



**HAL**  
open science

# Infinets: The parallel syntax for non-wellfounded proof-theory

Abhishek De, Alexis Saurin

► **To cite this version:**

Abhishek De, Alexis Saurin. Infinets: The parallel syntax for non-wellfounded proof-theory. 2021.  
hal-02166199v2

**HAL Id: hal-02166199**

**<https://hal.science/hal-02166199v2>**

Preprint submitted on 23 Sep 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Infinets: The parallel syntax for non-wellfounded proof-theory

Abhishek De<sup>1</sup> and Alexis Saurin<sup>2</sup> \*

<sup>1</sup> IRIF, Université de Paris, France

<sup>2</sup> IRIF, CNRS, Université de Paris, France

**Abstract.** Logics based on the  $\mu$ -calculus are used to model inductive and coinductive reasoning and to verify reactive systems. A well-structured proof-theory is needed in order to apply such logics to the study of programming languages with (co)inductive data types and automated (co)inductive theorem proving. While traditional proof system suffers some defects, non-wellfounded (or infinitary) and circular proofs have been recognized as a valuable alternative, and significant progress have been made in this direction in recent years. Such proofs are non-wellfounded sequent derivations together with a global validity condition expressed in terms of *progressing threads*.

The present paper investigates a discrepancy found in such proof systems, between the sequential nature of sequent proofs and the parallel structure of threads: various proof attempts may have the exact threading structure while differing in the order of inference rules applications. The paper introduces infinets, that are proof-nets for non-wellfounded proofs in the setting of multiplicative linear logic with least and greatest fixed-points ( $\mu\text{MLL}^\infty$ ) and study their correctness and sequentialization.

**Keywords:** circular proofs · non-wellfounded proofs · fixed points · mu-calculus · linear logic · proof-nets · induction and coinduction

Inductive and coinductive reasoning is pervasive in computer science to specify and reason about infinite data as well as reactive properties. Developing appropriate proof systems amenable to automated reasoning over (co)inductive statements is therefore important for designing programs as well as for analyzing computational systems. Various logical settings have been introduced to reason about such inductive and coinductive statements, both at the level of the logical languages modelling (co)induction (such as Martin L of’s inductive predicates or fixed-point logics, also known as  $\mu$ -calculi) and at the level of the proof-theoretical framework considered (finite proofs with explicit (co)induction rules *à la* Park [23] or infinite, non-wellfounded proofs with fixed-point unfoldings) [6–8, 4, 1, 2]. Moreover, such proof systems have been considered over classical logic [6, 8], intuitionistic logic [9], linear-time or branching-time temporal logic [19, 18, 25, 26, 13–15] or linear logic [24, 16, 4, 3, 14].

---

\* This project has received funding from the European Union’s Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 754362. Partially funded by ANR Project RAPIDO, ANR-14-CE25-0007.

Logics based on the  $\mu$ -calculus have been particularly successful in modelling inductive and coinductive reasoning and for the verification of reactive systems. While the model-theory of the  $\mu$ -calculus has been well-studied, its proof-theory still deserves further investigations. Indeed, while explicit induction rules are simple to formulate (For instance, figure 1 shows the introduction rule *à la* Park for a coinductive property) the treatment of (co)inductive reasoning brings some highly complex proof objects.

At least two fundamental technical shortcomings prevent the application of traditional  $\mu$ -calculus-based proof-systems for the study of programming languages with (co)inductive data types and automated (co)inductive theorem proving and call for alternative proposals of proof systems supporting (co)induction. Firstly, the fixed point introduction rules break the subformula property which is highly problematic for automated proof construction: at each coinduction rule, one shall guess an invariant (in the same way as one has to guess an appropriate induction hypothesis in usual mathematical reasoning). Secondly,  $(\nu_{\text{inv}})$  actually hides a cut rule that *cannot* be eliminated, which is problematic for extending the Curry-Howard correspondence to fixed-point logics.

Non-wellfounded proof systems have been proposed as an alternative [6–8] to explicit (co)induction. By having the coinduction rule with simple fixed-point unfoldings and allowing for non-wellfounded branches, those proof systems address the problem of the subformula property for the cut-free systems: the set of subformula is then known as the set Fischer-Ladner subformulas, incorporating fixed-points unfolding, but preserving finiteness of the subformula space. Moreover, the cut-elimination dynamics for inductive-coinductive rules becomes much simpler. A particularly interesting subclass of non-wellfounded proofs, is that of circular, or cyclic proofs, that have infinite but *regular* derivations trees: they have attracted a lot of attention for retaining the simplicity of the inferences of non-wellfounded proof systems but finitely representable making it possible to have an algorithmic treatment of such proof objects. However, in those proof systems when considering all possible infinite, non-wellfounded derivations (*a.k.a.* pre-proofs), it is straightforward to derive any sequent  $\Gamma$  (see fig. 2). Such **pre-proofs** are therefore unsound: one needs to impose a validity criterion to sieve the logically valid proofs from the unsound ones. This condition will actually reflect the inductive and coinductive nature of our two fixed-point connectives: a standard approach [6–8, 24, 3] is to consider a pre-proof to be valid if every infinite branch is supported by an infinitely progressing thread. As a result, the logical correctness of circular proofs becomes non-local, much in the spirit of correctness criteria for proof-nets [17, 12].

However the structure of non-wellfounded proofs has to be further investigated: the present work stems from the observation of a discrepancy between

$$\frac{\vdash G[\nu X.G/X], \Delta}{\vdash \nu X.G, \Delta} (\nu)$$

Fig. 1: Coinduction rule

$$\frac{\frac{\vdots}{\vdash \mu X.X} (\mu) \quad \frac{\vdots}{\vdash \nu X.X, \Gamma} (\nu)}{\vdash \Gamma} (\text{cut})$$

Fig. 2: An unsound proof

the sequential nature of sequent proofs and the parallel structure of threads. An immediate consequence is that various proof attempts may have the exact same threading structure but differ in the order of inference rule applications; moreover, cut-elimination is known to fail with more expressive thread conditions. In this paper, we propose a theory of proof-nets for non-wellfounded proofs of  $\mu\text{MLL}^\infty$ .

**Organization of the paper.** In Section 1, we recall the necessary background from [3] on linear logic with least and greatest fixed points and its non-wellfounded proofs, we only present the unit-free multiplicative setting which is the framework in which we will define our proof-nets. In Section 2 we adapt Curien’s proof-nets [10] to a very simple extension of  $\text{MLL}$ ,  $\mu\text{MLL}^*$ , in which fixed-points inferences are unfoldings and only wellfounded proofs are allowed; this allows us to set the first definitions of proof-nets and extend correctness criterion, sequentialization and cut-elimination to this setting but most importantly it sets the proof-net formalism that will be used for the extension to non-wellfounded derivations. Infinites are introduced in Section 3 as an extension of the  $\mu\text{MLL}^*$  proof-nets of the previous section. A correctness criterion is defined in Section 4 which is shown to be sound (every proof-nets obtained from a sequent (pre-)proof is correct). The completeness of the criterion (*i.e.* sequentialization theorem) is addressed in Section 5. We quotient proofs differing in the order of rule application in Section 6 and establish proof-nets as canonical proof objects. We conclude in Section 8 and comment on related works and future works. An appendix provides details on some of the prerequisites and proof details.

**Notation.** Let  $[n]$  denote the set  $\{1, 2, \dots, n\}$ . For any sequence  $S$ , let  $\text{Inf}(S)$  be the terms of  $S$  that appears infinitely often in  $S$ . Given a finite alphabet  $\Sigma$ ,  $\Sigma^*$  and  $\Sigma^\omega$  are the set of finite and infinite words over  $\Sigma$  respectively.  $\Sigma^\infty = \Sigma^* \cup \Sigma^\omega$ . We denote the empty word by  $\epsilon$ . Given two words  $u, u'$  (finite or infinite) we denote by  $u \cap u'$  the greatest common prefix of  $u$  and  $u'$  and  $u \sqsubseteq u'$  if  $u$  is a prefix of  $u'$ . Given a language,  $\mathcal{L} \subseteq \Sigma^\infty$ ,  $\bar{\mathcal{L}} \subseteq \Sigma^\infty$  is the prefix closure of  $\mathcal{L}$ . Given two node-labelled trees  $T_1$  and  $T_2$ , we define  $d(T_1, T_2) = \frac{1}{2^\delta}$  where  $\delta$  is the minimal depth of the nodes at which they differ.

## 1 Background

We denote the multiplicative additive fragment of linear logic by  $\text{MALL}$  and the multiplicative fragment by  $\text{MLL}$ . The non-wellfounded extension of  $\text{MALL}$  with least and greatest fixed points operators,  $\mu\text{MALL}^\infty$ , was introduced in [3, 14]. Proof-nets for additives and units are more cumbersome, so, in the current presentation, we will only consider the unit-free multiplicative fragment which we denote by  $\mu\text{MLL}^\infty$ .

**Definition 1.** *Given an infinite set of atoms  $\mathcal{A} = \{A, B, \dots\}$ , and an infinite set of propositional variables  $\mathcal{V} = \{X, Y, \dots\}$  s.t.  $\mathcal{A} \cap \mathcal{V} = \emptyset$ ,  $\mu\text{MLL}$  pre-formulas are given by following grammar:*

$$\phi, \psi ::= A \mid A^\perp \mid X \mid \phi \wp \psi \mid \phi \otimes \psi \mid \sigma X. \phi$$

where  $A \in \mathcal{A}$  and  $X \in \mathcal{V}$ , and  $\sigma \in \{\mu, \nu\}$ ;  $\sigma$  binds the variable  $X$  in  $\phi$ . Free and bound variables, and capture-avoiding substitution are defined as usual. When a pre-formula is closed (i.e. no free variables), we simply call it a **formula**.

Note that negation is not a part of the syntax, so that we do not need any positivity condition on fixed-points expressions. We define negation,  $(\bullet)^\perp$ , as a meta-operation on the pre-formulas and will use it only on formulas.

**Definition 2.** **Negation** of a pre-formula  $\phi$ ,  $\phi^\perp$ , is the involution satisfying:

$$(\phi \otimes \psi)^\perp = \psi^\perp \wp \phi^\perp, X^\perp = X, (\mu X.\phi)^\perp = \nu X.\phi^\perp.$$

*Example 1.* As a running example, we will consider the formulas

$$\phi = A \wp A^\perp \in \text{MLL} \text{ and } \psi = \nu X.X \otimes \phi \in \mu\text{MLL}^\infty.$$

Observe that  $\phi^\perp = A^\perp \otimes A$  as usual in MLL and by def. 2,  $\psi^\perp = \mu X.X \wp \phi^\perp$ .

The reader may find it surprising to define  $X^\perp = X$ , but it is harmless since our proof system only deals with formulas. Note that we have  $(F[X/G])^\perp = F^\perp[X/G^\perp]$ .

**Definition 3.** An **address** is a word in  $\{l, r, i\}^*$ . Negation extends over addresses as the morphism satisfying  $l^\perp = r$ ,  $r^\perp = l$ , and  $i^\perp = i$ . We say that  $\alpha'$  is a **sub-address** of  $\alpha$  if  $\alpha' \sqsubseteq \alpha$ . We say that  $\alpha$  and  $\beta$  are **disjoint** if  $\alpha \cap \beta$  is not equal to  $\alpha$  or  $\beta$ .

**Definition 4.** A **formula occurrence** (denoted by  $F, G, \dots$ ) is given by a formula  $\phi$  and an address  $\alpha$ , and written  $\phi_\alpha$ . Let  $\text{addr}(\phi_\alpha) = \alpha$ . We say that occurrences are **disjoint** when their addresses are. Operations on formulas are extended to occurrences as follows:  $\phi_\alpha^\perp = \phi_{\alpha^\perp}^\perp$ , for any  $\star \in \{\wp, \otimes\}$ ,  $F \star G = (\phi \star \psi)_\alpha$  if  $F = \phi_{\alpha l}$  and  $G = \psi_{\alpha r}$ , and for  $\sigma \in \{\mu, \nu\}$ ,  $\sigma X.F = (\sigma X.\phi)_\alpha$  if  $F = \phi_{\alpha i}$ . Substitution of occurrences forgets addresses i.e.  $(\phi_\alpha)[\psi_\beta/X] = (\phi[\psi/X])_\alpha$ . Finally, we use  $[\bullet]$  to denote the address erasure operation on occurrences.

Fixed-points logics come with a notion of subformulas (and suboccurrences) slightly different from usual:

**Definition 5.** The **Fischer-Ladner closure** of a formula occurrence  $F$ ,  $\text{FL}(F)$ , is the least set of formula occurrences s.t.  $F \in \text{FL}(F)$ ,  $G_1 \star G_2 \in \text{FL}(F) \implies G_1, G_2 \in \text{FL}(F)$  for  $\star \in \{\wp, \otimes\}$ , and  $\sigma X.G \in \text{FL}(F) \implies G[\sigma X.G/X] \in \text{FL}(F)$  for  $\sigma \in \{\mu, \nu\}$ . We say that  $G$  is a **FL-suboccurrence** of  $F$  (denoted  $G \leq F$ ) if  $G \in \text{FL}(F)$  and  $G$  is an **immediate FL-suboccurrence** of  $F$  (denoted  $G < F$ ) if  $G \leq F$  and for every  $H$  s.t.  $G \leq H \leq F$  either  $H = G$  or  $H = F$ . The **FL-subformulas** of  $F$  are elements of  $\{\phi \mid \phi = [G \in \text{FL}(F)]\}$ .

Clearly, we could have defined Fischer-Ladner closure on the level of formulas. By abuse of notation, we will sometimes use  $\text{FL}(\bullet)$ ,  $\leq$ ,  $<$  on formulas.

*Remark 1.* Observe that for any  $F$ , the number of FL-subformulas of  $F$  is finite.

$$\begin{array}{c}
 \frac{\vdash [F] = [G]^\perp}{\vdash F, G} \text{ (ax)} \quad \frac{\vdash F, \Delta_1 \quad \vdash F^\perp, \Delta_2}{\vdash \Delta_1, \Delta_2} \text{ (cut)} \quad \frac{\vdash F, G, \Delta}{\vdash F \wp G, \Delta} \text{ ()} \\
 \\
 \frac{\vdash G[\nu X.G/X], \Delta}{\vdash \nu X.G, \Delta} \text{ (\nu)} \quad \frac{\vdash F, \Delta_1 \quad \vdash G, \Delta_2}{\vdash F \otimes G, \Delta_1, \Delta_2} \text{ (\otimes)} \quad \frac{\vdash G[\mu X.G/X], \Delta}{\vdash \mu X.G, \Delta} \text{ (\mu)}
 \end{array}$$

 Fig. 3: Inference rules for  $\mu\text{MLL}^\infty$ 

The usual notion of subformula (say in MLL for example) is obtained by traversing the syntax tree of a formula. In the same way, the notion of FL-subformula can be obtained by traversing the graph of the formula (*resp.* occurrence).

**Definition 6.** The **FL-graph** of a formula  $\phi$ , denoted  $\mathfrak{G}(\phi)$ , is the graph obtained from  $\text{FL}(\phi)$  by identifying the nodes of bound variable occurrences with their binders (i.e.  $\phi \rightarrow \psi$  if  $\phi \leq \psi$ ).

*Example 2.* The graphs of the formulas  $\phi$  and  $\psi$  of example 1 are the following:

$$\begin{array}{ccc}
 \mathfrak{G}(\phi) = \wp & & \mathfrak{G}(\psi) = \nu X. \\
 \swarrow \quad \searrow & & \downarrow \\
 A \quad A^\perp & & \otimes \longrightarrow \mathfrak{G}(\phi)
 \end{array}$$

Observe that the graph of a MLL formula is acyclic corresponding to the usual syntax tree but the graph of a  $\mu\text{MLL}^\infty$  formula could potentially contain a cycle.

As usual with classical linear logic  $\Gamma, \phi \vdash \Delta$  is provable iff the sequent  $\Gamma \vdash \phi^\perp, \Delta$  is provable. Hence, it is enough to consider the one-sided proof system of LL. A one-sided sequent is an expression  $\vdash \Delta$  where  $\Delta$  is a finite set of pairwise disjoint formula occurrences.

**Definition 7.** A **pre-proof** of  $\mu\text{MLL}^\infty$  is a possibly infinite tree generated from the inference rules given in fig. 3.

**Definition 8.** A **thread** of a formula occurrence  $F$  is a sequence  $t = \{F_i\}_{i \in I}$  where  $I \in \omega + 1$ ,  $F_0 = F$ , and for every  $i \in I$  s.t.  $i + 1 \in I$  either  $F_i$  is suboccurrence of  $F_{i+1}$  or  $F_i = F_{i+1}$ . We denote by  $[t]$  the sequence  $\{[F_i]\}_{i \in I}$  where  $t = \{F_i\}_{i \in I}$ . A thread  $t$  is said to be **valid** if  $\min(\text{Inf}([t]))$  is a  $\nu$ -formula where minimum is taken in the  $\leq$  ordering.

*Remark 2.* Observe that for any infinite thread  $t$  of a formula occurrence  $F$ ,  $\text{Inf}([t])$  is non-empty since  $F$  has finitely many FL-subformulas.

**Definition 9.** A  $\mu\text{MLL}^\infty$  **proof** is a pre-proof in which every infinite branch contains a valid thread. A **circular pre-proof** is a regular  $\mu\text{MLL}^\infty$  pre-proof i.e. one which has a finite number of distinct subtrees.

*Example 3.* The following non-wellfounded pre-proof of the sequent  $\vdash \psi_\alpha$  ( $\alpha$  is an arbitrary address) is circular.

$$\frac{\star \quad \frac{\frac{\vdash \psi_{\alpha il}}{\vdash \psi_{\alpha il}} \quad \frac{\frac{\vdash A_{\alpha ir l}, A_{\alpha ir r}^\perp}{\vdash A \wp A_{\alpha ir}^\perp} (\wp)}{\vdash \psi \otimes (A \wp A^\perp)_{\alpha i}} (\otimes)}{\star \vdash \psi_\alpha} (\nu)}{\star \vdash \psi_\alpha} (\text{ax})$$

It is indeed valid because the only infinite thread  $\{\psi_{\alpha(il)^n}\}_{n=0}^\infty$  is validating.

## 2 A first taste of proof-nets in logics with fixed points

Proof-nets are a geometrical method of representing proofs, introduced by Girard that eliminates two forms of bureaucracy which differentiates sequent proofs: irrelevant syntactical features and the order of rules. As a stepping stone, we first consider proof nets in  $\mu\text{MLL}^*$  which is the proof system with the same inference rules as  $\mu\text{MLL}^\infty$  (fig. 3) but with finite proofs. This logic is strictly weaker than  $\mu\text{MLL}^\infty$ .

Proof-nets are usually defined as vertex labelled, edge labelled directed multi-graphs. In this presentation a proof structure is “almost” a forest (*i.e.* a collection of trees) with the leaves joined by axioms or cuts. We use a different presentation due to Curien [10] to separate the forest of syntax trees and the space of axiom links for reasons that will become clearer later.

**Definition 10.** A **syntax tree** of a formula occurrence  $F$  is the (possibly infinite) unfolding tree of  $\mathfrak{G}(F)$ . The syntax tree induces a prefix closed language,  $\mathcal{L}_F \subset \{l, r, i\}^\infty$  s.t. there is a natural bijection between the finite (resp. infinite) words in  $\mathcal{L}$  and the finite (resp. infinite) paths of the tree. A **partial syntax tree**,  $F^U$ , is a subtree of the syntax tree of the formula occurrence,  $F$ , such that the set of words,  $U \subseteq \mathcal{L}_F$ , s.t.  $U$  represents a “frontier” of the syntax tree of  $F$  *i.e.* any  $u, u' \in U$  are pairwise disjoint and for every  $uav \in U$ , there is a  $v'$  s.t.  $ua^\perp v' \in U$ . For a finite  $u \in \overline{U}$ , we denote by  $(F, u)$  the unique suboccurrence of  $F$  with the address  $\text{addr}(F).u$ .

*Example 4.* The syntax tree of  $\psi$  is the unfolding of  $\mathfrak{G}(\psi)$  and induces the language  $\overline{i(li)^*r(l+r)} + (il)^\omega$ . Further, given an arbitrary address  $\alpha$ ,  $\psi_\alpha^{\{ili, irl, irr\}}$  is a partial syntax tree whereas  $\psi_\alpha^{\{ilil, irl, irr\}}$  is not. If  $u = ililir$  then  $(\psi_\alpha, u) = A \wp A_{\alpha u}^\perp$ .

**Definition 11.** A **proof structure** is given by  $[\Theta']\{B_i^{U_i}\}_{i \in I}[\Theta]$ , where

- $I$  is a finite index set;
- for every  $i \in I$ ,  $B_i$  is a formula occurrence,  $B_i^{U_i}$  is a partial syntax tree with  $U_i \subset \{l, r, i\}^*$ ;
- $\Theta'$  is a (possibly empty) collection of disjoint subsets of  $\{B_i\}_{i \in I}$  of the form  $\{C, C^\perp\}$ ;

- $\Theta$  is a partition of  $\bigcup_{i \in I} \{\alpha_i u_i \mid \text{addr}(B_i) = \alpha_i, u_i \in U_i\}$  s.t. the partitions are of the form  $\{\alpha_i u_i, \alpha_j u_j\}$  with  $[(B_i, u_i)] = [(B_j, u_j)]^\perp$ .

Each class of  $\Theta$  represents an axiom, each of class of  $\Theta'$  represents a cut, and  $\{B_i\}_{i \in I} \setminus \bigcup_{\theta \in \Theta'} \theta$  are the conclusions of the proof structure.

**Definition 12.** Let  $\pi$  be a  $\mu\text{MLL}^*$  proof. **Desequentialization** of  $\pi$ , denoted  $\text{Deseq}(\pi)$ , is defined by induction on the structure of the proof:

- The base case is a proof with only an ax rule. Say the proof is  $F, G^\perp$  then

$$\text{Deseq}(\pi) = [\emptyset]\{F, G^\perp\}[\{\{F, G^\perp\}\}]$$

- If  $\text{Deseq}(\pi_1) = [\Theta'_1]\Gamma_1 \cup \{F^U\}[\Theta_1]$  and  $\text{Deseq}(\pi_2) = [\Theta'_2]\Gamma_2 \cup \{F^{\perp U'}\}[\Theta_2]$ , then  $\text{Deseq}(\pi) = [\Theta'_1 \cup \Theta'_2 \cup \{F, F^\perp\}]\Gamma_1 \cup \Gamma_2[\Theta_1 \cup \Theta_2]$  where the proof  $\pi$  is

$$\frac{\frac{\pi_1}{\vdash \Gamma, F} \quad \frac{\pi_2}{\vdash \Delta, F^\perp}}{\vdash \Gamma, \Delta} (\text{cut})$$

- If  $\text{Deseq}(\pi_1) = [\Theta'_1]\Gamma_1 \cup \{F^U\}[\Theta_1]$  with  $\text{addr}(F) = \alpha l$  and  $\text{Deseq}(\pi_2) = [\Theta'_2]\Gamma_2 \cup \{G^{U'}\}[\Theta_2]$  with  $\text{addr}(G) = \alpha r$ , then  $\text{Deseq}(\pi) = [\Theta'_1 \cup \Theta'_2]\Gamma_1 \cup \Gamma_2 \cup \{F \otimes G^{l \cdot U + r \cdot U'}\}[\Theta_1 \cup \Theta_2]$  with  $\text{addr}(F \otimes G) = \alpha$  where the proof  $\pi$  is

$$\frac{\frac{\pi_1}{\vdash \Gamma, F} \quad \frac{\pi_2}{\vdash \Delta, G}}{\vdash \Gamma, \Delta, F \otimes G} (\otimes)$$

- If  $\text{Deseq}(\pi_0) = [\Theta'_0]\Gamma_0 \cup \{F^U, G^{U'}\}[\Theta_0]$  with  $\text{addr}(F) = \alpha l, \text{addr}(G) = \alpha r$  then  $\text{Deseq}(\pi) = [\Theta_0]\Gamma_0 \cup \{F \wp G^{l \cdot U + r \cdot U'}\}[\Theta_0]$  with  $\text{addr}(F \wp G) = \alpha$  where the proof  $\pi$  is

$$\frac{\frac{\pi_0}{\vdash \Gamma, F, G}}{\vdash \Gamma, F \wp G} (\wp)$$

- If  $\text{Deseq}(\pi_0) = [\Theta'_0]\Gamma_0 \cup \{F[X/F]^U\}[\Theta_0]$  with  $\text{addr}(F[X/F]) = \alpha i$  then  $\text{Deseq}(\pi) = [\Theta_0]\Gamma_0 \cup \{\mu X.F^{i \cdot U}\}[\Theta_0]$  with  $\text{addr}(\mu X.F) = \alpha$  where the proof  $\pi$  is

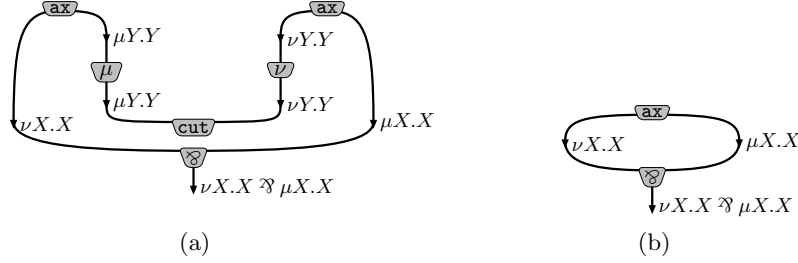
$$\frac{\frac{\pi_0}{\vdash \Gamma, F[X/F]}}{\vdash \Gamma, \mu X.F} (\mu)$$

- The case for  $\nu$  follows exactly as  $\mu$ .

*Example 5.* Consider the following proof  $\pi$  of the sequent  $\vdash \nu X.X \wp \mu X.X$ .

$$\frac{\frac{\frac{\frac{\frac{\vdash \nu X.X_{\alpha l}, \mu Y.Y_{\beta i}}{\vdash \nu X.X_{\alpha l}, \mu Y.Y_{\beta}} (\mu)}{\vdash \nu X.X_{\alpha l}, \mu Y.Y_{\beta i}} (\text{ax})}{\vdash \nu Y.Y_{\beta^\perp i}, \mu X.X_{\alpha r}} (\text{ax})}{\vdash \nu Y.Y_{\beta^\perp}, \mu X.X_{\alpha r}} (\nu)}{\vdash \nu X.X_{\alpha l}, \mu X.X_{\alpha r}} (\text{cut})}{\vdash \nu X.X \wp \mu X.X_\alpha} (\wp)$$



Fig. 4: Graph of  $\mu\text{MLL}^\infty$  proof structures

We choose  $\alpha, \beta$  s.t. they are disjoint. We have that  $\text{Deseq}(\pi) = [\Theta']\Gamma[\Theta]$  s.t.

$$\begin{aligned} \Theta' &= \{ \{ \mu Y.Y_\beta, \nu Y.Y_{\beta^\perp} \} \} & \Theta &= \{ \{ \alpha l, \beta i \}, \{ \alpha r, \beta^\perp i \} \} \\ \Gamma &= \{ \nu X.X \wp \mu X.X_\alpha^{\{l,r\}}, \mu Y.Y_\beta^{\{i\}}, \nu Y.Y_{\beta^\perp}^{\{i\}} \} \end{aligned}$$

**Definition 13 (Graph of proof structure).** Let  $\mathcal{S} = [\Theta']\{B_i^{U_i}\}_{i \in I}[\Theta]$  be a proof structure. The graph of  $\mathcal{S}$  denoted  $\text{Gr}(\mathcal{S})$  is the graph formed by:

- taking the transpose of the partial syntax tree  $\{B_i^{U_i}\}_{i \in I}$ ;
- for each  $\{B_i, B_j\} \in \Theta'$ , adding a node labelled  $\text{cut}$  with two incoming edges from  $(B_i, \epsilon)$  and  $(B_j, \epsilon)$ ;
- for each  $\{\alpha_i u_i, \alpha_j u_j\} \in \Theta$ , adding a node labelled  $\text{ax}$  with two outgoing edges to  $(B_i, u_i)$  and  $(B_j, u_j)$  where  $\text{addr}(B_i)$  and  $\text{addr}(B_j)$  is  $\alpha_i$  and  $\alpha_j$  resp.

*Example 6.* The graph of the proof structure in example 5 is Fig. 4a.

$\text{Gr}(\mathcal{S})$  are exactly the proof structures that we obtain from directly lifting the formalism of MLL proof nets à la Girard to  $\mu\text{MLL}^*$ .

As usual in the theory of proof nets, we need a correctness criterion on the  $\mu\text{MLL}^*$  proof structure to exactly characterize the class of proof nets. The following correctness criterion lifts to  $\mu\text{MLL}^*$  a criterion first investigated by Danos and Regnier [12]. We present it in a slightly different syntax using the notion of orthogonal partitions [11, 12].

**Definition 14.** Let  $P_1$  and  $P_2$  be partitions of a set  $S$ . The graph induced by  $P_1$  and  $P_2$  is defined as the undirected bipartite multigraph  $(P_1, P_2, E)$  s.t. for every  $p \in P_1$  and  $p' \in P_2$ ,  $(p, p') \in E$  if  $p \cap p' \neq \emptyset$ . Finally,  $P_1$  and  $P_2$  are said to be **orthogonal** to each other if the graph induced by them is acyclic and connected.

**Definition 15.** Given a proof structure,  $\mathcal{S} = [\Theta']\{B_i^{U_i}\}_{i \in I}[\Theta]$ , define a sets of **switchings** of  $\mathcal{S}$ ,  $sw = \{sw_i\}_{i \in I}$  s.t.  $\forall i \in I$ ,  $sw_i : P_i \rightarrow \{l, r\}$  is a function over  $P_i$ , the  $\wp$  nodes of  $B_i^{U_i}$ . The **switching graph**  $\mathcal{S}^{sw}$  associated with  $sw$  is formed by:

- taking the partial syntax tree  $\{B_i^{U_i}\}_{i \in I}$  as an undirected graph;

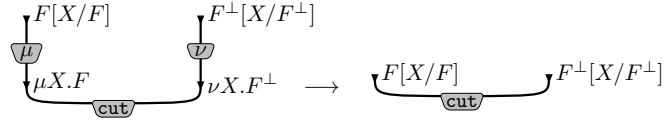
- for each  $\{B_i, B_j\} \in \Theta'$ , adding a node labelled *cut* with two edges to  $(B_i, \epsilon)$  and  $(B_j, \epsilon)$ ;
- for each node  $(B_i, u) \in P_i$ , removing the edge between  $(B_i, u)$  and  $(B_i, u \cdot sw((B_i, u)))$ .

Let  $\Theta_S^{sw}$  be the partition over  $\bigcup_{i \in I} \{\alpha_i u_i \mid \text{addr}(B_i) = \alpha_i, u_i \in U_i\}$  induced by the connected component of  $\mathcal{S}^{sw}$ .

**Definition 16.** A proof structure,  $\mathcal{S}$ , is said to be **OR-correct** if for any switching  $sw$ ,  $\Theta_S^{sw}$  and  $\Theta$  is orthogonal. The graph induced by  $\Theta_S^{sw}$  and  $\Theta$  is called a **correction graph** of  $\mathcal{S}$ .

**Proposition 1.** Let  $\pi$  be a  $\mu\text{MLL}^*$  proof. Then  $\text{Deseq}(\pi)$  is an OR-correct proof structure. Conversely, given an OR-correct  $\mu\text{MLL}^*$  proof structure, it can be sequentialized into a  $\mu\text{MLL}^*$  sequent proof.

**Definition 17.**  $\mu\text{MLL}^*$  **cut-reduction rules** is obtained by adding the following rule to the usual cut-reduction rules for MLL proof nets:



**Proposition 2.** Cut elimination on  $\mu\text{MLL}^*$  proof-nets preserves correctness and is strongly normalizing and confluent.

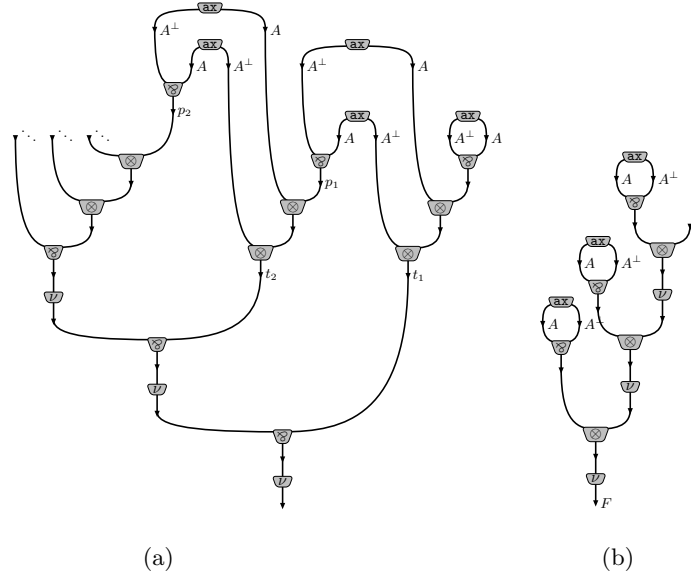
The proofs of propositions 1, 2 are straightforward extensions from MLL.

*Example 7.* The proof structure in example 5 after cut-elimination produces the proof structure in Fig. 4b.

Now the question is how this translates to non-wellfounded proofs. Consider the proof in example 3. Firstly observe that there is no finite proof of this sequent *i.e.* it is not provable in  $\mu\text{MLL}^*$ . Now, if we naively translate it into a proof structure using the same recipe as definition 12, we have

$$[\emptyset] \left\{ \psi_{\alpha}^{\{i(li)^* r(l+r) + (il)^{\omega}\}} \right\} \left[ \{\alpha i(li)^n r l, i(li)^n r r\}_{n \geq 0} \right].$$

Observe that  $(il)^{\omega}$  is not in any partition. In fact, it represents a thread in an infinite branch and must be accounted for. Hence the partition should be equipped to account for the threads invariant by an infinite branch in a proof (in particular, in the example above there should be a singleton partition  $\{(il)^{\omega}\}$ ). This is also the reason we will use the graphical presentation for non-wellfounded proof-nets since we would potentially need to join two infinite paths by a node which is unclear graph-theoretically. However we will sometimes draw the “graph” of non-wellfounded proof-nets for ease of presentation by using ellipsis points to brush the technical difficulty under the carpet (for example Fig. 5b represents the proof net we discussed above).

Fig. 5: Graph of  $\mu\text{MLL}^\infty$  NWFPS

### 3 Infinets

We will now lift our formalism for defining proof nets for  $\mu\text{MLL}^*$  to  $\mu\text{MLL}^\infty$ .

**Definition 18.** A non-wellfounded proof structure (NWFPS) is given by  $[\Theta']\{B_i^{U_i}\}_{i \in I}[\Theta]$ , where

- $I$  is a possibly infinite index set;
- for every  $i \in I$ ,  $B_i$  is a formula occurrence,  $B_i^{U_i}$  is a partial syntax tree;
- $\Theta'$  is a (possibly empty) collection of disjoint subsets of  $\{B_i\}_{i \in I}$  of the form  $\{C, C^\perp\}$ ;
- $\Theta$  is a partition of  $\bigcup_{i \in I} \{\alpha_i u_i \mid \text{addr}(B_i) = \alpha_i, u_i \in U_i\}$  s.t. the partitions are one of the following forms:
  - $\{\alpha_i u_i, \alpha_j u_j\}$  s.t.  $u_i, u_j$  are finite and  $\lceil (B_i, u_i) \rceil = \lceil (B_j, u_j) \rceil^\perp$ .
  - It contains an elements of the form  $\alpha_i u_i$  s.t.  $u$  is an infinite word;
- $\{B_i\}_{i \in I} \setminus \bigcup_{\theta \in \Theta'} \theta$  is necessarily finite.

Intuitively, each class of  $\Theta$  represents either an axiom or an infinite branch in a sequentialization. In fact, infinite words in a partition corresponds exactly to the infinite threads in a proofs. Hence it is also straightforward to define a valid NWFPS.

**Definition 19.** Let  $\pi$  be a pre-proof of the  $\mu\text{MLL}^\infty$  sequent  $\vdash \Gamma$ .  $\text{Deseq}(\pi)$  is the (co)inductive translation of  $\pi$  into a NWFPS,  $[\Theta']\Gamma'[\Theta]$  s.t.  $\Theta'$  are the cut

formulas in  $\pi$ ,  $\Gamma' = \Gamma \cup \bigcup_{\theta \in \Theta} \theta$ , and  $\Theta$  is a partition of the set of formula occurrences in  $\pi$  s.t. two formula occurrences are in the same partition if they are conclusions of an  $\text{ax}$  rule in  $\pi$ . A NWFPS that is the desequentialization of a  $\mu\text{MLL}^\infty$  (pre-)proof is called an (valid) **infinet**.

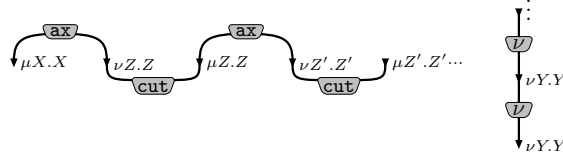
## 4 Correctness criteria

The OR-correctness of a NWFPS is defined as in def. 15 and def. 16 (up to the fact that the switching can be an infinite set of switching functions). However this straightforward translation is not enough to ensure soundness.

*Example 8.* Consider the following sequent proof with infinitely many cuts.

$$\frac{\frac{}{\mu X.X, \nu Z.Z} (\text{ax}) \quad \frac{\star \vdash \mu Z.Z, \nu Y.Y}{\vdash \mu Z.Z, \nu Y.Y} (\nu)}{\star \vdash \mu X.X, \nu Y.Y} (\text{cut})$$

Observe that this structure is not OR-correct:



Consequently, we restrict ourselves to NWFPS with at most finitely many cuts. The proof structures discussed in the rest of the paper have finitely many cuts unless otherwise mentioned.

*Example 9.* Consider the proof structure of the sequent  $\vdash \nu X.X \wp (A^\perp \otimes (A \otimes (A^\perp \wp A)))$  in fig. 5a. Note that for the sake of readability, edge labels have been concealed. This proof structure is OR-correct but it is not sequentializable. Consider the  $\otimes$  node labelled  $t_1$ . In any sequentialization it should be above  $p_1$ , which should be above  $t_2$ , which in turn should be above  $p_2$  and so on. This is absurd since even in a non-wellfounded proof every rule is executed at a finite depth.

Hence we impose a “lock-free” condition (borrowing the terminology from concurrent programs) on NWFPS.

**Definition 20.** Let  $[\Theta']\{B_i^{U_i}\}_{i \in I}[\Theta]$  be a NWFPS. For any  $u_i \in U_i, u_j \in U_j$ , we say that  $(u_i, u_j)$  is a **coherent pair** if there exists  $\theta \in \Theta$  s.t.  $\{\alpha_i u_i, \alpha_j u_j\} \subseteq \theta$  where  $\text{addr}(B_i) = \alpha_i$  and  $\text{addr}(B_j) = \alpha_j$ .

**Definition 21.** A **switching path** is an undirected path in a partial syntax tree s.t. it does not go consecutively through the two premises of a  $\wp$  formula. A **strong switching path** is a switching path whose first edge is not the premise of a  $\wp$  node. We denote by  $\text{src}(\bullet), \text{tgt}(\bullet)$  the source and target of a switching path resp. Two switching paths  $\gamma, \gamma'$  are said to be **compatible** if  $\gamma'$  is strong and  $\text{tgt}(\gamma) = \text{src}(\gamma')$ .

**Proposition 3.** *If  $\gamma, \gamma'$  are compatible switching paths, then their concatenation  $\gamma \cdot \gamma'$  is a switching path. Furthermore, if  $\gamma$  is strong, then  $\gamma \cdot \gamma'$  is also strong.*

The underlying undirected path of any path in a partial syntax tree is a switching path. We call such paths **straight switching paths**. In particular, the path from any node,  $n$ , to the root is a straight switching path. We denote it by  $\delta(n)$ . By abuse of notation, we will also sometimes write  $\delta((B_i, u))$  where  $u$  is infinite to mean the infinite path from the root of  $B_i^{U_i}$  following  $u$ , although technically  $(B_i, u)$  is not a node per se. Observe that any straight switching path in a partial syntax tree,  $F^U$ , can be represented by a pair of words  $(u, u') \in \overline{U}^2$  s.t.  $u \sqsubset u'$ . Intuitively, it means that the path is from  $(F, u)$  to  $(F, u')$ .

**Definition 22.** *A switching sequence is a sequence  $\sigma = \{\gamma_i\}_{i=1}^n$  s.t.  $\gamma_i$ s are disjoint switching paths and for every  $i \in [n-1]$ , either  $\gamma_i, \gamma_{i+1}$  are compatible or they are straight and the word pairs corresponding to them,  $(u_i, u'_i)$  and  $(u_{i+1}, u'_{i+1})$ , are s.t.  $(u'_i, u'_{i+1})$  is a coherent pair. Two nodes,  $N$  and  $N'$ , are said to be connected by the switching sequence,  $\sigma$ , if  $\text{src}(\gamma_1) = N$  and  $\text{tgt}(\gamma_n) = N'$ . We say the switching sequence is cyclic if  $\text{src}(\gamma_1) = \text{tgt}(\gamma_n)$ .*

**Definition 23.** *Let  $\mathcal{S} = [\Theta']\{B_i^{U_i}\}[\Theta]$  be a proof structure. Let  $T = \{(B_i, u_i) \mid u_i \in \overline{U}_i; (B_i, u_i) \text{ is a } \otimes \text{ formula}\}$  and let  $P = \{(B_i, u_i) \mid u_i \in \overline{U}_i; (B_i, u_i) \text{ is a } \wp \text{ formula}\}$ . The **dependency graph** of  $\mathcal{S}$ ,  $D(\mathcal{S})$ , is the directed graph  $(V, E)$  s.t.  $V = T \uplus P$ , for every  $v \in V$  and  $p \in P$ ,  $(p, v) \in E$  if the premises of  $p$  are connected by a switching sequence containing  $v$ , and, for every  $v, v' \in V$ ,  $(v, v') \in E$  if  $v' \in \text{FL}(v)$ .*

**Proposition 4 (Bagnol et al. [5]).** *If  $\mathcal{S}$  is OR-correct then  $D(\mathcal{S})$  is acyclic.*

From prop. 4, we can impose an order on the nodes of an OR-correct proof structure,  $\mathcal{S}$ , namely,  $n_1 < n_2$  if  $n_1 \rightarrow n_2$  in  $D(\mathcal{S})$ , denoted as  $<_{D(\mathcal{S})}$ .

**Definition 24.** *A NWFPS,  $\mathcal{S}$ , is said to be **deeply lock-free** if  $<_{D(\mathcal{S})}$  has no infinite descending chains.*

*Example 10.* Consider the proof structure,  $\mathcal{S} = [\emptyset]\{\nu X.X \wp X_\alpha^L, A \otimes B_\beta^{\{l,r\}}\}[\Theta]$  where,  $L = (i(l+r))^\omega$ ,  $\Theta = \{\{\alpha(il)^\omega, \beta l\}, \alpha \cdot (L \setminus (il)^\omega) \cup \{\beta r\}\}$ .

Observe that  $\mathcal{S}$  is OR-correct and deeply lock-free. But  $\mathcal{S}$  cannot be sequentialized into a sequent proof, because a potential sequentialization has a  $\otimes$  rule at a finite depth, then either there are some subsoccurrences of  $\nu X.X \wp X_\alpha$  in the left premise in which case  $A$  cannot reside with only the left-branch in  $\Theta$ , or, there are some subsoccurrences of  $\nu X.X \wp X_\alpha$  in the left premise in which case  $A$  cannot reside with any infinite branch in  $\Theta$ .

**Definition 25.** *A NWFPS,  $\mathcal{S} = [\Theta']\{B_i^{U_i}\}_{i \in I}[\Theta]$ , is said to be **widely lock-free** if there is a function  $f : \mathbb{N} \rightarrow \mathbb{N}$  s.t. for every  $(B_i, u) \in P$  and  $(B_j, v) \in T$  if  $((B_i, u), (B_j, v)) \in E$ ,  $f(|v|) \geq |u|$  where  $D(\mathcal{S}) = (T \uplus P, E)$ . We call such a function a **wait function** of  $\mathcal{S}$ . A proof structure is simply called **lock-free** if it is both deeply and widely lock-free.*

*Remark 3.* The wait function of a NWFPS need not be unique (if one exists).

**Proposition 5.** *An infinit is an OR-correct lock-free NWFPS.*

*Proof (Sketch).* Let  $\pi$  be a  $\mu\text{MLL}^\infty$  pre-proof with finitely many cuts and let  $\text{Deseq}(\pi) = \mathcal{S}$ . OR-correctness of  $\mathcal{S}$  is proved by (co)induction on the structure of  $\pi$  (see appendix sec. C.1 for more details). To prove that  $\mathcal{S}$  is deeply lock-free, we show that  $\prec_{D(\mathcal{S})} \subseteq \prec_{O(\pi)}$  where  $n_1 \prec_{O(\pi)} n_2$  if  $n_1$  is introduced below  $n_2$  in  $\pi$  (see prop. 5 in [5]). To prove that  $\mathcal{S}$  is widely lock-free, we will define a wait function,  $f$ . Let  $n \in \mathbb{N}$ . Let  $d$  be the maximal (in depth)  $\otimes$  inference in  $\pi$  s.t. the address,  $u$ , of the formula introduced satisfies  $|u| \leq n$ . Consider the set,  $\{P_i\}$  of all  $\wp$  inferences under  $d$  where  $P_i$  is the formula introduced. Then define  $f(n) := \max\{|\alpha_i| \mid \text{addr}(P_i) = \alpha_i\}$ . Check that  $f$  is indeed a wait function and hence we are done.

## 5 Sequentialization

In this section we show that any NWFPS satisfying the correctness criterion introduced in section 4 is indeed sequentializable. Since we deal with finitely many cuts, without loss of generality, we can assume that we have cut-free proof structures due to the standard trick shown in Fig. 6.

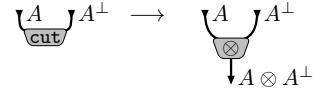


Fig. 6: Translating cuts to tensors

So, in this section, we will write NWFPS without the left component. We try to adapt the standard proof for MLL but the straightforward adaptation is not *fair* since we may never explore one branch by forever prioritizing the sequentialization of another infinite branch. We restore fairness by a time-stamping algorithm.

**Definition 26.** *Let  $\mathcal{S} = \Gamma[\Theta]$  be an OR-correct NWFPS. The root,  $B_i$ , of a tree in  $\Gamma$  is said to be **splitting** if:*

- $\Gamma = \{B_i^\epsilon, B_j^\epsilon\}$ ,
- $B_i$  is a  $\wp, \mu$  or  $\nu$  formula, or,
- $B_i$  is a  $\otimes$  formula and there exists  $\Theta_1, \Theta_2$  s.t.  $\Theta = \Theta_1 \uplus \Theta_2$  and  $\mathcal{S}_1 = \Gamma_1[\Theta_1]$ ,  $\mathcal{S}_2 = \Gamma_2[\Theta_2]$  are OR-correct NWFPS where  $\Gamma_1 = \Gamma \setminus \{B_i^{U_i}\} \cup \{(B_i, l)^{U_i}\}$ ,  $\Gamma_2 = \Gamma \setminus \{B_i^{U_i}\} \cup \{(B_i, r)^{U_i}\}$  and  $U_i = lU_i + rU_i$ .

**Proposition 6.** *Let  $\mathcal{S} = \Gamma[\Theta]$  be an OR-correct NWFPS and  $B_i$  be a splitting tensor in  $\mathcal{S}$ . If  $\mathcal{S}$  is lock-free then so is  $\mathcal{S}_1$  and  $\mathcal{S}_2$  as defined in def. 26.*

*Dated sequentialization process.* We time-stamp each node of  $\Gamma$  to indicate the time when it will be sequentialized. Formally, we have  $(\mathcal{S}, \tau)$  where  $\tau$  is a function s.t.  $\tau : \{(B_i, u) \mid u \in U_i\}_{i \in I} \rightarrow \mathbb{N} \cup \{\infty\}$  where  $\Gamma = \{B_i^{U_i}\}_{i \in I}$  and  $\infty > n$  for all  $n \in \mathbb{N}$ . Define the minimal finite image,  $\min$ , as

$$\min(\tau) := \min\{n \in \mathbb{N} \mid \exists i \in I, u \in U_i \text{ s.t. } \tau((B_i, u)) = n\}.$$

We will describe the sequentialization process. Suppose we are given  $\mathcal{S}(= \Gamma[\Theta], \tau)$ . We maintain the following invariant:

$$\begin{aligned} \mathcal{S} \text{ is cut-free, OR-correct and lock-free;} \\ \tau((B_i, u)) \neq \infty \text{ iff } (B_i, u) \text{ is splitting in } \mathcal{S}. \end{aligned} \quad (\star)$$

Assume that  $\Gamma$  contains a splitting root,  $B_j$ , st.  $\tau(B_j) = \min(\tau)$ .

- If  $\Gamma = \{B_i^c, B_j^c\}$  then we stop successfully with the proof reduced to an ax.
- If  $B_j$  is a  $\mathfrak{A}$ , (co)recursively apply the sequentialization process to  $\mathcal{S}_0(= \Gamma_0[\Theta], \tau_0)$  where  $\Gamma_0 = \Gamma \setminus \{B_j^{U_j}\} \cup \{(B_j, l)^{U_l}, (B_j, r)^{U_r}\}$ ,  $U_j = lU_l + rU_r$ , and

$$\tau_0((B_i, u)) = \begin{cases} t & \text{if } (B_i, u) \text{ is splitting in } \mathcal{S}_0; \\ \tau((B_i, u)) & \text{otherwise.} \end{cases}$$

where for each splitting  $(B_i, u)$ ,  $t$  is arbitrarily chosen to be any natural number greater than  $\tau(B_j)$ . We apply a  $\mathfrak{A}$  rule on the obtained proof.

- If  $B_j$  is a  $\mu$  (resp.  $\nu$ ) formula, (co)recursively apply the sequentialization process to  $(\mathcal{S}_0 = \Gamma_0[\Theta], \tau_0)$  where  $\Gamma_0 = \Gamma \setminus \{B_j^{U_j}\} \cup \{(B_j, i)^{U_i}\}$ ,  $U_j = iU_i$ , and  $\tau_0$  is defined as above. We apply a  $\mu$  (resp.  $\nu$ ) rule on the obtained proof.
- If  $B_j$  is a  $\otimes$  formula we (co)recursively apply the sequentialization process to  $(\mathcal{S}_1, \tau_1)$  and  $(\mathcal{S}_2, \tau_2)$  where  $\mathcal{S}_1, \mathcal{S}_2$  are as defined in def. 26 and  $\tau_1, \tau_2$  are defined as above. We apply a  $\otimes$  rule on the two obtained proofs.

Observe that the invariant  $(\star)$  is maintained in this (co)recursive process.

To start the sequentialization, we initialize  $\tau$  by assigning arbitrary natural numbers to splitting nodes and  $\infty$  to the other nodes.

**Proposition 7.** *Let  $\gamma$  be a switching path in  $B_j^{U_j} \in \Gamma$ . Then there exists a switching sw s.t.  $\gamma$  is also a path in the switching graph,  $\mathcal{S}^{sw}$ .*

**Proposition 8.** *If  $\mathcal{S}$  is a NWFPS containing a cyclic switching sequence, then there is switching of  $\mathcal{S}$ , s.t. the corresponding correction graph is contains a cycle.*

**Proposition 9.** *Let  $T$  be a non-splitting conclusion in an OR-correct NWFPS. Then there exists a  $\mathfrak{A}$  formula,  $P$ , s.t. there exists disjoint switching path sequences,  $\sigma, \sigma'$ , from  $T$  to  $P$  which both start with a premise of  $T$  and end with a premise of  $P$ . We call  $(P, \sigma, \sigma')$  the **witness** for  $T$ .*

**Lemma 1.** *Let  $\mathcal{S}$  be a cut-free OR-correct NWFPS.  $\mathcal{S}$  contains a splitting root.*

**Lemma 2.** *The sequentialization assigns a finite natural number to every formula i.e.  $\tau((B_i, u)) \neq \infty$  after some iterations of the process described above.*

Lemma. 1 crucially uses OR-correctness and lemma. 2 crucially uses lock-freeness. They are proved in full details in the appendix. Lemma. 1 ensures productivity of the aforementioned sequentialization process while lemma. 2 ensures that every inference in a NWFPS is ultimately executed. From that, we conclude the following theorem.

**Theorem 1.** *Let  $\mathcal{S} = [\Theta']\Gamma[\Theta]$  be an OR-correct lock-free NWFPS s.t.  $\Theta' = \emptyset$ . Then  $\mathcal{S}$  is an infinitet.*

*Remark 4.* Observe that the choice of the time-stamping function at each step of our sequentialization is non-deterministic. By considering appropriate time-stamping functions we can generate all sequentializations. The detailed study is beyond the scope of the present paper.

## 6 Canonicity

We started investigating proof nets for non-wellfounded proofs since we expected that the proof net formalism would quotient sequent proofs that are equivalent up to a permutation of inferences. At this point, we carry out that sanity check.

Consider the following proofs  $\pi_1$  and  $\pi'_1$ .

$$\frac{\frac{\pi_2}{\vdash \Gamma, F[X/F], A}(\mu)}{\vdash \Gamma, \mu X.F, A}(\mu) \quad \frac{\pi_3}{\vdash B, \Delta}}{\pi_1 \vdash \Gamma, \mu X.F, A \otimes B, \Delta}(\otimes) \qquad \frac{\frac{\pi_2}{\vdash \Gamma, F[X/F], A} \quad \frac{\pi_3}{\vdash B, \Delta}}{\vdash \Gamma, F[X/F], A \otimes B, \Delta}(\otimes)}{\pi'_1 \vdash \Gamma, \mu X.F, A \otimes B, \Delta}(\mu)$$

We say that  $\pi \rightsquigarrow_{(\mu, \otimes_L)} \pi'$  if  $\pi$  is a proof with  $\pi_1$  as a subproof at a finite depth and  $\pi'$  is  $\pi$  where  $\pi'_1$  has been replaced by  $\pi'_1$ . Observe that we can define  $\rightsquigarrow_{\square}$  for every  $\square \in P \times P$  where  $P = \{\mu, \nu, \wp, \otimes_*, \text{cut}_* \mid * \in \{L, R\}\}$ . Let  $\sim_{\blacksquare} = \bigcup_{\square \in S} \rightsquigarrow_{\square}$ .

Observe that the usual notion of equivalence by permutation, *viz.*  $\sim = (\sim_{\blacksquare})^*$  does not characterize equivalence by infinites. Consider the following two proofs,  $\pi_1$  and  $\pi_2$ , s.t.  $\pi_1 \not\sim \pi_2$  which have the same infinitet,

$$[\emptyset]\{\mu X.X_{\alpha}^{i\omega}, \nu X.X_{\beta}^{i\omega}\}[\{\{\alpha i\omega, \beta i\omega\}\}].$$

$$\frac{\frac{\pi_1 \vdash \mu X.X, \nu X.X}{\vdash \mu X.X, \nu X.X}(\mu)}{\pi_1 \vdash \mu X.X, \nu X.X}(\nu) \qquad \frac{\frac{\pi_2 \vdash \mu X.X, \nu X.X}{\vdash \mu X.X, \nu X.X}(\nu)}{\pi_2 \vdash \mu X.X, \nu X.X}(\mu)$$

Suppose we allow infinite permutations. We say that  $\pi (\sim_{\blacksquare})^{\omega} \pi'$  if there exists a proof  $\pi''$  (not necessarily different from  $\pi, \pi'$ ) and two sequence of proofs,  $\{\pi_i\}_{i=0}^{\infty}$  and  $\{\pi'_i\}_{i=0}^{\infty}$ , s.t.  $\pi_0 = \pi, \pi'_0 = \pi'$ , for every  $i, \pi_i \sim_{\blacksquare} \pi_{i+1}$  and  $\pi'_i \sim_{\blacksquare} \pi'_{i+1}$ , and  $d(\pi_i, \pi'') \rightarrow 0, d(\pi'_i, \pi'') \rightarrow 0$  as  $i \rightarrow \infty$ . Consider the following three proofs of  $\vdash A \otimes B, \nu Y.Y$ .

$$\frac{\frac{\pi}{\vdash A} \quad \frac{\vdots}{\vdash B, \nu Y.Y}(\nu)}{\vdash A \otimes B, \nu Y.Y}(\otimes) \quad (\sim_{\blacksquare})^{\omega} \quad \frac{\vdots}{\vdash A \otimes B, \nu Y.Y}(\nu) \quad (\sim_{\blacksquare})^{\omega} \quad \frac{\frac{\pi'}{\vdash A} \quad \frac{\vdots}{\vdash B, \nu Y.Y}(\nu)}{\vdash A \otimes B, \nu Y.Y}(\otimes)$$

Note that  $\pi$  and  $\pi'$  can have different computation behaviour (say, for example  $A = (X^{\perp} \wp X^{\perp}) \wp (X \otimes X)$  and  $\pi$  corresponds to **true** while  $\pi'$  corresponds to



false (see example 11 in appendix A)). Hence equating these proofs is absurd. To exactly capture equivalence by proof nets we need to refine this equivalence. To do that we introduce the notion of an active occurrence. We say that for a permutation step  $\rightsquigarrow_{(r_i, r'_i)}$ , the formula occurrence  $F_i$  introduced by the rule  $r'_i$  is the **active occurrence** in that step.

We say that  $\pi(\sim_{\blacksquare})_{\text{fair}}^{\omega} \pi'$  if there exists a sequence of proofs  $\{\pi_i\}_{i=0}^{\infty}$  s.t.  $\pi_0 = \pi$ , for every  $i$ ,  $\pi_i \rightsquigarrow_{(r_i, r'_i)} \pi_{i+1}$ , the set of addresses of the active occurrences occurring infinitely often is empty,  $\text{Inf}(\{\text{addr}(F_i)\}_{i=0}^{\infty}) = \emptyset$  and  $d(\pi_i, \pi') \rightarrow 0$  as  $i \rightarrow \infty$ . Let  $\sim^{\infty} = (\sim_{\blacksquare})^* \cup (\sim_{\blacksquare})_{\text{fair}}^{\omega}$ .

**Proposition 10.**  $\pi_1 \sim^{\infty} \pi_2$  iff  $\text{Deseq}(\pi_1) = \text{Deseq}(\pi_2)$ .

## 7 Cut Elimination

In this section we provide cut elimination results albeit with two crucial restrictions: firstly, we consider only finitely many cuts as in the rest of the paper and secondly, we consider proofs with no axioms and no atoms. An infinet  $S = [\Theta']\Gamma[\Theta]$  is said to be  $\eta^{\infty}$ -**expanded** if it does not contain any axioms or atoms i.e. every  $\theta \in \Theta$  contains only infinite words. Any infinet can be made  $\eta^{\infty}$ -expanded in a way akin to  $\eta$ -expansion of axioms in MLL (see appendix sec. B). There are two issues to be resolved to obtain the result: first, to specify the notion of a normal form and second, formulate how to reach that.

**Proposition 11.** Let  $S = [\Theta']\Gamma[\Theta]$  be an  $\eta^{\infty}$ -expanded infinet. Let  $\{C, C^{\perp}\} \in \Theta$  and  $B_i^{U_i}, B_j^{U_j} \in \Gamma$  s.t.  $B_i = C = B_j^{\perp}$ .  $U_i = U_j^{\perp}$  i.e.  $u \in U_i$  iff  $u^{\perp} \in U_j$ .

*Proof (Sketch).* Since  $B_i = B_j^{\perp}$ , their syntax trees are orthogonal. Since  $S$  is  $\eta^{\infty}$ -expanded,  $U_i$  (resp.  $U_j$ ) is actually the full syntax tree. Hence  $U_i = U_j^{\perp}$ .

**Definition 27.** Let  $S_0 = [\Theta'_0]\Gamma_0[\Theta_0]$  be a  $\eta^{\infty}$ -expanded infinet. Let  $\{C, C^{\perp}\} \in \Theta'_0$  and  $B_i^{U_i}, B_j^{U_j} \in \Gamma_0$  s.t.  $B_i = C = B_j^{\perp}$ . A **big-step**  $\{C, C^{\perp}\}$  **elimination** on  $S_0$  produces non-wellfounded proof-structure  $S_1 = [\Theta'_1]\Gamma_1[\Theta_1]$  where,

- $\Theta'_1 = \Theta'_0 \setminus \{\{C, C^{\perp}\}\}$
- $\Gamma_1 = \Gamma_0 \setminus \{B_i^{U_i}, B_j^{U_j}\}$
- If  $\theta \in \Theta_0$  s.t.  $\theta \cap U_i = \emptyset$  and  $\theta \cap U_j = \emptyset$ , then  $\theta \in \Theta_1$ . If  $u \in \theta \cap U_i$  then  $\theta \cup \theta' \setminus \{u, u^{\perp}\} \in \Theta_1$  where  $\theta' \in \Theta_0$  and  $u^{\perp} \in \theta' \cap U_j$ .

*Remark 5.* Definition 27 is well-defined because of proposition 11.

**Proposition 12.** A big-step operation on a valid infinet produces a valid infinet.

Given  $S = [\Theta']\Gamma[\Theta]$ , an  $\eta^{\infty}$ -expanded infinet, we can extend the definition of a big-step  $\{C, C^{\perp}\}$  elimination on  $S$ , for any  $\{C, C^{\perp}\} \in \Theta'$ , to a big-step  $C$  elimination on  $S$ , for  $\mathcal{C} \subseteq \Theta'$ . We call the big-step  $\Theta'$  elimination on  $S$  the **normal form** of  $S$  and denote it by  $\llbracket S \rrbracket$ .

We will now construct a suitable distance on infinites with the goal that this distance will converge for cut reduction sequences. Note that our distance will only be well-defined over infinites with the same normal form. Given two such infinites, we will first identify partitions with common conclusions *i.e.* given  $S_1 = [\Theta'_1]\Gamma_1[\Theta_1]$  and  $S_2 = [\Theta'_2]\Gamma_2[\Theta_2]$ , we define relation  $R \subseteq \Theta_1 \times \Theta_2$  such that for every  $\theta_1 \in \Theta_1$  and  $\theta_2 \in \Theta_2$ ,  $\theta_1 R \theta_2$  iff there exists  $u \in \theta_1 \cup \theta_2$  such that

$$u \in (\Gamma_1 \cup \Gamma_2) \setminus \bigcup_{\theta \in \Theta_1, \Theta_2} \theta$$

Observe that  $R$  is symmetric. Now we will fix a measure on partitions. Define

$$f(\theta_1, \theta_2) = \begin{cases} 0 & \text{if } \theta_1 = \theta_2; \\ (\min\{|\alpha| \mid \exists u \in \{l, r, i\}^\infty, B \in \Gamma_1 \cup \Gamma_2 \text{addr}(B) = \alpha, \alpha u \in \theta_1 \Delta \theta_2\})^{-1} & \text{otherwise.} \end{cases}$$

**Lemma 3.** *The set of all valid infinites with the same normal form is a metric space.*

We can now define the limit of an infinite sequence of valid infinites with the same normal form in the standard way: we say that  $\{\mathcal{S}_i\}_{i=0}^\infty$  converges to  $\mathcal{S}$  if  $d(\mathcal{S}_i, \mathcal{S}) \rightarrow 0$  as  $i \rightarrow \infty$  (where  $d$  is the metric obtained from lemma 3).

**Definition 28.** *A sequence of infinites,  $\{\mathcal{S}_i\}_{i=0}^\infty$ , is called a **reduction sequence** if for every  $i > 0$ ,  $\mathcal{S}_i \rightarrow \mathcal{S}_{i+1}$  by the cut reduction rules in definition 17. A reduction sequence is said to be **fair** if for every  $i$ , for every cut  $\{C, C^\perp\}$  in  $\mathcal{S}_i$ , there is a  $j > i$  such that  $C'$  is a suboccurrence of  $C$  where  $\{C', C'^\perp\}$  is the cut being reduced in the step  $\mathcal{S}_j \rightarrow \mathcal{S}_{j+1}$ .*

**Theorem 2.** *Let  $\{\mathcal{S}_i\}_{i=0}^\infty$  be a fair reduction sequence s.t.  $\mathcal{S}_0$  is valid. Then, it converges to  $\llbracket \mathcal{S}_0 \rrbracket$ .*

**Corollary 1.** *If two reduction sequences starting from a valid  $\eta^\infty$ -expanded infinitet,  $\mathcal{S}$ , converges to  $\mathcal{S}_1$  and  $\mathcal{S}_2$ , then all fair reduction sequences starting from  $\mathcal{S}_1$  and  $\mathcal{S}_2$  resp. converge to  $\llbracket \mathcal{S} \rrbracket$ .*

## 8 Conclusion

In this paper, we introduced infinitary proof-nets for multiplicative linear logic with least and greatest fixed points with non-well-founded proofs. We defined a correctness criterion and showed its soundness and completeness in characterizing those proof structures which come from non-wellfounded sequent (pre)proofs. We also give a partial cut elimination result.

*Related and future works.* The closest works we know of are Montelatici's polarized proof nets with cycles [21] and Mellies' work on higher-order parity automata [20] which considers a  $\lambda Y$ -calculus and an infinitary  $\lambda$ -calculus endowed with parity conditions, therefore quotienting some of the non-determinism of sequent-calculus albeit in the case of intuitionistic logic. Our work is a first step in developing a general theory of non-wellfounded and circular proof-nets:

- We plan to extend our formalism to capture proofs with infinitely many cuts: a first step being to strength the correctness criterion and then trying to capture *most* sequentializations since the standard trick of converting cuts to tensors would yield infinitely many conclusions.
- We plan to carry an investigation of the notion of circularity in proof-nets: while one can capture circular proofs as finitely representable proof nets, there are non-wellfounded proofs which are not circular but which are, once desequentialized. The most simple example of this question are the proofs of  $\vdash \nu X.X \wp X$  which contain sequents of unbounded size and are therefore not circular.

## References

1. David Baelde. On the proof theory of regular fixed points. In Martin Giese and Arild Waaler, editors, *Automated Reasoning with Analytic Tableaux and Related Methods, 18th International Conference, TABLEAUX 2009, Oslo, Norway, July 6-10, 2009. Proceedings*, volume 5607 of *Lecture Notes in Computer Science*, pages 93–107. Springer, 2009.
2. David Baelde. Least and greatest fixed points in linear logic. *ACM Transactions on Computational Logic (TOCL)*, 13(1):2, 2012.
3. David Baelde, Amina Doumane, and Alexis Saurin. Infinitary proof theory: the multiplicative additive case. In *25th EACSL Annual Conference on Computer Science Logic, CSL 2016, August 29 - September 1, 2016, Marseille, France*, volume 62 of *LIPICs*, pages 42:1–42:17. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016.
4. David Baelde and Dale Miller. Least and greatest fixed points in linear logic. In Nachum Dershowitz and Andrei Voronkov, editors, *Logic for Programming, Artificial Intelligence, and Reasoning, 14th International Conference, LPAR 2007, Yerevan, Armenia, October 15-19, 2007, Proceedings*, volume 4790 of *Lecture Notes in Computer Science*, pages 92–106. Springer, 2007.
5. Marc Bagnol, Amina Doumane, and Alexis Saurin. On the dependencies of logical rules. In Andrew M. Pitts, editor, *Foundations of Software Science and Computation Structures - 18th International Conference, FoSSaCS 2015, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2015, London, UK, April 11-18, 2015. Proceedings*, volume 9034 of *Lecture Notes in Computer Science*, pages 436–450. Springer, 2015.
6. James Brotherston. *Sequent Calculus Proof Systems for Inductive Definitions*. PhD thesis, University of Edinburgh, November 2006.
7. James Brotherston and Alex Simpson. Complete sequent calculi for induction and infinite descent. In *22nd IEEE Symposium on Logic in Computer Science (LICS 2007), 10-12 July 2007, Wroclaw, Poland, Proceedings*, pages 51–62. IEEE Computer Society, 2007.
8. James Brotherston and Alex Simpson. Sequent calculi for induction and infinite descent. *J. Log. Comput.*, 21(6):1177–1216, 2011.
9. Pierre Clairambault. Least and greatest fixpoints in game semantics. In *FOSSACS*, volume 5504 of *Lecture Notes in Computer Science*, pages 16–31. Springer, 2009.
10. Pierre-Louis Curien. Introduction to linear logic and ludics, part ii, 2006.

11. Vincent Danos. *Une application de la logique linéaire à l'étude des processus de normalisation (principalement du  $\lambda$ -calcul)*. Thèse de doctorat, Université Denis Diderot, Paris 7, 1990.
12. Vincent Danos and Laurent Regnier. The structure of multiplicatives. *Archive for Mathematical Logic*, 28:181–203, 1989.
13. Christian Dax, Martin Hofmann, and Martin Lange. A proof system for the linear time  $\mu$ -calculus. In S. Arun-Kumar and Naveen Garg, editors, *FSTTCS 2006: Foundations of Software Technology and Theoretical Computer Science, 26th International Conference, Kolkata, India, December 13-15, 2006, Proceedings*, volume 4337 of *Lecture Notes in Computer Science*, pages 273–284. Springer, 2006.
14. Amina Doumane. *On the infinitary proof theory of logics with fixed points. (Théorie de la démonstration infinitaire pour les logiques à points fixes)*. PhD thesis, Paris Diderot University, France, 2017.
15. Amina Doumane, David Baelde, Lucca Hirschi, and Alexis Saurin. Towards Completeness via Proof Search in the Linear Time  $\mu$ -Calculus. Accepted for publication at LICS, January 2016.
16. Jérôme Fortier and Luigi Santocanale. Cuts for circular proofs: semantics and cut-elimination. In Simona Ronchi Della Rocca, editor, *Computer Science Logic 2013 (CSL 2013), CSL 2013, September 2-5, 2013, Torino, Italy*, volume 23 of *LIPICs*, pages 248–262. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2013.
17. Jean-Yves Girard. Linear logic. *Theor. Comput. Sci.*, 50:1–102, 1987.
18. Roope Kaivola. A simple decision method for the linear time  $\mu$ -calculus. In Jörg Desel, editor, *Structures in Concurrency Theory, Workshops in Computing*, pages 190–204. Springer London, 1995.
19. Dexter Kozen. Results on the propositional  $\mu$ -calculus. *Theoretical Computer Science*, 27:333–354, 1983.
20. Paul-André Melliès. Higher-order parity automata. In *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20-23, 2017*, pages 1–12. IEEE Computer Society, 2017.
21. Raphaël Montelatici. Polarized proof nets with cycles and fixpoints semantics. In Martin Hofmann, editor, *Typed Lambda Calculi and Applications, 6th International Conference, TLCA 2003, Valencia, Spain, June 10-12, 2003, Proceedings.*, volume 2701 of *Lecture Notes in Computer Science*, pages 256–270. Springer, 2003.
22. Reiji Nakajima. Infinite normal forms for the  $\lambda$ -calculus. In C. Böhm, editor,  *$\lambda$ -Calculus and Computer Science Theory*, pages 62–82, Berlin, Heidelberg, 1975. Springer Berlin Heidelberg.
23. David Park. Fixpoint induction and proofs of program properties. *Machine intelligence*, 5(59-78):5–3, 1969.
24. Luigi Santocanale. A calculus of circular proofs and its categorical semantics. In Mogens Nielsen and Uffe Engberg, editors, *Foundations of Software Science and Computation Structures*, volume 2303 of *Lecture Notes in Computer Science*, pages 357–371. Springer, 2002.
25. Igor Walukiewicz. On completeness of the  $\mu$ -calculus. In *LICS*, pages 136–146. IEEE Computer Society, 1993.
26. Igor Walukiewicz. Completeness of Kozen's axiomatisation of the propositional  $\mu$ -calculus. In *Proceedings, 10th Annual IEEE Symposium on Logic in Computer Science, San Diego, California, USA, June 26-29, 1995*, pages 14–24. IEEE Computer Society, 1995.

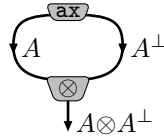
**Acknowledgement.** We are indebted to the anonymous reviewers for providing insightful comments which has immensely enhanced the presentation of the paper.

## A MLL proof nets à la Girard

**Definition 29.** A **proof structure** of MLL is a vertex-labelled and edge-labelled directed multigraph where the nodes are labelled by rules  $\{\mathbf{ax}, \mathbf{cut}, \otimes, \wp, \mathbf{c}\}$  and the edges are labelled by formulas such that:

- Each node labelled  $\mathbf{ax}$  has two outgoing edges labelled  $F$  and  $F^\perp$  for some  $F$  and no incoming edges.
- Each node labelled  $\mathbf{cut}$  has incoming edges labelled  $F$  and  $F^\perp$  for some  $F$  and no outgoing edges.
- Each node labelled  $\otimes$  has two incoming edges labelled  $F$  and  $G$  respectively (from left to right) and one outgoing edge labelled  $F \otimes G$ , for some  $F$  and  $G$ .
- Each node labelled  $\wp$  has two incoming edges labelled  $F$  and  $G$  respectively (from left to right) and one outgoing edge labelled  $F \wp G$ , for some  $F$  and  $G$ .
- Each node labelled  $\mathbf{c}$  has exactly one incoming edge and no outgoing edges.

A sequent proof,  $\pi$ , in MLL can be translated into a proof structure  $\mathbf{Deseq}(\pi)$  such that there is a bijection between internal nodes of  $\mathbf{Deseq}(\pi)$  and the rules of  $\pi$  which are not exchange rules. However, not every proof structure represents (or is the translation of) a MLL proof. For example,

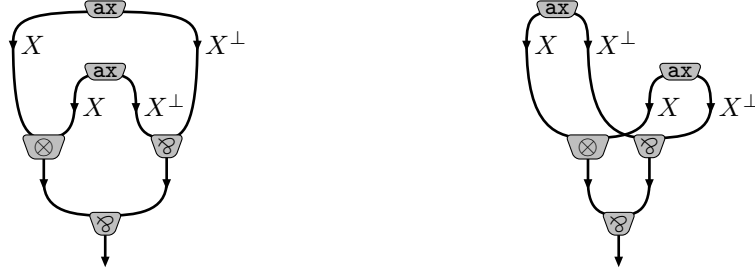


Hence we need to impose a correctness criterion to delineate a subset of “correct” proof structures which belong to the image of the translation  $\mathbf{Deseq}(\bullet)$ . There are several correctness criteria known in the literature. We will present the Danos-Regnier correctness criterion (DR-correctness).

**Definition 30.** Given a proof structure  $\mathcal{S}$ , let  $P$  be the set of its  $\wp$  nodes, a **switching** of  $\mathcal{S}$  is a function  $sw : P \rightarrow \{\text{left}, \text{right}\}$ . The **switching graph**  $\mathcal{S}^{sw}$  associated with  $sw$  is the labelled directed multigraph obtained from  $\mathcal{S}$  by modifying the target of the  $sw(p)$  premise of each node  $p \in P$  into a new  $\mathbf{c}$  node.

A proof structure  $\mathcal{S}$  is said to be **DR-correct** if for any switching  $sw$ ,  $\mathcal{S}^{sw}$  is acyclic and connected as an undirected graph. A correct proof structure is called a **proof net**.

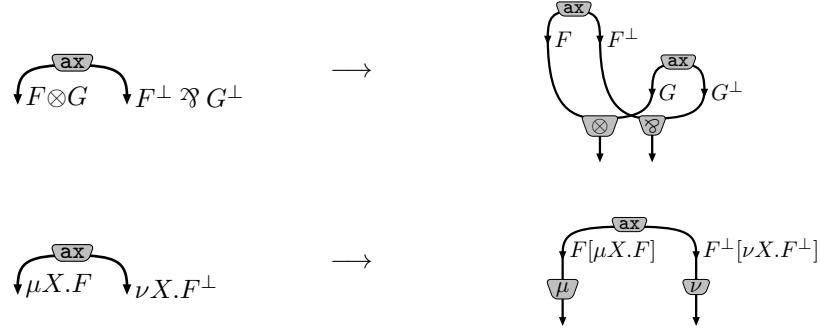
*Example 11.* Two proof nets of the sequent  $\vdash (X \otimes X) \wp (X^\perp \wp X^\perp)$  which are usually used to represent booleans.



## B Infinitary $\eta$ -expansion

In this section we will show that any infinet can be converted to an  $\eta^\infty$ -expanded infinet. First consider the morphism  $A \mapsto \mu X.X$  for all  $A \in \mathcal{A}$  where  $\mathcal{A}$  is the infinite set of atoms. Observe that now our syntax is atom-free.

**Definition 31.** *The **infinitary  $\eta$ -expansion** of an infinet,  $\mathcal{S}$ , is defined as the limit,  $\mathcal{S}^\eta$ , of the (co)inductive application of the following rewriting rules:*



We could have defined the infinitary  $\eta$ -expansion on the level of sequent proofs and indeed,  $\mathcal{S}^\eta$  would be isomorphic to the infinet of the infinitary  $\eta$ -expansion of  $\text{Deseq}(\mathcal{S})$ . On the level of  $\lambda$ -calculus, if an infinet corresponds to a Böhm tree, then its infinitary  $\eta$ -expansion would correspond to Nakajima trees [22].

## C Further Proofs and Clarifications

### C.1 Proof of Prop. 5

*Proof.* Let  $\mathcal{S}(= [\theta']I[\theta])$ . Fix a switching  $sw$  of  $\mathcal{S}$  and consequently obtain partition the  $\theta_{sw}$ . We will show by (co)induction on the structure of  $\pi$  that  $\theta^{sw}$  and  $\theta$  are orthogonal.



- Let  $u \in \Theta_1$  and  $v \in (\Theta_{\mathcal{S}_2}^{sw_2} \setminus \theta_G)$ . By the first case, there are unique paths,  $\gamma, \tau$  connecting  $u$  and  $\theta_F \uplus \theta_G$  and connecting  $v$  and  $\theta_F \uplus \theta_G$  respectively. Hence  $u$  and  $v$  are connected by  $\gamma\tau^{-1}$ . In fact, every simple path connecting  $u$  and  $v$  must pass through  $\theta_F \uplus \theta_G$ . Hence there is exactly one simple path between  $u$  and  $v$ . Symmetrically, for  $u \in \Theta_2$  and  $v \in (\Theta_{\mathcal{S}_1}^{sw_1} \setminus \theta_F)$ .

So,  $\mathcal{S}$  is OR-correct.

## C.2 Proof of Lemma. 1

*Proof.* Let  $\mathcal{S} = \Gamma[\Theta]$  be a DR-correct non-wellfounded proof structure. We are going to build a path leading to a splitting root.  $\Theta$  has be non-empty; so choose  $\theta \in \Theta$ . Choose  $(B, u) \in \theta$ .

If  $(B, u)$  is the root of a tree in  $\Gamma$  then it is splitting and we are done. Otherwise follow  $\delta((B, u))$  to reach a root node  $R_0$ . ( $\star$ ) If  $R$  is a  $\mathfrak{A}, \mu, \nu$  or a splitting  $\otimes$  node, we are done. Let  $R$  be non-splitting. By prop. 9, there is a correctness triple  $(P_0, \sigma_0, \sigma'_0)$ . Starting from  $P$ , we then follow  $\delta(P_0)$  and we reach a root node  $R_1$ . We can now continue as before from ( $\star$ ). If we stop, it means we have reached a splitting root and we are done. Otherwise the sequence of root nodes  $\{R_i\}$  visited is infinite. But since  $\Gamma$  is finite we must visit the same node twice; we will show that this is not possible.

Suppose  $R_i$  is the first root node to repeat *i.e.* there exists  $R_j$  for  $j < i$  such that  $R_j = R_i$  and for all  $0 \leq \ell < \ell' < i$ ,  $R_\ell \neq R_{\ell'}$ . Then  $\sigma_j, \delta(P_j), \sigma_{j+1}, \dots, \delta(P_{i-1})$  is a cyclic switching sequence. But then by, prop. 8, the correction graph of  $\mathcal{S}$  contains a cycle. This contradicts DR-correctness of  $\mathcal{S}$ .

## C.3 Proof of Lemma. 2

*Proof.* We will prove by contradiction. Suppose there are nodes which are never assigned a finite natural number by the time stamping algorithm. Let  $t = (B_i, u)$  be the minimal node like that *i.e.* for all such other nodes  $(B_j, u')$ , we have  $|u| \leq |u'|$ . ( $\star$ ) Then, by construction, after finite iterations of the sequentialization process, it becomes a conclusion of a proof structure,  $\mathcal{S}'$  that needs to be sequentialized. Hence  $t$  is a non-splitting root in  $\mathcal{S}$ . By prop. 9, there is a correctness pair  $(\sigma, \sigma', P)$ . Consider the set of all such  $\mathfrak{A}$  formulas,  $\mathcal{P} = \{p_i\}_{i \in I}$ . Suppose  $I$  is infinite. Since  $\Gamma$  is finite, there is a partial syntax tree,  $B_i^{U_i}$ , in  $\Gamma$  such that there is set  $V \subseteq \overline{U_i}$  such that  $V' = \{(B_i, v) \mid v \in V\} \cap \mathcal{P}$  is infinite. Let  $f$  be a wait function of  $\mathcal{S}$ . Then,  $f(|u|) > |v|$  for all  $v \in V'$  which is absurd. Hence  $I$  is finite.

Observe that for every  $i \in I$ ,  $p_i <_{D(\mathcal{S})} t$ . Now, choose  $p \in \mathcal{P}$  and follow  $\delta(P_i)$  to the conclusion,  $c$ , of  $p$ . If  $c$  splitting, then after finite iterations of the sequentialization process, we have a proof structure,  $\mathcal{S}''$  that needs to be sequentialized such there is conclusion in  $\mathcal{S}''$  that is a premise of  $c$ . We can continue like this until the conclusion of  $p$  is a non-splitting  $\otimes$  formula  $t'$  (otherwise  $p$  will become eventually splitting and we choose another  $p' \in \mathcal{P}$ ). Observe that  $t' <_{D(\mathcal{S})} p$ . We continue as before from ( $\star$ ). But then we have an infinite descending chain



in  $<_{D(\mathcal{S})}$ . This contradicts the lock-freeness of  $\mathcal{S}$ . So, after finitely many iterations every  $p \in \mathcal{P}$  will be sequentialized and hence  $t$  will be splitting and will be assigned a finite time-stamp.

#### C.4 Proof of Prop. 10

*Proof.* The if part follows from the observation that every rule  $\pi_1$  is ultimately executed in  $\pi_2$  on the same formulas as  $\pi_1$  (possibly preponed or postposed by finitely many steps) and the fact that permutation does not change the axiom partition.

For the only-if part, let  $\pi_i = [\Theta'_i] \Gamma_i [\Theta_i]$ .  $[\Theta'_1] \Gamma_1 = [\Theta'_2] \Gamma_2$  implies that same rules are applied on same formulas in  $\pi_1, \pi_2$ .  $[\Theta_1] = [\Theta_2]$  implies that same formulas are the premises of the tensor and cut rules in  $\pi_1, \pi_2$ . Hence  $\pi_1 \sim^\infty \pi_2$ .

#### C.5 Proof of Prop. 12

*Proof.* We know that  $\mathcal{S}_0$  is a valid infinet. Suppose  $\mathcal{S}_1$  is not an infinet. Then, the correction graph of  $\mathcal{S}_1$  contains a cycle involving  $\bar{\theta}$  s.t.  $\bar{\theta} \in \Theta_1 \setminus \Theta_0$ . Let  $\bar{\theta} = \theta \cup \theta' \setminus \{u, u^\perp\}$  where  $\theta, \theta' \in \Theta_0$ .

Notice that  $\theta, \theta'$  are validating since  $\mathcal{S}_0$  is valid. If  $u$  is the validating thread in  $\theta$ , then  $u^\perp$  cannot be the validating thread in  $\theta'$ . Hence there is a validating thread in  $\theta'$  that will be retained in  $\theta \cup \theta' \setminus \{u, u^\perp\}$ . So, it is non-empty and validating.

Now, a path between  $\bar{\theta}$  and  $\bar{\theta}$  in the correction graph of  $\mathcal{S}_1$  is a path between  $\theta$  and  $\theta'$  in the correction graph of  $\mathcal{S}_0$ . But there is a switching of  $\mathcal{S}_0$  s.t. there is another distinct path between  $\theta$  and  $\theta'$  which goes through the partitions corresponding to the connected components containing  $u, u^\perp$  resp. Hence there is a cycle in the correction graph of  $\mathcal{S}_0$ . Contradiction! Hence  $\mathcal{S}_1$  is OR-correct and valid.

If there is a deadlock in  $\mathcal{S}_1$  then there is a deadlock in  $\mathcal{S}_0$ , so lock-freeness of  $\mathcal{S}_1$  follows from the lock-freeness of  $\mathcal{S}_0$ . Therefore,  $\mathcal{S}_1$  is an infinet.