



HAL
open science

Eliminating infinitely many cuts in non-wellfounded MLL proof-nets

Abhishek De, Luc Pellissier, Alexis Saurin

► **To cite this version:**

Abhishek De, Luc Pellissier, Alexis Saurin. Eliminating infinitely many cuts in non-wellfounded MLL proof-nets. 2021. hal-03235591

HAL Id: hal-03235591

<https://hal.science/hal-03235591v1>

Preprint submitted on 25 May 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.

Eliminating infinitely many cuts in non-wellfounded μ MLL proof-nets

ABHISHEK DE, Université de Paris, IRIF, CNRS, France

LUC PELLISSIER, LACL, Université Paris Est Créteil, France

ALEXIS SAURIN, Université de Paris, IRIF, CNRS, France

Non-wellfounded and circular proofs have been recognised over the past decade as a valuable tool to study logics expressing (co)inductive properties, *e.g.* μ -calculus. Such proofs are non-wellfounded sequent derivations together with a global validity condition expressed in terms of *progressing threads*. While the cut-free fragment of circular proofs is satisfactory, cuts are poorly treated and the non-canonicity of sequent proofs becomes a major issue in the non-wellfounded setting. The present paper develops the theory of *infinets* – proof-nets for non-wellfounded proofs – allowing *infinets* with infinitely many cuts therefore solving a crucial shortcoming of the previous work [21], characterising sequentialisation and proving a cut-elimination theorem. To that end, we provide an alternate cut reduction for non-wellfounded sequent calculus.

ACM Reference Format:

Abhishek De, Luc Pellissier, and Alexis Saurin. 2021. Eliminating infinitely many cuts in non-wellfounded μ MLL proof-nets. 1, 1 (May 2021), 41 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

1 INTRODUCTION

Fixed points. Fixed point theory occurs in just about every field of computer science, including program analysis [31], games [14, 36], automata theory [32, 37] and especially in programming language theory because the semantics of recursion can be described by fixed points [41]. In computational logic, inductive and coinductive reasoning can be modelled in two paradigms: Martin-Löf’s inductive predicates [11, 13, 33] and fixed-point logics [3, 5, 8, 26, 40]. Non-wellfounded and circular proof systems [3, 7, 25, 26, 40] have received much attention in recent years: such proof systems allow finitely-branching possibly infinitely deep derivation trees. However, when considering all possible non-wellfounded derivations (*a.k.a. pre-proofs*), the resulting system is inconsistent: one can derive any sequent (see fig. 4a) and thus needs to impose a global validity criterion to sieve the logically valid proofs from the unsound ones. Typically, it requires that every infinite branch is supported by some thread tracing some formula in a bottom-up manner and witnessing infinitely many progress points of a coinductive property. Furthermore, in this non-wellfounded setting, termination of the cut-elimination procedure shall be replaced by *productivity* *i.e.* that arbitrarily large prefixes of the result can be computed in a finite number of steps. The aforementioned validity condition is a sufficient, but non-necessary, condition for productivity of cut-elimination. A particularly useful fragment of non-wellfounded derivations is that of infinite but regular derivation trees, known as *circular*, or *cyclic*, derivations.

Authors’ addresses: Abhishek De, Université de Paris, IRIF, CNRS, F-75013, Paris, France, abhishek.de@irif.fr; Luc Pellissier, LACL, Université Paris Est Créteil, F-94010, Créteil, France, luc.pellissier@lacl.fr; Alexis Saurin, Université de Paris, IRIF, CNRS, F-75013, Paris, France, alexis.saurin@irif.fr.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2021 Association for Computing Machinery.

Manuscript submitted to ACM

Manuscript submitted to ACM

1

```

CoInductive Stream := Cons : nat → Stream → Stream.
CoFixpoint f0 (n : nat) : Stream := Cons n (f0 (n+1)).
CoFixpoint f1 (n : nat) : Stream := let s := f1 (n+1) in
  Cons n (match s with Cons h t ⇒ Cons h t end).
CoFixpoint f2 (n : nat) : Stream := let s := f2 (n+1) in
  (match s with Cons h t ⇒ Cons n (Cons h t) end).
CoFixpoint f3 (n : nat) : Stream := let s := f3 (n+1) in
  (match s with Cons h t ⇒ Cons h (Cons n t) end).

```

Fig. 1. Some productive and non-productive definitions

Productivity, from a proofs/programs perspective. Non-wellfounded proofs are currently handled in several proof-systems, such as Coq, whose type checker checks for a guard condition to ensure productivity. In Figure 1, we show several Coq coinductive definitions:

- f_0 is the only valid Coq coinductive definition; $(f_0 \ n)$ computes the streams of naturals starting from n .
- f_1 is a productive term, even though it is rejected by Coq type-checker as it fails to pass its guard condition. It computes the same stream as f_0 .
- f_2 is not productive only through reductions, and one needs to introduce a commutation rule: $\text{match } e_1 \text{ with } p \Rightarrow \text{Cons } (h, t) \rightsquigarrow \text{Cons } (h, \text{match } e_1 \text{ with } p \Rightarrow t)$ if pattern p does not occur free in h (and symmetrically with t).
- f_3 is not productive: producing the first element of $(f_3 \ n)$ requires to already have produced the first element of each stream $(f_3 \ k)$ for $k > n$.

Proof theory of fixed point logics can tell us about the computational behaviour of these programs: following the guiding principles of the Curry-Howard correspondence, notice first that (co)inductive types can be encoded as formulas of the μ -calculus. In μ MALL (linear logic extended with least and greatest fixed points), one can represent the type of nats and streams as $N = \mu X. 1 \oplus X$ and $S = \nu Y. N \otimes Y$.¹ We can represent natural numbers as well as the successor function as the cut-free proofs respectively $\pi_k, k \in \mathbb{N}$ of type $\vdash N$ and π_{succ} of type $N \vdash N$ presented in fig. 2. Also, naturals are duplicable resources *i.e.* there is a proof π_{dup} of $N \vdash N \otimes N$ such that, when cut with π_k , returns a pair of (proofs representing) natural numbers.

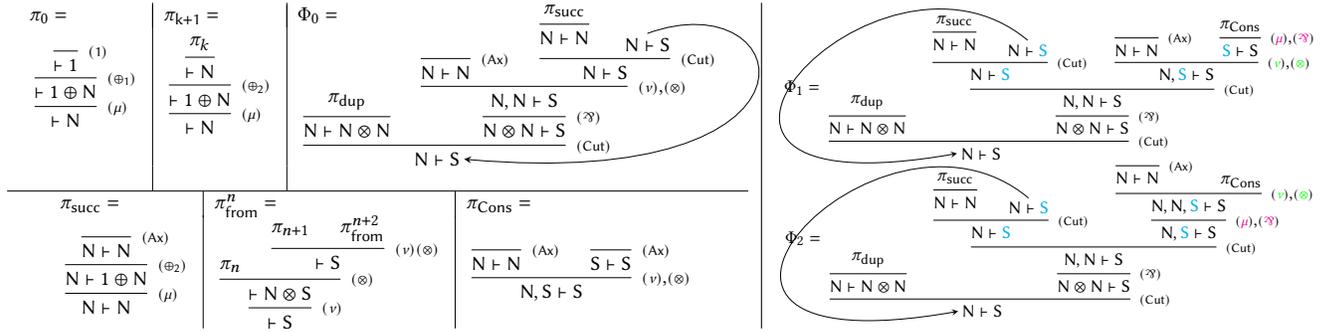
$$\pi_{\text{dup}} = \frac{\frac{\frac{\pi_0}{\vdash N} \quad \frac{\pi_0}{\vdash N}}{\vdash N \otimes N} (\otimes) \quad \frac{\frac{\frac{\pi_{\text{succ}}}{N \vdash N} \quad \frac{\pi_{\text{succ}}}{N \vdash N}}{N, N \vdash N \otimes N} (\otimes) \quad \frac{\frac{\frac{\pi_{\text{succ}}}{N \vdash N} \quad \frac{\pi_{\text{succ}}}{N \vdash N}}{N \otimes N \vdash N \otimes N} (\otimes)}{N \vdash N \otimes N} (\text{Cut})}{\frac{1 \oplus N \vdash N \otimes N}{N \vdash N \otimes N} (\&)}}{\frac{1 \oplus N \vdash N \otimes N}{N \vdash N \otimes N} (\vee)} (\perp)$$

The following proposition, the proof of which is a routine check, characterizes the properties of π_{succ} and π_{dup} :

PROPOSITION 1. *For any natural number k , we have:*

$$\frac{\pi_k \quad \pi_{\text{succ}}}{\vdash N} (\text{Cut}) \longrightarrow_c^* \pi_{k+1}$$

¹While we use a one-sided presentation of the sequent calculus in the technical developments of this paper, as common with classical LL, we show the encoding of the above programs using two-sided sequents for clarity: $F_1, \dots, F_n \vdash G_1, \dots, G_m$ should be read as $\vdash F_1^+, \dots, F_n^+, G_1, \dots, G_m$ as usual and left inference rules are written via the right rule of their dual connective.

Fig. 2. μ MALL[∞] encodings of nat, of basics functions on nats and streams and of f_0 , f_1 and f_2 from fig. 1

and

$$\frac{\pi_k \quad \pi_{\text{dup}}}{\vdash N \otimes N} \text{ (Cut)} \longrightarrow_c^* \frac{\pi_k \quad \pi_k}{\vdash N \otimes N} \text{ (}\otimes\text{)}$$

Similarly, one can represent streams of nats as cut-free *non-wellfounded* derivations of $\vdash S$ (e.g. in fig. 2 π_{from}^n represents the streams of successive nats starting from n) and functions on streams (e.g. π_{Cons} represents the cons on streams): in those derivations, we may have infinitely deep branches as for π_{from}^n whose right-most branch is infinite.

One can encode the above coinductive programs as *circular* derivations² of $N \vdash S$, as shown in fig. 2: Φ_0 , Φ_1 and Φ_2 represent respectively f_0 , f_1 and f_2 (the encoding of f_3 is shown in Appendix). One can, and will in the following, consider cut-reduction relation \longrightarrow_c over those non-wellfounded proofs. For instance, the proof obtained by cutting Φ_0 with π_n will induce an infinite cut-reduction sequence *converging* to π_{from}^n ; the same happens when reducing a cut between Φ_1 and π_n : those are productive. On the other hand, if π' is obtained by cutting Φ_2 with π_n every derivation that is reached by reduction sequence from π' will have a cut as its last inference: cut cannot be eliminated from that proof, it is a non-productive computation.³ Interestingly, the difference between Φ_1 and Φ_2 is limited to the relative order of the (v, \otimes) inferences and the (μ, \circledast) depicted in green and red on fig. 2 and that difference, altogether with non-wellfoundedness of the sequent derivations, will make the difference between a productive and a non-productive cut-elimination. A simpler example of the same phenomenon will be given in fig. 4d and 4e and discussed below.

Proof-nets. As we have seen, some seemingly irrelevant differences (the relative order of the application of rules) induce widely varying behaviour in μ MALL sequent calculus. This phenomenon is related to the fact that the sequent calculus for LL is non-canonical: a LL proof may be reduced to two cut-free proofs π_1 and π_2 which are different but guaranteed to be equal up to irrelevant permutations of inference rules⁴. In other words, the permutations are denotationally trivial i.e. $\llbracket \pi_1 \rrbracket = \llbracket \pi_2 \rrbracket$ in any semantics. Proof-nets [29] were devised to overcome this sequentality. A proof-net can be seen as a graph whose nodes are inference rules, which are thus not ordered, and consequently less sequential than sequent calculus proofs. As they are canonical, proof-nets are well-suited to represent computation.

²Notice that derivations for the Φ_{iS} contain back-edges, denoting the fact that the derivation tree is infinite but regular.

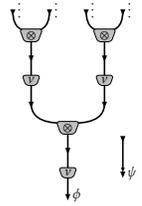
³Indeed, to produce the first element of the stream the green (v, \otimes) inferences, one first needs to make the red (μ, \circledast) inferences interact over the cut but this requires first partially eliminating the cut of the derivation at the source of the back-edge, which is essentially the proof we started with: this infinite chain of dependencies causes the non-productivity.

⁴Normalisation for LL sits thus in the middle between classical sequent calculus LK – in which a proof (Lafont’s critical pair) can be reduced to any two proofs of the same sequent – and natural deduction [28, 38] or λ -calculus [16] normalisation which are confluent.

$$\begin{aligned}
\Phi_0 = & \frac{\frac{\frac{\pi_{\text{succ}}}{N \vdash N} \quad N \vdash S}{N \vdash S} \text{ (Cut)} \quad \frac{\overline{N \vdash N} \text{ (Ax)} \quad \frac{N \vdash S}{N \vdash S} \text{ (Cut)}}{N, N \vdash S} \text{ (v), (\otimes)}}{\frac{\pi_{\text{dup}}}{N \vdash N \otimes N} \quad \frac{N, N \vdash S}{N \otimes N \vdash S} \text{ (\textcircled{?})}}{N \vdash S} \text{ (Cut)}} \\
\Phi_1 = & \frac{\frac{\frac{\pi_{\text{succ}}}{N \vdash N} \quad N \vdash S}{N \vdash S} \text{ (Cut)} \quad \frac{\overline{N \vdash N} \text{ (Ax)} \quad \frac{\overline{S \vdash S} \text{ (Ax)}}{N, S \vdash S} \text{ (\mu), (\textcircled{?})}}{N, S \vdash S} \text{ (v), (\otimes)}}{\frac{\pi_{\text{dup}}}{N \vdash N \otimes N} \quad \frac{N, N \vdash S}{N \otimes N \vdash S} \text{ (\textcircled{?})}}{N \vdash S} \text{ (Cut)}} \\
\Phi_2 = & \frac{\frac{\frac{\pi_{\text{succ}}}{N \vdash N} \quad N \vdash S}{N \vdash S} \text{ (Cut)} \quad \frac{\overline{N \vdash N} \text{ (Ax)} \quad \frac{\overline{S \vdash S} \text{ (Ax)}}{N, S \vdash S} \text{ (\mu), (\textcircled{?})}}{N, S \vdash S} \text{ (v), (\otimes)}}{\frac{\pi_{\text{dup}}}{N \vdash N \otimes N} \quad \frac{N, N \vdash S}{N \otimes N \vdash S} \text{ (\textcircled{?})}}{N \vdash S} \text{ (Cut)}} \\
\Phi_3 = & \frac{\frac{\frac{\pi_{\text{succ}}}{N \vdash N} \quad N \vdash S}{N \vdash S} \text{ (Cut)} \quad \frac{\overline{N \vdash N} \text{ (Ax)} \quad \frac{\overline{S \vdash S} \text{ (Ax)}}{N, S \vdash S} \text{ (\mu), (\textcircled{?})}}{N, S \vdash S} \text{ (v), (\otimes)}}{\frac{\pi_{\text{dup}}}{N \vdash N \otimes N} \quad \frac{N, N \vdash S}{N \otimes N \vdash S} \text{ (\textcircled{?})}}{N \vdash S} \text{ (Cut)}}
\end{aligned}$$

Fig. 3. μMALL^∞ encodings of the Coq functions of Figure 1

Infinets. In [21], the authors defined *infinets*, canonical objects that capture exactly the equivalence classes of pre-proofs under the equivalence by (possibly infinite) permutation of inferences (*a.k.a.* permutative equivalence). Compared to MLL, more structure is needed in order to have suitable non-wellfounded proof structures for μMLL^∞ . Consider the two proofs in fig. 4b and fig. 4c⁵. They are not permutatively equivalent: no permutation will change the contents of the premises of a tensor. However, naively desequentialising them by forgetting the order of inferences and keeping only the subformula ordering ends up in the same proof structure (drawn on the left. The formula ψ is left hanging as it is not connected to any cell of the infinet) which means the proof-net equivalence would be coarser than permutation equivalence. Indeed, the fact that ψ resides with either the rightmost or the leftmost infinite branch is lost in translation: in order to be faithful, more structure in the form of “infinite axioms” is present in *infinets*. As usual axioms encapsulate

⁵Omitting the indices for the time being.

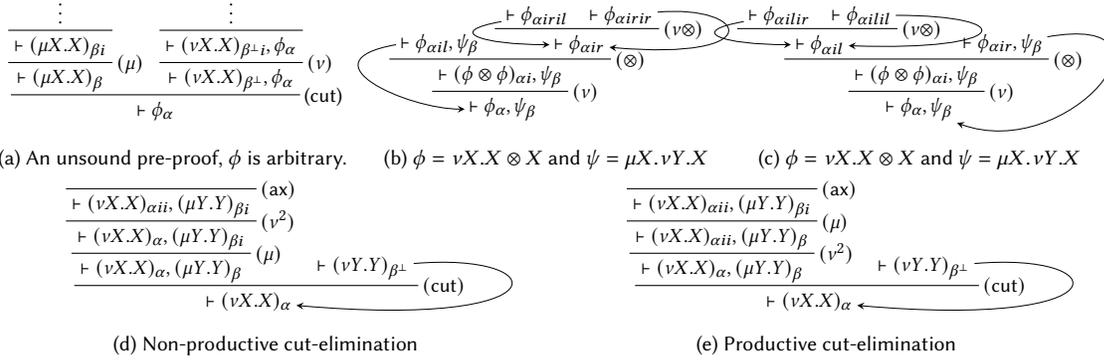


Fig. 4. Non-wellfounded derivations. The back-edges between two sequents with the same underlying formulas are a way to represent regular proofs: the derivation above the pointing sequent is equal to the one above the pointed one, up to address substitution. α and β are arbitrary addresses.

the information which formulas end up in which leaf of the proof tree, infinite axioms encapsulate the information which formulas end up in which infinite branch of the non-wellfounded proof tree.

The non-canonicity of sequent calculus manifests itself more critically in the non-wellfounded setting: productivity of cut-elimination is not preserved by permutative equivalence [6], as already noticed with fig. 2. The two pre-proofs in figs. 4d and 4e witness the same phenomenon with simpler proof objects (they use neither additive nor multiplicative connectives, only fixed points): they are permutatively equivalent but cut-elimination is productive only in the latter (fig. 7). However they have the same infinet on which the cut-reduction rules that we propose can be applied (fig. 17). Consequently, we believe that infinets are the proper framework for dealing with unrestricted cuts and more expressive validity conditions (such as bouncing-validity). Understanding the impact of those permutations and how to quotient them properly is a deep motivation for this work and for our investigation of proof-nets for non-wellfounded proofs: we aim at benefiting from the canonicity of proof-nets to improve the dynamics of non-wellfounded derivation *wrt.* cut-elimination.

Infinitely many cuts. The handling of the cuts in [21] has been rudimentary: only finitely many cuts are considered and cut-elimination is basically interpreted as an infinitary abstract rewriting system with a metric: at first, one guesses the normal form (*a.k.a.* big-step) then, a transfinite reduction sequence of small steps is shown to converge to the big-step in the limit. To guess the limit, one has to sacrifice some structure *viz.* η -expand all axioms rendering the calculus without axioms. This is a strong limitation when one see that examples as simple as Φ_0, Φ_1 and Φ_2 contain infinitely many cuts. The present paper provides a full treatment of cuts and axioms for non-wellfounded proofs.

Current trends and related works. Several strands of research around non-wellfounded and circular proofs are related to the present work in various ways:

Expressing theories with circular proofs is a fruitful research direction [19, 20, 23]. Circular proofs also help understanding the meta-theory of traditional fixed points logics such as the linear-time or modal μ -calculi [3, 24, 25].

Validity criteria are subject of active investigations. For instance, Baelde et al.[6] introduce *bouncing-validity*: threads can bounce on axioms and cuts describing paths in proofs and validate more circular derivations.

The denotational semantics of circular proofs is yet to be fully understood and is a challenging direction [26, 30, 40].

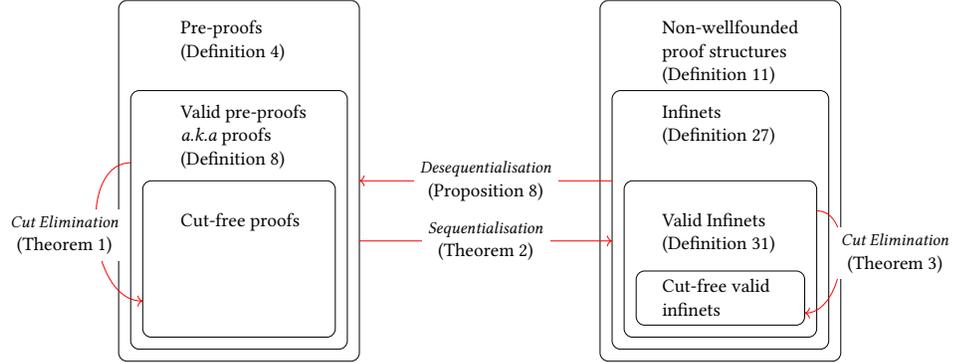


Fig. 5. Schemata of the contributions

Interactive and automated theorem proving are impacted by circular proofs – which use a form of implicit induction– [4, 15, 27, 34, 39], be it for building automated proofs of (co)inductive statements or for easing the manipulation of coinduction in proof assistants such as Coq. This active research topic can be impacted by advances on the fine-grained structure of circular proofs [12, 22] or approaches such as copatterns [2].

Alternative styles for handling non-wellfounded proofs are actively researched to provide solutions to some of the limitations of sequent proofs. In addition to De and Saurin’s line of work on proof-nets [21], an interesting direction is that of Das on designing deep-inference proof systems for circular reasoning [18].

The present work – devising proof-nets for non-wellfounded proof theory – nestles itself naturally in the last category, but is also related to other approaches: (i) proof-nets are closer to semantics than sequent proofs and infinites can potentially impact the understanding of the denotational invariants of non-wellfounded proofs and the design of their denotational semantics. (ii) the validity criteria could be improved as the bouncing-validity condition will strongly benefit from less sequential proof structures.

Organisation of the contributions. This work strengthens the definition for non-wellfounded proof structures—and their correctness criterion—to accommodate infinitely many cuts: in this situation, some infinite axioms are only virtually present, and made explicit through cut-elimination. Our main contribution is the cut-elimination result for infinites with atoms and infinitely many cuts by reconciling the locality of the big-step and non-locality of the small-step. To prove that result we need to provide an alternate cut reduction system of μMLL^∞ sequent calculus. The contributions are summarised in fig. 5.

In section 2, we recall the necessary background on fixed-point logics and non-wellfounded proof theory. In section 3, we provide a new cut-elimination result for μMLL^∞ sequent calculus which is an alternative to Baelde et al [6, 7] cut-elimination and is more suited for proof-nets. In section 4, we informally discuss the necessary structure to be added to MLL proof-nets. In section 5, we formalize this intuition and define non-wellfounded proof structures. We state the correctness criterion in section 6, which leads us to develop and adapt the theory of kingdoms to non-wellfounded structures, an additional contribution of the work. This allows us to provide a sequentialisation procedure extending that of [21]. We introduce a new cut reduction system in section 7 and establish the cut-elimination theorem by showing productivity of cut-reduction for valid infinites. Section 8 concludes with future directions.

2 BACKGROUND

μ MALL $^\infty$, the non-wellfounded extension of MALL, the multiplicative additive fragment of linear logic, with least and greatest fixed points operators, was introduced in [7, 25]. In this paper, we only consider the unit-free multiplicative fragment which we call μ MLL $^\infty$. In this section, we recall some basic definitions.

DEFINITION 1. Given two disjoint infinite sets of atoms $\mathcal{A} = \{A, B, \dots\}$, and of propositional variables $\mathcal{V} = \{X, Y, \dots\}$, μ MLL $^\infty$ **pre-formulas** are given by the following grammar (where $A \in \mathcal{A}, X \in \mathcal{V}$):

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

μ, ν bind the variable X in ϕ . Free and bound variables, as well as capture-avoiding substitution are defined as usual. The subformula ordering is denoted \leq . A closed pre-formula (i.e. no free variables), is called a **formula**.

We define negation, $(\bullet)^\perp$, as a meta-operation on pre-formulas (with $X^\perp = X$) and will use it only on formulas. As it is not part of the syntax, we do not need any positivity condition on the fixed-point expressions. As expected, the least and greatest fixed point are the dual of each other.

The system is classical, hence, it is enough to consider a one-sided proof system. However, in order to keep track of progressing *threads* (i.e. *validity*) and also while translating into proof nets, it is useful to distinguish occurrences of the same formula within a sequent. A μ MLL $^\infty$ **sequent** is an expression $\vdash \Delta$ where Δ is a finite set of pairwise *disjoint formula occurrences*. We will now define these terms introduced.

DEFINITION 2. An **(in)finite address** is a (in)finite word in $\{l, r, i\}^{\infty 6}$. Negation extends over addresses as the morphism satisfying $l^\perp = r, r^\perp = l$, and $i^\perp = i$. We call α' a **sub-address** of α if α is a prefix of α' . α and β are said to be **disjoint** if the greatest common prefix of α and β is not equal to α or β .

DEFINITION 3. A **formula occurrence** (denoted by F, G, \dots) is given by a formula ϕ and a finite address α , written ϕ_α . Let $\text{addr}(\phi_\alpha) = \alpha$. Operations on formulas extend to occurrences: $\phi_\alpha^\perp = \phi_{\alpha^\perp}^\perp$; for $\star \in \{\wp, \otimes\}$, $F \star G = (\phi \star \psi)_\alpha$ if $F = \phi_\alpha$ and $G = \psi_\alpha$; for $\sigma \in \{\mu, \nu\}$, $\sigma X.F = (\sigma X.\phi)_\alpha$ if $F = \phi_\alpha$. Substitution of occurrences forgets addresses i.e. $(\phi_\alpha)[\psi_\beta/X] = (\phi[\psi/X])_\alpha$. We say that occurrences are **disjoint** when their addresses are. Given F, G two occurrences, G is called a **suboccurrence** of F (written $G \sqsubseteq F$) if $\text{addr}(G)$ is a subaddress of $\text{addr}(F)$. Finally, $[\bullet]$ denotes the address erasure operation on occurrences.

We are now ready to define the derivation trees of μ MLL $^\infty$.

DEFINITION 4. A **pre-proof** of μ MLL $^\infty$ is a possibly infinite tree generated from the inferences of unit-free multiplicative linear logic and the following rules:

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

Given a pre-proof, π , $\text{addr}(\pi) \subseteq \{l, r, i\}^{\infty}$ is a set of addresses s.t. a finite address $\alpha \in \text{addr}(\pi)$ iff there is an occurrence F , in an axiom in π with $\text{addr}(F) = \alpha$ and an infinite address $\alpha \in \text{addr}(\pi)$ iff all the strict prefixes of α are addresses of occurrences appearing in π .

EXAMPLE 1. The three trees in fig. 4 are pre-proofs. If we call π the proof in fig. 4b, we have that $\text{addr}(\pi) = \alpha.(i(l+r))^\omega \cup \{\beta\}$.

⁶For any finite set of alphabets, A, A^∞ denotes the set of finite and infinite words made of letters from A .

We will directly present examples on formula, leaving the reader to add the addresses in a coherent way.

Infinitary proof systems depart peculiarly from their wellfounded counterparts: in spite of the rules being locally sound, it is possible to derive any sequent, as in Figure 4a. We impose a global validity criterion on pre-proofs. Valid pre-proofs are simply called **proofs**.

DEFINITION 5. Let $\gamma = (s_i)_{i \in \omega}$ be an infinite branch of a pre-proof. A **thread** of γ is a sequence $\tau = \{F_i\}_{i \in \omega}$ such that there exists $j \geq 0$ such that for all $i < \omega$, we have $F_i \in s_{i+j}$ and either F_i is suboccurrence of F_{i+1} or $F_i = F_{i+1}$. We denote the sequence of formulas $\{[F_i]\}_{i \in I}$ by $[\tau]$.

PROPOSITION 2 ([25], PG. 53). The \leq ordering is a total order on the set $\text{Inf}([\tau])$ where $\text{Inf}(\tau)$ is the set of formulas occurring infinitely often in τ .

DEFINITION 6. A thread is said to be **straight** if it is not ultimately constant. A straight thread, τ , is said to be **valid** if $\min(\text{Inf}([\tau]))$ is a v -formula.

DEFINITION 7. An infinite branch of a pre-proof is called **real** if it has at least one straight thread and **virtual** otherwise. It is **valid** if it has a valid thread.

DEFINITION 8. Let π be a μMLL^∞ pre-proof. It is **straight-thread valid** if all its infinite branches are valid.

REMARK 1. Observe that a proof does not have virtual branches. Furthermore, if a pre-proof has virtual branches, then it has infinitely many cuts.

EXAMPLE 2. Consider the pre-proof (say π) in fig. 4b. The left-most branch contains two threads: one following the formula ϕ , the other the formula ψ . The first one is straight and valid, the second is not: accordingly, this branch is supported by a valid thread. As it is also the case of the other infinite branches, π is a proof.

3 CUT-ELIMINATION IN μMLL^∞ SEQUENT CALCULUS

In finitary proof theory, cut elimination may proceed by reducing topmost cuts but there is no such thing, in general, as a topmost cut in non-wellfounded proof-theory. Previous cut-elimination results relied on reduction of bottom-most cuts [6, 7, 25, 26] using a generalized cut-rule, the multicut, which abstracts over a finite subtree made of cut (and axiom) rules. In those approaches, when two cuts are immediately above one another, they are merged instead of being permuted. The following is an example of a multicut rule: the red lines indicate the context and the blue lines indicate two cuts that have been merged.

$$\frac{\frac{\frac{\vdash F', G}{\vdash F, G} \quad \frac{\vdash G^\perp, H}{\vdash G^\perp, H}}{\vdash F, G, H} \quad \frac{\vdash H^\perp, K'}{\vdash H^\perp, K'}}{\vdash F, K'} \text{ (mcut)}$$

A less sequential approach to cut-elimination. While the multicut brings uniformity in the treatment of cut-elimination in sequent calculus, it is not well-suited for our purpose of developing a canonical and parallel treatment of cuts in non-wellfounded proof systems. It is indeed better-suited to use the usual cut-rule to draw a comparison between cut-reductions in sequent systems and in proof-nets, as we will do in the last sections of the paper. To serve this purpose, we develop here an alternative approach to cut-elimination for non-wellfounded proof which avoids the use of the multicut but on the standard cut instead and we will prove a new cut-elimination result in this case. We shall simply retain however a degenerated case of the multi-cut, the *unary* case, used to perform lazily the cut-axiom reduction and relocation of addresses. Indeed, as we work with explicit occurrences, the cut/ax case is as follows:

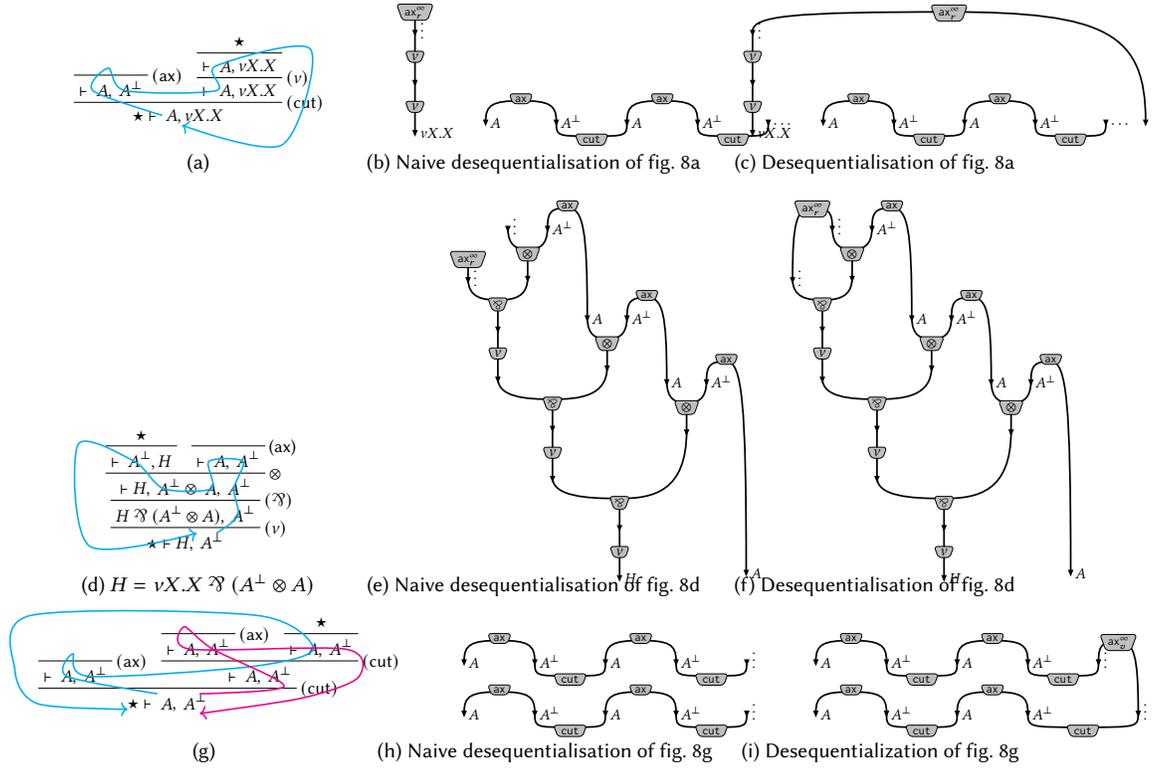


Fig. 8. Naive and faithful desequentialisations of μMLL^∞ simple pre-proofs. Back-edges are depicted using pointers (\star). Red and blue curves indicate trips.

Consider the proof structure in fig. 8b, the naive desequentialisation of the pre-proof in fig. 8a. There is an infinite axiom “above” the undirected ray of v nodes (which corresponds to a straight thread) and there is an undirected ray, ρ , of alternating axioms and cuts. Observe that every A introduced by a cut resides with the straight thread in the only infinite branch in fig. 8a: this is lost in translation. Since infinite axioms capture the invariant of an infinite branch, ρ should be included in the infinite axiom in a correct desequentialisation as in fig. 8c. Paths like ρ alternating through axioms and cuts are called *visitable paths*.

Visitable paths can be formed using tensor nodes as well: a similar situation can be reproduced using tensors and a fixed point formula. Consider the proof structure in fig. 8e, the naive desequentialisation of the pre-proof in fig. 8b. Here as well the visitable path of alternating axioms and tensors (in red) should be included in the infinite axiom above the undirected ray of v nodes.

When translating pre-proofs into proof structures, one needs to recognise these paths on pre-proofs. We therefore introduce the notion of trips.

DEFINITION 10. Given a pre-proof π , a **trip** starting from F_1 is a sequence $\tau = \{(s_i, F_i, d_i)\}_{i \in \omega}$ where s_i is a sequent in π , $F_i \in s_i$ and $d_i \in \{\uparrow, \downarrow\}$ such that $d_1 = \uparrow$ and for every $i < \omega$ exactly one of the following holds:

- $d_i = d_{i+1} = \uparrow$, s_{i+1} is a premise of s_i and $F_{i+1} \sqsubseteq F_i$.
- $d_i = d_{i+1} = \downarrow$, s_i is a premise of s_{i+1} and $F_i \sqsubseteq F_{i+1}$.

- $d_i = \uparrow, d_{i+1} = \downarrow, s_i = s_{i+1} = \{F_i, F_{i+1}\}$ and s_i is conclusion of a (ax) rule.
- $d_i = \uparrow, d_{i+1} = \downarrow, s_i = s_{i+1}$ and there exist straight threads τ and τ' that start from F_i and F_{i+1} respectively and belong to the same infinite branch.
- $d_i = \downarrow, d_{i+1} = \uparrow, s_i$ and s_{i+1} are the premises of a (cut) rule on F_i and F_{i+1} .
- $d_i = \downarrow, d_{i+1} = \uparrow, s_i$ and s_{i+1} are the premises of a (\otimes) rule on F_i and F_{i+1} .

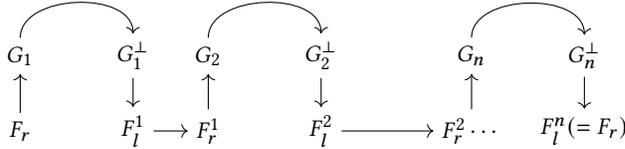
Further, τ satisfies the following conditions:

- For every $i, j < \omega$, there does not exist a sequent, s , in π such that $F_i \wp F_j \in s$.
- There are infinitely many terms of the form (s, F, \downarrow) .

Informally, a trip is sequence of pointed sequents (i.e. sequents with a principal formula occurrence) with directions tracing a path bouncing on axioms, cuts or tensors that does not go through two premises of a \wp rule: anticipating the next section, trips are special types of switching paths. The trips in pre-proofs of fig. 8 are indicated as coloured lines.

PROPOSITION 3. *Let $((s_i, F_i, d_i))_{i \in \omega}$ be a trip of a pre-proof π . Then there exists an infinite branch that has infinitely many common terms with $(s_i)_{i \in \omega}$. Furthermore, if there exist two such infinite branches γ_1 and γ_2 then $\gamma_1 = \gamma_2$ after finitely many terms.*

PROOF. Suppose not. Then it contains infinitely many sequents from two diverging infinite branches, γ_1 and γ_2 , of π . Let γ be the finite common prefix of γ_1 and γ_2 . By construction, there is a tensor or a cut rule in γ such that it has premises s_l and s_r respectively and it introduces F_l and F_r respectively such that s_l and s_r occurs infinitely often in the trip. By construction, during a downward travel via s_l (resp. s_r), in order to change directions, the trip must be through F_l (resp. F_r) rather than any other occurrence of s_l (resp. s_r). Similarly, during the immediately succeeding upward travel, the trip must be through (s_r, F_r, \uparrow) . So, there is a finite trip starting from (s_r, F_r, \uparrow) and (s_r, F_r, \downarrow) . Schematically this finite trip looks like the following. We will show by induction on n that such a finite trip cannot exist.



Base Case We have $F_l^1 = F_r$. Then, G_1 and G_1^\perp are suboccurrences of F_r . Let G' be the suboccurrence which is the greatest common prefix of G_1 and G_1^\perp in the FL-graph of F_r . The outermost operator of G' must be either a parr or a tensor. In case it is a parr, the finite trip goes through two premises of G' which is not allowed. If it is a tensor, then G_1 and G_1^\perp go to different sequents and there cannot be axiom with them.

Induction Step Then, G_1 and G_n^\perp are suboccurrences of F_r . Let G' be the suboccurrence which is the greatest common prefix of G_1 and G_n^\perp in the FL-graph of F_r . The outermost operator of G' must be a tensor (parr is ruled out by the same logic as above). Let $G' = H \otimes H'$. Since the finite trip goes through G' , $\{H, H'\} = \{F_l^m, F_r^m\}$ for some $m < n$. But then we can follow F_r^m down to F_r (since G' is a suboccurrence of F_r). Hence we must have a shorter finite trip of the requisite form. By induction hypothesis that does not exist.

Hence a trip can be associated with at most one infinite branch. Further observe by the above argument every sequent is repeated at most finitely often in a trip, so a trip visits higher and higher (or deeper depending on one's perspective) sequents. Hence there is an infinite branch associated with every trip. \square

From proposition 3 we can associate a unique maximal infinite branch with a trip. Observe that trips correspond to visitable paths (up to a choice of axioms to bounce on) and the infinite branch associated with a trip corresponds to the infinite axiom above the corresponding visitable path.

Consider the pre-proof in fig. 8c and its corresponding proof structure in fig. 8h. They have two trips and visitable paths respectively. We can argue as before that an infinite axiom should be atop the two visitable paths, however no infinite axiom is available. We need to introduce a new infinite axiom which we call a *virtual axiom* as in fig. 8i. We will thus distinguish between infinite axioms that are supported by a straight thread (which we will call real axioms) and infinite axioms supported by visitable paths (virtual axioms). The intuition that an infinite axiom is the invariant of an infinite branch still carries forward.

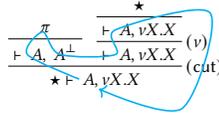


Fig. 9. Let π be the proof in fig. 8(c)

Let us ruminate about the procedure to desequentialise the pre-proof in fig. 9. In the first parse of the proof, one shall recognise the infinitely many visitable paths from the infinitely many trips (each occurrence of π generating two trips). Then, the second parse will illuminate the fact there is a “higher order trip” (indicated in blue in fig. 9) which should correspond to a visitable path⁷ that resides with the infinite undirected v -ray in a real axiom. We distinguish this visitable path from others, as it is only revealed in the second parse of the proof. We call such visitable paths, *higher order visitable paths*. To simplify the presentation, we will only consider pre-proofs without higher order trips which we call *simple pre-proofs*. Our results scale to general pre-proofs but all our key ideas and techniques can be illustrated over simple pre-proofs. We bookend this discussion summarizing the terms introduced:

Pre-proofs	Non-wellfounded Proof Structures
Axioms	Finite axiom links
Real branches	Real axioms
Virtual branches	Virtual axioms
Trips	Visitable paths
Higher order trips	Higher order visitable paths

REMARK 2. *The infinets introduced in [21] do not have this crucial additional structure of visitable paths (rendering the original definition of μMLL^∞ proof structures in that paper incomplete), which can feature even in proof structures with finitely many cuts.*

5 NON-WELLFOUNDED PROOF STRUCTURES

NWFPS. In section 4 we have treated infinitary proof structures as infinite graphs. This is imprecise since infinite axioms—nodes which join infinite paths in their infinite directions—are meaningless graphically. We formalise our intuition by adapting the formalism of [17] to our setting.

⁷Imagine bouncing through π as bouncing on a generalized axiom.

DEFINITION 11. A **non-wellfounded proof structure** (abbreviated *NWFPS*) is a 6-tuple $(\{F_i^{U_i}\}_{i \in I}, \mathfrak{R}, \Theta_f, \Theta_r, V, \Theta_v)$ which respectively comprises of a set of partial syntax trees, a set of cuts, a set of finite axioms, a set of real axioms, a set of visitable ends, and a set of virtual axioms. A *NWFPS* is denoted by \mathcal{R}, \mathcal{S} , etc.

We will now explain the meaning of each term.

DEFINITION 12. A **syntax tree** of a formula occurrence F is the (possibly infinite) unfolding tree of the Fischer-Ladner graph of 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}_F 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 , s.t. $U \subseteq \mathcal{L}_F$ and U represents a bar 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 $\text{addr}(F).u$.

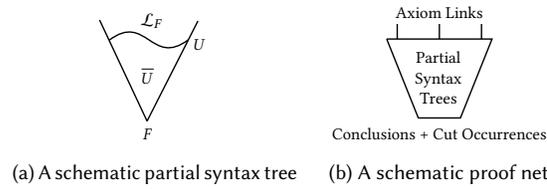


Fig. 10. Illustration of partial syntax trees and proof nets

We illustrate a schematic partial syntax tree in fig. 10a. MLL proof nets without cuts can be seen as a forest of partial syntax trees of the occurrences in the conclusion sequent and axiom links between their leaves as in fig. 10b. To incorporate cuts we need to add the partial syntax tree of the cut occurrences (along with the axioms links involving their leaves) and links between dual cut occurrences. Hence a proof-net would be a 3-tuple $(\{F_i^{U_i}\}_{i \in I}, \mathfrak{R}, \Theta)$ where:

- for all $i \in I$, $F_i^{U_i}$ is a partial syntax tree; $\{F_i\}_{i \in I}$ is called the set of **doors**.
- \mathfrak{R} is the set of cuts i.e. a (possibly empty) set of disjoint subsets of $\{F_i\}_{i \in I}$ of the form $\{C, C^\perp\}$; and,
- Θ is the set of axiom links i.e. a partition of the set of leaves, $\mathcal{L} = \bigcup_{i \in I} \{\alpha_i u_i \mid \text{addr}(F_i) = \alpha_i, u_i \in U_i\}$ such that each cell is pair of dual addresses i.e. of the form $\{\alpha_i u_i, \alpha_j u_j\}$ such that $\lceil (F_i, u_i) \rceil = \lceil (F_j, u_j) \rceil$.

However, recall from section 4 that axioms are finite, real or virtual. So, in μMLL^∞ , $\Theta = \Theta_f \uplus \Theta_r \uplus \Theta_v$ where:

- Θ_f is the set of finite axioms: its elements are pairs of finite dual addresses from \mathcal{L} . They are links between leaves of partial syntax trees denoted by ax in fig. 8.
- Θ_r is the set of real axioms: the invariant of an infinite branch of a pre-proof supported by a straight thread. Its elements necessarily contain at least one infinite address and might contain visitable paths. They are denoted by ax_r^∞ in figs. 8c and 8f.
- Θ_v is the set of virtual axioms: the invariant of an infinite branch of a pre-proof supported by a visitable thread. Its elements necessarily contain visitable paths and might contain finite addresses from \mathcal{L} . They are denoted by ax_v^∞ in fig. 8i.

Hence Θ is a partition over $\mathcal{L} \cup V$ where V is the set of visitable paths. Observe that permutation of inference rules in a pre-proof can induce a permutation of terms in its trips. However, if we only take the principal formulas of a trip (without repetition), this sequence is invariant. Further observe that this sequence is fixed by only stating the points of alternation of directions i.e. the cuts and tensors for \downarrow to \uparrow and finite and real axioms for \uparrow to \downarrow . Hence V is a set of

sequences of alternating tensors (or cuts) and axioms. We must impose a sanity condition since any such sequence cannot be a visitable path. Now recall we have higher-order visitable path which go through visitable paths. Hence one needs to define the set of visitable paths incrementally.

The **level 0 approximant** of a NWFPs has components $(\{F_i^{U_i}\}_{i \in I}, \mathfrak{R}, \Theta_f, \Theta_r)$ where $\Theta_f \cup \Theta_r$ contains subsets of \mathcal{L} : it is the NWFPs minus its visitable paths.

EXAMPLE 4. Consider the NWFPs in fig. 8f. Its level 0 approximant is of the form $(\{H^{U_1}, (A_{\beta}^{\perp})^{U_2}\}, \emptyset, \{\theta_n\}_{n \geq 0}, \{(il)^{\omega}\})$ such that α and β are disjoint, $U_1 = (il)^* .ir(l+r) + (il)^{\omega}$, U_2 is simply $\{\epsilon\}$, $\theta_0 = \{\alpha ir^2, \beta\}$ and for every $n > 0$, $\theta_n = \{(il)^{n-1} ir^2, (il)^{n-1} ir l\}$.

On top of a level 0 approximant we will augment the information of visitable paths and potentially add virtual axioms. In the process, real axioms might also be augmented with visitable paths. We will now formalize the notion of a level 0 visitable path (which is a visitable path that does not go through other visitable paths).

DEFINITION 13. Given a level 0 approximant, a **level 0 visitable path** is an infinite sequence $\{t_i\}_{i \in \mathbb{N}}$ such that if i is odd then t_i is either a tensor formula occurrence (F_j, u) for some $j \in I$ and $u \in \overline{U_j}$ or an element of \mathfrak{R} and if i is even then $t_i \in \Theta_f \uplus \Theta_r$. Further, there exists a pair of infinite sequences of addresses, $(\{l_i\}_{i \in \mathbb{N}}, \{r_i\}_{i \in \mathbb{N}})$ such that for every $i \in \mathbb{N}$:

- If $t_{2i-1} = (F_j, u)$ is a tensor formula, then u is a prefix of l_i and r_i .
- If $t_{2i-1} = \{C, C^{\perp}\}$, then $\text{addr}(C)$ and $\text{addr}(C^{\perp})$ are prefixes of l_i and r_i respectively.
- $\{r_i, l_{i+1}\} \subseteq t_{2i}$.

Two visitable paths are equivalent if they share a common tail⁸. Their equivalence classes are called **visitable ends**.

REMARK 3. In definition 13, the choice to start tensors or cuts instead of axioms is arbitrary.

DEFINITION 14. A **level 1 approximant** is a 6-tuple $(\{F_i^{U_i}\}_{i \in I}, \mathfrak{R}, \Theta_f, \Theta_r, V, \Theta_v)$ which respectively comprises of a set of partial syntax trees, a set of cuts, a set of finite axioms, a set of real axioms, a set of visitable ends, and a set of virtual axioms such that the following hold:

- $\{F_i\}_{i \in I} \setminus \bigcup_{\kappa \in \mathfrak{R}} \kappa$ is finite.
- $\Theta_f \uplus \Theta_r \uplus \Theta_v$ is a partition of the set $\bigcup_{i \in I} \{\alpha_i u_i \mid \text{addr}(F_i) = \alpha_i, u_i \in U_i\} \cup V$ such Θ_f has only elements containing finite dual addresses, elements of Θ_r necessarily contain an infinite address, and, elements of Θ_v do not contain infinite addresses and necessarily contain at least one visitable end.

One can similarly define a level λ visitable path and a level λ approximant for any ordinal λ . It is helpful to remember the following maxim for any successor ordinal λ :

Level $\lambda + 1$ approximant = Level λ approximant + level λ visitable path

Just like a smooth function is the sum of all its n^{th} order approximants (Taylor expansion), a non-wellfounded proof structure (abbreviated NWFPs) is **well-defined** if it is the union of all its approximants.

DEFINITION 15. A **simple NWFPs** is a NWFPs that is identical to its level 1 approximant.

In the rest of the paper, we refer to simple NWFPs as NWFPs unless explicitly mentioned.⁹

⁸In the parlance of infinite graph theory, subrays of a ray are called its tail and ends are rays quotiented by a common tail.

⁹NWFPs are treated in full generality in the appendix (section 5.1).

EXAMPLE 5. In fig. 8f, we have the visitable path $\rho = \{t_n\}_{n \in \mathbb{N}^*}$ such that

$$t_n = \begin{cases} \theta' & n = 1; \\ \theta \lfloor \frac{n}{2} \rfloor & n \text{ is odd}; \\ (B_1, (il) \lfloor \frac{n}{2} \rfloor ir) & n \text{ is even.} \end{cases}$$

Check that every other visitable path we can produce is a suffix of this. Hence there is only one visitable end (say, $[\rho]$). Observe that we need to augment θ_0 by adding this visitable end i.e. $\theta_0 = \{ \{(il)^\omega, [\rho] \}$. There are no virtual axioms, hence $\Theta_v = \emptyset$. Check that $\Theta_f \cup \Theta_r \cup \Theta_v$ is a partition of $\alpha U_1 \cup \beta U_2 \cup \{[\rho]\}$. Furthermore, there cannot be any visitable paths of level 2. Hence this is a simple NWFFS.

Desequentialization. Desequentialization of a μ MLL $^\infty$ pre-proof is not as straightforward as in MLL. Let π be a pre-proof of the μ MLL $^\infty$ sequent $\vdash \Gamma$. We will translate to a level 0 approximant first:

- for any cut in π that introduces two occurrences, C and C^\perp , $\{C, C^\perp\} \in \mathfrak{R}$.
- $\{F_i\}_{i \in I} = \Gamma \cup \bigcup_{\kappa \in \mathfrak{R}} \kappa$.
- for every $i \in I$, $U_i = \text{addr}(F_i)^{-1} \text{addr}(\pi)$.
- for every axiom linking (F_i, u_i) to (F_j, u_j) , $\{\text{addr}(F_i).u_i, \text{addr}(F_j).u_j\} \in \Theta_f$.
- for every real infinite branch γ in π , $\{\text{addr}(F_i).u_i\}_{i \in I'} \in \Theta$ where it is the largest set such that $I' \subseteq I$ and either u_i is infinite and $\text{addr}(F_i).u_i$ corresponds to a straight thread of γ or u_i is finite and (F_i, u_i) occurs in infinitely many sequents along γ .

To produce the set of visitable ends of the desequentialization, given a trip in π , we collect the points of alternation of directions which gives us a sequence of cuts or tensor and axioms: this is a visitable path in the level 0 approximant. This gives us a set of visitable path, which in turn will give us a set of visitable ends, V . We are now ready to define the desequentialization of a pre-proof π .

DEFINITION 16. The **desequentialization** of π , $(\{F_i^{U_i}\}_{i \in I}, \mathfrak{R}, \Theta_f, \Theta_r, V, \Theta_v)$, denoted $\text{Deseq}(\pi)$, is such that:

- $(\{F_i^{U_i}\}_{i \in I}, \mathfrak{R}, \Theta_f, \Theta_r)$ is the level 0 approximant as described above.
- V is the set of visitable ends of the level 0 approximant.
- For every real infinite branch, γ , in π that has a trip associated with it, let θ be its corresponding real infinite axiom. Update θ as $\theta \cup V'$ where V' is the largest set such that for every $v \in V' \subseteq V$ there exists a trip ρ associated with γ such that v is obtained from ρ .
- For every virtual infinite branch, γ , in π , $\{\text{addr}(F_i).u_i\}_{i \in I'} \cup V' \in \Theta_v$ is the largest such that $I' \subseteq I$, every u_i is finite, (F_i, u_i) occurs in infinitely many sequents along γ and for every $v \in V' \subseteq V$ there exists a trip ρ associated with γ such that v is obtained from ρ .

EXAMPLE 6. If we desequentialize the proof in fig. 8d we get the NWFFS described in examples 4 and 5.

5.1 Treating non-simple proofs

We will first give a more granular definition of higher order trips and then go on to define higher order visitable paths. Let trips as defined in definition 10 be known as level 0 trips. We refine the definition of higher order trips to define a level $\lambda + 1$ trip for a successor ordinal λ .



Fig. 11. The operator loc changes the address of ϕ from α to β .

DEFINITION 17. Given a pre-proof π , a **level $\lambda + 1$ trip** is an infinite sequence $\tau = \{(s_i, F_i, d_i)\}_{i \in \omega}$ where s_i is a sequent in π , $F_i \in s_i$ and $d_i \in \{\uparrow, \downarrow, \uparrow^\omega, \downarrow^\omega\}$ such that for every $i, j < \omega$ exactly one of the following holds:

- either $d_i = \uparrow, d_{i+1} = \downarrow, s_i = s_{i+1}$ and there exist level λ' trips τ and τ' (where $\lambda' \leq \lambda$), starting from F_i and F_{i+1} and they have the same infinite branch associated with them; or,
- they fall into one of the conditions of definition 10.

Further, τ satisfies the following conditions:

- For every $i, j < \omega$, there does not exist a sequent, s , in π such that $F_i \not\approx F_j \in s$.
- There are infinitely many terms of the form (s, F, \downarrow) .

The set of all level λ trips for a limit ordinal λ is the union of the sets of all level λ' trips for every $\lambda' < \lambda$.

Observe that proposition 3 will still hold in this case. Correspondingly, we will define a level k visitable path. To define a level $\lambda + 1$ visitable path, we assume that we have the level λ approximant *i.e.* the level 0 approximant along with all level λ visitable paths and virtual axioms pertaining to them.

DEFINITION 18. Given a level λ approximant, a **level $\lambda + 1$ visitable path** is an infinite sequence $\{t_i\}_{i \in \mathbb{N}}$ such that if i is odd then t_i is either a tensor formula occurrence (B_j, u) for some $j \in I$ and $u \in \overline{U}_j$ or an element of \mathfrak{R} and if i is even then $t_i \in \Theta_f \uplus \Theta_r \in \Theta_v$ such that:

- either there is a switching path from t_{2i} from t_{2i+2} via t_{2i+1} à la definition 13,
- or, there are level λ' visitable paths (where $\lambda' \leq \lambda$) starting from t_{2i} and t_{2i+2} respectively such that the ends corresponding to them reside in t_{2i+1} .

The set of all level λ visitable paths for a limit ordinal λ is the union of the sets of all level λ' visitable paths for every $\lambda' < \lambda$. Visitable ends are defined on the set of all visitable paths. The level of an end is the level of the visitable path of the largest level in it. Consequently the progress from simple NWFPs to general NWFPs happens through a simple embellishment of V , the set of visitable ends.

Desequentialisation to a level 0 approximant is unperturbed. The rest of the desequentialisation happens recursively. We recursively recognize visitable paths and add new virtual axioms or embellish existing infinite axioms. One can think of this procedure as starting with a level 0 approximant and an empty set for the visitable paths and virtual axioms and imagine process of adding visitable ends and embellishing and/or creating infinite axioms as a monotonic function on the space of NWFPs getting better approximations in every iteration. In essence, for each λ we create a level λ approximant of desequentialisation. Monotonicity guarantees a fixed point *i.e.* we are guaranteed to stabilize for some λ when we neither add new visitable paths nor augment/create infinite axioms.

Let $(\{B_i^{U_i}\}_{i \in I}, \mathfrak{R}, \Theta_f, \Theta_r, V, \Theta_v)$ be level λ approximant of the desequentialisation of π for some successor ordinal λ . The level $\lambda + 1$ approximant of the desequentialisation of π is $(\{B_i^{U_i}\}_{i \in I}, \mathfrak{R}, \Theta_f, \Theta_r', V', \Theta_v')$ such that:

- Given level λ approximant, one can compute V_λ , the set of level λ visitable ends. $V' = V \cup V_\lambda$.
- For every real infinite branch, γ , in π that has a trip associated with it, let θ be the real infinite axiom corresponding to it. Update θ as $\theta \cup V''$ where V'' is the largest set such that for every $v \in V'' \subseteq V'$ there is a set of trips of π with a common tail associated with γ .
- For every virtual infinite branch, γ , in π , $\{\text{addr}(B_i).u_i\}_{i \in I'} \cup V'' \in \Theta_v$ is the largest such that $I' \subseteq I$, every u_i is finite, (B_i, u_i) occurs in infinitely many sequents along γ and for every $v \in V'' \subseteq V'$ there is a set of trips of π with a common tail associated with γ .

Desequentialisation to a level λ approximant for a limit ordinal λ is given by taking component-wise unions of level λ' approximants for $\lambda' < \lambda$.

5.2 An illustration of desequentialisation

Consider the proof in fig. 8d (more specifically the unfolding of that circular proof). Let us call it π . Let $(\{B_1^{U_1}, B_2^{U_2}\}, \emptyset, \Theta_f, \Theta_r)$ be the straight fragment. Since we have two conclusions and no cuts, there are two partial systanx trees. Let $B_1 = H$ and $B_2 = A_\beta^\perp$ such that α and β are disjoint. Now we trace the suboccurrences of B_1 and B_2 . We get that $U_1 = (il)^*.ir(l+r) + (il)^\omega$ and U_2 is simply $\{\epsilon\}$. Now we look at each axiom link and observe that $\theta' = \{\alpha ir^2, \beta\} \in \Theta_f$ and for every $n \geq 0$, $\theta_n = \{(il)^{n+1}ir^2, (il)^n ir l\} \in \Theta_f$. Now observe that the left-most branch of π is infinite and has exactly one straight thread. Hence $\theta'' \{\{(il)^\omega\} \in \Theta_r$.

By reading off the alternating tensors and finite axioms along the trip indicated in fig. 8d, we obtain the visitable path $\rho = \{t_n\}_{n \in \mathbb{N}^*}$ such that

$$t_n = \begin{cases} \theta' & n = 1; \\ \theta_{\lfloor \frac{n}{2} \rfloor} & n \text{ is odd}; \\ (B_1, (il)^{\lfloor \frac{n}{2} \rfloor} ir) & n \text{ is even}. \end{cases}$$

Check that every other visitable path we can produce is a suffix of this. Hence there is only one visitable end (say, $[\rho]$). Now observe every element of $[\rho]$ is associated with the left-most branch of π . Hence we augment θ'' by adding this visitable end *i.e.* $\theta'' = \{\{(il)^\omega, [\rho]\}$. There are no virtual axioms, hence $\Theta_v = \emptyset$. Check that $\Theta_f \cup \Theta_r \cup \Theta_v$ is a partition of $\alpha U_1 \cup \beta U_2 \cup \{[\rho]\}$.

5.3 Non-wellfounded substructures

A proof structure \mathcal{S} is called a **substructure** of a proof structure \mathcal{R} if there is an injective map $m : \mathcal{S} \rightarrow \mathcal{R}$ preserving links. Since NWFPs are not presented graphically, we need to specify the exact nature of this injection: Let $\mathcal{S} = (\{B_i^{U_i}\}_{i \in I}, \mathfrak{R}, \Theta_f, \Theta_r, V, \Theta_v)$ and $\mathcal{R} = (\{B_i^{U'_i}\}_{i \in I}, \mathfrak{R}', \Theta'_f, \Theta'_r, V', \Theta'_v)$. Then, $m : \{(B_i, u) \mid u \in \overline{U_i}, i \in I\} \rightarrow \{(B'_i, u) \mid u \in \overline{U'_i}, i \in I'\}$ such that:

- For all $i \in I$, $[B_i] = [m(B_i)]$.
- For all $i \in I$, $U_i = U'_j$ where $m(B_i) = B'_j$.
- For all $\{C, C^\perp\} \in \mathfrak{R}$, we have $\{m(C), m(C^\perp)\} \in \mathfrak{R}'$. Hence we can lift m to cuts writing $m(\{C, C^\perp\}) = \{m(C), m(C^\perp)\}$.
- for all $\theta \in \Theta_f$, we have $\theta' \in \Theta'_f$ where $\theta' = \{\text{addr}(m(B))u \mid \alpha u \in \theta, \text{addr}(B) = \alpha\}$. Hence we can lift m to finite axioms by writing $m(\theta) = \theta'$. Exactly the same follows for real axioms that do not contain a visitable end.
- Let T be the set of all visitable paths in \mathcal{S} . Then $\{\{m(t_i)\}_{i \in \mathbb{N}} \mid \{t_i\}_{i \in \mathbb{N}} \in T\}$ is the set of visitable paths of \mathcal{R} . Hence we can lift m to visitable ends.
- Now we can lift m to rest of the real axioms and virtual axioms.

In other words, the sub-NWFPs of a NWFPs, \mathcal{R} is an injective map between the set of their formula occurrences that can be lifted to the other components. We define the intersection, $\mathcal{S} \cap \mathcal{S}'$ of two sub-NWFPs, \mathcal{S} and \mathcal{S}' . Let $m : \mathcal{S} \rightarrow \mathcal{R}$ and $m' : \mathcal{S}' \rightarrow \mathcal{R}$ be the injective maps specifying the two sub-NWFPs. Consider \mathcal{R} restricted to the set of formula occurrences $\text{Im}(m) \cap \text{Im}(m')$. Observe that the trivial identity map induces a sub-NWFPs. We call this sub-NWFPs, $\mathcal{S} \cap \mathcal{S}'$.

PROPOSITION 4. *Let \mathcal{R} be a DR-correct NWFPS. For any occurrence F in \mathcal{R} , $k(F)$ exists and is unique.*

PROOF. The idea is to show that there is at least one sub-NWFPS with F as the lowermost occurrence. Then, we will show uniqueness.

Suppose F is a suboccurrence of the door, F_i , in \mathcal{R} . Consider the partial syntax tree, $F_i^{U_i}$ in \mathcal{R} . Consider a bar of this tree with F . We cut the tree at this bar and take the subtree under each element of the bar to be new partial syntax trees. If F_i is not a cut occurrence then it is easy to check that this gives us a sub-NWFPS. If $\{F_i, F_j\}$ is a cut, take a switching of \mathcal{R} and disconnect the cut. Now remove the connected component containing F_j and restore the premisses of the parrs erased by the switching. This gives us a sub-NWFPS with F as a door. Let \mathcal{S} be the set of sub-NWFPS with F as a door. Then,

$$k(F) = \bigcap_{S \in \mathcal{S}} S$$

□

PROPOSITION 5. *Let \mathcal{R} be a DR-correct NWFPS. Then, the relation \ll is a partial order on $\{(B_i, u) \mid u \in \overline{U_i} \setminus U_i, i \in I\}$.*

PROOF. **Reflexivity:** Follows by definition.

Anti-symmetry: Suppose $X \in k(Y)$ and $Y \in k(X)$. Consider $k(X) \cap k(Y)$. This is a sub-NWFPS that contains X as a door but is smaller than or equal to $k(X)$. Hence, $k(X) = k(X) \cap k(Y)$. Similarly, $k(Y) = k(X) \cap k(Y)$. Hence, $k(X) = k(Y)$. But k is obtained by injective maps, this implies $X = Y$.

Transitivity: Suppose $X \in k(Y)$ and $Y \in k(Z)$. Consider $k(Y) \cap k(Z)$. This is a sub-NWFPS that contains Y as a door but is smaller than or equal to $k(Y)$. Hence $k(Y) = k(Y) \cap k(Z) \subset k(Z)$. Hence $x \in k(Z)$. □

5.4 Treating NWFPS with locs

Recall from section 3 we are in an extended system *i.e.* μMLL^∞ with $(\text{Loc}(i))$. Hence we need add new relocation cells with one premise and one conclusion, changing the addresses (as illustrated in fig. 11). Formally NWFPS have one more component *loc*: a bijection between a finite subset L of \mathcal{L} and a finite subset C of doors such that the underlying formulæ of the image and the antecedent are equal. However, since the geometry of NWFPS is completely unaffected by the presence of finitely many *loc* nodes we will ignore them. To carry cut-elimination, we introduce non-canonical NWFPS that contain new relocalization cells.

DEFINITION 19 (NON-CANONICAL NWFPS). *A simple NWFPS is a 7-tuple*

$$(\{B_i^{U_i}\}_{i \in I}, \mathfrak{R}, \Theta_f, \Theta_r, V, \Theta_v, \text{loc})$$

which respectively comprises of a set of partial syntax trees, a set of cuts, a set of finite axioms, a set of real axioms, a set of visitable ends, a set of virtual axioms and a relocation function such that the following hold:

- *loc is a bijection between a finite subset L of the leaves of the partial trees and a finite subset C of the conclusions such that the underlying formulæ of the image and the antecedent are equal;*
- *$\bigcup_{\kappa \in \mathfrak{R}} \kappa$ is disjoint from the image of *loc*;*
- *$\{B_i\}_{i \in I} \setminus \bigcup_{\kappa \in \mathfrak{R}} \kappa$ is finite;*
- *a visitable path is an infinite sequence $\{t_i\}_{i \in \mathbb{N}}$ such that if i is odd then t_i is*
 - *either a tensor formula occurrence (B_j, u) for some $j \in I$ and $u \in \overline{U_j}$*
 - *or an element of \mathfrak{R}*

- or in the image of loc
 - and if i is even then $t_i \in \Theta_f \uplus \Theta_r$ or in the domain of definition of loc . Further, there exists a pair of infinite sequences of addresses, $(\{l_i\}_{i \in \mathbb{N}}, \{r_i\}_{i \in \mathbb{N}})$ such that for every $i \in \mathbb{N}$:
 - If $t_{2i} = (B_j, u)$ is a tensor formula, then u is a prefix of l_i and r_i .
 - If $t_{2i} = \{C, C^\perp\}$, then $\text{addr}(C)$ and $\text{addr}(C^\perp)$ are prefixes of l_i and r_i respectively.
 - $\{r_i, l_{i+1}\} \subseteq t_{2i+1}$.
- Two visitable paths are said to be equivalent if they share a common tail. The equivalence classes of visitable paths are called **visitable ends**.
- $\Theta_f \uplus \Theta_r \uplus \Theta_v$ is a partition of the set

$$\left(\bigcup_{i \in I} \{\alpha_i u_i \mid \text{addr}(B_i) = \alpha_i, u_i \in U_i\} \cup V \right) \setminus C$$

such Θ_f has only elements containing finite dual addresses, elements of Θ_r necessarily contain an infinite address, and, elements of Θ_v do not contain infinite addresses and necessarily contain at least one visitable end.

Informally, loc represents the possibility to change the addresses locally. It can be simulated inside canonical NWFPS. If we decided to represent non-canonical NWFPS, we would do so by adding a type of cell, with one premise and one conclusion, changing the addresses.

PROPOSITION 6. A non-canonical NWFPS defines a NWFPS by replacing all the loc by ax/cut .

This is a form of η -expansion of relocation. Relocations are inside kingdoms.

6 CORRECTNESS CRITERION

In this section, we develop a correctness criterion on NWFPS, strengthening the correctness criterion in [21] to account for visitable paths. It has two conditions:

DR-correct: The switching graph is acyclic and connected.

Lock-free: The dependence graph [9, 21] is locally finite and contains no ray as a subgraph.

DR-correctness. Because μMLL^∞ contains MLL, DR-correctness is necessary. However, in the setting of NWFPS, DR-correctness needs to be rephrased in terms of orthogonal partitions. The reason why it is not sufficient is more subtle: the presence of infinitely many vertices in a NWFPS leads to pathological cases where in order to sequentialise a certain vertex needs to wait for infinitely many vertices to sequentialise.

DEFINITION 20. Given a NWFPS, $\mathcal{R} = (\{F_i\}_{i \in I}, \{U_i\}_{i \in I}, \mathfrak{R}, \Theta_f, \Theta_r, V, \Theta_v)$, a **switching**, sw , of \mathcal{R} is set of functions $\{sw_i : P_i \rightarrow \{l, r\}\}_{i \in I}$ s.t. for every $i \in I$, $P_i \subseteq \overline{U_i}$ and $\lceil (F_i, p) \rceil$ is a \mathfrak{R} -formula for all $p \in P_i$.

Observe that a switching of a NWFPS can be fixed on its level 0 approximant.

DEFINITION 21. Let sw be a switching of \mathcal{R} . Let u be a substring of a word w in $\overline{U_i}$. Then, u is said to be **unbroken** if for all $j \in \{1, \dots, n-1\}$, $sw_i(vu_1 \dots u_j) \neq u_{j+1}$ where $u = u_1 \dots u_n$ and vu is a prefix of w for some word v .

Fix a switching, sw , of \mathcal{R} . Let $\text{SW} \subseteq \mathcal{L}^2$ such that $(x, y) \in \text{SW}$ iff either $x = y$ or one of the following holds:

- $w = x \cap y \neq \epsilon^{10}$. Let $wu = x$ and $wv = y$. Then, u and v are unbroken;

¹⁰i.e. x and y are not disjoint and the w is their largest common prefix.

- $x = \alpha u$ and $y = \alpha' v$ such that $\text{addr}(C) = \alpha$, $\text{addr}(C^\perp) = \alpha'$, $\{C, C^\perp\} \in \mathfrak{R}$ and u, v are unbroken.

Observe that SW is an equivalence. If we see the elements of \mathcal{L} as the collection of leaves of the partial syntax trees of a proof net, cells of SW are the connected components of that proof net under the switching sw and without axiom links.

DEFINITION 22. *The level 0 orthogonal graph of \mathcal{R} for the switching, sw , (denoted $G_0^{sw}(\mathcal{R})$) is the undirected bipartite multigraph, $(\Theta, [\text{SW}], E_0)$, where $\Theta = \Theta_f \cup \Theta_r \cup \Theta_v$ i.e. the axioms of \mathcal{R} , $[\text{SW}]$ is the set of equivalence classes of SW and $(x, y) \in E_0$ iff $x \cap y \neq \emptyset$.*

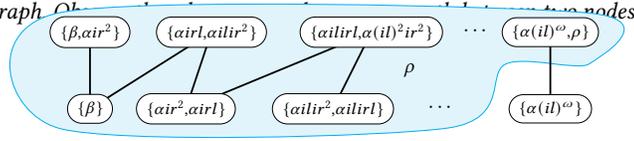
Observe that for a NWFPS with no visitable paths, DR-correctness as stated in [21] means that, for every switching, the orthogonal graph is acyclic and connected. However, as discussed in section 4, this is not enough in general and visitable paths need to be incorporated into orthogonal graphs.

PROPOSITION 7. *There is a one-one correspondence between $E_{\mathcal{R}} = \bigcup_{sw} \{[\rho] \mid [\rho] \text{ is an end in } G_0^{sw}(\mathcal{R})\}$ and the set of visitable ends of \mathcal{R} .*

DEFINITION 23. *Given a level 0 orthogonal graph $G_0^{sw}(\mathcal{R}) = (\Theta, [\text{SW}], E_0)$, the level 1 orthogonal graph (denoted $G_1^{sw}(\mathcal{R})$) is the undirected hybridgraph¹¹ $(\Theta, [\text{SW}], E_0, E_1)$ such that for every $\theta \in \Theta_r \cup \Theta_v$ that contains a visitable end, $\{\theta\} \cup S \in E_1$ where $S \subseteq \Theta \cup [\text{SW}]$ is the set of all nodes appearing in every end of θ . A **pure path** in $G_1^{sw}(\mathcal{R})$ is path comprised of only E_0 or E_1 but not both.*

DEFINITION 24. *A NWFPS, \mathcal{R} , is said to be **DR-correct** if for any switching sw , between any two nodes of $G_1^{sw}(\mathcal{R})$ there is exactly one pure path.*

EXAMPLE 7. *We illustrate checking DR-correctness when all the \mathfrak{A} s switches of the proof structure in fig. 8f are switched to the left. The following figure (without the coloured region) is the level 0 orthogonal graph corresponding to it. It has two connected components and one of them has an infinite path (denoted by ρ). The cyan region is an extra hyperedge added in the level 1 orthogonal graph. Observe that this hyperedge connects two nodes.*



Lock-freeness. We reformulate the condition in [21] (for finitely many cuts) through non-wellfounded substructures (sub-NWFPS) and kingdoms (adapted from [10]). Viewed as graphs, a sub-NWFPS¹² is a subgraph that is a NWFPS. The kingdom of a formula occurrence, F , is the upward-closed sub-NWFPS starting from F .

DEFINITION 25. *Let \mathcal{R} be a DR-correct NWFPS. Given B_i and $u \in \overline{U}_i$, the **kingdom**, $k(B_i, u)$, of (B_i, u) is the smallest DR-correct sub-NWFPS of \mathcal{R} with (B_i, u) as one of its doors. We define a relation on the set of occurrences $\{(B_i, u) \mid u \in \overline{U}_i, i \in I\}$ by $X \ll Y$ iff $X \in k(Y)$.*

EXAMPLE 8. *The NWFPS in fig. 12 is DR-correct. The kingdom $K(p_1)$ and $K(t_2)$ are drawn in cyan and magenta. Observe that $t_1 \ll p_1 \ll t_2 \ll p_2 \ll \dots$*

DEFINITION 26. *A NWFPS is **lock-free** if $\{m \mid n \ll m\}$ is finite for all n .*

¹¹We coin the term *hybridgraph* for a graph with both normal edges and hyperedges.

¹²See Section 5.3

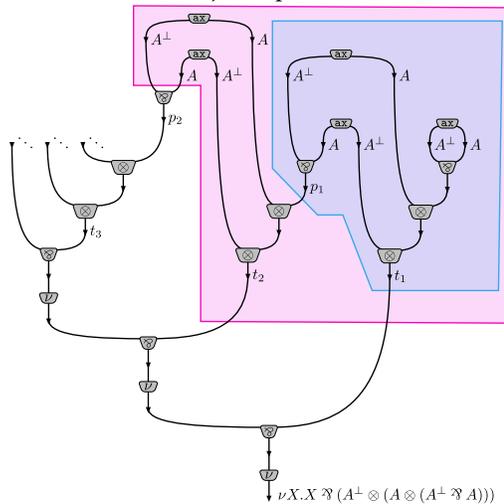


Fig. 12. A DR-correct NWFPS exhibiting kingdoms

DEFINITION 27. A NWFPS is an **infinet** if it is DR-correct and lock-free.

PROPOSITION 8. Let π be a pre-proof. $\text{Deseq}(\pi)$ is an infinet.

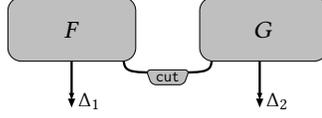
PROOF. The crux of the proof is to show DR-correctness. For lock-freeness, observe that if an occurrence, A , first appears in a sequent above the sequent of the first appearance of an occurrence B then A cannot be in the kingdom of B . We will first outline the game-plan and then lay the details.

- The orthogonal graph is acyclic.
- If the orthogonal graph is disconnected then there exists visitable paths.
- If there are no visitable paths we are done. Suppose not.
- The corresponding 1st level orthogonal graph is acyclic. The argument for this basically is the same for acyclicity of the orthogonal graph.
- If the 1st level orthogonal graph is disconnected then there exists higher order visitable paths. The proof for this mimics that of the second item.
- Since a simple proof does not have higher order visitable paths, we are done.

So, we basically need to prove the first two items. Let us start with the first. Suppose the orthogonal graph has a cycle. Although the graph could potentially be infinite, a cycle must be finite by definition (in fact, with even number of vertices since the graph is bipartite). Let the cycle be $r_0, r_1, \dots, r_n, r_0$ where for all $j \in \{0, \dots, \lfloor \frac{n}{2} \rfloor\}$ we have that $r_{2j} \in [R_0]$ and $r_{2j+1} \in [R_0^{sw}]$. Choose $u_j \in r_{2j} \cap r_{2j+1}$. Since u_j, u_{j+1} are in same cell of R_0^{sw} , they must be connected by a tensor or a cut node. For each j , call this particular node t_j . Observe that t_j, t_{j+1} cannot be separated by a tensor or a cut because they share two pieces of an axiom, r_{2j} . So, t_j and t_{j+1} must happen one after the other. This leads to a contradiction in order of introduction of the t_j 's.

For the second item assume the orthogonal graph is disconnected. Say, in the graph of the NWFPS, occurrences F and G are disconnected. (This can be stated via elements of $[R_0]$ and $[R_0^{sw}]$ but we avoid that for the sake of clarity. We will keep referring to the graph of the NWFPS for the rest of the proof but note that this can be made rigorous. Since F, G are suboccurrences of some conclusions which started out in the same sequent, there must have been a point in the course

of the pre-proof where a tensor or cut separated them or some occurrences of which they were suboccurrences. WLOG it is a cut. Schematically we are in the following situation



But we assumed that the graph is disconnected. So either the connection between A and the left premise of the cut or the connected between B and the right premise of the cut is snapped. We now continue the same argument. If this procedure stops, we end up in a connected graph which is not possible. Thus this procedure continues and we have produced a infinite sequence of cuts or tensors. We claim that this is a visitable path. Cuts can only be connected via axioms. Hence if we have infinitely many cuts we are done. Tensors can either be connected as suboccurrences or via axioms. If we have infinitely many tensors connected as suboccurrences, we have a straight thread that must reside in an infinite branch with A and B . This is absurd. Hence if there are infinitely many tensors, they are connected via axioms. Therefore there exists visitable paths. \square

Sequentialisation. We now give an informal description of the corecursive definition of sequentialisation. The technique in [21] basically follows the standard procedure for MLL but with a guarantee of *fairness* preventing a situation where the exploration of a branch is forgotten since the sequentialisation of another branch is forever prioritized. Fairness is ensured by time-stamping the doors of the infinet with elements of $\mathbb{N} \cup \{\infty\}$, which dictates that at any particular step the node with the least time-stamp is to be sequentialised.

We strengthen this time-stamping to account for infinitely many cuts. Given an infinet, we treat cuts as tensors: a “quasi” infinet (say \mathcal{R}) with potentially infinitely many conclusions. We carefully initialize the time-stamping such that infinitely many numbers are free to be used as time-stamps at later stages of the sequentialisation. Consider $t_{\mathcal{R}}$ which injectively time-stamps every maximal door in the \ll ordering by powers of two¹³ and every other door by ∞ . $\text{SEQUENTIALISE}(\mathcal{R}, t_{\mathcal{R}})$ chooses the door, F , with least time-stamp, applies the corresponding rule on the finite prefix of the sequentialisation being built, and relaunches $\text{SEQUENTIALISE}(\mathcal{R}', t_{\mathcal{R}'})$ for every sub-infinet, \mathcal{R}' and the time-stamping that results from removing F from \mathcal{R} .

THEOREM 2. *Given an infinet, \mathcal{R} , and a proper time-stamping, $t_{\mathcal{R}}$, $\text{Deseq}(\text{SEQUENTIALISE}(\mathcal{R}, t_{\mathcal{R}})) = \mathcal{R}$.*

6.1 Details on sequentialisation

We formally define the sequentialisation function.

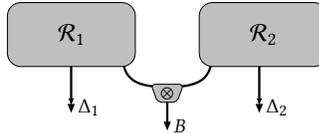
DEFINITION 28. *Let \mathcal{R} be an infinet and $t_{\mathcal{R}}$ a time-stamping which injectively time-stamps every maximal door in the \ll ordering by powers of two and every other door by ∞ . We define the pre-proof $\text{SEQUENTIALISE}(\mathcal{R}, t_{\mathcal{R}})$ is defined in Figure 13, SEQUENTIALISE being a corecursive function.*

REMARK 4. *Assuming one considers a fragment of infinet in which the necessary operations are computable (such as the transformation from $t_{\mathcal{R}}$ to $t_{\mathcal{R}'}$ for instance), the previous function can be turned into a corecursive algorithm, which is obviously not the case for the full set of infinets.*

function $\text{SEQUENTIALISE}(\mathcal{R}, t_{\mathcal{R}})$
 Choose a conclusion F in \mathcal{R} s.t. $t_{\mathcal{R}}(F) \neq \infty$.
if \mathcal{R} is an axiom link between F and some G **then**
 return $\frac{}{\vdash F, G}$ (ax)
else if $F = G \wp H$ **then**
 \mathcal{R}' is \mathcal{R} with \wp node above F removed and new conclusions, G and H .
 $t_{\mathcal{R}'}(x) = \begin{cases} t_{\mathcal{R}}(x) & \text{if } x \in \Gamma \wedge t_{\mathcal{R}}(x) \neq \infty; \\ \infty & \text{if } x \text{ is not maximal in } \ll \text{ ordering;} \\ t & \text{if } t_{\mathcal{R}}(x) = \infty \vee x = G \vee x = H. \end{cases}$
 where t is a fresh time-stamp larger than $t_{\mathcal{R}}(F)$.
 return $\frac{\text{SEQUENTIALISE}(\mathcal{R}', t_{\mathcal{R}'})}{\vdash \Gamma, G \wp H}$ (\wp)
else if $F = G \otimes H$ **then**
 $\mathcal{R}_1, \mathcal{R}_2$ are the two infinets with conclusions $\Gamma \cup \{G\}$ and $\Delta \cup \{H\}$ respectively that we get by removing the \otimes node above F in \mathcal{R} . $t_{\mathcal{R}_1}$ and $t_{\mathcal{R}_2}$ are $t_{\mathcal{R}}$ extended with timestamps for G and H respectively. G (resp. H) is time-stamped by t larger than $t_{\mathcal{R}}(F)$ if it is maximal in \ll ordering in \mathcal{R}_1 (resp. \mathcal{R}_2) and by ∞ otherwise.
 return $\frac{\pi_1 \quad \pi_2}{\vdash \Gamma, \Delta, G \otimes H}$ (\otimes) with $\pi_1 = \text{SEQUENTIALISE}(\mathcal{R}_1, t_{\mathcal{R}_1})$ and $\pi_2 = \text{SEQUENTIALISE}(\mathcal{R}_2, t_{\mathcal{R}_2})$.
else if $F = \sigma X.G(X)$ **then**
 \mathcal{R}' is \mathcal{R} with σ node above F removed and new conclusion $G[F/X]$.
 $t_{\mathcal{R}'}$ is $t_{\mathcal{R}}$ with a fresh timestamp for $G[F/X]$ larger than $t_{\mathcal{R}}(F)$.
 return $\frac{\text{SEQUENTIALISE}(\mathcal{R}', t_{\mathcal{R}'})}{\vdash \Gamma, \sigma X.G(X)}$ (σ)
end if
end function

Fig. 13. The function SEQUENTIALISE

PROPOSITION 9. *Let \mathcal{R} be an infinet. If a door, F , of the form $G \otimes H$, is maximal in the \ll ordering, then there exists infinets, \mathcal{R}_1 and \mathcal{R}_2 with doors G and H respectively such that \mathcal{R} can be construed as:*



PROOF. We will prove the contrapositive. Let x be a non-splitting conclusion. Since x is not splitting there is a parr node such that there is a switching path from each premise of the parr node to each premise of x . Follow this parr node to a conclusion, y . Observe that $x \ll y$ and hence x is not maximal. \square

LEMMA 1. $\text{Deseq}(\text{SEQUENTIALISE}(\mathcal{R}, t_{\mathcal{R}}))$ is a sub-infinet of \mathcal{R} .

PROOF. Proposition 9 ensures the local sanity of each step of sequentialisation *i.e.* it ensures that we don't prematurely sequentialise a node. By proposition 5 and lock-freeness we have that if \mathcal{R} is an infinet then the maximal element in the kingdom ordering exists. It is easy to check that at each step \mathcal{R}' , $\mathcal{R}_1, \mathcal{R}_2$ (depending on the case) is an infinet. Hence if we assume that every iterative step is $O(1)$ (for example obtaining $t_{\mathcal{R}'}$ from $t_{\mathcal{R}}$), then SEQUENTIALISE is

¹³This labelling is arbitrary: any cofinite sequence in place of powers of two works.

productive and produces a pre-proof, $\pi_{\mathcal{R}}$. Since this pre-proof is made of elements of \mathcal{R} , $\text{Deseq}(\pi_{\mathcal{R}})$ is a sub-infinet of \mathcal{R} by construction. \square

LEMMA 2. *The time-stamping assigns a finite natural number to every occurrence of the infinet that one starts with after some iterations of the sequentialisation process.*

PROOF. We will prove by contradiction. Suppose there are nodes which are never assigned a finite natural number by the time stamping algorithm. Among them choose t such that it has the least distance from a conclusion in the original infinet.

Then after finite iterations of the sequentialisation process, it becomes the conclusion (otherwise we would have found a node with even a lesser distance). Since it is not assigned a finite number, it is not maximal in the kingdom ordering. Consider $S = \{s \mid t \ll s\}$. Since t is not maximal S is non-empty. By lock-freeness S is finite.

Now, choose any $s \in S$ and follow it to its conclusion, c . If c is splitting, then after finite iterations of the sequentialisation process, we have a proof structure, \mathcal{S} , that needs to be sequentialised such there is conclusion in \mathcal{S} which is a premise of c . We can continue like this until the conclusion of s (say t') is non-splitting (otherwise s will become eventually splitting and we continue the sequentialisation by choosing another element $s' \in S$). We note that $t \ll s \ll t'$ and t' is not maximal. We continue as before when we encountered a non-maximal node t . If it is possible to continue this procedure ad infinitum we create an infinite ascending chain in \ll contradicting lock-freeness. So, after finitely many iterations every element of S will be sequentialised and hence t will be maximal and will be assigned a finite time-stamp. \square

6.2 Correctness of non-simple proofs

We will now explain checking DR-correctness on general nwfps. We recursively construct the level $\lambda + 1$ orthogonal graph. Proposition 7 lifts to level λ orthogonal graphs for any ordinal λ . An infinite simple path in a level λ orthogonal graph corresponds to a level λ visitable path. Assuming that we have a level λ orthogonal graph, this helps us construct a level $\lambda + 1$ orthogonal graph à la definition 23.

DEFINITION 29. *Given a level λ orthogonal graph $G_{\lambda}^{sw}(\mathcal{R}) = (\Theta, [SW], \{E_i\}_{i \leq \lambda})$, the **level $\lambda + 1$ orthogonal graph** (denoted $G_{\lambda+1}^{sw}(\mathcal{R})$) is the undirected hybridgraph $(\Theta, [SW], \{E_i\}_{i \leq \lambda+1})$ such that for every end, $[\rho]$ of G_{λ}^{sw} , $\{\theta\} \cup S \in E_{\lambda+1}$ where $S \subseteq \Theta \cup [SW]$ is the set of all nodes appearing in every ray of $[\rho]$ and $\theta \in \Theta$ is the axiom containing v , the visitable end corresponding to $[\rho]$.*

Pure paths are defined as usual. Observe that the set of nodes is invariant in the construction from a level λ to a level $\lambda + 1$ i.e. only new hyperedges are added at each step. Hence this can be seen as a monotonic graph rewriting. Consequently, we have a graph in the limit which we simply call $G^{sw}(\mathcal{R})$.

DEFINITION 30. *A nwfps, \mathcal{R} , is said to be DR-correct if for all switchings, sw , between any two nodes of $G^{sw}(\mathcal{R})$ there is exactly one pure path.*

7 CUT-ELIMINATION FOR VALID INFINETS

We now provide the main result of this paper: cut-elimination on infinets. As discussed in section 1, validity is sufficient (but not necessary, see figs. 4d and 4e) for its productivity. We retain the notion of validity in the sequent calculus and simply lift definition 8 to nwfps.

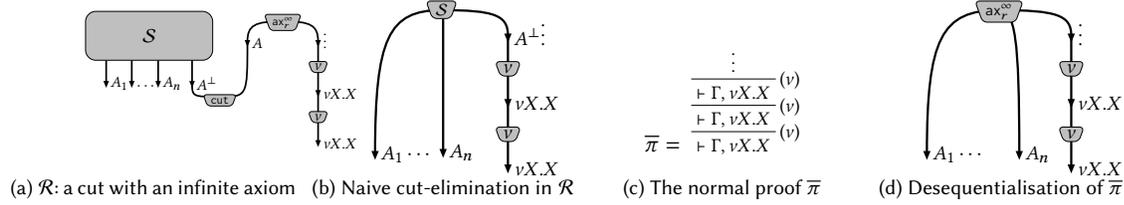


Fig. 14. Exhibiting the necessity of the kingdom erasure rule

$$\begin{aligned}
 \pi &= \frac{\frac{\pi'}{\vdash \Gamma, A^\perp} \quad \frac{\vdots}{\vdash \Gamma, vX.X, A}^{(v)} \quad \frac{\vdots}{\vdash \Gamma, vX.X, A}^{(v)}}{\vdash \Gamma, vX.X}^{(\text{Cut})} \sim \frac{\frac{\pi'}{\vdash \Gamma, A^\perp} \quad \frac{\vdots}{\vdash \Gamma, vX.X, A}^{(v)}}{\vdash \Gamma, vX.X}^{(\text{Cut})} \\
 &\sim \frac{\frac{\pi'}{\vdash \Gamma, A^\perp} \quad \vdots}{\vdash \Gamma, vX.X}^{(\text{Cut})} \sim \dots
 \end{aligned}$$

Fig. 15. A productive infinite reduction

DEFINITION 31. A NWFPS \mathcal{R} is **valid** if $\Theta_v = \emptyset$ and for $\theta \in \Theta_r$, there is $au \in \theta$ such that $\text{addr}(F_i) = \alpha$ and (F_i, v) is a v -formula for infinitely many prefixes v of u .

PROPOSITION 10. π is a proof iff $\text{Deseq}(\pi)$ is a valid infinit.

Cut-reduction rules. The cut-elimination procedure for infinets is adapted from MLL: during cut-elimination, finite axioms interact with cuts by annihilating one another, replaced by a wire. To understand the rule for infinite axioms, consider the infinet, \mathcal{R} , in fig. 14a. The straightforward adaptation of the finitary rule makes no sense, as it would result in reducing \mathcal{R} to the object in fig. 14b which is not an infinet: first, it requires to put a structure \mathcal{S} atop of an infinite path of v -cells; second, the types of this infinite path do not match¹⁴. To justify a better rule, let us see the situation in sequent calculus: consider a sequentialisation π as in fig. 15 of \mathcal{R} (where $\text{Deseq}(\pi') = \mathcal{S}$ and $\Gamma = \{A_1, \dots, A_n\}$).

The infinite axiom is represented in π by the infinite branch and the only way to make it interact with π' (in the way \mathcal{S} interacts with the infinite axiom) using the rules in definition 9 is by commuting the cut with one v -rule. Iterating such permutations builds the infinite sequence of proofs of fig. 15 which all desequentialise to $\bar{\pi}$.

This sequence converges to the proof in fig. 14c, where π' has been deleted and Γ is supported by the infinite branch. Desequentialised, this yields the proof-structure in fig. 14d. So, an infinitary axiom and a cut interact by removing the whole subinfinet “above” the cut. Now recall that the kingdom of an occurrence is the subinfinet that is always sequentialised above it. Hence, the subinfinet that has to be erased is indeed a kingdom. Although this operation will be represented by a single rule it does not correspond to one step of cut-elimination in the sequent calculus but to an infinite sequence of permutations. The complete set of cut-reduction rules are the ones illustrated in fig. 16 and

¹⁴Along the infinite v -ray, the types ought to remain equal to $vX.X$, but change to A^\perp above.

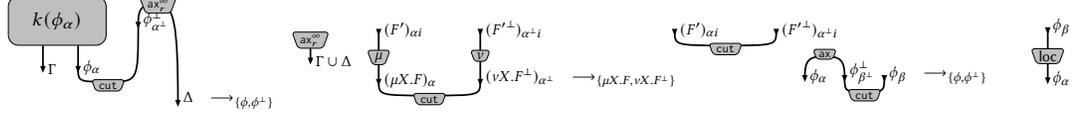


Fig. 16. Cut reduction rules for μMLL^∞ infinites. $k(\phi_\alpha)$ denotes the kingdom of the occurrence ϕ_α .

the usual \otimes/\otimes rule of MLL cut reduction. A sequence of infinites, $(\mathcal{R}_i)_{i \geq 0}$, is called a **reduction sequence** if for all i , $\mathcal{R}_i \rightarrow_\kappa \mathcal{R}_{i+1}$ for some cut κ in \mathcal{R}_i .

DEFINITION 32 (BABY-STEP CUT-REDUCTION RULES). *Let*

$$\mathcal{R} = (\{B_i^{U_i}\}_{i \in I}, \mathfrak{R}, \Theta_f, \Theta_r, V, \Theta_v, \text{loc})$$

be a non-canonical NWFPS. Suppose there exists $\{\phi_\alpha, \phi_\beta^\perp\} \in \mathfrak{R}$.

finite axiom if there exists $a \in \Theta_f$ such that $\phi_\alpha \in a$, then, $a = \{\phi_\alpha, \phi_{\alpha^\perp}^\perp\}$.

- loc is not defined on ϕ_β^\perp , as ϕ_β^\perp is part of a cut. Hence, we can consider the extension loc' of the loc function defined also on it, by $\text{loc}'(\phi_\beta^\perp) = \phi_{\alpha^\perp}^\perp$;
- ϕ_α is a conclusion, but its partial syntax tree is limited to itself;

Hence,

$$\mathcal{R}' = ((\{B_i^{U_i}\}_{i \in I} \setminus \{\phi_\alpha\}, \mathfrak{R} \setminus \{\{\phi_\alpha, \phi_\beta^\perp\}\}, \Theta_f \setminus \{\{\phi_\alpha, \phi_{\alpha^\perp}^\perp\}\}, \Theta_r, V, \Theta_v, \text{loc}')$$

is a non-canonical NWFPS: indeed, V is a set of equivalence classes of visitable paths in the level 0 approximant of \mathcal{R} and for any of these equivalence classes, an infinity of representatives are also visitable paths in the level 0 approximant of \mathcal{R}' . Here and after, we will not distinguish.

multiplicatives if $\phi = \psi \otimes \varphi$ and $\alpha \notin U_i, \beta \notin U_i$, then

- the partial syntax trees $B_{\phi_\alpha}^{U_{\phi_\alpha}}$ and $B_{\phi_\beta^\perp}^{U_{\phi_\beta^\perp}}$ are actually disjoint unions:

$$\begin{aligned} B_{\phi_\alpha}^{U_{\phi_\alpha}} &= B_{\psi_{\alpha,l}}^{U_{\psi_{\alpha,l}}} \cup B_{\varphi_{\alpha,r}}^{U_{\varphi_{\alpha,r}}} \cup \phi_\alpha \\ B_{\phi_\beta^\perp}^{U_{\phi_\beta^\perp}} &= B_{\psi_{\beta,r}^\perp}^{U_{\psi_{\beta,r}^\perp}} \cup B_{\varphi_{\beta,l}^\perp}^{U_{\varphi_{\beta,l}^\perp}} \cup \phi_\beta^\perp \end{aligned}$$

Hence,

$$\mathcal{B} = \{B_i^{U_i}\}_{i \in I} \setminus \{B_{\phi_\alpha}^{U_{\phi_\alpha}}, B_{\phi_\beta^\perp}^{U_{\phi_\beta^\perp}}\} \cup \{B_{\psi_{\alpha,l}}^{U_{\psi_{\alpha,l}}}, B_{\varphi_{\alpha,r}}^{U_{\varphi_{\alpha,r}}}, B_{\psi_{\beta,r}^\perp}^{U_{\psi_{\beta,r}^\perp}}, B_{\varphi_{\beta,l}^\perp}^{U_{\varphi_{\beta,l}^\perp}}\}$$

is a set of partial syntax trees.

- we set $\mathfrak{R}' = \mathfrak{R} \setminus \{\{\phi_\alpha, \phi_\beta^\perp\}\} \cup \{\{\psi_{\alpha,l}, \psi_{\beta,l}^\perp\}\} \cup \{\{\varphi_{\alpha,r}, \varphi_{\beta,r}^\perp\}\}$

Hence,

$$\mathcal{R}' = (\mathcal{B}, \mathfrak{R}', \Theta_f, \Theta_r, V, \Theta_v, \text{loc})$$

is a non-canonical NWFPS.

fix-point if $\phi = \mu X.F(X)$ and $\alpha \notin U_i, \beta \notin U_i$, then

- let

$$\mathcal{B} = \{B_i^{U_i}\}_{i \in I} \setminus \{B_{\phi_\alpha}^{U_{\phi_\alpha}}, B_{\phi_\beta^\perp}^{U_{\phi_\beta^\perp}}\} \cup \{B_{\phi_\alpha}^{U_{\phi_\alpha}} \setminus \{\phi_\alpha\}, B_{\phi_\beta^\perp}^{U_{\phi_\beta^\perp}} \setminus \{\phi_\beta^\perp\}\}$$

- the set $\mathfrak{R}' = \mathfrak{R} \setminus \{\{\phi_\alpha, \phi_\beta^\perp\}\} \cup \{\{F(\phi)_{\alpha,i}, (F(\phi)_{\beta,i}^\perp)\}\}$ is a set of pair of conclusions of \mathcal{B}

Hence,

$$\mathcal{R}' = (\mathcal{B}, \mathfrak{R}', \Theta_f, \Theta_r, V, \Theta_v, \text{loc})$$

is a non-canonical NWFPS.

infinite axiom if there exists a real infinite axiom $a \in \Theta_r$ such that $\alpha \in a$, then, let $K(\phi_\beta)$ be the kingdom of the occurrence ϕ_β . It is a non-canonical NWFPS:

$$K(\phi_\beta) = (\mathcal{B}^K, \mathfrak{R}^K, \Theta_f^K, \Theta_r^K, V^K, \Theta_v^K, \text{loc})$$

where $\mathcal{B}^K = \{B_{i,K}^{U_i}\}_{i \in J}$. All the formulæ $B_{i,K}$ are occurrences in \mathcal{R} . Let:

- $U_i(K)$ be the set of addresses in U_i that are suffixes of the addresses in \mathcal{B}^K . We set $U'_i = U_i \cup U_i(K)$ for all $i \in I$.
- $F'(K) = (K) \setminus \mathfrak{R}^K$
- $\Theta'_f = \Theta_f \setminus \Theta_f^K$
- $\Theta'_r = \Theta_r \setminus \Theta_r^K$
- $V' = V \setminus V^K$
- $\Theta' = \Theta \setminus \Theta_v^K$
- loc' is the restriction of loc to the axioms in Θ'_f

Hence,

$$\mathcal{R}' = (\{B_i^{U'_i}\}_{i \in I}, \mathfrak{R}', \Theta'_f, \Theta'_r, V', \Theta'_v, \text{loc}')$$

in a non-canonical NWFPS.

relocation if ϕ_α is a leaf and in the domain of loc , then let

- $\mathcal{B}' = \{B_i^{U_i}\}_{i \in I} \setminus \{\phi_\alpha\}$;
- $\mathfrak{R}' = \mathfrak{R} \setminus \{\{\phi_\beta^\perp, \phi_\alpha\}\} \cup \{\{\phi_\beta^\perp, \phi_\gamma\}\}$ where ϕ_γ is such that $\text{loc}(\phi_\alpha) = \phi_\gamma$;
- loc' is the restriction of loc to its domain without ϕ_α .

Hence,

$$\mathcal{R}' = (\mathcal{B}', \mathfrak{R}', \Theta_f, \Theta_r, V, \Theta_v, \text{loc}')$$

is a non-canonical NWFPS.

Else, if there exists ϕ_α and ϕ_β such that $\text{loc}(\phi_\alpha) = \phi_\beta$ and:

relocation-axiom there exists $a \in \Theta_f$ such that $\phi_\beta \in a$.

- $\mathcal{B}' = \{B_i^{U_i}\}_{i \in I} \setminus \{\phi_\beta\}$;
- let $\{\phi_\beta, \phi_\gamma^\perp\} = a$. We set $a' = \{\phi_\alpha, \phi_\gamma^\perp\}$ and $\Theta'_f = \Theta_f \setminus a \cup a'$.
- loc' is the restriction to loc undefined on ϕ_α .

Hence,

$$\mathcal{R}' = (\mathcal{B}', \mathfrak{R}, \Theta'_f, \Theta_r, V, \Theta_v, \text{loc}')$$

is a non-canonical NWFPS.

relocation-multiplicative if $\phi = \psi \otimes \varphi$ and $\beta \notin U_{\phi_\beta}$, then the partial syntax tree $B_{\phi_\beta}^{U_{\phi_\beta}}$ is actually the disjoint union:

$$B_{\phi_\beta}^{U_{\phi_\beta}} = B_{\psi_{\beta.l}}^{U_{\psi_{\beta.l}}} \cup B_{\varphi_{\beta.r}}^{U_{\varphi_{\beta.r}}} \cup \{\phi_\beta\}$$

- let $\mathcal{B}' = \{B_i^{U_i}\}_{i \in I} \setminus \{B_{\phi_\beta}^{U_{\phi_\beta}}\} \cup \{B_{\psi_{\beta.l}}^{U_{\psi_{\beta.l}}}, B_{\varphi_{\beta.r}}^{U_{\varphi_{\beta.r}}}\}$
- loc' is defined as loc undefined on ϕ_α extended with $\text{loc}(\psi_{\alpha.l}) = \psi_{\beta.l}$ and $\text{loc}(\varphi_{\alpha.r}) = \varphi_{\beta.r}$.

Hence,

$$\mathcal{R}' = (\mathcal{B}', \mathfrak{A}, \Theta_f, \Theta_r, V, \Theta_v, \text{loc}')$$

is a non-canonical NWFPS.

relocation-fixpoint if $\phi = \mu X.F(X)$ and $\beta \notin U_{\phi_\beta}$, then the partial syntax tree $B_{\phi_\beta}^{U_{\phi_\beta}}$ is actually of the form:

$$B_{\phi_\beta} = B_{F(\phi)_{\beta.i}}^{U_{F(\phi)_{\beta.i}}} \cup \{\phi_\beta\}$$

- let $\mathcal{B}' = \{B_i^{U_i}\}_{i \in I} \setminus \{B_{\phi_\beta}^{U_{\phi_\beta}}\} \cup \{B_{F(\phi)_{\beta.i}}^{U_{F(\phi)_{\beta.i}}}\}$
- loc' is defined as loc undefined on ϕ_α extended with $\text{loc}(F(\phi)_{\alpha.i}) = F(\phi)_{\beta.i}$.

Hence,

$$\mathcal{R}' = (\mathcal{B}', \mathfrak{A}, \Theta_f, \Theta_r, V, \Theta_v, \text{loc}')$$

is a non-canonical NWFPS.

relocation-infinite axiom if $\phi_\beta \in a \in \Theta_r$, then let $a' = a \setminus \{\phi_\beta\} \cup \{\phi_\alpha\}$ and loc' defined as equal to loc but undefined on ϕ_α ;

$$\mathcal{R}' = (\{B_i^{U_i}\}_{i \in I} \setminus \{\phi_\beta\}, \mathfrak{A}, \Theta_f, \Theta_r, V, \Theta_v, \text{loc}')$$

is a non-canonical NWFPS.

relocation-relocation if loc is defined on ϕ_β with $\text{loc}(\phi_\beta) = \phi_\gamma$, then we set

- loc' as the restriction of loc , undefined on ϕ_α and ϕ_β .

$$\mathcal{R}' = (\{B_i^{U_i}\}_{i \in I} \setminus \{\phi_\beta\}, \mathfrak{A}, \Theta_f, \Theta_r, V, \Theta_v, \text{loc}')$$

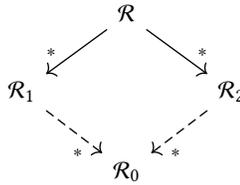
is a non-canonical NWFPS.

In all the cases above, we set

$$\mathcal{R} \rightarrow \mathcal{R}'.$$

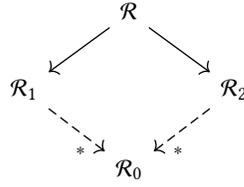
This is a baby-step reduction for NWFPS, as reducing it produces a non-canonical NWFPS. Nonetheless, we will prove after introducing limits of reduction sequences that, if \mathcal{R} is an NWFPS, reducing to \mathcal{R}_1 , a non-canonical NWFPS, there exists a reduction sequence involving only loc -rules such that its limit is an NWFPS.

PROPOSITION 11. *Cut-reduction on infinets is confluent.*



The proof of confluence leads us to consider a kingdom $k(F)$ as a generalized axiom, whose internal dynamics along cut-elimination does not interact with the exterior and can be merged with an infinite axiom: viewed in that light, there are only three reduction rules in infinets: reducing multiplicative cuts, reducing cuts on fixed-point formulas and cuts against generalized axioms (be they finite, infinite, or kingdoms) which erase the kingdom of one of the premise.

PROPOSITION 12. *Cut-reduction on infinets is locally confluent.*



PROOF. Let \mathcal{R} be an infinet; and $\mathcal{R} \rightarrow \mathcal{R}_1$ and $\mathcal{R} \rightarrow \mathcal{R}_2$ be two reductions. If none of the reductions are involving an infinitary axiom, the proof is as usual: if the redexes do not overlap, then the diamond can be closed, if they overlap, both of the reductions are of the form ax/cut, and hence, the reducts are already equal.

If one of the reduction reduces an infinite axiom against a cut, then, we have three possibilities:

- (1) the redexes are disjoint. In that case, local confluence follows from the same argument as above;
- (2) the reductions both are of the form infinite axiom vs. cut, and the infinite axiom is shared by the two redexes. In that case, in \mathcal{R}_1 , the kingdom above the other premise of one cut is erased and its conclusions are attached to the infinite axiom, while the same happens to the other cut in \mathcal{R}_2 . Closing the diamond is done by erasing the kingdom of the premise of the remaining cut in both cases;
- (3) the redexes overlap, but not by sharing an infinite axiom. In that case, one redex is actually included in the other: indeed, every redex is included in a kingdom.

□

Although not finite, the reduction relation is finite on finite NWFPS, and the reduction sequences can be rewritten as the composition of a reduction sequence on a finite NWFPS cut with an infinite constant one. Hence, the confluence.

Limits of reduction sequences. To prove that an infinite sequence of these reductions converges to some infinet, it is possible to define a topology on the set of infinets, which accounts for the cuts moving upwards during the cut-reduction procedure, by giving weights to cuts. One way to achieve it is to consider the heights of the cuts in a sequentialisation: basically, we use a sequentialisation to give a tree-like ordering to a proof-structure, and hence, a notion of distance compatible with the reduction. This method works for straight thread valid infinets, as we have theorem 1 for the sequent calculus.

Thus, infinitary cut-elimination is carried out in **valid** and **correct** NWFPS: correctness to use the tree topology of the sequentialisations (π and π' are at a distance $\leq 2^{-h}$ if they coincide up to height h); validity to ensure productivity.

As the reductions we introduced for infinets do not correspond to a single step of cut-reduction in the sequent calculus, we introduce a new reduction in the sequent calculus. We define the family of relations $\{\Rightarrow_h \mid h \in \mathbb{N}\}$ on μMLL^∞ proofs such that $\pi_0 \Rightarrow_h \pi'$ if the restrictions of π_0 and π' below height h coincide and

- either π' is the limit of an infinite sequence $(\pi_i)_{i \geq 0}$ such that for all $i \geq 0$, π_{i+1} is obtained from π_i by a permutation
- or there exists a finite sequence $(\pi_i)_{i \leq n}$ such that for all $i \leq n-1$, π_{i+1} is obtained from π_i by a permutation of an inference rule, and π' can be obtained from π_n by an external cut-reduction.

DEFINITION 33. *We say that $\pi \Rightarrow_h \pi'$ is a **sequentialisation of a reduction**, $\mathcal{R} \rightarrow_\kappa \mathcal{R}'$, if $\text{Deseq}(\pi) = \mathcal{R}$ and $\text{Deseq}(\pi') = \mathcal{R}'$ and h is maximal (i.e. for every $h' > h$, $\pi \not\Rightarrow_{h'} \pi'$).*

We also extend Def. 33 to define a sequentialisation of a reduction sequence.

LEMMA 3. *Let \mathcal{R}_0 be valid and $(\mathcal{R}_i)_{i \geq 0}$ be a reduction sequence. Every sequentialisation of $(\mathcal{R}_i)_{i \geq 0}$ has a limit which is a proof. Furthermore, the limits π and π' of two sequentialisations of the reduction sequence satisfy $\text{Deseq}(\pi) = \text{Deseq}(\pi')$.*

PROOF. We prove it by observing the following facts:

- for every reduction step, there exists a sequentialisation; we prove it by case analysis, by first noting that \mathcal{S}_0 is a valid infinet, and hence can be sequentialised: if the reduction is \otimes/\mathfrak{Y} , ν/μ or cut/ax , the cells in the proof-structure have corresponding rules in the sequent calculus proof. By doing some permutations, these rules can be put immediately above a cut rule. Among all these permutations, some are such that the untouched height is maximal: the ones that only do permutations above the cut.

For the $\infty\text{ax}/\text{cut}$ reductions, the reduction can be simulated by an infinite number of permutations above the cut rule in the sequent calculus proof, while not modifying anything below.

- given a sequentialisation of the reduction sequence

$$\begin{array}{ccccccc}
 \mathcal{S}_0 & \longrightarrow & \mathcal{S}_1 & \longrightarrow & \mathcal{S}_2 & \longrightarrow & \cdots & \longrightarrow & \mathcal{S}_k & \longrightarrow & \cdots \\
 \uparrow & & \uparrow & & \uparrow & & & & \uparrow & & \\
 \pi_0 & \xRightarrow{h_0} & \pi_1 & \xRightarrow{h_1} & \pi_2 & \xRightarrow{h_2} & \cdots & \xRightarrow{h_{k-1}} & \pi_k & \xRightarrow{h_k} & \cdots
 \end{array}$$

we can take the sequence $(h_i)_{i \geq 0}$ as going to the infinity.

Indeed, each reduction step of \Rightarrow_h is either a standard cut-reduction step preceded by a finite number of permutations or an infinite sequence of permutations that erases a subproof by pushing it above an infinite branch. We know [7, Theorem 28] that cut-elimination is productive for the cut-reduction in the sequent calculus, and adding the erasure of branches by pushing them along infinite cut does not perturbate this result.

The only thing to prove is that the permutations before any elimination does not mess with productivity. It is not the case, indeed: consider a lowest-height cut in π_0 . It will be eliminated at a reduction step i . As all the $(h_j)_{0 \leq j \leq i}$ are all maximal, no permutation in the sequence $(\pi_j)_{0 \leq j \leq i}$ have touched it, hence, the cut is untouched in π_i , and either deleted or replaced in π_{i+1} by a cut at higher height. As there are only finitely many cuts at this height h_i , there exists a j such that for all $k > j$, $h_k > h_i$.

- We still have to prove that two choices of initial sequentialisation π_0 and π'_0 yield two limits that desequentialise to the same infinet.

Let $H > 0$ be an height. As $(h_i)_{i > 0}$ goes to the infinity, there exists a $j > 0$ such that for all $k \geq j$, $h_k > H$. The rules below H in π_j are all permuted in π'_j and can be above h'_j : they have no reason to be stabilized. Nonetheless, there exists a $j' \geq j$ such that they are stabilized. Hence, the image of the desequentialisation of the restriction of π_j to height H is included (as a sub-NWFPs) in the image of the desequentialisation of the restriction of $\pi'_{j'}$ to height $h'_{j'}$.

So, ultimately, the two limits are equal.

□

DEFINITION 34. *Let \mathcal{R}_0 be valid and $(\mathcal{R}_i)_{i \geq 0}$ be a reduction sequence. The **limit** of $(\mathcal{R}_i)_{i \geq 0}$ is the desequentialisation of the limit of a sequentialisation of $(\mathcal{R}_i)_{i \geq 0}$.*

Fair sequences. Just as in the sequent calculus, not every reduction sequences converge towards a cut-free infinet: it is possible to never reduce some cuts.

DEFINITION 35. A reduction sequence $(\mathcal{R}_i)_{i \geq 0}$ is **fair** if for every $i \geq 0$ and r such that $\mathcal{R}_i \rightarrow_r \mathcal{R}'$, there is some $j \geq i$, such that r cannot be reduced in \mathcal{R}_j , i.e. there is no infinit, \mathcal{R}'' such that $\mathcal{R}_j \rightarrow_r \mathcal{R}''$.

EXAMPLE 9. Consider the desequentialisation of the pre-proofs in figs. 4d and 4e and the steps of cut reduction on it in fig. 17 using the rules in fig. 16. Observe it closely resembles the reduction sequence in example 3.

THEOREM 3. Let \mathcal{R}_0 be a valid infinit and $(\mathcal{R}_i)_{i \geq 0}$ be a fair reduction sequence. Its limit is a valid cut-free infinit.

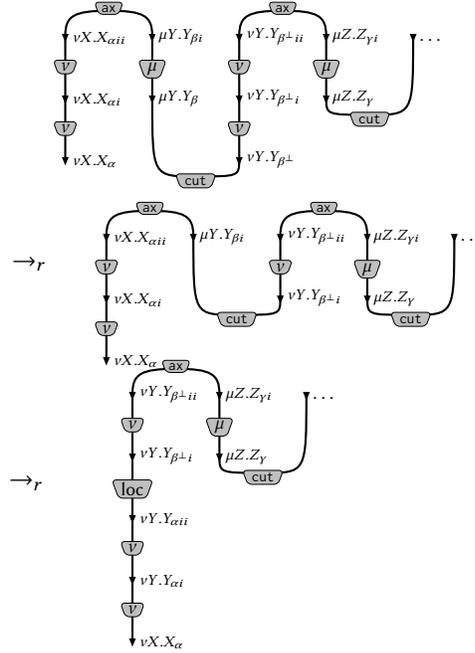


Fig. 17. Cut reduction on the desequentialisation of the pre-proofs in figs. 4d and 4e

8 CONCLUSION AND FUTURE WORK

We have developed the parallel syntax of non-wellfounded proof theory by generalizing the non-wellfounded proof structures in [21] to account for the presence of infinitely many cuts; and provided a cut-elimination result on these structures. Indeed the requirement of containing only finitely many cuts in [21] prevents to consider any circular pre-proof having a cut between the target and the source of a back-edge, as in fig. 4e. With our present generalization and confluent cut-elimination, we hope to contribute to a better understanding of non-wellfounded proofs, from both a syntactical and semantical point of view.

In future work we plan to improve our cut-elimination result, express more flexible validity conditions and understand what are circular infinits. We sketch few ideas.

Bouncing-valid infinits and cut-elimination. The motivation for investigating proof nets for μMLL^∞ is cut-elimination in bouncing-valid proof objects, for which the situation is not satisfactory: on the one hand, we have no criterion for proof-nets; on the other hand, criteria for the sequent calculus are not as strong as they could be. This work on cut-elimination for straight-thread valid proof-nets is a good stepping stone: our technique for defining the limit of a

sequence of reduction is independent from the validity criterion—we are just unable for the time being to characterize correctly its domain of validity, which should correspond to a notion of bouncing-validity. A further approach will be to investigate other cut-elimination proof methods developed in the infinitary setting, for instance Mints’ *continuous cut-elimination* [35].

Finitely presentable (f.p.) infinets. Regular (or circular) pre-proofs are those that can be represented by back-edges: they are a f.p. fragment of general pre-proofs. Algorithmically such structures are interesting: they allow one to explore decidability questions (checking validity for example). Thus a meaningful formulation of *f.p.* infinets is worth exploring. Other than decidability questions, there are several interesting directions in this endeavour:

- (1) Firstly, can the class of infinets that are the desequentialisation of some circular pre-proofs be characterized? Let us call this class *circular infinets*.
- (2) Further, this raises the question whether the Brotherston-Simpson conjecture can be addressed via proof nets. In other words if we define proof nets for finitary μ MLL with Park’s rules, the connection between these proof nets and circular infinets could shed new light on finitization techniques.
- (3) We can imagine having a fragment of *f.p.* infinets, called regular infinets, by considering finitely many cuts, regular partial syntax trees and the set of axioms given by a regular transducer. We conjecture that circular infinets \subseteq regular infinets.
- (4) Circularity of proofs is not preserved by cut-elimination. Hence, if it turns out that cut-elimination in regular infinets is decidable and produces regular infinets, then this partially culminates the quest for the class of *f.p.* proof objects of μ MALL[∞] that are closed under cut-elimination.

REFERENCES

- [1] Martín Abadi, Luca Cardelli, Pierre-Louis Curien, and Jean-Jacques Lévy. Explicit substitutions. *Journal of Functional Programming*, 1(4):375–416, 1991.
- [2] Andreas Abel and Brigitte Pientka. Wellfounded recursion with copatterns: a unified approach to termination and productivity. In Greg Morrisett and Tarmo Uustalu, editors, *ACM SIGPLAN International Conference on Functional Programming, ICFP’13, Boston, MA, USA - September 25 - 27, 2013*, pages 185–196. ACM, 2013.
- [3] Bahareh Afshari, Gerhard Jäger, and Graham E. Leigh. An infinitary treatment of full mu-calculus. In Rosalie Iemhoff, Michael Moortgat, and Ruy de Queiroz, editors, *Logic, Language, Information, and Computation*, pages 17–34, Berlin, Heidelberg, 2019. Springer Berlin Heidelberg.
- [4] Jürgen Avenhaus, Ulrich Kühler, Tobias Schmidt-Samoa, and Claus-Peter Wirth. How to prove inductive theorems? QUODLIBET! In *Automated Deduction - CADE-19, 19th International Conference on Automated Deduction Miami Beach, FL, USA, July 28 - August 2, 2003, Proceedings*, volume 2741 of *Lecture Notes in Computer Science*, pages 328–333. Springer, 2003.
- [5] 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, TABLEUX 2009, Oslo, Norway, July 6-10, 2009. Proceedings*, volume 5607 of *Lecture Notes in Computer Science*, pages 93–107. Springer, 2009.
- [6] David Baelde, Amina Doumane, Denis Kuperberg, and Alexis Saurin. Bouncing threads for circular and non-wellfounded proofs. <https://arxiv.org/abs/2005.08257>, 2020.
- [7] 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.
- [8] 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.
- [9] Marc Bagnol, Amina Doumane, and Alexis Saurin. On the dependencies of logical rules. In *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*, pages 436–450, 2015.
- [10] G. Bellin and J. van de Wiele. Subnets of proof-nets in MLL’. In *Proceedings of the Workshop on Advances in Linear Logic*, page 249–270, USA, 1995. Cambridge University Press.

- [11] James Brotherston. Cyclic proofs for first-order logic with inductive definitions. In Bernhard Beckert, editor, *Automated Reasoning with Analytic Tableaux and Related Methods*, pages 78–92, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.
- [12] James Brotherston, Nikos Gorogiannis, and Rasmus Lerchedahl Petersen. A generic cyclic theorem prover. In *Programming Languages and Systems - 10th Asian Symposium, APLAS 2012, Kyoto, Japan, December 11-13, 2012. Proceedings*, volume 7705 of *Lecture Notes in Computer Science*, pages 350–367. Springer, 2012.
- [13] James Brotherston and Alex Simpson. Sequent calculi for induction and infinite descent. *Journal of Logic and Computation*, 21(6):1177–1216, 10 2010.
- [14] J. Richard Buchi and Lawrence H. Landweber. Solving sequential conditions by finite-state strategies. *Transactions of the American Mathematical Society*, 138:295–311, 1969.
- [15] Alan Bundy. The automation of proof by mathematical induction. In *Handbook of Automated Reasoning (in 2 volumes)*, pages 845–911. Elsevier and MIT Press, 2001.
- [16] Alonzo Church and J. B. Rosser. Some properties of conversion. *Transactions of the American Mathematical Society*, 39(3):472–482, 1936.
- [17] Pierre-Louis Curien. Introduction to linear logic and ludics, part ii, 2006.
- [18] Anupam Das. Structure vs. invariants in proofs: project announcement. Talk at CiSS-19, <http://www.cse.chalmers.se/~bahafs/CiSS2019/programme.html>, 2019.
- [19] Anupam Das, Amina Doumane, and Damien Pous. Left-handed completeness for kleene algebra, via cyclic proofs. In *LPAR*, volume 57 of *EPIc Series in Computing*, pages 271–289. EasyChair, 2018.
- [20] Anupam Das and Damien Pous. A cut-free cyclic proof system for kleene algebra. In Renate A. Schmidt and Cláudia Nalon, editors, *Automated Reasoning with Analytic Tableaux and Related Methods*, pages 261–277, Cham, 2017. Springer International Publishing.
- [21] Abhishek De and Alexis Saurin. Infinets: The parallel syntax for non-wellfounded proof-theory. In Serenella Cerrito and Andrei Popescu, editors, *TABLEAUX 2019*, pages 297–316. Springer International Publishing, 2019.
- [22] Farzaneh Derakhshan and Frank Pfenning. Circular Proofs as Session-Typed Processes: A Local Validity Condition. *arXiv e-prints*, page arXiv:1908.01909, August 2019.
- [23] Simon Docherty and Reuben N. S. Rowe. A non-wellfounded, labelled proof system for propositional dynamic logic. In Serenella Cerrito and Andrei Popescu, editors, *Automated Reasoning with Analytic Tableaux and Related Methods*, pages 335–352, Cham, 2019. Springer International Publishing.
- [24] Amina Doumane. Constructive completeness for the linear-time μ -calculus. In *LICS*, pages 1–12. IEEE Computer Society, 2017.
- [25] 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.
- [26] Jérôme Fortier and Luigi Santocanale. Cuts for circular proofs: semantics and cut-elimination. In *CSL*, 2013.
- [27] Eduardo Giménez. Structural recursive definitions in type theory. In K. G. Larsen, S. Skyum, and G. Winskel, editors, *Proceedings 25th Int. Coll. on Automata, Languages and Programming, ICALP'98, Aalborg, Denmark, 13–17 July 1998*, volume 1443 of *LNCS*, pages 397–408. Springer-Verlag, Berlin, 1998.
- [28] Jean-Yves Girard. Linear logic. *Theor. Comput. Sci.*, 50(1):1–102, January 1987.
- [29] Jean-Yves Girard. Proof-nets: The parallel syntax for proof-theory. In *Logic and Algebra*, pages 97–124. Marcel Dekker, 1996.
- [30] Farzad Jafarrahmani. Denotational semantics of linear logic with least and greatest fixpoint. Master’s thesis, Université Paris Diderot, 2018.
- [31] Gary A. Kildall. A unified approach to global program optimization. In *Proceedings of the 1st Annual ACM SIGACT-SIGPLAN Symposium on Principles of Programming Languages, POPL '73*, page 194–206, New York, NY, USA, 1973. Association for Computing Machinery.
- [32] Dexter Kozen. Results on the propositional μ -calculus. *Theoretical Computer Science*, 27(3):333 – 354, 1983. Special Issue Ninth International Colloquium on Automata, Languages and Programming (ICALP) Aarhus, Summer 1982.
- [33] Per Martin-Löf. An intuitionistic theory of types: Predicative part. In H.E. Rose and J.C. Shepherdson, editors, *Logic Colloquium '73*, volume 80 of *Studies in Logic and the Foundations of Mathematics*, pages 73 – 118. Elsevier, 1975.
- [34] Ralph Matthes. Monotone fixed-point types and strong normalization. In Georg Gottlob, Etienne Grandjean, and Katrin Seyr, editors, *Computer Science Logic*, pages 298–312, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg.
- [35] Grigori E Mints. Finite investigations of transfinite derivations. *Journal of Soviet Mathematics*, 10(4):548–596, 1978.
- [36] J. F. Nash. Equilibrium points in N -person games. *Proceedings of the National Academy of Sciences of the United States of America*, 36(48-49), 1950.
- [37] Damian Niwiński. Fixed point characterization of infinite behavior of finite-state systems. *Theor. Comput. Sci.*, 189(1–2):1–69, December 1997.
- [38] Dag Prawitz. *Natural Deduction: A Proof-Theoretical Study*. Dover Publications, 1965.
- [39] Martin Protzen. Lazy generation of induction hypotheses. In Alan Bundy, editor, *Automated Deduction — CADE-12*, pages 42–56, Berlin, Heidelberg, 1994. Springer Berlin Heidelberg.
- [40] 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.
- [41] Dana Scott. Outline of a mathematical theory of computation. Technical Report PRG02, OUCI, November 1970.

$$\begin{array}{c}
\frac{\frac{C}{\vdash \Delta, F' \quad \vdash \Gamma, G'}{\vdash \Delta, \Gamma, F' \otimes G'}^{(\otimes)} \quad \text{mcut}(\iota, \perp\!\!\!\perp)}{\vdash \Sigma_\Delta, \Sigma_\Gamma, F \otimes G} \quad \xrightarrow{r} \quad \frac{\frac{C_\Delta \quad \vdash \Delta, F'}{\vdash \Sigma_\Delta, F} \text{mcut}(\iota', \perp\!\!\!\perp) \quad \frac{C_\Gamma \quad \vdash \Gamma, G'}{\vdash \Sigma_\Gamma, G} \text{mcut}(\iota'', \perp\!\!\!\perp)}{\vdash \Sigma_\Delta, \Sigma_\Gamma, F \otimes G}^{(\otimes)} \\
\\
\frac{C \quad \frac{\vdash \Delta, F', G'}{\vdash \Delta, F' \wp G'}^{(\wp)} \quad \text{mcut}(\iota, \perp\!\!\!\perp)}{\vdash \Sigma, F \wp G} \quad \xrightarrow{r} \quad \frac{C \quad \vdash \Delta, F', G'}{\vdash \Sigma, F, G} \text{mcut}(\iota', \perp\!\!\!\perp)}{\vdash \Sigma, F \wp G}^{(\wp)} \\
\\
\frac{C \quad \frac{\vdash \Delta, F'[\sigma X.F'/X]}{\vdash \Delta, \sigma X.F'}^{(\sigma)} \quad \text{mcut}(\iota, \perp\!\!\!\perp)}{\vdash \Sigma, \sigma X.F} \quad \xrightarrow{r} \quad \frac{C \quad \vdash \Delta, F'[\sigma X.F'/X]}{\vdash \Sigma, F[\sigma X.F/X]} \text{mcut}(\iota', \perp\!\!\!\perp)}{\vdash \Sigma, \sigma X.F}^{(\sigma)}
\end{array}$$

In the first reduction $((\otimes)/(\text{mcut}))$ we require that $\iota(F \otimes G) = F' \otimes G'$ and take ι' and ι'' that coincide with ι on Σ_Δ and Σ_Γ respectively, and such that $\iota'(F) = F'$ and $\iota''(G) = G'$. In the other reductions ι and ι' are similarly constrained.

Fig. 18. (mcut) Commutation reduction rules, where $r = (\text{ext}, F)$ and F is the principal occurrence.

A DETAILS ON CUT-ELIMINATION FOR μMLL^∞ SEQUENT CALCULUS

Multicut reduction rules. For the sake of completeness we will spell out the multicut reduction system for unit-free μMLL^∞ .

DEFINITION 36. Given sequents s, s_1, \dots, s_n where $n > 0$ and such that s_i, s_j are disjoint for all $i \neq j$, a **multicut** of conclusion s and premisses $(s_i)_{i \in [1;n]}$ is given by an injection $\iota : s \mapsto \cup_{i \in [1;n]} s_i$ and a binary relation $\perp\!\!\!\perp \subseteq (\cup_{i \in [1;n]} s_i)^2$ such that:

- For all $F \in s$, $\iota(F) \equiv F$.
- For all $F, G \in \cup_{i \in [1;n]} s_i$, $F \perp\!\!\!\perp G$ implies $F \equiv G^\perp$.
- $\text{dom}(\perp\!\!\!\perp) = (\cup_{i \in [1;n]} s_i) \setminus \text{im}(\iota)$.
- Given two sequents s_i and s_j , we say that they are $\perp\!\!\!\perp$ -connected on a pair of formula occurrences (F, G) when $F \in s_i$ and $G \in s_j$ such that $F \perp\!\!\!\perp G$. We say that they are $\perp\!\!\!\perp$ -connected, and we write $s_i \perp\!\!\!\perp s_j$, when they are $\perp\!\!\!\perp$ -connected on some F, G . The relation $\perp\!\!\!\perp$ on sequents must satisfy two conditions:
 - two sequents must be $\perp\!\!\!\perp$ -connected on at most one pair of occurrences F, G ;
 - the graph of the relation $\perp\!\!\!\perp$ must be connected and acyclic.

We write this multicut rule as:

$$\frac{s_1 \quad \dots \quad s_n}{s} \text{mcut}(\iota, \perp\!\!\!\perp)$$

DEFINITION 37. **Mcut commutation** are defined in fig. 18. In the first commutation rule, the sets C_Δ and C_Γ are the subsets of C which are respectively connected to Δ and Γ respectively. More precisely,

$$C_\Delta = \{s \mid \exists s', s \perp\!\!\!\perp^* s' \text{ and } s' \text{ is } \perp\!\!\!\perp\text{-connected to } \vdash \Delta, \Gamma, F \otimes G \text{ on an occurrence of } \Delta\},$$

where $\perp\!\!\!\perp^*$ is the symmetric transitive closure of the relation $\perp\!\!\!\perp$ on sequents. C_Γ is defined similarly.

$$\begin{array}{c}
\frac{C \quad \frac{\frac{\vdash \Delta, F \quad \vdash \Gamma, G}{\vdash \Delta, \Gamma, F \otimes G} (\otimes) \quad \frac{\vdash \Theta, F'^{\perp}, G'^{\perp}}{\vdash \Theta, F'^{\perp} \wp G'^{\perp}} (\wp)}{\vdash \Sigma} \text{mcut}(i, \perp)}{\vdash \Sigma} \\
\longrightarrow_r \quad \frac{C \quad \frac{\vdash \Delta, F \quad \vdash \Gamma, G \quad \vdash \Theta, F'^{\perp}, G'^{\perp}}{\vdash \Sigma} \text{mcut}(i, \perp')}{\vdash \Sigma}
\end{array}$$

where $F \otimes G \perp G'^{\perp} \wp G'^{\perp}$ and \perp' coincides with \perp except for $F \perp' F'^{\perp}$ and $G \perp' G'^{\perp}$

$$\begin{array}{c}
\frac{C \quad \frac{\frac{\vdash \Delta, F'[\mu X.F'/X]}{\vdash \Delta, \mu X.F'} (\mu) \quad \frac{\vdash \Gamma, F^{\perp}[\nu X.F^{\perp}/X]}{\vdash \Gamma, \nu X.F^{\perp}} (\nu)}{\vdash \Sigma} \text{mcut}(i, \perp)}{\vdash \Sigma} \\
\longrightarrow_r \quad \frac{C \quad \frac{\vdash \Delta, F'[\mu X.F'/X] \quad \vdash \Gamma, F^{\perp}[\nu X.F^{\perp}/X]}{\vdash \Sigma} \text{mcut}(i, \perp')}{\vdash \Sigma}
\end{array}$$

where $\mu X.F' \perp \nu X.F^{\perp}$ and \perp' coincides with \perp except for $F'[\mu X.F'/X] \perp' F^{\perp}[\nu X.F^{\perp}/X]$

Fig. 19. Principal reductions, where $r = (\text{princ}, \{F, F'^{\perp}\})$ with $\{F, F'^{\perp}\}$ the principal occurrences that have been reduced.

DEFINITION 38. **Internal reductions** are the **principal reductions** given in fig. 19 together with the following two reductions:

– the merge (mcut)/(Cut) reduction

$$\begin{array}{c}
\frac{C \quad \frac{\frac{\vdash \Delta, F \quad \vdash \Gamma, F^{\perp}}{\vdash \Delta, \Gamma} (\text{Cut})}{\vdash \Sigma} \text{mcut}(i, \perp)}{\vdash \Sigma} \\
\longrightarrow_r \quad \frac{C \quad \frac{\vdash \Delta, F \quad \vdash \Gamma, F^{\perp}}{\vdash \Sigma} \text{mcut}(i, \perp')}{\vdash \Sigma}
\end{array}$$

where \perp' extends \perp with $F \perp' F^{\perp}$ and $r = (\text{merge}, \{F, F^{\perp}\})$.

– the axiom reduction (mcut)/(Ax)

$$\begin{array}{c}
\frac{C \quad \frac{\overline{\vdash F, F'^{\perp}} (\text{Ax}) \quad \vdash F'', \Gamma}{\vdash \Sigma} \text{mcut}(i, \perp)}{\vdash \Sigma} \\
\longrightarrow_r \quad \frac{C \quad \frac{\vdash F'', \Gamma}{\vdash \Sigma} \text{mcut}(i', \perp')}{\vdash \Sigma}
\end{array}$$

where $r = (\text{CutAx}, \{F, F'^{\perp}\})$, $F'^{\perp} \perp F''$ and i', \perp' are defined as follows:

- for all $G \in \Sigma$, if $\iota(G) = F$ then $\iota'(G) = F''$, otherwise $\iota'(G) = \iota(G)$;
- $\perp' = \perp \setminus \{(F'^{\perp}, F'')\} \cup \{(F'', G) \mid (F, G) \in \perp\} \cup \{(G, F'') \mid (G, F) \in \perp\}$.

DEFINITION 39. A **reduction sequence** is a finite or infinite sequence $\sigma = (\pi_i, r_i)_{i \in 1+\lambda}$ with $\lambda \in \omega + 1$, where the π_i are μMLL_m^{∞} pre-proofs, the r_i are labels identifying multicut reduction rules and, for all $i \in \lambda$, $\pi_i \xrightarrow{r_i} \pi_{i+1}$. The sequence is **fair** if for all $i \in \lambda$ and r such that $\pi_i \xrightarrow{r} \pi'$ there is some $j \in \lambda$ such that $j \geq i$ and $\pi_j \xrightarrow{r} \pi_{j+1}$.

We can now state the multicut-elimination theorem for μMLL^{∞} [6, 7, 25]:

THEOREM 4 (μMLL^∞ (MULTI)CUT-ELIMINATION). *Fair reduction sequences on μMLL_m^∞ proofs produce cut-free μMLL^∞ proofs.*

Cut-elimination reductions for μMLL^∞ . We will detail the cut reduction rules and the cut-elimination theorem warts and all for μMLL^∞ avoiding the use of multicut. Some of the rules were given in section 3. The following is the full system.

DEFINITION 40 ((Cut)-(Cut) COMMUTATION).

$$\frac{\frac{\frac{\pi_1}{\Gamma, G^\perp} \quad \frac{\frac{\pi_2}{\vdash \Delta, F, G} \quad \frac{\pi_3}{\vdash \Sigma, F^\perp}}{\vdash \Delta, \Sigma, G} \text{ (Cut)}}{\vdash \Gamma, \Delta, \Sigma} \text{ (Cut)}}{\frac{\frac{\frac{\pi_1}{\Gamma, G^\perp} \quad \frac{\pi_2}{\vdash \Delta, F, G}}{\vdash \Gamma, \Delta, F} \text{ (Cut)} \quad \frac{\pi_3}{\vdash \Sigma, F^\perp}}{\vdash \Gamma, \Delta, \Sigma} \text{ (Cut)}}{r}}$$

with $r = (\text{cut} - \text{comm}, ((G^\perp, G), (F, F^\perp)))$, together with the other ways to associate the cuts, for instance:

$$\frac{\frac{\frac{\frac{\pi_1}{\vdash \Delta, F, G} \quad \frac{\pi_2}{\vdash \Sigma, F^\perp}}{\vdash \Delta, \Sigma, G} \text{ (Cut)} \quad \frac{\pi_3}{\Gamma, G^\perp}}{\vdash \Gamma, \Delta, \Sigma} \text{ (Cut)}}{\frac{\frac{\frac{\pi_1}{\vdash \Delta, F, G} \quad \frac{\pi_3}{\Gamma, G^\perp}}{\vdash \Gamma, \Delta, F} \text{ (Cut)} \quad \frac{\pi_2}{\vdash \Sigma, F^\perp}}{\vdash \Gamma, \Delta, \Sigma} \text{ (Cut)}}{r}}$$

and the like.

DEFINITION 41 (PRINCIPAL LOGICAL CASES FOR CUT-REDUCTION). *The rules are depicted in fig. 20, together with the symmetric rules, where $r = (\text{principal}, (C, C^\perp))$ with (C, C^\perp) the principal formulas that have been reduced.*

DEFINITION 42 (PRINCIPAL (Cut)/(Ax) CASE FOR CUT-REDUCTION).

$$\frac{\frac{\frac{}{\vdash F, G^\perp} \text{ (Ax)} \quad \frac{\pi}{\vdash G, \Gamma}}{\vdash F, \Gamma} \text{ (Cut)}}{\frac{\pi}{\vdash G, \Gamma}} \xrightarrow{r} \frac{\pi}{\vdash F, \Gamma} \text{ (Loc(i))}$$

with $F \equiv G$ and $\iota : \begin{cases} F \mapsto G \\ H \mapsto H \end{cases} \quad H \in \Gamma$ and $r = (\text{AXCut}, (F, G))$.

DEFINITION 43 (COMMUTATION OF LOGICAL RULES WITH (Cut)). *Together with the symmetric cases, the above reductions in fig. 21 define the cut commutation rules, where $r = (\text{ext}, F)$ and F is the formula occurrence that is principal after the rule application.*

$$\begin{array}{c}
\frac{\frac{\pi_1}{\vdash \Gamma, F} \quad \frac{\pi_2}{\vdash \Delta, G}}{\vdash \Gamma, \Delta, F \otimes G} \text{ } (\otimes) \quad \frac{\frac{\pi_3}{\vdash \Theta, G^\perp, F^\perp}}{\vdash \Theta, G^\perp \wp F^\perp} \text{ } (\wp)}{\vdash \Gamma, \Delta, \Theta} \text{ } (\text{Cut}) \quad \xrightarrow{r} \quad \frac{\frac{\pi_1}{\vdash \Gamma, F} \quad \frac{\frac{\pi_2}{\vdash \Delta, G} \quad \frac{\pi_3}{\vdash \Theta, G^\perp, F^\perp}}{\vdash \Delta, \Theta, F^\perp} \text{ } (\text{Cut})}}{\vdash \Gamma, \Delta, \Theta} \text{ } (\text{Cut}) \\
\\
\frac{\frac{\pi_1}{\vdash \Gamma, G^\perp, F^\perp} \text{ } (\wp)}{\vdash \Gamma, G^\perp \wp F^\perp} \text{ } (\wp) \quad \frac{\frac{\pi_2}{\vdash \Delta, F} \quad \frac{\pi_3}{\vdash \Theta, G}}{\vdash \Delta, \Theta, F \otimes G} \text{ } (\otimes)}{\vdash \Gamma, \Delta, \Theta} \text{ } (\text{Cut}) \quad \xrightarrow{r} \quad \frac{\frac{\frac{\pi_1}{\vdash \Gamma, G^\perp, F^\perp} \quad \frac{\pi_2}{\vdash \Delta, F}}{\vdash \Gamma, \Delta, G^\perp} \text{ } (\text{Cut}) \quad \frac{\pi_3}{\vdash \Theta, G}}{\vdash \Gamma, \Delta, \Theta} \text{ } (\text{Cut})}{} \\
\\
\frac{\frac{\pi_1}{\vdash \Gamma, F[\mu X.F/X]} \text{ } (\mu)}{\vdash \Gamma, \mu X.F} \text{ } (\mu) \quad \frac{\frac{\pi_2}{\vdash \Delta, F^\perp[\nu X.F^\perp/X]} \text{ } (\nu)}{\vdash \Delta, \nu X.F^\perp} \text{ } (\nu)}{\vdash \Gamma, \Delta} \text{ } (\text{Cut}) \quad \xrightarrow{r} \quad \frac{\frac{\pi_1}{\vdash \Gamma, F[\mu X.F/X]} \quad \frac{\pi_2}{\vdash \Delta, F^\perp[\nu X.F^\perp/X]}}{\vdash \Gamma, \Delta} \text{ } (\text{Cut}) \\
\\
\frac{\frac{\pi_1}{\vdash \Gamma, F[\nu X.F/X]} \text{ } (\nu)}{\vdash \Gamma, \nu X.F} \text{ } (\nu) \quad \frac{\frac{\pi_2}{\vdash \Delta, F^\perp[\mu X.F^\perp/X]} \text{ } (\mu)}{\vdash \Delta, \mu X.F^\perp} \text{ } (\mu)}{\vdash \Gamma, \Delta} \text{ } (\text{Cut}) \quad \xrightarrow{r} \quad \frac{\frac{\pi_1}{\vdash \Gamma, F[\nu X.F/X]} \quad \frac{\pi_2}{\vdash \Delta, F^\perp[\mu X.F^\perp/X]}}{\vdash \Gamma, \Delta} \text{ } (\text{Cut})
\end{array}$$

Fig. 20. Principal logical cases for cut-elimination

$$\begin{array}{c}
\frac{\frac{\frac{\pi_1}{\vdash \Gamma, F[\mu X.F/X], G} \text{ } (\mu)}{\vdash \Gamma, \mu X.F, G} \text{ } (\text{Cut}) \quad \frac{\pi_2}{\vdash G^\perp, \Delta}}{\vdash \Gamma, \mu X.F, \Delta} \text{ } (\text{Cut}) \quad \xrightarrow{r} \quad \frac{\frac{\frac{\pi_1}{\vdash \Gamma, F[\mu X.F/X], G} \quad \frac{\pi_2}{\vdash G^\perp, \Delta}}{\vdash \Gamma, F[\mu X.F/X], \Delta} \text{ } (\text{Cut}) \quad \frac{\pi_3}{\vdash \Gamma, \mu X.F, \Delta} \text{ } (\mu)}{\vdash \Gamma, \mu X.F, \Delta} \text{ } (\text{Cut}) \\
\\
\frac{\frac{\frac{\pi_1}{\vdash \Gamma, F[\nu X.F/X], G} \text{ } (\nu)}{\vdash \Gamma, \nu X.F, G} \text{ } (\text{Cut}) \quad \frac{\pi_2}{\vdash G^\perp, \Delta}}{\vdash \Gamma, \nu X.F, \Delta} \text{ } (\text{Cut}) \quad \xrightarrow{r} \quad \frac{\frac{\frac{\pi_1}{\vdash \Gamma, F[\nu X.F/X], G} \quad \frac{\pi_2}{\vdash G^\perp, \Delta}}{\vdash \Gamma, F[\nu X.F/X], \Delta} \text{ } (\text{Cut}) \quad \frac{\pi_3}{\vdash \Gamma, \nu X.F, \Delta} \text{ } (\nu)}{\vdash \Gamma, \nu X.F, \Delta} \text{ } (\text{Cut}) \\
\\
\frac{\frac{\frac{\pi_1}{\vdash \Gamma, F} \quad \frac{\pi_2}{\vdash \Delta, H, G}}{\vdash \Gamma, \Delta, H, F \otimes G} \text{ } (\otimes) \quad \frac{\pi_3}{\vdash \Sigma, H^\perp}}{\vdash \Gamma, \Delta, \Sigma, F \otimes G} \text{ } (\text{Cut}) \quad \xrightarrow{r} \quad \frac{\frac{\frac{\pi_1}{\vdash \Gamma, F} \quad \frac{\frac{\pi_2}{\vdash \Delta, H, G} \quad \frac{\pi_3}{\vdash \Sigma, H^\perp}}{\vdash \Delta, \Sigma, G}}{\vdash \Gamma, \Delta, \Sigma, F \otimes G} \text{ } (\otimes)}{\vdash \Gamma, \Delta, \Sigma, F \otimes G} \text{ } (\text{Cut}) \\
\\
\frac{\frac{\frac{\pi_1}{\vdash \Gamma, H^\perp} \quad \frac{\pi_2}{\vdash \Delta, F, G, H} \text{ } (\wp)}{\vdash \Gamma, H^\perp \wp \Delta, F, G, H} \text{ } (\text{Cut}) \quad \frac{\pi_3}{\vdash \Gamma, \Delta, F \wp G}}{\vdash \Gamma, \Delta, F \wp G} \text{ } (\text{Cut}) \quad \xrightarrow{r} \quad \frac{\frac{\frac{\pi_1}{\vdash \Gamma, H^\perp} \quad \frac{\pi_2}{\vdash \Delta, F, G, H}}{\vdash \Gamma, \Delta, F, G} \text{ } (\text{Cut}) \quad \frac{\pi_3}{\vdash \Gamma, \Delta, F \wp G} \text{ } (\wp)}{\vdash \Gamma, \Delta, F \wp G} \text{ } (\text{Cut})
\end{array}$$

Fig. 21. Commutations of cuts and logical rules

DEFINITION 44 ((Loc)-COMMUTATION RULES). *Together with the symmetric cases, the reductions in fig. 22 define the (Loc)-commutation rules.*

NOTATION 1. *We use the following notations:*

- \longrightarrow_m for the multicut reduction;
- \longrightarrow_c for the cut reduction (together with the Loc rules);
- $\longrightarrow_{\text{merge}}$ for the (mcut) merge reduction.

$$\begin{array}{c}
\frac{\pi}{\vdash \Gamma', F'[\mu X.F'/X]} \\
\frac{\vdash \Gamma', \mu X.F'}{\vdash \Gamma, \mu X.F} \text{ (Loc}(i))} \xrightarrow{r} \frac{\frac{\pi}{\vdash \Gamma', F'[\mu X.F'/X]} \text{ (Loc}(\kappa))}{\vdash \Gamma, \mu X.F} \text{ (}\mu\text{)} \\
\text{with } \iota \text{ such that } \iota(\mu X.F) = \mu X.F' \quad \text{and} \quad \iota(\Gamma) = \Gamma' \text{ and } \kappa = \begin{cases} G \mapsto \iota(G) & \text{if } G \in \Gamma \\ F[\mu X.F'/X] \mapsto F'[\mu X.F'/X] \end{cases}
\end{array}$$

$$\begin{array}{c}
\frac{\pi}{\vdash \Gamma', F'[\nu X.F'/X]} \\
\frac{\vdash \Gamma', \nu X.F'}{\vdash \Gamma, \nu X.F} \text{ (Loc}(i))} \xrightarrow{r} \frac{\frac{\pi}{\vdash \Gamma', F'[\nu X.F'/X]} \text{ (Loc}(\kappa))}{\vdash \Gamma, \nu X.F} \text{ (}\nu\text{)} \\
\text{with } \iota \text{ such that } \iota(\nu X.F) = \nu X.F' \quad \text{and} \quad \iota(\Gamma) = \Gamma' \text{ and } \kappa = \begin{cases} G \mapsto \iota(G) & \text{if } G \in \Gamma \\ F[\nu X.F'/X] \mapsto F'[\nu X.F'/X] \end{cases}
\end{array}$$

$$\begin{array}{c}
\frac{\frac{\pi_1}{\vdash \Gamma', F'} \quad \frac{\pi_2}{\vdash \Delta', G'}}{\vdash \Gamma', \Delta', F' \otimes G'} \text{ (}\otimes\text{)} \\
\frac{\vdash \Gamma, \Delta, F \otimes G}{\vdash \Gamma, \Delta, F \otimes G} \text{ (Loc}(i))} \xrightarrow{r} \frac{\frac{\frac{\pi_1}{\vdash \Gamma', F'} \text{ (Loc}(\kappa)) \quad \frac{\pi_2}{\vdash \Delta', G'} \text{ (Loc}(\lambda))}{\vdash \Gamma, \Delta, F \otimes G} \text{ (}\otimes\text{)}}{\vdash \Gamma, \Delta, F \otimes G} \\
\text{with } \kappa = \begin{cases} H \mapsto \iota(H) & \text{if } H \in \Gamma \\ F \mapsto F' \end{cases} \quad \text{and} \quad \lambda = \begin{cases} H \mapsto \iota(H) & \text{if } H \in \Delta \\ G \mapsto G' \end{cases}
\end{array}$$

$$\begin{array}{c}
\frac{\pi}{\vdash \Gamma', F', G'} \\
\frac{\vdash \Gamma', F' \wp G'}{\vdash \Gamma, F \wp G} \text{ (Loc}(i))} \text{ (}\wp\text{)} \xrightarrow{r} \frac{\frac{\pi}{\vdash \Gamma', F', G'} \text{ (Loc}(\kappa))}{\vdash \Gamma, F \wp G} \text{ (}\wp\text{)} \\
\text{with } \iota = \begin{cases} H \mapsto \iota(H) & \text{if } H \in \Gamma \\ F \mapsto F' \\ G \mapsto G' \end{cases}
\end{array}$$

$$\begin{array}{c}
\frac{\frac{\pi_1}{\vdash \Gamma', C} \quad \frac{\pi_2}{\vdash \Delta', C^\perp}}{\vdash \Gamma', \Delta'} \text{ (Cut)} \\
\frac{\vdash \Gamma, \Delta}{\vdash \Gamma, \Delta} \text{ (Loc}(i))} \xrightarrow{r} \frac{\frac{\frac{\pi_1}{\vdash \Gamma', C} \text{ (Loc}(\kappa)) \quad \frac{\pi_2}{\vdash \Delta', C^\perp} \text{ (Loc}(\lambda))}{\vdash \Gamma, \Delta} \text{ (Cut)}}{\vdash \Gamma, \Delta} \\
\text{with } \kappa = \begin{cases} H \mapsto \iota(H) & \text{if } H \in \Gamma \\ C \mapsto C \end{cases} \quad \text{and} \quad \lambda = \begin{cases} H \mapsto \iota(H) & \text{if } H \in \Delta \\ C^\perp \mapsto C^\perp \end{cases}
\end{array}$$

$$\begin{array}{c}
\frac{\pi}{\vdash \Gamma''} \text{ (Loc}(\kappa)) \xrightarrow{r} \frac{\pi}{\vdash \Gamma''} \text{ (Loc}(\kappa \circ i)) \\
\frac{\vdash \Gamma'}{\vdash \Gamma} \text{ (Loc}(i))
\end{array}
\qquad
\begin{array}{c}
\frac{\vdash F', G'}{\vdash F, G} \text{ (Ax)} \xrightarrow{r} \frac{\vdash F, G}{\vdash F, G} \text{ (Ax)} \\
\frac{\vdash \Gamma'}{\vdash \Gamma} \text{ (Loc}(i))
\end{array}$$

Fig. 22. Commutation of logical rules with relocations

– $\longrightarrow_{\text{comm}}$ for the (Cut) commutation reduction.

DEFINITION 45 ($[\pi]$). If $\pi \in \mu\text{MLL}_m^\infty$, $[\pi]$ is the proof where each unary (resp. binary) (mcut) has been replaced with a (Loc) (resp. a (Cut) and possibly a (Loc)) rule as follows:

- $\frac{s'}{s} (\text{mcut})(\iota, \emptyset)$ is replaced by $\frac{s'}{s} (\text{Loc}(\iota))$
- $\frac{s_1 \quad s_2}{s} (\text{mcut})(\iota, \perp)$ is replaced by $\frac{s_1 \quad s_2}{s} (\text{Cut})$ if $\iota = \text{id}$ and by $\frac{s_1 \quad s_2}{s'} (\text{Cut})$ otherwise, with $s' = \iota(s)$.

DEFINITION 46 ($\text{CSeq}(\pi_m)$). Given a μMLL_m^∞ proof π_m , we define $\text{CSeq}(\pi_m) = \{[\pi'_m] \mid \pi_m \longleftarrow_{\text{merge}}^* \pi'_m \not\longleftarrow_{\text{merge}}^*\}$

REMARK 5. While $[\pi] \in \mu\text{MLL}_m^\infty$ in general, elements of $\text{CSeq}(\pi_m)$ are actually all proofs in μMLL^∞ (ie without any multicut).

LEMMA 4. Let π_m be a μMLL_m^∞ proof. If $\pi \in \text{CSeq}(\pi_m)$, then every occurrence of a sequent in π_m also occurs in π .

PROOF. This is trivial as any merge antireduction does not change the premises nor the conclusion of the multicut involved in the merge rule and as μMLL^∞ proofs having no antecedent by merge have only binary and unary (mcut) inferences. \square

LEMMA 5. Let π_m be a μMLL_m^∞ proof and let r be an occurrence of (mcut) and s_1, s_2 be two sequents premises of r such that $s_1 \perp\!\!\!\perp s_2$, then there exists some $\pi \in \text{CSeq}(\pi_m)$ containing an occurrence of (Cut) of the form: $\frac{s_1 \quad s_2}{s} (\text{Cut})$

PROOF. Consider some $\pi_0 \in \text{CSeq}(\pi_m)$ such that the number n_c of (Cut) on a path from s_1 to s_2 is minimal. If n_c is equal to 1 we are done and if it is greater than one, we see by case analysis that a (Cut) -commutation case can reduce π_0 to some π_1 which is also in $\text{CSeq}(\pi_m)$ and distance between s_1 and s_2 less than n_c contradicting minimality from which we conclude.

The same proof method works by taking any element from $\text{CSeq}(\pi_m)$ and performing $\longrightarrow_{\text{comm}}$ reduction diminishing the distance between s_1 and s_2 . \square

LEMMA 6. Let $\pi_m \in \mu\text{MLL}_m^\infty$ and $\pi, \pi' \in \text{CSeq}(\pi_m)$, then $\pi \longrightarrow_{\text{comm}} \pi'$.

PROOF. This is a simple induction on the size of the (Cut) -tree (ie subtree of (Cut) inferences) resulting from the merge antireduction, which depends only on the multicut and not on the choice of an element in $\text{CSeq}(\pi_m)$: one can use $\longrightarrow_{\text{comm}}$ to move any cut to the root and reduce from π to π' . \square

PROPOSITION 13. If $\pi \in \mu\text{MLL}^\infty$ and $\pi_m \in \mu\text{MLL}_m^\infty$ containing at most one mcut per branch such that $\pi \in \text{CSeq}(\pi_m)$, if there exists π'_m such that $\pi_m \longrightarrow_m \pi'_m$, then there exists $\pi' \in \mu\text{MLL}^\infty$ such that $\pi \longrightarrow_c^{0/1} \pi'$ and $\pi' \in \text{CSeq}(\pi'_m)$.

PROOF. Simple case analysis: simply take $\pi' = \pi$ in case of a merge rule and otherwise, perform the corresponding cut-reduction, possibly beginning with (Cut) commutation rules to expose the redex and possibly followed with (Loc) for preserving the structure of the resulting proof. \square

The following lemma is trivial by analyzing all cases of (mcut) -reduction which all preserve this property of having at most one (mcut) per branch.

LEMMA 7. If $\pi_m \in \mu\text{MLL}_m^\infty$ has at most one (mcut) per branch and $\pi_m \longrightarrow_m \pi'_m$, then π'_m has at most one (mcut) per branch.

CONTENTS

Abstract	1
1 Introduction	1
2 Background	7
3 Cut-elimination in μ MLL $^\infty$ sequent calculus	8
4 Trips	9
5 Non-wellfounded Proof Structures	12
5.1 Treating non-simple proofs	15
5.2 An illustration of desequentialisation	17
5.3 Non-wellfounded substructures	17
5.4 Treating NWFPS with locs	18
6 Correctness Criterion	19
6.1 Details on sequentialisation	22
6.2 Correctness of non-simple proofs	24
7 Cut-elimination for valid infinets	24
8 Conclusion and Future Work	31
References	32
A Details on Cut-elimination for μ MLL $^\infty$ sequent calculus	34
Contents	41