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 [7] for his famous lecture 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.

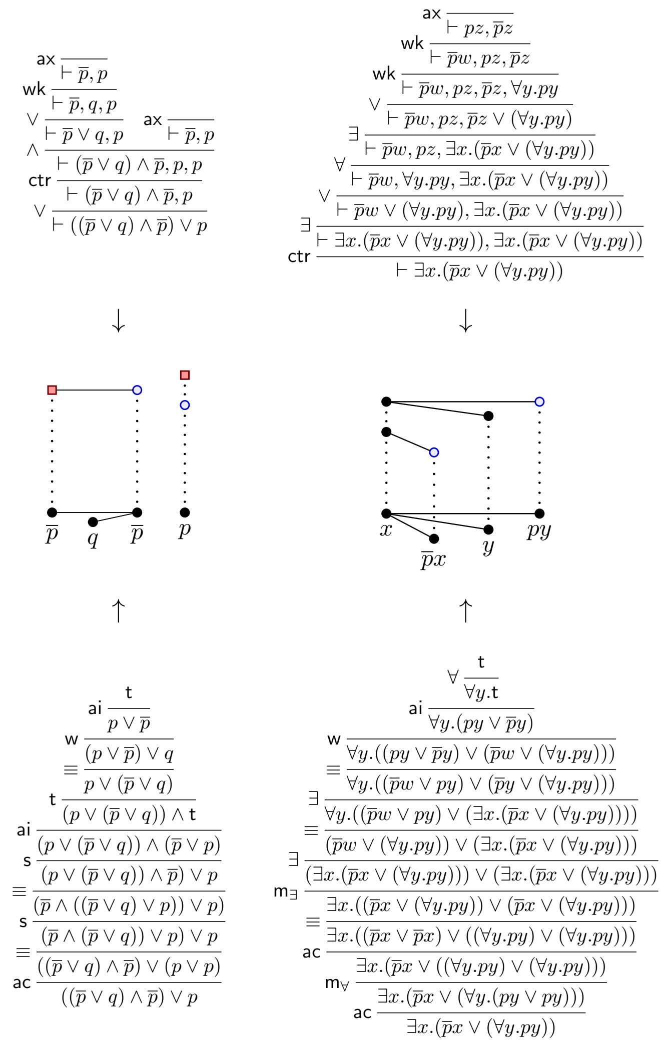


Fig. 1. Left: syntactic proofs in sequent calculus (above) and the calculus of structures (below) which translate to the same propositional combinatorial proof (centre). Right: syntactic proofs in sequent calculus (above) and the new calculus KS1 introduced in this paper (below), which translate to the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

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 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 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 $\mathsf{KS1}$, for first-order logic, that (a) reflects every combinatorial proof, i.e., there is a surjection from $\mathsf{KS1}$ 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 to 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.

II. Preliminaries: First-order Logic

A. Terms and Formulas

Fix pairwise disjoint countably infinite sets $\textsc{var} = \{x, y, z, \ldots\}$ of variables, $\textsc{fun} = \{f, g, \ldots\}$ of function symbols, and $\textsc{pred} = \{p, q, \ldots\}$ of predicate symbols. Each function symbol and each predicate symbol has a finite arity. The grammars below generate the set $\textsc{term}$ of ***terms***, denoted by $s, t, u, \ldots$, the set $\textsc{atom}$ of ***atoms***, denoted by $a, b, c, \ldots$, and the set $\textsc{form}$ of ***formulas***, denoted by $A, B, C, \ldots$:

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

where the arity of f and p is n. For better readability we often omit parentheses and write simply $ft_1 \ldots t_n$ or $pt_1 \ldots t_n$. We consider the truth constants t (*true*) and 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:

$$\begin{array}{lll} \bar{\bar{a}} = a & \bar{\mathsf{t}} = \mathsf{f} & \overline{p(t_1, \ldots, t_n)} = \bar{p}(t_1, \ldots, t_n) \\ & \bar{\mathsf{f}} = \mathsf{t} & \overline{\bar{p}(t_1, \ldots, t_n)} = p(t_1, \ldots, t_n) \\ & \overline{\exists x.A} = \forall x.\overline{A} & \overline{A \wedge B} = \overline{A} \vee \overline{B} \\ & \overline{\forall x.A} = \exists x.\overline{A} & \overline{A \vee B} = \overline{A} \wedge \overline{B} \end{array}$$

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

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 $\widehat{A}$.

A ***substitution*** is a function $\sigma \colon \textsc{var} \to \textsc{term}$ that is the identity almost everywhere. We denote substitutions as $\sigma = [x_1/t_1, \ldots, x_n/t_n]$, where $\sigma(x_i) = t_i$ for $i = 1..n$ and $\sigma(x) = x$ for all $x \notin \{x_1, \ldots, x_n\}$. 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 $\rho(x) \in \textsc{var}$ for all variables x.

B. Sequent Calculus $\mathsf{LK1}$

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

In this paper we use the sequent calculus $\mathsf{LK1}$, 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. $\mathsf{LK1}$ *is sound and complete for first-order logic.*

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

$$
\begin{array}{ccc}
\mathsf{ax}\,\dfrac{}{\vdash a, \overline{a}} & \vee\,\dfrac{\vdash \Gamma, A, B}{\vdash \Gamma, A \vee B} & \wedge\,\dfrac{\vdash \Gamma, A \quad \vdash B, \Delta}{\vdash \Gamma, A \wedge B, \Delta} \\[2ex]
\mathsf{t}\,\dfrac{}{\vdash \mathsf{t}} & \mathsf{f}\,\dfrac{\vdash \Gamma}{\vdash \Gamma, \mathsf{f}} & \mathsf{mix}\,\dfrac{\vdash \Gamma \quad \vdash \Delta}{\vdash \Gamma, \Delta} \\[2ex]
\exists\,\dfrac{\vdash \Gamma, A[x/t]}{\vdash \Gamma, \exists x.A} & \forall\,\dfrac{\vdash \Gamma, A}{\vdash \Gamma, \forall x.A}\,(x \text{ not free in } \Gamma) & \\[2ex]
\mathsf{ctr}\,\dfrac{\vdash \Gamma, A, A}{\vdash \Gamma, A} & \mathsf{wk}\,\dfrac{\vdash \Gamma}{\vdash \Gamma, A} &
\end{array}
$$

Fig. 2. Sequent calculi $\mathsf{LK1}$ (all rules) and $\mathsf{MLL1^X}$ (rules in the dashed box)

The linear fragment of $\mathsf{LK1}$, 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 $\mathsf{MLL1^X}$ (shown in Figure 2 in the dashed box).

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

$$
\mathsf{cut}\,\frac{\vdash \Gamma, A \quad \vdash \overline{A}, \Delta}{\vdash \Gamma, \Delta} \tag{1}
$$

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

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 $\{v, w\}$.

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*** $\varphi\colon \mathcal{G} \to \mathcal{H}$ is a function $\varphi\colon V_\mathcal{G} \to V_\mathcal{H}$ 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***.

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 $\hat{(\cdot)}\colon V_\mathcal{G} \to \hat{V_\mathcal{G}}$ 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*** $\varphi\colon \langle V_\mathcal{G}, E_\mathcal{G}\rangle \to \langle V_\mathcal{H}, E_\mathcal{H}\rangle$ is a function $\varphi\colon V_\mathcal{G} \to V_\mathcal{H}$ 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 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

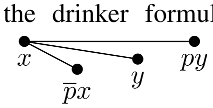
A cograph is ***logical*** if every vertex is labelled by either an atom or variable, and it has at least one atom-labelled vertex. An atom-labelled vertex is called a ***literal*** and a variable-labelled vertex is called a ***binder***. A binder labelled with x is called an ***x-binder***. The ***scope*** 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 in which all binders are legal. The ***binding graph*** of a fograph $\mathcal{G}$ is the directed graph $\vec{\mathcal{G}} = \langle V_\mathcal{G}, \{(b, l) \mid b \text{ binds } l\}\rangle$.

We define a mapping $[\![\cdot]\!]$ from formulas to (labelled) graphs, inductively as follows:

$$
\begin{array}{ll}
[\![a]\!] = \bullet a \quad \text{(for any atom } a) & \\
[\![A \vee B]\!] = [\![A]\!] + [\![B]\!] & [\![\exists x.A]\!] = \bullet x \times [\![A]\!] \\
[\![A \wedge B]\!] = [\![A]\!] \times [\![B]\!] & [\![\forall x.A]\!] = \bullet x + [\![A]\!]
\end{array}
$$

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

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

Lemma 6. *If* A *is a rectified formula then* $[\![A]\!]$ *is a fograph.*

Proof. That $[\![A]\!]$ is a logical cograph follows immediately from the definition and Theorem 3. The fact that every binder of $[\![A]\!]$ is legal can be proved by structural induction on A. $\square$

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

We define a congruence relation $\equiv$ on formulas by the following equations:

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

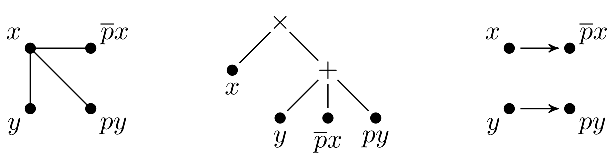
where x must not be free in B in the last two equations. Two formulas A and B are ***equivalent*** if $A \equiv B$. We have the following theorem:

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

$$A \equiv B \iff [\![A]\!] = [\![B]\!]$$

Proof. 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.



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

IV. First-order Combinatorial Proofs

A. Fonets

Two atoms are ***pre-dual*** if they are not $\mathtt{t}$ or $\mathtt{f}$, and their predicate symbols are dual (e.g. $p(x, y)$ and $\bar{p}(y, z)$) and two literals are ***pre-dual*** if their labels (atoms) are pre-dual. A ***linked fograph*** $\langle \mathcal{C}, \sim_{\mathcal{C}} \rangle$ is a coloured fograph $\mathcal{C}$ such that every colour (i.e., equivalence class of $\sim_{\mathcal{C}}$), called a ***link***, consists of two pre-dual literals, and every literal is either $\mathtt{t}$-labelled or in a link. Hence, in a linked fograph no vertex is labelled $\mathtt{f}$.

Let $\mathcal{C}$ be a linked fograph. The set of links can be seen as a unification problem by identifying dual predicate symbols. A ***dualizer*** of $\mathcal{C}$ is a substitution δ unifying all the links of $\mathcal{C}$. Since a first-order unification problem is either unsolvable or has a most general unifier, we can define the notion of ***most general dualizer***. A ***dependency*** is a pair $\{\bullet x, \bullet y\}$ of an existential binder $\bullet x$ and a universal binder $\bullet y$ such that the most general dualizer assigns to x a term containing y. A ***leap*** is either a link or a dependency. The ***leap graph*** $\mathcal{C}^{\mathsf{L}}$ 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 $\mathcal{C}^{\mathsf{L}}$.

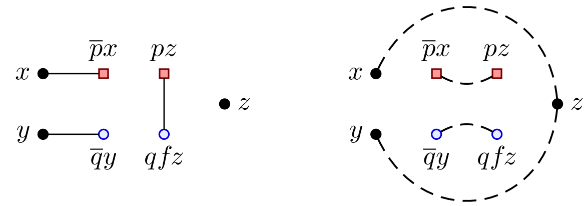


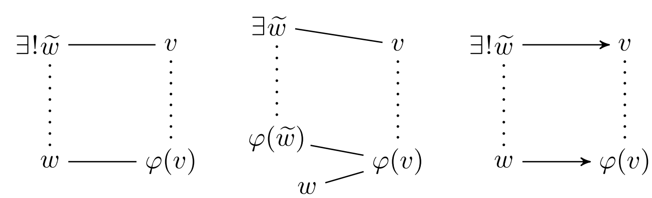
Fig. 3. A fonet (left) with dualizer $[x/z, y/fz]$ and its leap graph (right).

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 $\varphi\colon \langle V_{\mathcal{G}}, E_{\mathcal{G}} \rangle \to \langle V_{\mathcal{H}}, E_{\mathcal{H}} \rangle$ 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 $\varphi\colon \mathcal{G} \to \mathcal{H}$ 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 $\varphi\colon \mathcal{G} \to \mathcal{H}$ ***preserves existentials*** if for all existential binders b in $\mathcal{G}$, the binder $\varphi(b)$ is existential in $\mathcal{H}$.

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

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 labelling in the upper graph, as below-right.

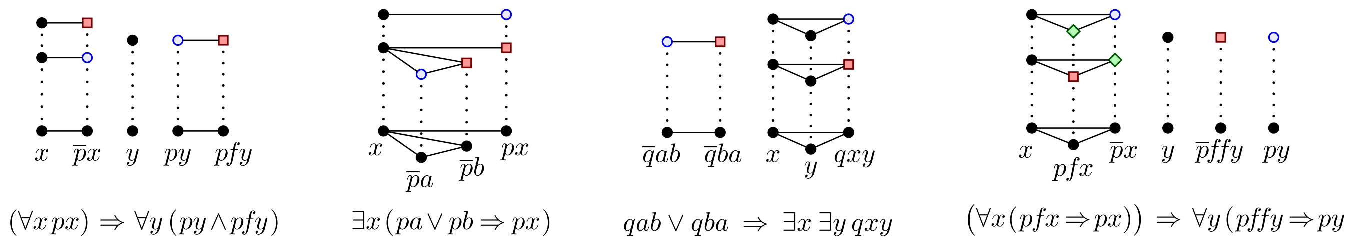


Fig. 4. Four combinatorial proofs, each shown above the formula proved. Here x and y are variables, f is a unary function symbol, a and b are constants (nullary function symbols), p is a unary predicate symbol, and q is a binary predicate symbol.

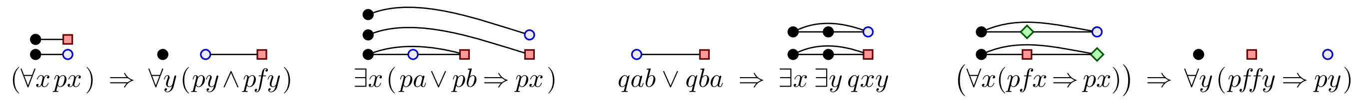
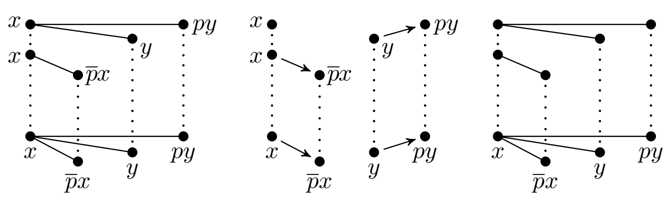


Fig. 5. Condensed forms of the four combinatorial proofs in Figure 4. We do not show the lower graph, and indicate the mapping by the position of the vertices of the upper graph.



Definition 13. A ***first-order combinatorial proof*** (***FOCP***) of a fograph $\mathcal{G}$ is a skew bifibration $\varphi\colon \mathcal{C} \to \mathcal{G}$ where $\mathcal{C}$ is a fonet. A ***first-order combinatorial proof*** of a formula A is a combinatorial proof of its graph $[\![A]\!]$.

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

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 ρ_φ which was forced to be the identity in [18].

$$\forall\,\frac{S\{\mathsf{t}\}}{S\{\forall x.\mathsf{t}\}} \qquad \mathsf{ai}\,\frac{S\{\mathsf{t}\}}{S\{a \vee \bar{a}\}} \qquad \mathsf{t}\,\frac{S\{A\}}{S\{A \wedge \mathsf{t}\}}$$

$$\mathsf{s}\,\frac{S\{A \wedge (B \vee C)\}}{S\{(A \wedge B) \vee C\}} \qquad \mathsf{mix}\,\frac{S\{A \wedge B\}}{S\{A \vee B\}}$$

$$\exists\,\frac{S\{A[x/t]\}}{S\{\exists x.A\}} \qquad \equiv\,\frac{S\{B\}}{S\{A\}}\ (\text{where } A \equiv B)$$

$$\mathsf{w}\,\frac{S\{A\}}{S\{A \vee B\}} \qquad \mathsf{m}\,\frac{S\{(A \wedge C) \vee (B \wedge D)\}}{S\{(A \vee B) \wedge (C \vee D)\}} \qquad \mathsf{ac}\,\frac{S\{a \vee a\}}{S\{a\}}$$

$$\mathsf{m}_\forall\,\frac{S\{(\forall x.A) \vee (\forall x.B)\}}{S\{\forall x.(A \vee B)\}} \qquad \mathsf{m}_\exists\,\frac{S\{(\exists x.A) \vee (\exists x.B)\}}{S\{\exists x.(A \vee B)\}}$$

$$\mathsf{w}_\forall\,\frac{S\{A\}}{S\{\forall x.A\}}\ (x \text{ not free in } A) \qquad \mathsf{c}_\forall\,\frac{S\{\forall x.\forall x.A\}}{S\{\forall x.A\}}$$

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

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 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 $\mathsf{KS1}$ that we introduce in this paper. It is a slight variation of the systems presented by Brünnler [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, (iii) an explicit rule for the equivalence, and (iv) new inference rules $\mathsf{w}_\forall$ and $\mathsf{c}_\forall$. The reason behind these design choices is to obtain the correspondence with combinatorial proofs and the full completeness result.

We consider here only the cut-free fragment, as cut-elimination for deep inference systems has already been discussed elsewhere (e.g. [22], [31]).[1] As with the sequent system $\mathsf{LK1}$, we also need for $\mathsf{KS1}$ the *linear fragment*, $\mathsf{MLS1^X}$, and that is shown in Figure 6 in the dashed box.

We write $\mathsf{S}\,\begin{array}{c} B \\ \Big\Vert\,\Phi \\ A \end{array}$ to denote a derivation Φ from B to A using

[1] In 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.

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

We will for some results also employ the general (non-atomic) version of the contraction rule:

$$\mathsf{c}\,\frac{S\{A \vee A\}}{S\{A\}} \tag{3}$$

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. $\mathsf{KS1}$ *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* $\begin{array}{c}\mathsf{t}\\ \mathsf{KS1}\,\Vert\,\Phi\\ A\end{array}$ *there are* f*-free formulas* $A_1, \ldots, A_5$ *and a derivation*

$$\begin{array}{rc}
& \mathsf{t} \\
\{\forall,\mathsf{ai},\mathsf{t}\} & \Vert \\
& A_5 \\
\{\mathsf{s},\mathsf{mix},\equiv\} & \Vert \\
& A_4 \\
\{\exists\} & \Vert \\
& A_3 \\
\{\mathsf{m},\mathsf{m}_\forall,\mathsf{m}_\exists,\equiv\} & \Vert \\
& A_2 \\
\{\mathsf{ac},\mathsf{c}_\forall\} & \Vert \\
& A_1 \\
\{\mathsf{w},\mathsf{w}_\forall,\equiv\} & \Vert \\
& A
\end{array}$$

This theorem is stronger than the existing decompositions for first-order logic, which either separate 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 rules $\mathsf{w}_\forall$ and $\mathsf{c}_\forall$ in system $\mathsf{KS1}$, as these rules are derivable with the other rules. However, they are needed to obtain this decomposition. Figure 7 shows an example of a decomposed derivation in $\mathsf{KS1}$ of the formula $(\exists x.\bar{p}x) \vee (\forall y.(py \wedge pfy))$.

A weaker version of Theorem 17 will also be useful:

Theorem 18. *For every derivation* $\begin{array}{c}\mathsf{t}\\ \mathsf{KS1}\,\Vert\,\Phi\\ A\end{array}$ *there is a formula* A' *with no occurrence of* f *and a derivation*

$$\begin{array}{rc}
& \mathsf{t} \\
\mathsf{MLS1}^{\mathsf{X}} & \Vert \\
& A' \\
\{\mathsf{w},\mathsf{c},\equiv\} & \Vert \\
& A
\end{array}$$

Here A' corresponds to A_3 of Theorem 17.

$$\begin{array}{rc}
& \mathsf{t} \\ \cline{2-2}
\forall & \forall y.\mathsf{t} \\ \cline{2-2}
\mathsf{t} & \forall y.(\mathsf{t} \wedge \mathsf{t}) \\ \cline{2-2}
\mathsf{ai} & \forall y.((\bar{p}y \vee py) \wedge \mathsf{t}) \\ \cline{2-2}
\mathsf{ai} & \forall y.((\bar{p}y \vee py) \wedge (pfy \vee \bar{p}fy)) \\ \cline{2-2}
\equiv & \forall y.(\bar{p}y \vee (py \wedge (pfy \vee \bar{p}fy))) \\ \cline{2-2}
\mathsf{s} & \forall y.(\bar{p}y \vee ((py \wedge pfy) \vee \bar{p}fy)) \\ \cline{2-2}
\equiv & \forall y.((\bar{p}y \vee \bar{p}fy) \vee (py \wedge pfy)) \\ \cline{2-2}
\exists & \forall y.((\bar{p}y \vee (\exists x.\bar{p}x)) \vee (py \wedge pfy)) \\ \cline{2-2}
\exists & \forall y.(((\exists x.\bar{p}x) \vee (\exists x.\bar{p}x)) \vee (py \wedge pfy)) \\ \cline{2-2}
\equiv & ((\exists x.\bar{p}x) \vee (\exists x.\bar{p}x)) \vee (\forall y.(py \wedge pfy) \\ \cline{2-2}
\mathsf{m}_\exists & (\exists x.(\bar{p}x \vee \bar{p}x) \vee (\forall y.(py \wedge pfy)) \\ \cline{2-2}
\mathsf{ac} & (\exists x.\bar{p}x) \vee (\forall y.(py \wedge pfy))
\end{array}$$

Fig. 7. Example derivation in decomposed form of Theorem 17

We now establish the connection between derivations in $\mathsf{KS1}$ and combinatorial proofs.

Theorem 19. *Let* $\varphi\colon \mathcal{C} \to \mathcal{A}$ *be a combinatorial proof and let* A *be a formula with* $\mathcal{A} = [\![A]\!]$*. Then there is a derivation*

$$\begin{array}{rc}
& \mathsf{t} \\
\mathsf{MLS1}^{\mathsf{X}} & \Vert\,\Phi_1 \\
& A' \\
\{\mathsf{w},\mathsf{w}_\forall,\mathsf{ac},\mathsf{c}_\forall,\mathsf{m},\mathsf{m}_\forall,\mathsf{m}_\exists,\equiv\} & \Vert\,\Phi_2 \\
& A
\end{array} \tag{4}$$

for some $A' \equiv C\rho_\varphi$ *where* C *is a formula with* $[\![C]\!] = \mathcal{C}$ *and* ρ_φ *is the variable renaming substitution induced by* φ*. Conversely, whenever we have a derivation as in* (4) *above, such that* f *does not occur in* A'*, then there is a combinatorial proof* $\varphi\colon \mathcal{C} \to [\![A]\!]$ *such that* $\mathcal{C} = [\![\widehat{A'}]\!]$.

Furthermore, in the proof of Theorem 19, 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. To give an example, consider the derivation in Figure 7 which corresponds to the left-most combinatorial proof in Figures 4 and 5.

Thus, combinatorial proofs are closely related to derivations of the form (4), and since by Theorem 17 every derivation can be transformed into that form, we can say that combinatorial proofs provide a canonical proof representation for first-order logic, similarly to what proof nets are for linear logic [33].

Finally, Theorems 16, 17 and 19 imply Theorem 14, which means that we have here an alternative proof of the soundness and completeness for first-order combinatorial proofs which is simpler than the one given in [18], and improves with completeness being full (a surjection from syntactic KS1 proofs onto combinatorial proofs).

VII. TRANSLATING BETWEEN LK1 AND KS1

We prove Theorems 16, 17, and 18, mainly by translating derivations to and from the sequent calculus, and by rule permutation arguments.

A. The Linear Fragments $\mathsf{MLL1^X}$ and $\mathsf{MLS1^X}$

We show that $\mathsf{MLL1^X}$ and $\mathsf{MLS1^X}$ are equivalent.

Lemma 20. *If $\vdash \Gamma$ is provable in $\mathsf{MLL1^X}$ then $\bigvee(\Gamma)$ is provable in $\mathsf{MLS1^X}$.*

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

$$\equiv \dfrac{\dfrac{\forall \dfrac{\mathsf{t}}{\forall x.\mathsf{t}}}{\forall x. \bigvee(\Delta) \vee A}\,\mathsf{MLS1^X}\Big\Vert}{\bigvee(\Delta) \vee \forall x.A}$$

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

Lemma 21. *Let $\mathsf{r}\,\dfrac{S\{A\}}{S\{B\}}$ be an inference rule in $\mathsf{MLS1^X}$. Then the sequent $\vdash \overline{A}, B$ is provable in $\mathsf{MLL1^X}$.*

Proof. This is a straightforward exercise. □

Lemma 22. *Let A, B be formulas, and let $S\{\cdot\}$ be a (positive) context. If $\vdash \overline{A}, B$ is provable in $\mathsf{MLL1^X}$, then so is $\vdash \overline{S\{A\}}, S\{B\}$.*

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

Lemma 23. *If a formula C is provable in $\mathsf{MLS1^X}$ then $\vdash C$ is provable in $\mathsf{MLL1^X}$.*

Proof. We proceed by induction on the number of inference steps in the proof of C in $\mathsf{MLS1^X}$. Consider the bottommost rule instance $\mathsf{r}\,\dfrac{S\{A\}}{S\{B\}}$. By induction hypothesis we have a $\mathsf{MLL1^X}$ proof Π of $\vdash S\{A\}$. By Lemmas 21 and 22, we have a $\mathsf{MLL1^X}$ proof of $\vdash \overline{S\{A\}}, S\{B\}$. We can compose them via cut:

$$\mathsf{cut}\,\dfrac{\vdash S\{A\} \quad \vdash \overline{S\{A\}}, S\{B\}}{\vdash S\{B\}}$$

and then apply Theorem 2. □

B. Contraction and Weakening

The first observation here is that Lemmas 20–23 from above also hold for LK1 and KS1. We therefore immediately have:

Theorem 24. *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 25. *The c-rule is derivable in $\{\mathsf{ac}, \mathsf{m}, \mathsf{m}_\forall, \mathsf{m}_\exists, \equiv\}$.*

Proof. This can be shown by a straightforward induction on A (for details, see e.g. [29]). □

Lemma 26. *$\mathsf{w}_\forall, \mathsf{c}_\forall, \mathsf{m}, \mathsf{m}_\forall, \mathsf{m}_\exists$ are derivable in $\{\mathsf{w}, \mathsf{c}, \equiv\}$.*

Proof. We only show the cases for $\mathsf{w}_\forall$ and $\mathsf{c}_\forall$ (for the others see [29]):

$$\mathsf{c}\,\dfrac{\equiv\,\dfrac{\mathsf{w}\,\dfrac{A}{A \vee (\forall x.A)}}{\forall x.(A \vee A)}}{\forall x.A} \qquad \mathsf{c}\,\dfrac{\equiv\,\dfrac{\mathsf{w}\,\dfrac{\forall x.\forall x.A}{\forall x.((\forall x.A) \vee A)}}{(\forall x.A) \vee (\forall x.A)}}{\forall x.A} \tag{5}$$

where in the first derivation, x is not free in A, and in the second one not free in $\forall x.A$. □

Lemma 27. *Let A and B be formulas. Then*

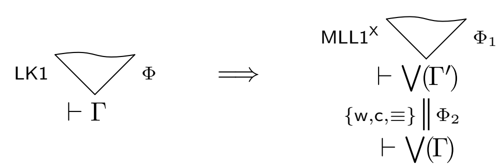
$$\begin{array}{c} A \\ \{\mathsf{w},\mathsf{c},\equiv\}\Big\Vert \\ B \end{array} \iff \begin{array}{c} A \\ \{\mathsf{w},\mathsf{w}_\forall,\mathsf{ac},\mathsf{c}_\forall,\mathsf{m},\mathsf{m}_\forall,\mathsf{m}_\exists,\equiv\}\Big\Vert \\ B \end{array}$$

Proof. Immediately from Lemmas 25 and 26. □

Remark 28. Observe that Lemma 27 would also hold with the rules $\mathsf{w}_\forall$ and $\mathsf{c}_\forall$ removed.

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' not containing any f, such that there is a derivation as shown on the right below:*



Proof. First, we can replace every instance of the f-rule in Φ by wk. Then the instances of wk and ctr are replaced by w and c, which can then be permuted down. Details are in Appendix A. □

Lemma 30. *For every derivation $\begin{array}{c} \mathsf{t} \\ \mathsf{MLS1^X}\Big\Vert \\ A \end{array}$ there are formulas A' and A'' such that*

$$\begin{array}{c} \mathsf{t} \\ \{\forall,\mathsf{ai},\mathsf{t}\}\Big\Vert \\ A'' \\ \{\mathsf{s},\mathsf{mix},\equiv\}\Big\Vert \\ A' \\ \{\exists\}\Big\Vert \\ A \end{array}$$

Proof. First, observe that the $\exists$ rule can be permuted under 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 all rules in $\{\forall, \mathsf{ai}, \mathsf{t}\}$ can be permuted over the rules in $\{\mathsf{s}, \mathsf{mix}, \equiv\}$, which is straightforward (see [35] for details). $\square$

Lemma 31. *For every derivation* $\begin{array}{c} A \\ {\scriptstyle\{\mathsf{w},\mathsf{w}_\forall,\mathsf{ac},\mathsf{c}_\forall,\mathsf{m},\mathsf{m}_\forall,\mathsf{m}_\exists,\equiv\}}\Big\Vert \\ B \end{array}$ *there are formulas A' and B' such that*

$$\begin{array}{c} A \\ {\scriptstyle\{\mathsf{m},\mathsf{m}_\forall,\mathsf{m}_\exists,\equiv\}}\Big\Vert \\ A' \\ {\scriptstyle\{\mathsf{ac},\mathsf{c}_\forall\}}\Big\Vert \\ B' \\ {\scriptstyle\{\mathsf{w},\mathsf{w}_\forall,\equiv\}}\Big\Vert \\ B \end{array}$$

Proof. We first permute all instances of w and $\mathsf{w}_\forall$ to the bottom of the derivation and then permute in a second step the rules c and $\mathsf{c}_\forall$ below $\{\mathsf{m}, \mathsf{m}_\forall, \mathsf{m}_\exists\}$. This involves a tedious but straightforward case analysis. However, unlike most other rule permutations in this paper, this has not been done before in the deep inference literature. For this reason, we give the full case analysis in Appendix B. Note that this Lemma is the reason for the presence of the rules $\mathsf{w}_\forall$ and $\mathsf{c}_\forall$, as without them the permutation cases in (5) could not be resolved. $\square$

We can now complete the proof of Theorems 17 and 18.

Proof of Theorem 18. Assume we have a proof of A in $\mathsf{KS1}$. By Theorem 24 we have a proof of $\vdash A$ in $\mathsf{LK1}$ to which we can apply Theorem 29. Finally, we apply Lemma 20 to get the desired shape. $\square$

Proof of Theorem 17. Assume we have a proof of A in $\mathsf{KS1}$. We first apply Theorem 18, and then Lemma 30 to the upper half and Lemmas 27 and 31 to the lower half. $\square$

VIII. Fonets and Linear Proofs

A. *From* $\mathsf{MLL1}^{\mathsf{X}}$ *Proofs to Fonets*

Let Π be a $\mathsf{MLL1}^{\mathsf{X}}$ proof of a rectified sequent $\vdash \Gamma$ not containing f. We now show how Π is translated into a linked fograph $[\![\Pi]\!] = \langle [\![\Gamma]\!], \sim_\Pi \rangle$. We proceed inductively, making a case analysis on the last rule in Π. At the same time we are constructing a dualizer δ_Π, so that in the end we can conclude that $[\![\Pi]\!]$ is in fact a fonet.

1) Π is $\mathsf{ax}\,\dfrac{}{\vdash a, \bar{a}}$: Then the only link is $\{a, \bar{a}\}$, and δ_Π is empty.
2) Π is $\mathsf{t}\,\dfrac{}{\vdash \mathsf{t}}$: Then $\sim_\Pi$ and δ_Π are both empty.
3) The last rule in Π is $\mathsf{mix}\,\dfrac{\vdash \Gamma' \quad \vdash \Gamma''}{\vdash \Gamma', \Gamma''}$: By induction hypothesis, we have proofs Π' and Π'' of Γ' and Γ'', respectively. We have $[\![\Gamma]\!] = [\![\Gamma']\!] + [\![\Gamma'']\!]$ and we can let $\sim_\Pi = \sim_{\Pi'} \cup \sim_{\Pi''}$ and $\delta_\Pi = \delta_{\Pi'} \cup \delta_{\Pi''}$.
4) The last rule in Π is $\vee\,\dfrac{\vdash \Gamma_1, A, B}{\vdash \Gamma_1, A \vee B}$: By induction hypothesis, there is a proof Π' of $\Gamma' = \Gamma_1, A, B$. We have $[\![\Gamma]\!] = [\![\Gamma']\!]$ and let $\sim_\Pi = \sim_{\Pi'}$ and $\delta_\Pi = \delta_{\Pi'}$.
5) The last rule in Π is $\wedge\,\dfrac{\vdash \Gamma_1, A \quad \vdash B, \Gamma_2}{\vdash \Gamma_1, A \wedge B, \Gamma_2}$: By induction hypothesis, we have proofs Π' and Π'' of $\Gamma' = \Gamma_1, A$ and $\Gamma'' = B, \Gamma_2$, respectively. We have $[\![\Gamma]\!] = [\![\Gamma_1]\!] + ([\![A]\!] \times [\![B]\!]) + [\![\Gamma_2]\!]$ and we let $\sim_\Pi = \sim_{\Pi'} \cup \sim_{\Pi''}$ and $\delta_\Pi = \delta_{\Pi'} \cup \delta_{\Pi''}$.
6) The last rule in Π is $\exists\,\dfrac{\vdash \Gamma_1, A[x/t]}{\vdash \Gamma_1, \exists x.A}$: By induction hypothesis, there is a proof Π' of $\Gamma' = \Gamma_1, A[x/t]$. For each atom in $\Gamma' = \Gamma_1, A[x/t]$, there is a corresponding atom in $\Gamma = \Gamma_1, \exists x.A$. We can therefore define the linking $\sim_\Pi$ from the linking $\sim_{\Pi'}$ via this correspondence. Then, we let δ_Π be $\delta_{\Pi'} + [x/t]$. Since Γ is rectified x does not yet occur in $\delta_{\Pi'}$. Hence δ_Π is a dualizer of $[\![\Pi]\!]$.
7) The last rule in Π is $\forall\,\dfrac{\vdash \Gamma_1, A}{\vdash \Gamma_1, \forall x.A}$ (x not free in Γ_1) : By induction hypothesis, there is a proof Π' of $\Gamma' = \Gamma_1, A$, which has the same atoms as in $\Gamma = \Gamma_1, \forall x.A$. Hence, we can let $\sim_\Pi = \sim_{\Pi'}$ and $\delta_\Pi = \delta_{\Pi'}$.

Theorem 32. *If Π is a $\mathsf{MLL1}^{\mathsf{X}}$ proof of a rectified f-free sequent $\vdash \Gamma$, then $[\![\Pi]\!]$ is a fonet and δ_Π a dualizer for it.*

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

B. *From* $\mathsf{MLS1}^{\mathsf{X}}$ *Proofs to Fonets*

There is a more direct path from a $\mathsf{MLL1}^{\mathsf{X}}$ proof Π of a rectified sequent Γ to the linked fograph $[\![\Pi]\!]$: simply take the fograph $[\![\Gamma]\!]$, and let the equivalence classes of $\sim_\Pi$ 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 $\mathsf{MLS1}^{\mathsf{X}}$ derivation into fonets, we employ exactly that direct path.

First observe that in a derivation in $\mathsf{MLS1}^{\mathsf{X}}$ where the conclusion is rectified, every line is also rectified, as the only rules involving bound variables are $\forall$ and $\exists$ which both remove a binder. Therefore, we can call such a derivation ***rectified***, and for a non-rectified $\mathsf{MLS1}^{\mathsf{X}}$ derivation Φ we can define its ***rectification*** $\widehat{\Phi}$ inductively, by rectifying each line, proceeding step-wise from conclusion to premise.[2]

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

A rectified derivation $\begin{array}{c}\mathsf{t}\\ \mathsf{MLS1}^{\mathsf{X}}\Big\Vert\,\Phi\\ A\end{array}$ determines a substitution 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 $\sim_\Phi$ 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 $\sim_\Phi$ the ***linking*** of Φ, and define $[\![\Phi]\!] = \langle [\![A]\!], \sim_\Phi \rangle$.

Theorem 33. *Let* $\begin{array}{c}\mathsf{t}\\ \mathsf{MLS1}^{\mathsf{X}}\Big\Vert\,\Phi\\ A\end{array}$ *be a rectified derivation where A is f-free. Then $[\![\Phi]\!]$ is a fonet and δ_Φ a dualizer for it.*

For proving this theorem, we have to show that no inference rule in $\mathsf{MLS1}^{\mathsf{X}}$ can introduce a bimatching. To simplify the argument, we introduce the ***frame*** [36] of the linked 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 $[\![C]\!] = \mathcal{C}$, to which we exhaustively apply the following subformula rewriting steps, to obtain a sequent $C^\star$:

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 $\overline{q}_{i,j} \wedge \exists x_i.A$, 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 $pt_1 \dots t_n$ (resp. $\overline{p}t_1 \dots t_n$) with a nullary predicate symbol p (resp. $\overline{p}$)

Then $\sim_{C^\star}$ consists of the pairs induced by $\sim_{\mathcal{C}}$ and the new pairs $\{q_{i,j}, \overline{q}_{i,j}\}$ introduced in step 1 above. We call $C^\star$ the ***frame*** of C and we define the ***frame*** of $\mathcal{C}$, denoted $\mathcal{C}^\star$, as $\langle [\![C^\star]\!], \sim_{C^\star} \rangle$.

Lemma 34. *If a linked fograph $\mathcal{C}$ has an induced bimatching then so does its frame $\mathcal{C}^\star$.*

Proof. Immediately from the construction of the frame. $\square$

Proof of Theorem 33. From Φ we construct a derivation $\Phi^\star$ of $A^\star$ in the propositional fragment of $\mathsf{MLS1}^{\mathsf{X}}$, such that $[\![\Phi^\star]\!] = [\![\Phi]\!]^\star$. The rules $\mathsf{ai}, \mathsf{t}, \mathsf{mix}$ 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]

$$\forall\,\frac{S\{\mathsf{t}\}}{S\{\forall y_j.\mathsf{t}\}} \quad\rightsquigarrow\quad \begin{array}{c}\mathsf{t}\\ {\scriptstyle\{\mathsf{ai},\mathsf{t}\}}\Big\Vert\,{\scriptstyle\Psi_1}\\ S\{(q_{h_1,j} \vee \overline{q}_{h_1,j}) \wedge \cdots \wedge (q_{h_n,j} \vee \overline{q}_{h_n,j}) \wedge \mathsf{t}\}\\ {\scriptstyle\{\mathsf{s},\equiv\}}\Big\Vert\,{\scriptstyle\Psi_2}\\ S\{q_{h_1,j} \vee \cdots \vee q_{h_n,j} \vee (\overline{q}_{h_1,j} \wedge \cdots \wedge \overline{q}_{h_n,j} \wedge \mathsf{t})\}\end{array}$$

where $h_1, \dots, h_n$ range over the indices of the existential binders dependent on that y_j. It is easy to see how Ψ_1 is constructed. The construction of Ψ_2, using s and $\equiv$, is standard, see, e.g. [37], [34], [38], [35]. Then, every occurrence of $\forall y_j.F$ is replaced by $q_{h_1,j} \vee \cdots \vee q_{h_n,j} \vee (\overline{q}_{h_1,j} \wedge \cdots \wedge \overline{q}_{h_n,j} \wedge F)$ in the derivation below that $\forall$-instance. Now, observe that all instances of the $\exists$-rule introducing x_i dependent on y_j must occur below in the derivation (otherwise Φ would not be rectified). Now consider such an instance $\exists\,\dfrac{S\{B[x_i/t]\}}{S\{\exists x_i.B\}}$. Its 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

$$\begin{array}{c} S_0\{\overline{q}_{i,k_1} \wedge S_1\{\overline{q}_{i,k_2} \wedge \cdots S_{l-1}\{\overline{q}_{i,k_l} \wedge S_l\{B'\}\}\cdots\}\}\\ {\scriptstyle\{\mathsf{s},\equiv\}}\Big\Vert\,{\scriptstyle\Psi_3}\\ S_0\{S_1\{\cdots S_{l-1}\{S_l\{\overline{q}_{i,k_1} \wedge \overline{q}_{i,k_2} \wedge \cdots q_{i,k_l} \wedge B'\}\}\cdots\}\}\end{array}$$

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

Doing this to all instances of the rules $\forall$ and $\exists$ in Φ yields indeed a propositional derivation $\Phi^\star$ with $[\![\Phi^\star]\!] = [\![\Phi]\!]^\star$. It has been shown by Retoré [39] and rediscovered by Straßburger [35] that $[\![\Phi^\star]\!] = \langle [\![C^\star]\!], \sim_{\Phi^\star} \rangle$ cannot contain an induced bimatching. By Lemma 34, $[\![\Phi]\!]$ does not have an induced bimatching either. Furthermore, it follows from the definition of δ_Φ that it is a dualizer for $[\![\Phi]\!]$. $\square$

Remark 35. There is an alternative path of proving Theorem 33 by translating Φ to an $\mathsf{MLL1}^{\mathsf{X}}$-proof Π, observing that this process preserves the linking and the dualizer. However, for this, we have to extend the construction from the previous subsection 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 [36].

C. From Fonets to $\mathsf{MLL1}^{\mathsf{X}}$ *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 $\mathsf{MLL1}^{\mathsf{X}}$ such that $[\![\Pi]\!] = \langle \mathcal{C}, \sim_{\mathcal{C}} \rangle$. In the proof net literature, this operation is also called *sequentialization*. The basic idea behind our sequentialization is to use the frame of $\mathcal{C}$, to which we can apply the *splitting tensor theorem*, and then reconstruct the sequent proof Π.

Let Γ be a propositional sequent and $\sim_\Gamma$ be a linking for $[\![\Gamma]\!]$. 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 $[\![\Gamma', A]\!]$ and $\sim_2$ is a linking for $[\![B, \Gamma'']\!]$, i.e., removing the $\wedge$ from $A \wedge B$ splits the linked fograph $\langle [\![\Gamma]\!], \sim_\Gamma \rangle$ into two fographs. We say that $\langle [\![\Gamma]\!], \sim_\Gamma \rangle$ is ***mixed***

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

iff $\Gamma = \Gamma', \Gamma''$ and $\sim_\Gamma = \sim_1 \cup \sim_2$, such that $\sim_1$ is a linking for $[\![\Gamma']\!]$ and $\sim_2$ is a linking for $[\![\Gamma'']\!]$. Finally, $\langle [\![\Gamma]\!], \sim_\Gamma \rangle$ is ***splittable*** if it is mixed or has a splitting tensor.

Theorem 36. *Let Γ be a f-free propositional sequent containing only atoms and $\wedge$-formulas, and $\sim_\Gamma$ be a linking for $[\![\Gamma]\!]$. If $\langle [\![\Gamma]\!], \sim_\Gamma \rangle$ does not induce a bimatching then it is splittable.*

This is the well-known splitting-tensor-theorem [19], [40], adapted for the presence of mix. In the setting of linked cographs, it has first been proved by Retoré [41], [42] and then rediscovered by Hughes [9]. We use it now for our sequentialization:

Theorem 37. *Let $\langle \mathcal{C}, \sim_\mathcal{C} \rangle$ be a fonet, and let Γ be a sequent with $[\![\Gamma]\!] = \mathcal{C}$. Then there is an $\mathsf{MLL1^X}$-proof Π of Γ, such that $[\![\Pi]\!] = \langle \mathcal{C}, \sim_\mathcal{C} \rangle$.*

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 $\mathsf{MLL1^X}$ 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 $\langle [\![\Gamma^\star]\!], \sim_{\Gamma^\star} \rangle$ 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 $\Gamma^\star$ 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. $\square$

D. *From Fonets to* $\mathsf{MLS1^X}$ *Proofs*

We can now straightforwardly obtain the same result for $\mathsf{MLS1^X}$:

Theorem 38. *Let $\langle \mathcal{C}, \sim_\mathcal{C} \rangle$ be a fonet, and let C be a formula with $[\![C]\!] = \mathcal{C}$. Then there is a derivation $\mathsf{MLS1^X} \overset{\mathsf{t}}{\underset{C}{\Big\|}} \Phi$ such that $[\![\Phi]\!] = \langle \mathcal{C}, \sim_\mathcal{C} \rangle$.*

Proof. We apply Theorem 37 to obtain a sequent proof Π of $\vdash C$ with $[\![\Pi]\!] = \langle \mathcal{C}, \sim_\mathcal{C} \rangle$. Then we apply Lemma 20, observing that the translation from $\mathsf{MLL1^X}$ to $\mathsf{MLS1^X}$ preserves linking and dualizer. $\square$

Remark 39. Note that it is also possible to do a direct "sequentialization" into the deep inference system $\mathsf{MLS1^X}$, using the techniques presented in [35] and [43].

IX. Skew Bifibrations and Resource Management

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

$$\widehat{\mathsf{c}_\forall}\,\frac{S\{\forall y.\forall x.Ax\}}{S\{\forall x.Ax\}} \qquad \widehat{\mathsf{m}_\forall}\,\frac{S\{(\forall y.Ay) \vee (\forall z.Bz)\}}{S\{\forall x.(Ax \vee Bx)\}} \qquad \widehat{\mathsf{m}_\exists}\,\frac{S\{(\exists y.Ay) \vee (\exists z.Bz)\}}{S\{\exists x.(Ax \vee Bx)\}}$$

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 $\widehat{\mathsf{m}_\forall}$ or $\widehat{\mathsf{m}_\exists}$ (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 $\mathsf{m}_\forall$ and $\mathsf{m}_\exists$ as special cases of $\widehat{\mathsf{m}_\forall}$ and $\widehat{\mathsf{m}_\exists}$, respectively. And similarly, if $x = y$ then $\mathsf{c}_\forall$ is a special case of $\widehat{\mathsf{c}_\forall}$.

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

For each instance $\mathsf{r}\,\dfrac{Q}{P}$ of an inference rule in $\{\mathsf{w}, \mathsf{w}_\forall, \mathsf{ac}, \widehat{\mathsf{c}_\forall}, \mathsf{m}, \widehat{\mathsf{m}_\forall}, \widehat{\mathsf{m}_\exists}, \equiv\}$ we can define the ***induced map*** $\lfloor \mathsf{r} \rfloor \colon V_{[\![Q]\!]} \to V_{[\![P]\!]}$ which acts as the identity for $\mathsf{r} \in \{\mathsf{m}, \equiv\}$ and as the canonical injection for $\mathsf{r} \in \{\mathsf{w}, \mathsf{w}_\forall\}$. For $\mathsf{r} = \mathsf{ac}$ it maps the vertices corresponding to the two atoms in the premise to the vertex of the contracted atom in the conclusion, and for $\mathsf{r} \in \{\widehat{\mathsf{c}_\forall}, \widehat{\mathsf{m}_\forall}, \widehat{\mathsf{m}_\exists}\}$ it maps the two vertices corresponding to the quantifiers in the premise to the one in the conclusion (and acts as the identity on all other vertices). For a derivation Φ in $\{\mathsf{w}, \mathsf{w}_\forall, \mathsf{ac}, \widehat{\mathsf{c}_\forall}, \mathsf{m}, \widehat{\mathsf{m}_\forall}, \widehat{\mathsf{m}_\exists}, \equiv\}$ we can then define the ***induced map*** $\lfloor \Phi \rfloor$ as the composition of the induced maps of the rule instances in Φ.

Lemma 40. *Let $\{\mathsf{w}, \mathsf{w}_\forall, \mathsf{ac}, \mathsf{c}_\forall, \mathsf{m}, \mathsf{m}_\forall, \mathsf{m}_\exists, \equiv\} \overset{A}{\underset{B}{\Big\|}} \Phi$ be given. Then there is a rectified derivation $\{\mathsf{w}, \mathsf{w}_\forall, \mathsf{ac}, \widehat{\mathsf{c}_\forall}, \mathsf{m}, \widehat{\mathsf{m}_\forall}, \widehat{\mathsf{m}_\exists}, \equiv\} \overset{\widehat{A}}{\underset{\widehat{B}}{\Big\|}} \widehat{\Phi}$, such that the induced maps $\lfloor \Phi \rfloor \colon [\![A]\!] \to [\![B]\!]$ and $\lfloor \widehat{\Phi} \rfloor \colon [\![\widehat{A}]\!] \to [\![\widehat{B}]\!]$ are equal up to a variable renaming of the vertex labels.*

Proof. Immediate from the definition. $\square$

A. From Contraction and Weakening to Skew Bifibrations

Lemma 41. *Let* $\begin{array}{c} A \\ {\scriptstyle \{\mathsf{w},\mathsf{w}_\forall,\mathsf{ac},\widehat{\mathsf{c}_\forall},\mathsf{m},\widehat{\mathsf{m}_\forall},\widehat{\mathsf{m}_\exists},\equiv\}} \Big\Vert {\scriptstyle \Phi} \\ B \end{array}$ *be a rectified derivation. Then the induced map* $\lfloor\Phi\rfloor\colon [\![A]\!] \to [\![B]\!]$ *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 $[\![A^\circ]\!] = [\![A]\!]$. For this, we introduce new propositional variables that have the same names as the (first-order) variables $x \in \mathsf{VAR}$. Then A° is defined inductively by:

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

Lemma 42. *For every formula A, we have* $[\![A^\circ]\!] = [\![A]\!]$.

Proof. Straightforward induction on A. □

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

Proof of Lemma 41. First, observe that for every inference rule $\mathsf{r} \in \{\mathsf{w},\mathsf{w}_\forall,\mathsf{ac},\widehat{\mathsf{c}_\forall},\mathsf{m},\widehat{\mathsf{m}_\forall},\widehat{\mathsf{m}_\exists},\equiv\}$ the induced map $\lfloor\mathsf{r}\rfloor\colon V_{[\![Q]\!]} \to V_{[\![P]\!]}$ defines an existential-preserving graph homomorphism $[\![Q]\!] \to [\![P]\!]$ and a fibration on the corresponding binding graphs. Therefore, their composition $\lfloor\Phi\rfloor$ has the same properties of 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. The instances of the rules $\widehat{\mathsf{m}_\forall}$ and $\widehat{\mathsf{m}_\exists}$ are replaced by:

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

respectively, where $\widehat{\mathsf{ac}}$ is a 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, ac, m, or $\equiv$ is translated to an instance of the same rule, $\widehat{\mathsf{c}_\forall}$ is translated to $\widehat{\mathsf{ac}}$, and $\mathsf{w}_\forall$ is translated to w.

This gives us a derivation $\begin{array}{c} A^\circ \\ {\scriptstyle \{\mathsf{w},\mathsf{ac},\widehat{\mathsf{ac}},\mathsf{m},\equiv^\circ\}} \Big\Vert {\scriptstyle \Phi^\circ} \\ B^\circ \end{array}$ such that $\lfloor\Phi^\circ\rfloor = \lfloor\Phi\rfloor$. It has been shown in [23] that $\lfloor\Phi^\circ\rfloor$ is a skew fibration. Hence, $\lfloor\Phi\rfloor$ is a skew fibration. □

B. From Skew Bifibrations to Contraction and Weakening

Lemma 43. *Let $\mathcal{A}$ and $\mathcal{B}$ be fographs, let $\varphi\colon \mathcal{A} \to \mathcal{B}$ be a skew bifibration, and let A and B be formulas with $[\![A]\!] = \mathcal{A}$ and $[\![B]\!] = \mathcal{B}$. Then there are derivations*

$$\begin{array}{c} A \\ {\scriptstyle \{\mathsf{w},\mathsf{w}_\forall,\mathsf{ac},\widehat{\mathsf{c}_\forall},\mathsf{m},\widehat{\mathsf{m}_\forall},\widehat{\mathsf{m}_\exists},\equiv\}} \Big\Vert {\scriptstyle \widehat{\Phi}} \\ B \end{array} \quad \textit{and} \quad \begin{array}{c} A\rho_\varphi \\ {\scriptstyle \{\mathsf{w},\mathsf{w}_\forall,\mathsf{ac},\mathsf{c}_\forall,\mathsf{m},\mathsf{m}_\forall,\mathsf{m}_\exists,\equiv\}} \Big\Vert {\scriptstyle \Phi} \\ B \end{array}$$

such that $\lfloor\widehat{\Phi}\rfloor = \varphi$ and $\widehat{\Phi}$ is a rectification of Φ, and ρ_φ is the substitution induced by φ.

In the proof of this lemma, we make use of the following concept: Let $\begin{array}{c} P \\ {\scriptstyle \mathsf{S}} \Big\Vert {\scriptstyle \Psi} \\ Q \end{array}$ be a derivation where P and Q are propositional formulas (possibly using variable $x \in \mathsf{VAR}$ 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 $\begin{array}{c} C \\ {\scriptstyle \mathsf{S}'} \Big\Vert {\scriptstyle \Psi'} \\ D \end{array}$.

We say a fograph homomorphism $\varphi\colon \mathcal{G} \to \mathcal{H}$ is ***full*** if for all $v, w \in V_\mathcal{G}$, we have that $\varphi(v)\varphi(w) \in E_\mathcal{H}$ implies $vw \in E_\mathcal{G}$.

Lemma 44. *Let $\varphi\colon \mathcal{G} \to \mathcal{H}$ be full and injective skew bifibration such that ρ_φ is the identity substitution, and let G and H be formulas with $[\![G]\!] = \mathcal{G}$ and $[\![H]\!] = \mathcal{H}$. Then there is a derivation* $\begin{array}{c} G \\ {\scriptstyle \{\mathsf{w},\mathsf{w}_\forall,\equiv\}} \Big\Vert {\scriptstyle \Phi} \\ H \end{array}$.

Proof. By [23, Proposition 7.6.1], we have a derivation $\begin{array}{c} G^\circ \\ {\scriptstyle \{\mathsf{w},\equiv^\circ\}} \Big\Vert {\scriptstyle \Psi} \\ H^\circ \end{array}$. In order to lift Ψ, we need to reorganize the instances of w. If H contains a subformula $\forall x.A$ which is not present in G, the w-instances in Ψ could introduce the parts of the propositional encoding $x \vee A$ independently. We say that an instance r_1 of w in Φ is *in the scope* of an instance r_2 of w if r_1 introduced formulas that contain a free variable x (i.e., x occurs in a term in a predicate) and r_2 introduces the atom x as a subformula (i.e. the propositional encoding of the binder x). We can now permute the w-instances in Ψ such that whenever a rule instance r_1 is in the scope on an instance r_2, then r_2 occurs below r_1 in Ψ. Then we can lift Ψ stepwise. First, observe that each line of Ψ is $\equiv^\circ$-equivalent to the propositional encoding P° of a first-order formula P. We now have to show that each instance of w in Ψ is indeed the image of a correct application of w or $\mathsf{w}_\forall$ in first-order logic. If we have a w of the form

$$\mathsf{w}\,\frac{S^\circ\{A^\circ\}}{S^\circ\{x \vee A^\circ\}} \qquad \text{or} \qquad \mathsf{w}\,\frac{S^\circ\{A^\circ\}}{S^\circ\{(x \vee B^\circ) \vee A^\circ\}}$$

then x cannot occur freely in A, as otherwise the fibration property would be violated. We can therefore lift these instances to

$$\mathsf{w}_\forall\,\frac{S\{A\}}{S\{\forall x.A\}} \qquad \text{or} \qquad \mathsf{w}\,\frac{S\{A\}}{S\{(\forall x.B) \vee A\}}$$

respectively. If a weakening happens inside a subformula $x \vee C^\circ$ or $x \wedge C^\circ$ in Ψ, then there are the following cases:

$$\mathsf{w}\,\frac{S^\circ\{x \vee C^\circ\}}{S^\circ\{x \vee D^\circ \vee C^\circ\}} \quad \mathsf{w}\,\frac{S^\circ\{x \wedge C^\circ\}}{S^\circ\{x \wedge (D^\circ \vee C^\circ)\}} \quad \mathsf{w}\,\frac{S^\circ\{x \wedge C^\circ\}}{S^\circ\{(x \vee D^\circ) \wedge C^\circ\}}$$

The first two cases can be lifted to

$$\mathsf{w}\,\frac{S\{\forall x.C\}}{S\{\forall x.(D \vee C)\}} \quad \text{and} \quad \mathsf{w}\,\frac{S\{\exists x.C\}}{S\{\exists x.(D \vee C)\}}$$

respectively. But 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. All other situations can be lifted trivially, giving us $\begin{array}{c} G \\ {\scriptstyle \{\mathsf{w},\mathsf{w}_\forall,\equiv\}}\big\Vert {\scriptstyle \Phi} \\ H \end{array}$ as desired. □

Lemma 45. *Let $\varphi\colon \mathcal{G} \to \mathcal{H}$ be a surjective skew bifibration, and let G and H be formulas with $[\![G]\!] = \mathcal{G}$ and $[\![H]\!] = \mathcal{H}$. Then there is a derivation*

$$\begin{array}{c} G\rho_\varphi \\ {\scriptstyle \{\mathsf{ac},\mathsf{c}_\forall,\mathsf{m},\mathsf{m}_\forall,\mathsf{m}_\exists,\equiv\}}\big\Vert {\scriptstyle \Phi} \\ H \end{array}$$

where ρ_φ is the substitution induced by φ.

Proof. By [44, Proposition 7.5], there is a derivation $\begin{array}{c} (G\sigma_\varphi)^\circ \\ {\scriptstyle \{\mathsf{ac},\mathsf{m},\equiv^\circ\}}\big\Vert {\scriptstyle \Psi} \\ H^\circ \end{array}$. We can lift Ψ to a first-order derivation in $\{\mathsf{ac}, \mathsf{c}_\forall, \mathsf{m}, \mathsf{m}_\forall, \mathsf{m}_\exists, \equiv\}$, in a similar way as in the previous lemma. The technical details are in Appendix C. □

Proof of Lemma 43. Let $V'_\mathcal{B} \subseteq V_\mathcal{B}$ be the image of φ, and let $\mathcal{B}_1$ be the subgraph of $\mathcal{B}$ induced by $V'_\mathcal{B}$. Hence, we have two maps $\varphi''\colon \mathcal{A} \to \mathcal{B}_1$ being a surjection and $\varphi'\colon \mathcal{B}_1 \to \mathcal{B}$ being a full injection. Both, φ' and φ'' remain skew bifibrations. Furthermore, $\mathcal{B}_1$ is also a fograph. Let B_1 be a formula with $[\![B_1]\!] = \mathcal{B}_1$. We can apply Lemmas 44 and 45 to obtain derivations

$$\begin{array}{c} B_1 \\ {\scriptstyle \{\mathsf{w},\mathsf{w}_\forall,\equiv\}}\big\Vert {\scriptstyle \Phi'} \\ B \end{array} \quad \text{and} \quad \begin{array}{c} A\rho_{\varphi''} \\ {\scriptstyle \{\mathsf{ac},\mathsf{c}_\forall,\mathsf{m},\mathsf{m}_\forall,\mathsf{m}_\exists,\equiv\}}\big\Vert {\scriptstyle \Phi''} \\ B_1 \end{array}$$

As $\rho_{\varphi'}$ is the identity, we have $\rho_{\varphi''} = \rho_\varphi$. Hence, the composition of Φ'' and Φ' is the desired derivation Φ. Then $\widehat{\Phi}$ can be constructed by rectifying Φ, where the variables to be used in A are already given. That $\varphi = \lfloor\widehat{\Phi}\rfloor$ follows immediately from the construction. □

X. Summary and Proof of Main Result

The only theorem of Section VI that has not yet been proved is Theorem 19 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 19. First, assume we have a combinatorial proof $\varphi\colon \mathcal{C} \to \mathcal{A}$ and a formula A with $\mathcal{A} = [\![A]\!]$. Let C be a formula with $[\![C]\!] = \mathcal{C}$, and let ρ_φ be the substitution induced by φ. By Lemma 43 there is a derivation

$$\begin{array}{c} C\rho_\varphi \\ {\scriptstyle \{\mathsf{w},\mathsf{w}_\forall,\mathsf{ac},\mathsf{c}_\forall,\mathsf{m},\mathsf{m}_\forall,\mathsf{m}_\exists,\equiv\}}\big\Vert {\scriptstyle \Phi_2} \\ A \end{array}$$

Since $\mathcal{C}$ is a fonet, we have by Theorem 38 a derivation

$$\begin{array}{c} \mathsf{t} \\ {\scriptstyle \mathsf{MLS1}^\mathsf{X}}\big\Vert {\scriptstyle \Phi'_1} \\ C \end{array}$$

This derivation remains valid if we apply the substitution ρ_φ to every line in Φ'_1, yielding the derivation Φ_1 of $C\rho_\varphi$ as desired.

Conversely, assume we have a decomposed derivation

$$\begin{array}{c} \mathsf{t} \\ {\scriptstyle \mathsf{MLS1}^\mathsf{X}}\big\Vert {\scriptstyle \Phi_1} \\ A' \\ {\scriptstyle \{\mathsf{w},\mathsf{w}_\forall,\mathsf{ac},\mathsf{c}_\forall,\mathsf{m},\mathsf{m}_\forall,\mathsf{m}_\exists,\equiv\}}\big\Vert {\scriptstyle \Phi_2} \\ A \end{array} \tag{6}$$

Then we can transform Φ_1 into a rectified form $\widehat{\Phi}_1$, proving $\widehat{A}'$. By Theorem 33, the linked fograph $[\![\widehat{\Phi}_1]\!] = \langle [\![\widehat{A}']\!], \sim_{\widehat{\Phi}_1} \rangle$ is a fonet. Then, by Lemma 40, there is a rectified derivation $\begin{array}{c} \widehat{A'} \\ {\scriptstyle \{\mathsf{w},\mathsf{w}_\forall,\mathsf{ac},\widehat{\mathsf{c}_\forall},\mathsf{m},\widehat{\mathsf{m}_\forall},\widehat{\mathsf{m}_\exists},\equiv\}}\big\Vert {\scriptstyle \widehat{\Phi_2}} \\ \widehat{A} \end{array}$ whose induced map $\lfloor\widehat{\Phi_2}\rfloor\colon [\![\widehat{A'}]\!] \to [\![\widehat{A}]\!]$ is the same as the induced map $\lfloor\Phi_2\rfloor\colon [\![A']\!] \to [\![A]\!]$ of Φ_2. By Lemma 41, this map is a skew bifibration. Hence, we have a combinatorial proof $\varphi\colon \mathcal{C} \to [\![A]\!]$ with $\mathcal{C} = [\![\widehat{A'}]\!]$. □

Note that Theorem 19 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 normalisation of combinatorial proofs is underdeveloped. There exist two different procedures for cut elimination for combinatorial proofs in classical propositional logic [10], [12], but both have their insufficiencies, and they have not been extended to other logics.

Nonetheless, we hope to get new insights in the normalisation of classical first-order proofs through our work on combinatorial proofs.

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] R. Thiele, “Hilbert’s Twenty-fourth Problem,” *American Mathematical Monthly*, vol. 110, pp. 1–24, 2003.
[8] D. Hilbert, “Mathematische Probleme,” *Nachrichten der Königlichen Gesellschaft der Wissenschaften zu Göttingen, mathematisch-physikalische Klasse*, vol. 3, pp. 253–297, 1900.
[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,” *Theoretical Computer Science*, 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 comtemporanee*, 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] L. Straßburger, “Linear logic and Noncommutativity in the Calculus of Structures,” Ph.D. dissertation, Technische Universität Dresden, 2003.
[36] 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.
[37] A. A. Tubella and L. Straßburger, “Introduction to Deep Inference,” 2019, lecture notes for ESSLLI’19. [Online]. Available: https://hal.inria.fr/hal-02390267
[38] 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.
[39] C. Retoré, “Pomset Logic as a Calculus of Directed Cographs,” INRIA, Research Report RR-3714, 1999. [Online]. Available: https://hal.inria.fr/inria-00072953
[40] V. Danos and L. Regnier, “The Structure of Multiplicatives,” *Archive for Mathematical Logic*, vol. 28, no. 3, pp. 181–203, 1989.
[41] C. Retoré, “Handsome Proof-nets: Perfect Matchings and Cographs,” *Theoretical Computer Science*, vol. 294, no. 3, pp. 473–488, 2003.
[42] 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
[43] L. Straßburger, “Deep Inference and Expansion Trees for Second-order Multiplicative Linear Logic,” *Mathematical Structures in Computer Science*, vol. 29, pp. 1030–1060, 2019.
[44] 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.
[45] F. Lamarche, “Exploring the Gap Between Linear and Classical Logic,” *Theory and Applications of Categories*, vol. 18, no. 18, pp. 473–535, 2007.

APPENDIX

A. Proof of Theorem 29

Proof of Theorem 29. Write $\mathsf{fv}(A)$ for the set of variables which occur free in A.

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

First, if an instance of $\mathsf{wk}\,\dfrac{\vdash \Gamma}{\vdash \Gamma, A}$ 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.

$$\wedge\,\dfrac{\mathsf{wk}\,\dfrac{\vdash \Gamma}{\vdash \Gamma, A} \quad \vdash B, \Delta}{\vdash \Gamma, A \wedge B, \Delta} \rightsquigarrow \mathsf{wk}\,\dfrac{\mathsf{wk}\,\dfrac{\vdash \Gamma}{\vdots}}{\vdash \Gamma, A \wedge B, \Delta}$$

$$\vee\,\dfrac{\mathsf{wk}\,\dfrac{\vdash \Gamma, A}{\vdash \Gamma, A, B}}{\vdash \Gamma, A \vee B} \rightsquigarrow \mathsf{w}\,\dfrac{\vdash \Gamma, A}{\vdash \Gamma, A \vee B}$$

$$\exists\,\dfrac{\mathsf{wk}\,\dfrac{\vdash \Gamma}{\vdash \Gamma, A[x/t]}}{\vdash \Gamma, \exists x.A} \rightsquigarrow \mathsf{wk}\,\dfrac{\vdash \Gamma}{\vdash \Gamma, \exists x.A}$$

$$\forall\,\dfrac{\mathsf{wk}\,\dfrac{\vdash \Gamma}{\vdash \Gamma, A}}{\vdash \Gamma, \forall x.A} \rightsquigarrow \mathsf{wk}\,\dfrac{\vdash \Gamma}{\vdash \Gamma, \forall x.A}$$

$$\mathsf{ctr}\,\dfrac{\mathsf{wk}\,\dfrac{\vdash \Gamma, A}{\vdash \Gamma, A, A}}{\vdash \Gamma, A} \rightsquigarrow \vdash \Gamma, A$$

Note that in the case of $\vee$, we use the deep rule w which can be permuted under 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:

$$\mathsf{ctr}\,\dfrac{\vdash \Gamma, A, A}{\vdash \Gamma, A} \rightsquigarrow \mathsf{c}\,\dfrac{\vee\,\dfrac{\vdash \Gamma, A, A}{\vdash \Gamma, A \vee A}}{\vdash \Gamma, A}$$

Knowing that c can be permuted under all the rules of $\mathsf{MLL1^X}$, we eventually obtain a derivation:

$$\begin{array}{c} \mathsf{MLL1^X} \;\; \Phi_1' \\ \vdash \Gamma' \\ \{\mathsf{wk},\mathsf{w},\mathsf{c},\equiv\} \, \Big\| \, \Phi_2' \\ \vdash \Gamma \end{array}$$

Note that $\equiv$ is required here since the permutation of formulas is implicit in $\mathsf{MLL1^X}$.

By transforming each sequent of Φ_2' into its corresponding formula, and by considering the following rewriting rule:

$$\mathsf{wk}\,\dfrac{\vdash \Gamma}{\vdash \Gamma, A} \rightsquigarrow \mathsf{w}\,\dfrac{\vdash \bigvee(\Gamma)}{\vdash \bigvee(\Gamma) \vee A}$$

, we obtain a derivation

$$\begin{array}{c} \mathsf{MLL1^X} \;\; \Phi_1 \\ \vdash \bigvee(\Gamma') \\ \{\mathsf{w},\mathsf{c},\equiv\} \, \Big\| \, \Phi_2 \\ \vdash \bigvee(\Gamma) \end{array}$$

where Φ_1 can be obtained from Φ_1' by applying the $\vee$ rule. $\square$

B. Rule permutation for the proof of Lemma 31

We construct a rewriting system based on rule permutation on derivations in $\{\mathsf{w}, \mathsf{w}_\forall, \mathsf{ac}, \mathsf{c}_\forall, \mathsf{m}, \mathsf{m}_\forall, \mathsf{m}_\exists, \equiv\}$ that allows us to reach a derivation of the form

$$\begin{array}{c} A \\ \{\mathsf{m},\mathsf{m}_\forall,\mathsf{m}_\exists,\equiv\} \, \Big\| \\ A' \\ \{\mathsf{ac},\mathsf{c}_\forall\} \, \Big\| \\ B' \\ \{\mathsf{w},\mathsf{w}_\forall,\equiv\} \, \Big\| \\ B \end{array}$$

from any derivation. Intuitively, we want to move all the instances of $\mathsf{r} \in \{\mathsf{w}, \mathsf{w}_\forall\}$ downwards and all the instances of $\mathsf{r}' \in \{\mathsf{m}, \mathsf{m}_\forall, \mathsf{m}_\exists\}$ upwards.

We first study the interactions between two rules. Certain cases are unsolved at this stage, and they are considered later when we study the interactions between two non-$\equiv$ rule instances separated by $\equiv$. Only non-trivial cases are presented here:

- $\mathsf{r}_1/\mathsf{r}_2$, where $\mathsf{r}_1 \in \{\mathsf{w}, \mathsf{w}_\forall\}$ and $\mathsf{r}_2 \in \{\mathsf{ac}, \mathsf{c}_\forall, \mathsf{m}, \mathsf{m}_\forall, \mathsf{m}_\exists\}$:

$$\mathsf{ac}\,\dfrac{\mathsf{w}\,\dfrac{a}{a \vee a}}{a} \rightsquigarrow a$$

$$\mathsf{m}\,\dfrac{\mathsf{w}\,\dfrac{A \wedge C}{(A \wedge C) \vee (B \wedge D)}}{(A \vee B) \wedge (C \vee D)} \rightsquigarrow \mathsf{w}\,\dfrac{\mathsf{w}\,\dfrac{A \wedge C}{(A \vee B) \wedge C}}{(A \vee B) \wedge (C \vee D)}$$

$$\mathsf{m}_\forall\,\dfrac{\mathsf{w}\,\dfrac{\forall x.A}{(\forall x.A) \vee (\forall x.B)}}{\forall x.(A \vee B)} \rightsquigarrow \mathsf{w}\,\dfrac{\forall x.A}{\forall x.(A \vee B)}$$

$$\mathsf{c}_\forall\,\dfrac{\mathsf{w}_\forall\,\dfrac{\forall x.A}{\forall x.\forall x.A}}{\forall x.A} \rightsquigarrow \forall x.A$$

$$\mathsf{m}_\forall\,\dfrac{\mathsf{w}_\forall\,\dfrac{A \vee (\forall x.B)}{(\forall x.A) \vee (\forall x.B)}}{\forall x.(A \vee B)} \rightsquigarrow \equiv\,\dfrac{A \vee (\forall x.B)}{\forall x.(A \vee B)}$$

where in the last case, x is not free in A.

- $\mathsf{r}_1/\mathsf{r}_2$, where $\mathsf{r}_1 \in \{\mathsf{ac}, \mathsf{c}_\forall\}$ and $\mathsf{r}_2 \in \{\mathsf{m}, \mathsf{m}_\forall, \mathsf{m}_\exists\}$:

$$\mathsf{c}_\forall\,\cfrac{S\{(\forall x.\forall x.A) \vee (\forall x.B)\}}{\mathsf{m}_\forall\,\cfrac{S\{(\forall x.A) \vee (\forall x.B)\}}{S\{\forall x.(A \vee B)\}}} \rightsquigarrow \mathsf{m}_\forall\,\cfrac{S\{(\forall x.\forall x.A) \vee (\forall x.B)\}}{\equiv\,\cfrac{S\{\forall x.(\forall x.A \vee B)\}}{\mathsf{m}_\forall\,\cfrac{S\{(\forall x.A) \vee (\forall x.B)\}}{S\{\forall x.(A \vee B)\}}}}$$

- $\mathsf{c}_\forall/\equiv$:

$$\mathsf{c}_\forall\,\cfrac{\forall x.\forall x.\forall y.A}{\equiv\,\cfrac{\forall x.\forall y.A}{\forall y.\forall x.A}} \rightsquigarrow \equiv\,\cfrac{\forall x.\forall x.\forall y.A}{\mathsf{c}_\forall\,\cfrac{\forall y.\forall x.\forall x.A}{\forall y.\forall x.A}}$$

$$\mathsf{c}_\forall\,\cfrac{\forall x.\forall x.(A \vee B)}{\equiv\,\cfrac{\forall x.(A \vee B)}{(\forall x.A) \vee B}} \rightsquigarrow \equiv\,\cfrac{\forall x.\forall x.(A \vee B)}{\mathsf{c}_\forall\,\cfrac{(\forall x.\forall x.A) \vee B}{(\forall x.A) \vee B}}$$

$$\mathsf{c}_\forall\,\cfrac{(\forall x.\forall x.A) \vee B}{\equiv\,\cfrac{(\forall x.A) \vee B}{\forall x.(A \vee B)}} \rightsquigarrow \equiv\,\cfrac{(\forall x.\forall x.A) \vee B}{\mathsf{c}_\forall\,\cfrac{\forall x.\forall x.(A \vee B)}{\forall x.(A \vee B)}}$$

where in the last two cases, x is not free in B.

- $\mathsf{w}/\equiv$:

$$\mathsf{w}\,\cfrac{A}{\equiv\,\cfrac{A \vee B}{B \vee A}}$$

$$\mathsf{w}\,\cfrac{A \vee C}{\equiv\,\cfrac{(A \vee B) \vee C)}{A \vee (B \vee C)}}$$

$$\mathsf{w}\,\cfrac{\forall x.A}{\equiv\,\cfrac{\forall x.(A \vee B)}{(\forall x.A) \vee B}} \rightsquigarrow \mathsf{w}\,\cfrac{\forall x.A}{(\forall x.A) \vee B}$$

$$\mathsf{w}\,\cfrac{\forall x.B}{\equiv\,\cfrac{\forall x.(B \vee A)}{(\forall x.A) \vee B}}$$

$$\mathsf{w}\,\cfrac{\forall x.A}{\equiv\,\cfrac{(\forall x.A) \vee B}{\forall x.(A \vee B)}} \rightsquigarrow \mathsf{w}\,\cfrac{\forall x.A}{\forall x.(A \vee B)}$$

$$\mathsf{w}\,\cfrac{B}{\equiv\,\cfrac{B \vee (\forall x.A)}{\forall x.(A \vee B)}}$$

where in the last four cases, x is not free in B.

- $\mathsf{w}_\forall/\equiv$:
In the following two cases, we assume $x \neq y$ (otherwise they are trivial).

$$\mathsf{w}_\forall\,\cfrac{\forall y.A}{\equiv\,\cfrac{\forall x.\forall y.A}{\forall y.\forall x.A}}\,(x \notin \mathsf{fv}(\forall y.A)) \rightsquigarrow \mathsf{w}_\forall\,\cfrac{\forall y.A}{\forall y.\forall x.A}\,(x \notin \mathsf{fv}(A))$$

$$\mathsf{w}_\forall\,\cfrac{\forall y.A}{\equiv\,\cfrac{\forall y.\forall x.A}{\forall x.\forall y.A}}\,(x \notin \mathsf{fv}(A)) \rightsquigarrow \mathsf{w}_\forall\,\cfrac{\forall y.A}{\forall x.\forall y.A}\,(x \notin \mathsf{fv}(\forall y.A))$$

$$\mathsf{w}_\forall\,\cfrac{A \vee B}{\equiv\,\cfrac{\forall x.(A \vee B)}{(\forall x.A) \vee B}} \rightsquigarrow \mathsf{w}_\forall\,\cfrac{A \vee B}{(\forall x.A) \vee B}$$

$$\mathsf{w}_\forall\,\cfrac{A \vee B}{\equiv\,\cfrac{(\forall x.A) \vee B}{\forall x.(A \vee B)}} \rightsquigarrow \mathsf{w}_\forall\,\cfrac{A \vee B}{\forall x.(A \vee B)}$$

where in the last two cases, the constraint on x on the left-hand side implies that of the right-hand side.

- $\equiv/\mathsf{c}_\forall$:

$$\equiv\,\cfrac{\forall x.\forall y.\forall x.A}{\mathsf{c}_\forall\,\cfrac{\forall x.\forall x.\forall y.A}{\forall x.\forall y.A}}$$

$$\equiv\,\cfrac{\forall x.\forall y.\forall x.A}{\mathsf{c}_\forall\,\cfrac{\forall y.\forall x.\forall x.A}{\forall y.\forall x.A}}$$

$$\equiv\,\cfrac{\forall x.((\forall x.A) \vee B)}{\mathsf{c}_\forall\,\cfrac{(\forall x.\forall x.A) \vee B}{(\forall x.A) \vee B}}\,(x \notin \mathsf{fv}(B))$$

$$\equiv\,\cfrac{\forall x.((\forall x.A) \vee B)}{\mathsf{c}_\forall\,\cfrac{\forall x.\forall x.(A \vee B)}{\forall x.(A \vee B)}}\,(x \notin \mathsf{fv}(B))$$

- $\equiv/\mathsf{m}$:

$$\equiv\,\cfrac{(C \wedge A) \vee (B \wedge D)}{\mathsf{m}\,\cfrac{(A \wedge C) \vee (B \wedge D)}{(A \vee B) \wedge (C \vee D)}}$$

$$\equiv\,\cfrac{(B \wedge D) \vee (A \wedge C)}{\mathsf{m}\,\cfrac{(A \wedge C) \vee (B \wedge D)}{(A \vee B) \wedge (C \vee D)}} \rightsquigarrow \mathsf{m}\,\cfrac{(B \wedge D) \vee (A \wedge C)}{\equiv\,\cfrac{(B \vee A) \wedge (D \vee C)}{(A \vee B) \wedge (C \vee D)}}$$

$$\equiv\,\cfrac{((A \wedge C) \wedge E) \vee (B \wedge D)}{\mathsf{m}\,\cfrac{(A \wedge (C \wedge E)) \vee (B \wedge D)}{(A \vee B) \wedge ((C \wedge E) \vee D)}}$$

$$\equiv\,\cfrac{(\forall x.(A \wedge C)) \vee (B \wedge D)}{\mathsf{m}\,\cfrac{\forall x.((A \wedge C) \vee (B \wedge D))}{\forall x.((A \vee B) \wedge (C \vee D))}}\,(x \notin \mathsf{fv}(B \wedge D))$$

- $\equiv/\mathsf{m}_\forall$:

$$\equiv\,\cfrac{(\forall x.B) \vee (\forall x.A)}{\mathsf{m}_\forall\,\cfrac{(\forall x.A) \vee (\forall x.B)}{\forall x.(A \vee B)}} \rightsquigarrow \mathsf{m}_\forall\,\cfrac{(\forall x.B) \vee (\forall x.A)}{\equiv\,\cfrac{\forall x.(B \vee A)}{\forall x.(A \vee B)}}$$

$$\mathsf{m}_\forall\,\dfrac{\equiv\,\dfrac{(\forall y.\forall x.A)\lor(\forall x.B)}{(\forall x.\forall y.A)\lor(\forall x.B)}}{\forall x.((\forall y.A)\lor B)}$$

$$\mathsf{m}_\forall\,\dfrac{\equiv\,\dfrac{\forall x.(A\lor(\forall x.B))}{(\forall x.A)\lor(\forall x.B)}}{\forall x.(A\lor B)}$$

- $\equiv/\mathsf{m}_\exists$: similar to $\equiv/\mathsf{m}_\forall$

Interactions between two non-$\equiv$ rules with the presence of $\equiv$ in between:

- $\mathsf{c}_\forall/\equiv/\mathsf{r}$ where $\mathsf{r}\in\{\mathsf{m},\mathsf{m}_\forall,\mathsf{m}_\exists\}$: First permute $\mathsf{c}_\forall$ under $\equiv$ and then permute $\mathsf{c}_\forall$ under r.
- $\mathsf{ac}/\equiv/\mathsf{r}$ where $\mathsf{r}\in\{\mathsf{m},\mathsf{m}_\forall,\mathsf{m}_\exists\}$: First permute ac under $\equiv$ and then permute ac under r.
- $\mathsf{w}/\equiv/\mathsf{c}_\forall$:

$$\mathsf{c}_\forall\,\dfrac{\equiv\,\dfrac{\mathsf{w}\,\dfrac{\forall x.\forall x.A}{\forall x.((\forall x.A)\lor B)}}{(\forall x.\forall x.A)\lor B}}{(\forall x.A)\lor B}\rightsquigarrow\mathsf{w}\,\dfrac{\mathsf{c}_\forall\,\dfrac{\forall x.\forall x.A}{(\forall x.A)\lor B}}{(\forall x.A)\lor B}$$

$$\mathsf{c}_\forall\,\dfrac{\equiv\,\dfrac{\mathsf{w}\,\dfrac{\forall x.B}{\forall x.(B\lor(\forall x.A))}}{(\forall x.\forall x.A)\lor B}}{(\forall x.A)\lor B}\rightsquigarrow\equiv\,\dfrac{\mathsf{w}\,\dfrac{\forall x.B}{\forall x.(B\lor A)}}{(\forall x.A)\lor B}$$

$$\mathsf{c}_\forall\,\dfrac{\equiv\,\dfrac{\mathsf{w}\,\dfrac{\forall x.\forall x.A}{\forall x.((\forall x.A)\lor B)}}{\forall x.\forall x.(A\lor B)}}{\forall x.(A\lor B)}\rightsquigarrow\mathsf{w}\,\dfrac{\mathsf{c}_\forall\,\dfrac{\forall x.\forall x.A}{\forall x.A}}{\forall x.(A\lor B)}$$

$$\mathsf{c}_\forall\,\dfrac{\equiv\,\dfrac{\mathsf{w}\,\dfrac{\forall x.B}{\forall x.(B\lor(\forall x.A))}}{\forall x.\forall x.(A\lor B)}}{\forall x.(A\lor B)}\rightsquigarrow\equiv\,\dfrac{\mathsf{w}\,\dfrac{\forall x.B}{\forall x.(B\lor A)}}{\forall x.(A\lor B)}$$

where in all four cases, x is not free in B.

- $\mathsf{w}/\equiv/\mathsf{ac}$:

$$\mathsf{ac}\,\dfrac{\equiv\,\dfrac{\mathsf{w}\,\dfrac{a\lor B}{(a\lor B)\lor a}}{(a\lor a)\lor B}}{a\lor B}\rightsquigarrow a\lor B$$

$$\mathsf{ac}\,\dfrac{\equiv\,\dfrac{\mathsf{w}\,\dfrac{a}{a\lor(a\lor B)}}{(a\lor a)\lor B}}{a\lor B}\rightsquigarrow\mathsf{w}\,\dfrac{a}{a\lor B}$$

$$\mathsf{ac}\,\dfrac{\equiv\,\dfrac{\mathsf{w}\,\dfrac{\forall x.a}{(\forall x.a)\lor a}}{\forall x.(a\lor a)}}{\forall x.a}\;(x\notin\mathsf{fv}(a))\rightsquigarrow\forall x.a$$

$$\mathsf{ac}\,\dfrac{\equiv\,\dfrac{\mathsf{w}\,\dfrac{a}{a\lor(\forall x.a)}}{\forall x.(a\lor a)}}{\forall x.a}\;(x\notin\mathsf{fv}(a))\rightsquigarrow\mathsf{w}_\forall\,\dfrac{a}{\forall x.a}\;(x\notin\mathsf{fv}(a))$$

- $\mathsf{w}/\equiv/\mathsf{m}$:

$$\mathsf{m}\,\dfrac{\equiv\,\dfrac{\mathsf{w}\,\dfrac{C\land A}{(C\land A)\lor(B\land D)}}{(A\land C)\lor(B\land D)}}{(A\lor B)\land(C\lor D)}\rightsquigarrow\mathsf{w}\,\dfrac{\mathsf{w}\,\dfrac{\equiv\,\dfrac{C\land A}{A\land C}}{(A\lor B)\land C}}{(A\lor B)\land(C\lor D)}$$

$$\mathsf{m}\,\dfrac{\equiv\,\dfrac{\mathsf{w}\,\dfrac{B\land D}{(B\land D)\lor(\forall x.(A\land C))}}{\forall x.((A\land C)\lor(B\land D))}}{\forall x.((A\lor B)\land(C\lor D))}\rightsquigarrow\equiv\,\dfrac{\mathsf{w}\,\dfrac{\mathsf{w}\,\dfrac{\forall\,\dfrac{B\land D}{\forall x.(B\land D)}}{\forall x.((B\lor A)\land D)}}{\forall x.((B\lor A)\land(D\lor C))}}{\forall x.((A\lor B)\land(C\lor D))}$$

where in the second case, x is free in $B\land D$.

- $\mathsf{w}/\equiv/\mathsf{m}_\forall$:

$$\mathsf{m}_\forall\,\dfrac{\equiv\,\dfrac{\mathsf{w}\,\dfrac{\forall x.B}{(\forall x.B)\lor(\forall x.A)}}{(\forall x.A)\lor(\forall x.B)}}{\forall x.(A\lor B)}\rightsquigarrow\equiv\,\dfrac{\mathsf{w}\,\dfrac{\forall x.B}{\forall x.(B\lor A)}}{\forall x.(A\lor B)}$$

$$\mathsf{m}_\forall\,\dfrac{\equiv\,\dfrac{\mathsf{w}\,\dfrac{\forall x.\forall x.A}{\forall x.((\forall x.A)\lor B)}}{(\forall x.A)\lor(\forall x.B)}}{\forall x.(A\lor B)}\rightsquigarrow\mathsf{w}\,\dfrac{\mathsf{c}_\forall\,\dfrac{\forall x.\forall x.A}{\forall x.A}}{\forall x.(A\lor B)}$$

- $\mathsf{w}/\equiv/\mathsf{m}_\exists$:

$$\mathsf{m}_\exists\,\dfrac{\equiv\,\dfrac{\mathsf{w}\,\dfrac{\exists x.B}{(\exists x.B)\lor(\exists x.A)}}{(\exists x.A)\lor(\exists x.B)}}{\exists x.(A\lor B)}\rightsquigarrow\equiv\,\dfrac{\mathsf{w}\,\dfrac{\exists x.B}{\exists x.(B\lor A)}}{\exists x.(A\lor B)}$$

C. Proof of Lemma 45

Proof of Lemma 45. By [44, Proposition 7.5], there is a derivation $\begin{array}{c}(G\rho_\varphi)^\circ\\ {\scriptstyle\{\mathsf{ac},\mathsf{m},\equiv^\circ\}}\big\Vert{\scriptstyle\Psi}\\ H^\circ\end{array}$, We plan to show that Ψ can be lifted to $\{\mathsf{ac},\mathsf{c}_\forall,\mathsf{m},\mathsf{m}_\forall,\mathsf{m}_\exists,\equiv\}$. However, observe that not every formula occurring in Ψ 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 ac can duplicate an atom $x\in\textsc{var}$. Let us write ac_x for such instances. The problem with (i) is that we could have the following situation

$$\mathsf{m}\,\dfrac{\equiv^\circ\,\dfrac{S\{(x\land(E\land C))\lor(x\land(F\land D))\}}{S\{((x\land E)\land C)\lor((x\land F)\land D)\}}}{S\{((x\land E)\lor(x\land F))\land(C\lor D)\}}\tag{7}$$

where x occurs in $C \vee D$. Then premise and conclusion are both propositional encodings, but the whole derivation cannot be lifted. However, since we demand that the mapping is a fibration (and therefore a homomorphism) on the binding graphs, there must be another instance of m further below in the derivation:

$$\mathsf{m}\,\frac{S'\{(x \wedge E) \vee (x \wedge F)\}}{S'\{(x \vee x) \wedge (E \vee F)\}} \tag{8}$$

We can permute both instances via the following more general scheme (see [23], [45] for a general discussion on permutations of the m-rule):

$$\mathsf{m}\,\dfrac{\mathsf{m}\,\dfrac{S\{(G \wedge E \wedge C) \vee (G \wedge F \wedge D)\}}{S\{((G \wedge E) \vee (H \wedge F)) \wedge (C \vee D)\}}}{S\{(G \vee H) \wedge (E \vee F) \wedge (C \vee D)\}} \leftrightarrow \mathsf{m}\,\dfrac{\mathsf{m}\,\dfrac{S\{(G \wedge E \wedge C) \vee (G \wedge F \wedge D)\}}{S\{(G \vee H) \wedge ((E \wedge C) \vee (F \wedge D))\}}}{S\{(G \vee H) \wedge (E \vee F) \wedge (C \vee D)\}} \tag{9}$$

We omitted some instances of $\equiv^\circ$ and some parentheses. We now call instances of m as in (7) ***illegal***, and we can transform Ψ through m-permutations (9) 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

$$\mathsf{ac}_x^{\equiv}\,\frac{S_0\{S_1\{x\} \vee S_2\{x\}\}}{S\{x\}} \tag{10}$$

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 $\mathsf{ac}_x^{\equiv}$ permutes over $\equiv$, ac, and other instances of $\mathsf{ac}_x^{\equiv}$, and over instances of m if they occur inside S_0 or S_1 or S_2. The only situation in which $\mathsf{ac}_x^{\equiv}$ cannot be permuted up is the following:

$$\mathsf{ac}_x^{\equiv}\,\dfrac{\mathsf{m}\,\dfrac{S\{(R_1\{x\} \wedge C) \vee (R_2\{x\} \wedge D)\}}{S\{(R_1\{x\} \vee R_2\{x\}) \wedge (C \vee D)\}}}{S\{R\{x\} \wedge (C \vee D)\}} \tag{11}$$

We can therefore assume that all instances of ac_x, that contract an atom $x \in \textsc{var}$ are either at the top of Ψ or below a m-instance as in (11). We now lift Ψ to $\{\mathsf{ac}, \mathsf{c}_\forall, \mathsf{m}, \mathsf{m}_\forall, \mathsf{m}_\exists, \equiv\}$, proceed by induction on the height of Ψ, 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 (10) 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

    $$\mathsf{ac}_x^{\equiv}\,\frac{S^\circ\{(x \vee E^\circ) \vee (x \vee F^\circ)\}}{S^\circ\{x \vee (E^\circ \vee F^\circ)\}}$$

    which can be lifted to

    $$\mathsf{m}_\forall\,\frac{S\{(\forall x.E) \vee (\forall x.F)\}}{S\{\forall x.(E \vee F)\}}$$

    where $S^\circ\{\cdot\}, E^\circ, F^\circ$ are the propositional encodings of $S\{\cdot\}, E, F$, respectively.
  - E° is empty and F° is non-empty: We have

    $$\mathsf{ac}_x^{\equiv}\,\frac{S^\circ\{x \vee (x \vee F^\circ)\}}{S^\circ\{x \vee F^\circ)\}}$$

    which can be lifted to

    $$\mathsf{c}_\forall\,\frac{S\{\forall x.\forall x.F\}}{S\{\forall x.F\}}$$
  - 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 $x \in \textsc{var}$, 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 $x \in \textsc{var}$, 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 $x \in \textsc{var}$, then we are in the following special case of (11):

    $$\mathsf{ac}_x^{\equiv}\,\dfrac{\mathsf{m}\,\dfrac{S\{(x \wedge C) \vee (x \wedge D)\}}{S\{(x \vee x) \wedge (C \vee D)\}}}{S\{x \wedge (C \vee D)\}}$$

    which can be lifted immediately to

    $$\mathsf{m}_\exists\,\frac{S\{(\exists x.C) \vee (\exists x.D)\}}{S\{\exists x.(C \vee D)\}}$$
  - We have a situation (11) 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 $\mathsf{ac}_x^{\equiv}$ would not be correct.) That means, we have:

    $$\mathsf{ac}_x^{\equiv}\,\dfrac{\mathsf{m}\,\dfrac{S\{((x \vee E) \wedge C) \vee ((x \vee F) \wedge D)\}}{S\{((x \vee E) \vee (x \vee F)) \wedge (C \vee D)\}}}{S\{(x \vee E \vee F) \wedge (C \vee D)\}}$$

    which can be lifted to

    $$\mathsf{m}_\forall\,\dfrac{\mathsf{m}\,\dfrac{S\{((\forall x.E) \wedge C) \vee ((\forall x.F) \wedge D)\}}{S\{((\forall x.E) \vee (\forall x.F)) \wedge (C \vee D)\}}}{S\{(\forall x.(E \vee F)) \wedge (C \vee D)\}}$$
  - 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 Ψ can be lifted to $\begin{array}{c} G\rho_{\varphi} \\ {\scriptstyle \{\mathsf{ac},\mathsf{c}_{\forall},\mathsf{m},\mathsf{m}_{\forall},\mathsf{m}_{\exists},\equiv\}}\Big\Vert\,\Phi \\ H \end{array}$. $\square$