Theory and Applications of Categories, Vol. 6, No. 9, 1999, pp. 105–146.

NATURAL DEDUCTION AND COHERENCE FOR NON-SYMMETRIC LINEARLY DISTRIBUTIVE CATEGORIES

Dedicated to Joachim Lambek

ROBERT R. SCHNECK

ABSTRACT. In this paper certain proof-theoretic techniques of [BCST] are applied to *non-symmetric* linearly distributive categories, corresponding to *non-commutative* negation-free multiplicative linear logic (mLL). First, the correctness criterion for the two-sided proof nets developed in [BCST] is adjusted to work in the non-commutative setting. Second, these proof nets are used to represent morphisms in a (non-symmetric) linearly distributive category; a notion of proof-net equivalence is developed which permits a considerable sharpening of the previous coherence results concerning these categories, including a decision procedure for the equality of maps when there is a certain restriction on the units. In particular a decision procedure is obtained for the equivalence of proofs in non-commutative negation-free mLL without non-logical axioms.

1. Introduction

Linearly distributive categories were introduced (as "weakly distributive categories") in [CS91], where these categories were shown to be a natural categorical setting for multiplicative linear logic (mLL) without negation. The sequel [BCST] further introduced a natural deduction system for the proof theory of this logic—that is, a version of proof nets suitable for studying the negation-free logic. Without negation, it was easier to isolate the problem of the units in coherence results: using these proof nets, the authors settled the coherence problem for symmetric linearly distributive categories, by providing a decision procedure for equivalence of morphisms.

However, this result was obtained only for *symmetric* linearly distributive categories (corresponding to *commutative* mLL without negation, and *non-planar* proof nets). Certain additional difficulties arise with *planar* proof nets (*non-symmetric* linearly distributive categories, *non-commutative* mLL without negation), which prevent an immediate generalization from the commutative case. Furthermore, there is also a problem with the generalization, to the non-commutative case, of the correctness criterion for proof nets.

This paper addresses these two problems. I describe a working correctness criterion for planar proof nets. (A criterion for one-sided planar proof nets, without units or non-logical axioms, was first given in [A95], using the notion of trips.) However, a complete

This material is based upon work supported in part under a Winston Churchill Scholarship, and in part under a National Science Foundation Graduate Research Fellowship. The author wishes to thank J.R.B. Cockett and R.A.G. Seely for their encouragement of this work.

Received by the editors 1999 January 30 and, in revised form, 1999 September 8.

Published on 1999 November 30.

1991 Mathematics Subject Classification: 03B60, 03F05, 03F07, 03G30, 18D10, 18D15.

Key words and phrases: categorical proof theory, linear logic, monoidal categories.

decision procedure for coherence in non-symmetric linearly distributive categories has still not been achieved. I present a modification of the notion of net-equivalence which considerably narrows the problem; in particular, a decision procedure is achieved given a certain restriction on morphisms involving the units.

The main content of the paper is a study of two rewrite systems: the sequentialization procedure which provides the correctness criterion for planar proof nets, and the net-equivalence rewrite system which forms the backbone of the coherence results. Appendix A describes the small amount of necessary information from rewriting theory.

This paper follows the pattern of [BCST]. In the interest of being somewhat self-contained I begin with a definition of linearly distributive categories, and a description of the fragment of linear logic to which they correspond. Following a definition of circuits, which are the underlying structures of these proof nets, I develop the new sequentialization procedure, and show that it is a normalizing rewrite system for proof nets. Next I develop a rewrite system which defines circuit equivalence, modified for the non-commutative case. I note the fact that the proof from [BCST], that planar proof nets form a linearly distributive category, carries over correctly given the modifications. Finally I show that the circuit equivalence rewrite system is normalizing, and discuss how this relates to a decision procedure for the coherence of non-symmetric linearly distributive categories.

A reader already familiar with the ideas of [CS91] and [BCST] should concentrate on sections 4–6 for the new material.

2. Preliminaries

2.1. Notation. *We shall use symbols* $\otimes$ *(*tensor *or* times*) and* $\oplus$ *(*cotensor, plus, *or* par*) to refer to the two tensors of linearly distributive categories, and of multiplicative linear logic; the units of the tensors are respectively* $\top$ *(*top, true, unit*) and* $\bot$ *(*bottom, false, counit*). Note that this conflicts with Girard's notation in [G87] (where par would be reverse-ampersand and the tensor unit* $\mathbf{1}$*), but is consistent with the notation of [BCST].*

2.2. Definition. *A* ***linearly distributive category*** C *is a category with two tensors* $\otimes$ *and* $\oplus$*; each tensor is equipped with a unit object, an associativity natural isomorphism, and left and right unit natural isomorphisms:*

$$(\otimes, \top, a_\otimes, u^L_\otimes, u^R_\otimes)$$

$$a_\otimes \colon (A \otimes B) \otimes C \to A \otimes (B \otimes C)$$

$$u^L_\otimes \colon \top \otimes A \to A$$

$$u^R_\otimes \colon A \otimes \top \to A$$

$$(\oplus, \bot, a_\oplus, u^L_\oplus, u^R_\oplus)$$

$$a_\oplus \colon (A \oplus B) \oplus C \to A \oplus (B \oplus C)$$

$$u^L_\oplus \colon \bot \oplus A \to A$$

$$u^R_\oplus \colon A \oplus \bot \to A.$$

The two tensors are linked by two linear distribution natural transformations:

$$\delta_L\colon A \otimes (B \oplus C) \to (A \otimes B) \oplus C$$
$$\delta_R\colon (B \oplus C) \otimes A \to B \oplus (C \otimes A).$$

Furthermore the following coherence conditions must be satisfied:

$$a_\otimes; 1 \otimes u^L_\otimes = u^R_\otimes \otimes 1 \tag{1}$$
$$a_\otimes; a_\otimes = a_\otimes \otimes 1; a_\otimes; 1 \otimes a_\otimes \tag{2}$$
$$a_\oplus; 1 \oplus u^L_\oplus = u^R_\oplus \oplus 1 \tag{3}$$
$$a_\oplus; a_\oplus = a_\oplus \oplus 1; a_\oplus; 1 \oplus a_\oplus \tag{4}$$
$$u^L_\otimes = \delta_L; u^L_\otimes \oplus 1 \tag{5}$$
$$u^R_\otimes = \delta_R; 1 \oplus u^R_\otimes \tag{6}$$
$$u^L_\oplus \oplus 1 = \delta_R; u^L_\oplus \tag{7}$$
$$1 \otimes u^R_\oplus = \delta_L; u^R_\oplus \tag{8}$$
$$\delta_L; a_\otimes \oplus 1 = a_\otimes; 1 \otimes \delta_L; \delta_L \tag{9}$$
$$a_\otimes; \delta_R = \delta_R \otimes 1; \delta_R; 1 \oplus a_\otimes \tag{10}$$
$$\delta_R; a_\oplus = a_\oplus \otimes 1; \delta_R; 1 \oplus \delta_R \tag{11}$$
$$1 \otimes a_\oplus; \delta_L = \delta_L; \delta_L \oplus 1; a_\oplus \tag{12}$$
$$\delta_L; \delta_R \oplus 1; a_\oplus = \delta_R; 1 \oplus \delta_L \tag{13}$$
$$a_\otimes; 1 \otimes \delta_R; \delta_L = \delta_L \otimes 1; \delta_R. \tag{14}$$

Also note that a *symmetric* linearly distributive category is one in which the tensors are symmetric (in which case further coherence conditions need to be introduced). A more detailed exposition can be found in [CS91].

2.3. mLL without negation. Figure 1 lists the rules for a sequent calculus presentation of non-commutative mLL without negation. Notice, in particular, the non-commutative restriction on cut: the usual cut rule separates into four rules. Since there is no negation we will be working with two-sided sequents. This sequent calculus was first given by Abrusci in [A91].

A link between linearly distributive categories and linear logic is indicated by the "resource-sensitive" character of the linear distributivities, as compared to an ordinary distribution. In [CS91] and [BCST] the connection is fully developed. To be more precise: there is an equivalence between (the 2-categories of) linearly distributive categories and two-tensor polycategories, which are a conservative extension of polycategories, a known presentation of the above sequent calculus; furthermore $*$-autonomous categories (corresponding to full multiplicative linear logic) are a conservative extension of linearly distributive categories. This indicates that linearly distributive categories are an appropriate setting in which to study the negation-free fragment of mLL.

identity axiom $A \vdash A$

units

$$(\top L)\ \frac{\Gamma_1, \Gamma_2 \vdash \Gamma_3}{\Gamma_1, \top, \Gamma_2 \vdash \Gamma_3} \qquad (\top R)\ \vdash \top$$

$$(\perp L)\ \perp \vdash \qquad (\perp R)\ \frac{\Gamma_1 \vdash \Gamma_2, \Gamma_3}{\Gamma_1 \vdash \Gamma_2, \perp, \Gamma_3}$$

tensor and par

$$(\otimes L)\ \frac{\Gamma_1, A, B, \Gamma_2 \vdash \Gamma_3}{\Gamma_1, A \otimes B, \Gamma_2 \vdash \Gamma_3} \qquad (\otimes R)\ \frac{\Gamma_1 \vdash \Gamma_2, A \quad \Delta_1 \vdash B, \Delta_2}{\Gamma_1, \Delta_1 \vdash \Gamma_2, A \otimes B, \Delta_2}$$

$$(\oplus L)\ \frac{\Gamma_1, A \vdash \Gamma_2 \quad B, \Delta_1 \vdash \Delta_2}{\Gamma_1, A \oplus B, \Delta_1 \vdash \Gamma_2, \Delta_2} \qquad (\oplus R)\ \frac{\Gamma_1 \vdash \Gamma_2, A, B, \Gamma_3}{\Gamma_1 \vdash \Gamma_2, A \oplus B, \Gamma_3}$$

CUT

$$\frac{\Gamma_1 \vdash A \quad \Delta_1, A, \Delta_2 \vdash \Delta_3}{\Delta_1, \Gamma_1, \Delta_2 \vdash \Delta_3} \qquad \frac{\Gamma_1 \vdash \Gamma_2, A, \Gamma_3 \quad A \vdash \Delta_1}{\Gamma_1 \vdash \Gamma_2, \Delta_1, \Gamma_3}$$

$$\frac{\Gamma_1 \vdash \Gamma_2, A \quad A, \Delta_1 \vdash \Delta_2}{\Gamma_1, \Delta_1 \vdash \Gamma_2, \Delta_2} \qquad \frac{\Gamma_1, A \vdash \Gamma_2 \quad \Delta_1 \vdash A, \Delta_2}{\Gamma_1, \Delta_1 \vdash \Gamma_2, \Delta_2}$$

Figure 1: Sequent calculus of mLL without negation

2.4. Linear bicategories. We can also consider the effect of taking formulas to be *typed relations* with domain and codomain in some class. Tensor and par can then be viewed as compositions on formulas, where the domain and codomain must match, e.g.:

$$A^{X \to Y} \otimes B^{Y \to Z} = (A \otimes B)^{X \to Z}.$$

The logic of these generalized relations is naturally non-commutative. The category-theoretic presentation of such data is a **linear bicategory**, which has 0-cells (corresponding to types), 1-cells (formulas), and 2-cells (proofs), with two different horizontal compositions on 1-cells (tensor and par) in addition to a vertical composition on 2-cells. Coherence conditions similar to those of linearly distributive categories must be satisfied; see [CKS] for a full exposition.

Looking ahead to planar circuits, we can introduce the 0-cells by labeling the regions of the plane in which the circuit is located (where the input and output wires are considered to extend to the edge of the planar area containing the circuit). Wires are then viewed as 1-cells, mapping from the region on the left to the region on the right. The circuit equivalences are unchanged. Thus, all of the development below, concerning planar circuits and linearly distributive categories, applies without change to planar circuits with typed regions and linear bicategories.

3. Circuit diagrams

We now consider a notion of proof nets as a system of natural deduction for non-commutative mLL without negation. In the absence of negation, it is natural to use two-sided nets. Pictorially the proof structures are presented as graphs in which edges (wires) represent formulas, and nodes (components, links, or boxes) represent logical rules and additional axioms. The non-commutative nature of the logic leads to the restriction to *planar* proof structures: that is, wires are not allowed to cross. To differentiate from the usual proof structures (as in [G87]) we shall refer to our structures suggestively as **circuit diagrams**.

Similar proof nets (without units or non-logical axioms) were developed in [A91].

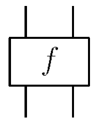
Although in this paper we shall primarily work with the graphical representation of circuit diagrams, it should be kept in mind that there is an underlying formalism supporting the proof structures. In particular the circuits are more than just graphs; *there is a specific order to the components in a given circuit*, although we will work up to equivalence modulo exchanging the order of non-interacting subcircuits (see below). The underlying formal syntax (which acts as a term calculus) is explained in [BCST].

3.1. Formulas and components. Begin with a set of **atomic formulas** $\mathcal{A}$ and **components** $\mathcal{C}$. From the atomic formulas build the **positive linear formulas** inductively as follows:

- $A \in \mathcal{A}$ is a formula;
- $\top$ and $\bot$ are formulas;
- if A and B are formulas, then $A \otimes B$ and $A \oplus B$ are formulas.

The types of the wires in a circuit are taken from these formulas.

Each component $f \in \mathcal{C}$ is equipped with two lists of types, one for the **input ports** and one for the **output ports**. We represent the component in a circuit as a box with wires of the correct types attached to the ports:



Logically, the components in $\mathcal{C}$ correspond to non-logical axioms such that the input formulas entail the output formulas; categorically, to morphisms from the tensor of the input formulas to the par of the output formulas (the formulas are the categorical objects).

There are also polymorphic components, corresponding to logical rules; call these **links** to distinguish them from the additional components in $\mathcal{C}$. The links are represented in Figure 2.

Note that the tensor-elimination and par-introduction links are drawn with a small arc; this is a reminder that the links are **switching** in the sense of Girard [G87], and the net criterion of Danos-Regnier [DR89] (discussed below). Also note the **thinning links** used for top-elimination and bottom-introduction. Formally the dotted line should be considered simply an integral part of the glyph used to represent a thinning link graphically. It is certainly not a special kind of edge. This representation will be evocative

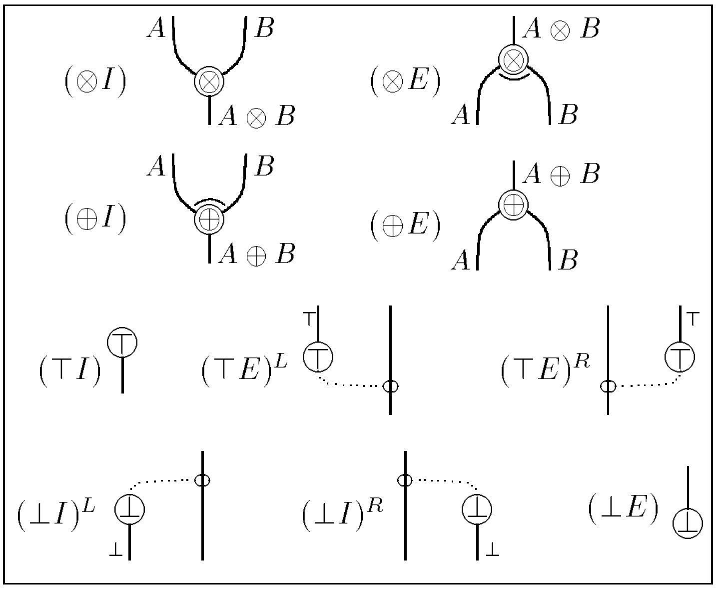
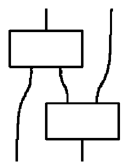


Figure 2: Links

of the special role of thinning links in the proof-net equivalence introduced later in the paper.

3.2. Juxtaposition and exchange. Components (and links) with correctly typed wires as inputs and outputs are the simplest circuits. Larger circuits are built inductively by a process of **juxtaposition**. Simply attach (some of) the output wires of one component/subcircuit to (some of) the input wires of another, in such a way that the types of the wires match. The extra input and output wires extend past the other component/subcircuit to become additional inputs or outputs of the new juxtaposed circuit. The operation is obvious in a picture:



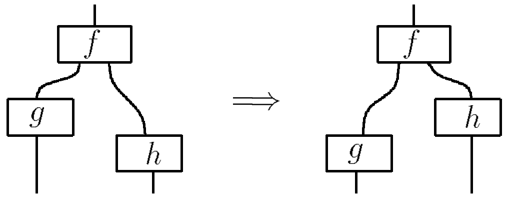
It is permissible to juxtapose without any attachments; then for definiteness, one needs to specify the order of the inputs and of the outputs of the juxtaposed circuit.

It is at this point that the non-commutative nature of the system is introduced, in the requirement that juxtaposition be *planar*; simply, wires are not allowed to cross. Also note that circuit juxtaposition is an associative operation.

We also allow the operation of **circuit exchange**[1]; two non-interacting components

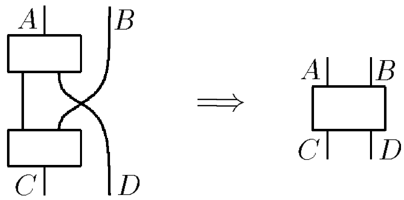
[1]Not to be confused with sequent-calculus exchange in a commutative system, which could be modelled by allowing changes in the order of the inputs and outputs of components.

(and by generalization subcircuits) can float past each other. For example:

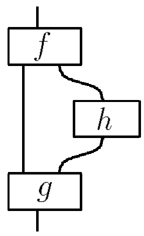


Circuits are considered equivalent up to juxtaposition reassociation and exchange of non-interacting subcircuits. This equivalence is decidable.

3.3. Rules of surgery. The core concern of this paper is working with rewrite systems on circuits. Rewrites are specified as redex/reduct pairs, which are called **rules of surgery**. When applying a rule of surgery one must be able to collect the components of the redex by circuit exchange. It is at this point that the order of components can have repercussions. As an example, consider the following possible rule of surgery (a non-planar cut):



In the following circuit, the rule cannot be applied to components f and g, since the location of component h prevents the collection of the redex, even though the appropriate graph-adjacencies are there:



A more complete discussion of circuits can be found in [BCST].

4. Sequentialization

The circuits described so far are the underlying proof structures for our natural deduction system. Only certain circuit diagrams actually represent proofs. These circuits are called **proof nets**. We check the correctness of a circuit by a process of **sequentialization**.

The sequentialization process is given as a rewrite system of redex/reduct pairs. A subcircuit, corresponding to one of the redexes, is collected (technically, using circuit

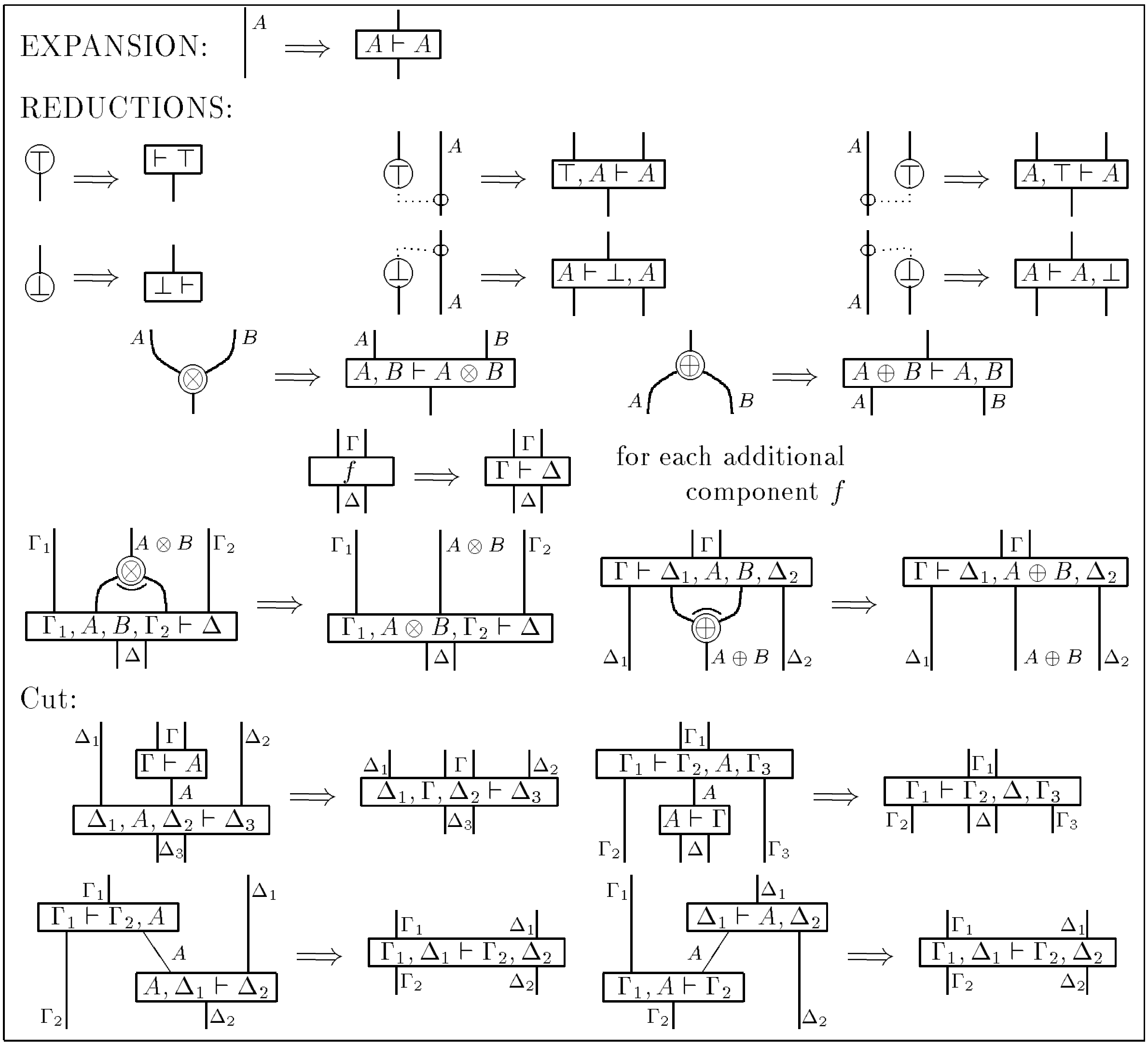


Figure 3: Sequentialization rewrites

exchange and juxtaposition reassociation) and replaced with a *sequent box* representing the sequent in which the input formulas of the redex entail the output formulas. If the entire circuit can be collected into a single sequent, the circuit is **sequentializable** and thus a proof net.

Note that proof nets actually correspond, not just to sequents, but to entire sequent calculus *proofs* in the fragment of linear logic we are considering. One can easily reconstruct a sequent calculus proof from the steps of the sequentialization process.

4.1. Original rewrite system. Figure 3 portrays the rewrites given in [BCST] for the sequentialization procedure. Again note the planar restriction on cut. We will use $\mathcal{X}$ to refer to the expansions in the system, and $\mathcal{R}$ to refer to the reductions.

(For non-planar circuits we also model sequent calculus exchange by having the inputs and outputs of the boxes be multi-sets rather than lists. In particular this removes any restriction on the cut rule.)

In the non-planar case, this system was proved to be confluent on sequentializable circuits. That is, if a circuit *can* be sequentialized, any order of sequentialization steps will succeed; if not, there may be several different ways for the process to become stuck. It was thought that the proof of confluence transferred correctly to planar circuits. In fact this is not the case, as the example in Figure 4 shows. In this example, the reduced circuit on the right will sequentialize fully with the rules we have; but no rules apply to the circuit on the left. Because of the complications this introduces, particularly with respect to the notion of circuit equivalence to be discussed below, it seems preferable to regard this apparently non-sequentializable circuit as a proof net. Thus we need to have more sequentialization rules.

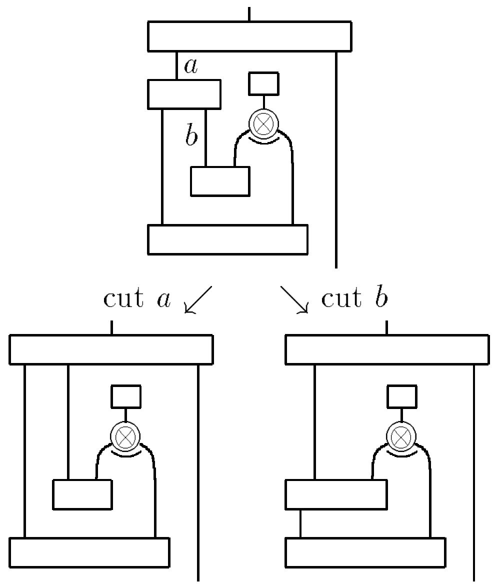


Figure 4: Sequentialization not confluent

4.2. New rewrite system. The problem evidently arises due to the non-commutative restriction on cut—working with non-planar rules, we could simply cut the offending wires in the above situation. It seems also that the *association* of the inputs and outputs of a box are more important in the absence of exchange: recall that from one perspective a box maps the tensor of its inputs to the par of its outputs, but how associated? This helps to motivate the rules we will add to the system. These rules can also be considered as ways to pass to an equivalent circuit from which we can sequentialize further using the original rules; this will be made precise in Lemma 5.7.

The rules in Figure 5 are *equations* or *$\mathcal{E}$-moves*; they may be performed in either direction. This increases the complexity of the rewrite system: a normal form can only be specified up to equivalence modulo the equations, and in the process of rewriting one may need to continuously switch back and forth between $\mathcal{E}$-equivalent forms.

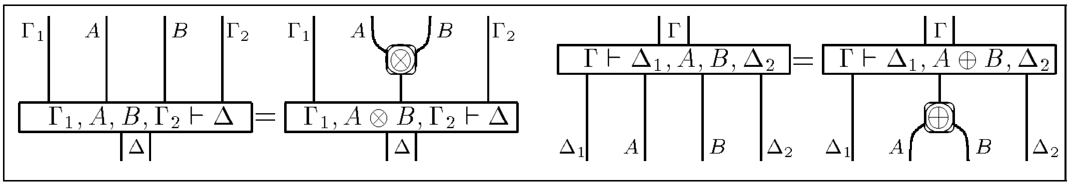


Figure 5: New sequentialization equations

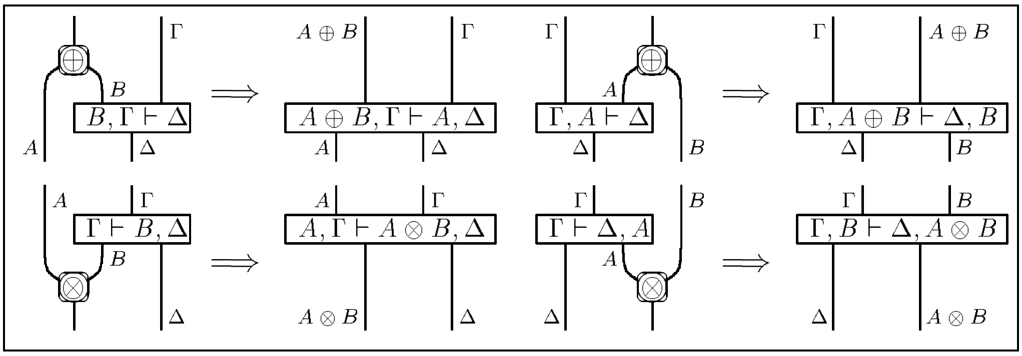


Figure 6: New sequentialization reductions

These rules model reassociation by allowing a box to extend and retract times or plus links between any two adjacent inputs or output. We mark these plus and times links to distinguish them from the links originally in the circuit. In particular an original link may not be pulled into a box by an $\mathcal{E}$-move, and a marked link (called a *reassociation link*) may not be replaced with a box by one of the original $\mathcal{R}$-moves[2]. However, note that this distinction is to be considered a temporary device used only in establishing whether a circuit is sequentializable; we will not have two kinds of tensor or par links in any other setting.

Naturally we want to be able to do something with these reassociation links; motivated by the problematic example above, add the new reduction rules in Figure 6 (which, again,

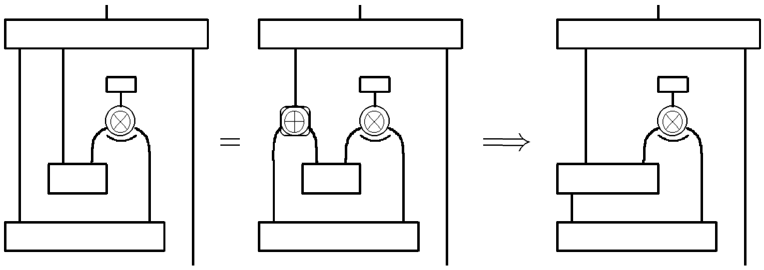
[2]This distinction may seem unnecessary. After all, the effect of applying an $\mathcal{E}$-move to an original link could be duplicated by using a reduction to replace it with a box, and then cutting the intervening wire; and the effect of replacing a reassociation link with a box could be duplicated by expanding the wire behind the reassociation link, then pulling it into the new box with an $\mathcal{E}$-move. The distinction should probably be regarded as a purely technical device to ensure that the system's reductions, expansions, and equations have certain appropriate properties which imply confluence.

For suppose that the $\mathcal{E}$-moves applied to the original links. Then the system would not be *reduction $\mathcal{E}$-terminating*, because as just noted it would be possible to duplicate one direction of an equation with a series of reductions. One might attempt to avoid this problem by forbidding the reduction rules which box non-switching tensor and par, since those rules could be duplicated with an expansion and an equation. However, with this change the system would not be *$\mathcal{X}$-reducing*, since one could find a divergence of two reductions where an expansion is required in any convergence.

In practice, keeping the two kinds of links apart is never an issue; the sensible way of starting a sequentialization is to box all the non-switching links immediately, and from there one need only consider boxes, switching links, and reassociation links.

apply only to the reassociation plus and times links, not the original ones).

This certainly solves the problem example. In a loose sense, these rules allow us to undo the wrong cut and redo it in the right way:



It will now be shown that these rules are enough for confluence.

4.3. Confluence. First observe that once a plus or times link is extended by one of the equations, there will always be a box into which to retract it (possibly after retracting other reassociation links that were later extended), regardless of intervening applications of rules. It is thus easy to see that an $\mathcal{E}$-equivalence class of circuits contains a unique **fully retracted form**, in which all the reassociation plus and times links are pulled back into boxes.

In terms of retracted forms, any application of the new reduction rules will have the form of Figure 7, or the corresponding operation for plus on the right, or times on the left or right. Figure 8 shows the derivation of this transformation; note that the desired circuit exchange will always be possible in a planar circuit. Call this derived rule *jump*, and say that the wire of type A in Figure 8 (the *jump* wire) *jumps along* the wire of type B (the *along* wire). The jump wire must be adjacent to the along wire (as inputs or outputs of a box) in any application of the jump rule.

Observe that the jump rule is a non-local operation; it cannot be formulated in terms of collecting a redex and substituting a reduct, since intervening components may prevent such a collection. Thus, the system is formulated using the local equations and reductions instead. Nonetheless, since in proving the confluence of the system we have to work up to $\mathcal{E}$-equivalence, it is convenient to refer to the retracted forms, and to what amounts

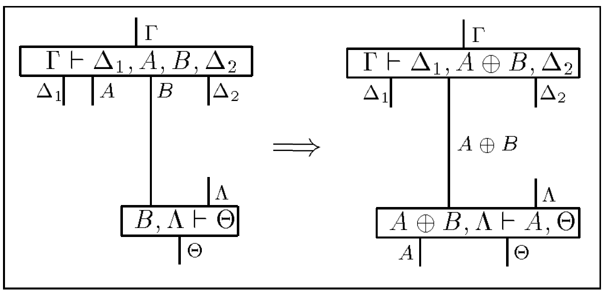


Figure 7: The jump rule

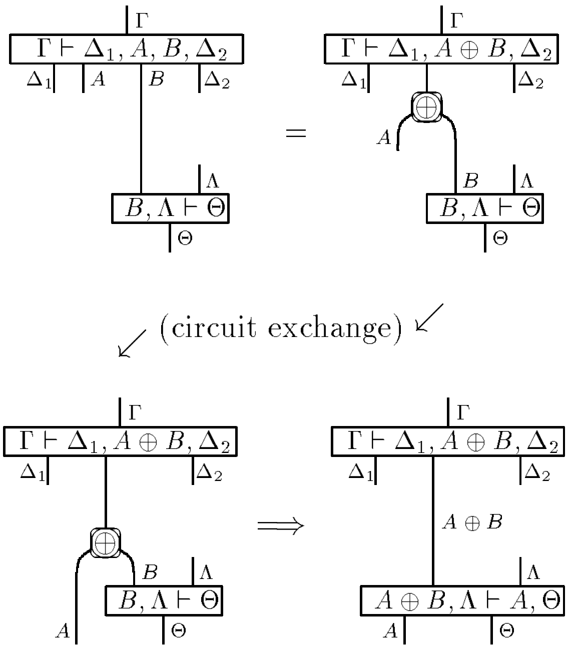
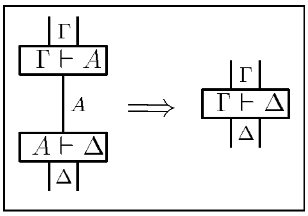


Figure 8: Derivation of the jump rule

to applications of these non-local rules. (The non-local character of the rule may make it seem that the rule is non-planar, but we are indeed still requiring planarity.)

Next observe that the rewrite system can be simplified by removing most kinds of cut. Allow only the most simple cuts, where there are no inputs or outputs adjacent to the cut wire:



The full power of non-commutative cut can be reproduced, using a series of jumps along the cut wire to remove any inputs or outputs adjacent to it, followed by an application of the restricted cut. It is convenient, in proving confluence, to leave the full cut out of the system. However, as we have any non-commutative cut as a derived rule, we will not hesitate to use the full form when needed.

We are now ready to prove that we can check the sequentializability of a circuit in an effective way.

4.4. Theorem. *Given any sequentializable circuit, one can rewrite in any order to achieve sequentialization, with a guaranteed bound on the number of reductions and irreducible expansions. Moreover, it is possible to determine whether a circuit is sequentializable in only a bounded number of moves including equations.*

Proof. First note that the theorem allows that given a non-sequentializable circuit, it is possible that there are multiple ways for the process to end (none of them single sequents).

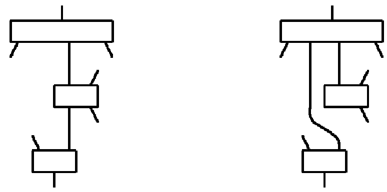
Following Appendix A, for the first statement it suffices to show:

- $\mathcal{R}$ is $\mathcal{E}$-terminating;
- $\mathcal{X} \cup \mathcal{R}$ is locally $\mathcal{E}$-confluent and $\mathcal{X}$-reducing; and
- $\mathcal{X}/\mathcal{R}$ is expansion $\mathcal{E}$-terminating;

where we are considering the rewrite system only on sequentializable circuits.

• **$\mathcal{R}$ is $\mathcal{E}$-terminating.** Assign to each sequent box a **height** and a **depth**. The height of a box is the maximum, under circuit exchange, of the number of boxes that come after it in the circuit; depth is the maximum number that can come before it. Roughly, the height of a box B is the greatest number of boxes we can manage to draw below B in a graphical representation of a circuit; similarly, the depth is the greatest number we can draw above B.

It is easy to see that the jump rule can only *decrease* the height or depth of any box. Any ordering on the boxes after the jump could be reproduced in the circuit before the jump, e.g.:



after jump *before* jump

If we can circuit-exchange n boxes past B after the jump, we can certainly do the same (or more) before the jump.

Moreover, considering the circuits in fully retracted form, a par jump (a jump downward) sends an output wire from a box of greater height to a box of strictly less height, since the along wire keeps the ending box after the starting box in any configuration. Similarly a tensor jump (a jump upward) sends an input wire from a box of greater depth to a box of strictly less depth.

Thus, an application of the jump rule always reduces the quantity

$$\sum_{\text{boxes}} [\text{depth} \cdot \text{number of inputs} + \text{height} \cdot \text{number of outputs}]$$

where inputs and outputs are counted as for the fully retracted form of each circuit.

Now observe that any reduction reduces the lexicographic order on

- the number of (non-equation) links/components in the circuit;
- the number of sequent boxes;

- $\sum_{\text{boxes}}$[depth $\cdot$ number of inputs + height $\cdot$ number of outputs], using fully retracted values.

The third value is reduced by any jump (which leaves the first two values unchanged); the second value is reduced by any cut (which leaves the first value unchanged); the first value is reduced by any other reduction.

Moreover it is clear (since all relevant values are counted in fully retracted form) that this quantity is not changed by $\mathcal{E}$-moves. It follows that the reductions are $\mathcal{E}$-terminating.

We will use this value again to obtain confluence; refer to the above triple as the *circuit value*.

• **The expansion.** Technically $\mathcal{X}$*-reducing* and *expansion* $\mathcal{E}$*-terminating* only make sense after local confluence is established. However it will simplify the presentation to get the lone expansion out of the way from the beginning; we can do this by noting the following.

In a *sequentializable* circuit containing *any* links or components, the expansion is always reducible. For, if the expanded wire is next to a box, the expansion box can be cut into that box; if it next to a non-switching link, the link can be boxed and then the intervening wire can be cut. If the expansion is next to a switching link, we know by sequentializability that the switching link will eventually be absorbed into a box, and the expansion box can be cut at that time. It is easy to see that in each case the same result could be obtained using only reductions. (To be precise, this shows only that the *last* expansion of the sequentialization process is reducible; an immediate induction on the number of expansions completes the argument.)

Thus the expansion is only irreducible when applied to a circuit which is a single wire (and at that point, sequentialization is complete). So if we can prove that $\mathcal{R}$ is locally $\mathcal{E}$-confluent, then it will follow immediately that $\mathcal{X} \cup \mathcal{R}$ is locally $\mathcal{E}$-confluent and $\mathcal{X}$-reducing, and that $\mathcal{X}/\mathcal{R}$ is expansion $\mathcal{E}$-terminating.

• **$\mathcal{R}$ is locally $\mathcal{E}$-confluent.** To deal with the $\mathcal{E}$-moves in determining possible divergences, simply treat all circuits as if in fully retracted form, and allow jump as a non-local move.

Confluence is proved by induction on the circuit value defined above. Clearly confluence holds on sole sequent boxes (circuit value $(0, 1, 0)$). Assume that confluence holds on all sequentializable circuits of value less than α; now take a circuit of value α and consider the possible divergences.

First observe that boxing a link/component can never lead to a divergence. This leaves

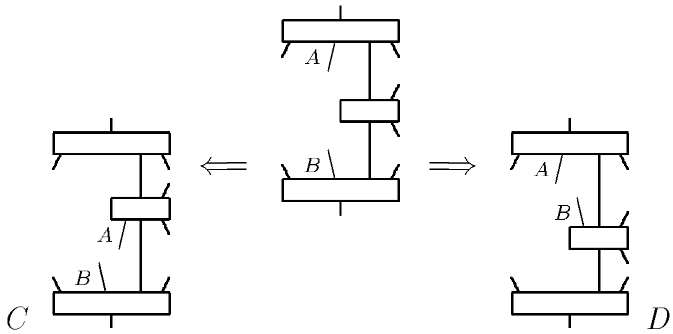
1. absorbing a switching link;
2. (restricted) cut;
3. (non-local) jump.

Note the following consequence of planarity: if the correct configuration for an application of these rules exists non-locally (that is, within the circuit diagram just viewed as a graph), it is always possible to do the circuit exchanges necessary to collect the redex. (For the jump rule equations will also be necessary.) It follows that a divergence can occur *only* between two redexes sharing boxes or wires. This makes listing the critical divergences much easier.

Now we proceed by exhibiting examples of the various critical divergences which can arise, and how to resolve them. The easy cases, where the two moves can be done, though not simultaneously, in either order for the same result, are omitted. Also note that a convergence can be considered valid as long as the configurations of boxes and wires are the same, even if some of the types of the internal wires are different (perhaps $A \oplus (B \oplus C)$ rather than $(A \oplus B) \oplus C$). Since the circuit is assumed to be sequentializable, confluence up to changing the types of internal wires does not really give anything away, since such wires will eventually be cut (thus giving confluence in a stricter sense as well).

For most cases, finding a convergence will not require the induction hypothesis. Figures 9–13 portray convergences for all but one of the non-trivial critical divergences.

The one difficult case is when there are competing jumps to the same box. An example of such a divergence would resemble the following:



Given a divergence of this sort, call the two possible circuits so obtained C and D. We may assume that one of C or D is sequentializable. (If not, there would be another critical divergence between C and the first step of the "correct" sequentialization, the resolution of which would show C to be sequentializable.)

Without loss of generality, suppose C is sequentializable. Since C is a reduction of the original circuit, it has a lower circuit value. Then, by the induction hypothesis, we can sequentialize in any order on C.

Our strategy will be to follow a sequentialization of C, mimicking each step also on the circuit D, until a convergence occurs. This may go so far as showing there is a convergence by showing that both circuits can be sequentialized all the way to single sequent boxes! We must ensure that the two circuits remain similar enough that sequentialization steps on one can be successfully mimicked on the other. In fact, we shall maintain the following three invariants:

1. We maintain that the circuits are identical up to a *divergent stack*, consisting of a stack of boxes with *central wires* connecting all along. Call the other wires in the

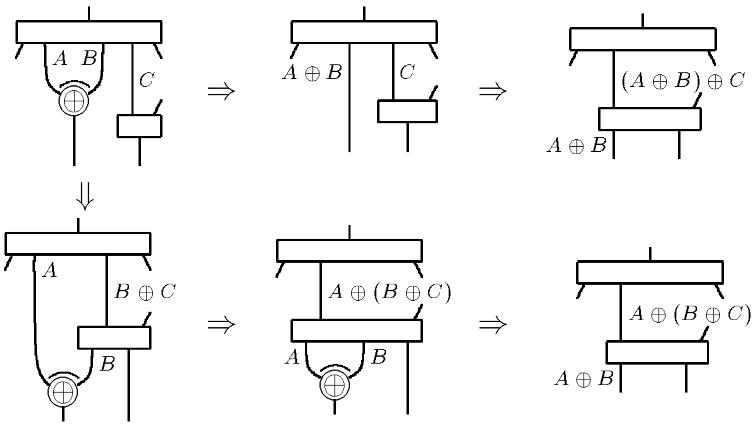


Figure 9: Switching link absorption broken by jump

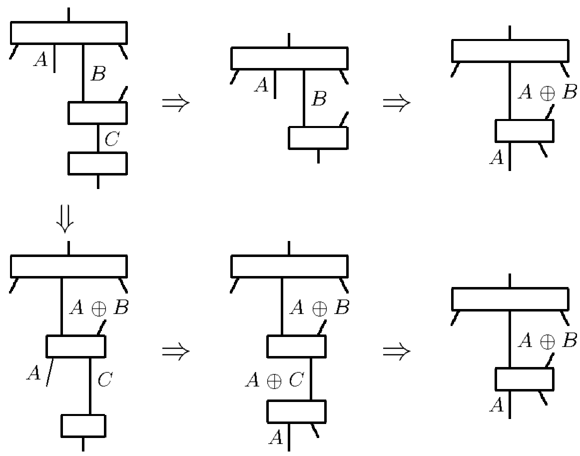


Figure 10: Cut interrupted by jump

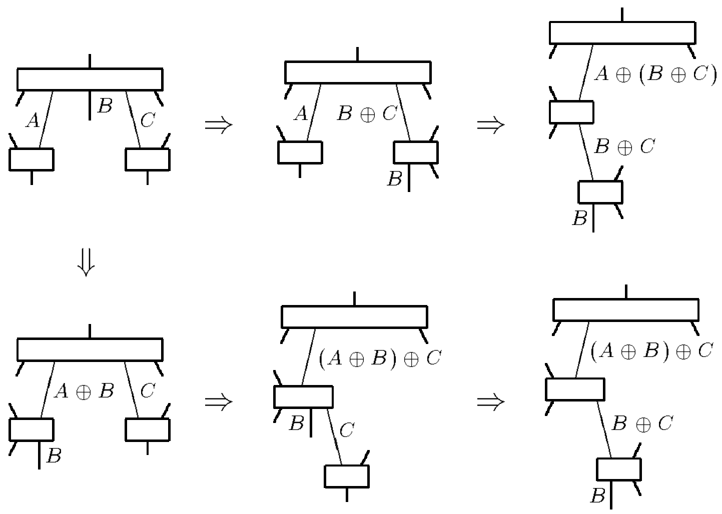


Figure 11: Two jumps, same jump wire

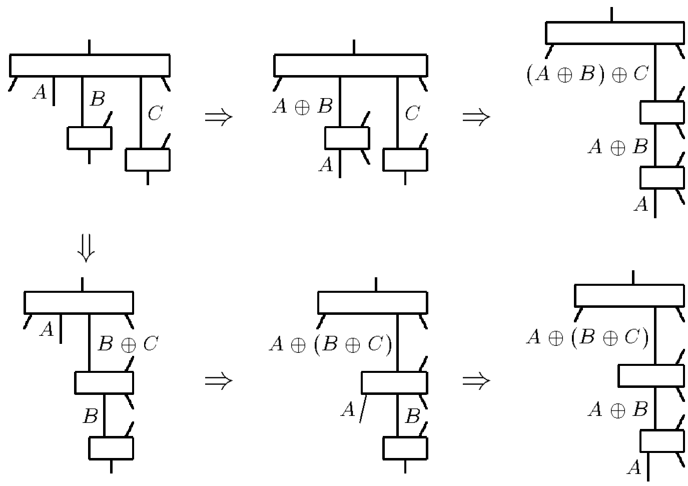


Figure 12: Along wire jumps

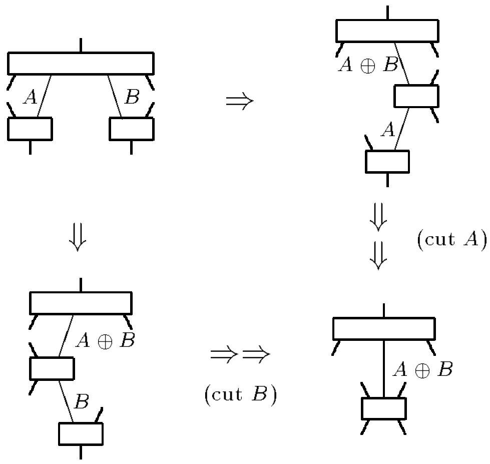


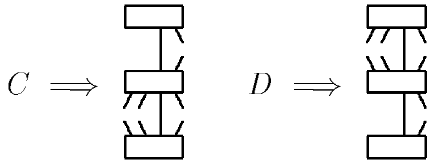
Figure 13: Mutual jump

stack (which connect to the non-divergent parts of the circuits) *side wires*. (Note that the inputs at the top of the stack and the outputs at the bottom of the stack are not divergent. Also note that the number of boxes in the stack is not invariant.)

2. Call two wires *adjacent* if they are consecutive inputs or outputs of a box. Then we maintain that two side wires adjacent in one circuit are adjacent in the other, and that a side wire adjacent to a central wire in one circuit is adjacent to a central wire in the other; and further, that the left-to-right ordering of the adjacencies is the same in both circuits. (Here we are referring to the side wire of C and the side wire of D, which connect to the non-divergent parts of the circuits in the same way, as if they were the same wire.)

3. View the side wires as if collected into sets, by grouping the adjacent inputs or outputs of each box. We will have distinct sets on the left and right sides of the central wires. We maintain that the top-to-bottom ordering of the sets on a given side of the central wires will be the same in both circuits.

Observe that at the beginning of this process the two circuits are unlikely to satisfy the adjacency condition of the second invariant; a series of jumps of the side wires is necessary. Returning to the particular C and D portrayed in the example divergence above, we would need to make jumps as follows (where only the divergent stack is portrayed):



In general, maintaining the adjacency condition may mean inserting jumps, after performing a sequentialization step of C and mimicking it with an corresponding step on D.

Because sequentialization is confluent on C we can add these jumps if necessary without losing sequentializability.

To motivate that there should be a convergence, observe that the divergent stacks of the two circuits differ essentially only the relative position of the side wires to the left of the central wires, with respect to those to the right of the central wires. But because of planarity, there is no interaction across the central wires. In particular, any rewrite depends on (at most) two adjacent wires, so as long as the adjacencies are maintained, we will continue to be able to mimic sequentialization steps in both circuits.

If the first invariant above can be maintained throughout a sequentialization of C, then eventually a convergence will be reached. For, at some point in the sequentialization of C, all the central wires of the divergent stack will be cut. This means that there will be no side wires in C, and hence none in D as the side wires all connect to the non-divergent part of each circuit. Thus all the central wires of D can be cut, and the non-divergent part will be all of the circuit.

It remains only to see that a sequentialization of C can always proceed (and be mimicked on D) in a way which preserves these invariants. Here again it helps to impose a restricted cut. There are several cases:

- Sequentialization steps applied in the non-divergent part (these include all reductions which only put a box around links) can be mimicked directly, and do not affect the invariants.

- Absorbing a switching link into a box of the divergent stack can be mimicked directly (since the two wires leading to the switching link must be adjacent in both circuits), and does not affect the invariants.

- Jumping a side wire along another side wire can be mimicked directly (by considering adjacency) and does not affect the invariants.

- Jumping a side wire along a central wire, or creating a new side wire by jumping a wire into the divergent stack, can always be mimicked directly (by the adjacency condition and also the third invariant concerning the top-to-bottom order of sets of side wires). However, in order to maintain the invariants it may be necessary to further jump the wire in one of the circuits. In particular, if in one circuit the wire is jumped into an established set of side wires, it must also reach that set in the other; and if in one circuit the wire is jumped out of the divergent section, it must be jumped out in the other. That this is always possible follows from the third invariant.

  An example of what this might look like, at some stage of attempting to find a convergence for some particular C' and D', is given by Figure 14. The step leading up to the portrayed configuration would be a jump bringing the A wire up into the divergent section. Only the divergent stack is portrayed.

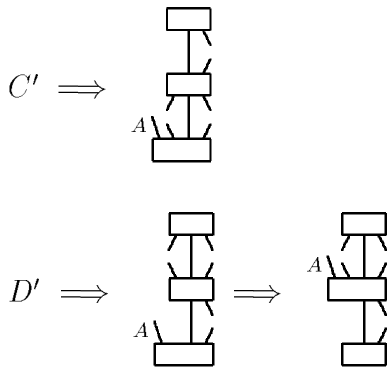


Figure 14: Maintaining adjacencies

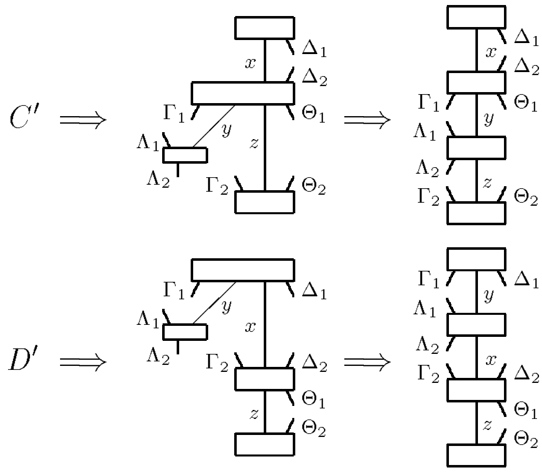


Figure 15: Jumping a central wire

- Cuts of central wires may not always be possible to mimic. However, not mimicking these cuts doesn't affect the invariants. It is best to simply cut central wires whenever possible in either circuit.

- Finally, consider jumps of the central wires along side wires. These jumps can be mimicked directly, but the divergent stack will grow by one level after the jump—it will now include the box jumped to, and the along wire of the jump as a central wire. An example of what this might look like, at some stage of attempting to find a convergence for some particular C' and D', is given by Figure 15. Only the divergent stacks are portrayed.

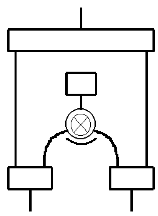
This completes the proof of confluence.

There remains the final statement of the theorem, i.e., that we can bound the number of $\mathcal{E}$-moves we need to apply in order to check sequentialization. Note that there are only a finite number of circuits $\mathcal{E}$-equivalent to a given circuit, because there are only a finite number of ways to fully determine the associations of the inputs and outputs of each box. It follows that we can check whether a reduction can be applied to any $\mathcal{E}$-equivalent form, with a bound on the number of equations it is necessary to apply. The proof that $\mathcal{R}$ is $\mathcal{E}$-terminating applies to all circuits (not just sequentializable ones); thus we can always reach a circuit, where no reductions can be applied to any $\mathcal{E}$-equivalent form, with a bound on the number of moves needed. The result follows. ∎

4.5. The net criterion. A typical criterion for a circuit to be a proof net follows along the lines of the following (see [DR89]):

4.6. Definition. *Regard each switching component in a circuit as having a switch which chooses between the non-tensored ports: the inputs of the* $(\oplus I)$ *or the outputs of the* $(\otimes E)$*. One port is considered to be connected, the other disconnected. Then the circuit satisfies the* ***net criterion*** *if for any choice of switch settings, the undirected graph determined by the connected wires is acyclic and connected.*

It is proved in [BCST] that, for non-planar circuits, sequentializability is equivalent to satisfying the net criterion. However the net criterion is not a sufficient condition for a planar circuit to be sequentializable, as the following example shows.



(The example given in [BCST] to support this claim turns out to be sequentializable given the modifications of this paper.) In fact, a planar circuit is sequentializable if and only it it satisfies the commutative net criterion and also the following *region criterion*:

4.7. Definition. *A planar circuit divides the planar area containing it into regions; that is, if the circuit is viewed as a graph embedded in a planar area, the edges of the graph (and the border of the area containing it) divide that area into regions in the usual sense.*

Call a region closed *if it is surrounded by the circuit, and* open *if it reaches the border of the area containing the circuit. Further say that a switching link* opens *onto the region between its non-tensored ports: the region above and between the inputs of a* $(\oplus I)$*, or the region below and between the outputs of a* $(\otimes E)$*.*

Then a circuit satisfies the ***region criterion*** *if every switching link opens onto a closed region, and every closed region has one and only one switching link which opens onto it.*

I intend to publish a proof of this result in a sequel.

5. Circuit equivalence

Now we introduce rules of surgery which define an equivalence relation on proof nets. Proof nets modulo this circuit equivalence will define a (non-symmetric) linearly distributive category—in fact, they will define the free such category on the polygraph of components on which the circuits are based. Equivalent proof nets will thus correspond to a single morphism of a non-symmetric linearly distributive category, and by extension to a single proof in non-commutative mLL without negation.

5.1. Rules of surgery. First we consider the original set of circuit-equivalence rules of surgery given in [BCST]. There is an important assumption underlying these rules: *a rule of surgery can only be applied to a proof net if it preserves sequentializability.* This is essentially an element of non-localness in the apparently local rules. However, this non-localness can be isolated into a very few cases: namely, the equations which rewire thinning links past switching components. It is immediate to check that all the other rules automatically preserve sequentializability.

The reductions are listed in Figure 16; the expansions (which can be thought of as "expressing the type of the wire") are listed in Figure 17.

The equation rules deal with the manipulation of thinning links. This manipulation is called *rewiring*; it is the core of this approach to coherence. To motivate this, note that the thinning links represent where a unit has been introduced into the proof by thinning. There are typically many steps in a proof at which this might occur, without (so one would desire) essentially changing the proof.

The first set of equation rules (Figure 18) deals with rewiring past components and all non-switching links (tensor-introduction, cotensor-elimination and all the unit links). These are called "box rewiring" rules. Note the difference in behavior between the top and the bottom: only the top can be rewired around the top of a component, and only a bottom around the bottom of a component.

Finally there is a set of equation rules (Figure 19) dealing with rewiring past switching links. Note the important fact that neither unit can be rewired directly across the "opening" of a switching link (that is, between the link's non-tensored ports). Also recall that these are the only rules that require one to check that sequentializability is preserved

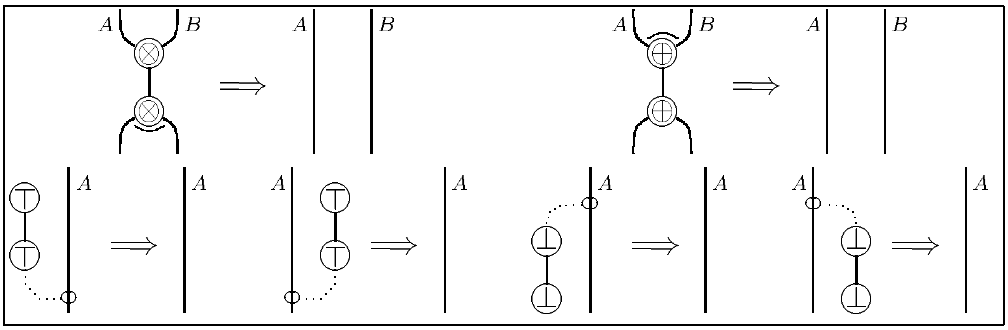


Figure 16: Reductions

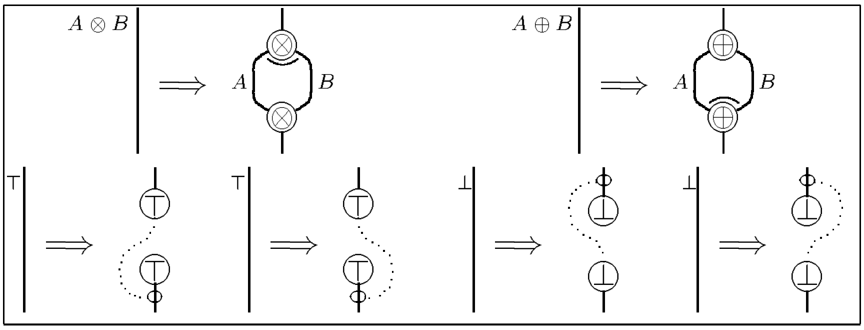


Figure 17: Expansions

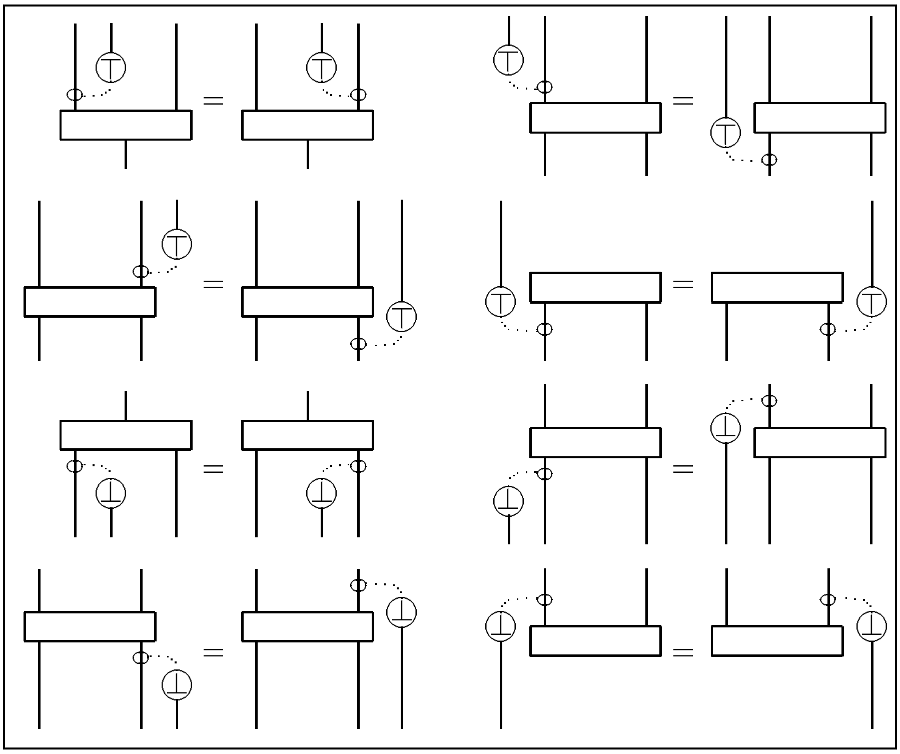


Figure 18: Box rewiring equations

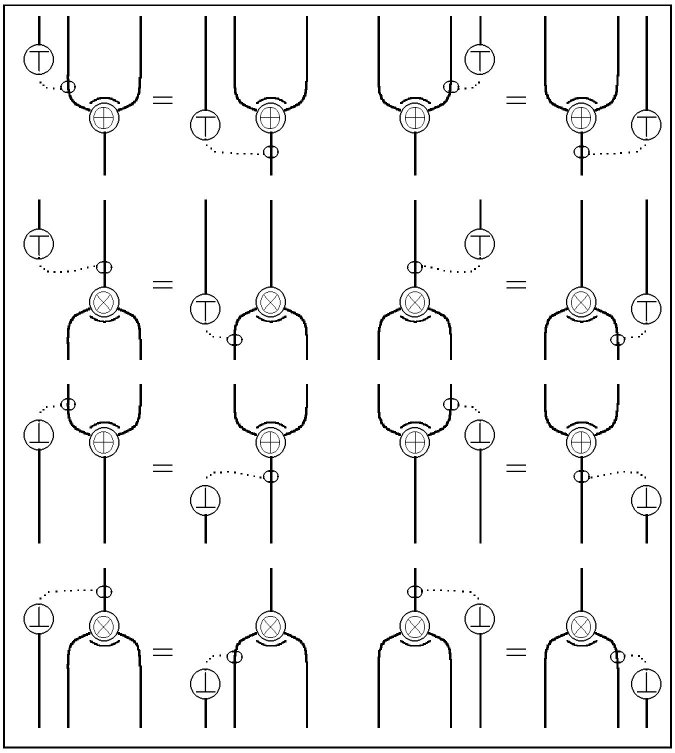


Figure 19: Equations for rewiring past switching links

by the rewrite.

(Additional rewirings must be added to the system for the commutative case. In particular, in non-planar nets, tops can be rewired around the bottom edges of components, and bottoms around top edges; furthermore there are rules to reverse the left/right orientation of a thinning link. The result is that possible rewirings in *non-planar* nets are not restricted by the handedness of the thinning link or the type of unit involved; in fact, rewiring is not even restricted by the order in which several thinning links are attached to a single wire. The situation is quite different in the non-commutative case.)

5.2. Planar modifications: negation links. The rewrite system from [BCST] developed above is not confluent on planar proof nets, as the authors were aware. Consider the series of circuits, equivalent modulo rewiring, portrayed in Figure 20. It is easy to see that if the obvious reduction were applied to eliminate the "floating unit barbell" from the first or last circuit in the series, no further rewrites would apply to the circuit.

In the commutative scenario these two circuits would in fact be equivalent. But as planar circuits there is nothing more to be done with them. This kind of undesirable behavior seems to be a direct result of the fact that, in the planar case, certain rewiring moves can be made by tops but not by bottoms and vice versa.

In [BCST] this problem was avoided by admitting the unit reduction rules as equations—then, one can add and remove unit barbells at will, and in that way units can be moved past difficult situations. The authors point out, however, that this is an

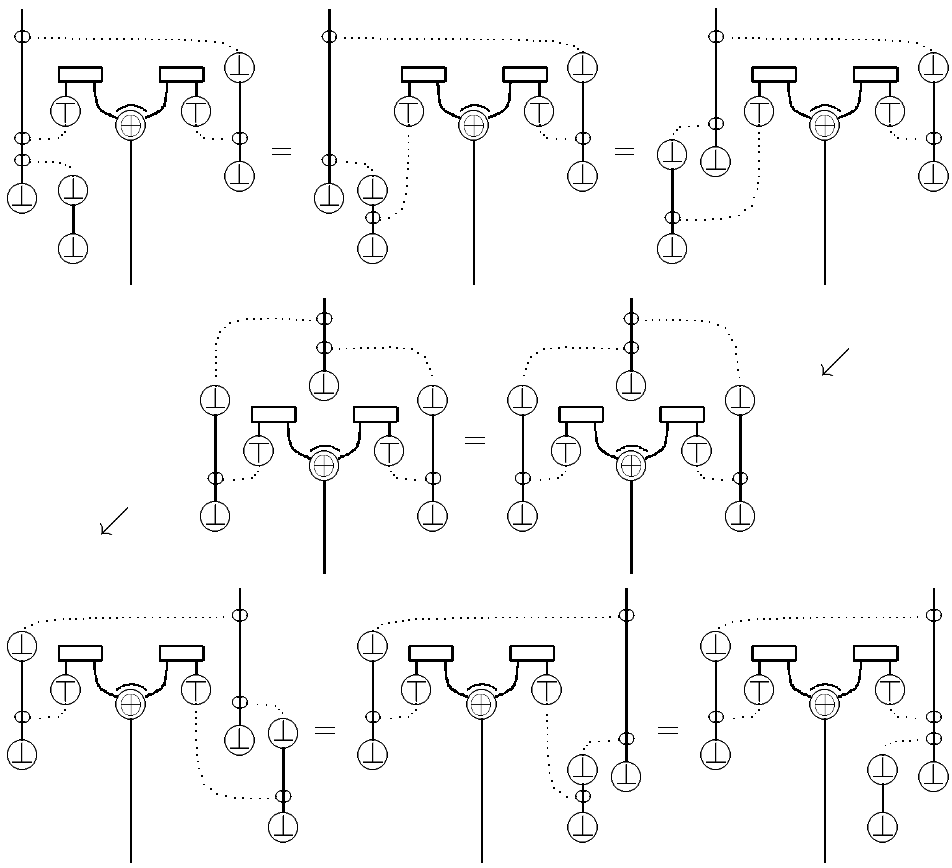


Figure 20: Circuit equivalence not confluent

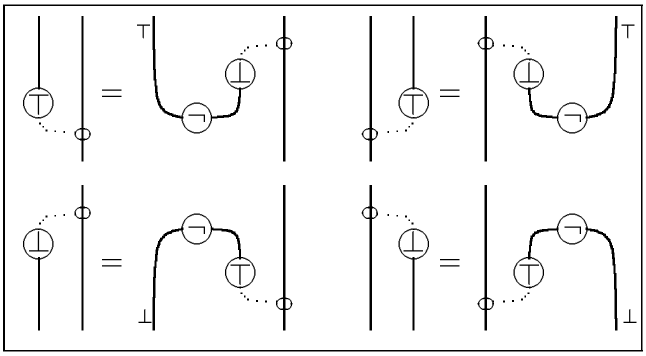


Figure 21: Negation link equation rules

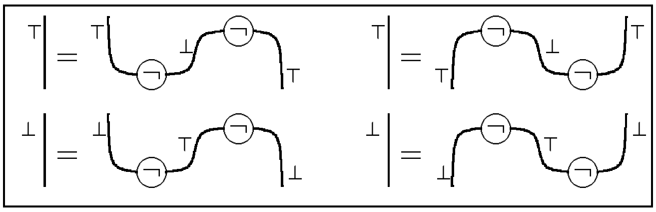


Figure 22: Negation link equation rules (double negation)

unsatisfying solution, because the possibility of an arbitrary number of floating barbells makes a decision procedure for coherence seem unlikely.

A more manageable means of interchanging the units is created by introducing a *metalogical* link that can only be introduced in a controlled way. This link should, in a certain sense, be viewed as an abbreviation for a unit barbell (this is made more precise in Lemma 5.6 below). Due to the manner in which this link interchanges the dual units, call the link a *negation* link. Add the equations in Figure 21 to the rewrite system to deal with it. By means of these equations, the top and bottom links have identical mobility. This resolves problems like the non-confluent example above. For convenience, add further equations, listed in Figure 22, to add or remove double negations.

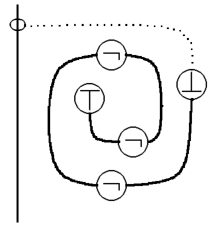
Refer to a series of negation links ending in a thinning link as a *negation link chain.* A negation link chain forms a kind of extended thinning link which is more flexible. It is convenient to restrict the negation links in a circuit to those occurring in negation link chains. The equations for negation links next to thinning links clearly maintain this restriction; the double negation rules can be appropriately restricted by requiring that they only be applied within a negation link chain.[3]

By imagining a trip in the circuit along a negation link chain, we can assign a hand-

[3]In terms of the redex/reduct presentation, this can be done by replacing each of the rules in Figure 22 with six rules, one for each of the various ways that either a negation link or a thinning link can be attached, at either the input or the output, to both redex and reduct. These are: a negation link at the input to the left or right, a negation link at the output to the left or right, and at the appropriate side a thinning link to the left or right.

edness to each link in the chain (including its final thinning link), according to whether the trip makes a left or right turn at the link. Using the equations, we can remove any negation links which fail to reverse the handedness of the next connection. The negation links left after such an operation will be those that produce a *winding* in a negation link chain. A negation link chain is said to be **tightly wound** when all the negation links reverse the handedness of the next connection in the chain.

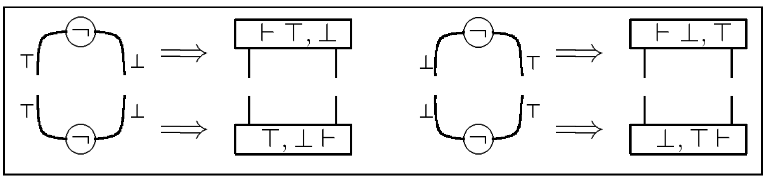
Consider a negation link chain ending in a top-introduction or bottom-elimination link, which is like a unit barbell that somehow manages to get wound up, e.g.:



In keeping with the idea that negation link chains should behave like chains of unit barbells, we should be able to reduce such structures. To this end, extend the unit reductions to infinite series of reductions to remove any such chains. Note that we need series for top and bottom units, left and right side of wire, *and* clockwise and counterclockwise winding, but due to the equations we may assume that the chain is tightly wound.

Observe that no matter how such an "extended barbell" is wound around other components in the circuit, one can always use circuit exchange to collect it into a single winding, because it is only attached to the rest of the circuit at the end. Thus the redex/reduct presentation for the rules is retained.

For the purposes of sequentialization and rewiring, the negation link is a non-switching link. Add the following rules to the sequentialization rewrite system (clearly that rewrite system's important properties are unchanged):



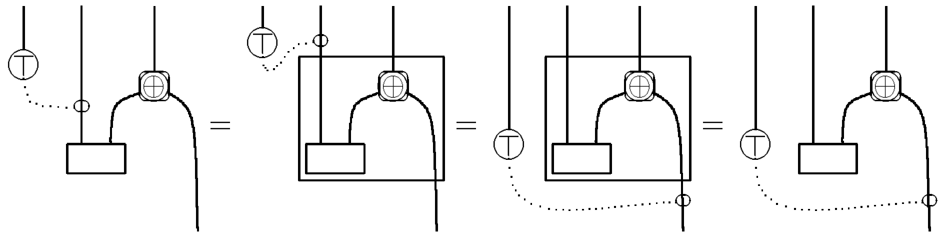
This expands our notion of proof net to include negation links. Again, it is convenient to restrict proof nets to be only those sequentializable circuits in which all negation links occur within negation link chains.

5.3. Sequent box rewiring. Next it is proven that the "box rewiring" rules can be extended so that the boxes can represent, not just components and non-switching links, but *sequent boxes* which can be collected during the sequentialization process. This powerful derived rule lets us move the thinning links in large steps. This will be needed to establish the coherence result in section 6.

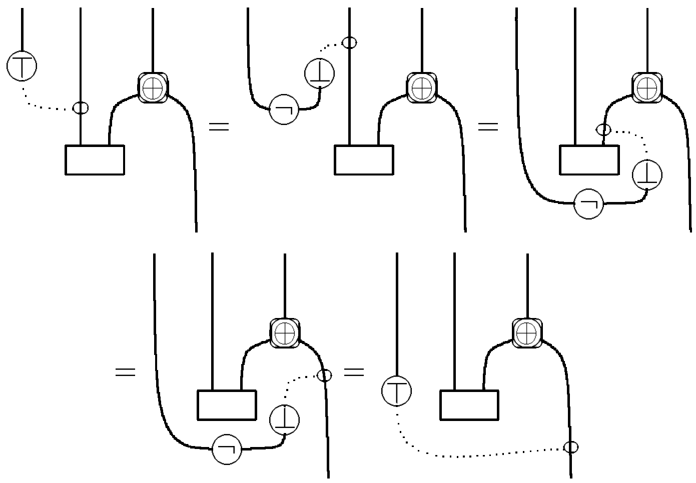
5.4. Proposition. *The box rewiring rules apply to sequent boxes, in the following sense. Consider a circuit C, and suppose a circuit D can be reached from C by applying the sequentialization rules. If a thinning link can be rewired between two positions in D, by allowing the box rewiring rules to apply to the sequent boxes, then the thinning link can be rewired between the corresponding positions*[4] *in the original circuit C.*

Proof. It suffices to check that for each sequentialization rule, all rewirings (including box rewirings around sequent boxes) which are possible after the rule are also possible before the rule. This is immediate for the rules that box non-switching links, and straightforward for most of the other rules. (It is helpful to heavily restrict the cut rule as was done in the proof that sequentialization is confluent.)

The only tricky case is a jump when the box involved has either no inputs or no outputs, for example:



In [BCST][5] this was handled by adding a floating barbell; instead negation links can now be used as follows:



This completes the proof. ∎

[4]Making precise the required notion of correspondence is not difficult: for each possible position for a thinning link to connect to each sequentialization reduct, assign an appropriate position in the corresponding redex. The choice is obvious in all cases except perhaps for the expansion and the equations, where new wires are created. For the expansion both the input wire and the output wire correspond to the original unexpanded wire; for the equations, the left side and the right side of the new wire correspond to the left side of the leftmost, and the right side of the rightmost, of the two distinct wires which are involved in the equation.

[5]At that time the tricky case was a cut, because jump didn't yet exist as such.

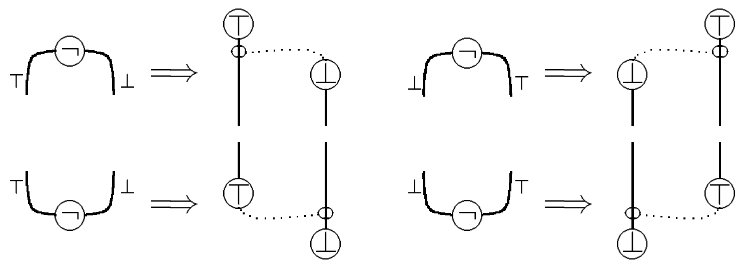
5.5. Proof nets as linearly distributive categories. Planar proof nets with one input and one output can be regarded as a category, where each sequentializable circuit is a morphism from the type of the input wire to the type of the output wire.[6] Composition is simply circuit juxtaposition, which as noted is an associative operation; sequentializability is preserved under juxtaposition since if each subcircuit can be sequentialized, then the two sequent boxes so obtained can be cut together to sequentialize the full circuit. The identities are the single wires.

Now, equivalence of circuits under the rewrite system defined above forms a congruence on the category described so far. Denote by $\mathrm{PNet}(\mathcal{C})$ the category of planar proof nets based on the set of components $\mathcal{C}$, quotiented by circuit equivalence.

It is now shown that the new equivalences using the negation links do not collapse or create equivalence classes of nets.

5.6. Lemma. *Any net involving a negation link is equivalent to a net not involving negation. Furthermore two nets, which do not involve any negation links and which are equivalent under the above rules of surgery (including the negation rules), are also equivalent under the original rules (not including the negation rules, but including the unit reductions as bidirectional equations).*

Proof. Consider the following translation:



By this translation all the negation links can be removed from a circuit. In fact, because negation links are always adjacent to other negation links or to thinning links, each negation link will be replaced by a thinning link attached to a unit barbell. This new circuit is equivalent to the original: for every link attached to a unit barbell in the new circuit, where the original circuit has a negation link, simply introduce a negation, move the new dual unit off the barbell, and then eliminate the barbell.

Furthermore, referring to the translation, it is easily seen that the equations which introduce and eliminate negation links can be simulated by introducing and eliminating unit barbells; and the negation link chain reductions can be simulated by a series of barbell eliminations. Thus it is possible to show the equivalence of two equivalent nets, which lack negation links, without using the negation rules. ∎

[6]We could view proof nets with multiple inputs or outputs as the morphisms of a *polycategory* as defined in [CS91]. However that paper also showed that no information will be gained by doing so. In fact, simply considering the inputs to be tensored together and the outputs parred together results in a morphism in a linearly distributive category; the associations do need to be made explicit to obtain a definite morphism.

We have also changed the sequentialization system; by adding the jump rule, more circuits are proof nets. However, the following lemma shows that the jump rule does not add equivalence classes of nets.

5.7. LEMMA. *Any proof net is equivalent to a net which is sequentializable without using the jump rule.*

Proof. It is not hard to show that it suffices to establish the following result: that if a net C sequentializes to a net D in one application of the jump rule, then C is equivalent to a net C', which sequentializes to D without using the jump rule. Suppose that the jump is a par jump (a jump downward). Then obtain C' from C by running a par reduction *backwards* between the jump wire and the along wire; that is, join the jump wire and along wire with a par-introduction link, followed immediately by a par-elimination link. Note that C' is sequentializable if C is. Now C can be obtained from C' by one application of a par reduction. It is clear how to emulate the effect of the jump on C' without using the jump rule. The case of a tensor jump uses a (backwards) tensor reduction similarly. ∎

Given these facts, the results of [BCST] will carry through, namely

5.8. THEOREM. *PNet($\mathcal{C}$) is the free non-symmetric linearly distributive category generated by the polygraph of components $\mathcal{C}$.*

Refer to [BCST] for the proof.

6. Coherence

We now come to the main result, which addresses the equivalence of morphisms in non-symmetric linearly distributive categories, and by extension proofs in non-commutative negation-free mLL. Theorem 5.8 above states that such a category, free on a polygraph of morphisms/components $\mathcal{C}$, is just PNet($\mathcal{C}$), planar proof nets under the circuit equivalence defined in the previous section. To approach the problem of coherence, it is first proved that normal forms of the circuit equivalence rewrite system can be achieved in an effective way.

6.1. THEOREM. *Each equivalence class of circuits in PNet($\mathcal{C}$) contains a unique expanded normal form up to $\mathcal{E}$-equivalence; one can reach the normal form of a circuit by applying rewrites, in any order, with a guaranteed bound on the number of reductions and irreducible expansions. Furthermore there is an procedure to execute the normalization in a bounded number of moves including equations.*

Proof. This proof follows the pattern of the coherence result for non-planar proof nets in [BCST]. Following Appendix A, for the first statement it suffices to show

- $\mathcal{R}$ is $\mathcal{E}$-terminating;
- $\mathcal{X} \cup \mathcal{R}$ is locally $\mathcal{E}$-confluent and $\mathcal{X}$-reducing; and
- $\mathcal{X}/\mathcal{R}$ is expansion $\mathcal{E}$-terminating.

• $\mathcal{R}$ **is** $\mathcal{E}$**-terminating.** This holds because every reduction decreases the number of links other than negation links in the circuit, whereas the equations change only the number of negation links.

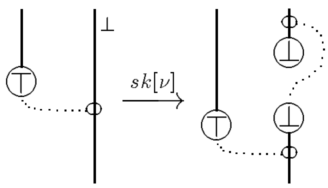
• $\mathcal{X} \cup \mathcal{R}$ **is locally** $\mathcal{E}$**-confluent and** $\mathcal{X}$**-reducing.** Consider the following notion:

6.2. Definition. *The* ***skeleton*** *of a net is the graph obtained from the net by: first, disconnecting all thinning links; and second, replacing each negation link chain with a single wire, by removing each negation link in turn from the thinning-link end of the now disconnected chain, and reversing the unit type of the (disconnected) thinning link with each such removal.*

Clearly any two $\mathcal{E}$-equivalent nets have the same skeleton. Reductions and expansions can be applied to the skeleton of a net; for any such move ν on the skeleton, define a transformation $sk[\nu]$ on the original net, which leads to a net with the reduced or expanded skeleton, but also moves the thinning links in an appropriate way. This means that any thinning links, which prevent a direct application of the skeletal move ν to the original net, need to be rewired out of the way first.

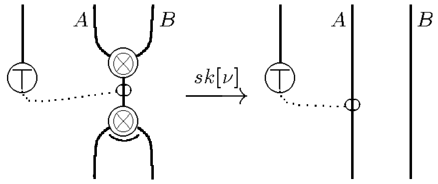
A skeletal expansion on the unit wire of a thinning link may lead to ambiguity, because the thinning link could correspond to a negation link chain in the original net, and the corresponding $sk[\nu]$ move could thus be an expansion on any of several different wires. However, it is easy to see that any expansion in a negation link chain can be followed by a sequence of reductions and equations that return the net to its state before the expansion. Finding a convergence for a divergence involving such an expansion is thus trivial. We can therefore safely neglect $sk[\nu]$ for these expansions.

Defining $sk[\nu]$ for other expansions is easy: just put all the thinning links on the expanded wire directly below the expansion, for instance:



(Note that there is some arbitrary choice in the way $sk[\nu]$ is defined; we could just as easily put all the thinning links above the expansion. It will not matter exactly how we do it as long as it is definite.)

For the tensor and cotensor reductions, simply rewire any thinning links on the reduced wire out of the way (whether up or down, they end up in the same place after the reduction):

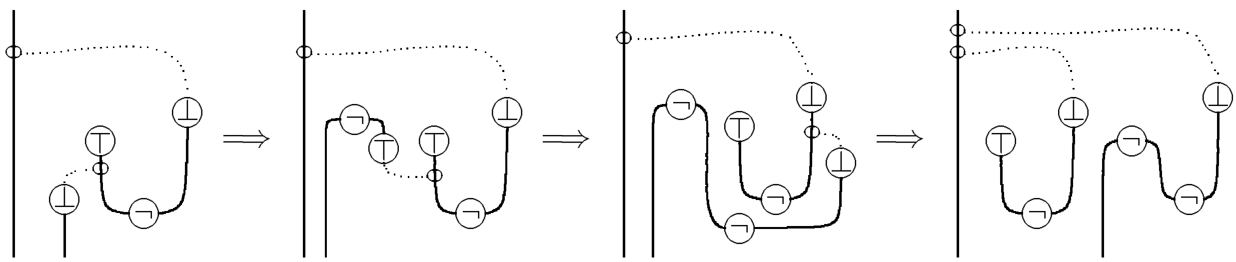


The remaining cases are the unit reductions. Here, a simple unit reduction on the skeleton may correspond to an complicated negation link chain reduction on the original net. All the thinning links which are attached to the chain must be pushed off of it. This needs to be done in a definite way so that we can show (shortly) that $sk[\nu]$ moves commute with rewirings. Define $sk[\nu]$ so that each thinning link pushed off "copies" the negation link structure of the chain to be reduced. Apply the following two steps to each link attached to the chain, starting with the one closest to the thinning-link end of the chain:

1. Ensure that the type of the unit of the attached thinning link matches the type of the wire to which it is attached; if a change is required, this means adding a negation link at the end of the attached thinning link.

2. Rewire the attached thinning link past the next negation link of the chain. This may mean adding another negation link to the attached link, in order to give it enough mobility. (If this is the case, the types will already match, as required by the first step, during the next stage.)

Repeat these steps until all thinning links are rewired off of the chain. An example:



Next it is necessary to show that, with appropriate rewirings, we can obtain the same result from a skeletal reduction or expansion even after rewiring on the original net. Thus for any transformation $sk[\nu]$ and any equation e, we define a series of equations $sk[\nu](e)$ such that the following diagram commutes:

$$\begin{array}{ccc} n_1 & \xrightarrow{sk[\nu]} & n_1' \\ {\scriptstyle e}\downarrow & & \downarrow{\scriptstyle sk[\nu](e)} \\ n_2 & \xrightarrow{sk[\nu]} & n_2' \end{array} \tag{15}$$

If e is a rewiring of a thinning link not affected by $sk[\nu]$ (that is, not on the reduced or expanded wire before or after e, and not across a link removed by a reduction), clearly $sk[\nu](e)$ is the same rewiring. This is also the case if e simply changes the negation link configuration of a wire attached to a tensor/cotensor reduction or an expansion, or if

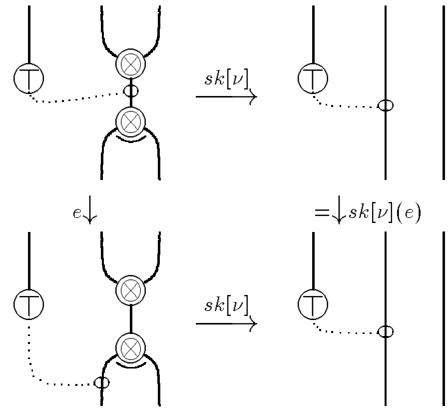


Figure 23: $sk[\nu](e)$ for tensor or cotensor reduction

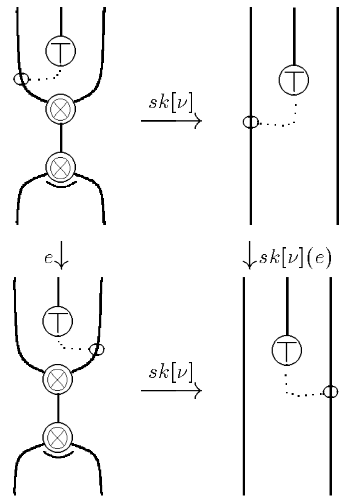


Figure 24: $sk[\nu](e)$ for tensor or cotensor reduction

e adds or removes a double negation along the negation link chain of a thinning link attached to a unit reduction.

If e moves a thinning link onto or off of the primary wire of a tensor or cotensor reduction, $sk[\nu](e)$ is the identity. See Figure 23.

If e moves a thinning link around the inside of the non-switching link in a tensor or cotensor reduction, $sk[\nu](e)$ is more complicated. We use the fact that we have a sequentializable circuit to assert that a sequent box can be formed between the adjacent wires of the switching link. Then $sk[\nu](e)$ is a "sequent box rewiring" (Proposition 5.4) to move the thinning link across this box. See Figure 24.

If e moves a thinning link onto or off of an expanded wire, $sk[\nu](e)$ is typically the same rewiring, but we may have to add rewirings (which clearly exist as "sequent box rewirings") to move the link past the now-expanded section. See Figure 25.

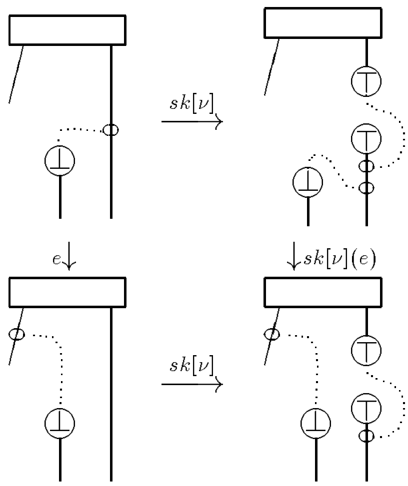


Figure 25: $sk[\nu](e)$ for expansions

We have now taken care of all cases except where ν is a unit reduction. Here there are two possibilities: either e applies to a thinning link which is attached to the negation link chain to be removed, or e applies to that negation link chain itself. Recall that $sk[\nu]$ copies the negation link configuration of the reduced chain onto each link pushed off of the chain. It is not difficult to see that, using this fact, we can just have $sk[\nu](e)$ insert whatever equations are necessary to emulate the negation link configuration for the circuit after e. The next few paragraphs spell out the various cases in detail.

Consider the case where e applies to a link which is attached to the structure to be reduced. If e pushes the link onto or off of the chain, then either $sk[\nu]$ would have the same result before or after e, or the type of unit would be changed by the addition of a negation link before being pushed off the chain. So $sk[\nu](e)$ is either the identity, or the addition or removal of a negation link at the affected thinning link.

Similarly, if e moves the link along the negation link chain in either direction, or if e adds or removes a negation link, then $sk[\nu]$ would either have the same result, or else the result will differ only by a double negation. So $sk[\nu](e)$ is either the identity, or the addition or removal of two opposing negation links at the appropriate point.

If e moves a link, which is attached to the "far end" of an wound barbell, around to the other side of the barbell, then the end result will be the same (a copy of the negation link structure will end up the same read along either side).

In the other case, if e applies to the thinning link which is to be removed by a unit reduction, then $sk[\nu](e)$ will be the same rewiring applied to each link pushed off of the reduced chain; in the case of a double negation along the negation link chain, $sk[\nu](e)$ is the same rewiring applied only to the links that will be pushed past that section. This works again because $sk[\nu]$ copies the negation link configuration of the reduced chain onto each of the attached links.

Now note that $sk[\nu](e)$ can be extended to apply to any series of equations e^*: define $sk[\nu](e_1; e_2)$ to be $sk[\nu](e_1); sk[\nu](e_2)$, and similarly for any series. These transformations can also be extended to apply to series of skeletal reductions and expansions: define $sk[\nu_1; \nu_2]$ to be $sk[\nu_1]; sk[\nu_2]$ and then $sk[\nu_1; \nu_2](e^*)$ to be $sk[\nu_2](sk[\nu_1](e^*))$. That these definitions "work" is easily seen by plugging together copies of the commutative diagram (15).

Finally, we use these transformations to obtain confluence. Consider any divergence

$$n_1' \xleftarrow{\nu_1} n_1 \stackrel{e^*}{=} n_2 \xrightarrow{\nu_2} n_2'.$$

Observe that the only cases where ν_1 and ν_2 cannot happen consecutively *in the skeleton* are when one is a reduction, and the other is an expansion of the wire to be reduced. But clearly all such expansions have the property that they can be followed by a sequence of reductions and equations which returns the net to its state before the expansion. Thus finding a convergence is trivial. As mentioned before, divergences involving a unit expansion along a negation link chain can be taken care of similarly.

For any other divergence construct the following diagram:

$$\begin{tikzcd}
 & n_1 \arrow[r, equal, "e^*"] \arrow[dl, "\nu_1"'] \arrow[dr, dotted] & n_2 \arrow[dl, dotted] \arrow[dr, "\nu_2"] & \\
n_1' \arrow[r, equal, "{sk[\nu_1](e^*)}"] \arrow[dr, dotted] & n_{12} \arrow[dr, "\nu_2"] & n_{21} \arrow[r, equal, "{sk[\nu_2](e^*)}"] \arrow[dl, "\nu_1"'] & n_2' \arrow[dl, dotted] \\
 & n_{21}' \arrow[r, equal, "{sk[\nu_1;\nu_2](e^*)}"'] & n_{12}' &
\end{tikzcd}$$

The dotted arrows represent $sk[\nu]$ moves. The solid arrows provide the required convergence. This is immediately $\mathcal{X}$-reducing.[7]

[7]There are two slight simplifications here. First, we are applying the commutative diagram (15) as if it applied in the case where one of the horizontal arrows is actually ν rather than $sk[\nu]$. Indeed, in most cases where ν is a valid expansion or reduction, $\nu = sk[\nu]$. However, our convention for $sk[\nu]$ with ν an expansion makes this equality false when ν is an expansion *below* a thinning link. In these cases, $sk[\nu]$ is just ν followed by an obvious sequence of equations, so it is easy to adjust the diagram by adding these

• **$\mathcal{X}/\mathcal{R}$ is expansion $\mathcal{E}$-terminating.** Call a wire in a circuit *unexpandable* if it is connected to the output port of an introduction link or the input port of an elimination link, and *expandable* otherwise. Clearly any expansion on an unexpandable wire will be reducible. For each circuit C let $v(C)$ be the total number of logical connectives ($\otimes$, $\oplus$, $\top$, $\perp$) in the types of all the expandable wires in C. Further let $v'(C)$ be $v(D)$, where D is the skeleton of a reduced $\mathcal{E}$-normal form of C (which exists by the preceding parts of the proof). Clearly $v'(C)$ is unchanged by reductions, equations, or reducible expansions; thus to prove the desired result it suffices to show that $v'(C)$ is strictly reduced by any irreducible expansion.

Let C' be the result of applying an irreducible expansion to C. We wish to show that this expansion could be "postponed" to D, in the sense that there is a circuit D', obtained from D by a single (skeletal) expansion, which is the skeleton of a (perhaps not fully) reduced form of C'. Observe that it is possible to set up a correspondence between the wires in the skeleton of C and the wires in the skeleton of its reduced $\mathcal{E}$-normal form, D, by fixing a reduction sequence, and considering each reduction to destroy the reduced wire and, in the case of the tensor and cotensor reductions, to join other wires which were previously distinct. Then any expansion in C on a wire which is eventually destroyed is reducible: the expansion doesn't prevent the application of any of the reductions in the sequence up to the one which would have destroyed its wire, but at that point a pair of reductions clearly has the same effect. So any irreducible expansion in C is on a wire which is preserved in D; it is now easy to see that that the circuit C' reduces to a circuit, whose skeleton is obtained by applying the corresponding expansion to D. Let D' be this skeletal circuit.

Clearly $v'(C') = v'(D') < v(D')$, so it suffices to show that $v(D') < v(D) = v'(C)$. Since D' is obtained from D by one expansion, it clearly is enough to show that $v(\text{reduct}) < v(\text{redex})$ for any expansion, which is indeed the case (note that the wires in the reduct, which have the same type as the expanded wire, have become unexpandable).

To see that the final statement of the theorem holds (i.e., that we can bound the number of equations needed to normalize a net), observe that a net is in expanded $\mathcal{E}$-normal form precisely when its skeleton is in expanded normal form. The preceding proof of confluence makes it clear that one can rewrite a net, so that its skeleton is in expanded normal form, without referring to arbitrary sequences of $\mathcal{E}$-moves. ∎

---

equations where appropriate.

(In spite of this, the diagram is correct in implying that ν_1 is applicable to n_{21} and ν_2 to n_{12}; this is because $sk[\nu]$ was defined so that thinning links will not be moved to a wire, where they prevent a reduction or expansion which could have happened simultaneously.)

The second simplification is the assumption that $\nu_1; sk[\nu_2]$ and $sk[nu_2]; \nu_1$ both lead from n_1 to the same circuit n'_{21}, and similarly with 1 and 2 reversed (these are the two large squares in the diagram). In most cases, when two skeletal moves can happen consecutively in either order, then the corresponding $sk[\nu]$ moves will have the same effect on thinning links in either order, as required for this assumption to be correct. Again, the convention for $sk[\nu]$ with expansions can make this invalid; but again, the resulting circuits will differ by an obvious sequence of equations, so the diagram can be adjusted by inserting these equations at the appropriate place.

Two nets are equivalent when they have $\mathcal{E}$-equivalent expanded normal forms; that is, when they reduce to nets with the same skeleton, and rewirings can be applied to make their thinning link arrangements the same. In order to provide a decision procedure for the equivalence of proof nets, it remains to provide a decision procedure for $\mathcal{E}$-equivalence.

As in [BCST] this is the point where we run into difficulty. Now, though, the problem is considerably more confined. The difficulty is that there can be an infinite number of distinct $\mathcal{E}$-equivalent circuits in expanded normal form, due to the ability to add negation links. This prevents an immediate decision procedure.

However, we can put any expanded normal form through a second level of reduction, in which the negation link rewirings are treated as one-directional reductions to remove "unnecessary" negation links. As noted in section 5.2, this leads to a net in which all the negation link chains are *tightly wound.*

If the $\mathcal{E}$-equivalence classes of only tightly wound configurations were finite, a decision procedure would follow easily.

In general, this is not the case. There exist infinite families of $\mathcal{E}$-equivalent circuits differing only in the winding of the thinning links; Figure 26 gives an example. Refer to Figure 28 for a breakdown of the rewiring used to prove equivalence in this example.

To illustrate the complexity of the situation, there are also easy examples of infinite families of *inequivalent* circuits differing only in the winding of the thinning links; an example is given in Figure 27.

Note that planar net skeletons which admit infinitely many winding configurations (equivalent or not) will all have a "floater": some subcircuit which is disconnected from the body of the net (that is, the inputs or outputs of the net) except via thinning links. Without such a floater there is simply nothing to wind about. If the unit wire of a

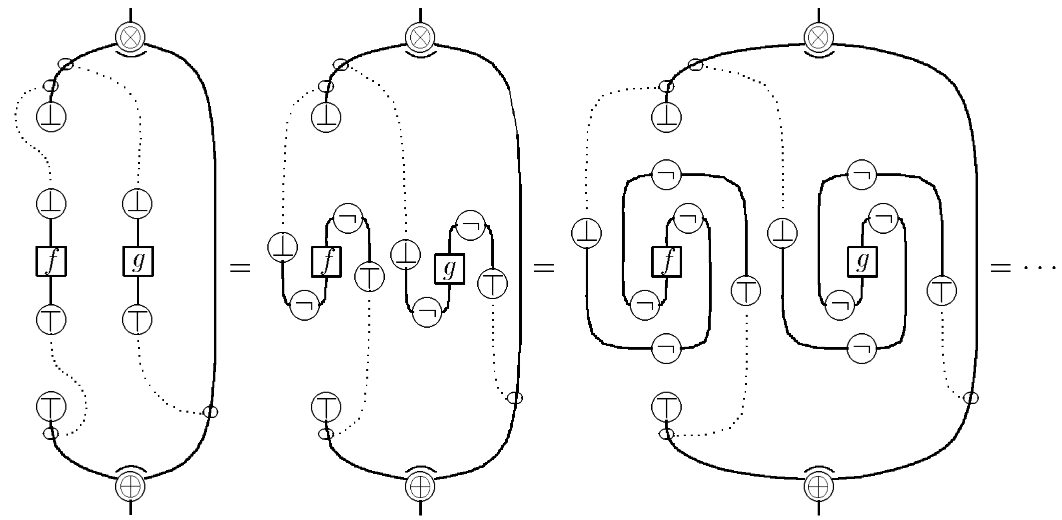


Figure 26: An infinite family of $\mathcal{E}$-equivalent nets

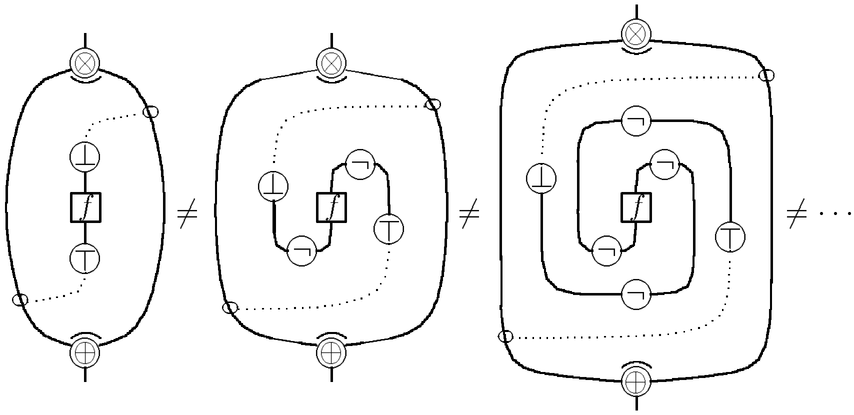


Figure 27: An infinite family of $\mathcal{E}$-inequivalent nets

thinning link is anchored (that is, if there is a path to the inputs or outputs without using thinning links), the geometry of the situation forbids an infinite number of tightly wound configurations from being possible.

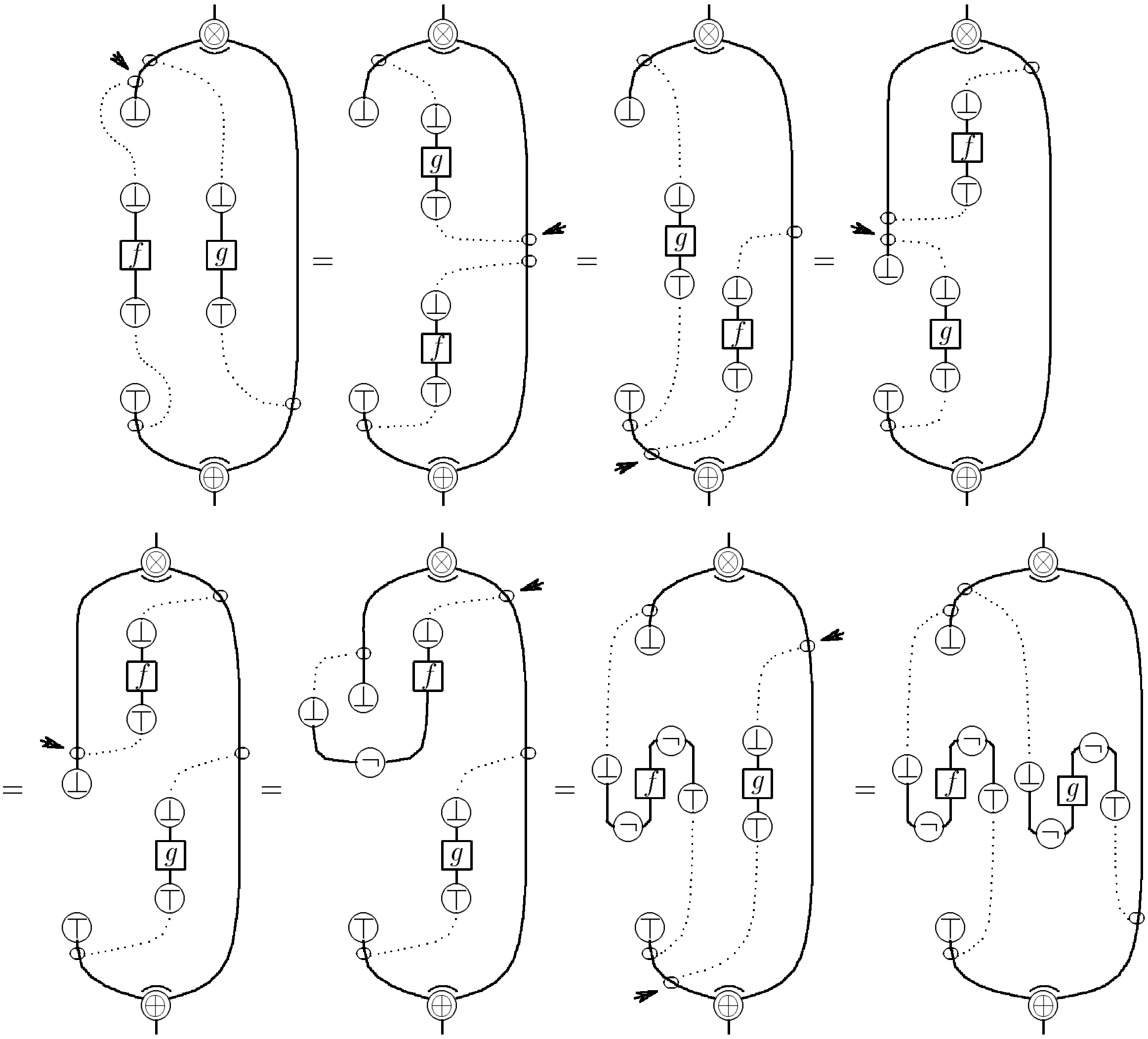
Furthermore, the presence of a floater in a normalized net depends on the existence of additional components with the units appearing in their inputs or outputs. Without such components any floater can only contain logical links. The following argument shows that any such floater must be reducible.

Suppose that the floater contains a bottom-introduction thinning link at its extremity. We ignore any thinning links attached to the floater itself, since we can use the skeleton moves. So if we follow the bottom wire down, the first link that we hit will be either a bottom-elimination, or a tensor-introduction or par-introduction. If we hit a bottom-elimination we have a barbell which can be reduced. If we hit a tensor-introduction or par-introduction, we continue to follow the wire downward; the types of the wires ensure that, eventually, an introduction link will be followed by the corresponding elimination link, implying a reduction. A similar argument holds for floaters containing only top-elimination thinning links.

These observations lead to the following result.

6.3. Corollary. *There is a decision procedure for the equivalence of morphisms in PNet($\mathcal{C}$), if $\mathcal{C}$ does not contain any components with units among its inputs or outputs.*

In particular, there is a decision procedure for the equivalence of proofs in non-commutative negation-free mLL without non-logical axioms. Further exploration of the phenomenon of winding will be required for an unrestricted decision procedure.



The arrows show which link is rewired to get to the next diagram.

Figure 28: $\mathcal{E}$-equivalent winding

A. Rewrite systems

The purpose of this section is to present the necessary background on the rewrite systems used in the paper, in particular *expansion/reduction systems modulo equations.* This is based on Appendix B of [BCST]. All results will be stated without proof.

A.1. Definition. *A **rewrite system** on a set N consists of a directed graph on N. The edges of the graph are the **rewrites**.*

In an expansion/reduction system modulo equations, one works with three such systems simultaneously: the **reductions** $\mathcal{R}$, the **expansions** $\mathcal{X}$, and the **equations** $\mathcal{E}$. The equations are considered to be usable in either orientation (that is, also include a system $\mathcal{E}^{\mathrm{op}}$ whose edges correspond to the edges of $\mathcal{E}$, but in the opposite direction). Denote the full system by $\mathcal{X}/\mathcal{R}$ *modulo* $\mathcal{E}$.

It is often important to consider which elements of N can be reached from each other via the equations:

A.2. Definition. *Two elements $n_1, n_2 \in N$ are **$\mathcal{E}$-equivalent** if there is a series of $\mathcal{E} \cup \mathcal{E}^{op}$ rewrites between them.*

The following are several important properties of this kind of rewrite system.

A.3. Definition. *Say that $\mathcal{R}$ is **$\mathcal{E}$-terminating at** $n \in N$ when there is a bound on the number of $\mathcal{R}$-moves in any $\mathcal{R} \cup \mathcal{E} \cup \mathcal{E}^{op}$ rewrite chain leaving n; also $\mathcal{R}$ is simply **$\mathcal{E}$-terminating** when it is $\mathcal{E}$-terminating for all $n \in N$.*

A.4. Definition. *Say that $\mathcal{R} \cup \mathcal{X}$ is **locally $\mathcal{E}$-confluent** if for any divergence of the form*

$$n_1 \xleftarrow{\nu_1} n \overset{e^*}{=} n' \xrightarrow{\nu_2} n_2$$

where ν_1, ν_2 are single *rewrites in $\mathcal{R} \cup \mathcal{X}$, and e^* is a series of moves in $\mathcal{E} \cup \mathcal{E}^{op}$, then there exists a convergence*

$$n_1 \xrightarrow{\nu_2'^*} n'' \xleftarrow{\nu_1'^*} n_2$$

where $\nu_1'^*$, $\nu_2'^*$ are series of moves in the full system $\mathcal{R} \cup \mathcal{X} \cup \mathcal{E} \cup \mathcal{E}^{op}$.

*Further say that $\mathcal{R} \cup \mathcal{X}$ is **$\mathcal{X}$-reducing** when there exists such a convergence in which $\nu_1'^*$ contains at most one expansion only if ν_1 is an expansion (and none otherwise), and $\nu_2'^*$ contains at most one expansion only if ν_2 is an expansion (and none otherwise).*

A.5. Definition. *An **expansion/reduction system modulo equations** $\mathcal{X}/\mathcal{R}$ modulo $\mathcal{E}$ is a collection of rewrite systems $(\mathcal{X}, \mathcal{R}, \mathcal{E}, \mathcal{E}^{op})$ in which*

- *$\mathcal{R}$ is $\mathcal{E}$-terminating;*
- *$\mathcal{R} \cup \mathcal{X}$ is locally $\mathcal{E}$-confluent and $\mathcal{X}$-reducing.*[8]

The following is to build up to the definition of *expansion $\mathcal{E}$-terminating.*

[8]In [BCST] a different definition is given in terms of non-local confluence, then this characterization is given as Theorem B.6.

A.6. Definition. *An element $n \in N$ is in* ***reduced $\mathcal{E}$-normal form*** *when there are no reductions leading from n or any element $\mathcal{E}$-equivalent to n.*

A.7. Proposition. *In an expansion/reduction system modulo equations, there is a unique reduced $\mathcal{E}$-normal form (up to $\mathcal{E}$-equivalence) in each $(\mathcal{R} \cup \mathcal{E} \cup \mathcal{E}^{op})$-equivalence class on N; call this* ***the reduced $\mathcal{E}$-normal form*** *of any element of the equivalence class. Furthermore one can reach the normal form from any element of the equivalence class by applying $(\mathcal{R} \cup \mathcal{E} \cup \mathcal{E}^{op})$-rewrites in any order, and one is guaranteed a bound on the number of reductions.*

A.8. Definition. *An expansion $n_1 \xrightarrow{x} n_2$ is said to be* ***reducible*** *if there are series ν_1^*, ν_2^* in $\mathcal{R} \cup \mathcal{E} \cup \mathcal{E}^{op}$ such that $n_1 \xrightarrow{x} n_2 \xrightarrow{\nu_1^*} n_3$ and $n_1 \xrightarrow{\nu_2^*} n_3$; it is* ***irreducible*** *otherwise.*

In an expansion/reduction system modulo equations, it follows that $n_1 \xrightarrow{x} n_2$ is reducible if the reduced $\mathcal{E}$-normal form of n_1 is $\mathcal{E}$-equivalent to the reduced $\mathcal{E}$-normal form of n_2, and irreducible otherwise.

A.9. Definition. *Say that $\mathcal{X}/\mathcal{R}$ is* ***expansion $\mathcal{E}$-terminating at*** *$n \in N$ when there is a bound on the number of irreducible expansions in any rewrite chain (of the full system) leaving n; also $\mathcal{X}/\mathcal{R}$ is* ***expansion $\mathcal{E}$-terminating*** *if it is expansion $\mathcal{E}$-terminating at each $n \in N$.*

The following is the most important property of these systems for the results in this paper.

A.10. Definition. *An element $n \in N$ is in* ***expanded $\mathcal{E}$-normal form*** *when there are no reductions, and only reducible expansions, leading from n or any element $\mathcal{E}$-equivalent to n.*

A.11. Proposition. *In an expansion $\mathcal{E}$-terminating expansion/reduction system modulo equations, there is a unique expanded $\mathcal{E}$-normal form (up to $\mathcal{E}$-equivalence) in each $(\mathcal{R} \cup \mathcal{X} \cup \mathcal{E} \cup \mathcal{E}^{op})$-equivalence class on N; we then call this* ***the expanded $\mathcal{E}$-normal form*** *of any element of the equivalence class. Furthermore one can reach the normal form from any element of the equivalence class by applying $(\mathcal{R} \cup \mathcal{X} \cup \mathcal{E} \cup \mathcal{E}^{op})$-rewrites in any order, and one is guaranteed a a bound on the number of reductions and irreducible expansions.*

Thus, in a system for which

- $\mathcal{R}$ is $\mathcal{E}$-terminating;
- $\mathcal{X} \cup \mathcal{R}$ is locally confluent and $\mathcal{X}$-reducing; and
- $\mathcal{X}/\mathcal{R}$ is expansion $\mathcal{E}$-terminating;

every $n \in N$ has a unique expanded $\mathcal{E}$-normal form up to $\mathcal{E}$-equivalence. Furthermore one can reach this normal form from n by applying rewrites, in any order, with a guaranteed bound on the number of reductions and irreducible expansions. Note that there is not in general a guaranteed bound on the number of equations, so further work may be required to establish that normal forms can be reached in an effective way.

References

[A91] V.M. Abrusci, *Phase Semantics and Sequent Calculus for Pure Noncommutative Classical Linear Propositional Logic.* Journal of Symbolic Logic **56** (1991) 1403–1451.

[A95] V.M. Abrusci, *Noncommutative Proof Nets.* In: J.-Y. Girard, Y. Lafont, L. Regnier, eds., *Advances in Linear Logic.* London Mathematical Society Lecture Note Series **222** (1995) 271–296.

[BCST] R. Blute, J.R.B. Cockett, R.A.G. Seely, and T. Trimble, *Natural deduction and coherence for weakly distributive categories.* Journal of Pure and Applied Algebra **113** (1996) 229–296.

[CS91] J.R.B. Cockett and R.A.G. Seely, *Weakly distributive categories.* In: M.P. Fourman, P.T. Johnstone, A.M. Pitts, eds., *Applications of Categories to Computer Science.* London Mathematical Society Lecture Note Series **177** (1992) 45–65. Expanded version in: Journal of Pure and Applied Algebra **144** (1997) 133–173.

[CKS] J.R.B. Cockett, J. Koslowski, and R.A.G. Seely, *Introduction to linear bicategories.* To appear in Mathematical Structures in Computer Science. Preprint (1999).

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

[G87] J.-Y. Girard, *Linear logic.* Theoretical Computer Science **50** (1987) 1–102.

Group in Logic and the Methodology of Science
731 Evans Hall #3840
University of California Berkeley
Berkeley CA 94720-3840
USA
Email: `schneck@math.berkeley.edu`

This article may be accessed via WWW at `http://www.tac.mta.ca/tac/` or by anonymous ftp at `ftp://ftp.tac.mta.ca/pub/tac/html/volumes/6/n9/n9.{dvi,ps}`