Unification nets: canonical proof net quantifiers

Dominic J. D. Hughes
Stanford & UC Berkeley*

Abstract

Proof nets for MLL (unit-free Multiplicative Linear Logic) are concise graphical representations of proofs which are *canonical* in the sense that they abstract away syntactic redundancy such as the order of non-interacting rules. We argue that Girard's extension to MLL1 (first-order MLL) fails to be canonical because of redundant existential witnesses, and present canonical MLL1 proof nets called *unification nets* without them. For example, while there are infinitely many cut-free Girard nets $\forall x\, Px \vdash \exists x\, Px$, one per arbitrary witness for $\exists x$, there is a unique cut-free unification net, with no specified witness.

Cut elimination for unification nets is local and linear time, while Girard's is non-local and exponential time. Since some unification nets are exponentially smaller than corresponding Girard nets and sequent proofs, technical delicacy is required to ensure correctness is polynomial-time (quadratic).

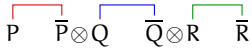
These results transcend MLL1 via a methodological insight: for canonical quantifiers, the standard *parallel/sequential* dichotomy of proof nets is insufficient; an *implicit/explicit witness* dichotomy is needed. Current work extends unification nets to additives and uses them to extend *combinatorial proofs* [*Proofs without syntax*, Annals of Mathematics, 2006] to classical first-order logic.

1 Introduction

Girard's elegant proof nets [Gir87, DR89] are concise graphical representations of proofs in MLL (unit-free Multiplicative Linear Logic). For example, the two MLL proofs

$$\dfrac{\dfrac{\overline{P},\overline{P} \quad \overline{Q},\overline{Q}}{P, \overline{P}\otimes Q, \overline{Q}} \otimes \quad \overline{R},\overline{R}}{P, \overline{P}\otimes Q, \overline{Q}\otimes R, \overline{R}}\otimes \qquad \dfrac{\overline{P},\overline{P} \quad \dfrac{\overline{Q},\overline{Q} \quad \overline{R},\overline{R}}{Q, \overline{Q}\otimes R, \overline{R}}\otimes}{P, \overline{P}\otimes Q, \overline{Q}\otimes R, \overline{R}}\otimes$$

translate to the same MLL proof net:



MLL proof nets are *canonical* in the sense that they abstract away syntactic redundancy such as the order of non-interacting rules. For example, the two proofs above differ only in the order they introduce non-interacting tensors $\overline{P}\otimes Q$ and $\overline{Q}\otimes R$; the proof net abstracts away this arbitrary choice. Such syntactic redundancies are not merely subjective aesthetic failures: as noted by Girard [Gir96], they burden sequent calculus cut elimination with endless mechanical rule commutations. By purging these commutations, cut elimination for MLL proof nets is local (each reduction being a local graph rewrite) and linear time (eliminating all cuts taking time linear in the size of the net). In contrast, cut elimination for MLL sequent calculus is non-local and at best quadratic.

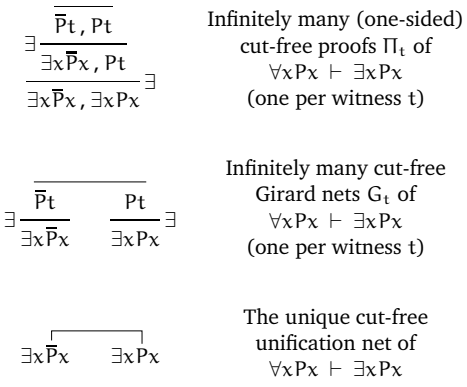


Figure 1. Illustrating unification net canonicity.

Girard extended MLL proof nets with quantifiers, to MLL1 (first-order MLL), over a series of three papers [Gir87, Gir88, Gir91]. He reiterated them in his book *The Blind Spot* [Gir11], choosing one for the cover picture, and characterizing them as "*The only really satisfactory extension of proof-nets*" [Gir11, Ch. 11].

At first glance, they do indeed appear satisfactory: like the MLL nets they extend, they abstract away the redundant order of non-interacting rules using *parallelism* [Gir96]. However, we argue that they fail to be canonical (hence fail to be satisfactory) due to redundant existential witnesses, inherited from sequent calculus. For example, consider $\forall x Px \vdash \exists x Px$, whose one-sided form is $\exists x \overline{P}x, \exists x Px$. Figure 1 (top) shows an infinite family of cut-free MLL1 proofs Π_t, one per existential witness term $t = z, a, f(z,a), f(g(z), h(a,b))$, etc. The choice of t is arbitrary, hence redundant. Correspondingly, there is an infinite family of cut-free Girard nets G_t (Figure 1 centre), one per witness term t, since Girard nets inherit redundant existential witnesses from sequent calculus.[1]

We present canonical MLL1 proof nets called *unification nets*, or *unets* for short, free of redundant existential witnesses. Fig. 1 (bottom) illustrates canonicity: in contrast to the infinite families of cut-free sequent proofs and Girard nets, there is a unique cut-free unification net of $\forall x Px \vdash \exists x Px$. Fig. 2 shows another comparison: an MLL1 proof with two axioms, the corresponding Girard net with two axiom links, and the corresponding unification net with two links ⌐¬. Unlike a Girard axiom link, a unification net link can go between atoms which are not strictly dual, such as $Q(h(z,a))$ and $\overline{Q}(y)$ in Fig. 2.

*I pursued this research as a Visiting Scholar at Stanford then Berkeley. I'm grateful to my hosts, Vaughan Pratt (Stanford Computer Science), Sol Feferman (Stanford Mathematics) and Wes Holliday (Berkeley Logic Group). Thanks to Marc Bagnol, Willem Heijltjes and Lutz Straßburger for valuable feedback, and to Dale Miller for inviting me to present this work at the LIX Colloquium 2013. In memoriam Sol Feferman (1928–2016).

[1] The [Gir96] variant introduces additional redundancy not even present in sequent calculus, due to explicit witness annotations $\exists_t$ even in the case of vacuous quantifiers, so by *Girard net* we shall always mean the [Gir91] and [Gir11] variant.

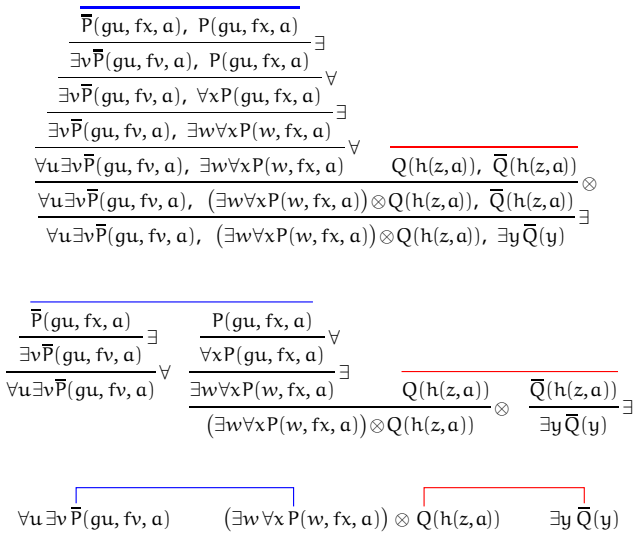


Figure 2. An MLL1 proof, its Girard net, and its unification net.

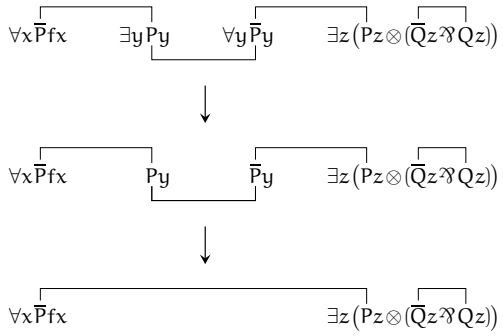


Figure 3. Unification net cut elimination is local.

1.1 Local linear-time cut elimination

Unification net cut elimination is local and linear time, while Girard's is non-local and exponential time and space. Fig. 3 shows the two-step cut elimination of a unification net with one cut ⌴. Each step is a purely local graph rewrite. Fig. 4 shows the corresponding Girard cut elimination. The first step is non-local, since it substitutes $\mathrm{f}x$ for y globally. Chaining such substitutions, each duplicating a term, causes exponential growth (see Appendix A).

1.2 Brevity

By leaving witnesseses implicit, unification nets are more concise than sequent proofs and Girard nets. Some sequents require exponentially large cut-free proofs and Girard nets (see App. B), *i.e.*, cut-free sequent proofs and Girard nets are not *polynomially bounded* [CR79]. In contrast, cut-free unification nets are only linearly larger than their underlying sequents, hence they are polynomially bounded (indeed linearly bounded).

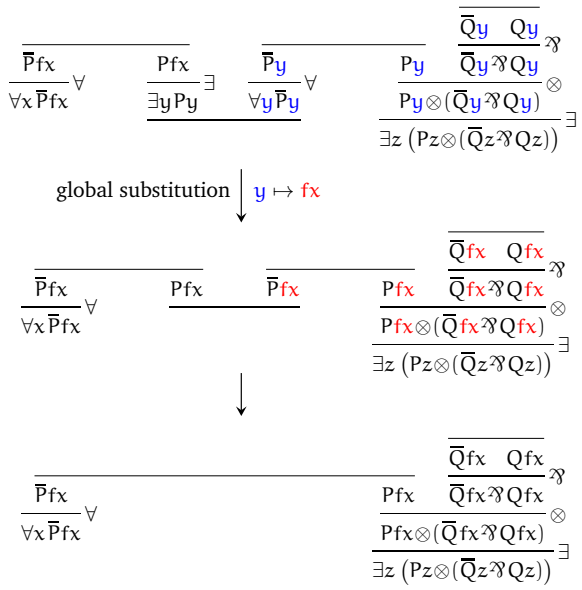


Figure 4. Girard net cut elimination is not local.

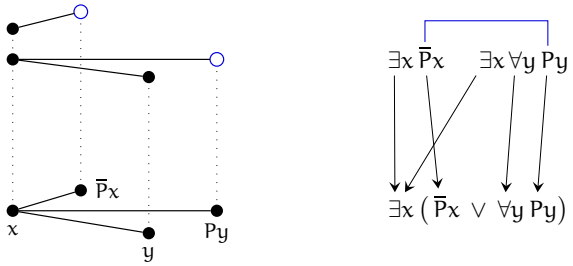
1.3 Beyond sequentialization

Fig. 5 summarizes the relationship between MLL1 sequent calculus, Girard nets, and unification nets. The lower-left corner, *unification calculus* (§6.2), is a variant of MLL1 sequent calculus in which, like unification nets, existential witnesses remain implicit; it was conceived to fill out the commuting square.

The east-west axis is the standard *parallel/sequential* dichotomy of proof nets [Gir96]: sequent calculus and unification calculus are sequential (*west*), with redundant order on non-interacting rules; Girard nets and unification nets are parallel (*east*), abstracting away this redundancy. The north-south axis is an *implicit/explicit witness* dichotomy: sequent calculus and Girard nets have redundant explicit witnesses (*north*); unification calculus and unification nets abstract away this redundancy by leaving witnesses implicit (*south*).

1.4 Combinatorial proofs for classical first-order logic

Proof without syntax [Hug06a] reformulated classical propositional logic in terms of *combinatorial proofs* rather than syntactic proofs. A key motivation for the present paper on unification nets was as a stepping stone towards extending combinatorial proofs to classical first-order logic (in preparation; see also [Hug14]). A first-order combinatorial proof of Smullyan's drinker paradox $\exists x(\mathrm{P}x \Rightarrow \forall y\,\mathrm{P}y)$ is shown below-left.



The lower graph abstracts the proved formula $\exists x(\mathrm{P}x \Rightarrow \forall y\,\mathrm{P}y)$, the upper graph abstracts a unification net, and the dotted

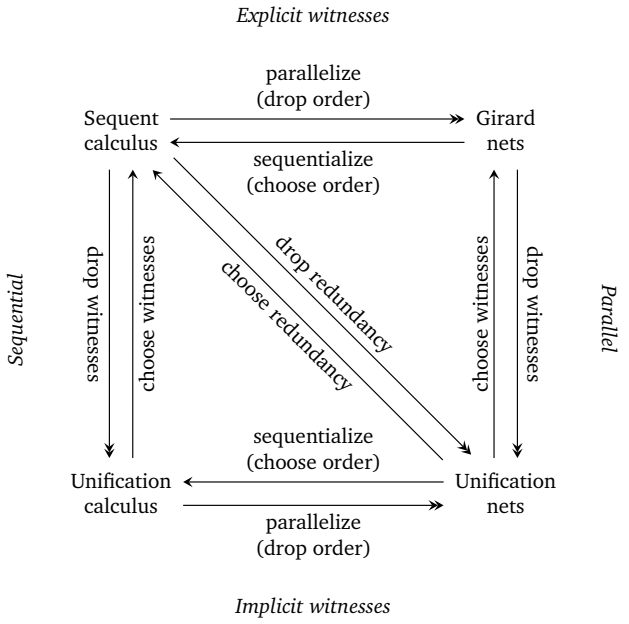


Figure 5. Relating MLL1 sequent calculus, Girard nets, and unification nets. Each double-headed arrow is a surjection between cut-free sub-systems. The diagram commutes from top-left to bottom-right: the surjection from cut-free MLL1 sequent calculus proofs to cut-free unification nets factorizes through both cut-free Girard nets (eliminating order redundancy first) and cut-free unification calculus (eliminating witness redundancy first).

lines indicate a *skew fibration*, a lax graph fibration. As in the original propositional case [Hug06b] (see also [Car10, Str17]), a skew fibration parallelizes all contraction and weakening in a proof. By using a *semi-combinatorial* style [Hug06b, §2.1], above-right, the upper unification net becomes more apparent. See [Hug14] for other first-order examples in semi-combinatorial style.

1.5 Extending unification nets to additives

Joint work in progress aims to extend unification nets to ALL1 (first-order Additive Linear Logic without units). Since the examples in Figure 1 involve no multiplicative connective, they are also additive; thus the unification net in Figure 1 is both multiplicative and additive.

1.6 Technical delicacy for p-time (quadratic) correctness

Since some unification nets are exponentially smaller than corresponding Girard nets and sequent proofs (App. B) technical delicacy is required to ensure that correctness is polynomial time. Theorem 3 (p. 6) shows that it is at worst quadratic time.

1.7 Canonicity Theorem

The two cut-free MLL1 proofs

$$\exists\dfrac{\exists\dfrac{\overline{\overline{P}a\,,\,Pa}}{\exists x\overline{P}x\,,\,Pa}}{\exists x\overline{P}x\,,\,\exists xPx}\exists \qquad\qquad \exists\dfrac{\exists\dfrac{\overline{\overline{P}fc\,,\,Pfc}}{\overline{P}fc\,,\,\exists xPx}\exists}{\exists x\overline{P}x\,,\,\exists xPx}$$

are *equivalent* in the sense that the left yields the right by commuting the order of the $\exists$ rules and replacing one arbitrary choice of existential witness, a, by another, fc. While they have distinct Girard nets (because Girard nets inherit redundant explicit witnesses), they have the same unification net (from Fig. 1): $\exists x\overline{\overline{P}x\ \exists xPx}$. In §4 we formalize this notion of proof equivalence and prove a *Canonicity Theorem* (Thm. 4, p. 7): two cut-free MLL1 proofs are equivalent if and only if they have the same unification net.

1.8 Related work

Unification in the context of first-order logic goes back to Herbrand's theorem [Her30]. Robinson's resolution [Rob65] is a seminal work. Our links between predicates which are not strictly dual (*e.g.* $\overline{P}x$ and Pfy) are akin to the first-order connections or matings employed in automated theorem proving [Bib81, And81]. In fact, Bibel in [Bib81, p. 4] coined *link* as an alternative term for a connection; we have adopted this terminology. The roots of first-order connections/matings with unification can be traced back to Quine [Qui55] and Prawitz [Pra70].

Our *leaps* from $\exists x$ vertices to $\forall y$ vertices play a similar role to Girard's *jumps* between $\forall y$ vertices and occurrences of witnesses containing y, but in a more rarefied context without explicit witnesses. Both leaps and jumps capture dependencies between $\forall$ rules and $\exists$ rules in a proof, and the interaction between tensors and quantifiers. Bellin and van de Wiele [BW95] add a condition on eigenvariables to Girard's MLL1 net definition [Gir91] to streamline kingdoms and empires. Since we leave witnesses implicit, and have no need for eigenvariables, we do not need an analogous condition.

Abstract representations of first-order quantifiers with explicit witnesses for classical logic have been presented by Heijltjes [Hei10] (extending Miller's expansion trees [Mil84]) and McKinley [McK10]. Straßburger presents proof nets for second-order MLL in [Str09]. First-order proof nets with explicit witnesses are employed in linguistic analysis, for example, [Moo02]; it would be interesting to see if using unification nets could lead to simplification.

2 MLL1

As in [Gir91], we work with MLL1 (first-order Multiplicative Linear Logic without units). We adopt the following conventions: term variables x, y, z; n-ary function symbols f, g, h ($n \geqslant 1$); constants (0-ary function symbols) a, b, c; terms s, t, u; n-ary predicate symbols P, Q, R ($n \geqslant 0$); formulas A, B, C; sequents Γ, Δ, Σ. Fix an arity-preserving ***negation*** or ***duality*** function $\overline{\ }$ on predicate symbols such that $\overline{\overline{P}} = P$ and $\overline{P} \neq P$ for all P. A ***predicate*** or ***atom*** is an expression $Pt_1 \dots t_n$ for any n-ary predicate symbol P and terms t_i. We may insert parentheses to increase readability, *e.g.*, $Pffy = P(ffy) = P\big(f(f(y))\big)$ if f is a unary (1-ary) function symbol. ***Formulas*** are generated from atoms by binary connectives tensor $\otimes$ and par $\parr$ and unary quantifiers $\forall x$ and $\exists x$ for each variable x. Negation extends to formulas by $\overline{P\,t_1 \dots t_n} = \overline{P}\,t_1 \dots t_n$, $\overline{A \otimes B} = \overline{A} \parr \overline{B}$, $\overline{A \parr B} = \overline{A} \otimes \overline{B}$, $\overline{\exists x A} = \forall x \overline{A}$, $\overline{\forall x A} = \exists x \overline{A}$.

We identify a formula with its parse tree, a directed tree with leaves labelled by atoms and internal vertices by connec-

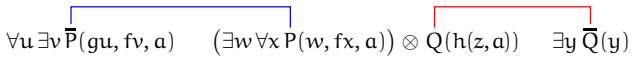


Figure 6. A linking with mgu (most general unifier) $[v\mapsto x, w\mapsto gu, y\mapsto h(z,a)]$, hence precedences $v \curvearrowright x$ and $w \curvearrowright u$.

tives and quantifiers, and edges directed towards the root. A ***sequent*** is a disjoint union of formulas.[2] We write comma for disjoint union. Sequents are proved using the following rules, where $A[x\mapsto t]$ denotes the result of simultaneously substituting the term t for all free occurrences of x in A.

$$\frac{}{P,\overline{P}}\,\text{ax} \qquad \frac{\Gamma, A, B}{\Gamma, A \parr B}\,\parr \qquad \frac{\Gamma, A[x\mapsto t]}{\Gamma, \exists x A}\,\exists$$

$$\frac{\Gamma, A \quad \overline{A}, \Delta}{\Gamma, \Delta}\,\text{cut} \qquad \frac{\Gamma, A \quad B, \Delta}{\Gamma, A\otimes B, \Delta}\,\otimes \qquad \frac{\Gamma, A}{\Gamma, \forall x A}\,\forall\ (x \text{ not free in } \Gamma)$$

These are the standard rules for first-order multiplicative linear logic [Gir87, Gir88, Gir91], omitting turnstile $\vdash$ (redundant in a right-sided calculus) and the exchange rule (redundant since we treat sequents as labelled forests). A sequent just above a rule is a ***hypothesis*** of a rule, and the sequent just below the rule is its ***conclusion***. The conclusion of a proof is its final sequent (the conclusion of its final rule).

A quantifier is ***vacuous*** if it binds no variable. For example, in $\forall x\exists y\forall z Pzc$ both $\forall x$ and $\exists y$ are vacuous, but $\forall z$ is not. An instance $\frac{\Gamma, A[x\mapsto t]}{\Gamma, \exists x A}$ of an $\exists$ rule in a proof Π is ***vacuous*** if x does not occur free in A; otherwise its ***witness*** is t (recoverable from $A[x\mapsto t]$ and A since A has at least one x). If every $\exists$ rule in Π introduces a distinct bound variable, we can unambiguously say that x is vacuous or has witness t (since x unambiguously determines the $\exists$ rule instance).

A sequent is ***clean*** if all quantifed variables are distinct from each other and from all free variables. *E.g.* $\exists x Px, \forall y Qzy$ is clean but $\exists x Px, \forall x Qzx$ and $\exists x Px, Qx$ are not. In a clean sequent, an ***existential*** (resp. ***universal***) variable is one bound by an existential (resp. universal) quantifier. For example, in $\forall x\overline{P}fx, \exists y Qyz$ the variables x, y, and z are universal, existential and free (respectively).

3 Cut-free unification nets

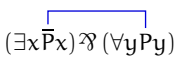
A ***link*** on a sequent Γ is a pair $\{l,l'\}$ of leaves in Γ whose predicate symbols are dual. A ***linking*** on Γ is a set of disjoint links on Γ whose union contains every leaf of Γ. We draw a link $\{l,l'\}$ as an undirected edge between the predicate symbols of l and l'. For example, a linking with two links is shown in Figure 6 (copied from the bottom of Fig. 2).

Let λ be a linking on Γ. Without loss of generality, assume Γ is clean (renaming bound variables if necessary, *e.g.* $\exists x Px, Qx$ becomes $\exists y Py, Qx$). A ***unifier*** for λ is an assignment of terms to existential variables which equalizes the term sequences at either end of every link, *e.g.*, $\sigma = [v\mapsto x, w\mapsto gu, y\mapsto h(z,a)]$ is a unifier for the Fig. 6 linking since upon substituting according to σ the first link has the three-term sequence (gu, fx, a) at either end, and the second has the one-term sequence $(h(z,a))$.

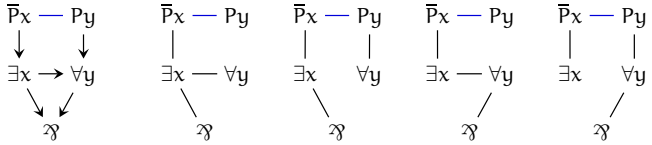
The formal unification problem is as follows. A link between $P(s_1,\ldots,s_n)$ and $\overline{P}(t_1,\ldots,t_n)$ determines n equations $s_i = t_i$. Taking the union across all links, we obtain a set of equations E. Solve E for the existential variables (treating free and universal variables as constants). For example, the left link $\overline{P}(gu, fv, a)\ \overline{P}(w, fx, a)$ in Fig. 6 determines three equations $gu = w$, $fv = fx$ and $a = a$, and the right link $\overline{Q(h(z,a))}\ \overline{Q}(y)$ yields $h(z,a) = y$, so E is $\{gu = w,\ fv = fx,\ a = a,\ h(z,a) = y\}$. Solve E for existential variables v, w, y (treating the universal u and x as constants): $[v\mapsto x, w\mapsto gu, y\mapsto h(z,a)]$.

A linking is ***unifiable*** if it has a unifier. Unifiability can be determined in linear time [MM76]. The ***most general unifier*** or ***mgu*** yields every other unifier by substitution. For example, the mgu of $\exists x\overline{Px}\ \exists y\overline{Py}$ is $\sigma = [x\mapsto\alpha, y\mapsto\alpha]$ for α a free variable: every unifier is $\sigma_t = [x\mapsto t, y\mapsto t]$ for some term t, and σ yields σ_t by substituting t for α, *i.e.*, $\sigma_t = \sigma[\alpha\mapsto t]$. The mgu is defined up to free variable renaming [LMM88]: $[x\mapsto\beta, y\mapsto\beta]$ also represents the mgu, for any other free variable β.

Let λ be a unifiable linking on a sequent Γ. Without loss of generality, assume Γ is clean. A ***precedence*** $x \curvearrowright y$ is an existential variable x and a universal variable y such that the mgu of λ assigns to x a term containing y. For example, the precedences of the linking in Fig. 6 are $v \curvearrowright x$ and $w \curvearrowright u$. The ***graph*** $\mathcal{G}(\lambda)$ of λ is the labelled directed forest Γ together with an undirected edge between leaves l and l' for every link $\{l,l'\}$ in λ, and a directed edge $\exists x \to \forall y$, called a ***leap***, for every precedence $x \curvearrowright y$. A ***switching*** of λ is any derivative of $\mathcal{G}(\lambda)$ obtained by deleting all but one edge into each $\parr$ and $\forall$ and undirecting remaining edges. For example, the graph of



is below-left, followed by its four switchings.



A linking is ***correct*** if it is unifiable and its switchings are trees (acyclic and connected). For example, the linking above is correct: all four switchings are trees. In §3.3 we prove that correctness can be verified in quadratic time, despite the fact that constructing an explicit mgu, used to extract leaps, may take exponential time and space.

A ***cut-free unification net*** on Γ is a correct linking on Γ.

The following two linkings show that the interaction of $\otimes$ and $\forall$ is a necessary part of correctness, via leaps and switchings. Although the linkings differ only by exchanging $\otimes$ for $\parr$, the left is correct, but the right is not.[3]

[2] We follow standard graph theory and work with graphs up to isomorphism, *i.e.*, modulo renaming of vertices. Thus disjoint union becomes associative.

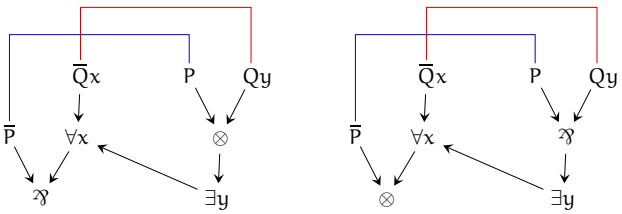
[3] The left is an instance of prenex extrusion $A\otimes\exists x B \vdash \exists x(A\otimes B)$, provable in MLL1 ($x$ not free in A), while the right is an instance of the unprovable $A\parr\exists x B \vdash \exists x(A\parr B)$. The right x was renamed to y to avoid ambiguity.

$$\frac{}{\{\{P,\overline{P}\}\} \rhd P,\overline{P}}\,\text{ax} \qquad \frac{\theta \rhd \Gamma, A, B}{\theta \rhd \Gamma, A \parr B}\,\parr \qquad \frac{\theta \rhd \Gamma, A[x\mapsto t]}{\theta \rhd \Gamma, \exists x A}\,\exists$$

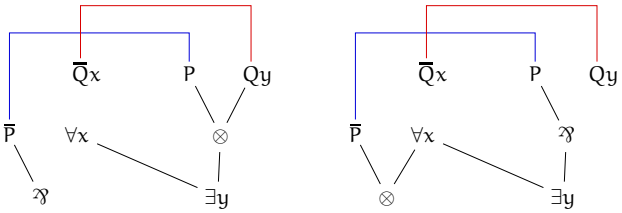
$$\frac{\theta \rhd \Gamma, A \qquad \phi \rhd B, \Delta}{\theta \cup \phi \rhd \Gamma, A \otimes B, \Delta}\,\otimes \qquad \frac{\theta \rhd \Gamma, A}{\theta \rhd \Gamma, \forall x A}\,\forall \text{ (x not free in } \Gamma)$$

Figure 7. Inductive translation of a cut-free MLL1 proof.

The respective graphs are



The respective switchings obtained by choosing the left $\parr$ edge and $\forall x$ leap are:



All four switchings of the left linking are trees (including the one shown above-left). However, the right linking has a non-tree switching, shown above-right. Thus one cannot hope for a factorized correctness criterion treating the propositional and first-order parts independently, for example, verifying separately that the underlying propositional MLL linking is correct (true for both linkings above), and that quantifier precedence $\curvearrowright$ (together with the subformula relation on quantifiers) is acyclic (also true for both linkings above).

3.1 Translation from cut-free proofs

Every cut-free proof Π of a sequent Γ translates to a cut-free unification net $\lfloor\Pi\rfloor$ on Γ in a manner similar to MLL proof nets: track dual pairs of predicate symbols from each axiom down the proof to form links on Γ.

A formal inductive definition of $\lfloor\Pi\rfloor$ is shown in Figure 7, where $\theta \rhd \Gamma$ asserts that θ is a cut-free unification net on Γ, and we make two simplifying assumptions (without loss of generality): in the $\otimes$ case θ and ϕ are disjoint, and in the $\forall$ and $\exists$ cases the leaf vertices of A, $\forall xA$, $\exists xA$ and $A[x\mapsto t]$ are the same (only their labels vary). Figure 2 shows the translation of a proof to the unification net of Figure 6, together with the corresponding Girard net for comparison. The proof of the following theorem is a routine induction:

Theorem 1. *The translation $\lfloor\Pi\rfloor$ of a cut-free MLL1 proof Π is a well-defined cut-free unification net.*

3.2 Cut-free surjectivity theorem

In standard proof net theory, a surjectivity theorem of the following form would typically be called a *sequentialization* theorem. However, as remarked in the Introduction and emphasized in Fig. 5, for unification nets the (non-deterministic) inverse of the surjection is both sequentialization (choice of rule orderings) and explicit witness assignment (choice of witnesses). Thus we simply label the theorem as *surjectivity*.

Theorem 2 (Cut-free Surjectivity)**.** *The translation from cut-free proofs to cut-free unification nets is surjective.*

We prove this theorem via an MLL encoding of a unification net, called the *frame*, via which we appeal to the standard MLL splitting tensor theorem.

Let θ be a unification net on Γ. Define the ***frame*** of θ by exhaustively applying the following subformula rewrites, in order, to obtain a linking θ_m on an MLL sequent Γ_m:

(1) *Encode every precedence $\curvearrowright$ as a new link.* Iterate through the precedences $x \curvearrowright y$ one by one. For each such precedence $x \curvearrowright y$, with corresponding subformulas $\exists xA$ and $\forall yB$, add a link as follows. Let Q be a fresh predicate symbol (distinct per precedence). Replace $\exists xA$ by $Q \otimes \exists xA$ and $\forall yB$ by $\overline{Q} \parr \forall yB$, and add a link between Q and $\overline{Q}$.

(2) *Delete quantifiers.* After (1) replace every subformula of the form $\forall yA$ or $\exists xA$ by A. (We no longer need their leaps, because we encoded leaps as links in step 1.)

(3) *Delete terms.* After (2) replace each predicate $Pt_1 \dots t_n$ by a nullary predicate symbol P.

For example, the frame of the unification net θ shown below-left is the MLL linking θ_m below-right:

Lemma 1. *Let θ be a unification net on Γ. The frame θ_m on Γ_m is an MLL proof net.*

Proof. Switchings correspond before and after the frame construction. □

Let θ be a unification net on Γ. A $\otimes$ root vertex v ***splits*** if deleting v (and its two edges) from the graph $\mathcal{G}(\theta)$ disconnects it into two connected components.

Lemma 2. *No tensor added during the frame construction splits: if θ is a unification net on Γ and θ_m is its frame on Γ_m, then no tensor of the graph $\mathcal{G}(\theta_m)$ added during step (1) of the frame construction splits.*

Proof. Lemma 1 and a routine analysis of switchings. □

Proof of Theorem 2 (Cut-free Surjectivity). Let θ be a cut-free unification net on Γ. W.l.o.g. Γ is clean. Proceed by induction on the number of connectives and quantifiers in Γ. In the base case Γ is $Pt_1 \dots t_n, \overline{P}t_1 \dots t_n$ for an n-ary predicate symbol P and terms t_i, so the corresponding axiom translates to θ, a single link. For the induction step, let $\mathcal{G}$ be the graph of θ.

($\parr$). Suppose Γ is $\Delta, A \parr B$. Let Γ' be Δ, A, B and define θ' on Γ' by the same links as θ (identifying the leaves of Γ' and Γ). The linking θ' is a unification net because (a) the mgu of θ is also that of θ' (all quantifiers and terms remain untouched, so

the unification problem is identical) and (b) every switching of θ' is a tree, since a non-tree would induce a non-tree switching of θ by adding an edge to the deleted $\parr$ down from the root of A (or B). Appeal to induction with θ' for a cut-free proof Π' whose translation is θ'. Appending the $\parr$ rule $\frac{\Delta, A, B}{\Delta, A \parr B}$ yields a cut-free proof Π, whose translation is θ because all links pass through the $\parr$ rule.

($\forall$). Suppose Γ is $\Delta, \forall x A$. Let Γ' be Δ, A and define θ' on Γ' by the same links as θ (identifying the leaves of Γ' and Γ). The mgu of θ is also that of θ' since x has only transitioned from universal to free (hence the unification problem is identical). Every switching of θ' is a tree, since a non-tree would induce a non-tree switching of θ by adding an edge down from the root of A to the deleted $\forall x$. Appeal to induction with θ' for a cut-free proof Π' whose translation is θ'. Appending the $\forall$ rule $\frac{\Delta, A}{\Delta, \forall x A}$ yields a cut-free proof Π, whose translation is θ because all links pass through the $\forall$ rule.

($\exists$). Suppose $\mathcal{G}$ has a root $\exists$ with no outward leap, say $\exists x$. Let σ be the mgu of θ, assigning t to x. Delete $\exists x$ by replacing the corresponding formula $\exists x A$ in Γ by $A[x \mapsto t]$ to form Γ', write down a final $\exists$ rule inferring Γ from Γ', and appeal to induction with θ' on Γ'. We obtain the mgu of θ' on Γ' by deleting the assignment $x \mapsto t$ from σ and replacing every other assignment $y \mapsto u$ with $y \mapsto u'$ for $u' = u[x \mapsto t]$. Every switching of the graph $\mathcal{G}'$ of θ' on Γ' is a tree because each induces a switching in $\mathcal{G}$ (since the deleted $\exists x$ was a root of Γ and every leap in $\mathcal{G}'$ is also a leap in $\mathcal{G}$).

($\exists\otimes$). Otherwise every root of $\mathcal{G}$ is either an $\exists$ with an outward leap or a $\otimes$. Let θ_m on Γ_m be the frame of θ on Γ. By the standard MLL splitting tensor theorem[4] [Gir87, Thm. 2.9.7], some $\otimes$ root ν of θ_m on Γ_m splits. By Lemma 2 ν is a $\otimes$ in Γ, and since every root $\exists$ has an outward leap, ν is a root (since no root $\otimes$ of Γ_m can result from step 2 in the frame construction deleting an $\exists$ below it). Thus ν splits in $\mathcal{G}$: deleting ν (and its two incoming edges) disconnects $\mathcal{G}$ into $\mathcal{G}_1$ and $\mathcal{G}_2$. Let Γ_i be the underlying sequent of $\mathcal{G}_i$ and θ_i the respective restriction of θ. Since ν splits, each θ_i is a unification net: its mgu is by restriction from θ, and any non-tree switching of θ_i would induce a non-tree switching of θ. Write down a $\otimes$ rule $\frac{\Gamma_1 \quad \Gamma_2}{\Gamma}$ and appeal to induction with θ_1 on Γ_1 and θ_2 on Γ_2. □

3.3 Correctness is at worst quadratic time

Define the ***size*** of a sequent Γ as the number of registers $|\Gamma|$ required to memorize Γ on a random access machine (*cf.* [Gue99]). In any non-redundant representation, $|\Gamma|$ is linear in the number of occurrences of symbols (predicate symbols, function symbols, variables, connectives and quantifiers) in Γ.[5] We analyze the worst-case asymptotic complexity of verifying the correctness of a cut-free unification net on Γ in terms of $|\Gamma|$.

Unifiability can be verified in linear time [MM76]. However, a standard mgu $[x_1 \mapsto t_1, \ldots, x_n \mapsto t_n]$ may take exponential time to construct, and be exponential in size. Since such an mgu generates the leaps in the correctness criterion, via precedences, correctness is naively exponential time and space.

[4] Every MLL net with a $\otimes$ and no root $\parr$ has a root $\otimes$ which splits.

[5] Although the number of distinct symbols in the logic is infinite, only a finite number k occur in any given sequent. We assume the symbols are enumerated $1, \ldots, k$. (This avoids an artificial inflation of $|\Gamma|$, which would make the complexity problem easier.)

Theorem 3. *The correctness of a cut-free unification net can be verfied in quadratic time.*

Proof. The primary unification algorithm of [MM76] provides in linear time an assignment $[x_1 \mapsto u_1, \ldots, x_n \mapsto u_n]$ with x_i not in u_j for $i \leqslant j$ such that the mgu σ is $[x_1 \mapsto t_1, \ldots, x_n \mapsto t_n]$ for $t_i = u_i[x_{i+1} \mapsto u_{i+1}] \ldots [x_n \mapsto u_n]$ (the sequential composition of $n-i$ single-variable substitutions applied to u_i). We shall extract all precedences of σ via transitive closure, without having to construct σ. Let $\{y_{i1}, \ldots, y_{im_i}\}$ be the set of variables occurring in u_i (existential, universal and free), and define u'_i as $f_i y_{i1} \ldots y_{im_i}$ for a fresh m_i-ary function symbol f_i. The assignment $\sigma' = [x_1 \mapsto t'_1, \ldots, x_n \mapsto t'_n]$ for $t'_i = u'_i[x_{i+1} \mapsto u'_{i+1}] \ldots [x_n \mapsto u'_n]$ has the same precedences as σ but can be constructed in quadratic time since each x_j appears at most once in each u'_i. Thus we can construct the linking graph in quadratic time. The linking graph determines a contractibility graph [Dan90] with $\parr$s and $\forall$s as switched nodes, and leaves, $\otimes$s and $\exists$s as unswitched nodes, checkable in linear time [Gue99]. Hence the overall complexity of correctness is at worst quadratic. □

4 Canonicity Theorem

For background on this section, see §1.7. Let Π be a proof of Γ. Without loss of generality, assume Γ is clean. Thus every $\exists$ rule introduces a distinct existential variable. Let x be an existential variable in Π and let ρ be the $\exists$ rule $\frac{\Gamma, A[x \mapsto t]}{\Gamma, \exists x A}$ introducing x. The ***scope*** of x in Π is every occurrence of t above ρ which descends to an occurrence of x in $\exists x A$ in the conclusion of ρ. Given a term u, define the ***witness replacement*** $\Pi[x \mapsto u]$ by replacing every occurrence of t in the scope of x by u. For example, if Π is below-left then $\Pi[x \mapsto hzb]$ is below-centre (not a well-formed proof) and $\Pi[x \mapsto hzb][y \mapsto hzb]$ is below-right (a well-formed proof):

$$\exists\dfrac{\exists\dfrac{\overline{\overline{P}fc\,,\,Pfc}}{\overline{P}fc\,,\,\exists y Py}\exists}{\exists x\overline{P}x\,,\,\exists y Py} \quad \xrightarrow{x \mapsto hzb} \quad \exists\dfrac{\exists\dfrac{\overline{\overline{P}hzb\,,\,Pfc}}{\overline{P}hzb\,,\,\exists y Py}\exists}{\exists x\overline{P}x\,,\,\exists y Py} \quad \xrightarrow{y \mapsto hzb} \quad \exists\dfrac{\exists\dfrac{\overline{\overline{P}hzb\,,\,Phzb}}{\overline{P}hzb\,,\,\exists y Py}\exists}{\exists x\overline{P}x\,,\,\exists y Py}$$

If $\sigma = [x_1 \mapsto t_1, \ldots, x_n \mapsto t_n]$ is an assigment of terms to existential variables in Π, define $\Pi\sigma = \Pi[x_1 \mapsto t_1] \ldots [x_n \mapsto t_n]$, the ***re-witnessing*** of Π along σ. This is well-defined modulo the choice of ordering of the x_i because scopes of distinct existential variables cannot overlap. For example, $\Pi[x \mapsto hzb, y \mapsto hzb]$ above-right is a re-witnessing of Π above-left.

Two cut-free MLL1 proofs are ***commutation-equivalent*** if one yields the other by a sequence of (zero or more) rule commutations. We use the standard notion of rule commutation in MLL1, for example,

$$\forall\dfrac{\dfrac{\Gamma, A, B}{\Gamma, \forall x A, B} \qquad C, \Delta}{\Gamma, \forall x A, B \otimes C, \Delta}\otimes \quad \leftrightarrow \quad \dfrac{\dfrac{\Gamma, A, B \qquad C, \Delta}{\Gamma, A, B \otimes C, \Delta}\otimes}{\forall\dfrac{\Gamma, A, B \otimes C, \Delta}{\Gamma, \forall x A, B \otimes C, \Delta}}$$

We only admit a commutation if it results in a well-formed proof. For technical convenience we assume every sequent in a cut-free proof is clean. Otherwise we may falsely reject a commutation due to variable clashes. For example, we would have to reject the left-to-right commutation above if x were

free in C (to avoid breaking the ∀-rule side condition on the right); by assuming all sequents are clean, we avoid such artefacts. There is nonetheless one commutation where we must be careful, even when all sequents are clean:

$$\exists\dfrac{\dfrac{\Gamma, A[x\mapsto t], B}{\Gamma, \exists x A, B}\forall}{\Gamma, \exists x A, \forall y B} \quad\leftrightarrow\quad \dfrac{\dfrac{\Gamma, A[x\mapsto t], B}{\Gamma, A[x\mapsto t], \forall y B}\forall}{\exists\dfrac{}{\Gamma, \exists x A, \forall y B}}$$

The left-to-right direction is unavailable if t contains y (breaking the side-condition of the ∀-rule on the right).

Two cut-free MLL1 proofs are ***equivalent*** if one yields the other by a sequence of rule commutations and/or re-witnessings. For example, the two proofs at the beginning of §1.7 are equivalent but not commutation-equivalent.

Theorem 4 (Canonicity)**.** *Two cut-free MLL1 proofs are equivalent (modulo rule commutations and re-witnessings) if and only if they have the same unification net.*

The proof follows from a number of auxiliary results below, whose individual proofs are routine and generally omitted.

Let θ be a cut-free unification net on Γ and $\mathcal{G}(\theta)$ its graph. A root v of Γ is ***ready*** if any of the following cases hold: v is a $\parr$ or ∀; v is an ∃ with no outgoing leap in $\mathcal{G}(\theta)$; v is a $\otimes$ which splits $\mathcal{G}(\theta)$. A rule ρ ***commutes downwards*** if a commutation rewrite applies with ρ as the upper rule.

Lemma 3. *Let ρ be a penultimate logical rule in a cut-free proof Π introducing a vertex v. If v is ready in the unification net of Π, then ρ commutes downwards.*

Let Π be a cut-free proof of Γ and v a vertex of Γ. Since MLL1 has no contraction or weakening, a unique rule in Π introduces v. By iterating Lemma 3 we obtain:

Lemma 4. *Let θ be the unification net of a cut-free proof Π. If v is a ready vertex in θ, then Π is commutation-equivalent to a cut-free proof Π' whose final rule introduces v.*

Lemma 5. *Let σ be the mgu of the unification net of a cut-free proof Π. The re-witnessing $\Pi\sigma$ is a well-defined cut-free proof.*

Let Π be cut-free proof. Without loss of generality, its conclusion Γ is clean, hence every quantifer rule introduces a distinct bound variable. Define the ***witness assignment*** σ_Π of Π by $\sigma_\Pi(x) = x$ if the existential rule introducing x is vacuous, otherwise set $\sigma_\Pi(x)$ to be the witness of x. Lemma 4 yields:

Lemma 6. *Suppose Π and Π' are cut-free proofs with the same witness assignment and the same unification net. Then Π and Π' are commutation-equivalent.*

Proof of Thoerem 4 (Canonicity). Let Π and Π' be cut-free proofs with the same unification net, whose mgu is σ. By Lem. 5 the re-witnessings $\Pi\sigma$ and $\Pi'\sigma$ are well-defined cut-free proofs, which are commutation-equivalent by Lem. 6 since they have the same witness assigment, σ. Thus Π and Π' are equivalent modulo rule commutations and re-witnessings. □

5 Cut

Extending unification nets with cuts comes essentially for free, as in the propositional case [Gir87] where one treats a cut as a tensor $A \underline{\ \overline{A}} \approx A\otimes\overline{A}$. For quantifiers one must generalize slightly, to an existentially closed tensor: $A \underline{\ \overline{A}} \approx \exists\underline{x}(A\otimes\overline{A})$ where $\exists\underline{x} = \exists x_1 \dots \exists x_n$ for $x_1 \dots x_n$ the free variables in A.

A ***cut*** $A \underline{\ \overline{A}}$ is a disjoint union of dual formulas A and $\overline{A}$, the ***cut formulas***, with an undirected edge between their roots, a ***cut edge***. A ***cut sequent*** is a disjoint union of a sequent and zero or more cuts. Let Δ be a cut sequent. A ***link*** on Δ is a pair $\{l, l'\}$ of leaves in Δ whose predicate symbols are dual. A ***linking*** on Δ is a set of disjoint links on Δ whose union contains every leaf of Δ. Figure 3 shows three examples.

We consider every free variable of A (hence also $\overline{A}$) to be bound in the cut $A \underline{\ \overline{A}}$. Such bound variables are the ***cut variables*** of $A \underline{\ \overline{A}}$. Their renaming is analogous to renaming of existential or universal variables. The (cut-free) ***encoding*** of a cut $A \underline{\ \overline{A}}$ is the existentially closed tensor $\exists\underline{x}(A\otimes\overline{A})$ where $\exists\underline{x}$ denotes $\exists x_1 \exists x_2 \dots \exists x_n$ for $x_1 \dots x_n$ the free variables in A. (For definiteness, we assume a fixed order of the x_i.) For technical convenience, and without loss of generality, we assume the leaves of the encoding are identical to the leaves of the cut. (For example, if $Px \underline{\ \overline{P}x}$ is the cut whose leaves are l and l', labelled Px and $\overline{P}x$, respectively, then the encoding is $\exists x(Px\otimes\overline{P}x)$ with the same leaves l and l', still labelled Px and $\overline{P}x$, respectively.) The ***encoding*** of a cut sequent Δ is the sequent $\Delta^\otimes$ obtained by replacing each cut by its encoding.

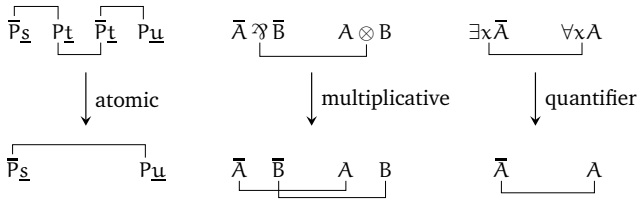
Let θ be a linking on a cut sequent Δ. By our assumption that the leaves remain unchanged by encoding, θ also constitutes a (cut-free) linking on $\Delta^\otimes$. The linking θ on Δ is ***correct*** if θ is correct (in the cut-free sense of §3) on $\Delta^\otimes$. A ***unification net*** (or ***unet*** for short) on a cut sequent Δ is a correct linking on Δ.

Since cut-free encoding is linear time, the following is a corollary of cut-free quadratic-time correctness:

Theorem 5. *Unification net correctness can be verified in quadratic time.*

5.1 Cut elimination

A ***cut reduction*** on a unification net is a subgraph rewrite of any of the following forms:



Here $\underline{t}$ denotes any sequence of terms. We refer to the upper subgraphs as ***redexes***.

Theorem 6. *Reducing a cut from a unification net yields a unification net.*

To prove this theorem we shall require auxiliary definitions and a key lemma concerning the reduction of a quantifier cut.

A ***cycle*** in the graph of a linking is a subgraph C with vertex set $\{v_1, \dots, v_n\}$ for $n \geqslant 2$, all v_i distinct, and an edge (directed

or undirected) between v_i and v_{i+1} for all $i \pmod n$, such that if $n = 2$ then $\mathcal{C}$ contains two distinct edges between v_1 and v_2; $\mathcal{C}$ is a ***switching cycle*** [HG03] if it contains at most one directed edge into any given $\parr$ or $\forall$ vertex.

Lemma 7. *Let θ be a unification net on $\Gamma, \exists x \overline{A}\ \forall x A$ and let the linking θ' on $\Gamma, \overline{A}\ A$ result from reducing the distinguished quantifier cut. Then the graph of θ' has no switching cycle.*

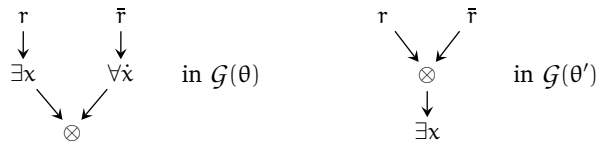
Proof. The respective cut-free encodings are

$$\Gamma^{\otimes}, \exists x_1 \dots \exists x_n (\exists x \overline{A} \otimes \forall x A) \qquad \Gamma^{\otimes}, \exists x_1 \dots \exists x_n \exists x (\overline{A} \otimes A)$$

where the additional $\exists x$ in the latter is because, without loss of generality, x is free in A: when x is not free in A, the result is trivial since $\exists x$ and $\forall x$ in the redex cut $\exists x \overline{A}\ \forall x A$ are vacuous, so topologically inert. To avoid bound variable conflict, rename $\forall x$ to $\forall \dot{x}$, as in $\Gamma^{\otimes}, \exists x_1 \dots \exists x_n (\exists x \overline{A} \otimes \forall \dot{x} \dot{A})$, where $\dot{A}$ is the result of substituting $\dot{x}$ for x in A.

Let σ be a unifier for θ. Thus $\sigma = [z_1 \mapsto t_1, \dots, z_k \mapsto t_k, x \mapsto t]$, where the z_i include the x_j. The term t assigned to x cannot contain $\dot{x}$, or there would be a switching cycle due to the resulting precedence $x \curvearrowright \dot{x}$, via the $\otimes$ of the encoding of $\exists x \overline{A}\ \forall x A$. Let t'_i result from substituting t for $\dot{x}$ in t_i. Define $\sigma' = [z_1 \mapsto t'_1, \dots, z_k \mapsto t'_k, x \mapsto t]$. This is a well-defined unifier for θ' since none of the t'_i contains $\dot{x}$ (because t did not contain $\dot{x}$). Without loss of generality, σ' is an mgu.

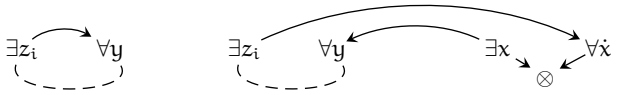
We must prove that $\mathcal{G}(\theta')$ has no switching cycle. Suppose $\mathcal{C}'$ were such. We consider three cases according to whether there are zero, one or two (or more) leaps in $\mathcal{C}'$ which are not in $\mathcal{G}(\theta)$. Let r and $\bar{r}$ be the root vertices of A and $\overline{A}$, respectively, and assume that $\dot{A}$ has the same vertices as A. We assume $\mathcal{G}(\theta)$ and $\mathcal{G}(\theta')$ have the same vertices, except for the necessary difference around the tensors of the encodings of the two cuts:



For technical convenience assume the vertex of $\exists x$ is the same in each case.

Case (0): every leap of $\mathcal{C}'$ is in $\mathcal{G}(\theta)$. Define a switching cycle in $\mathcal{G}(\theta)$ from $\mathcal{C}'$ by, if necessary, re-routing a traversal of the tensor of the encoding of $A\ \overline{A}$ to the tensor of the encoding of $\forall \dot{x} \dot{A}\ \exists x A$.

Case (1): $\mathcal{C}'$ contains a single leap $\exists z_i \to \forall y$ which does not occur in $\mathcal{G}(\theta)$. (The leap must be from an $\exists z_i$ since both σ and σ' assign $x \mapsto t$.) This is depicted below-left, where the dashed line represents one or more edges in $\mathcal{C}'$.



The leap $\exists z_i \to \forall y$ came from a precedence $z \curvearrowright y$ present in θ' but not in θ. Such an additional precedence can arise only from the construction of t'_i by substituting t for $\dot{x}$ in t_i, hence y must be in t, so $x \curvearrowright y$ is a precedence of θ (since $x \mapsto t$ is in σ), with a corresponding leap $\exists x \to \forall y$ in $\mathcal{G}(\theta)$. Since t_i contains $\dot{x}$, there is a precedence $x_i \curvearrowright \dot{x}$, hence a leap $\exists z_i \to \forall \dot{x}$ in $\mathcal{G}(\theta)$. Thus we can construct a switching cycle in $\mathcal{G}(\theta)$ as above-right.

Case (2+): there are two or more leaps in $\mathcal{C}' \subseteq \mathcal{G}(\theta')$ not present in $\mathcal{G}(\theta)$, say (without loss of generality) $\exists z_1 \to \forall y_1$ and $\exists z_2 \to \forall y_2$. Either (a) the leaps are in the same direction around $\mathcal{C}'$, as shown below-left, or (b) they are in opposite directions, as shown below-right.

Reasoning for each $\exists z_i$ and $\forall y_i$ as in case (1) for $\exists z_i$ and $\forall y$, we have leaps $\exists x \to \forall y_i$ and $\exists z_i \to \forall x$. Thus, in $\mathcal{G}(\theta)$, if (a), we can construct the switching cycle below-left, and if (b), the switching cycle below-right.

□

Proof of Theorem 6. Each of the three reductions preserves the difference between the number of links and the number of $\otimes$s and cuts, thus (see *e.g.* [HG05, §4.7.1]) to confirm a switching is a tree we need only check that it is acyclic. Acyclicity of all switchings is equivalent [HG05, §4.7.2] to there being no switching cycle in the graph of the linking.

Atomic case: an atomic cut reduction takes θ on $\Gamma, P\underline{t}\ \overline{P}\underline{t}$ to θ' on Γ. Let σ be mgu for θ, which by definition equalizes the term sequences $\underline{s}$ and $\underline{t}$ (due to the left link in the redex) and $\underline{t}$ and $\underline{u}$ (due to the right link). By transitivity σ equalizes $\underline{s}$ and $\underline{u}$, thus the restriction σ' of σ to existential variables in θ' is an mgu for θ'. A switching cycle $\mathcal{C}'$ of $\mathcal{G}(\theta')$ induces a corresponding switching cycle $\mathcal{C}$ of $\mathcal{G}(\theta)$: since σ' is a restriction of σ, every leap in $\mathcal{C}'$ determines a corresponding leap in $\mathcal{C}$; if $\mathcal{C}'$ passes through the new link $\overline{P\underline{s}}\ P\underline{u}$, in $\mathcal{C}$ go instead between $\overline{P}\underline{s}$ and $P\underline{u}$ via the cut $P\underline{t}\ \overline{P}\underline{t}$ (*i.e.*, via the $\otimes$ of its encoding $\exists \underline{x}(P\underline{t} \otimes \overline{P}\underline{t})$).

Multiplicative case: a multiplicative cut reduction takes θ on $\Gamma, \overline{A} \parr \overline{B}\ A \otimes B$ to θ' on $\Gamma, \overline{A}\ A, \overline{B}\ B$. There is no change in mgu, precedences or leaps, so the reasoning of the usual multiplicative case [Gir96] goes through directly.

Quantifier case: Lemma 7. □

Theorem 7 (Strong normalization)**.** *Every sequence of cut reductions terminates.*

Proof. Each reduction decreases the size of the cut sequent. □

Write $(\theta, \Delta) \xrightarrow{c} (\theta', \Delta')$ if the unification net θ on Δ yields θ' on Δ' by reducing the cut whose cut edge is c.

Lemma 8 (Diamond)**.** *If $(\theta_1, \Delta_1) \xleftarrow{c_1} (\theta, \Delta) \xrightarrow{c_2} (\theta_2, \Delta_2)$ for distinct cut edges c_1 and c_2, then there exists a unique (θ', Δ') such that $(\theta_1, \Delta_1) \xrightarrow{c_2} (\theta', \Delta') \xleftarrow{c_1} (\theta_2, \Delta_2)$.*

Proof. A routine case analysis, relying on the fact that cut reduction is local and cuts are pairwise disjoint. □

Theorem 8 (Confluence)**.** *Cut reduction is confluent.*

Proof. Theorem 7 (Strong normalization) and Lemma 8. □

Theorem 9 (Linear time cut elimination)**.** *Eliminating all cuts from a unification net on Δ is linear time in the size of Δ.*

Proof. Every cut reduction is local, decreasing the size of the cut sequent, and cuts are pairwise disjoint. □

5.2 Surjectivity Theorem with cut

Corresponding to the existentially closed tensor encoding, we extend the cut rule by keeping the cut formulas in the conclusion and allowing a substitution σ of the cut formulas A and $\overline{A}$ in the hypotheses:

$$\frac{\Gamma, A\sigma \qquad \overline{A}\sigma, \Delta}{\Gamma, \underbrace{A \; \overline{A}}, \Delta}\text{cut}$$

Here σ is any substitution of terms for free variables in A and $\overline{A}$. Define MLL1$^{\sqcup}$ as the extension of MLL1 with this rule. Two examples are below:

$$\dfrac{\dfrac{}{Pfx, \overline{P}fx} \quad \dfrac{\overline{Pfx, \overline{P}fx}}{Pfx, \exists z \overline{P}z}\exists}{Pfx, \underbrace{\overline{P}fx \;\; Pfx}, \exists z \overline{P}z}\text{cut} \qquad\qquad \dfrac{\dfrac{}{Pfx, \overline{P}fx} \quad \dfrac{\overline{Pfx, \overline{P}fx}}{Pfx, \exists z \overline{P}z}\exists}{Pfx, \underbrace{\overline{P}y \;\; Py}, \exists z \overline{P}z}\text{cut}$$

In the left example σ is trivial, so that $A\sigma = A$, and in the right example $\sigma = [y \mapsto fx]$.

Theorem 10. *The translation from* MLL1$^{\sqcup}$ *proofs to linkings is a surjection onto unification nets.*

Proof. Follows from Theorem 2 via cut-free encoding. □

6 Factorization through Girard nets and unification calculus

We factorize the surjection $\lfloor - \rfloor$ from cut-free MLL1 proofs onto cut-free unification nets defined in §3.1 in two different ways: the two outer paths of the commuting square in Fig. 5. For technical convenience, throughout this section we assume every sequent is clean.

6.1 Factorization through cut-free Girard nets

Define the translation of a cut-free Girard net on Γ to a linking on Γ in the same manner as the translation of a cut-free proof: track the axiom links down to links on the concluding sequent.

Lemma 9. *The translation of a cut-free Girard net is a cut-free unification net.*

The proof is a routine induction, by using the explicit witnesses in the Girard net to ensure unifiability.

Theorem 11. *The translation from cut-free Girard nets to cut-free unification nets is surjective.*

Proof. Let θ be a cut-free unification net on Γ with mgu σ. We unfold θ into a cut-free Girard net G by working upwards from each root of Γ.

We first unfold each formula A in Γ to a fragment of G with concluding formula A. Define the *unfolding* $\widehat{A}$ of a formula A as the following tree, alternating between Girard-links and formulas, whose root, called the *conclusion* of $\widehat{A}$, is the formula A. If A is an atom, then $\widehat{A} = A$. If $A = B \otimes C$, define $\widehat{A}$ as $\frac{\widehat{B} \quad \widehat{C}}{B \otimes C}$, the disjoint union of $\widehat{B}$ and $\widehat{C}$ and a $\otimes$-link taking the conclusions B and C of $\widehat{B}$ and $\widehat{C}$ as hypotheses and $A = B \otimes C$ as its conclusion. If $A = B \parr C$ define $\widehat{A}$ analogously, with $\parr$ in place of $\otimes$. If $A = \forall x B$, define $\widehat{A}$ as $\frac{\widehat{B}}{\forall x B}$, the tree $\widehat{B}$ with a $\forall$-link taking the conclusion B of $\widehat{B}$ as its hypothesis and $A = \forall x B$ as its conclusion. If $A = \exists x B$, define $\widehat{A}$ as $\frac{\widehat{B}[x \mapsto t]}{\exists x B}$, where t is the the term assigned to x by the mgu σ and $\widehat{B}[x \mapsto t]$ is the result of substituting t for x in every formula in $\widehat{B}$. Thus $\widehat{A}$ is the tree $\widehat{B}[x \mapsto t]$ and an $\exists$-link whose hypothesis is the conclusion $B[x \mapsto t]$ of $\widehat{B}[x \mapsto t]$ and whose conclusion is $A = \exists x B$. Define the unfolding $\widehat{\Gamma}$ of Γ as the disjoint union of the unfoldings of its formulas. By induction, the atoms in $\widehat{\Gamma}$ are in bijection with the leaves of Γ. Define G from $\widehat{\Gamma}$ as follows: for each link $\{l, l'\}$ in θ between a pair of leaves in Γ, add a Girard axiom-link $\widehat{l} \;\; \widehat{l'}$ between the corresponding pair of atoms in $\widehat{\Gamma}$.

We must show that G is a cut-free Girard net. First we prove that the atoms of each axiom-link in G are strictly dual. Each axiom-link $\widehat{L}$ in G is derived from a link L in θ between leaves $P(t_1, \ldots, t_n)$ and $\overline{P}(t'_1, \ldots, t'_n)$. Since the mgu σ equalizes corresponding term sequences, we have $t_i\sigma = t'_i\sigma$. The same substitutions of existential variables applied in the unfolding of formulas in $\widehat{\Gamma}$, so the axiom-link $\widehat{L}$ in G is between $P(t_1\sigma, \ldots, t_n\sigma)$ and $\overline{P}(t'_1\sigma, \ldots, t'_n\sigma)$, strictly dual.

We must show that G has no switching cycle. W.l.o.g. every jump from a formula A with an eigenvariable x to the conclusion $\forall x B$ of the corresponding $\forall x$-link can move to an edge from either (a) the hypothesis B of the $\forall x$-link, or (b) the hypothesis C of a $\exists$-link: if A is above $\forall x B$, choose (a), following the path between A and B; otherwise A must have a $\exists$-link below it which prevents the eigenvariable x from being free in the conclusion, and we choose (b), following the path between A and the hypothesis C of the $\exists$-link. Thus G-switchings and θ-switchings correspond.

Since Γ is assumed clean, G satisfies the requisite closure condition (by suitably renaming free variables to constants).

By induction, since the translation from cut-free Girard nets to cut-free unification nets defined in Lemma 9 uses the converse steps to those above (removing rather than adding witnesses), G translates to θ. □

Theorem 12. *The surjection from cut-free MLL1 proofs to cut-free unification nets factorizes through cut-free Girard nets.*

Proof. The surjection from cut-free MLL1 proofs to cut-free Girard nets, followed by that from cut-free Girard nets to cut-free unification nets, is the translation $\lfloor - \rfloor$ from cut-free MLL1 proofs to unification nets defined in §3.1 (the diagonal of Fig. 5) because each is defined by the same tracking of dual predicate symbols down from axioms. □

6.2 Factorization through cut-free unification calculus

Let Π be a proof of Γ. Define its ***skeleton*** as $\Pi\iota$ for ι the identity on the non-vacuous existential variables of Π. (Re-witnessing $\Pi\sigma$ was defined in §4.) Define a ***unification proof*** of Γ as a skeleton of a proof of Γ, and define ***unification calculus*** as the

MLL1 proof system comprising unification proofs.[6] In general, a skeleton will not be a well-formed sequent calculus proof since its axioms can be ill-formed, with non-dual predicates $Pt_1 \dots t_n$ and $\overline{P}u_1 \dots u_1$.

Theorem 13. *The correctness of a unification proof can be verified in polynomial time.*

Proof. Check the unifiability of the (ill-formed) axioms of the unification proof U. If they are not unifiable, U is invalid. Otherwise, let σ be an mgu, and verify that $U\sigma$ is a well-defined MLL1 proof. Naively this is exponential time (since constructing the mgu is exponential time and space, in general). However, we can use the same technique as in the quadratic-time complexity proof (Theorem 3) to build a sequential mgu, then lazily confirm that every rule of $U\sigma$ would be a well-formed rule were we to actually carry out the substitution σ at each rule (verifying that the predicates in every link become dual, and the $\forall$ rule side condition on free variables holds). □

Theorem 14. *The surjection from cut-free MLL1 to cut-free unification nets factorizes through cut-free unification calculus.*

Proof. Instead of extracting links directly from a proof Π, first take the skeleton $\Pi\iota$ (dropping explicit witnesses) then extract the links from $\Pi\iota$. Since the (ill-formed) axiom rules of $\Pi\iota$ are those of Π, only with some terms substituted, we extract the same links as going directly from Π. □

A Girard's cut elimination is exp-time/space

Let G be the Girard net

$$\dfrac{\dfrac{\overline{\overline{P}x} \quad Px}{\forall x \overline{P}x} \quad \dfrac{\overline{\overline{P}fxx} \quad Pfxx}{Px \otimes \overline{P}fxx}}{\exists x (Px \otimes \overline{P}fxx)} \quad \dfrac{\overline{Pfxx}}{\exists x\, Px}$$

for f a binary function symbol and G^n the result of cutting n copies of G against one another using $n-1$ cuts, each between copies of $\forall x\, \overline{P}x$ and $\exists x\, Px$, renaming to ensure unique eigenvariables. The cut-free normal form of G^n has a term with 2^n occurrences of x, so is exponentially larger than G^n.

B Exponentially large cut-free Girard nets

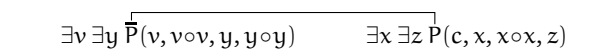
Here is a minimal cut-free Girard net on

$$\exists v\, \exists y\, \overline{P}(v, v \circ v, y, y \circ y),\ \exists x\, \exists z\, P(c, x, x \circ x, z)$$

for P a 4-ary predicate, $\circ$ an infix binary function symbol, and abbreviations $c^2 = c \circ c$, $c^4 = c^2 \circ c^2$, $c^8 = c^4 \circ c^4$:

$$\exists \dfrac{\exists \dfrac{\overline{\overline{P}\left(c, c^2, c^4, c^8\right)}}{\exists y\, \overline{P}(c, c^2, y, y \circ y)}}{\exists v\, \exists y\, \overline{P}(v, v \circ v, y, y \circ y)} \quad \dfrac{\dfrac{P\left(c, c^2, c^4, c^8\right)}{\exists z\, P\left(c, c^2, c^4, z\right)} \exists}{\exists x\, \exists z\, P(c, x, x \circ x, z)} \exists$$

Its axiom link is exponentially larger than the sequent: in the general case with P n-ary, the axiom link has $2(2^n - 1)$ copies of c. This example has no connective, so also shows that Girard's first-order additive nets [Gir96] with explicit existential witnesses grow exponentially; indeed, it shows that quantifier-only sequent proofs and Girard nets do. The corresponding cut-free unification net grows only linearly with n, here $n = 4$:



References

[And81] P. B. Andrews. Theorem proving via general matings. *J. ACM*, 28, 1981.

[Bib81] W. Bibel. On matrices with connections. *J. ACM*, 28(4), 1981.

[BW95] G. Bellin and J. van de Wiele. Subnets of proof-nets in MLL$^-$. In *Advances in Linear Logic*. CUP, 1995.

[Car10] A. Carbone. A new mapping between combinatorial proofs and sequent calculus proofs read out from logical flow graphs. *Inf. Comput.*, 208(5), 2010.

[CR79] S. A. Cook and R. A. Reckhow. The relative efficiency of propositional proof systems. *J. Symb. Logic*, 44:36–50, 1979.

[Dan90] V. Danos. *La logique linéaire appliquée à l'étude de divers processus de normal. et princip. du lambda calcul*. PhD thesis, U. Paris, 1990.

[DR89] V. Danos and L. Regnier. The structure of multiplicatives. *Archive for Mathematical Logic*, 28:181–203, 1989.

[Gir87] J.-Y. Girard. Linear logic. *Theor. Comp. Sci.*, 50:1–102, 1987.

[Gir88] J.-Y. Girard. Quantifiers in linear logic. In *Temi e prospettive della logica e della filosofia della scienza contemporanee*, volume 1. CLUEB, Bologna, 1988.

[Gir91] J.-Y. Girard. Quantifiers in linear logic II. *Nuovi problemi della logica e della filosofia della scienza*, 2, 1991.

[Gir96] J.-Y. Girard. Proof-nets: the parallel syntax for proof theory. In *Logic and Algebra*, volume 180 of *Lec. Notes In Pure and Applied Math*. 1996.

[Gir11] J.-Y. Girard. *The Blind Spot: Lectures on Logic*. Eur. Math. Soc., 2011.

[Gue99] S. Guerrini. Correctness of multiplicative proof nets is linear. In *Proc. Logic in Computer Science '99*, 1999.

[Hei10] W. Heijltjes. Classical proof forestry. *Annals of Pure and Applied Logic*, 161(11), 2010.

[Her30] J. Herbrand. *Recherches sur la théorie de la démonstration*. PhD thesis, Sorbonne, Paris, 1930.

[HG03] D. J. D. Hughes and R. J. van Glabbeek. Proof nets for unit-free multiplicative additive linear logic (Extended abstract). In *Proc. Logic in Comp. Sci. '03*, 2003.

[HG05] D. J. D. Hughes and R. J. van Glabbeek. Proof nets for unit-free multiplicative-additive linear logic. *ACM Trans. Comput. Log.*, 6, 2005.

[Hug06a] D. J. D. Hughes. Proofs without syntax. *Annals of Mathematics*, 143:1065–1076, 2006.

[Hug06b] D. J. D. Hughes. Towards Hilbert's 24$^{\text{th}}$ Problem: Combinatorial Proof Invariants. In *Proc. WOLLiC'06*, volume 165 of *Lec. Notes in Comp. Sci.*, 2006.

[Hug14] D. J. D. Hughes. First-order proofs without syntax. Slides for the Berkeley Logic Colloquium, October 17, 2014. `http://boole.stanford.edu/~dominic/fopws-blc-2014`, 2014.

[LMM88] J.-L. Lassez, M. J. Maher, and K. Marriott. Unification revisited. In *Foundations of deductive databases and logic programming*. 1988.

[McK10] R. McKinley. Expansion nets: Proof nets for for propositional classical logic. In *LPAR 17*, volume 6397 of *LNCS*, 2010.

[Mil84] D. A. Miller. Expansion tree proofs and their conversion to natural deduction proofs. *Lec. Notes in Comp. Sci.*, 170, 1984.

[MM76] A. Martelli and U. Montanari. Unification in linear time and space: a structured presentation. Technical report, U. Pisa, 1976.

[Moo02] R. Moot. *Proof nets for lingusitic analysis*. PhD thesis, Utrecht, 2002.

[Pra70] D. Prawitz. A proof procedure with matrix reduction. *Lecture Notes in Mathematics*, 125, 1970.

[Qui55] W. V. Quine. A proof procedure for quantification theory. *J. Symbolic Logic*, 20(2), 1955.

[Rob65] J. A. Robinson. A machine-oriented logic based on the resolution principle. *J. ACM*, 12(1), 1965.

[Str09] L. Straßburger. Some observations on the proof theory of second order propositional multiplicative linear logic. In *TLCA '09*, 2009.

[Str17] L. Straßburger. Combinatorial Flows and Their Normalisation. In *Proc. FSCD 2017*, volume 84 of *Leibniz Int. Proc. Informat.*, 2017.

[6] One could investigate cut elimination on unification calculus by mimicking sequent calculus cut elimination without the explicit witnesses. Since witnesses are absent, such a cut elimination is likely to be polynomial-time.