11. Let \mathcal{G} and \mathcal{H} be fographs. A **skew bifibration**
 $\varphi: \mathcal{G} \rightarrow \mathcal{H}$ is an existential-preserving fograph homomor-
 phism that is a skew fibration on $\langle V_{\mathcal{G}}, E_{\mathcal{G}} \rangle \rightarrow \langle V_{\mathcal{H}}, E_{\mathcal{H}} \rangle$ and
 a fibration on the binding graphs $\vec{\mathcal{G}} \rightarrow \vec{\mathcal{H}}$.



$$\exists x (pa \vee pb \Rightarrow px) \quad (\forall x px) \Rightarrow \forall y (py \wedge pfy) \quad qab \vee qba \Rightarrow \exists x \exists y qxy \quad (\forall x (pfx \Rightarrow px)) \Rightarrow \forall y (pffy \Rightarrow py)$$

Fig. 4. Four combinatorial proofs, each shown above the formula proved. Here x and y are variables, f is a unary function symbol, a and b are constants (nullary function symbols), p is a unary predicate symbol, and q is a binary predicate symbol.

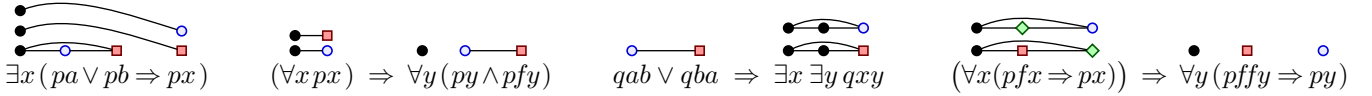
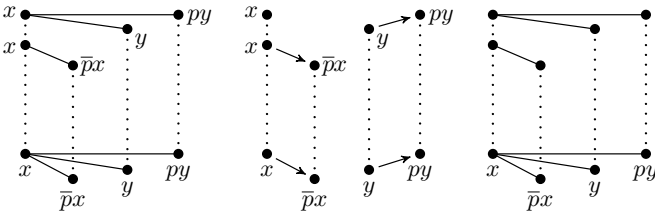


Fig. 5. Condensed forms of the four combinatorial proofs in Fig. 4.

Example 12. Below-left is a skew bifibration, whose binding fibration is below-centre. When the labels on the source fograph can be inferred (modulo renaming), we often omit the labeling in the upper graph, as below-right, which refer to as *condensed form*. [[[Dominic: I said “can be inferred” to account for e.g. $\llbracket \forall x_1 \forall x_2 (px_i \vee px_j \vee px_k \vee px_l) \rrbracket \rightarrow \llbracket \forall x px \rrbracket$ whose unlabelled skeleton is the skeleton of many distinct formula variants (“who binds who?”).]]]]



Definition 13. A *first-order combinatorial proof (FOCP)* of a fograph \mathcal{G} is a skew bifibration $\varphi: \mathcal{C} \rightarrow \mathcal{G}$ where \mathcal{C} is a fonet. A *first-order combinatorial proof* of a formula A is a combinatorial proof of its graph $\llbracket A \rrbracket$.

Figure 4 shows examples of FOCPs (taken from [18]), each above the formula it proves. The same FOCPs are shown in Figure 5 in condensed form.

Theorem 14 ([18]). *FOCPs are sound and complete for first-order logic.*

Remark 15. Our definition of FOCP is slightly more lax than the original definition of [18], as we allow for a variable renaming σ_φ which was forced to be the identity in [18].

V. FIRST-ORDER DEEP INFERENCE SYSTEM KS1

In contrast to standard proof formalisms, like sequent calculi or tableaux, where inference rules decompose the principal formula along its root connective, *deep inference rules* apply like rewriting rules inside any (positive) formula or sequent *context*, which is denoted as $S\{\cdot\}$, and which is

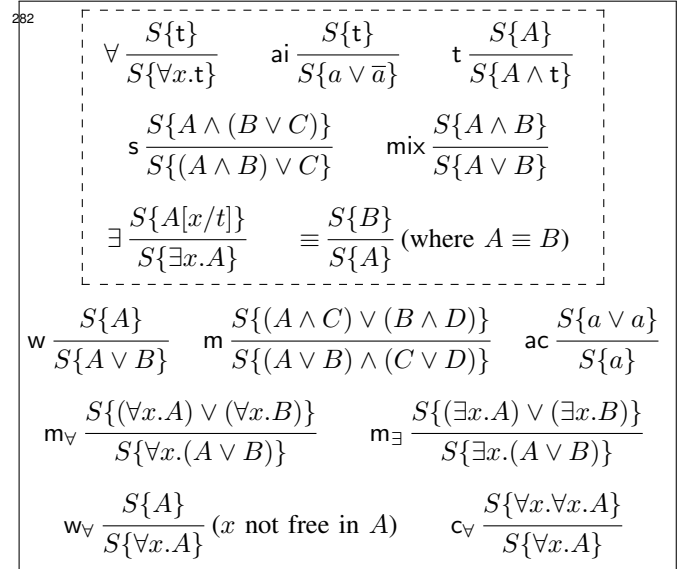


Fig. 6. Deep inference systems KS1 (all rules) and $MLS1^X$ (rules in the dashed box)

a formula (resp. sequent) with exactly one occurrence of the *hole* $\{\cdot\}$ in the position of an atom. Then $S\{A\}$ is the result of replacing the hole $\{\cdot\}$ in $S\{\cdot\}$ with A .

Figure 6 shows the inference rules for the deep inference system KS1 that we introduce in this paper. It is a slight variation of the systems presented by Br  nnler [29] and Ralph [30] in their PhD-theses. The main differences are (i) the explicit presence of the mix-rule, (ii) a different choice of how the formula equivalence \equiv is defined, and (iii) an explicit rule for the equivalence.

We consider here only the cut-free fragment, as cut-elimination for deep inference systems has already been dis-

cussed elsewhere (e.g. [22], [31]).¹

As with the sequent system LK1, we also need for KS1 the *linear fragment*, MLS1^\times , and that is shown in Figure 6 in the dashed box.

B

Write $s \Vdash_\Phi$ to denote a derivation Φ from B to A using the

A

rules from system S . A formula A is **provable** in a system S if there is a derivation in S from t to A .

In the course of this paper we will employ the general (non-atomic) version of the contraction rule:

$$c \frac{S\{A \vee A\}}{S\{A\}}$$

VI. MAIN RESULTS

We state the main results of this paper here, and prove them in later sections. The first is routine and expected, but needs to be proved nonetheless:

Theorem 16. *KS1 is sound and complete for first-order logic.*

Our second result is more surprising, as it is a very strong decomposition result for first order logic.

Theorem 17. *For every derivation $\text{KS1} \Vdash_\Phi$ there are formulas A_1, \dots, A_5 , such that there is a derivation:*

$$\begin{array}{c} t \\ \{\forall, \text{ai}, t\} \Vdash \\ A_5 \\ \{s, \text{mix}, \equiv\} \Vdash \\ A_4 \\ \{\exists\} \Vdash \\ A_3 \\ \{m, m_\forall, m_\exists, \equiv\} \Vdash \\ A_2 \\ \{ac, c_\forall\} \Vdash \\ A_1 \\ \{w, w_\forall, \equiv\} \Vdash \\ A \end{array}$$

This theorem is stronger than the existing decompositions for first-order logic, which either separated only atomic contraction and atomic weakening [29] or only contraction [30] or only the quantifiers in form of a Herbrand theorem [32], [30].

Theorem 17 is also the reason why we have the rule w_\forall and c_\forall in system KS1, as these rules are derivable with the other rules. However, they are needed to obtain this decomposition.

¹In the deep inference literature, the cut-free fragment is also called the *down-fragment*. But as we do not discuss the *up-fragment* here, we omit the down-arrows \downarrow in the rule names.

Example 18. Below is an example of a decomposed derivation in KS1 of the formula $(\exists x. \bar{p}(x)) \vee (\forall y. (p(y) \wedge p(f(y))))$:

$$\begin{array}{c} t \\ \forall y. t \\ t \frac{}{\forall y. (t \wedge t)} \\ \text{ai} \frac{}{\forall y. ((\bar{p}y \vee py) \wedge t)} \\ \text{ai} \frac{}{\forall y. ((\bar{p}y \vee py) \wedge (pfy \vee \bar{p}fy))} \\ \equiv \frac{}{\forall y. (\bar{p}y \vee (py \wedge (pfy \vee \bar{p}fy)))} \\ s \frac{}{\forall y. (\bar{p}y \vee ((py \wedge pfy) \vee \bar{p}fy))} \\ \equiv \frac{}{\forall y. ((\bar{p}y \vee \bar{p}fy) \vee (py \wedge pfy))} \\ \exists \frac{}{\forall y. ((\bar{p}y \vee (\exists x. \bar{p}x)) \vee (py \wedge pfy))} \\ \exists \frac{}{\forall y. (((\exists x. \bar{p}x) \vee (\exists x. \bar{p}x)) \vee (py \wedge pfy))} \\ \equiv \frac{}{((\exists x. \bar{p}x) \vee (\exists x. \bar{p}x)) \vee (\forall y. (py \wedge pfy))} \\ m_\exists \frac{}{(\exists x. (\bar{p}x \vee \bar{p}x) \vee (\forall y. (py \wedge pfy)))} \\ ac \frac{}{(\exists x. \bar{p}x) \vee (\forall y. (py \wedge pfy))} \end{array}$$

There is a weaker version of Theorem 17 that will also be useful:

Theorem 19. *For every derivation $\text{KS1} \Vdash_\Phi$ there is a formula A_1 , such that there is a derivation:*

$$\begin{array}{c} t \\ \text{MLS1}^\times \Vdash \\ A_1 \\ \{w, c, \equiv\} \Vdash \\ A \end{array}$$

Let us now establish the connection between derivations in KS1 and combinatorial proofs.

Theorem 20. *Let $\varphi: \mathcal{C} \rightarrow \mathcal{A}$ be a combinatorial proof and let A be a formula with $\mathcal{A} = \llbracket A \rrbracket$. Then there is a derivation*

$$\begin{array}{c} t \\ \text{MLS1}^\times \Vdash_{\Phi_1} \\ A' \\ \{w, w_\forall, ac, c_\forall, m, m_\forall, m_\exists, \equiv\} \Vdash_{\Phi_2} \\ A \end{array} \quad (3)$$

for some $A' \equiv C\sigma_\varphi$ where C is a formula with $\llbracket C \rrbracket = \mathcal{C}$ and σ_φ is the variable renaming substitution induced by φ . Conversely, whenever we have a derivation as in (9) above, then there is a combinatorial proof $\varphi: \mathcal{C} \rightarrow \llbracket A \rrbracket$ such that $C = \llbracket A' \rrbracket$.

Furthermore, in the proof of Theorem 20, we will see that (i) the links in the fonet \mathcal{C} correspond precisely to the pairs of atoms that meet in the instances of the ai-rule in the derivation Φ_1 , and (ii) the "flow-graph" of Φ_2 that traces the quantifier- and atom-occurrences in the derivation corresponds exactly to the vertex-mapping induced by φ .

Thus, combinatorial proofs are closely related to derivations of the form (9), and since by Theorem 17 every derivation can

be transformed into that form, we can say that combinatorial proofs form a canonical proof representation for first order logic, similarly to what proof nets are for linear logic [33].

Finally, Theorems 16, 17 and 20 imply Theorem 14, which means that we have here an alternative proof of the soundness and completeness of first-order combinatorial proofs which is simpler than the one given in [18].

VII. TRANSLATING BETWEEN LK1 AND KS1

In this section we prove Theorems 16, 17, and 19, mainly by translating derivations to and from the sequent calculus and by rule permutation arguments.

A. The Linear Fragments $MLL1^X$ and $MLS1^X$

In this section we show the equivalence of $MLL1^X$ and $MLS1^X$.

Lemma 21. *If $\vdash \Gamma$ is provable in $MLL1^X$ then $\bigvee(\Gamma)$ is provable in $MLS1^X$.*

Proof. This is a straightforward induction on the proof of $\vdash \Gamma$ in $MLL1^X$, making a case analysis on the bottommost rule instance. We show here only the case of $\forall \frac{\vdash \Delta, A}{\vdash \Delta, \forall x.A}$ (all other cases are simpler and have been shown before, e.g. [29]): By induction hypothesis, there is a proof of $\bigvee(\Delta) \vee A$ in $MLS1^X$. We can prefix every line in that proof by $\forall x$ and then compose the following derivation:

$$\begin{aligned} & \forall \frac{t}{\forall x.t} \\ & \text{MLS1}^X \parallel \\ & \frac{\forall x. \bigvee(\Delta) \vee A}{\bigvee(\Delta) \vee \forall x.A} \end{aligned}$$

where we can apply the \equiv -rule because x is not free in Δ . \square

Lemma 22. *Let $r \frac{S\{A\}}{S\{B\}}$ be an inference rule in $MLS1^X$ other than ai. Then the sequent $\vdash \overline{A}, B$ is provable in $MLL1^X$.*

Proof. This is a straightforward exercise that we leave to the reader. (Note that the ax-rule can be applied to $\vdash f, t$ in the cases of $r = \forall$.) \square

Lemma 23. *Let A, B be formulas, and let $S\{\cdot\}$ be a (positive) context. If $\vdash \overline{A}, B$ is provable in $MLL1^X$, then so is $\vdash \overline{S\{A\}}, S\{B\}$.*

Proof. Straightforward induction on $S\{\cdot\}$. (see e.g. [34]) \square

Lemma 24. *If a formula C is provable in $MLS1^X$ then $\vdash C$ is provable in $MLL1^X$.*

Proof. We proceed by induction on the number of inference steps in the proof of C in $MLS1^X$. Consider the bottommost rule instance $r \frac{S\{A\}}{S\{B\}}$. By induction hypothesis we have a $MLL1^X$ proof Π of $\vdash S\{A\}$. If r is ai $\frac{S\{t\}}{S\{a \vee \overline{a}\}}$, we replace in Π all corresponding occurrences of t with $a \vee \overline{a}$ and the

rule instance $t \frac{}{\vdash t}$ with the derivation $\frac{\text{ax } \vdash a, \overline{a}}{\vdash a \vee \overline{a}}$. This gives

us a proof of $\vdash S\{a \vee \overline{a}\}$. In all other cases, by Lemmas 22 and 23, we have a $MLL1^X$ proof of $\vdash \overline{S\{A\}}, S\{B\}$. We can compose them via cut:

$$\text{cut} \frac{\vdash S\{A\} \quad \vdash \overline{S\{A\}}, S\{B\}}{\vdash S\{B\}}$$

and then apply Theorem 2. \square

B. Contraction and Weakening

The first observation here is that Lemmas 21–24 from above also hold for LK1 and KS1. We therefore immediately have:

Theorem 25. *For every sequent Γ , we have that $\vdash \Gamma$ is provable in LK1 if and only if $\bigvee(\Gamma)$ is provable in KS1.*

Then Theorem 16 is an immediate consequence. Let us now proceed with providing further lemmas that will be needed for the other results.

Lemma 26. *The c-rule is derivable in $\{ac, m, m_\forall, m_\exists, \equiv\}$.*

Proof. We show that there is always a derivation

$$\begin{array}{c} A \vee A \\ s \parallel \\ A \end{array}$$

, where $S = \{ac, m, m_\forall, m_\exists, \equiv\}$, by induction on A :

- If $A = a$, then we have $ac \frac{a \vee a}{a}$.

$$m \frac{(B \wedge C) \vee (B \wedge C)}{(B \vee B) \wedge (C \vee C)}$$

- If $A = B \wedge C$, then we have

$$s \parallel \frac{B \wedge (C \vee C)}{B \wedge C}$$

$$\equiv \frac{(B \vee C) \vee (B \vee C)}{(B \vee B) \vee (C \vee C)}$$

- If $A = B \vee C$, then we have

$$s \parallel \frac{B \vee (C \vee C)}{B \vee C}$$

$$m_\exists \frac{(\exists x.A') \vee (\exists x.A')}{\exists x.(A' \vee A')}$$

- If $A = \exists x.A'$, then we have

$$s \parallel \frac{\exists x.A'}{(\forall x.A') \vee (\forall x.A')}$$

$$m_\forall \frac{(\forall x.A') \vee (\forall x.A')}{\forall x.(A' \vee A')}$$

- If $A = \forall x.A'$, then we have

$$s \parallel \forall x.A'$$

416 **TODO:** **Jui-Hsuan:** done. Maybe just keep one
 417 case. **Lutz:** yes, but we do that at the end. don't think about
 space right now. \square

418 **Lemma 27.** $w_{\forall}, c_{\forall}, m, m_{\forall}, m_{\exists}$ are derivable in $\{w, c, \equiv\}$.

419 *Proof.* **TODO:**

We have the following derivations:

$$\begin{array}{c}
 \frac{w}{\frac{\frac{\frac{\forall x. \forall x. A}{\forall x. ((\forall x. A) \vee A)}{\equiv} (\forall x. A) \vee (\forall x. A)}{c} \forall x. A} \\
 (x \notin fv(\forall x. A)) \\
 \\
 \frac{w}{\frac{\frac{(A \wedge C) \vee (B \wedge D)}{((A \vee B) \wedge C) \vee (B \wedge D)}{w} ((A \vee B) \wedge (C \vee D)) \vee (B \wedge D)}{w} ((A \vee B) \wedge (C \vee D)) \vee ((B \vee A) \wedge D)}{w} ((A \vee B) \wedge (C \vee D)) \vee ((B \vee A) \wedge (D \vee C))}{\equiv} ((A \vee B) \wedge (C \vee D)) \vee ((A \vee B) \wedge (C \vee D))}{c} (A \vee B) \wedge (C \vee D) \\
 \\
 \frac{w}{\frac{\frac{(\exists x. A) \vee (\exists x. B)}{(\exists x. (A \vee B)) \vee (\exists x. B)}{w} (\exists x. (A \vee B)) \vee (\exists x. (B \vee A))}{\equiv} \exists x. (A \vee B) \vee (\exists x. (A \vee B))}{c} \exists x. (A \vee B) \\
 \\
 \frac{w}{\frac{\frac{(\forall x. A) \vee (\forall x. B)}{(\forall x. (A \vee B)) \vee (\forall x. B)}{w} (\forall x. (A \vee B)) \vee (\forall x. (B \vee A))}{\equiv} (\forall x. (A \vee B)) \vee (\forall x. (A \vee B))}{c} \forall x. (A \vee B)
 \end{array}$$

420 **Jui-Hsuan:** done. If needed, we can introduce the notion
 421 of open deduction to reduce the size of derivations... **Lutz:**
 422 I was thinking about that, but (i) it is probably not worth the
 423 effort, as we won't have many derivations, and (ii) it is hard to
 424 define rectified derivations this way. \square

Lemma 28. Let A and B be formulas. Then

$$\frac{A}{\{w, c, \equiv\} \parallel B} \iff \frac{A}{\{w, w_{\forall}, ac, c_{\forall}, m, m_{\forall}, m_{\exists}, \equiv\} \parallel B}$$

425 *Proof.* This follows immediately from Lemmas 26 and 27.

426 \square

427 **C. Rule Permutations**

Theorem 29. Let Γ be a sequent. If $\vdash \Gamma$ is provable in LK1 (as depicted on the left below) then there is a sequent Γ' such that there is a derivation as shown on the right below:

415

$$\text{LK1} \frac{\triangle}{\vdash \Gamma} \Phi \implies \text{MLL1}^x \frac{\triangle}{\vdash \Gamma'} \Phi_1 \quad \frac{\{w, c, \equiv\} \parallel \Phi_2}{\vdash \vee(\Gamma)}$$

428 *Proof.* Note that the instances of w, c in Φ_2 are deep, but inside sequent contexts. 429

First, if an instance of $wk \frac{\vdash \Gamma}{\vdash \Gamma, A}$ is followed by a rule in which A is not in the principal formula, it can be permuted downwards. Otherwise, the proof can be transformed using the following rewriting rules. 430
431
432
433

$$\begin{array}{c}
 \text{wk} \frac{\vdash \Gamma}{\vdash \Gamma, A} \wedge \frac{\vdash \Gamma, A}{\vdash \Gamma, A \wedge B, \Delta} \rightsquigarrow \text{wk} \frac{\vdash \Gamma}{\vdash \Gamma, A \wedge B, \Delta} \\
 \\
 \text{wk} \frac{\vdash \Gamma, A}{\vdash \Gamma, A, B} \vee \frac{\vdash \Gamma, A}{\vdash \Gamma, A \vee B} \rightsquigarrow w \frac{\vdash \Gamma, A}{\vdash \Gamma, A \vee B} \\
 \\
 \text{wk} \frac{\vdash \Gamma}{\vdash \Gamma, A[x/t]} \exists \frac{\vdash \Gamma, A}{\vdash \Gamma, \exists x. A} \rightsquigarrow \text{wk} \frac{\vdash \Gamma}{\vdash \Gamma, \exists x. A} \\
 \\
 \text{wk} \frac{\vdash \Gamma}{\vdash \Gamma, A} \forall \frac{\vdash \Gamma, A}{\vdash \Gamma, \forall x. A} \rightsquigarrow \text{wk} \frac{\vdash \Gamma}{\vdash \Gamma, \forall x. A} \\
 \\
 \text{wk} \frac{\vdash \Gamma, A}{\vdash \Gamma, A, A} \text{ctr} \rightsquigarrow \vdash \Gamma, A
 \end{array}$$

Note that in the case of \vee , we use the deep rule w which can be permuted down over all the rules. By using these rewriting rules, we can eventually get a derivation with all the instances of wk and w at the bottom. Now observe that the instances of ctr in Φ can be transformed using the following rule: 434
435
436
437
438

$$\text{ctr} \frac{\vdash \Gamma, A, A}{\vdash \Gamma, A} \rightsquigarrow \frac{\vdash \Gamma, A, A}{c \frac{\vdash \Gamma, A \vee A}{\vdash \Gamma, A}}$$

Knowing that c can be permuted down over all the rules of MLL1^x , we eventually obtain a derivation:

$$\text{MLL1}^x \frac{\triangle}{\vdash \Gamma_0} \Phi'_1 \quad \frac{\{wk, w, c, \equiv\} \parallel \Phi'_2}{\vdash \Gamma}$$

Note that \equiv is required here since the permutation of formulas is implicit in MLL1^x . 439
440

By transforming each sequent of Φ'_2 into its corresponding formula, and by considering the following rewriting rule:

$$\text{wk} \frac{\vdash \Gamma}{\vdash \Gamma, A} \rightsquigarrow \text{w} \frac{\vdash \bigvee(\Gamma)}{\vdash \bigvee(\Gamma) \vee A}$$

, we obtain a derivation

$$\begin{array}{c} \text{MLL1}^x \quad \triangle \quad \Phi_1 \\ \vdash \Gamma' \\ \{w, c, \equiv\} \parallel \Phi_2 \\ \vdash \bigvee(\Gamma) \end{array}$$

where $\Gamma' = \bigvee(\Gamma_0)$ and Φ_1 can be obtained from Φ'_1 by applying the \vee rule. **TO CHECK:** **Jui-Hsuan:** This might be a bit long... \square

Lemma 30. For every derivation $\text{MLS1}^x \parallel \frac{t}{A}$ there are formulas A' and A'' such that

$$\begin{array}{c} t \\ \{v, ai, t\} \parallel \\ A'' \\ \{s, mix, \equiv\} \parallel \\ A' \\ \{\exists\} \parallel \\ A \end{array}$$

Proof. First, observe that the \exists rule can be permuted downwards over all the other rules since $A[x/t]$ has the same structure as A and none of the other rules has a premise of the form $S\{\exists x.A\}$. It suffices now to prove that for all $r_1 \in \{v, ai, t\}$, for all $r_2 \in \{s, mix, \equiv\}$, we can permute r_1 upwards over r_2 . We show some cases here, and leave the others to the reader.

$$\begin{array}{c} \frac{S\{A \wedge (t \vee C)\}}{S\{(A \wedge t) \vee C\}} \xrightarrow{s} \frac{S\{A \wedge (t \vee C)\}}{S\{(A \wedge (a \vee \bar{a})) \vee C\}} \xrightarrow{ai} \frac{S\{A \wedge (t \vee C)\}}{S\{(A \wedge (a \vee \bar{a})) \vee C\}} \\ \frac{S\{A \wedge B\}}{S\{A \vee B\}} \xrightarrow{mix} \frac{S\{A \wedge B\}}{S\{(A \vee (B \wedge t))\}} \xrightarrow{t} \frac{S\{A \wedge B\}}{S\{(A \vee (B \wedge t))\}} \end{array}$$

TO CHECK: \square

Lemma 31. For every derivation $\{w, ac, c_v, m, m_v, m_\exists, \equiv\} \parallel \frac{A}{B}$ there are formulas A' and B' such that

$$\begin{array}{c} A \\ \{m, m_v, m_\exists, \equiv\} \parallel \\ A' \\ \{ac, c_v\} \parallel \\ B' \\ \{w, \equiv\} \parallel \\ B \end{array}$$

Proof. First, a derivation consisting of an instance of w followed by $r \in \{ac, c_v, m, m_v, m_\exists\}$ can be either replaced by a derivation consisting of w only or the instance of w can be permuted downwards. We show some cases here, and leave the others to the reader.

$$\begin{array}{c} \frac{w}{m_v} \frac{S\{\forall x.A\}}{S\{(\forall x.A) \vee (\forall x.B)\}} \rightsquigarrow \frac{w}{m_v} \frac{S\{\forall x.A\}}{S\{\forall x.(A \vee B)\}} \\ \frac{w}{m} \frac{S\{A \wedge C\}}{S\{(A \wedge C) \vee (B \wedge D)\}} \rightsquigarrow \frac{w}{m} \frac{S\{A \wedge C\}}{S\{(A \vee B) \wedge (C \vee D)\}} \\ \frac{w}{ac} \frac{S\{a\}}{S\{a \vee a\}} \rightsquigarrow S\{a\} \end{array}$$

For $r_1 \in \{m, m_v, m_\exists\}$, $r_2 \in \{ac, c_v\}$, r_1 can be permuted upwards over r_2 (with some \equiv inserted). The only non-trivial case is shown below:

$$\frac{c_v}{m_v} \frac{S\{(\forall x.\forall x.A) \vee (\forall x.B)\}}{S\{(\forall x.A) \vee (\forall x.B)\}} \rightsquigarrow \frac{m_v}{m_v} \frac{S\{(\forall x.\forall x.A) \vee (\forall x.B)\}}{S\{(\forall x.A) \vee (\forall x.B)\}} \rightsquigarrow \frac{m_v}{m_v} \frac{S\{(\forall x.\forall x.A) \vee (\forall x.B)\}}{S\{\forall x.(A \vee B)\}}$$

TODO: permutation with \equiv (c_v / \equiv)

$$\begin{array}{c} \frac{c_v}{\equiv} \frac{\forall x.\forall x.\forall y.A}{\forall y.\forall x.A} \rightsquigarrow \frac{\equiv}{c_v} \frac{\forall x.\forall x.\forall y.A}{\forall y.\forall x.A} \\ \frac{c_v}{\equiv} \frac{\forall x.\forall x.(A \vee B)}{\forall x.(A \vee B)} \rightsquigarrow \frac{\equiv}{c_v} \frac{\forall x.\forall x.(A \vee B)}{(\forall x.A) \vee B} (x \notin fv(B)) \\ \frac{c_v}{\equiv} \frac{(\forall x.\forall x.A) \vee B}{(\forall x.A) \vee B} \rightsquigarrow \frac{\equiv}{c_v} \frac{(\forall x.\forall x.A) \vee B}{\forall x.(A \vee B)} (x \notin fv(B)) \end{array}$$

(w / \equiv)

$$\frac{w}{\equiv} \frac{A}{A \vee B} \rightsquigarrow \frac{w}{\equiv} \frac{A}{B \vee A}$$

$$\frac{w}{\equiv} \frac{A \vee C}{(A \vee B) \vee C} \rightsquigarrow \frac{w}{\equiv} \frac{A \vee C}{A \vee (B \vee C)}$$

$$\frac{w}{\equiv} \frac{\forall x.A}{\forall x.(A \vee B)} \rightsquigarrow \frac{w}{\equiv} \frac{\forall x.A}{(\forall x.A) \vee B} (x \notin fv(B))$$

$$\frac{w \frac{\forall x.A}{(\forall x.A) \vee B}}{\forall x.(A \vee B)} (x \notin fv(B)) \rightsquigarrow w \frac{\forall x.A}{\forall x.(A \vee B)}$$

 (\equiv / c_{\forall})

$$\begin{aligned} & \frac{\forall x.\forall y.\forall x.A}{\forall x.\forall x.\forall y.A} \\ & \frac{c_{\forall}}{\forall x.\forall y.A} \end{aligned}$$

$$\begin{aligned} & \frac{\forall x.\forall y.\forall x.A}{\forall y.\forall x.\forall x.A} \\ & \frac{c_{\forall}}{\forall y.\forall x.A} \end{aligned}$$

$$\begin{aligned} & \frac{\forall x.((\forall x.A) \vee B)}{(\forall x.\forall x.A) \vee B} (x \notin fv(B)) \\ & \frac{c_{\forall}}{(\forall x.A) \vee B} \end{aligned}$$

$$\begin{aligned} & \frac{\forall x.((\forall x.A) \vee B)}{\forall x.\forall x.(A \vee B)} (x \notin fv(B)) \\ & \frac{c_{\forall}}{\forall x.(A \vee B)} \end{aligned}$$

449

 (\equiv / m)

$$\begin{aligned} & \frac{(C \wedge A) \vee (B \wedge D)}{(A \wedge C) \vee (B \wedge D)} \\ & \frac{m}{(A \vee B) \wedge (C \vee D)} \end{aligned}$$

$$\begin{aligned} & \frac{(B \wedge D) \vee (A \wedge C)}{(A \wedge C) \vee (B \wedge D)} \rightsquigarrow \frac{m}{(A \vee B) \wedge (C \vee D)} \frac{(B \wedge D) \vee (A \wedge C)}{(B \vee A) \wedge (D \vee C)} \\ & \frac{m}{(A \vee B) \wedge (C \vee D)} \end{aligned}$$

$$\begin{aligned} & \frac{((A \wedge C) \wedge E) \vee (B \wedge D)}{(A \wedge (C \wedge E)) \vee (B \wedge D)} \\ & \frac{m}{(A \vee B) \wedge ((C \wedge E) \vee D)} \end{aligned}$$

$$\begin{aligned} & \frac{(forallx.(A \wedge C)) \vee (B \wedge D)}{\forall x.((A \wedge C) \vee (B \wedge D))} (x \notin fv(B \wedge D)) \\ & \frac{m}{\forall x.((A \vee B) \wedge (C \vee D))} \end{aligned}$$

450

 (\equiv / m_{\forall})

$$\begin{aligned} & \frac{(\forall x.B) \vee (\forall x.A)}{(\forall x.A) \vee (\forall x.B)} \rightsquigarrow \frac{m_{\forall}}{\forall x.(B \vee A)} \frac{(\forall x.B) \vee (\forall x.A)}{\forall x.(A \vee B)} \\ & \frac{m_{\forall}}{\forall x.(A \vee B)} \end{aligned}$$

$$\begin{aligned} & \frac{(\forall y.\forall x.A) \vee (\forall x.B)}{(\forall x.\forall y.A) \vee (\forall x.B)} \\ & \frac{m_{\forall}}{\forall x.((\forall y.A) \vee B)} \end{aligned}$$

$$\begin{aligned} & \frac{\forall x.(A \vee (\forall x.B))}{(\forall x.A) \vee (\forall x.B)} \\ & \frac{m_{\forall}}{\forall x.(A \vee B)} \end{aligned}$$

We can now complete the proof of Theorems 17 and 19. □ 453

Proof of Theorem 19. Assume we have a proof of A in KS1. 454
 By Theorem 25 we have a proof of $\vdash A$ in LK1 to which we 455
 can apply Theorem 29. Finally, we apply Lemma 21 to get 456
 the desired shape. □ 457

Proof of Theorem 17. Assume we have a proof of A in KS1. 458
 We first apply Theorem 19, and then Lemma 30 to the upper 459
 half and Lemma 31 to the lower half. □ 460

VIII. FONETS AND LINEAR PROOFS 461

A. From MLL1^X Proofs to Fonets 462

Let Π be a MLL1^X proof of a rectified sequent $\vdash \Gamma$. We 463
 now show how Π is translated into a linked fograph $\llbracket \Pi \rrbracket =$ 464
 $\langle \llbracket \Gamma \rrbracket, \sim_{\Pi} \rangle$. We proceed inductively, making a case analysis on 465
 the last rule in Π . At the same time we are constructing a 466
 dualizer δ_{Π} , so that in the end we can conclude that $\llbracket \Pi \rrbracket$ is in 467
 fact a fonet. 468

1) Π is $\text{ax} \frac{}{\vdash a, \bar{a}}$: Then the only link is $\{a, \bar{a}\}$, and δ_{Π} is 469
 empty. 470

2) Π is $\text{t} \frac{}{\vdash \text{t}}$: Then \sim_{Π} and δ_{Π} are both empty. 471

3) The last rule in Π is $\text{mix} \frac{\vdash \Gamma' \quad \vdash \Gamma''}{\vdash \Gamma', \Gamma''}$: By induction 472
 hypothesis, we have proofs Π' and Π'' of Γ' and Γ'' , 473
 respectively. We have $\llbracket \Gamma \rrbracket = \llbracket \Gamma' \rrbracket + \llbracket \Gamma'' \rrbracket$ and let 474

$$\sim_{\Pi} = \sim_{\Pi'} \cup \sim_{\Pi''} \quad \text{and} \quad \delta_{\Pi} = \delta_{\Pi'} \cup \delta_{\Pi''}$$

4) The last rule in Π is $\vee \frac{\vdash \Gamma_1, A, B}{\vdash \Gamma_1, A \vee B}$: By induction 472
 hypothesis, there is proofs Π' of $\Gamma' = \Gamma_1, A, B$. We 473
 have $\llbracket \Gamma \rrbracket = \llbracket \Gamma' \rrbracket$ and let $\sim_{\Pi} = \sim_{\Pi'}$ and $\delta_{\Pi} = \delta_{\Pi'}$. 474

5) The last rule in Π is $\wedge \frac{\vdash \Gamma_1, A \quad \vdash \Gamma_2, B}{\vdash \Gamma_1, A \wedge B, \Gamma_2}$: By induction 475
 hypothesis, we have proofs Π' and Π'' of $\Gamma' = \Gamma_1, A$ 476
 and $\Gamma'' = \Gamma_2, B$, respectively. We have $\llbracket \Gamma \rrbracket = \llbracket \Gamma_1 \rrbracket +$ 477
 $(\llbracket A \rrbracket \times \llbracket B \rrbracket) + \llbracket \Gamma_2 \rrbracket$ and we let 478

$$\sim_{\Pi} = \sim_{\Pi'} \cup \sim_{\Pi''} \quad \text{and} \quad \delta_{\Pi} = \delta_{\Pi'} \cup \delta_{\Pi''}$$

6) The last rule in Π is $\exists \frac{\vdash \Gamma_1, A[x/t]}{\vdash \Gamma_1, \exists x.A}$: By induction 475
 hypothesis, there is a Π' of $\Gamma' = \Gamma_1, A[x/t]$. For each 476
 atom in $\Gamma' = \Gamma_1, A[x/t]$, there is a corresponding atom 477
 in $\Gamma = \Gamma_1, \exists x.A$. We can therefore define the linking \sim_{Π} 478
 from the linking $\sim_{\Pi'}$ via this correspondence. Then, we 479
 let δ_{Π} be $\delta_{\Pi'} + [x/t]$. Since Γ is rectified x does not yet 480
 occur in $\delta_{\Pi'}$. Hence δ_{Π} is a dualizer of $\llbracket \Pi \rrbracket$. 481

7) The last rule in Π is $\forall \frac{\vdash \Gamma_1, A}{\vdash \Gamma_1, \forall x.A}$ (x not free in Γ_1): 482
 By induction hypothesis, there is a proof Π' of $\Gamma' =$ 483
 Γ_1, A , which has the same atoms as in $\Gamma = \Gamma_1, \forall x.A$. 484
 Hence, we can let $\sim_{\Pi} = \sim_{\Pi'}$ and $\delta_{\Pi} = \delta_{\Pi'}$. 485

Theorem 32. If Π is a MLL1^\times proof of a rectified sequent $\vdash \Gamma$, then $\llbracket \Pi \rrbracket$ is a fonet and δ_Π is a dualizer for it.

Proof. We have to show that none of the operations above can introduce a bmatching. For cases 1–6, this is immediate. For case 7, observe that there is a potential dependency from each existential binder in $\llbracket \Gamma' \rrbracket$ to the new x -binder $\bullet x$ in $\llbracket \Gamma \rrbracket$. However, observe that this $\bullet x$ vertex is not connected to any vertex in $\llbracket \Gamma' \rrbracket$, and hence no such new dependency can be extended to a bmatching. That δ_Π is a dualizer for $\llbracket \Pi \rrbracket$ follows immediately from the construction. Hence, $\llbracket \Pi \rrbracket$ is a fonet. \square

B. From MLS1^\times Proofs to Fonets

There is a more direct path from a MLL1^\times proof Π of a rectified sequent Γ to the linked fograph $\llbracket \Pi \rrbracket$: simply take the fograph $\llbracket \Gamma \rrbracket$, and let the equivalence classes of \sim_Π be all the atom pairs that meet in an instance of ax , and δ_Π is simply the collection of all substitutions of all the instances of the \exists -rule in Π . We have chosen the more cumbersome path above because it gives us a direct proof of Theorem 32. However, for translating MLS1^\times derivation into fonets, we employ exactly that direct path.

A derivation Φ in MLS1^\times is **rectified** if every line in Φ is rectified.

Lemma 33. Let Φ be a MLS1^\times proof of a formula A . Then Φ is rectified iff A is rectified.

Proof. The only rules involving bound variables are \forall and \exists which both remove a binder (and all occurrences of the variable it binds). \square

Hence, for a non-rectified MLS1^\times derivation Φ in MLS1^\times we can define its **rectification** $\hat{\Phi}$ inductively, by rectifying each line, proceeding step-wise from conclusion to premise.²

A rectified derivation $\text{MLS1}^\times \upharpoonright_\Phi^t$ determines a substitution A which maps the existential bound variables occurring in A to the terms substituted for them in the instances of the \exists -rule in Φ . We denote this substitution with δ_Φ and call it the **dualizer** of Φ . Furthermore, every atom occurring in the conclusion A must be consumed by a unique instance of the rule ai in Φ . This allows us to define a (partial) equivalence relation \sim_Φ on the atom occurrences in A by $a \sim_\Phi b$ if a and b are consumed by the same instance of ai in Φ . We call \sim_Φ the **linking** of Φ , and define $\llbracket \Phi \rrbracket = \langle \llbracket A \rrbracket, \sim_\Phi \rangle$.

!!!TODO: example here!!!

Theorem 34. Let $\text{MLS1}^\times \upharpoonright_\Phi^t$ be a rectified derivation. Then $\llbracket \Phi \rrbracket$ is a fonet and δ_Φ a dualizer for it.

For proving this theorem, we have to show that no inference rule in MLS1^\times can introduce a bmatching. To simplify the

²As for formulas, the rectification of a derivation is unique up to renaming of bound variables.

argument, we introduce the **frame** [35] of the fograph \mathcal{C} , which is a linked (propositional) cograph in which the dependencies between the binders in \mathcal{C} are encoded as links.

More formally, let C be a formula with $\llbracket C \rrbracket = \mathcal{C}$, to which we exhaustively apply the following subformula rewriting steps, to obtain a sequent C^* :

- 1) **Encode dependencies as fresh links.** For each dependency $\{\bullet x_i, \bullet y_j\}$ in \mathcal{C} , with corresponding subformulas $\exists x_i.A$ and $\forall y_j.B$ in C , we pick a fresh (nullary) predicate symbol $q_{i,j}$, and then replace $\exists x_i.A$ by $\bar{q}_{i,j} \wedge \exists x_i.A$, and replace $\forall y_j.B$ by $q_{i,j} \vee \forall y_j.B$.
- 2) **Erase quantifiers.** After step 1, remove all the quantifiers, i.e., replace $\exists x_i.A$ by A and replace $\forall y_j.B$ by B everywhere.
- 3) **Simplify atoms.** After step 2, replace every predicate $p(t_1 \dots t_n)$ (resp. $\bar{p}(t_1 \dots t_n)$) with a nullary predicate symbol p (resp. \bar{p}).

The \sim_{C^*} consists of the pairs induced by $\sim_{\mathcal{C}}$ and the new pairs $\{q_{i,j}, \bar{q}_{i,j}\}$ introduced in step 1 above. We call C^* the **frame** of C and we define the **frame** of \mathcal{C} , denoted C^* , as $\langle \llbracket C^* \rrbracket, \sim_{C^*} \rangle$.

Lemma 35. A linked fograph \mathcal{C} has an induced bmatching iff its frame C^* has an induced bmatching.

Proof. This immediately follows from the construction of the frame. !!!Lutz: is it really an “iff”? It is easy to construct from a bmatching in \mathcal{C} a bmatching in the frame. (and I think we only need that direction). But what about the other direction?!!! \square

Proof of Theorem 34. From Φ we construct a derivation Φ^* of A^* in the propositional fragment of MLS1^\times , such that $\llbracket \Phi^* \rrbracket = \llbracket \Phi \rrbracket^*$. The rules ai , t , mix and s are translated trivially, and for \equiv , it suffices to observe that the frame construction is invariant under \equiv . Finally, for the rules \forall and \exists , proceed as follows. Every instance of \forall is replaced by the derivation on the right below:³

$$\forall \frac{S\{t\}}{S\{\forall y_j.t\}} \rightsquigarrow \frac{\text{t} \quad \{ \text{ai}, t \} \parallel \Psi_1}{S\{(q_{h_1,j} \vee \bar{q}_{h_1,j}) \wedge \dots \wedge (q_{h_j,j} \vee \bar{q}_{h_j,j}) \wedge t\}} \quad \{ \text{s}, \equiv \} \parallel \Psi_2 \\ S\{q_{h_1,j} \vee \dots \vee q_{h_j,j} \vee (\bar{q}_{h_1,j} \wedge \dots \wedge \bar{q}_{h_j,j} \wedge t)\}$$

where h_1, \dots, h_j range over the indices of the existential binders dependent on that y_j . It is easy to see how Ψ_1 is constructed, and for Ψ_2 see, e.g. [?], [34], [36] !!!Lutz: check if it is really there. otherwise [37]!!! Then, every occurrence of $\forall y_j.F$ is replaced by $q_{h_1,j} \vee \dots \vee q_{h_j,j} \vee (\bar{q}_{h_1,j} \wedge \dots \wedge \bar{q}_{h_j,j} \wedge F)$ in the derivation below that \forall -instance. Now, observe that all instances of the \exists -rule introducing x_i depend on y_j must occur below in the derivation (otherwise Φ would not be rectified). Now consider such an instance $\exists \frac{S\{B[x_i/t]\}}{S\{\exists x_i.B\}}$. Its

³For better readability we omit superfluous parentheses, knowing that we always have \equiv incorporating associativity and commutativity of \wedge and \vee .

context $S\{\cdot\}$ must contain all the $\forall y_j$ the $\exists x_i$ depends on, such that B is in their scope. Following the translation of the \forall rules above, we can therefore translate the \exists -rule instance by the following derivation

$$\begin{aligned} & S_0\{\bar{q}_{i,k_1} \wedge S_1\{\bar{q}_{i,k_2} \wedge \cdots S_{k_i-1}\{\bar{q}_{i,k_i} \wedge S_{k_i}\{B'\}\}\cdots\}\} \\ & \quad \quad \quad \{s, \equiv\} \parallel \Psi_3 \\ & S_0\{S_1\{\cdots S_{k_i-1}\{S_{k_i}\{\bar{q}_{i,k_1} \wedge \bar{q}_{i,k_2} \wedge \cdots q_{i,k_i} \wedge B'\}\}\cdots\}\} \end{aligned}$$

where k_1, \dots, k_i are the indices of the universal binders on which that x_i depends, and B' is B in which all predicates are replaced by nullary one (step 3 in the frame construction). The derivation Ψ_3 can be constructed in the same way as Ψ_2 above.

Doing this to all instances of the rules \forall and \exists in Φ yields indeed a propositional derivation Φ^* with $\llbracket \Phi^* \rrbracket = \llbracket \Phi \rrbracket^*$. It has been shown by Retoré [?] and rediscovered by Straßburger [37] that $\llbracket \Phi^* \rrbracket = \langle \llbracket C^* \rrbracket, \sim_{\Phi^*} \rangle$ can not contain an induced bimatching. By Lemma 37, $\llbracket \Phi \rrbracket$ does not have an induced bimatching either. Furthermore, it followed from the definition of δ_Φ that it is a dualizer for $\llbracket \Phi \rrbracket$. Hence $\llbracket \Phi \rrbracket$ is a fonet. \square

Remark 36. There is an alternative path of proving Theorem 34 by translating Φ to an MLL1^\times -proof Π , observing that this process preserves the linking and the dualizer. However, for this, we have to extend the construction above to the cut-rule, and then show that linking and dualizer of a sequent proof Π are invariant under cut elimination. This can be done similarly to unification nets in [35].

C. From Fonets to MLL1^\times Proofs

Now we are going to show how from a given fonet $\langle \mathcal{C}, \sim_{\mathcal{C}} \rangle$ we can construct a sequent proof Π in MLL1^\times such that $\llbracket \Pi \rrbracket = \langle \mathcal{C}, \sim_{\mathcal{C}} \rangle$. In the proof net literature, this operation is also called *sequentialization*. The basic idea behind our sequentialization is to construct a propositional linked cograph, called the **frame** [35] of \mathcal{C} , in which the dependencies between the binders in \mathcal{C} are encoded as links. Then we can apply the *splitting tensor theorem* to the frame, and then reconstruct the sequent proof Π . **[[Lutz: if the proof of thm 34 is verified, we can delete the frame-def here]]**

More formally, let Γ be a sequent with $\llbracket \Gamma \rrbracket = \mathcal{C}$, to which we exhaustively apply the following subformula rewriting steps, to obtain a sequent Γ^* :

- 1) **Encode dependencies as fresh links.** For each dependency $(\bullet x, \bullet y)$ in \mathcal{C} , with corresponding subformulas $\exists x A$ and $\forall y B$ in Γ , we pick a fresh (nullary) predicate symbol q , and then replace $\exists x A$ by $q \wedge \exists x A$, and replace $\forall y B$ by $\bar{q} \vee \forall y B$.
- 2) **Erase quantifiers.** After step 1, remove all the quantifiers, i.e., replace $\exists x A$ by A and replace $\forall y B$ by B everywhere.
- 3) **Simplify atoms.** After step 2, replace every predicate $p(t_1 \cdots t_n)$ (resp. $\bar{p}(t_1 \cdots t_n)$) with a nullary predicate symbol p (resp. \bar{p})

The \sim_{Γ^*} consists of the pairs induced by $\sim_{\mathcal{C}}$ and the new pairs $\{q, \bar{q}\}$ introduced in step 1 above. We call Γ^* the **frame** of Γ and we define the **frame** of \mathcal{C} , denoted \mathcal{C}^* , as $\langle \llbracket \Gamma^* \rrbracket, \sim_{\Gamma^*} \rangle$, and we immediately have the following:

Lemma 37. *A linked fograph \mathcal{C} induces a bimatching iff its frame \mathcal{C}^* has an induced bimatching.*

Let Γ be a propositional sequent and \sim_Γ be a linking for $\llbracket \Gamma \rrbracket$. A conjunction formula $A \wedge B$ is **splitting** or a **splitting tensor** if $\Gamma = \Gamma', A \wedge B, \Gamma''$ and $\sim_\Gamma = \sim_1 \cup \sim_2$, such that \sim_1 is a linking for $\llbracket \Gamma', A \rrbracket$ and \sim_2 is a linking for $\llbracket B, \Gamma'' \rrbracket$, i.e., removing the \wedge from $A \wedge B$ splits the linked fograph $\langle \llbracket \Gamma \rrbracket, \sim_\Gamma \rangle$ into two fographs. We say that $\langle \llbracket \Gamma \rrbracket, \sim_\Gamma \rangle$ is **mixed** iff $\Gamma = \Gamma', \Gamma''$ and $\sim_\Gamma = \sim_1 \cup \sim_2$, such that \sim_1 is a linking for $\llbracket \Gamma' \rrbracket$ and \sim_2 is a linking for $\llbracket \Gamma'' \rrbracket$. Finally, $\langle \llbracket \Gamma \rrbracket, \sim_\Gamma \rangle$ is **splittable** if it is mixed or has a splitting tensor.

The purpose of introducing the frame is the following theorem.

Theorem 38. *Let Γ be a propositional sequent containing only atoms and \wedge -formulas, and \sim_Γ be a linking for $\llbracket \Gamma \rrbracket$. If $\langle \llbracket \Gamma \rrbracket, \sim_\Gamma \rangle$ does not induce a bimatching then it is splittable.*

This is the well-know splitting-tensor-theorem [19], [38], adapted for the presence of mix. In the setting of linked cographs, it has first been proved by Retoré [39], [40]. We use it now for our sequentialization:

Theorem 39. *Let $\langle \mathcal{C}, \sim_{\mathcal{C}} \rangle$ be a fonet, and let Γ be a sequent with $\llbracket \Gamma \rrbracket = \mathcal{C}$. Then there is an MLL1^\times -proof Π of Γ , such that $\llbracket \Pi \rrbracket = \langle \mathcal{C}, \sim_{\mathcal{C}} \rangle$.*

Proof. Let $\delta_{\mathcal{C}}$ be the dualizer of $\langle \mathcal{C}, \sim_{\mathcal{C}} \rangle$. We proceed by induction on the size of Γ (i.e., the number of symbols in it, without counting the commas). If Γ contains a formula with \vee -root, or a formula $\forall x.A$, we can immediately apply the \vee -rule or the \forall -rule of MLL1^\times and proceed by induction hypothesis. If Γ contains a formula $\exists x.A$ such that the corresponding binder $\bullet x$ in \mathcal{C} has no dependency, then we can apply the \exists -rule, choosing the term t as determined by $\delta_{\mathcal{C}}$, and proceed by induction hypothesis. Hence, we can now assume that Γ contains only atoms, \wedge -formulas, or formulas of shape $\exists x.A$, where the vertex $\bullet x$ has dependencies. Then the frame $\langle \llbracket \Gamma^* \rrbracket, \sim_{\Gamma^*} \rangle$ does not induce a bimatching and contains only atoms and \wedge -formulas, and is therefore splittable. If it is mixed, then we can apply the mix-rule to Γ and apply the induction hypothesis to the two components. If it is not mixed then there must be a splitting tensor. If the splitting \wedge is already in Γ , then we can apply the \wedge -rule and proceed by induction hypothesis on the two branches. However, if Γ^* is not mixed and all splitting tensors are \wedge -formulas introduced in step 1 of the frame construction, then we get a contradiction as in that case there must be a \vee - or \forall -formula in Γ . **[[Lutz: can anyone give a good argument here?]]** \square

D. From Fonets to MLS1^\times Proofs

We can now straightforwardly obtain the same result for MLS1^\times :

Theorem 40. Let $\langle \mathcal{C}, \sim_{\mathcal{C}} \rangle$ be a fonet, and let C be a formula

with $\llbracket C \rrbracket = \mathcal{C}$. Then there is a derivation $\text{MLS1}^X \Vdash_{\Phi}^t C$ such that

$\llbracket \Phi \rrbracket = \langle \mathcal{C}, \sim_{\mathcal{C}} \rangle$.

Proof. We apply Theorem 39 to obtain a sequent proof Π of $\vdash C$ with $\llbracket \Pi \rrbracket = \langle \mathcal{C}, \sim_{\mathcal{C}} \rangle$. Then we apply Lemma 21, observing that the translation from MLL1^X to MLS1^X preserves linking and dualizer. \square

Remark 41. Note that it is also possible to do a direct “sequentialization” into the deep inference system MLS1^X , using the techniques presented in [37] and [41].

IX. SKEW BIFIBRATIONS AND RESOURCE MANAGEMENT

In this section we establish the relation between skew bifibrations and derivations in $\{w, ac, c_{\forall}, m, m_{\forall}, m_{\exists}, \equiv\}$. However, if a derivation Φ contains instances of the rules c_{\forall} , m_{\forall} , and m_{\exists} we can no longer naively define the rectification $\hat{\Phi}$ as in the previous section for MLS1^X , as these two rules cannot be applied if premise and conclusion are rectified. For this reason we define here rectified versions \hat{c}_{\forall} , \hat{m}_{\forall} and \hat{m}_{\exists} , shown below:

$$\hat{c}_{\forall} \frac{S\{\forall y. \forall x. Ax\}}{S\{\forall x. Ax\}} \quad \hat{m}_{\forall} \frac{S\{(\forall y. Ay) \vee (\forall z. Bz)\}}{S\{\forall x. (Ax \vee Bx)\}} \quad \hat{m}_{\exists} \frac{S\{(\exists y. Ay) \vee (\exists z. Bz)\}}{S\{\exists x. (Ax \vee Bx)\}}$$

Here, we use the notation $A \cdot$ for a formula A with occurrences of a placeholder \cdot for a variable. Then Ax stands for the results of replacing that placeholder with x , and also indicating that x must not occur in $A \cdot$. Then $\forall x. Ax$ and $\forall y. Ay$ are the same formula modulo renaming of the bound variable bound by the outermost \forall -quantifier. We also demand that the variables x , y , and z do not occur in the context $S\{\cdot\}$.

Note that in an instance of \hat{m}_{\forall} or \hat{m}_{\exists} (as shown above), we can have $x = y$ or $x = z$, but not both if the premise is rectified. If $x = y$ and $x = z$ we have m_{\forall} and m_{\exists} as special cases of \hat{m}_{\forall} and \hat{m}_{\exists} , respectively. And similarly, if $x = y$ then c_{\forall} is a special case of \hat{c}_{\forall} .

For a derivation Φ in $\{w, ac, c_{\forall}, m, m_{\forall}, m_{\exists}, \equiv\}$, we can now construct the **rectification** $\hat{\Phi}$ by rectifying each line of Φ , yielding a derivation in $\{w, ac, m, \hat{m}_{\forall}, \hat{m}_{\exists}, \equiv\}$.

For each instance $r \frac{Q}{P}$ of an inference rule in $\{w, ac, \hat{c}_{\forall}, m, \hat{m}_{\forall}, \hat{m}_{\exists}, \equiv\}$ we can define the **induced map** $\llbracket r \rrbracket: V_{\llbracket Q \rrbracket} \rightarrow V_{\llbracket P \rrbracket}$ which acts as the identity for $r \in \{m, \equiv\}$ and as the canonical injection for $r = w$. For $r = ac$ it maps the vertices corresponding to the two atoms in the premise to vertex of the contracted atom in the conclusion, and for $r \in \{\hat{c}_{\forall}, \hat{m}_{\forall}, \hat{m}_{\exists}\}$ it maps the two vertices corresponding to the quantifiers in the premise to the one in the conclusion (as acts as the identity on all other vertices). For a derivation Φ in $\{w, ac, \hat{c}_{\forall}, m, \hat{m}_{\forall}, \hat{m}_{\exists}, \equiv\}$ we can then define the **induced map** $\llbracket \Phi \rrbracket$ as the composition of the induced maps of the rule instances in Φ . **Jui-Hsuan:** maybe mention at least

that the induced maps define graph homomorphisms. Do we need to talk about the contexts $S\{\cdot\}$ here (induced maps act clearly as the identity on contexts but we need them for the composition)? **Lutz:** For the context, I already say it is the identity. For the homom, it comes later

Lemma 42. Let $\{w, ac, c_{\forall}, m, m_{\forall}, m_{\exists}, \equiv\} \Vdash_{\Phi}^A B$ be a derivation. Then there is a rectified derivation $\{w, \hat{ac}, \hat{c}_{\forall}, m, \hat{m}_{\forall}, \hat{m}_{\exists}, \equiv\} \Vdash_{\hat{\Phi}}^{\hat{A}} \hat{B}$, such that the induced maps $\llbracket \Phi \rrbracket: \llbracket A \rrbracket \rightarrow \llbracket B \rrbracket$ and $\llbracket \hat{\Phi} \rrbracket: \llbracket \hat{A} \rrbracket \rightarrow \llbracket \hat{B} \rrbracket$ are equal up to a variable renaming of the vertex labels.

Proof. Immediate from the definition. \square

TODO: example

A. From Contraction and Weakening to Skew Bifibrations

We say that a derivation Φ is **sane** if for every line Q in Φ we have that $\llbracket D \rrbracket$ is a fograph (i.e., all binders are legal). Clearly, every rectified derivation is sane, but not vice versa, as we might have multiple occurrences of bound variables in Q , such that $\llbracket Q \rrbracket$ is still a fograph.

Lemma 43. Let $\{w, ac, \hat{c}_{\forall}, m, \hat{m}_{\forall}, \hat{m}_{\exists}, \equiv\} \Vdash_{\Phi}^A B$ be a sane derivation. Then the induced map $\llbracket \Phi \rrbracket: \llbracket A \rrbracket \rightarrow \llbracket B \rrbracket$ is a skew bifibration.

Before we show the proof of this lemma, we introduce another useful concept: the **propositional encoding** A° of a formula A , which is a propositional formula with the property that $\llbracket A^{\circ} \rrbracket = \llbracket A \rrbracket$. For this, we introduce new propositional variables that have the same names as the (first-order) variables $x \in \text{VAR}$. Then A° is defined inductively by:

$$\begin{aligned} a^{\circ} &= a & (\forall x A)^{\circ} &= x \vee A^{\circ} \\ (A \vee B)^{\circ} &= A^{\circ} \vee B^{\circ} & (\exists x A)^{\circ} &= x \wedge A^{\circ} \\ (A \wedge B)^{\circ} &= A^{\circ} \wedge B^{\circ} \end{aligned}$$

Lemma 44. For every formula A , we have $\llbracket A^{\circ} \rrbracket = \llbracket A \rrbracket$.

Proof. Straightforward induction on A . \square

We use \equiv° to denote the restriction of \equiv to propositional formulas, i.e., the first two lines in (2).

Proof of Lemma 43. First, observe that for every inference rule $r \in \{w, ac, \hat{c}_{\forall}, m, \hat{m}_{\forall}, \hat{m}_{\exists}, \equiv\}$ the induced map $\llbracket r \rrbracket: V_{\llbracket Q \rrbracket} \rightarrow V_{\llbracket P \rrbracket}$ defines a existential preserving graph homomorphism $\llbracket Q \rrbracket \rightarrow \llbracket P \rrbracket$ and a fibration on the corresponding binding graphs. **Jui-Hsuan:** we may need to have some explication here. **Lutz:** no Therefore, their composition $\llbracket \Phi \rrbracket$ has the same properties fibration.

For showing that it is also a skew fibration, we construct for Φ its propositional encoding Φ° by translating every line into its propositional encoding. **Jui-Hsuan:** maybe mention that an instance of one of the other rules can be translated into an instance of the same rule. It’s trivial but may be worth

mentioning. $\llbracket \llbracket \text{Lutz: done below} \rrbracket \rrbracket$ The instances of the rules \widehat{m}_\forall and \widehat{m}_\exists are replaced in two steps by:

$$\widehat{ac} \frac{\frac{S\{(y \vee (Ay)^\circ) \vee (z \vee (Bz)^\circ)\}}{S\{(y \vee z) \vee ((Ay)^\circ \vee (Bz)^\circ)\}}}{S\{x \vee ((Ax)^\circ \vee (Bx)^\circ)\}}$$

and

$$\widehat{m} \frac{S\{(y \wedge (Ay)^\circ) \vee (z \wedge (Bz)^\circ)\}}{S\{(y \vee z) \wedge ((Ay)^\circ \vee (Bz)^\circ)\}} \widehat{ac} \frac{S\{(y \wedge (Ay)^\circ) \vee (z \wedge (Bz)^\circ)\}}{S\{x \wedge ((Ax)^\circ \vee (Bx)^\circ)\}}$$

respectively, where \widehat{ac} is a \widehat{ac} that renames the variables—the propositional variable, as well as the first-order variable of the same name—as everything is rectified, there is no ambiguity here. Any instance of a rule w , \widehat{ac} , \widehat{m} , or \equiv is translated to an instance of the same rule, and \widehat{c}_\forall is translated to \widehat{ac} .

This gives us a derivation $\{w, \widehat{ac}, \widehat{ac}, m, \equiv\} \parallel_{B^\circ}^{\Phi^\circ}$ such that

$[\Phi^\circ] = [\Phi]$. It has been shown in [23] that $[\Phi^\circ]$ is a skew fibration (see also [10], [42], [13]). Hence, $[\Phi]$ is a skew fibration. \square

B. From Skew Bifibrations to Contraction and Weakening

Lemma 45. *Let \mathcal{A} and \mathcal{B} be fographs, let $\varphi: \mathcal{A} \rightarrow \mathcal{B}$ be a skew bifibration, and let A and B be formulas with $\llbracket A \rrbracket = \mathcal{A}$ and $\llbracket B \rrbracket = \mathcal{B}$. Then there are derivations*

$$\frac{A}{\{w, \widehat{ac}, \widehat{c}_\forall, m, \widehat{m}_\forall, \widehat{m}_\exists, \equiv\} \parallel_{B^\circ}^{\widehat{\Phi}}} \text{ and } \frac{A\sigma_\varphi}{\{w, \widehat{ac}, \widehat{c}_\forall, m, m_\forall, m_\exists, \equiv\} \parallel_{B^\circ}^{\Phi}}$$

such that $[\widehat{\Phi}] = \varphi$ and $\widehat{\Phi}$ is a rectification of Φ , and σ_φ is the substitution induced by φ .

In the proof of this lemma, we make use of the following

concept: Let $s \parallel_{P,Q} \Psi$ be a derivation where P and Q are propositional formulas (possibly using variable $x \in \text{VAR}$ at the places of atoms). We say that Ψ can be *lifted* to S' if there are (first-order) formulas C and D such that $P = C^\circ$ and $Q = D^\circ$ and

there is a derivation $s' \parallel_{D,C}^{\Psi'}$.

Proof of Lemma 45. By Lemma 44 we have $\mathcal{A} = \llbracket A^\circ \rrbracket$ and $\mathcal{B} = \llbracket B^\circ \rrbracket$. Let $V'_B \subseteq V_B$ be the image of φ , and let B_1 be the subgraph of \mathcal{B} induced by V'_B . Hence, we have two maps $\varphi'': \mathcal{A} \rightarrow B_1$ being a surjection and $\varphi': B_1 \rightarrow \mathcal{B}$ being an injection that reflects edges. $\llbracket \text{Jui-Hsuan: what do you mean by "reflect edges"?} \rrbracket \llbracket \text{Lutz: edge downstairs implies edge upstairs} \rrbracket$ Both, φ' and φ'' remain skew bifibrations. Let us first look at φ' . Let \tilde{B}_1 be the propositional formula obtained from B° by removing all atoms that are not represented by vertices in V'_B . Then $\llbracket \tilde{B}_1 \rrbracket = B_1$. By [23, Proposition 7.6.1], we have

a derivation $\{w, \equiv\} \parallel_{\Phi_1^\circ}^{\tilde{B}_1}$. A subformula of B° is called *weak* if

it has been introduced by a weakening. All subformulas of a weak formula are also weak. Two weak subformulas B' and B'' of B° form a *weak pair* if $B^\circ \equiv S\{B' \vee B''\}$ for some context $S\{\cdot\}$. We can assume without loss of generality that whenever weak subformulas B' and B'' form a weak pair, they have been introduced by the same instance of w in Φ_1° .⁴ Now we show that Φ_1° can be lifted. For this, observe that whenever a weakening in Φ_1° deletes an atom $x \in \text{VAR}$, it must also delete all atoms in the scope of the corresponding quantifier, because φ' is a fibration on the binding graph. Hence, each line in Φ_1° is the propositional encoding P° of a first-order formula P . We now have to show that each instance of w is indeed a correct application in first-order logic. For this we have to inspect the cases a weakening happens inside a subformula $x \vee C$ or $x \wedge C$ in Φ_1° . There are the following cases:

$$\frac{S\{x \vee C\}}{S\{x \vee D \vee C\}} \quad \frac{S\{x \wedge C\}}{S\{x \wedge (D \vee C)\}} \quad \frac{S\{x \wedge C\}}{S\{(x \vee D) \wedge C\}}$$

In the first case the weakening happens inside the scope of a \forall -quantifier, and in the second case inside the scope of a \exists -quantifier. Both cases are unproblematic in first-order logic. However, in the third case an \exists -quantifier would be transformed into an \forall -quantifier. But as φ has to preserve existentials, this third case cannot occur. Thus we have a first

order derivation $\{w, \equiv\} \parallel_{B_1}^{\Phi_1}$ with $B_1^\circ = \tilde{B}_1$.

Let us now look at φ'' . Let $\mathcal{A}_1 = \mathcal{A}\sigma_\varphi$ be the graph obtained from \mathcal{A} by applying σ_φ to all the labels. Note that \mathcal{A}_1 is not necessarily a fograph, as binders might be illegal. But it still is a cograph, and we have a surjective skew fibration $\varphi'': \mathcal{A}_1 \rightarrow B_1$ that preserves the labels. Therefore, by [42,

Proposition 7.5], there is a derivation $\{ac, m, \equiv\} \parallel_{\Phi_2^\circ}^{\mathcal{A}_1^\circ}$, where

$\mathcal{A}_1^\circ = A^\circ\sigma_\varphi$ is the result of applying σ_φ to A° . Note that $\mathcal{A}_1^\circ = (A\sigma_\varphi)^\circ$ and B_1° are both propositional encodings. We plan to show that Φ_2 can be lifted to $\{ac, \widehat{c}_\forall, m, m_\forall, m_\exists, \equiv\}$. However, observe that not every formula occurring in Φ_2 is a propositional encoding. There are two reasons for this: (i) we might have $P \equiv^\circ Q$ where P is a propositional encoding but Q is not, and (ii) the rule \widehat{ac} can duplicate an atom $x \in \text{VAR}$. Let us write \widehat{ac}_x for such instances. The problem with (i) is that we could have the following situation

$$\begin{aligned} &\equiv^\circ \frac{S\{(x \wedge (E \wedge C)) \vee (x \wedge (F \wedge D))\}}{S\{((x \wedge E) \wedge C) \vee ((x \wedge F) \wedge D)\}} \\ &\quad \widehat{m} \frac{S\{((x \wedge E) \wedge C) \vee ((x \wedge F) \wedge D)\}}{S\{((x \wedge E) \vee (x \wedge F)) \wedge (C \vee D)\}} \end{aligned} \quad (4)$$

where x occurs in $C \vee D$. Then premise and conclusion are both propositional encodings, but the whole derivation cannot

⁴If Φ_1° is not of that shape, it can brought into this form by simple rule permutations, and then collapsing several weakenings into a single one.

be lifted. However, since we demand that the mapping is a fibration (and therefore a monomorphism) on the binding graphs, there must be another instance of m further below in the derivation:

$$m \frac{S'\{(x \wedge E) \vee (x \wedge F)\}}{S'\{(x \vee x) \wedge (E \vee F)\}} \quad (5)$$

We can permute both instances via the following more general scheme (see [?], [?] for a general discussion on permutations of the m -rule):

$$m \frac{S\{(G \wedge E \wedge C) \vee (G \wedge F \wedge D)\}}{S\{((G \wedge E) \vee (H \wedge F)) \wedge (C \vee D)\}} \leftrightarrow m \frac{S\{(G \wedge E \wedge C) \vee (G \wedge F \wedge D)\}}{S\{(G \vee H) \wedge ((E \wedge C) \vee (F \wedge D))\}} \quad (6)$$

We omitted some instances of \equiv° and some parentheses. We now call instances of m as in (4) *illegal*, and we can transform Φ_2° through m -permutations (6) into a derivation that does not contain any illegal m -instances. To address (ii), we also apply a permutation argument, permuting all instances of ac_x up until they either reach the top of the derivation or an instance of m which separates the two atoms in the premise. More precisely, we consider the following inference rule

$$ac_x \frac{S_0\{S_1\{x\} \vee S_2\{x\}\}}{S\{x\}} \quad (7)$$

where $S_1\{\cdot\} \equiv \{\cdot\} \vee E$ and $S_2\{\cdot\} \equiv \{\cdot\} \vee F$ and $S\{\cdot\} \equiv S_0\{\{\cdot\} \vee E \vee F\}$ for some formulas E and F , where E or F or both might be empty. The rule ac_x permutes over \equiv , ac , and other instances of ac_x , and over instances of m if they occur inside S_0 or S_1 or S_2 . The only situation in which ac_x cannot be permuted up is the following:

$$ac_x \frac{m \frac{S\{(R_1\{x\} \wedge C) \vee (R_2\{x\} \wedge D)\}}{S\{(R_1\{x\} \vee R_2\{x\}) \wedge (C \vee D)\}}}{S\{R\{x\} \wedge (C \vee D)\}} \quad (8)$$

We can therefore assume that all instances of ac_x , that contract an atom $x \in \text{VAR}$ are either at the top of Φ_2° or below a m -instance as in (8). We now lift Φ_2° to $\{ac, c_\forall, m, m_\forall, m_\exists, \equiv\}$, proceed by induction on the height of Φ_2° , beginning at the top, making a case analysis on the topmost rule that is not a \equiv .

- ac_x : We know that the premise of (7) is a propositional encoding. Hence, $S_1\{\cdot\} = \{\cdot\} \vee E^\circ$ and $S_2\{\cdot\} = \{\cdot\} \vee F^\circ$ and both x are universals, and $E^\circ \vee F^\circ$ contains all occurrences of x bound by that universal. We have the following subcases:

- E and F are both non-empty: We have

$$ac_x \frac{S^\circ\{(x \vee E^\circ) \vee (x \vee F^\circ)\}}{S^\circ\{x \vee (E^\circ \vee F^\circ)\}}$$

which can be lifted to

$$m_\forall \frac{S\{(\forall x.E) \vee (\forall x.F)\}}{S\{\forall x.(E \vee F)\}}$$

where $S^\circ\{\cdot\}$, E° , F° are the propositional encodings of $S\{\cdot\}$, E , F , respectively.

- E° is empty and F° is non-empty: We have

$$ac_x \frac{S^\circ\{x \vee (x \vee F^\circ)\}}{S^\circ\{x \vee F^\circ\}}$$

which can be lifted to

$$c_\forall \frac{S\{\forall x.\forall x.F\}}{S\{\forall x.F\}}$$

- E° is non-empty and F° is empty: This is similar to the previous case.
- E° and F° are both empty: This is impossible as the premise would not be a propositional encoding.

- ac (contracting an ordinary atom): This can trivially be lifted.
- m : There are several cases to consider.
 - If none of the four principal formulas in the premise is x or $x \vee F$ for some formula F and $x \in \text{VAR}$, then this instance of m can trivially be lifted, and we can proceed by induction hypothesis.
 - If exactly one of the four principal formulas in the premise is x for some $x \in \text{VAR}$, then this x is the encoding of an existential in the premise and of an universal in the conclusion. This is impossible, as φ has to preserve existentials.
 - If two of the four principal formulas in the premise are x for some $x \in \text{VAR}$, then we are in the following special case of (8):

$$ac_x \frac{m \frac{S\{(x \wedge C) \vee (x \wedge D)\}}{S\{(x \vee x) \wedge (C \vee D)\}}}{S\{x \wedge (C \vee D)\}}$$

which can be lifted immediately to

$$m_\exists \frac{S\{(\exists x.C) \vee (\exists x.D)\}}{S\{\exists x.(C \vee D)\}}$$

- We have a situation (8) where $R_1\{x\} \equiv x \vee E$ for some E and $R_2\{x\} \equiv x \vee F$ for some F with $R\{x\} \equiv x \vee E \vee F$ (Otherwise, the application of ac_x would not be correct.) That means, we have:

$$ac_x \frac{m \frac{S\{((x \vee E) \wedge C) \vee ((x \vee F) \wedge D)\}}{S\{((x \vee E) \vee (x \vee F)) \wedge (C \vee D)\}}}{S\{(x \vee E \vee F) \wedge (C \vee D)\}}$$

which can be lifted to

$$m_\forall \frac{m \frac{S\{((\forall x.E) \wedge C) \vee ((\forall x.F) \wedge D)\}}{S\{((\forall x.E) \vee (\forall x.F)) \wedge (C \vee D)\}}}{S\{(\forall x.(E \vee F)) \wedge (C \vee D)\}}$$

- In all other cases (e.g. exactly one of the principal formulas is of shape $x \vee F$ (and none is x), we can trivially lift the m -instance, as the quantifier structure is not affected.

Thus Φ_2° can be lifted to $\{\text{ac}, \text{c}_\forall, \text{m}, \text{m}_\forall, \text{m}_\exists, \equiv\} \parallel_{B_1}^{\text{A}\sigma_\varphi} \Phi_2$. We construct Φ by composing Φ_2 and Φ_1 . Then $\widehat{\Phi}$ can be constructed by rectifying Φ , where the variables to be used in A are already given. That $\varphi = \llbracket \widehat{\Phi} \rrbracket$ follows immediately from the construction. \square

X. SUMMARY AND PROOF OF MAIN RESULT

The only theorem of Section VI that has not yet been proved is Theorem 20 establishing the full correspondence between decomposed proofs in KS1 and combinatorial proofs. We show the proof here, by summarizing the results of the previous two Sections VIII and IX.

Proof of Theorem 20. First, assume we have a combinatorial proof $\varphi: \mathcal{C} \rightarrow \mathcal{A}$ be a combinatorial proof and a formula A with $\mathcal{A} = \llbracket A \rrbracket$. Let C be a formula with $\llbracket C \rrbracket = \mathcal{C}$, and let σ_φ be the substitution induced by φ . By Lemma 45 there is a derivation

$$\frac{\text{C}\sigma_\varphi}{\{\text{w}, \text{ac}, \text{c}_\forall, \text{m}, \text{m}_\forall, \text{m}_\exists, \equiv\} \parallel_{A} \Phi_2}$$

Since \mathcal{C} is a fonet, we have by Theorem 40 a derivation

$$\frac{\text{t}}{\text{MLS1}^x \parallel_{C} \Phi_1'}$$

This derivation remains valid if we apply the substitution σ_φ to every line in Φ_1' , yielding the derivation Φ_1 of $C\sigma_\varphi$ as desired.

Conversely, assume we have a decomposed derivation

$$\frac{\frac{\text{t}}{\text{MLS1}^x \parallel_{A'} \Phi_1}}{\{\text{w}, \text{ac}, \text{m}, \text{m}_\forall, \text{m}_\exists, \equiv\} \parallel_{A} \Phi_2} \quad (9)$$

Then we can transform Φ_1 into a rectified form $\widehat{\Phi}_1$, proving \widehat{A}' . By Theorem 34, the linked fograph $\llbracket \widehat{\Phi}_1 \rrbracket = \langle \llbracket \widehat{A}' \rrbracket, \sim_{\widehat{\Phi}_1} \rangle$ is a fonet. Then, by Lemma 42, there is a rectified derivation

$$\frac{\widehat{A}'}{\{\text{w}, \widehat{\text{ac}}, \widehat{\text{c}}_\forall, \widehat{\text{m}}, \widehat{\text{m}}_\forall, \widehat{\text{m}}_\exists, \equiv\} \parallel_{\widehat{A}} \widehat{\Phi}_2} \text{ whose induced map } \llbracket \widehat{\Phi}_2 \rrbracket: \llbracket \widehat{A}' \rrbracket \rightarrow \llbracket \widehat{A} \rrbracket$$

$\llbracket \widehat{A} \rrbracket$ is the same as the induced map $\llbracket \Phi_2 \rrbracket: \llbracket A' \rrbracket \rightarrow \llbracket A \rrbracket$ of Φ_2 . By Lemma 43, this map is a skew bifibration. Hence, we have a combinatorial proof $\varphi: \mathcal{C} \rightarrow \mathcal{A}$ with $\mathcal{C} = \llbracket A' \rrbracket$.

||Lutz: shit, something's wrong...|| \square

Note that Theorem 20 shows at the same time soundness, completeness, and full completeness, as

- 1) every proof in KS1 can be translated into a combinatorial proof, and
- 2) every combinatorial proof is the image of a KS1-proof under that translation.

XI. CONCLUSION

In this paper we have uncovered a close correspondence between first-order combinatorial proofs and decomposed deep inference derivations of system KS1, and we have shown that every proof in KS1 has such a decomposed form.

The most surprising discovery for us was that all technical difficulties in our work could be reduced (in a non-trivial way) to the propositional setting.

The obvious next step in our research is to investigate proof composition and normalisation of first-order combinatorial proofs. Even in the propositional setting, the normalization of combinatorial proofs is underdeveloped. There exist two different procedures for cut elimination [10], [12], but both have their insufficiencies, and there is no general theory.

||Lutz: do we want/can say more here?||

||TODO: mention CERES||

REFERENCES

- [1] G. Frege, *Begriffsschrift*. Louis Nebert, Halle, 1879, English Translation in: J. van Heijenoort (ed.), *From Frege to Gödel*, Harvard University Press: 1977.
- [2] D. Hilbert, “Die logischen Grundlagen der Mathematik,” *Mathematische Annalen*, vol. 88, pp. 151–165, 1922.
- [3] G. Gentzen, “Untersuchungen über das logische Schließen. I,” *Mathematische Zeitschrift*, vol. 39, pp. 176–210, 1935.
- [4] G. Gentzen, “Untersuchungen über das logische Schließen. II,” *Mathematische Zeitschrift*, vol. 39, pp. 405–431, 1935.
- [5] R. M. Smullyan, *First-Order Logic*. Berlin: Springer-Verlag, 1968.
- [6] J. A. Robinson, “A Machine-Oriented Logic Based on the Resolution Principle,” *Journal of the ACM*, vol. 12, pp. 23–41, 1965.
- [7] D. Hilbert, “Mathematische Probleme,” *Nachrichten der Königlichen Gesellschaft der Wissenschaften zu Göttingen, mathematisch-physikalische Klasse*, vol. 3, pp. 253–297, 1900.
- [8] R. Thiele, “Hilbert’s Twenty-fourth Problem,” *American Mathematical Monthly*, vol. 110, pp. 1–24, 2003.
- [9] D. Hughes, “Proofs Without Syntax,” *Annals of Mathematics*, vol. 164, no. 3, pp. 1065–1076, 2006.
- [10] D. Hughes, “Towards Hilbert’s 24th Problem: Combinatorial Proof Invariants:(preliminary version),” *Electronic Notes in Theoretical Computer Science*, vol. 165, pp. 37–63, 2006.
- [11] L. Straßburger, “The Problem of Proof Identity, and Why Computer Scientists Should Care About Hilbert’s 24th Problem,” *Philosophical Transactions of the Royal Society A*, vol. 377, no. 2140, p. 20180038, 2019.
- [12] L. Straßburger, “Combinatorial Flows and Their Normalisation,” in *2nd International Conference on Formal Structures for Computation and Deduction (FSCD 2017)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2017.
- [13] L. Straßburger, “Combinatorial Flows and Proof Compression,” Inria Saclay, Research Report RR-9048, 2017. [Online]. Available: <https://hal.inria.fr/hal-01498468>
- [14] M. Acclavio and L. Straßburger, “From Syntactic Proofs to Combinatorial Proofs,” in *Automated Reasoning - 9th International Joint Conference, IJCAR 2018, Held as Part of the Federated Logic Conference, FloC 2018, Oxford, UK, July 14-17, 2018, Proceedings*, D. Galmiche, S. Schulz, and R. Sebastiani, Eds., vol. 10900. Springer, 2018, pp. 481–497.
- [15] M. Acclavio and L. Straßburger, “On Combinatorial Proofs for Logics of Relevance and Entailment,” in *26th Workshop on Logic, Language, Information and Computation (WoLLIC 2019)*, R. Iemhoff and M. Moortgat, Eds. Springer, 2019.
- [16] M. Acclavio and L. Straßburger, “On Combinatorial Proofs for Modal Logic,” in *International Conference on Automated Reasoning with Analytic Tableaux and Related Methods*. Springer, 2019, pp. 223–240.
- [17] W. Heijltjes, D. Hughes, and L. Straßburger, “Intuitionistic Proofs Without Syntax,” in *2019 34th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*. IEEE, 2019, pp. 1–13.

- [18] D. Hughes, “First-order Proofs Without Syntax,” *arXiv preprint arXiv:1906.11236*, 2019.
- [19] J.-Y. Girard, “Linear Logic,” vol. 50, pp. 1–102, 1987.
- [20] S. Abramsky and R. Jagadeesan, “Games and Full Completeness for Multiplicative Linear Logic,” *Journal of Symbolic Logic*, vol. 59, no. 2, pp. 543–574, 1994.
- [21] J. Herbrand, “Recherches sur la Théorie de la Démonstration,” Ph.D. dissertation, University of Paris, 1930.
- [22] K. Brünnler, “Cut Elimination Inside a Deep Inference System for Classical Predicate Logic,” *Studia Logica*, vol. 82, no. 1, pp. 51–71, 2006.
- [23] L. Straßburger, “A Characterization of Medial as Rewriting Rule,” in *International Conference on Rewriting Techniques and Applications*. Springer, 2007, pp. 344–358.
- [24] A. S. Troelstra and H. Schwichtenberg, *Basic Proof Theory*. Cambridge University Press, 2000, no. 43.
- [25] J.-Y. Girard, “Quantifiers in Linear Logic,” *Temi e prospettive della logica e della filosofia della scienza contemporanea*, vol. 1, pp. 95–130, 1988.
- [26] A. Fleury and C. Retoré, “The Mix Rule,” *Math. Structures in Comp. Science*, vol. 4, no. 2, pp. 273–285, 1994.
- [27] G. Bellin, “Subnets of Proof-nets in Multiplicative Linear Logic with MIX,” *Mathematical Structures in Computer Science*, vol. 7, no. 6, pp. 663–699, 1997.
- [28] R. J. Duffin, “Topology of Series-parallel Networks,” *Journal of Mathematical Analysis and Applications*, vol. 10, no. 2, pp. 303–318, 1965.
- [29] K. Brünnler, “Deep Inference and Symmetry for Classical Proofs,” Ph.D. dissertation, Technische Universität Dresden, 2003.
- [30] B. Ralph, “Modular Normalisation of Classical Proofs,” Ph.D. dissertation, University of Bath, 2019.
- [31] A. A. Tubella and A. Guglielmi, “Subatomic Proof Systems: Splittable Systems,” *ACM Transactions on Computational Logic (TOCL)*, vol. 19, no. 1, pp. 1–33, 2018.
- [32] K. Brünnler, “Locality for Classical Logic,” *Notre Dame Journal of Formal Logic*, vol. 47, no. 4, pp. 557–580, 2006. [Online]. Available: <http://www.iam.unibe.ch/kai/Papers/LocalityClassical.pdf>
- [33] J.-Y. Girard, “Proof-nets: The Parallel Syntax for Proof-theory,” in *Logic and Algebra*, A. Ursini and P. Agliano, Eds. Marcel Dekker, New York, 1996.
- [34] A. Guglielmi and L. Straßburger, “Non-commutativity and MELL in The Calculus of Structures,” in *Computer Science Logic, CSL 2001*, ser. LNCS, L. Fribourg, Ed., vol. 2142. Springer-Verlag, 2001, pp. 54–68.
- [35] D. Hughes, “Unification Nets: Canonical Proof Net Quantifiers,” in *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science*, 2018, pp. 540–549.
- [36] K. Brünnler and A. F. Tiu, “A Local System for Classical Logic,” in *International Conference on Logic for Programming Artificial Intelligence and Reasoning*. Springer, 2001, pp. 347–361.
- [37] L. Straßburger, “Linear logic and Noncommutativity in the Calculus of Structures,” Ph.D. dissertation, Technische Universität Dresden, 2003.
- [38] V. Danos and L. Regnier, “The Structure of Multiplicatives,” *Archive for Mathematical Logic*, vol. 28, no. 3, pp. 181–203, 1989.
- [39] C. Retoré, “Handsome Proof-nets: Perfect Matchings and Cographs,” *Theoretical Computer Science*, vol. 294, no. 3, pp. 473–488, 2003.
- [40] C. Retoré, “Handsome Proof-nets: R&B-Graphs, Perfect Matchings and Series-parallel Graphs,” INRIA, Research Report RR-3652, 1999. [Online]. Available: <https://hal.inria.fr/inria-00073020>
- [41] L. Straßburger, “Deep Inference and Expansion Trees for Second-order Multiplicative Linear Logic,” *Mathematical Structures in Computer Science*, vol. 29, pp. 1030–1060, 2019.
- [42] B. Ralph and L. Straßburger, “Towards a Combinatorial Proof Theory,” in *International Conference on Automated Reasoning with Analytic Tableaux and Related Methods*. Springer, 2019, pp. 259–276.

APPENDIX

A. Rule permutation for the proof of Lemma 31

941 *B. Unification Nets*

942 **[[TODO:]]**

943 In this paragraph, we associate each formula A with its
944 **formula tree** $\mathcal{F}(A)$, a directed tree with leaves labelled by
945 atoms, internal nodes labelled by connectives and quantifiers,
946 and edges directed from leaves to the root. For a sequent
947 $\Gamma = A_1, \dots, A_n$, we denote with $\mathcal{F}(\Gamma)$, the forest formed by
948 $\mathcal{F}(A_1), \dots, \mathcal{F}(A_n)$, i.e., the disjoint union of $\mathcal{F}(A_i)$'s. The
949 **roots** of $\mathcal{F}(\Gamma)$ are the roots of A_i 's

950 Let Γ be a sequent in MLL1^\times . Consider the forest $\mathcal{F}(\Gamma)$.
951 A **link** on Γ is a pair of leaves whose atoms are pre-dual. A
952 **linking** λ on Γ is a set of disjoint links such that each leaf
953 of $\mathcal{F}(\Gamma)$ is either labelled by t or in exactly one link. Similar
954 to the set of links in linked fographs, a linking can be seen
955 as a unification problem, and a **dualizer** δ of the linking λ is
956 an assignment unifying all the links in λ . There exists a **most**
957 **general dualizer** of λ if λ has a dualizer. **[[Jui-Hsuan: Now**
958 **I use the same terminology as for linked fographs]]** **[[Lutz:**
959 **use δ for the dualizer (or even better, make it a macro)]]** A
960 **dependency** is a pair $(\bullet\exists x, \bullet\forall y)$ of nodes such that the most
961 general dualizer assigns to x a term containing y .

962 Let λ is a linking on Γ that has a dualizer. The **unification**
963 **structure** $\mathcal{U}(\lambda)$ associated with λ is the forest $\mathcal{F}(\Gamma)$ together
964 with an undirected edge between leaves l and l' for every link
965 $\{l, l'\}$ in λ and a directed edge from $\bullet\exists x$ to $\bullet\forall y$ for every
966 dependency $(\bullet\exists x, \bullet\forall y)$.

967 A **switching graph** of a unification structure $\mathcal{U}(\lambda)$ is any
968 derivative of $\mathcal{U}(\lambda)$ obtained by keeping only one edge into
969 each \vee and \forall and undirecting remaining edges. A linking is
970 **correct** if it is unifiable and all of the switching graphs of its
971 associated unification structure are acyclic.

972 **Definition 46.** A **unification net** on a sequent Γ is a correct
973 linking on Γ .

974 *C. Translation between Unification Nets and MLL1^\times*

975 **[[TODO:]]**

976 **Theorem 47.** *If a sequent is provable in MLL1^\times , then there*
977 *exists a unification net on it.*

978 *Proof.* We proceed by induction on the proof of $\vdash \Gamma$ in
979 MLL1^\times , making a case analysis on the bottommost rule
980 instance:

- 981 • $\text{ax} \frac{}{\vdash a, \bar{a}}$: the linking $\{a, \bar{a}\}$ is correct.
- 982 • $\text{t} \frac{}{\vdash t}$: the empty linking is correct.
- 983 • $\text{mix} \frac{\vdash \Gamma \quad \vdash \Delta}{\vdash \Gamma, \Delta}$: By induction hypothesis, there is a
984 correct linking on Γ and another one on Δ , their union
985 giving a correct linking on Γ, Δ .

- $\vee \frac{\vdash \Gamma, A, B}{\vdash \Gamma, A \vee B}$: By induction hypothesis, there is a correct
linking on Γ, A, B , and it is correct on $\Gamma, A \vee B$ as well. 987
- $\wedge \frac{\vdash \Gamma, A \quad \vdash B, \Delta}{\vdash \Gamma, A \wedge B, \Delta}$: By induction hypothesis, there is a 988
correct linking on Γ, A and another one on B, Δ , their 989
union giving a correct linking on $\Gamma, A \wedge B, \Delta$. 990
- $\exists \frac{\vdash \Gamma, A[x/t]}{\vdash \Gamma, \exists x.A}$: By induction hypothesis, there is a correct 991
linking λ on $\Gamma, A[x/t]$. For each atom in $\Gamma, A[x/t]$, there 992
is a corresponding atom in $\Gamma, \exists x.A$. There is therefore a 993
linking λ' on $\Gamma, \exists x.A$ obtained from λ via this correspon- 994
dence, and it is not difficult to check that λ' is correct as 995
well. 996
- $\forall \frac{\vdash \Gamma, A}{\vdash \Gamma, \forall x.A}$ (x not free in Γ) : By induction hypothesis, 997
there is a correct linking on Γ, A , and it is easy to see 998
that it is a correct linking on $\Gamma, \forall x.A$ as well. 999

This allows to define a translation $[\cdot]$ from proofs in MLL1^\times 1000
to unification nets. \square 1001

Theorem 48. *Any unification net can be obtained via the* 1002
translation $[\cdot]$ given in Theorem 47. 1003

To prove this theorem, we need some basic lemmas about 1004
connected components in switching graphs of unification nets. 1005

Lemma 49. *The number of connected components of an acyclic* 1006
graph $\mathcal{G} = \langle V_{\mathcal{G}}, E_{\mathcal{G}} \rangle$ is equal to $|E_{\mathcal{G}}| - |V_{\mathcal{G}}|$. 1007

Proof. By a straightforward induction on $|V_{\mathcal{G}}|$. \square 1008

Lemma 50. *The number of connected components is the same* 1009
for any switching graph of a unification net. 1010

Proof. An immediate consequence of Lemma 49. \square 1011

In the proof, we also use the notion of **frame** introduced by 1012
Hughes in [35]. 1013

Definition 51. Let λ be a unification net on an MLL1^\times sequent 1014
 Γ . We define the **frame** of λ by exhaustively applying the 1015
following subformula rewriting steps, to obtain a linking λ_m 1016
on an $\text{MLL} + \text{mix}$ sequent Γ_m : 1017

- 1) **Encode dependencies as fresh links.** For each depen- 1018
dency $\exists x \rightarrow \forall y$, with corresponding subformulas $\exists x.A$ 1019
and $\forall y.B$, we add a fresh link as follows. Let P be a fresh 1020
(nullary) predicate symbol. Replace $\exists x.A$ with $P \wedge \exists x.A$ 1021
and $\forall y.B$ with $\bar{P} \vee \forall y.B$, and add an axiom link between 1022
 P and \bar{P} . 1023
- 2) **Erase quantifiers.** After step 1, erase all the quantifiers. 1024
(We no longer need their leaps since they are encoded 1025
as links in step 1.) 1026
- 3) **Simplify atoms.** After step 2, replace every predicate 1027
 $Pt_1 \dots t_n$ with a nullary predicate symbol P . 1028

Note that the linking λ_m is a valid $\text{MLL} + \text{mix}$ proof net. 1029

Lemma 52. Suppose that λ is a MLL + mix proof net which is connected and such that any of its switching graphs is not connected. Then there exists a \vee node in $\mathcal{U}(\lambda)$ such that λ is correct on the sequent Γ' obtained from Γ by replacing this \vee by a \wedge .

Proof. Suppose that such a \vee node does not exist. Then it is clear that for any two nodes, there exists a switching graph containing a path between them and this path corresponds to an AE -path in [39]. By [39, Propostion 3], λ corresponds to a sequent proof that does not use mix, which implies the connectedness of the switching graphs of λ . Contradiction. **TO CHECK:** \square

Lemma 53. Suppose that λ is a MLL1^X proof net which is connected and such that any of its switching graphs is not connected. Then there exists a \vee node in $\mathcal{U}(\lambda)$ such that λ is correct on the sequent Γ' obtained from Γ by replacing this \vee by a \wedge .

Proof. Consider the frame λ_m of λ . The number of any switching graph of $\mathcal{U}(\lambda)$ is equal to that of $\mathcal{U}(\lambda_m)$. Apply Lemma 52 and it is clear that such \vee cannot be one of the fresh \vee 's added during the frame construction. \square

We can now give the proof of Theorem 48.

Proof of Theorem 48. Let λ be a unification net on Γ . We proceed by induction on the number of connected components of the unification structure $\mathcal{U}(\lambda)$:

- If there is only one connected component, we proceed by induction on the number k of connected components of any switching graph of $\mathcal{U}(\lambda)$. If $k = 1$, we obtain a proof Φ in MLL1^X such that $[\Phi] = \lambda$ by applying [35, Theorem 3]. If $k > 1$, using the Lemma 53, we obtain a sequent Γ' on which λ is correct by transforming a \vee node into a \wedge . By induction hypothesis, there is a proof Φ' in MLL1^X whose translation is λ . By considering the \wedge rule instance corresponding to the \wedge node in Φ' , we

$$\text{have: } \Phi' = \wedge \frac{\frac{\frac{\Phi_1}{\vdash \Delta_1, A} \quad \frac{\Phi_2}{\vdash B, \Delta_2}}{\vdash \Delta_1, A \wedge B, \Delta_2}}{\vdash \Gamma'}. \text{ We can thus obtain}$$

$$\text{a proof } \Phi \text{ of } \Gamma: \Phi = \frac{\text{mix} \frac{\frac{\frac{\Phi_1}{\vdash \Delta_1, A} \quad \frac{\Phi_2}{\vdash B, \Delta_2}}{\vdash \Delta_1, A, B, \Delta_2}}{\vdash \Delta_1, A \vee B, \Delta_2}}{\vdash \Gamma}$$

$[\Phi] = \lambda$.

- If there are $n > 1$ connected components, add a fresh \vee node connecting two formulas belonging to different

connected components of Γ to get a new sequent Γ' . Define a unification net λ' on Γ' using the same linking as λ . By induction hypothesis, since $\mathcal{U}(\lambda')$ has $n - 1$ connected components, there is a MLL1^X proof Φ' such that $[\Phi'] = \lambda'$. Consider the \vee rule instance corresponding to the \vee node in question. Since \vee is invertible, we can permute downwards this rule instance until it becomes the last rule of the proof (note that this transformation does not change the image of the proof by the translation $[\cdot]$) to get a new proof Φ'' of Γ' . By deleting the last rule instance from Φ'' , we obtain a proof Φ of Γ such that $[\Phi] = \lambda$. **TO CHECK:** \square

We proceed by induction on the number of connectives in Γ . In the base case, Γ is of the form

$$p_1(t_{11}, \dots, t_{1n_1}), \overline{p_1}(t_{11}, \dots, t_{1n_1}), \dots, p_k(t_{k1}, \dots, t_{kn_k}), \overline{p_k}(t_{k1}, \dots, t_{kn_k}), \underbrace{t, \dots, t}_{m \text{ times}}$$

and λ is the linking $\{(a_1, \overline{a_1}), \dots, (a_k, \overline{a_k})\}$, where $a_i = p_i(t_{i1}, \dots, t_{in_i})$, which equals to $[\Pi]$, where Π is the proof consisting of m instances of the t rule, n instances $\text{ax} \frac{}{\vdash a_i, \overline{a_i}}$ of the ax rule, and followed by $m + k - 1$ instances of the mix rule.

Now we consider the inductive cases:

- $\Gamma = \Delta, A \vee B$: Let $\Gamma' = \Delta, A, B$. Define λ' on Γ' using the same links as λ by identifying the leaves of $\mathcal{F}(\Gamma')$ with those of $\mathcal{F}(\Gamma)$. We now check that λ' is a unification net:
 - The most general dualizer of λ is also the most general dualizer of λ' as they correspond to the same unification problem. Hence, the unification structure $\mathcal{U}(\lambda')$ is equal to the restriction of $\mathcal{U}(\lambda)$ to the nodes of $\mathcal{F}(\Gamma')$.
 - Every switching graph of λ' is acyclic: if there were some switching graph of $\mathcal{U}(\lambda')$ containing a cycle, it would induce a switching graph of $\mathcal{U}(\lambda)$ containing also a cycle, by adding an edge from the root of $\mathcal{F}(A)$ to the \vee node in question.
- $\Gamma = \Delta, \forall x.A$: Let $\Gamma' = \Delta, A$. Define λ' on Γ' using the same links as λ . We now check that λ' is a unification net:
 - The most general dualizer of λ is also the most general dualizer of λ' as they correspond to the same unification problem.
 - Every switching graph of $\mathcal{U}(\lambda')$ is acyclic: if there were some switching graph of $\mathcal{U}(\lambda')$ containing a cycle, it would induce a switching graph of $\mathcal{U}(\lambda)$ containing also a cycle, by adding an edge from the root of $\mathcal{F}(A)$ to the \forall node in question.
- $\mathcal{F}(\Gamma)$ has a root $\exists x$ with no outgoing dependency edge:

D. Translation between Unification Nets and Fonets

E. First-order Logic

In this paper, we also use some *deep inference* [36] rules that are in the following form:

$$\frac{\vdash S\{A\}}{\vdash S\{B\}}$$

where $S\{ \}$ stands for a *context*, which corresponds to a sequent with a hole taking the place of an atom, and $S\{A\}$ represents the sequent or formula obtained by replacing the hole in $S\{ \}$ with the formula A . Formally,

$$C ::= \Box \mid A \vee C \mid C \wedge A \mid \exists x C \mid \forall x C.$$

$$S ::= C \mid A, S \mid S, A$$

where A is a formula. The above rule can be thus seen as the rewriting rule $A \rightarrow B$.

We use the notation $\parallel^{\mathcal{P}}$ for denoting that there is a derivation from premise $\vdash S\{A\}$ to conclusion $\vdash S\{B\}$ in system \mathcal{P} for any context S .

F. Graphs

G. First-order combinatorial proofs

H. MLL1^X and Unification Nets

In MLL1^X, terms, atoms, formulas are defined as in first-order logic. For simplicity, we choose to use \vee and \wedge instead of \wp and \otimes which are generally used in the presentation of linear logic. A formula A is identified with its *formula tree* $\mathcal{F}(A)$, a directed tree with leaves labelled by atoms, internal nodes labelled by connectives and quantifiers, and edges directed from leaves to the root. A *sequent* Γ is simply a disjoint union of formulas. We write comma for disjoint union.

Sequents are proved using the inference rules of MLL1^X:

$$\begin{array}{c} \frac{}{\vdash A, \neg A} \text{ ax} \quad \frac{\vdash \Gamma, A \quad \vdash \Delta, \neg A}{\vdash \Gamma, \Delta} \text{ cut} \\[10pt] \frac{\vdash \Gamma, A \quad \vdash \Delta, B}{\vdash \Gamma, \Delta, A \wedge B} \wedge \quad \frac{\vdash \Gamma, A, B}{\vdash \Gamma, A \vee B} \vee \\[10pt] \frac{\vdash \Gamma, A}{\vdash \Gamma, \forall x A} \forall (x \notin fv(\Gamma)) \quad \frac{\vdash \Gamma, A[t/x]}{\vdash \Gamma, \exists x A} \exists \end{array}$$

Fig. 7. Sequent calculus for MLL1^X

We also consider the mix rule:

$$\frac{\vdash \Gamma \quad \vdash \Delta}{\vdash \Gamma, \Delta} \text{ mix}$$

Let Γ be a sequent in MLL1 + mix. A *link* on Γ is a pair of leaves whose atoms are pre-dual. A *linking* on Γ is a set of disjoint links such that each leaf of Γ is in exactly one link. Similar to the set of links in the linked fograph, a

linking can be seen as a unification problem, and a link is said *unifiable* if the corresponding unification problem is solvable. *Dependencies* are defined as previously.

I. Decomposition Theorem

Consider the following deep inference rules [36]:

$$\frac{\vdash S\{A \vee A\}}{\vdash S\{A\}} \text{ c} \quad \frac{\vdash S\{f\}}{\vdash S\{A\}} \text{ w}$$

Note that the ctr (resp. wk) rule in LK is derivable in $\{c, \vee\}$ (resp. $\{w, f\}$) and that c and w rules permute downwards with the non-structural rules of LK.

$$\begin{array}{c} \frac{\vdash \Gamma, A, A}{\vdash \Gamma, A} \text{ ctr} \rightsquigarrow \frac{\vdash \Gamma, A, A}{\vdash \Gamma, A \vee A} \vee \text{ c} \\[10pt] \frac{\vdash \Gamma}{\vdash \Gamma, A} \text{ wk} \rightsquigarrow \frac{\vdash \Gamma}{\vdash \Gamma, f} \text{ f} \text{ w} \end{array}$$

We also give an example to show how rule permutation works:

$$\frac{\frac{\Gamma, A \vee A}{\Gamma, A} \text{ c} \quad \Delta, B}{\Gamma, \Delta, A \wedge B} \wedge \rightsquigarrow \frac{\frac{\Gamma, A \vee A \quad \Delta, B}{\Gamma, \Delta, (A \vee A) \wedge B} \wedge}{\Gamma, \Delta, A \wedge B} \text{ c}$$

We want to establish the following theorem:

Theorem 54. *Let Γ be a sequent. Then there is a proof of Π in LK + mix iff there is a proof of some sequent Δ in MLL1 + mix and a derivation from Δ to Γ consisting of the c and w rules only.*

Proof. (\Rightarrow) This direction comes from the above observation: it suffices to permute downwards all the instances of the c and w rules.

(\Leftarrow) We regard the proof in MLL1 + mix as a proof in LK + mix. Then we put the derivation consisting of only c and w under the proof in LK + mix. Now we try to permute all the instances c and w upwards with the rules of LK and mix. For the c part, the only non-trivial case is the permutation with the \vee rule where the formula generated is $A \vee A$.

$$\frac{\vdash \Gamma, A, A}{\vdash \Gamma, A \vee A} \vee \text{ c} \rightsquigarrow \frac{\vdash \Gamma, A, A}{\vdash \Gamma, A} \text{ ctr}$$

In this case, the permutation of this instance of c stops and we continue with the remaining instances.

For the w part, the only non-trivial case is the permutation with the f rule (or the instance of wk where f is introduced):

$$\frac{\frac{\vdash \Gamma}{\vdash \Gamma, f} \text{ f}}{\vdash \Gamma, A} \text{ w} \rightsquigarrow \frac{\vdash \Gamma}{\vdash \Gamma, A} \text{ wk}$$

In this case, the permutation of this instance of w stops and we continue with the remaining instances.

□

D. Hughes proves in [35] the soundness and completeness of unification nets with respect to MLL1 + mix. In the following, we establish the equivalence between unification nets and fonets.

J. Equivalence between unification nets and fonets

In the following, we usually confound a vertex with its label.

Definition 55. A *switching path* of a unification structure $U(\lambda)$ is a path in a switching graph of $U(\lambda)$.

Definition 56. A *switching path* of a formula tree $\mathcal{F}(A)$ is a path in $\mathcal{F}(A)$ that does not go through both incoming edges of a \vee .

Proposition 57. In a formula tree, the root is connected to every vertex by a switching path.

Now we give the key proposition relating a fograph to its corresponding formula tree.

Proposition 58. Let u and v be two distinct vertices of a fograph $\llbracket \llbracket A \rrbracket \rrbracket$, then we have the equivalence between:

- u and v are adjacent in $\llbracket \llbracket A \rrbracket \rrbracket$
- u and v are connected by a switching path of $\mathcal{F}(A)$, and if one of them is a universal quantifier, then the other is not a descendant of the former.

Proof. By induction on A .

- If A is an atom, trivial.
- If $A = A_1 \wedge A_2$, then we distinguish two cases:
 - u and v are both in A_1 (resp. A_2): trivial by the induction hypothesis.
 - one of them is in A_1 and the other is in A_2 : they are adjacent in $\llbracket \llbracket A \rrbracket \rrbracket$ by definition. By Proposition 57, the one in A_1 (resp. A_2) is connected to the vertex representing A_1 (resp. A_2) by a switching path. Together with the two edges incident to $A_1 \wedge A_2$, we obtain a switching path connecting u and v .
- If $A = A_1 \vee A_2$, then we distinguish two cases:
 - u and v are both in A_1 (resp. A_2): trivial by the induction hypothesis.
 - one of them is in A_1 and the other is in A_2 : they are not adjacent in $\llbracket \llbracket A \rrbracket \rrbracket$ by definition. It is clear that they are not connected by a switching path.
- If $A = \exists x A'$, then we distinguish two cases:
 - u and v are both in A' : trivial by the induction hypothesis.
 - one of them is $\exists x$ and the other is in A' : trivial by Proposition 57
- If $A = \forall x A'$, then we distinguish two cases:
 - u and v are both in A' : trivial by the induction hypothesis.
 - one of them is $\forall x$ and the other is in A' : they are not adjacent in $\llbracket \llbracket A \rrbracket \rrbracket$ by definition and it is clear that the former is a descendant of $\forall x$.

□

Proposition 59. If there exists an induced bimatching of the linked fograph $G = \llbracket \llbracket A \rrbracket \rrbracket$, then there exists a switching graph of the corresponding unification net which contains a cycle.

Proof. Suppose that there exists a set W inducing a bimatching in G . Then (W, E_G) and (W, L_G) are matchings.

Let E_W (resp. L_W) be the restriction of E_G (resp. L_G) to W . If $E_W \cap L_W \neq \emptyset$, then there exist u and v such that $uv \in E_G$ and $uv \in L_G$. By Proposition 58, there exists a switching path of the formula tree of A . Together with the leap uv , this path induces a cycle in a switching graph of the corresponding unification structure.

We can now suppose that E_W and L_W are disjoint. It is not difficult to see the existence of an alternating and elementary cycle in the bicoloured graph $(W, E_W \uplus L_W)$, i.e. a cycle of which the edges are alternately in E_W and L_W and containing no two equal vertices. By Proposition 58, this cycle induces a cycle in the unification structure. Now we want to construct a switching graph that contains this cycle.

Consider a universal quantifier $\forall x$. If $\forall x \notin W$, then we keep the incoming edge from its direct subformula and remove all the dependencies. Otherwise, since (W, L_G) is a matching, there exists a unique existential quantifier adjacent to $\forall x$ and we keep thus the corresponding edge in the unification structure.

Now consider a \vee . We distinguish three cases:

- the cycle goes through none of the two branches (incoming edges) of the \vee : we can choose an arbitrary switching for this \vee
- the cycle goes through exactly one branch: we choose the corresponding switching
- the cycle goes through both branches: this means that there exist $v_L \in W$ (resp. v_R) in the left (resp. right) branch, $u_L, u_R \in W$, such that $u_L v_L, u_R v_R \in E_W$ and that the corresponding switching path from u_L to v_L (resp. from u_R to v_R) goes through the left (resp. right) edge of \vee .

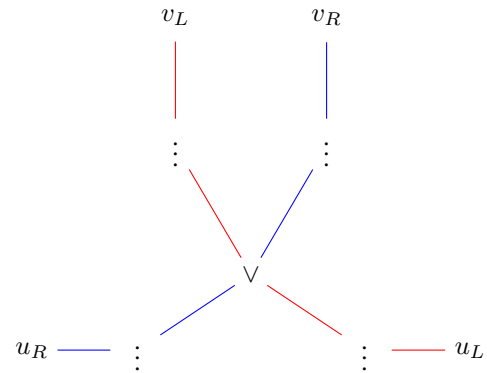


Fig. 8. A schema showing that the two branches of the same \forall cannot be used in the cycle at the same time.

The red (resp. blue) path is the switching path corresponding to the edge $u_L v_L$ (resp. $u_R v_R$) in E_W .

It is clear that u_L (resp. u_R) is not in the branches of the \vee . Otherwise, there will be no switching path from u_L to v_L

By Proposition 58, we know that u_L and u_R are not universal quantifiers which are ancestors the \vee and that there exist one switching path from u_L to v_L and one from u_R to v_R . In particular, there exist one switching path from u_L to the \vee and one from the \vee to v_R , and by concatenating the two, we obtain a switching path from u_L to v_R . By Proposition 58, u_L and v_R are thus adjacent in (W, E_G) , which is impossible since (W, E_W) is a matching.

Notice that the switching paths here are in the underlying formula tree. We have to verify that they are compatible with the choices of switching made for universal quantifiers. That is, if $uv \in E_W$, then for all the universal quantifiers $\forall x$ on the switching path (in the formula tree), we have chosen in the first part of the proof to keep the only edge from the child of $\forall x$ to itself. In fact, if there exists a universal quantifier $w \in W$ on the switching path $u \rightarrow v$, then one of u and v is not a descendant of w . Moreover, if u (resp. v) is a universal quantifier, then w is not in its scope. By Proposition 58, $\{wu, wv\} \cap E_W \neq \emptyset$, which is impossible since (W, E_W) is a matching. We have thus constructed a switching graph containing this cycle. \square

Proposition 60. *If one of the switching graphs of the unification structure of A contains a cycle or is not connected, then there exists an induced bimatching of the corresponding linked fograph.*

Proof. We use frames introduced by D. Hughes in Section 4 of [35].

Definition 61. Let θ be a unification structure on an $\text{MLL}^1 \times$ sequent Γ . We define the **frame** of θ by exhaustively applying the following subformula rewriting steps, to obtain a proof structure θ_m on an MLL sequent Γ_m :

- 1) **Encode dependencies as fresh links.** For each dependency $\exists x \rightarrow \forall y$, with corresponding subformulas $\exists xA$ and $\forall yB$, we add a fresh link as follows. Let P be a fresh (nullary) predicate symbol. Replace $\exists xA$ with $P \wedge \exists xA$ and $\forall yB$ with $\overline{P} \vee \forall yB$, and add an axiom link between P and \overline{P} .
- 2) **Erase quantifiers.** After step 1, erase all the quantifiers. (We no longer need their leaps since they are encoded as links in step 1.)
- 3) **Simplify atoms.** After step 2, replace every predicate $Pt_1 \dots t_n$ with a nullary predicate symbol P .

We have the following results:

Let u and v be atoms or quantifiers in a unification structure θ . Then they are connected by a switching path in the unification structure if, and only if, their corresponding nodes are connected by a switching path in θ_m .

Consider now a switching graph H of a unification structure θ of A .

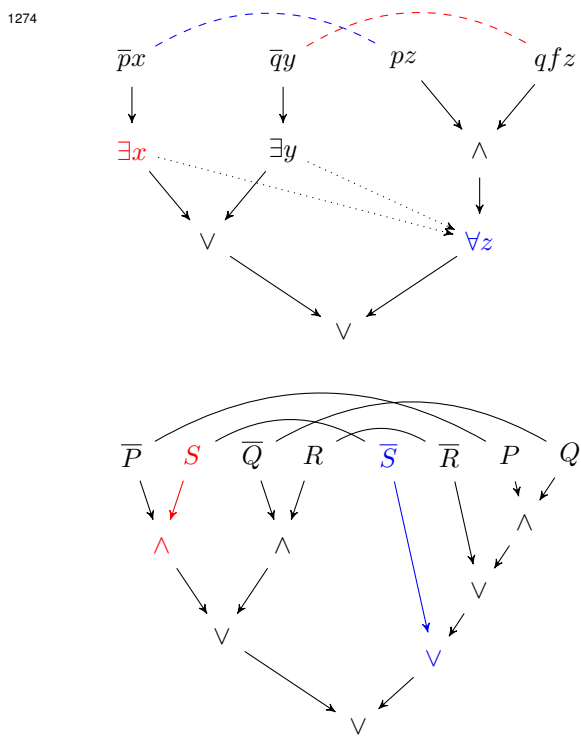


Fig. 9. A unification net and its frame. The colored part shows how the dependency $\exists x \rightarrow \forall z$ is transformed.

If H contains a cycle, then the corresponding switching graph of θ_m also contains a cycle. Hence, by applying the propositional results (Theorem 7) from [39], we conclude that there exists a chordless, alternating, and elementary cycle in the bicoloured graph $(W, E_W \uplus L_W)$, which corresponds to an induced bimatching in the linked fograph. (Note that the linked fograph (cograph) corresponding to θ_m is equivalent to the one corresponding to θ .) \square

K. From contraction/weakening to skew bifibrations

We first introduce the atomic contraction rule, the medial rule, and two rules on quantifiers.

$$\begin{array}{c}
 \frac{\vdash S\{a \vee a\}}{\vdash S\{a\}} \text{ ac} \qquad \frac{\vdash S\{(A \wedge B) \vee (C \wedge D)\}}{\vdash S\{(A \vee C) \wedge (B \vee D)\}} \text{ m} \\
 \frac{\vdash S\{\exists xA \vee \exists xB\}}{\vdash S\{\exists x(A \vee B)\}} \text{ m}_1\downarrow \qquad \frac{\vdash S\{\forall xA \vee \forall xB\}}{\vdash S\{\forall x(A \vee B)\}} \text{ m}_2\downarrow
 \end{array}$$

Here, we also consider the equivalence generated by the associativity, commutativity of \vee and the equations $t \vee A \equiv t$ and $f \vee A \equiv A$.

Now we have the following lemma:

Lemma 62. *The contraction rule c is derivable for $\{\text{ac}, \text{m}, \text{m}_1\downarrow, \text{m}_2\downarrow\}$.*

Proof. We prove that there is always $\frac{A \vee A}{\parallel_{\{ac, m, m_1 \downarrow, m_2 \downarrow\}} A}$ by structural induction on A .

- If $A = t$ or $A = f$, we have $\frac{\vdash S\{A \vee A\}}{\vdash S\{A\}} \equiv$ (the premiss and the conclusion are equivalent)
- If $A = a$, then we have $\frac{\vdash S\{a \vee a\}}{\vdash S\{a\}} \text{ ac}$
- If $A = A_1 \vee A_2$, then by the induction hypothesis, we have $\parallel_{\{ac, m, m_1 \downarrow, m_2 \downarrow\}} A_i$ for $i = 1, 2$.

$$\frac{\vdash S\{(A_1 \vee A_2) \vee (A_1 \vee A_2)\}}{\vdash S\{(A_1 \vee A_1) \vee (A_2 \vee A_2)\}} \equiv$$

Hence, we have

$$\frac{\vdash S\{A_1 \vee (A_2 \vee A_2)\}}{\vdash S\{A_1 \vee A_2\}} \equiv$$

- If $A = A_1 \wedge A_2$, then by the induction hypothesis, we have $\parallel_{\{ac, m, m_1 \downarrow, m_2 \downarrow\}} A_i$ for $i = 1, 2$.

$$\frac{\vdash S\{(A_1 \wedge A_2) \vee (A_1 \wedge A_2)\}}{\vdash S\{(A_1 \vee A_1) \wedge (A_2 \vee A_2)\}} m$$

Hence, we have

$$\frac{\vdash S\{A_1 \wedge (A_2 \vee A_2)\}}{\vdash S\{A_1 \wedge A_2\}} \equiv$$

- If $A = \exists x A'$, then by the induction hypothesis, we have $\parallel_{\{ac, m, m_1 \downarrow, m_2 \downarrow\}} A'$.

$$\frac{\vdash S\{\exists x A' \vee \exists x A'\}}{\vdash S\{\exists x (A' \vee A')\}} m_1 \downarrow$$

Hence, we have

$$\frac{\vdash S\{\exists x A'\}}{\vdash S\{\exists x A'\}} \equiv$$

- If $A = \forall x A'$, then by the induction hypothesis, we have $\parallel_{\{ac, m, m_1 \downarrow, m_2 \downarrow\}} A'$.

$$\frac{\vdash S\{\forall x A' \vee \forall x A'\}}{\vdash S\{\forall x (A' \vee A')\}} m_2 \downarrow$$

Hence, we have

$$\frac{\vdash S\{\forall x A'\}}{\vdash S\{\forall x A'\}} \equiv$$

Proof. We have:

$$\frac{\vdash S\{\exists x A\}}{\vdash S\{\exists x (A \vee f)\}} \equiv \text{ and } \frac{\vdash S\{\exists x B\}}{\vdash S\{\exists x (f \vee B)\}} \equiv$$

Thus, we have:

$$\frac{\vdash S\{\exists x A \vee \exists x B\}}{\vdash S\{\exists x (A \vee B)\}} c$$

Similar for $m_2 \downarrow$. \square

Now we define a propositional encoding for first-order formulas.

Definition 64. The propositional encoding A° of a formula A is defined inductively by:

$$\begin{aligned} a^\circ &= a \text{ for every atom } a \\ (A \vee B)^\circ &= A^\circ \vee B^\circ & (A \wedge B)^\circ &= A^\circ \wedge B^\circ \\ (\forall x A)^\circ &= U_x \vee A^\circ & (\exists x A)^\circ &= E_x \wedge A^\circ \end{aligned}$$

where U_x and E_x are fresh nullary atoms.

Similarly, we can define the propositional encoding S° of a context S inductively by setting $\square^\circ = \square$. Note that S° is also a context.

We have the following facts:

Proposition 65. For any context S and any formula A :

- A° is a formula containing no quantifier for any formula A .
- $\llbracket \llbracket A^\circ \rrbracket \rrbracket = \llbracket \llbracket A \rrbracket \rrbracket$ by confounding the atoms U_x, E_x with the variable x . Thus, a map $f : \llbracket \llbracket A^\circ \rrbracket \rrbracket \rightarrow \llbracket \llbracket B^\circ \rrbracket \rrbracket$ can be seen as a map $f : \llbracket \llbracket A \rrbracket \rrbracket \rightarrow \llbracket \llbracket B \rrbracket \rrbracket$.
- $(S\{A\})^\circ = S^\circ\{A^\circ\}$.

Proposition 66. Let A and B be two formulas such that $A \equiv B$.

$$\parallel_{\{w, c\}} A \quad \parallel_{\{w, c\}} B^\circ$$

Proof. Trivial by induction. \square

Lemma 67. Given two formulas A and B and a derivation $\Delta \parallel_{\{w, c\}} A$, then there exists a skew bifibration $G(A) \rightarrow G(B)$.

Proof. By Lemma 62, there exists a derivation $\Delta \parallel_{\{w, ac, m, m_1 \downarrow, m_2 \downarrow\}} A$.

$$\Delta \parallel_{\{w, ac, m, m_1 \downarrow, m_2 \downarrow\}} B$$

For each rule from $\{w, ac, m, m_1 \downarrow, m_2 \downarrow\}$, we define a map and show that it is a skew fibration.

- $\frac{\vdash S\{f\}}{\vdash S\{A\}} w$: the map wk maps f to anything and is identity elsewhere.
- $\frac{\vdash S\{a \vee a\}}{\vdash S\{a\}} ac$:

Lemma 63. The rules $m_1 \downarrow$ and $m_2 \downarrow$ are derivable for $\{w, c\}$.

the map ac maps the two a -labelled literals in the premise to the a -labelled literal in the conclusion.

- $\frac{\vdash S\{(A \wedge B) \vee (C \wedge D)\}}{\vdash S\{(A \vee C) \wedge (B \vee D)\}}$ m :
the map m is the canonical identity that maps A to A , \dots , D to D .
- $\frac{\vdash S\{\exists x A \vee \exists x B\}}{\vdash S\{\exists x(A \vee B)\}}$ $m_1 \downarrow$:
the map m_1 maps the two x -labelled binders in the premise to the x -labelled binder in the conclusion, A to A and B to B .
- $\frac{\vdash S\{\forall x A \vee \forall x B\}}{\vdash S\{\forall x(A \vee B)\}}$ $m_2 \downarrow$:
the map m_2 maps the two x -labelled binders in the premise to the x -labelled binder in the conclusion, A to A and B to B .

By considering propositional encodings, the maps defined are label-preserving skew fibrations on the underlying fographs according to [23].

Now we prove that each map $g \in \{wk, ac, m, m_1, m_2\}$ is a skew bifibration. To do that, it suffices to prove that g is a fibration between the corresponding binding graphs since it is already a skew fibration on the corresponding fographs and it is label-preserving and existential-preserving.

for each x -binder b in $\llbracket \llbracket B^\circ \rrbracket \rrbracket$, for each vertex $v \in V(\llbracket \llbracket A^\circ \rrbracket \rrbracket)$ such that $g(v)$ is bound by b , there exists a unique binder b' such that b' binds v .

- wk and m are clearly fibrations: the binding relations of the premise and the conclusion are exactly the same.
- ac is a fibration: suppose that a that in the conclusion a is bound by some quantifier b in S , then for each of its preimages by ac , there exists exactly one binder (in fact, b) in S that binds it.
- m_1 and m_2 are fibrations: in the conclusion, for every atom a in $A \vee B$ bound by the x -labelled quantifier, a has exactly one preimage and it is bound by the x -labelled quantifier in the premise.

Therefore, all of these maps are skew bifibrations and since skew bifibrations on fographs compose (Lemma 10.32, [18]), there exists a skew bifibration from $\llbracket \llbracket A \rrbracket \rrbracket$ to $\llbracket \llbracket B \rrbracket \rrbracket$. \square

Theorem 68. *If a formula A is provable in LK, then it has a combinatorial proof.*

Proof. By Theorem 54, there exists a formula A' such that there is a proof Π of A' in MLL1^X and a derivation D from A' to A consisting of the w and c rules only. The proof Π corresponds to a unique unification net which is equivalent to the fonet corresponding to Π , i.e., the fograph $\llbracket \llbracket A' \rrbracket \rrbracket$ together with the links of Π . By Lemma 67, there exists a skew bifibration $\llbracket \llbracket A' \rrbracket \rrbracket \rightarrow \llbracket \llbracket A \rrbracket \rrbracket$. We have thus a combinatorial proof of A . \square

From skew bifibrations to contraction/weakening

Theorem 69. *Let A and B be two formulas and $f : G(A) \rightarrow G(B)$ a skew bifibration. Then there exists a derivation $\Delta \parallel_{\{w, c\}}^A B$.*

f can be seen as a skew fibration from $G(A^\circ)$ to $G(B^\circ)$, which gives the existence of the propositions A' and B' , and of the following derivation:

$$\begin{array}{c} A^\circ \\ \Delta \parallel_m \\ A' \\ \Delta' \parallel_{ac} \\ B' \\ \Delta'' \parallel_w \\ B^\circ \end{array}$$

Lemma 70. *there exists B'' such that $B''^\circ = B'$.*

Proof. Consider the derivation Δ'' . If some U_x (or E_x) is introduced via weakening, then all the atoms it binds in B° should also be introduced via weakening. In fact, an atom of B° is introduced via weakening is equivalent to the fact that its corresponding vertex is not in the image of f . Since there is an edge from U_x (resp. E_x) to all the literals it binds in the binding graph $\llbracket \llbracket B \rrbracket \rrbracket$, if one of the atoms is in the image, U_x (resp. E_x) should also be in the image since f is a fibration on binding graphs.

This means that a such B'' can be obtained from B by erasing all the U_x and E_x introduced via weakening and all the atoms they bind. \square

We introduce new (atomic) symbols E_x^* and U_x^* which are used to represent disjunctions of E_x and U_x respectively.

We define a translation $(\cdot)^*$ inductively by:

- $(E_x \vee \dots \vee E_x)^* = E_x$
- $(U_x \vee \dots \vee U_x)^* = U_x$
- structural recursion in all the other cases.

Then the derivation:

$$\begin{array}{c} A^\circ \\ \Delta \parallel_m \\ A' \\ \Delta' \parallel_{ac} \\ B''^\circ \end{array}$$

can be translated to the derivation:

$$\begin{array}{c} A^{\circ*} \\ \Delta^* \parallel \\ B''^{\circ*} \end{array}$$

where Δ^* is the derivation obtained by replacing all the formulas F with F^* and by applying the following rule transformation:

$$\frac{S\{Q_x\}}{S\{Q_x\}} \text{ ac } \rightsquigarrow \frac{S\{Q_x\}}{S\{Q_x\}} =$$

$$\frac{S\{(E_x \wedge C) \vee (E_x \wedge D)\}}{S\{E_x \wedge (C \vee D)\}} \text{ m } \rightsquigarrow \frac{S\{(E_x \wedge C) \vee (E_x \wedge D)\}}{S\{E_x \wedge (C \vee D)\}} \text{ m}'_{1478}$$

1479 where Q_x stands for E_x or U_x .

Δ^* can now be transformed into a valid derivation Δ_1 by using the two transformation rules above and by applying them in a bottom-up style:

$$\begin{array}{c} A^{\circ*} \\ \Delta_1 \parallel_{\text{ac}, \text{m}, \text{m}'} \\ B''^{\circ*} \end{array}$$

1480 **Lemma 71.** *Every line of Δ_1 is a propositional encoding.*

1481 *Proof.* We proceed by bottom-up induction in the derivation. Clearly, $(B''^{\circ})^*$ is a propositional encoding as there is no disjunction of Q_x in it.

1484 First consider the ac rule: $\frac{C \vee C}{C} \text{ ac}$

1485 It is clear that if C is a propositional encoding, then so is $C \vee C$.

1486 Now consider the m rule:

$$\frac{S\{(C \wedge D) \vee (E \wedge F)\}}{S\{(C \vee E) \wedge (D \vee F)\}} \text{ m}$$

1489 Suppose that $(C \vee E) \wedge (D \vee F) = G^{\circ}$ for some G . Since $C \vee E$ cannot be Q_x (otherwise, the rule applied would be m'), G can be written as $G_1 \wedge G_2$ with $C \vee E = G_1^{\circ}$ and $D \vee F = G_2^{\circ}$.

1493 We have thus $G_i = \forall x_i H_i$ or $J_i \vee K_i (i = 1, 2)$.

1494 If $G_i = \forall x_i H_i$ for some i , then there will be a conjunction of U_x and some formula which can never be eliminated by the rules m, m' and ac. However, there exists no such conjunction in $A^{\circ*}$, which leads to a contradiction.

1498 Hence, G_i can be written as $J_i \vee K_i$ for $i = 1, 2$. We now have $(C \wedge D) \vee (E \wedge F) = ((J_1 \wedge J_2) \vee (K_1 \wedge K_2))^{\circ}$.

1500 Finally, consider the m' rule:

$$\frac{S\{(E_x \wedge C) \vee (E_x \wedge D)\}}{S\{E_x \wedge (C \vee D)\}} \text{ m}'$$

1502 Suppose that $E_x \wedge (C \vee D) = F^{\circ}$ for some F . It is clear that $F = \exists x G$ with $G^{\circ} = C \vee D$ for some G . We distinguish two cases:

- 1505 • $G = \forall y H$: in this case, $(E_x \wedge C) \vee (E_x \wedge D)$ has a subformula $(E_x \wedge U_y)$, which cannot be eliminated by the rules m, m' , ac. It is clear that $A^{\circ*}$ does not have a subformula of this form, which leads to a contradiction.
- 1509 • $G = G_1 \vee G_2$: in this case, $(E_x \wedge C) \vee (E_x \wedge D) = ((\exists x G_1) \vee (\exists x G_2))^{\circ}$.

1511 \square

- 1512 • If none of the four principal formulas in the premise is x or $x \vee F$ or $x \wedge F$ for some formula F and $x \in \text{VAR}$, then this instance of m can trivially be lifted, and we can proceed by induction hypothesis.
- 1516 • If exactly one of the four principal formulas in the premise is x for some $x \in \text{VAR}$, then this x is the

encoding of an existential in the premise and of an universal in the conclusion. This is impossible, as φ has to preserve existentials.

- If two of the four principal formulas in the premise are x for some $x \in \text{VAR}$, then we are in the following special case of (8):

$$\frac{\text{m} \frac{S\{(x \wedge C) \vee (x \wedge D)\}}{S\{(x \vee x) \wedge (C \vee D)\}}}{\text{ac}_x \frac{S\{(x \wedge C) \vee (x \wedge D)\}}{S\{x \wedge (C \vee D)\}}}$$

which can be lifted immediately to

$$\text{m}_{\exists} \frac{S\{(\exists x.C) \vee (\exists x.D)\}}{S\{\exists x.(C \vee D)\}}$$

- m/ac_x as in situation (8): We must have $R_1\{x\} \equiv x \vee E$ for some E and $R_2\{x\} \equiv x \vee F$ for some F with $R\{x\} \equiv x \vee E \vee F$. Otherwise, the application of ac_x would not be correct. We have the following four cases:
- E and F are both non-empty: Then (8) is (modulo omitted applications of \equiv):

$$\frac{\text{m} \frac{S\{((x \vee E) \wedge C) \vee ((x \vee F) \wedge D)\}}{S\{((x \vee E) \vee (x \vee F)) \wedge (C \vee D)\}}}{\text{ac}_x \frac{S\{((x \vee E) \wedge C) \vee ((x \vee F) \wedge D)\}}{S\{(x \vee E \vee F) \wedge (C \vee D)\}}}$$

which can be lifted to

$$\frac{\text{m} \frac{S\{((\forall x.E) \wedge C) \vee ((\forall x.F) \wedge D)\}}{S\{((\forall x.E) \vee (\forall x.F)) \wedge (C \vee D)\}}}{\text{m}_{\forall} \frac{S\{((\forall x.E) \wedge C) \vee ((\forall x.F) \wedge D)\}}{S\{(\forall x.(E \vee F)) \wedge (C \vee D)\}}}$$

1525 **[[Jui-Hsuan: maybe need some words to exclude the case in which C (or D) is a propositional variable.]]** 1526 **[[Lutz: shit. (you mean a “first order variable”) this actually can happen. then we have another m_{\exists}]]** 1527 1528

- E is empty and F is not: Then (8) becomes

$$\frac{\text{m} \frac{S\{(x \wedge C) \vee ((x \vee F) \wedge D)\}}{S\{(x \vee (x \vee F)) \wedge (C \vee D)\}}}{\text{ac}_x \frac{S\{(x \wedge C) \vee ((x \vee F) \wedge D)\}}{S\{(x \vee F) \wedge (C \vee D)\}}}$$

The conclusion is the propositional encoding of $S\{(\forall x.F) \wedge (C \vee D)\}$ and the premise is the propositional encoding of $S\{(\exists x.C) \vee ((\forall x.F) \vee D)\}$. Also note that no m-instance can break up the conjunction in $x \wedge C$ in the premise. Hence, φ maps an existential to a universal, which is ruled out by the definition. Hence, this case cannot occur.

- E is non-empty and F is empty: This case is similar to the previous subcase.
- E and F are both empty: Then (8) is

$$\frac{\text{m} \frac{S\{(x \wedge C) \vee (x \wedge D)\}}{S\{(x \vee x) \wedge (C \vee D)\}}}{\text{ac}_x \frac{S\{(x \wedge C) \vee (x \wedge D)\}}{S\{x \wedge (C \vee D)\}}}$$

which can be lifted immediately to

$$\text{m}_{\exists} \frac{S\{(\exists x.C) \vee (\exists x.D)\}}{S\{\exists x.(C \vee D)\}}$$