

Proof Nets for Second Order Intuitionistic Logic

Matteo Acclavio 

FORM, University of Southern Denmark, Denmark

Dylan Bettendroffer 

FoSS team, University of Sussex, Brighton, UK

Abstract

New proof nets for second-order intuitionistic logic are introduced with a simple correctness criterion inspired by Lamarche’s criterion for essential nets for propositional intuitionistic linear logic. Polynomial-time translation algorithms between sequent proofs and proof nets are provided. The proof equivalence induced by these proof nets is analyzed in comparison with rules permutations of the sequent calculus. A notion of canonical proof net is introduced, by which modularity is improved.

2012 ACM Subject Classification Replace ccsdesc macro with valid one

Keywords and phrases Proof Nets, Second Order Intuitionistic Logic, Proof Equivalence

Digital Object Identifier 10.4230/LIPIcs.CVIT.2016.23

1 Introduction

System F (or *second-order lambda calculus*), introduced by Girard in [16, 17] and independently by Reynolds [47], extends the simply-typed λ -calculus with second-order quantification. The proof of strong normalization for System F highlighted the need of careful attention on the role structural rules in the proof of cut-elimination for second-order intuitionistic logic. This led Girard to introduce *linear logic* [18], a substructural logic where resources are controlled via the use of *exponential* modalities allowing for a finer analysis of the role of structural rules in cut-elimination. *Proof nets* [18] were then introduced as a graph-based syntax for proofs in linear logic, capable of identifying proofs modulo inessential rules permutations, providing a formalism for proofs getting rid of the *syntactic bureaucracy* of sequent calculus proofs, analogously to what natural deduction does for intuitionistic logic. More precisely, proof nets identify sequent calculus proofs modulo the permutations of rules that are necessary to define cut-elimination steps as local rewriting steps on proofs, and allow recovering convergence of cut-elimination.

Since then, literature of proof nets has largely remained internal to linear logic [18, 12, 45, 46, 34, 8, 21, 22, 31, 40, 25, 29, 32, 26, 13], with rare works non-commutative logic with similar ‘linear’ flavors [48, 43, 39, 1, 44, 49] or classical logic [23]. However, at the time of writing, no proof net syntax has been proposed for intuitionistic logic without relying on encodings in linear logic, and, in particular, on the use of the exponential modalities to model structural rules [38, 37, 2, 3]. Furthermore, the literature on proof nets for second-order logic (e.g., [19, 20, 40, 50, 52, 41, 42]) mostly focusses on the multiplicative fragment, and the correctness criteria clearly struggle with the non-local dependencies between rules introduced by quantifiers, requiring the use of additional structures (e.g., jumps) to keep track of the dependencies between quantifiers rules in the proof net.

The research leading to this work originated from the question of whether it is possible to define a direct presentation of proof nets for propositional intuitionistic logic, with a topological correctness criterion reminiscent of the one by Danos and Regnier [12] for multiplicative linear logic, that is, formulated in terms of paths in the so-called *tests* of a proof net. Specifically, we were interested whether such criterion could be defined using a single test as in the criterion given by Lamarche [34] for essential nets for propositional



© Matteo Acclavio and Dylan Bettendroffer;
licensed under Creative Commons License CC-BY 4.0

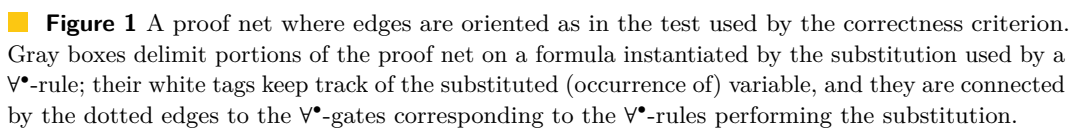
42nd Conference on Very Important Topics (CVIT 2016).

Editors: John Q. Open and Joan R. Access; Article No. 23; pp. 23:1–23:27

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany



intuitionistic linear logic, and whether such ‘single-test’ approach could be extended to second-order in this intuitionistic setting.

Contribution of the paper. We define proof nets for the (disjunction-free) fragment of second-order intuitionistic logic that underlies the typing discipline for System F without relying on encodings in linear logic. We provide a simple correctness criterion for these nets inspired by Lamarche’s criterion for essential nets for propositional intuitionistic linear logic [34], where the correctness criterion is defined in terms of paths in a single graph constructed from the proof net. For this purpose, we make use of *input/output polarities* as in [38, 34, 2] (not to be confused with the ones in [36, 37, 3]) providing an intuitive way to orient the edges in the test of a proof net, drastically restricting the search space of paths to be checked for correctness. See Figure 1 for an example of a proof net.

We also discuss the proof equivalence induced by our proof nets, and we introduce a finer notion of *canonical proof net* to guarantee with better modularity properties (on derivations) with respect to general proof nets, and to prove a full local canonicity result.

The two main challenges in defining a correctness criterion for proof nets for second-order intuitionistic logic are the following:

- Jumps are not needed in intuitionistic logic because of the different properties of its proof
 equivalence. However, they are strictly necessary in Lamarche’s criterion to keep track
 of the dependencies of weakening rules and to ensure the connectedness condition. We
 identify a simple condition that allows us to avoid jumps by exploiting the fact that the
 sequent-calculus transformation known as *excising* (see Figure 7) allows us to replace
 entire subproofs with weakenings in specific contexts. These weakenings then become the
 only components of the test that are not connected to the part of the net corresponding
 to the consequent of the sequent.
- Quantifiers introduce non-local dependencies between rules in proof nets, needed to ensure
 the substitutions required in the proof respect the side condition of the sequent calculus
 rules. These dependencies are typically dealt with some sort of jumps connecting the
 quantifiers related by some dependencies, thus increasing the number of graphs which need
 to be checked by the correctness criterion. Our approach allows us to keep the correctness
 criterion simple, only requiring to construct a single graph to check the correctness of a

proof net.

Why proof nets for intuitionistic logic? A legit question is why one should consider proof nets for intuitionistic logic, considering that natural deduction already provides a good account of proofs in intuitionistic logic.

A first reason, is the fact that the notion of proof equivalence induced by natural deduction may be considered too coarse, and that a finer notion of proof equivalence may be desirable: there are sequent calculus proofs differing by an exponential factor that are identified by a same natural deduction derivation. This because natural deduction not only identifies proofs differing for those *local* rule permutations that are necessary to deal with the aforementioned commutative cut-elimination steps, but also some *non-local* rule permutations. Those non-local rule permutations (see Remark 27) duplicate entire subproofs, leading to the exponential blow-up in the size of proofs identified by natural deduction, and which are not necessary to properly define cut-elimination steps as local rewriting steps. In this regard, *combinatorial proofs* [51] for propositional intuitionistic logic have shown to provide a finer account of proof equivalence for (cut-free) proofs modulo local rules permutations only, preventing the exponential blow-up of the size of proofs identified by natural deduction¹, but this syntax relies on a decomposition result of proofs into a linear and a structural part, which is not possible in the second-order case because of the presence of second-order quantification interacting with both the linear and the structural part of proofs.

A second reason is that the theory of proof nets is interested in the topological characterization of proofs as special families of graphs naturally constructed from the syntax of proofs, and in the study of the properties of those graphs. In [54, 7, 24, 15] this takes form of a topological characterization of the proof in natural deduction, and in [51] it requires a separation result that may seem as a partial result in this direction. In presence of second-order quantification, the topological characterization of proofs deals with more complex dependencies, and finding a simple correctness criterion for proof nets for second-order logic is not at all a trivial extension of the propositional case, and it is interesting in its own right. Further requiring a single graph to be checked for correctness, as in Lamarche's criterion, is even more challenging.

Structure of the paper. In Section 2 we recall some basic definitions and notations that will be used throughout the paper. In Section 3 we define (cut-free) proof nets, in Section 4 we give their correctness criterion and the sequentialization procedure. In Section 5 we discuss the proof equivalence induced by proof nets, and we define canonical proof nets, allowing for a stronger and modular notion of proof equivalence. Finally, in Section 6 we conclude by showing how to include the cut rule in our syntax, and we discuss future work. Due to space limitations, some proofs are only sketched in the body of the paper, but their detailed version can be found in the appendix.

2 Background

In this section, we recall some definitions and the notation we adopted in the paper.

2.1 Graphs

In this paper, we use the following notation for (undirected labeled simple) graphs.

¹ As a consequence of Section 5, the proof equivalence induced by our canonical proof nets for the propositional fragment is the same as the one induced by combinatorial proofs.

► **Definition 1.** A (labeled) **graph** is a triple $\mathcal{G} = \langle V_{\mathcal{G}}, \rightharpoonup_{\mathcal{G}}, \ell_{\mathcal{G}} \rangle$ made of:

- a set $V_{\mathcal{G}}$ of **vertices**;
- a set $\rightharpoonup_{\mathcal{G}}$ of **edges** of the form vw with **source** v and **target** w distinct vertices in $V_{\mathcal{G}}$;
- $\ell_{\mathcal{G}}$ is a **labeling function** $V_{\mathcal{G}} \mapsto \mathbb{S}$ assigning a **label** to each vertex.

We may write $v \rightharpoonup w$ if $vw \in \rightharpoonup_{\mathcal{G}}$ and we say that two vertices u and v are **adjacent** if $u \rightharpoonup v$ or $v \rightharpoonup u$. A graph is **bipartite** if the set of vertices can be partitioned into two (disjoint) sets such that no two vertices in the same set are adjacent. A **root** of $\mathcal{G}$ is a vertex such that vw is not in $\rightharpoonup$ for any $w \in V_{\mathcal{G}}^2$. We denote by $\mathfrak{R}_{\mathcal{G}}$ the set of roots of $\mathcal{G}$.

The **subgraph induced** by a set of vertices $V_{\mathcal{G}'} \subseteq V_{\mathcal{G}}$ has vertices $V_{\mathcal{G}'}$, edges those in $\rightharpoonup_{\mathcal{G}}$ connecting vertices in $V_{\mathcal{G}'}$, and labeling function the restriction of $\ell_{\mathcal{G}}$ to $V_{\mathcal{G}'}$.

A **path** p of length $n = |p|$ in $\mathcal{G}$ from u to v is a sequence of vertices $p_0 \cdots p_n$ such that $p_0 = u$ and $p_n = v$, and such that $p_i \rightharpoonup p_{i+1}$ for all $i \in \{0, \dots, n-1\}$. A **cycle** is a path p with length $n > 0$ such that $p_0 = p_n$. A **directed acyclic graph** is a graph containing no cycle.

2.2 Second order Intuitionistic Logic

We consider **formulas** generated by a countable set of propositional **atoms** $\mathbb{X} = \{X, Y, Z, \dots\}$, the binary connectives for **implication** ($\rightarrow$) and **conjunction** ($\wedge$), and by the **universal quantifier** ($\forall$) using the following grammar:

$$A, B ::= X \mid A \wedge B \mid A \rightarrow B \mid \forall X. A \quad X \in \mathbb{X} \quad (1)$$

The set of **free variables** of a formula A as the set $FV(A)$ of propositional variables in A not bound by a $\forall$ -quantifier.

► **Notation 2.** In this work, it is of crucial importance to distinguish different occurrences of the same variable in a formula. To avoid confusion, we may annotate different occurrences of the same formula A with indices (e.g., A^1, A^2) to distinguish them, while using pedicles to denote different formulas (e.g., A_1, A_2). For example, in the formula $(X_1 \wedge X_2^1) \rightarrow (X_2^2 \wedge Y)$ we have four different occurrences of propositional variables: one occurrence of X_1 , one occurrence of Y , and two distinct occurrences of X_2 .

A **sequent** $\Gamma \vdash A$ is a set of occurrences of formulas Γ , called **context**, and a formula A . From now on, we assume sequents to be **clean**, that is such that no variable occurs both free and bound it. The **parity** of a sub-formula, a connective, or a quantifier occurring in a sequent is the parity of the number times it occurs in the left of a $\rightarrow$ and $\vdash$ in the parse tree of the sequent. We write x° (resp. $x^\bullet$) to denote that the parity of x is even (resp. odd).

A **substitution** $\sigma = \{X_1 \leftarrow A_1, \dots, X_n \leftarrow A_n\}$ is a partial map with (finite) **domain** $\{X_1, \dots, X_n\} \subset \mathbb{X}$ and codomain the set of formulas. The formula $B\sigma$ is the formula obtained by simultaneously substituting each occurrence of X_i in B with the formula A_i for each $i \in \{1, \dots, n\}$. As standard, we assume that bound variables are renamed to avoid capture of free variables by the substitution.

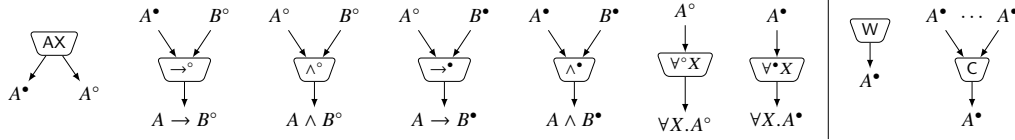
In Figure 2 we recall the rules of the sequent calculus LJ₂ for second order intuitionistic logic³. A formula occurrence is **active** (resp. **principal**) for a rule application if it occurs

² The use of the word *root* for vertices with no outgoing edges is justified by the fact that the ‘forest-like’ graphs we consider in this paper (proof nets and strategies) have edges oriented from premises to conclusions.

³ Note that *cleanness* is bottom-up preserved by rules. However, if we read the rules top-down, we have to assume formulas to be identified up to α -equivalence to ensure cleanness to be preserved by binary rules.

$$\begin{array}{c}
\text{AX} \frac{}{A \vdash A} \quad \wedge^\bullet \frac{\Gamma, A, B \vdash C}{\Gamma, A \wedge B \vdash C} \quad \rightarrow^\circ \frac{\Gamma, A \vdash B}{\Gamma \vdash A \rightarrow B} \quad \wedge^\circ \frac{\Gamma \vdash A \quad \Delta \vdash B}{\Gamma, \Delta \vdash A \wedge B} \quad \rightarrow^\bullet \frac{\Gamma \vdash A \quad B, \Delta \vdash C}{\Gamma, A \rightarrow B, \Delta \vdash C} \\
\\
\forall^\bullet \frac{\Gamma, A \{X \leftrightarrow B\} \vdash C}{\Gamma, \forall X. A \vdash C} \quad \forall^\circ \frac{\Gamma, \vdash A}{\Gamma \vdash \forall X. A} \quad X \notin \text{FV}(\Gamma) \quad \text{C} \frac{\Gamma, A, A \vdash C}{\Gamma, A \vdash C} \quad \text{W} \frac{\Gamma \vdash C}{\Gamma, A \vdash C}
\end{array}$$

■ **Figure 2** Sequent rules for LJ_2 .



■ **Figure 3** Local configuration of gates. A C-gate can have any positive number of premises.

154 in one of its premises (resp. in its conclusion) but not in its conclusion (resp. none of its
 155 premises). We may denote by $r \frac{\Gamma' \vdash A'}{\Gamma \vdash A}$ a bottom-up sequence of applications of the same
 156 rule r from the sequent $\Gamma \vdash A$ to the sequent $\Gamma' \vdash A'$.

157 3 Proof Nets for LJ_2

158 Proof nets are a graph-based syntax for proofs in linear logic, introduced by Girard [18] to
 159 abstract away from syntactic bureaucracy of sequent calculus proofs. The main idea is to
 160 replace inference rules in the sequent calculus with vertices of a graph, called *gates*, keeping
 161 track only of the dependencies between the active and principal formulas of the rules. In
 162 presence of quantifiers, the syntax of proof nets is usually enriched with additional edges to
 163 keep track of the binding relations of formulas in the proof net (see, e.g., [19, 20, 40, 50, 42]).
 164 We here use the same approach, by enriching the label of vertices corresponding to quantifier
 165 rules with the variable they bind. Moreover, as in other works on proof nets for intuitionistic
 166 linear logic (see, e.g., [34, 37, 2, 14]), we also use *polarization* to keep track of the parity of
 167 formulas in the sequent, which is crucial for the definition of the correctness criterion in the
 168 next section.

169 ► **Definition 3.** A *pre-net* is a bipartite graph with set of vertices partitioned into:

- 170 ■ **wires** labeled by *polarized formulas*, that is, formulas annotated with a polarity in
 171 $\{\circ, \bullet\}$ in the set $\{A^\circ, A^\bullet \mid A \text{ is a formula}\}$
- 172 ■ **gates** labeled by the rules of LJ_2 , where the rules for the quantifiers are additionally
 173 indexed by the variable they bind. That is, gates have labels in $\{\text{AX}, \wedge^\bullet, \wedge^\circ, \rightarrow^\bullet, \rightarrow^\circ, \text{C}, \text{W}\} \cup$
 174 $\{\forall^\circ X, \forall^\bullet X \mid X \in \mathbb{X}\}$

175 such that a gate and its adjacent vertices induce graphs as the ones in Figure 3.

176 We say that a vertex v in a pre-net is **below** another vertex w (resp. w is **above** v) if
 177 there is a path from w to v . A gate g is called a r -gate if its label is r ; its **premises** (resp.
 178 **conclusion**) are those vertices which are source (resp. target) of an edge connected to the
 179 gate. A **conclusion gate** is a gate with no gate below it; its conclusion is a $\circ$ -root or a
 180 $\bullet$ -root according to whether its conclusion has label A° (resp. $A^\bullet$). A **leaf** is a wire which is
 181 the conclusion of an AX- or W-gate, or no gate at all. An **eigenleaf** of a $\forall^\circ X$ -gate g is a leaf
 182 above g with label X° . Two leaves are **linked** if they are the conclusion of a same AX-gate.

183 The parse tree of a formula A can be naturally represented as a tree-like pre-net with
 184 leaves polarized variables, and internal nodes labeled by ‘polarized connectives’ and ‘polarized
 185 quantifiers’, that is, connectives and quantifiers labeled with $\circ$ (resp. $\bullet$) if the parity of their
 186 occurrence in A is even (resp. odd)⁴. Formally, this leads to the following definition.

187 ► **Definition 4.** *The **pre-net of a formula** A is the pre-net $\llbracket A \rrbracket$ with wires the subformulas
 188 of A labeled according to their parity, with gates the connectives and quantifiers of A labeled
 189 according to their parity (and the bound variable for quantifiers), and edges connecting each
 190 formula to the gate of the main logical connective or quantifier, and from this to its immediate
 191 subformulas as shown in Figure 3.*

192 In presence of quantifiers, the encoding of derivations also needs to include additional
 193 edges, called *pointers*, to keep track of the dependencies between quantifiers and the variables
 194 they bind. Their need is due to the fact that the rules for quantifiers involve substitutions of
 195 sub-formulas which are not necessarily immediate sub-formulas of the conclusion, but rather
 196 can be any formula above the quantifier.

197 *Nets* are defined as pre-net with pointers respecting some additional conditions, ensuring
 198 that the pointers are well-behaved with respect to the discipline of variable scoping and
 199 substitutions. In particular, we ask that pointers from a same $\forall^\bullet X$ -gate are (1) uniform, that
 200 is, their targets are labeled by the same formula (ignoring polarities), that (2) no variable
 201 can be both free and bound in a net, and that (3) no variable introduced by a substitution
 202 can be captured by the binding of another quantifier.

203 ► **Definition 5.** *A **pointer** in a pre-net is an edge from a $\forall^\bullet X$ -gate to a vertex above it. Its
 204 **tag** is an occurrence of the variable X ⁵. The **reach** of a pointer is the set of vertices that
 205 contains its target, and all the vertices above it that are not AX - or W -gates. The **reach** of a
 206 $\forall^\bullet X$ -gate $\mathbf{g}$ is the union of the reaches of all pointers with source $\mathbf{g}$. A $\forall^\circ Y$ -gate is **guarding**
 207 a $\forall^\bullet X$ -gate if one of its eigenleaves is linked to a leaf in the reach of the $\forall^\bullet X$ -gate.*

208 *A **net** is a pre-net provided with a set of pointers satisfying the following conditions:*
 209 1. **uniformity:** *all targets of pointers with the same source $\forall^\bullet X$ are labeled by the same*
 210 *formula A containing an occurrence of the variable X ;*
 211 2. **cleanness:** *if there is a $\forall^\circ X$ -gate or a $\forall^\bullet X$ -gate, then no formula labeling a root can*
 212 *contain free occurrences of X ;*
 213 3. **safeness:** *if the target w of a pointer vw is within the range of another pointer, then its*
 214 *source v must also be within the range of that pointer.*

215 In the propositional case, the proof net encoding a derivation is defined as the graph
 216 having a r -gate for each r -rule, connected to the wire encoding the principal and active
 217 formula of the rule similarly to how shown in Figure 3. However, the presence of quantifiers
 218 (and pointers) complicates the definition of an inductive translation. We define the translation
 219 from derivations to proof nets as the composition of two transformations: the first one,
 220 defined in Figure 4, defines a pre-net from a derivation, while the second one adds pointers
 221 to obtain a proof net.

⁴ The notation for the inference rules has been chosen to match the label of the corresponding connective or quantifier labeled according to its parity. For example, each even implication in a sequent can only be introduced by a $\rightarrow^\circ$ -rule, and the corresponding gate in the proof net is labeled by $\rightarrow^\circ$.

⁵ We assume each tag to be unique, that is, each is a different occurrence of the same variable. As a convention, we will use even (resp. odd) number to index tags according to if the target is labeled by a polarized formula A° (resp. $A^\bullet$).

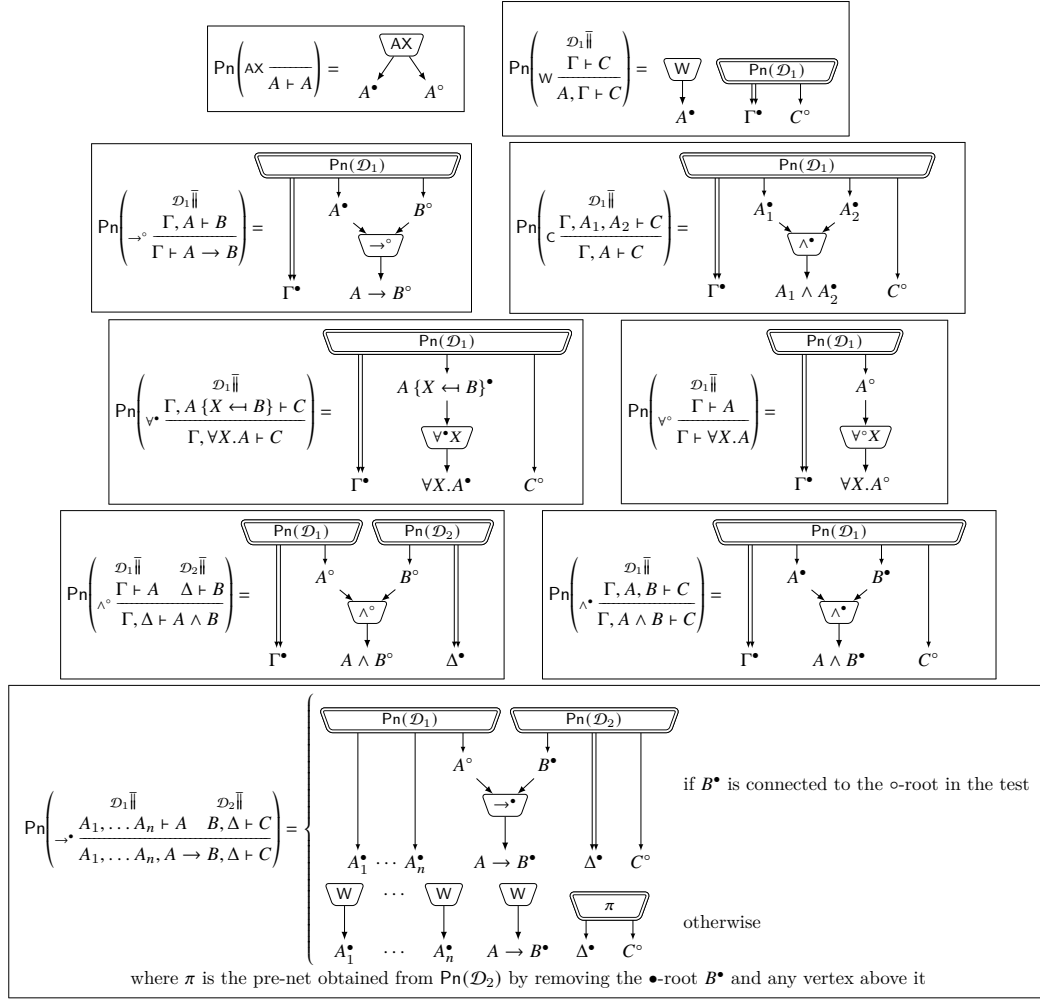
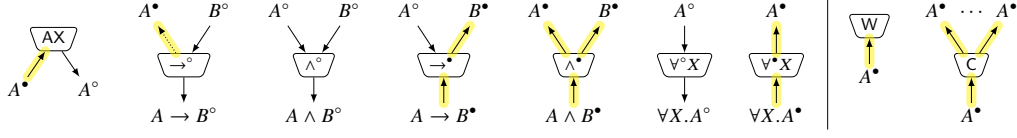


Figure 4 Translation of LJ_2 derivations to (non-canonical) nets.

Note that the translation from derivations to pre-net (Figure 4) requires a case analysis in the translation of the $\rightarrow^\bullet$ -rule, taking into account a special case to deal with the interaction of $\rightarrow^\bullet$ -rules with W-rules. In fact, the interaction of these two rules in intuitionistic logic, which we may refer to as *weakening propagation*, enforces a quotient of proofs which cannot be observed in linear logic due to the resource-sensitivity of this latter one, and it is a key factor on the simplicity of the correctness criterion for proof nets for intuitionistic logic. We comment more on this point in the following sections.

► **Definition 6** (Proof nets). Let $\mathcal{D}$ be a derivation in LJ_2 . The **pre-net** $\text{Pre}(\mathcal{D})$ is the pre-net defined by induction on $\mathcal{D}$ using the translation in Figure 4. The **proof net** of $\mathcal{D}$ is the net $\text{Pn}(\mathcal{D})$ obtained by adding to its **underlying pre-net** $\text{Pre}(\mathcal{D})$ a pointer from each $\forall^\bullet X$ -gate to all vertices above it that correspond to the (root of the) formula obtained by substituting the variable X bound by the quantifier.

► **Notation 7.** When drawing proof nets, we encapsulate the reach of a pointer inside a gray area and we add the tag of the pointer as a white circle at its bottom as follows  to



■ **Figure 5** Polarized orientation of the edges, where we highlighted those edges whose orientation is flipped with respect to the original orientation in the nets. The left premise of the $\rightarrow^\circ$ gate is dotted because it is removed in the test of a net.

ease the visualization of the substitution of the variable X bound by the quantifier. In this case, we use the tagging circle as source/target of the pointer and of any edge connecting the target of the pointer with a gate below it (see Figure 1). This notation is meant to visually keep track of the dependencies between quantifiers, the variables they bind, and the formula which may come from a substitution of a bound variable.

4 Correctness Criterion

In this section, we define a *correctness criterion* allowing us to decide whether a net is a proof net (i.e. a topological characterization of proof nets). As inspiration for our correctness criterion, we take Lamarche's correctness criterion for his proof nets for intuitionistic linear logic [34], called essential nets: we define the *polarized orientation* of a pre-net by reversing the orientation of edges connected to $A^\bullet$ -formulas (as done in [34, 37], but in the opposite direction).

Definition 8. The *polarized orientation* of a pre-net is the graph obtained by **flipping** the orientation of any edge with source or target a $A^\bullet$ -formula. That is, the polarized orientation of a pre-net is the graph obtained by replacing each edge uv , such that either u or v is labeled by a polarized formula $A^\bullet$, is replaced by the edge vu (see Figure 5).

We then define the *test* of a pre-net in the same way as the graph used in [34] to define the correctness criterion, that is, as the graph obtained by removing from the polarized orientation of a pre-net the left premises of $\rightarrow^\circ$ -gates, that is, the dotted edges in Figure 5.

Definition 9. The *test* of a net is the graph obtained by taking the polarized orientation of its underlying pre-net and removing the edge connecting each $\rightarrow^\circ$ -gate to its left premise.

Note that tests of linear logic essential nets are always connected graphs. This property is ensured by the proper use of additional edges, called *jumps*, connecting W -gates to some other vertices of the proof net. Jumps have the drawback of making proof nets non-canonical (even modulo the usual commutative conversions), as they introduce additional dependencies between rules, jeopardizing canonicity (see the 'disaster' causing in proof equivalence in multiplicative linear logic [25]). However, the coarser proof equivalence of intuitionistic logic allows us to get rid of jumps by weakening the functionality condition of essential nets (which usually relies on the existence of a path from any $\rightarrow^\circ$ -gate to the $\circ$ -root of the pre-net), and compensating this with a new condition, we call *excising*, which requires that every $\rightarrow^\bullet$ -gate admits a path to the $\circ$ -root in the test.

The intuition of why this trade-off works is related to the aforementioned *excising* interaction between weakening and $\rightarrow^\bullet$ (see Figure 6), connected to the weakening propagation we have in the proof translation. In linear logic, the rigid resource-sensitivity prevents such interaction, and is the reason why jumps are needed to ensure the connectivity of the test:

the information ‘after a jump’ cannot be ‘excised’, while in intuitionistic logic it can⁶. We discuss this issue in more detail in Section 5.

► **Definition 10.** *A net is **correct** if it satisfies the following conditions:*

1. **one-sidedness:** *there is a unique $\circ$ -root;*
2. **acyclicity:** *the test is an acyclic graph;*
3. **functionality:** *each $\rightarrow^\circ$ -gate is **functional**, that is, any path in the test reaching the $\circ$ -root from its left premise also contains its right premise;*
4. **excising:** *each $\rightarrow^\bullet$ -gate is **relevant**, that is, it has at least one path in the test from its right premise to the $\circ$ -root;*
5. **deadlock-free:** *no $\forall^\bullet X$ -gate is **stuck**, that is, all paths in the test from its premise to the $\circ$ -root contains all the $\forall^\circ Y$ -gates guarding it.*

► **Remark 11.** Condition 5 is the only condition in the correctness criterion which explicitly involves pointers. The other conditions are purely topological conditions on the underlying pre-net. In particular, the first three conditions are the same as the conditions of Lamarche’s correctness criterion for essential nets. The fourth condition is a new condition, which is not present in Lamarche’s criterion, but it suffices to characterize the weakening propagation in intuitionistic logic, and it is not needed in linear logic because of the presence of jumps. Together with the first three conditions, it is enough to have a correctness criterion for the propositional fragment of intuitionistic logic.

The last condition is a new condition, which deals with the dependencies created by the side condition of the $\forall^\circ$ -rule. Notice that this condition is defined for the $\forall^\bullet$ -gates, and not for the $\forall^\circ$ -gates. This is because, during sequentialization, the side condition of the $\forall^\circ$ -rule is enforced by the cleanness condition (Definition 5.2), which ensures that no variable can be captured by a $\forall^\circ$ -gate if it is not ‘in plain sight’ in the proof net, and this is enough to ensure that the $\forall^\bullet$ -rule can be applied in a correct way. This shifts attention from the $\forall^\circ$ -gates to the $\forall^\bullet$ -gates: during sequentialization, we can remove a $\forall^\bullet$ -gate only if it will not expose variables which are bound by a $\forall^\circ$ -gate, violating the cleanness condition. The name ‘deadlock-free’ is meant to capture this idea that if the condition is violated, then the $\forall^\bullet$ -gate cannot be sequentialized because of cyclic dependencies in the sequentialization order with some $\forall^\circ$ -gates. See Example 28 below.

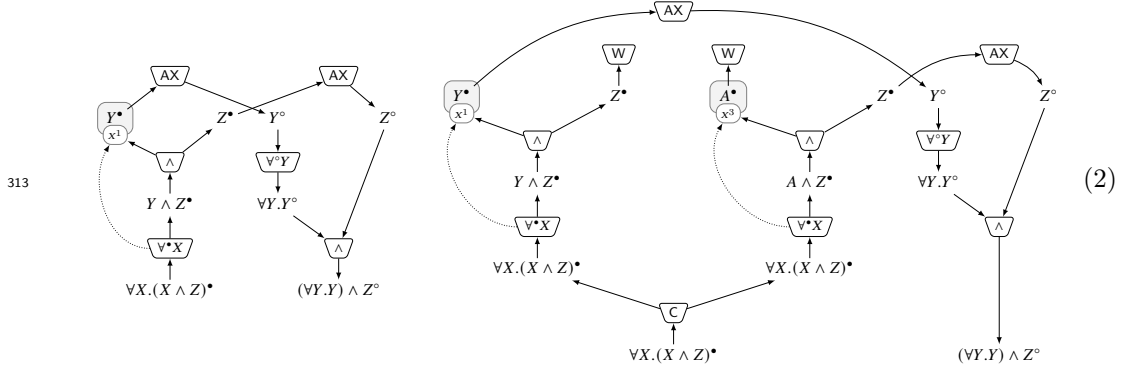
The remaining expected conditions on pointers, such as the uniformity, cleanness, and safeness conditions were already included in Definition 6. The design choice of separating the conditions on the underlying pre-net from the other conditions on pointers is motivated by the fact that we want to distinguish the topological conditions of the underlying pre-net capturing the fact that rules can be applied in a correct order, from the conditions ensuring that substitutions (associated to the $\forall^\bullet X$ -rule) are applied in a correct way.

► **Example 12.** Below, two nets of the same sequent $\forall X.(X \wedge X) \vdash (\forall Y.Y) \wedge Z$. The one on the left is not correct because the $\forall^\bullet X$ -gate is stuck: the path to the $\circ$ -root passing through the Z formula does not pass through the $\forall^\circ Y$ -gate, which is guarding the $\forall^\bullet X$ -gate. The one on the right is correct because in the left copy of $\forall X.(X \wedge X)$, the Z is discarded, preventing

⁶ In the game semantics for multiplicative exponential linear logic in [10], this same interaction of weakening and $\rightarrow^\bullet$ -rules is also enforced, as plays of a game can never continue playing a weakened formula, nor continue to the left premise of a $\rightarrow^\bullet$ -rule whose active formula in the right premise comes from weakening. However, this is not part of the denotational semantics of the logic, but rather a consequence of the translation from proofs to strategies, which enforces excising. As a consequence, full-completeness cannot be achieved in these games.

23:10 Proof Nets for Second Order Intuitionistic Logic

the existence of a path to the $\circ$ -root which does not pass through the $\forall^\circ Y$ -gate, while the $\forall^\bullet X$ -gate in the copy on the right is not stuck, as it has no path to the $\circ$ -root.



A detailed explanation is provided in Example 28 in Section A.

► **Proposition 13.** *Given a net provided with a set of pointers on it, it is possible to check in polynomial time (on the number of vertices of the net) whether it is a correct net.*

Proof. Uniformity condition and cleanness condition of the net can be checked in linear time, while proper instantiation can be checked in polynomial time with a traversal of the graph. At the same time, given a net, it is possible to check in polynomial time on the number of vertices of the net whether it is correct: one-sidedness can be checked in linear time, while acyclicity, functionality and excising can be checked in polynomial time with a traversal of the graph. ◀

To prove that the correctness criterion provides a sound and complete characterization of proof nets, we first show that any proof net is a correct net.

► **Lemma 14.** *A proof net is a correct net.*

Sketch of the proof. By induction on the structure of the derivation $\mathcal{D}$ used to define the proof net $\text{Pn}(\mathcal{D})$, reasoning on the bottom-most rule of the derivation. The only two cases requiring some care are the $\rightarrow^\bullet$ -rule and $\forall^\circ$ -rules. The first requires a case distinction whether the right premise of the $\rightarrow^\bullet$ -rule is connected to the $\circ$ -root in the test or not. The second one requires checking that the eigenleaf of a $\forall^\circ X$ -gate has its $\bullet$ -sibling in the reach of a $\forall^\bullet Y$ gate, following from cleanness of our sequents. ◀

4.1 Splitting and Sequentialization

To prove that any correct net is a proof net, we provide a **sequentialization** procedure, showing how to reconstruct a derivation in LJ_2 from a correct net. A key step in any sequentialization procedure for proof nets is to identify the last rule of the derivation, which corresponds to a conclusion gate of the net. For unary rules, this is straightforward, but for binary rules ($\rightarrow^\bullet$ and $\wedge^\circ$) this is more complex, as binary inference rules in LJ_2 requires to split the context in two parts. For this reason, we first prove a splitting lemma, showing that if the sequentialization procedure arrives at a point in which the net needs to be split in two, then it is always possible to find the right conclusion gate such that removing it does not break correctness.

► **Definition 15.** *A conclusion gate of a net is **splitting** if, after removing the gate and its conclusion, the remaining graph decomposes into two disjoint nets— i.e., the set of remaining*

vertices can be partitioned into two disjoint subsets such that no edge of the original net connects vertices belonging to different subsets.

To prove splitting, we adapt the proof of the splitting lemma for essential nets in [34] to our setting, and we show that the additional conditions of excising and deadlock-free are preserved by removing a splitting conclusion gate.

► **Definition 16.** Let x and y be vertices in a correct net $\mathcal{G}$. We write $x \preceq y$ if every path in the test of $\mathcal{G}$ from x to the $\circ$ -root contains y . The **empire** of a vertex x° is the set $[x]$ of vertices of the net which are $\preceq$ -smaller than x° , the $\preceq$ -minimal gates in $[x]$ are called **advisors** of x° , and we denote by $\pm(x)$ the set of advisors of x° . The **marshal** of a vertex y , denoted $\mathfrak{M}(y)$ is the $\preceq$ -smallest wire z° such that $y \in [z]$.

► **Lemma 17.** Let $\mathcal{G}$ be a correct net with $\circ$ -root r and z° a vertex of $\mathcal{G}$. Then:

1. the relation $\preceq$ is an order inducing a tree-like structure on the vertices of $\mathcal{G}$;
2. r is the $\preceq$ -maximum, and every $\rightarrow^\bullet$ -gate is $\preceq$ -smaller than r ;
3. if $x \in \pm(z)$, then $x^\bullet$. Moreover, if x is a premise of a gate $\mathfrak{g}$, then $z \neq \mathfrak{M}(\mathfrak{g})$;
4. if $\mathfrak{g}$ is a $\rightarrow^\circ$ -gate with premises x° and $y^\bullet$, then $\mathfrak{M}(x) \preceq \mathfrak{M}(y)$. If y is the premise of the $\preceq$ -greatest $\forall^\circ$ -gate guarding a $\forall^\bullet$ -gate $\mathfrak{g}$, then $\mathfrak{M}(\mathfrak{g}) \preceq y$.
5. if x is a premise of a gate with conclusion y , then $\mathfrak{M}(x) \preceq \mathfrak{M}(y)$. Moreover, if additionally $x^\bullet$ and $\mathfrak{M}(x) = r^\circ$, then $\mathfrak{M}(y) = r^\circ$ and $y^\bullet$.

Sketch of the proof. Item 1 is a consequence acyclicity, while Item 2 is a consequence of one-sidedness and excising. Item 3 is a consequence of the definition of the polarized orientation, while Items 4 and 5 are consequences of functionality and deadlock-free. ◀

► **Lemma 18 (Splitting).** Let $\mathcal{G}$ be a correct net with more than two AX-gate. If all conclusion gates of $\mathcal{G}$ have labels in $\{\text{AX}, \rightarrow^\bullet, \wedge^\circ, \forall^\bullet X\}$ and all conclusion $\forall^\bullet X$ -gates are guarded, then at least one of its conclusion gates is splitting.

Proof. We prove that either an advisor of the $\circ$ -root r° of $\mathcal{G}$ is the conclusion of a splitting $\rightarrow^\bullet$ -gate, or r° must be the conclusion of a splitting $\wedge^\circ$ -gate.

We first show that if the set $\mathcal{X}$ of conclusion $\rightarrow^\bullet$ -gates of $\mathcal{G}$ with marshal r° is non-empty, then there is $\mathfrak{g} \in \mathcal{X}$ whose left-premise $x_{\mathfrak{g}}$ is such that $\pm(x_{\mathfrak{g}}) \subset \pm(r)$. This is proven by contradiction. Take $\mathfrak{g} \in \mathcal{X}$ and its left-premise v . We show that if an advisor of v is not in $\pm(r)$, then such an advisor must be (strictly) above another advisor w of r° . If w is a $\forall^\bullet$ -gate, then it is guarded by hypothesis, and we should have $\mathfrak{M}(w) = r^\circ$ by Lemma 17.5; this is impossible because $\mathfrak{M}(w) \preceq y^\circ$ with y the premise of the $\preceq$ -greatest $\forall^\circ$ -gate guarding w . Since w cannot be the conclusion of a AX-gate, then it must be a $\rightarrow^\bullet$ -gate, and we can repeat the argument on its left premise, ultimately finding the desired $\rightarrow^\bullet$ -gate by finiteness and acyclicity of $\mathcal{G}$.

If $\mathcal{X}$ is non-empty, we select any advisor in it whose left premise x° is such that $\pm(x) \subset \pm(r)$, and we prove that $\mathfrak{g}$ is splitting by letting $\mathcal{G}_1$ (resp. $\mathcal{G}_2$) be the subgraph induced by the vertices in (resp. not in) the empire of x° . The correctness of $\mathcal{G}_2$ follows trivially from the one of $\mathcal{G}$. For $\mathcal{G}_1$, correctness follows from the fact that any path from a vertex y in $\mathcal{G}$ to its $\circ$ -root r° must pass through x° .

Otherwise, there is no advisor of the $\circ$ -root r° which has marshal r° itself and r° cannot be the conclusion an AX-gate, because the $\bullet$ -sibling of r° should be above another advisor whose marshal should necessarily be r° . Therefore, r° must be the conclusion of a $\wedge^\circ$ -gate, and we can conclude by letting $\mathcal{G}_1$ and $\mathcal{G}_2$ be the subgraphs induced by the empires of the left and right premise of r° , respectively. Correctness of $\mathcal{G}_1$ and $\mathcal{G}_2$ follows from the disjointness of the empires of the two premises, and the $\preceq$ -incomparability of their elements. ◀

$$\begin{array}{c}
\boxed{
\begin{array}{c}
\frac{\frac{\Gamma_1 \vdash C_1}{\Gamma_1, \Gamma', \Gamma_3, \Delta' \vdash C} \quad \frac{\frac{\Gamma_2 \vdash C_2 \quad \Gamma_3, \Delta \vdash C_3}{\Gamma_2, \Gamma_3, \Delta' \vdash C} \quad \frac{\Gamma_1 \vdash C_1 \quad \Gamma_2, \Delta_2 \vdash C_2}{\Gamma_1, \Gamma', \Delta_2 \vdash C_2}}{\Gamma_1, \Gamma', \Gamma_3, \Delta' \vdash C} \quad \sim \quad \frac{\frac{\Gamma_1 \vdash C_1 \quad \Gamma_2, \Delta_2 \vdash C_2}{\Gamma_1, \Gamma', \Delta_2 \vdash C_2} \quad \Gamma_3, \Delta_3 \vdash C_3}{\Gamma_1, \Gamma', \Gamma_3, \Delta' \vdash C} \\
\rho_2 \quad \rho_1 \quad \rho_1
\end{array}
}
\quad
\boxed{
\begin{array}{c}
\frac{\frac{\Gamma, \Sigma', \Delta' \vdash C'}{\Gamma, \Sigma', \Delta \vdash C} \quad \frac{\Gamma, \Sigma', \Delta' \vdash C'}{\Gamma, \Sigma, \Delta \vdash C}}{\Gamma, \Sigma, \Delta \vdash C} \quad \sim \quad \frac{\frac{\Gamma, \Sigma', \Delta' \vdash C'}{\Gamma, \Sigma, \Delta \vdash C} \quad \frac{\Gamma, \Sigma', \Delta' \vdash C'}{\Gamma, \Sigma, \Delta \vdash C}}{\Gamma, \Sigma, \Delta \vdash C} \\
\tau_1 \quad \tau_2 \quad \tau_1
\end{array}
}
\\
\boxed{
\begin{array}{c}
\frac{\frac{\Gamma, \Delta_1, \vdash C_1 \quad \Sigma' \vdash C_2}{\Gamma, \Delta', \Sigma \vdash C} \quad \frac{\Gamma, \Delta' \vdash C_1}{\Gamma, \Delta \vdash C_1 \quad \Sigma' \vdash C_2}}{\Gamma, \Delta, \Sigma \vdash C} \quad \sim \quad \frac{\frac{\Gamma, \Delta' \vdash C_1}{\Gamma, \Delta \vdash C_1 \quad \Sigma' \vdash C_2}}{A, \Gamma, \Delta \vdash C} \\
\rho_1 \quad \tau_2 \quad \rho_1
\end{array}
}
\quad
\boxed{
\begin{array}{c}
\frac{\frac{\Gamma \vdash C_0 \quad C_1, \Delta, A \vdash B}{\Gamma, \Sigma, \Delta', \vdash A'} \quad \frac{C_1, \Delta' \vdash A'}{C_1, \Delta \vdash A}}{\Gamma, \Sigma, \Delta \vdash A} \quad \sim \quad \frac{\Gamma \vdash C_0 \quad \frac{C_1, \Delta' \vdash A'}{C_1, \Delta \vdash A}}{\Gamma, \Sigma, \Delta \vdash A} \\
\rho_2 \quad \tau_1 \quad \rho_2
\end{array}
}
\end{array}$$

■ **Figure 6** Independent rules permutations in LJ_2 where $\rho_1 \in \{\wedge^\circ, \rightarrow^\bullet\}$ and $\rho_2 \in \{\rightarrow^\bullet\}$ and $\tau_1 \in \{\wedge^\bullet, \rightarrow^\circ, \forall^\bullet, \forall^\circ, \text{W}, \text{C}\}$ and $\tau_2 \in \{\wedge^\bullet, \forall^\bullet, \text{W}, \text{C}\}$ (symmetric cases omitted).

► **Lemma 19** (Sequentialization). *Let π be a correct net for a sequent $\Gamma \vdash C$. Then, there is a derivation $\mathcal{D}_\pi$ in LJ_2 with conclusion $\Gamma \vdash C$.*

Sketch of the proof. We define a *sequentialization* procedure to construct a derivation $\mathcal{D}_\pi$ in LJ_2 from a correct net π by induction on the number n of gates of π . If $n = 1$, then π is an AX-gate, and the corresponding derivation is the AX-rule. Otherwise, we proceed by sequentializing the conclusion gate of π which are not AX-gates: if any such gate has labels in $\{\rightarrow^\circ, \wedge^\bullet, \text{C}, \forall^\bullet X\}$ or it is a non-guarded $\forall^\bullet X$ -gates, then the derivation $\mathcal{D}_\pi$ is obtained by applying the corresponding rule of LJ_2 to the derivations $\mathcal{D}_{\pi'}$ of the sub-nets π' of π obtained by removing the conclusion gate. Otherwise, we can apply Lemma 18 to find a splitting $\rightarrow^\bullet$ or $\wedge^\circ$ -gate, and $\mathcal{D}$ is obtained by applying the corresponding rule of LJ_2 to the derivations $\mathcal{D}_{\pi_1}$ and $\mathcal{D}_{\pi_2}$ of the two sub-nets π_1 and π_2 of π obtained by removing the splitting conclusion gate. ◀

We conclude the section by formulating the equivalence between the definitions of proof net and correct net, consequence of Lemmas 14 and 19.

► **Theorem 20.** *A net is a proof net iff it is correct.*

5 Proof Equivalence

In this section, we discuss the proof equivalence in LJ_2 induced by proof nets.

► **Definition 21.** *We say that two (instances of) rules are **independent** if their set of active and principal formulas are disjoint. Independent rules permutations in LJ_2 are in Figure 6.*

While independent rules permutations are sufficient to get rid of the syntactic bureaucracy in the proof of cut-elimination in LJ_2 , additional rules permutations may be desirable to get a more canonical representation of proofs, and, they naturally arise in denotational models. Here, we consider the additional transformations on derivations in LJ_2 generated by the (non necessarily local) permutations in the bottom of Figure 7.

► **Definition 22.** *We define additional transformations on derivations in LJ_2 which are generated by the (non necessarily local) permutations at the bottom of Figure 6:*

- **weakening-contraction comonoid:** permutations of two nonindependent C-rules, or permutations of nonindependent C- and W-rules ‘cancelling each other’;
- **excising:** the permutation of a $\rightarrow^\bullet$ over a W, permuting a W below a $\rightarrow^\bullet$ by replacing the derivation of the left premise of the $\rightarrow^\bullet$ with a bunch of W;
- **weakening maximization:** permutations pushing the weakening below a $\wedge^\bullet$ or a $\forall^\bullet$;

We define the following equivalence relations on derivations in LJ_2 :

Weakening-contraction comonoid	
$\frac{w \frac{\Gamma, A \vdash C}{\Gamma, A, A \vdash C} \sim \Gamma \vdash A}{c \frac{\Gamma, A \vdash C}{\Gamma, A \vdash C}}$	$\frac{c \frac{\Gamma, A, A \vdash C}{\Gamma, A \vdash C} \sim \Gamma, A, A \vdash C}{w \frac{\Gamma, A \vdash C}{\Gamma, A, A \vdash C}}$
$\frac{c \frac{\Gamma, A^1, A^2, A^3 \vdash C}{\Gamma, A, A^3 \vdash C} \sim c \frac{\Gamma, A^1, A^2, A^3 \vdash C}{\Gamma, A^1, A \vdash C}}{c \frac{\Gamma, A \vdash C}{\Gamma, A \vdash C}}$	

Excising	Weakening maximization
$\frac{\frac{\mathcal{D} \parallel}{\Gamma \vdash A} \quad w \frac{\Delta \vdash C}{B, \Delta \vdash C} \sim w \frac{\Delta \vdash C}{\Gamma, A \rightarrow B, \Delta \vdash C}}{\rightarrow^* \frac{\Gamma, A \rightarrow B, \Delta \vdash C}{\Gamma, A \rightarrow B, \Delta \vdash C}}$	$\frac{w \frac{\Gamma \vdash C}{\Gamma, A, B \vdash C} \sim w \frac{\Gamma \vdash C}{\Gamma, A \wedge B \vdash C}}{\wedge^* \frac{\Gamma, A, B \vdash C}{\Gamma, A \wedge B \vdash C}} \quad \frac{w \frac{\Gamma \vdash C}{\Gamma, A \sigma \vdash C} \sim w \frac{\Gamma \vdash C}{\Gamma, \forall X. A \vdash C}}{\vee^* \frac{\Gamma, \forall X. A \vdash C}{\Gamma, \forall X. A \vdash C}}$

■ **Figure 7** Additional rules permutations in LJ₂.

- 422 ■ $\mathcal{D}_1 \sim \mathcal{D}_2$ if $\mathcal{D}_1$ and $\mathcal{D}_2$ are equivalent up to independent rules permutations and excising;
 423 ■ $\mathcal{D}_1 \approx \mathcal{D}_2$ if $\mathcal{D}_1$ and $\mathcal{D}_2$ are equivalent up to independent rules permutations, excising,
 424 weakening-contraction comonoid, and weakening maximization.

425 ► **Proposition 23.** Let $\mathcal{D}_1$ and $\mathcal{D}_2$ be two derivations. If $\mathcal{D}_1 \sim \mathcal{D}_2$, then $\text{Pn}(\mathcal{D}_1) = \text{Pn}(\mathcal{D}_2)$.

426 **Proof.** It suffices to prove that if $\mathcal{D}_1$ and $\mathcal{D}_2$ differ by a single transformation of the ones
 427 defining $\sim$, then $\text{Pn}(\mathcal{D}_1) = \text{Pn}(\mathcal{D}_2)$. The translation of derivations into proof nets allows us
 428 to easily check that gates corresponding to independent rules are not one above the other,
 429 while the special case for the translation of the $\rightarrow^*$ -rule allows us to capture the excising
 430 transformation. ◀

431 Proposition 23 shows that the proof nets of two derivations are the same if the two
 432 derivations are equivalent up to independent rules permutations, but it does not show that
 433 the converse is true, that is, that if the proof nets of two derivations are the same, then the
 434 two derivations are equivalent up to independent rules permutations. For a counterexample,
 435 consider the two following derivations and their proof net.

$$436 \quad \text{Pn} \left(\frac{\frac{\frac{\text{AX} \frac{}{D \vdash D}}{w \frac{}{B, C, D \vdash D}}}{\wedge^* \frac{}{B \wedge C, D \vdash D}}}{\rightarrow^* \frac{}{A, A \rightarrow (B \wedge C), D \vdash D}} \right) = \begin{array}{c} \boxed{W} \\ \downarrow \\ A^* \end{array} \quad \begin{array}{c} \boxed{W} \\ \downarrow \\ A \rightarrow (B \wedge C)^* \end{array} \quad \begin{array}{c} \boxed{AX} \\ \swarrow \searrow \\ D^* \quad D^\circ \end{array} = \text{Pn} \left(\frac{\frac{\frac{\text{AX} \frac{}{D \vdash D}}{w \frac{}{B \wedge C, B \wedge C, D \vdash D}}}{c \frac{}{B \wedge C, D \vdash D}}}{\rightarrow^* \frac{}{A, A \rightarrow (B \wedge C), D \vdash D}} \right) \quad (3)$$

437 The reason why the converse implication of Proposition 23 does not hold in proof nets, is
 438 because of the excising transformation is local, while the special case of the translation of
 439 the $\rightarrow^*$ -rule has a non-local trigger. In the example above, the two sub-derivations of right
 440 premises of the $\rightarrow^*$ -rules are not $\sim$ -equivalent. To fix this problem, it is sufficient to be able
 441 to permute all W-rule ‘as low as possible’ (see the three leftmost rewritings in Figure 8). This
 442 includes permuting the W-ruled below C-rules, which is part of the weakening-contraction
 443 comonoid, which, therefore, we also aim to capture with our syntax.

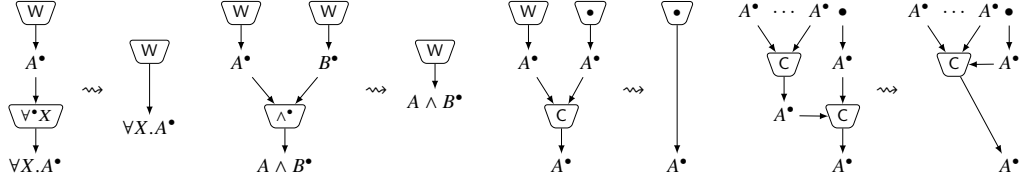
444 5.1 Canonical Proof Nets

445 In order to improve the modularity of proof equivalence for our proof nets, we define a
 446 canonical form for proof nets, which is obtained by pushing all W-gates as low as possible in
 447 the proof net.

448 ► **Definition 24.** A *canonical proof net* is a proof net such that:

- 449 ■ every C-gate has no premise which is the conclusion of another C-gate or a W-gate;

23:14 Proof Nets for Second Order Intuitionistic Logic



■ **Figure 8** Rewriting rules for canonization of proof nets.

450 ■ *there is no gate such that all its premises are conclusions of W-gates.*

451 The **canonization** of a proof net is the rewriting defined by the rules in Figure 8.

452 ► **Proposition 25.** *Canonization is convergent, and normal forms are canonical proof nets.*

453 **Proof.** Termination follows from the fact that each rewriting rule strictly reduces the number
454 of C-, W-, and $\nabla^\bullet$ -gates. Confluence can be shown by a straightforward verification of all the
455 critical pairs: the non-trivial ones are those involving the second and third rewriting rules,
456 which correspond to the comonoidal structure of the contraction and weakening rules, which
457 is clearly confluent (see [33]).

458 Moreover, the rewriting rules in Figure 8 preserve the correctness criterion. This is
459 straightforward to verify since the $\circ$ -root and none of the $\rightarrow^\bullet$ - and $\rightarrow^\circ$ -gates are involved
460 in the rewriting rules, and the effect of the rewriting rules on the test does not change
461 the topology of the context, that is, each rewriting rule preserves the existence of paths
462 connecting vertices in the pre-net on which they are applied. We conclude since normal
463 forms of canonization satisfy the conditions for proof nets being canonical. ◀

464 Note that a direct translation of derivations into canonical proof nets is possible, but it would
465 require to absorb the weakening-contraction comonoid and the weakening maximization
466 transformations into the translation of the $\wedge^\bullet$ -, $\nabla^\bullet$ -, and C-rules, which would make the
467 translation more complex, see Figure 10 in ??.

468 Canonical proof nets allows us to have a canonicity result with respect to the equivalence
469 relation $\approx$ on derivations in LJ_2 .

470 ► **Theorem 26** (Full Local Canonicity). *Let $\mathcal{D}_1$ and $\mathcal{D}_2$ be two derivations in LJ_2 . Then*
471 *$\mathcal{D}_1 \approx \mathcal{D}_2$ iff $\{\mathcal{D}_1\} = \{\mathcal{D}_2\}$.*

472 **Sketch of the proof.** The left-to-right implication is proven similarly to Proposition 23.
473 The weakening-contraction comonoid and the weakening maximization transformations are
474 captured by the rewriting rules in Figure 8.

475 The right-to-left implication, is proven by induction on number of gates in a proof net $\pi =$
476 $\text{Pn}(\mathcal{D})$, showing that if we pick a conclusion gate of π which has label in $\{\wedge^\bullet, \text{C}, \text{W}, \nabla^\circ X, \rightarrow^\circ\}$,
477 is a non-guarded $\nabla^\bullet X$ -gate, or is a splitting $\rightarrow^\bullet$ - or $\wedge^\circ$ -gate, then we can find a derivation $\mathcal{D}'$
478 with bottom-most rule corresponding to the selected conclusion gate and such that $\mathcal{D} \approx \mathcal{D}'$
479 and $\text{Pn}(\mathcal{D}') = \pi^7$. ◀

480 ► **Remark 27.** For the propositional fragment of LJ_2 , the proof equivalence captured by
481 canonical proof nets is the same as the same proof equivalence captured by combinatorial
482 proofs [51], and it is well-known that this latter is weaker than the proof equivalence

⁷ See [53], where the same technique has been used to prove that proof nets for (classical) multiplicative additive linear logic are canonical with respect to rules permutations.

captured by natural deduction (or, equivalently, lambda calculus). In particular, the following equivalence of derivations in LJ₂ is not captured by canonical proof nets, but it is captured by natural deduction.

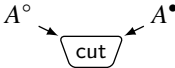
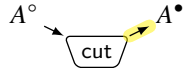
$$\rightarrow \cdot \frac{\frac{\mathcal{D} \parallel}{\Gamma \vdash A} \quad \frac{\mathcal{C} \quad \frac{B, B, \Delta \vdash C}{B, \Delta \vdash C}}{\Gamma, A \rightarrow B, \Delta \vdash C}}{\Gamma, A \rightarrow B, \Delta \vdash C} \sim \rightarrow \cdot \frac{\frac{\mathcal{D} \parallel}{\Gamma \vdash A} \quad \frac{\mathcal{D} \parallel \quad \frac{\Gamma \vdash A \quad B, B, \Delta \vdash C}{\Gamma, A \rightarrow B, B, \Delta \vdash C}}{\Gamma, A \rightarrow B, \Gamma, A \rightarrow B, \Delta \vdash C}}{\mathcal{C} \quad \frac{\Gamma, A \rightarrow B, \Gamma, A \rightarrow B, \Delta \vdash C}{\Gamma, A \rightarrow B, \Delta \vdash C}} \quad (4)$$

Note that this very equivalence is the one causing the unbounded growth of the size of derivations in LJ₂ during cut-elimination.

6 Conclusion

In this paper, we have defined proof nets for the cut-free fragment of second-order intuitionistic logic sufficient to define types for the System F, closing a long trip started by Girard's original motivation for the introduction of linear logic, and proof nets as its canonical syntax for proofs. The direct translation of derivations into proof nets allowed us formulate a very simple correctness criterion for our proof nets, based on paths in a graph obtained by reorienting certain edges of the proof net. We have also shown that our proof nets can be refined to provide canonical representations of proofs in LJ₂ up to a natural notion of proof equivalence, which is weaker than the one captured by natural deduction, but still strong enough to avoid the unbounded growth of the size of derivations during cut-elimination.

Adding the cut-rule. In the paper, we did not include the cut-rule in our syntax to avoid unnecessary mental overhead, and to keep the presentation as simple as possible. However, to include the cut-rule only requires considering a new shape for gates, which behave similarly to a $\rightarrow^\bullet$ -gate with no conclusion. When it comes to the definition of test, correctness criterion, splitting, and sequentialization, the extension of the results in the paper is straightforward.

sequent rule $\text{cut} \frac{\Gamma \vdash A \quad A, \Delta \vdash C}{\Gamma, \Sigma, \Delta \vdash C}$	cut-gate shape 	polarized orientation of cut 
---	---	--

Perspectives. We can establish a correspondence between the set of paths we use to check correctness for our proof nets and winning strategy in a Hughes' hypergames [30, 28], similarly to the correspondence already studied in [35]. Such a correspondence could be used to provide a purely combinatorial (i.e., not relying on categorical constructions) proofs of soundness and completeness for hypergames, and to design *combinatorial proofs* (in the sense of [51]) for the entire System F, which is still an open problem due to the absence of a decomposition theorem usually used to define combinatorial proofs [51, 4, 5, 9, 6]. We also plan to extend the *combinatorial lambda calculus* from [27] for the entire System F using these results, providing a new tool to study the dynamics of normalization of System F in which it is possible to separate the resource management from the linear reductions, similarly to the approach in Dal Lago and Olimpieri's in [11] for multiplicative exponential linear logic.

References

- 1 V. Michele Abrusci and Roberto Maieli. Proof nets for multiplicative cyclic linear logic and lambek calculus. *Mathematical Structures in Computer Science*, 29(6):733–762, 2019. doi:10.1017/S0960129518000300.

- 520 2 Beniamino Accattoli. Proof nets and the call-by-value λ -calculus. *Theoretical Computer Science*,
521 606:2–24, 2015. Logical and Semantic Frameworks with Applications. URL: <https://www.sciencedirect.com/science/article/pii/S0304397515007264>, doi:10.1016/j.tcs.2015.
522 08.006.
- 524 3 Beniamino Accattoli. Proof nets and the linear substitution calculus. In Bernd Fischer and
525 Tarmo Uustalu, editors, *Theoretical Aspects of Computing – ICTAC 2018*, pages 37–61, Cham,
526 2018. Springer International Publishing.
- 527 4 Matteo Acclavio and Lutz Straßburger. From syntactic proofs to combinatorial proofs. In
528 *International Joint Conference on Automated Reasoning*, pages 481–497. Springer, 2018.
- 529 5 Matteo Acclavio and Lutz Straßburger. On combinatorial proofs for modal logic. In Serenella
530 Cerrito and Andrei Popescu, editors, *Automated Reasoning with Analytic Tableaux and Related*
531 *Methods*, pages 223–240, Cham, 2019. Springer International Publishing.
- 532 6 Matteo Acclavio and Lutz Straßburger. Combinatorial proofs for constructive modal logic. In
533 Alessandra Palmigiano David Fernández-Duque and Sophie Pinchinat, editors, *Advances in*
534 *Modal Logic, Volume 14*, pages 15–36. College Publications, 2022. URL: <http://www.aiml.net/volumes/volume14/06-Acclavio-Strassburger.pdf>.
- 536 7 Z.M. Ariola and J.W. Klop. Cyclic lambda graph rewriting. In *Proceedings Ninth Annual*
537 *IEEE Symposium on Logic in Computer Science*, pages 416–425, 1994. doi:10.1109/LICS.
538 1994.316066.
- 539 8 Gianluigi Bellin and Jacques van de Wiele. Subnets of proof-nets in MLL^- . In *Advances in*
540 *Linear Logic*, pages 249–270. Cambridge University Press, 1995.
- 541 9 Ralph Benjamin and Lutz Straßburger. Towards a combinatorial proof theory. In *Tableaux*
542 *2019*. Springer, 2019.
- 543 10 Ugo Dal Lago and Olivier Laurent. Quantitative game semantics for linear logic. In Michael
544 Kaminski and Simone Martini, editors, *Computer Science Logic*, pages 230–245, Berlin,
545 Heidelberg, 2008. Springer Berlin Heidelberg. doi:10.1007/978-3-540-87531-4_18.
- 546 11 Ugo Dal Lago and Federico Olimpieri. Linearization via rewriting (long version), 2025. URL:
547 <https://arxiv.org/abs/2503.04408>, arXiv:2503.04408.
- 548 12 Vincent Danos and Laurent Regnier. The structure of multiplicatives. *Archive for Mathematical*
549 *Logic*, 28(3):181–203, 1989. doi:10.1007/BF01622878.
- 550 13 Abhishek De and Alexis Saurin. Infinets: The parallel syntax for non-wellfounded proof-theory.
551 In Serenella Cerrito and Andrei Popescu, editors, *Automated Reasoning with Analytic Tableaux*
552 *and Related Methods*, pages 297–316, Cham, 2019. Springer International Publishing.
- 553 14 Raffaele Di Donna and Lorenzo Tortora de Falco. On the role of connectivity in linear logic
554 proofs, 2025. URL: <https://arxiv.org/abs/2506.21678>, arXiv:2506.21678.
- 555 15 Herman Geuvers and Iris Loeb. Deduction graphs with universal quantification. *Elec-*
556 *tronic Notes in Theoretical Computer Science*, 203(1):93–108, 2008. Proceedings of the
557 Fourth International Workshop on Computing with Terms and Graphs (TERMGRAPH
558 2007). URL: <https://www.sciencedirect.com/science/article/pii/S1571066108001412>,
559 doi:10.1016/j.entcs.2008.03.036.
- 560 16 Jean-Yves Girard. *Interprétation fonctionnelle et élimination des coupures de l’arithmétique*
561 *d’ordre supérieur*. PhD thesis, University of Paris VII, 1972.
- 562 17 Jean-Yves Girard. The system f of variable types, fifteen years later. *Theoretical Computer*
563 *Science*, 45:159–192, 1986. URL: <https://www.sciencedirect.com/science/article/pii/S0304397586900447>, doi:10.1016/0304-3975(86)90044-7.
- 565 18 Jean-Yves Girard. Linear logic. *Theoretical computer science*, 50(1):1–101, 1987.
- 566 19 Jean-Yves Girard. Quantifiers in linear logic. In *Proceedings of the SILFS conference held in*
567 *Cesena*, volume 41, 1987.
- 568 20 Jean-Yves Girard. Quantifiers in linear logic ii. *Nuovi problemi della logica e della filosofia*
569 *della scienza*, 2(1), 1991.
- 570 21 Stefano Guerrini. A linear algorithm for mll proof net correctness and sequentialization. *The-*
571 *oretical Computer Science*, 412(20):1958–1978, 2011. Girard’s Festschrift. URL: <https://www>.

- sciencedirect.com/science/article/pii/S0304397510007127, doi:10.1016/j.tcs.2010.12.021.
- 22 Stefano Guerrini and Andrea Masini. Parsing mell proof nets. *Theoretical Computer Science*, 254(1):317–335, 2001. URL: <https://www.sciencedirect.com/science/article/pii/S0304397599002996>, doi:10.1016/S0304-3975(99)00299-6.
 - 23 Stefano Guerrini and Andrea Masini. Proof nets for classical logic. *Notre Dame Journal of Formal Logic*, 62(2):303–343, 2021.
 - 24 Willem Heijltjes. Graph rewriting for natural deduction and the proper treatment of variables. *Logic Group Preprint Series*, 254, 2007.
 - 25 Willem Heijltjes and Robin Houston. No proof nets for mll with units: proof equivalence in mll is pspace-complete. In *Proceedings of the Joint Meeting of the Twenty-Third EACSL Annual Conference on Computer Science Logic (CSL) and the Twenty-Ninth Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, CSL-LICS '14, New York, NY, USA, 2014. Association for Computing Machinery. doi:10.1145/2603088.2603126.
 - 26 Willem B. Heijltjes, Dominic J. D. Hughes, and Lutz Straßburger. Proof Nets for First-Order Additive Linear Logic. In Herman Geuvers, editor, *4th International Conference on Formal Structures for Computation and Deduction (FSCD 2019)*, volume 131 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 22:1–22:22, Dagstuhl, Germany, 2019. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. URL: <https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.FSCD.2019.22>, doi:10.4230/LIPIcs.FSCD.2019.22.
 - 27 Willem B. Heijltjes, Dominic J. D. Hughes, and Lutz Straßburger. Normalization Without Syntax. In Amy P. Felty, editor, *7th International Conference on Formal Structures for Computation and Deduction (FSCD 2022)*, volume 228 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 19:1–19:19, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. URL: <https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.FSCD.2022.19>, doi:10.4230/LIPIcs.FSCD.2022.19.
 - 28 D. Hughes. Games and definability for system f. In *Logic in Computer Science, Symposium on*, page 76, Los Alamitos, CA, USA, jul 1997. IEEE Computer Society. URL: <https://doi.ieeecomputersociety.org/10.1109/LICS.1997.614935>, doi:10.1109/LICS.1997.614935.
 - 29 Dominic Hughes and Willem Heijltjes. Conflict nets: Efficient locally canonical mall proof nets. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science, LICS '16*, pages 437–446, New York, NY, USA, 2016. Association for Computing Machinery. doi:10.1145/2933575.2934559.
 - 30 Dominic J. D. Hughes. *Hypergame Semantics: Full Completeness for System F*. PhD thesis, University of Oxford, 1999. URL: <http://boole.stanford.edu/~dominic/papers/thesis/hughesthesis.pdf>.
 - 31 Dominic J. D. Hughes and Rob J. Van Glabbeek. Proof nets for unit-free multiplicative-additive linear logic (extended abstract). In *Proceedings of the 18th Annual IEEE Symposium on Logic in Computer Science, LICS '03*, page 1, USA, 2003. IEEE Computer Society.
 - 32 Dominic J.D. Hughes. Unification nets: canonical proof net quantifiers. In *LICS 2018*, 2018.
 - 33 Yves Lafont. The finite model property for various fragments of linear logic. *Journal of Symbolic Logic*, 62(4):1202–1208, 1997. doi:10.2307/2275637.
 - 34 François Lamarche. Proof Nets for Intuitionistic Linear Logic: Essential Nets, 2008. URL: <https://hal.inria.fr/inria-00347336>.
 - 35 François Lamarche. From proof nets to games: extended abstract. *Electronic Notes in Theoretical Computer Science*, 3:107–119, 1996. Linear Logic 96 Tokyo Meeting. URL: <https://www.sciencedirect.com/science/article/pii/S1571066105804098>, doi:10.1016/S1571-0661(05)80409-8.
 - 36 Olivier Laurent. *Étude de la polarisation en logique*. Theses, Université de la Méditerranée - Aix-Marseille II, March 2002. URL: <https://theses.hal.science/tel-00007884>.

- 622 37 Olivier Laurent. Polarized proof-nets and $\lambda\mu$ -calculus. *Theoretical Computer Science*, 290(1):161–188, 2003. URL: <https://www.sciencedirect.com/science/article/pii/S0304397501002973>, doi:10.1016/S0304-3975(01)00297-3.
- 623
- 624 38 P. Malacaria and L. Regnier. Some results on the interpretation of λ -calculus in operator algebras. In *[1991] Proceedings Sixth Annual IEEE Symposium on Logic in Computer Science*, pages 63–72, 1991. doi:10.1109/LICS.1991.151631.
- 625
- 626
- 627 39 Richard Moot and Quintijn Puite. Proof nets for the multimodal lambek calculus. *Studia Logica*, 71(3):415–442, 2002. doi:10.1023/A:1020525032763.
- 628
- 629 40 Michele Pagani and Lorenzo Tortora de Falco. Strong normalization property for second order linear logic. *Theoretical Computer Science*, 411(2):410–444, 2010. URL: <https://www.sciencedirect.com/science/article/pii/S0304397509005180>, doi:10.1016/j.tcs.2009.07.053.
- 630
- 631
- 632
- 633 41 Paolo Pistone. Proof nets, coends and the yoneda isomorphism. *Electronic Proceedings in Theoretical Computer Science*, 292:148–167, April 2019. URL: <http://dx.doi.org/10.4204/EPTCS.292.9>, doi:10.4204/eptcs.292.9.
- 634
- 635 42 Adrien Ragot, Thomas Seiller, and Lorenzo Tortora de Falco. Correctness Criteria for Second Order Multiplicative Linear Logic. working paper or preprint, February 2025. URL: <https://hal.science/hal-04947011>.
- 636
- 637 43 Christian Retoré. *Réseaux et Séquents Ordonnés*. PhD thesis, Univ. Paris VII, 1993.
- 638
- 639 44 Christian Retoré. *Pomset Logic*, pages 299–345. Springer International Publishing, Cham, 2021. doi:10.1007/978-3-030-66545-6_9.
- 640
- 641 45 Christian Retoré. Perfect matchings and series-parallel graphs: multiplicatives proof nets as R&B-graphs: [extended abstract]. *Electronic Notes in Theoretical Computer Science*, 3:167–182, 1996. Linear Logic 96 Tokyo Meeting. URL: <https://www.sciencedirect.com/science/article/pii/S1571066105804165>, doi:10.1016/S1571-0661(05)80416-5.
- 642
- 643 46 Christian Retoré. Handsome proof-nets: perfect matchings and cographs. *Theoretical Computer Science*, 294(3):473–488, 2003. Linear Logic. URL: <https://www.sciencedirect.com/science/article/pii/S030439750100175X>, doi:10.1016/S0304-3975(01)00175-X.
- 644
- 645
- 646 47 John C. Reynolds. Towards a theory of type structure. In *Symposium on Programming*, 1974. URL: <https://api.semanticscholar.org/CorpusID:30450751>.
- 647
- 648 48 DIRK ROORDA. Proof nets for lambek calculus. *Journal of Logic and Computation*, 2(2):211–231, 05 1992. arXiv:<https://academic.oup.com/logcom/article-pdf/2/2/211/2745781/2-2-211.pdf>, doi:10.1093/logcom/2.2.211.
- 649
- 650 49 Mehrnoosh Sadrzadeh and Lutz Straßburger. Lambek calculus with banged atoms for parasitic gaps. In George Metcalfe, Thomas Studer, and Ruy de Queiroz, editors, *Logic, Language, Information, and Computation*, pages 193–209, Cham, 2024. Springer Nature Switzerland.
- 651
- 652 50 Lutz Strassburger. Deep inference and expansion trees for second-order multiplicative linear logic. *Mathematical Structures in Computer Science*, 29(8):1030–1060, 2019. doi:10.1017/S0960129518000385.
- 653
- 654
- 655 51 Lutz Straßburger, Willem Heijltjes, and Dominic J D Hughes. Intuitionistic proofs without syntax. In *LICS 2019 - 34th Annual ACM/IEEE Symposium on Logic in Computer Science*, pages 1–13, Vancouver, Canada, June 2019. IEEE. URL: <https://hal.inria.fr/hal-02386878>, doi:10.1109/LICS.2019.8785827.
- 656
- 657 52 Lutz Straßburger. Deep inference and expansion trees for second-order multiplicative linear logic. *Mathematical Structures in Computer Science*, 29(8):1030–1060, 2019. doi:10.1017/S0960129518000385.
- 658
- 659
- 660 53 Rob van Glabbeek and Dominic Hughes. MALL proof nets identify proofs modulo rule commutation, 2016. URL: <https://arxiv.org/abs/1609.04693>, arXiv:1609.04693.
- 661
- 662 54 PL WADSWORTH. Semantics and pragmatics of the lambda calculus. *PhD thesis, Oxford Univ.*, 1971.
- 663
- 664
- 665
- 666
- 667

- 704 ■ Otherwise, $B^\bullet$ is not connected to the $\circ$ -root of the proof net π_2 . In this case, by
 705 inductive hypothesis, in the test of the proof net translating the derivation $\mathcal{D}_2$ of the
 706 right premise, all maximal paths starting from $B^\bullet$ must end in a W -gate, and they
 707 cannot contain any $\rightarrow^\bullet$ -gate because of the excising condition (4). Then, $\text{Pn}(\mathcal{D}_2)$ is
 708 correct iff also the proof net π' obtained by removing any vertex reachable from $B^\bullet$
 709 in the test of $\text{Pn}(\mathcal{D}_2)$ is correct. We conclude since adding any number of W -gates
 710 together with their conclusions (and the edges connecting them) to π' also preserves
 711 correctness;
- 712 ■ the translation of $\forall^\circ$ adds a conclusion gate (and its conclusion). Acyclicity, functionality,
 713 excising, and one-sidedness follow immediately from the inductive hypothesis on the
 714 proof net of the premise, because the new conclusion gate is connected to the previous
 715 $\circ$ -root of the proof net of the premise. By the assumption of freshness of the variable Y
 716 in the $\forall^\circ Y$ -rule, cleanness follows immediately from the inductive hypothesis on the proof
 717 net of the premise, because any leaf becoming an eigenleaf of the new $\forall^\circ Y$ -gate must be
 718 linked to a leaf which must be in the reach of a $\forall^\bullet Y$ -gate. The deadlock-free condition
 719 then follows immediately from the fact that any path to the $\circ$ -root of the proof net must
 720 contain the new $\forall^\circ Y$ -gate;
- 721 ■ the translation of $\forall^\bullet$ adds a conclusion gate (and its conclusion) and a pointer from the
 722 $\forall^\bullet X$ -gate to each vertex corresponding to the formula instantiated by the substitution of
 723 an occurrence of the variable X bound by the quantifier. By construction, the target of
 724 each pointer is labeled by the same formula (modulo polarities), therefore uniformity is
 725 preserved. The cleanness and safeness follow by inductive hypothesis and by definition
 726 of substitution, which avoids capture of free variables. Then, correctness follows from
 727 inductive hypothesis on the proof net of the premise. In particular, the cleanness condition
 728 of the proof net of the premise ensures that no leaf in the reach of the $\forall^\bullet X$ -gate is linked to
 729 an eigenleaf of a $\forall^\circ Y$ -gate, and therefore the deadlock-free condition is trivially satisfied.
 730 ◀

731 ► **Lemma 17.** *Let $\mathcal{G}$ be a correct net with $\circ$ -root r and z° a vertex of $\mathcal{G}$. Then:*

- 732 1. *the relation $\preceq$ is an order inducing a tree-like structure on the vertices of $\mathcal{G}$;*
- 733 2. *r is the $\preceq$ -maximum, and every $\rightarrow^\bullet$ -gate is $\preceq$ -smaller than r ;*
- 734 3. *if $x \in \pm(z)$, then $x^\bullet$. Moreover, if x is a premise of a gate $\mathfrak{g}$, then $z \neq \mathfrak{g}(\mathfrak{g})$;*
- 735 4. *if $\mathfrak{g}$ is a $\rightarrow^\circ$ -gate with premises x° and $y^\bullet$, then $\mathfrak{g}(x) \preceq \mathfrak{g}(y)$. If y is the premise of the*
 736 *$\preceq$ -greatest $\forall^\circ$ -gate guarding a $\forall^\bullet$ -gate $\mathfrak{g}$, then $\mathfrak{g}(\mathfrak{g}) \preceq y$.*
- 737 5. *if x is a premise of a gate with conclusion y , then $\mathfrak{g}(x) \preceq \mathfrak{g}(y)$. Moreover, if additionally*
 738 *$x^\bullet$ and $\mathfrak{g}(x) = r^\circ$, then $\mathfrak{g}(y) = r^\circ$ and $y^\bullet$.*

739 **Proof.** We prove each item separately.

- 740 1. Reflexivity and transitivity of $\preceq$ are consequence of the definition, while anti-symmetry
 741 follows from acyclicity of the test (2). Moreover, if $x \preceq y_1, y_2$, then either $y_1 \preceq y_2$ or
 742 $y_2 \preceq y_1$ must hold by acyclicity of the test.
- 743 2. The $\circ$ -root is the $\preceq$ -maximum because of one-sidedness and excising guarantees that every
 744 $\rightarrow^\bullet$ -gate is $\preceq$ -smaller than the $\circ$ -root.
- 745 3. The polarized orientation of the test guarantees that every advisor is a $\bullet$ -vertex. If an
 746 advisor x of z is a premise of a gate $\mathfrak{g}$, then the gate is not in the empire of z (otherwise
 747 x would not be an advisor of z); therefore, $\mathfrak{g}(\mathfrak{g}) \neq z$.
- 748 4. The first follows from functionality, the second from deadlock-free.
- 749 5. By definition of the polarized orientation of the test and functionality. If, additionally, $x^\bullet$
 750 and $\mathfrak{g}(x) = r^\circ$, the only possibility for y to not be $y^\bullet$ is that the gate is a $\wedge^\circ$ -gate, but

in this case, by the previous point, $\mathring{\forall}(x)$ should be above the right-premise of the gate (contradicting the assumption that $\mathring{\forall}(x) = r^\circ$).

► **Lemma 19** (Sequentialization). *Let π be a correct net for a sequent $\Gamma \vdash C$. Then, there is a derivation $\mathcal{D}_\pi$ in LJ_2 with conclusion $\Gamma \vdash C$.*

Proof. We provide a *sequentialization* procedure to construct a derivation $\mathcal{D}_\pi$ in LJ_2 from a correct net π . It proceeds by induction on the number $|\pi|$ of gates of π .

If $|\pi| = 1$, then π can only be made of a AX-gate and its conclusions, then $\mathcal{D}_\pi$ is made of a single AX. Otherwise, π has at least a conclusion gate which is not an AX-gate, and we have the following cases:

- if one of the conclusion gates has label in $\{\rightarrow^\circ, \wedge^\bullet, W, \forall^\circ X\}$ (resp. is a C-gate with $n + 1$ premises), then the net π' obtained by removing that gate and its conclusion from π is still correct. Therefore, we can apply inductive hypothesis to π' to obtain a derivation $\mathcal{D}_{\pi'}$ in LJ_2 , and define the derivation $\mathcal{D}_\pi$ as the one obtained by applying the corresponding rule (resp. the C-rule n times) to the conclusion of $\mathcal{D}_{\pi'}$. In particular:
 - the functionality condition ensures that removing a $\rightarrow^\circ$ -gate does not break functionality nor excising, because any path from the left premise of the removed $\rightarrow^\circ$ -gate to the $\circ$ -root must contain its right premise, and therefore it is still present in the test of π' ;
 - the cleanness condition ensures that every time a $\forall^\circ X$ -gate is a conclusion gate, then none of its eigenleaves are linked to a leaf which is not in the reach of a $\forall^\bullet X$ -gate, therefore that the side condition of the $\forall^\circ X$ -rule is respected;
- if a conclusion gate is an unguarded $\forall^\bullet X$ -gate, then the net π' obtained by removing that gate, its conclusion, as well as the pointers with source that gate from π is still correct. In particular, the conditions on pointers in the definition of net (Definition 5) are trivially preserved by removing the pointers with source the removed $\forall^\bullet X$ -gate. Therefore, we can apply inductive hypothesis to π' to obtain a derivation $\mathcal{D}_{\pi'}$ in LJ_2 , and define the derivation $\mathcal{D}_\pi$ as the one obtained by applying the $\forall^\bullet X$ -rule to the conclusion of $\mathcal{D}_{\pi'}$, where the substitution is defined by the labels of the targets of the pointers with source the removed $\forall^\bullet X$ -gate in π .
- otherwise, the underlying pre-net of π satisfies the hypothesis of Lemma 18, therefore there is a splitting $\wedge^\circ$ or $\rightarrow^\bullet$ gate among the conclusion gates of π . If π_1 and π_2 are the two nets obtained by removing that gate and its conclusion from π , then we can apply inductive hypothesis to both π_1 and π_2 to obtain derivations $\mathcal{D}_{\pi_1}$ and $\mathcal{D}_{\pi_2}$ in LJ_2 , and define the derivation $\mathcal{D}_\pi$ as the one obtained by applying the corresponding rule to the conclusions of these derivations.

► **Theorem 26** (Full Local Canonicity). *Let $\mathcal{D}_1$ and $\mathcal{D}_2$ be two derivations in LJ_2 . Then $\mathcal{D}_1 \approx \mathcal{D}_2$ iff $\{\mathcal{D}_1\} = \{\mathcal{D}_2\}$.*

Proof. The left-to-right implication is proven similarly to Proposition 23. The weakening-contraction comonoid and the weakening maximization transformations are captured by the rewriting rules in Figure 8.

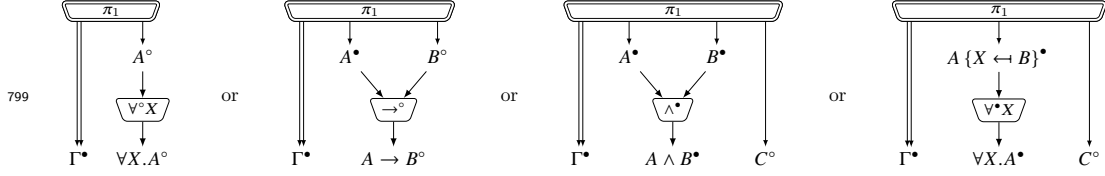
The right-to-left implication, is proven as in by induction on number of gates in a proof net $\pi = \text{Pn}(\mathcal{D})$, showing that if we pick a conclusion gate of π which has label in $\{\wedge^\bullet, C, W, \forall^\circ X, \rightarrow^\circ\}$, is a non-guarded $\forall^\bullet X$ -gate, or is a splitting $\rightarrow^\bullet$ - or $\wedge^\circ$ -gate, then we can

23:22 Proof Nets for Second Order Intuitionistic Logic

find a derivation $\mathcal{D}'$ with bottom-most rule corresponding to the selected conclusion gate and such that $\mathcal{D} \approx \mathcal{D}'$ and $\text{Pn}(\mathcal{D}') = \pi^8$.

We reason by cases on the label of a conclusion gate of π :

1. If π has one of the following forms



800 then the derivation $\mathcal{D}$ has the respective following shape

801

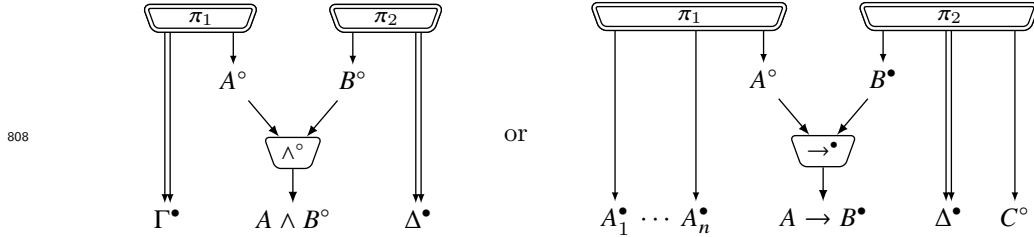
$$\begin{array}{c}
 \frac{\mathcal{D}'_{\text{pre}} \parallel \Gamma' \vdash A}{\Gamma' \vdash \forall X.A} \quad \forall^o \quad \frac{\mathcal{D}_{\text{post}} \parallel}{\Gamma \vdash \forall X.A} \\
 \frac{\mathcal{D}'_{\text{pre}} \parallel \Gamma', A \vdash B}{\Gamma' \vdash A \rightarrow B} \quad \rightarrow^o \quad \frac{\mathcal{D}_{\text{post}} \parallel}{\Gamma \vdash A \rightarrow B} \\
 \frac{\mathcal{D}'_{\text{pre}} \parallel \Gamma', A, B \vdash C'}{\Gamma', A \wedge B \vdash C'} \quad \wedge^o \quad \frac{\mathcal{D}_{\text{post}} \parallel}{\Gamma, A \wedge B \vdash C} \\
 \frac{\mathcal{D}'_{\text{pre}} \parallel \Gamma', \forall X.A \vdash C'}{\Gamma', \forall X.A \vdash C'} \quad \forall^o X \quad \frac{\mathcal{D}_{\text{post}} \parallel}{\Gamma, \forall X.A \vdash C}
 \end{array}$$

802 with none of the rules in $\mathcal{D}_{\text{post}}$ having the occurrence of A in the conclusion of the
 803 r -rule we are focussing on as active formula. We can then permute down r using rules
 804 permutations, obtaining a derivation $\mathcal{D}' \sim \mathcal{D}$ of the corresponding following form, as
 805 requested in the statement of the theorem:

806

$$\begin{array}{c}
 \frac{\mathcal{D}_{\text{pre}} \parallel \Gamma' \vdash A}{\Gamma \vdash \forall X.A} \quad \forall^o \quad \frac{\mathcal{D}_{\text{pre}} \parallel \Gamma', A \vdash B}{\Gamma \vdash A \rightarrow B} \quad \rightarrow^o \\
 \frac{\mathcal{D}_{\text{pre}} \parallel \Gamma', A, B \vdash C'}{\Gamma, A \wedge B \vdash C} \quad \wedge^o \quad \frac{\mathcal{D}_{\text{pre}} \parallel \Gamma', \forall X.A \vdash C'}{\Gamma, \forall X.A \vdash C} \quad \forall^o X
 \end{array}$$

807 2. If π has one of the following forms



809 then the derivation $\mathcal{D}$ has the respective following shape

810

$$\begin{array}{c}
 \frac{\mathcal{D}'_{\text{L-pre}} \parallel \Gamma' \vdash A \quad \mathcal{D}'_{\text{R-pre}} \parallel \Delta' \vdash B}{\Gamma', \Delta' \vdash A \wedge B} \quad \wedge^o \quad \frac{\mathcal{D}_{\text{post}} \parallel}{\Gamma, \Delta \vdash A \wedge B} \\
 \frac{\mathcal{D}'_{\text{L-pre}} \parallel \Gamma' \vdash A \quad \mathcal{D}'_{\text{R-pre}} \parallel B, \Delta' \vdash C'}{\Gamma', A \rightarrow B, \Delta' \vdash C'} \quad \rightarrow^o \quad \frac{\mathcal{D}_{\text{post}} \parallel}{\Gamma, A \rightarrow B, \Delta \vdash C}
 \end{array}$$

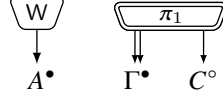
811 with none of the rules in $\mathcal{D}_{\text{post}}$ having the occurrence of A in the conclusion of the
 812 r -rule we are focussing on as active formula. We can then permute down r using rules

⁸ See [53], where the same technique has been used to prove that proof nets for (classical) multiplicative additive linear logic are canonical with respect to rules permutations.

permutations, obtaining a derivation $\mathcal{D}' \sim \mathcal{D}$ of the corresponding following form, as requested in the statement of the theorem:

$$\frac{\mathcal{D}_{L\text{-pre}} \parallel \quad \mathcal{D}_{R\text{-pre}} \parallel}{\wedge^\circ \frac{\Gamma \vdash A \quad \Delta \vdash B}{\Gamma, \Delta \vdash A \wedge B}} \quad \frac{\mathcal{D}_{L\text{-pre}} \parallel \quad \mathcal{D}_{R\text{-pre}} \parallel}{\rightarrow^\bullet \frac{\Gamma \vdash A \quad B, \Delta \vdash C}{\Gamma, A \rightarrow B, \Delta \vdash C}}$$

3. If π has the following form,



then we reason on the bottom-most rule r in $\mathcal{D}$ whose principal formula A corresponds to the conclusion of the W -gate of π :

- a. if $r = W$ -rule, then we proceed as in (1);
- b. if $r = \wedge^\bullet$ (resp. $r = \vee^\bullet$), then we have that $\mathcal{D}$ is of the following shape

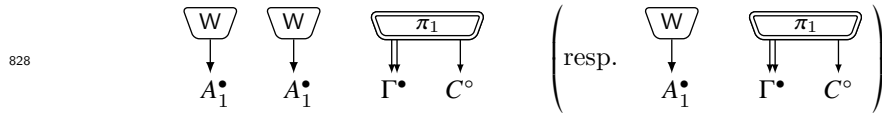
$$\wedge^\bullet \frac{\mathcal{D}'_{\text{pre}} \parallel \quad A_1, A_2, \Gamma' \vdash C'}{A_1, A_2, \Gamma' \vdash C'} \quad \text{if } r = \wedge^\bullet \quad \left(\text{resp. } \vee^\bullet \frac{\mathcal{D}'_{\text{pre}} \parallel \quad A_1, \Gamma' \vdash C'}{A_1, \Gamma' \vdash C'} \quad \text{if } r = \vee^\bullet \right)$$

$$\frac{\mathcal{D}_{\text{post}} \parallel}{A, \Gamma \vdash C}$$

with none of the rules in $\mathcal{D}_{\text{post}}$ having the occurrence of A in the conclusion of the W -rule we are focussing on as active formula. In this case, we can permute down r using rules permutations, obtaining a derivation $\mathcal{D}' \sim \mathcal{D}$ of the following form

$$\wedge^\bullet \frac{\mathcal{D}_{\text{pre}} \parallel \quad A_1, A_2, \Gamma \vdash C}{A, \Gamma \vdash C} \quad \text{if } r = \wedge^\bullet \quad \left(\text{resp. } \vee^\bullet \frac{\mathcal{D}_{\text{pre}} \parallel \quad A_1, \Gamma' \vdash C'}{A_1, \Gamma' \vdash C'} \quad \text{if } r = \vee^\bullet \right)$$

where the canonical proof net of $\mathcal{D}_{\text{pre}}$ is the following



We conclude by induction hypothesis, since $\mathcal{D}_{\text{pre}}$ has strictly fewer gates than $\mathcal{D}$.

- c. Otherwise, it must be the case that there is a *excising* $\rightarrow^\bullet$ -rule in $\mathcal{D}$, that is, a $\rightarrow^\bullet$ -rule whose active formula in the right premise is the conclusion of a W -rule. Thus, $\mathcal{D}$ has the following shape

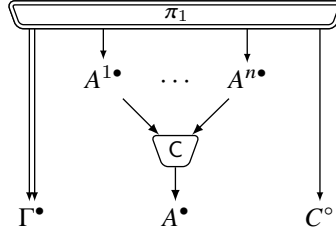
$$\frac{\mathcal{D}'_{L\text{-pre}} \parallel \quad \frac{\mathcal{D}'_{R\text{-pre}} \parallel \quad \Sigma \vdash D}{W \frac{\Sigma \vdash D}{C, \Sigma \vdash D}}}{\rightarrow^\bullet \frac{A, \Delta \vdash B \quad C, \Sigma \vdash D}{A, \Delta, B \rightarrow C, \Sigma \vdash D}} \quad \frac{\mathcal{D}_{\text{post}} \parallel}{A, \Gamma \vdash D}$$

where no rule in $\mathcal{D}_{\text{post}}$ has active formula A , otherwise we could assume that such rule could be a $\wedge^\bullet$ or a $\vee^\bullet$, and we would be in (3.3b). In this case, we can apply excising

to $\mathcal{D}$, obtaining a derivation $\mathcal{D}'' \approx \mathcal{D}$ as shown below on the left, and thus we can permute down the W -rule in $\mathcal{D}''$ as done in (3.3a) obtaining a derivation $\mathcal{D}' \sim \mathcal{D}''$ and conclude.

$$\mathcal{D}'' = \frac{\frac{\mathcal{D}'_{\text{R-pre}} \parallel}{\Sigma \vdash D} \quad \frac{\mathcal{D}_{\text{post}} \parallel}{A, \Gamma \vdash D}}{A, \Delta, B \rightarrow C, \Sigma \vdash D} W \quad \mathcal{D}' = \frac{\mathcal{D}_{\text{pre}} \parallel}{\Gamma \vdash D} W \frac{}{A, \Gamma \vdash D}$$

4. If π has the following form,



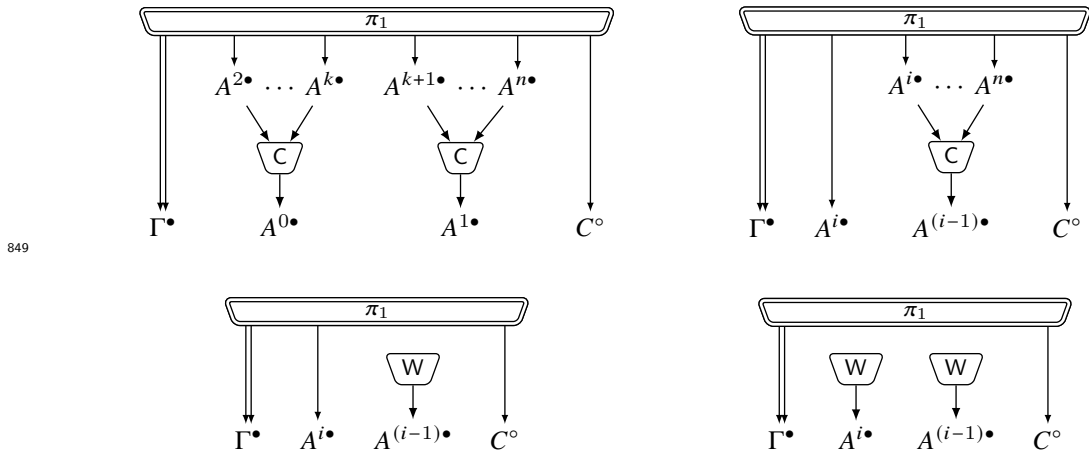
then the derivation $\mathcal{D}$ has the following shape

$$\frac{\frac{\mathcal{D}'_{\text{pre}} \parallel}{\Gamma', A^0, A^1 \vdash C'} \quad \frac{\mathcal{D}_{\text{post}} \parallel}{\Gamma, A \vdash C}}{C} C \frac{}{\Gamma', A \vdash C'}$$

with none of the rules in $\mathcal{D}_{\text{post}}$ having the occurrence of A in the conclusion of the C -rule we are focussing on as active formula. In this case, we can permute down the C -rule using rules permutations, obtaining a derivation $\mathcal{D}' \sim \mathcal{D}$ of the following form.

$$\mathcal{D}' = \frac{\mathcal{D}_{\text{pre}} \parallel}{\Gamma, A^0, A^1 \vdash C} C \frac{}{\Gamma, A \vdash C}$$

such that $\{\mathcal{D}_{\text{pre}}\}$ has one of the following forms



To conclude, it suffices to remark that the canonization of the proof net obtained by adding a C -gate with premises $A^{(i-1)\bullet}$ and $A^{(i)\bullet}$ to $\{\mathcal{D}_{\text{pre}}\}$ is the initial proof net π .

▶

853

854



857

861

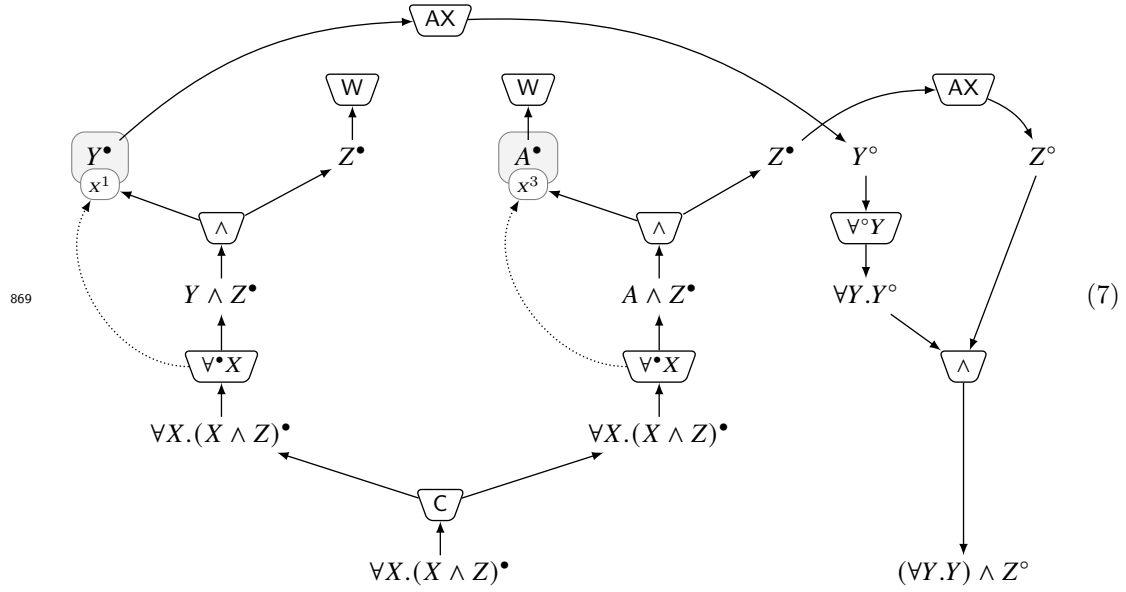
862

866

867

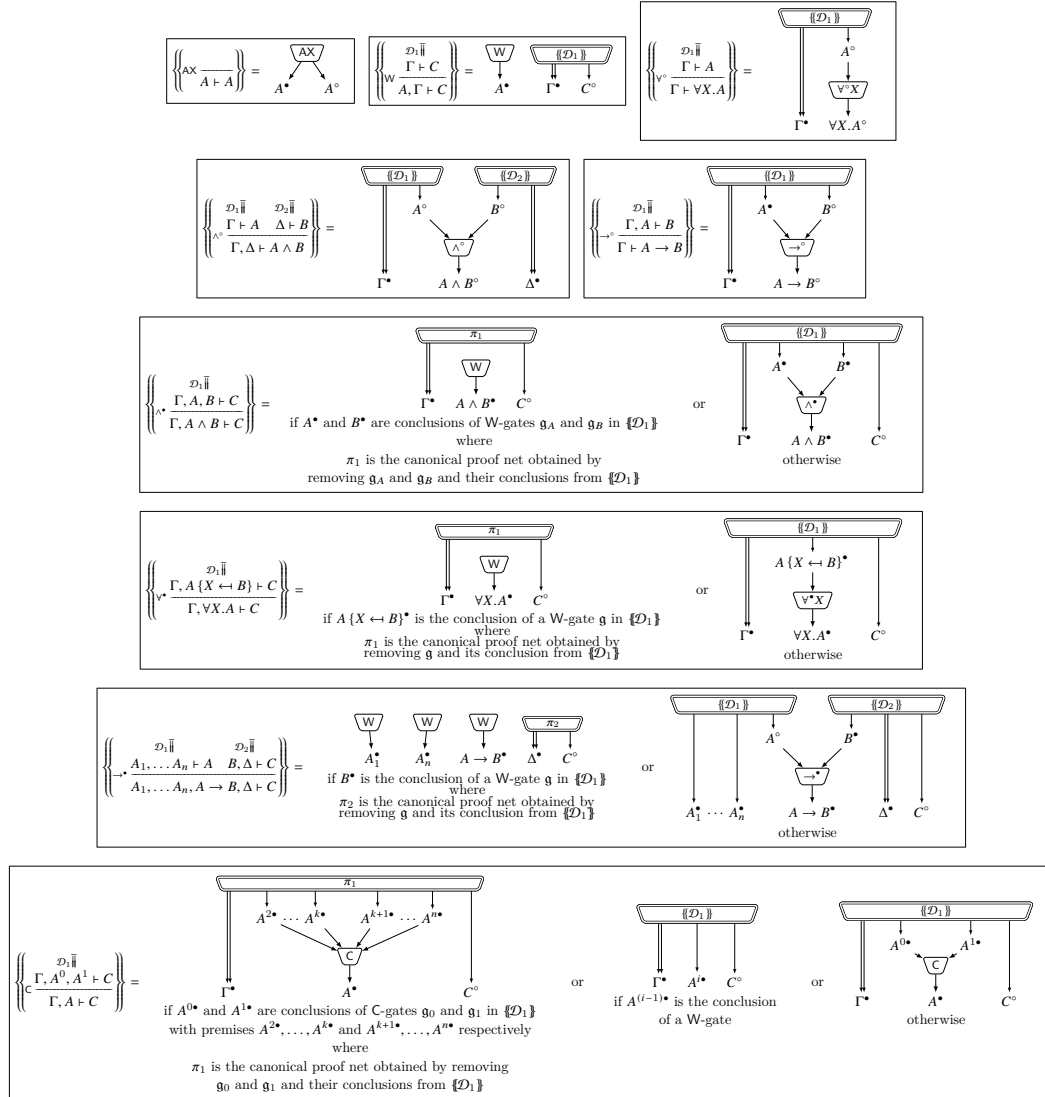
23:26 Proof Nets for Second Order Intuitionistic Logic

net for the same sequent is correct.



The way proof search in this case can succeed is due to the fact that the C allows us to create two copies of the formula $\forall X.(X \wedge Z)$. Then, we can now safely split the proof search context in two branches, one for each copy of $\forall X.(X \wedge Z)$, and then apply the $\forall^\circ X$ -rule in the branch containing $\forall Y.Y$ before applying the $\forall^\bullet X$ -rule, therefore now respecting the side condition of the $\forall^\circ X$ -rule.

$$\begin{array}{c}
 \text{875} \\
 \frac{\frac{\frac{\frac{\text{AX} \text{ --- } Y \vdash Y}{\text{W} \text{ --- } Y, Z \vdash Y}}{\wedge^\bullet \text{ --- } Y \wedge Z \vdash Y}}{\forall^\bullet \text{ --- } \forall X.(X \wedge Z) \vdash Y}}{\forall^\circ \text{ --- } \forall X.(X \wedge Z) \vdash \forall Y.Y} \\
 \frac{\frac{\frac{\frac{\text{AX} \text{ --- } Z \vdash Z}{\text{W} \text{ --- } A, Z \vdash Z}}{\wedge^\bullet \text{ --- } A \wedge Z \vdash Z}}{\forall^\bullet \text{ --- } \forall X.(X \wedge Z) \vdash Z}}{\forall^\circ \text{ --- } \forall X.(X \wedge Z) \vdash Z} \\
 \frac{\forall X.(X \wedge Z) \vdash \forall Y.Y \quad \forall X.(X \wedge Z) \vdash Z}{\wedge^\circ \text{ --- } \forall X.(X \wedge Z), \forall X.(X \wedge Z) \vdash (\forall Y.Y) \wedge Z} \\
 \frac{\quad}{\text{C} \text{ --- } \forall X.(X \wedge Z) \vdash (\forall Y.Y) \wedge Z}
 \end{array}$$



■ **Figure 10** Direct translation of LJ₂ derivations to canonical proof nets. Cases for the translation of the C-rule where only one of the two A^{i° is conclusion of a C-gate are omitted, but they are similar to the leftmost case.