Journal of Pure and Applied Algebra 216 (2012) 2386–2410

Contents lists available at SciVerse ScienceDirect

Journal of Pure and Applied Algebra

journal homepage: www.elsevier.com/locate/jpaa

Simple free star-autonomous categories and full coherence

Dominic J.D. Hughes

Stanford University, United States

ARTICLE INFO

Article history:
Received 7 July 2005
Received in revised form 6 February 2012
Available online 21 April 2012
Communicated by G. Rosolini

MSC: 18D10; 18D15; 03F52; 03B47

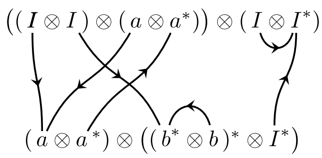
ABSTRACT

This paper gives a simple presentation of the free star-autonomous category over a category, based on Eilenberg–Kelly–MacLane graphs and Trimble rewiring, yielding a full coherence theorem: the commutativity of diagrams of canonical maps is decidable.

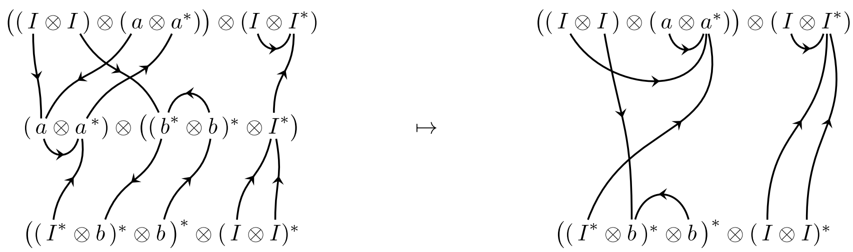
1. Introduction

Eilenberg–Kelly–MacLane graphs [10,24] elegantly describe certain morphisms of closed categories. This paper shows that little more is needed to present the free star-autonomous category [2] generated by a category, for a full coherence theorem: the commutativity of diagrams of canonical maps is decidable.

Given a set $\mathbb{A} = \{a, b, \ldots\}$ of generators, we define the category of $\mathbb{A}$-*linkings*: objects are star-autonomous *shapes* (expressions) over $\mathbb{A}$, such as $S = \big(a \otimes (b^* \otimes a^{**})\big)^{***} \otimes I^*$ (with I the unit), and a morphism $S \to T$ is a *linking*, a function from negative leaves to positive leaves, e.g.



is a morphism from the upper shape to the lower shape. Composition is simply path composition:



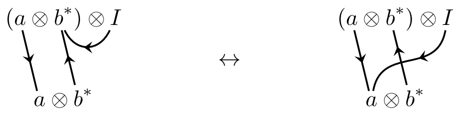
A leaf function qualifies as a linking only if it satisfies the standard criterion for multiplicative proof nets[1] [9,14], so simple as to be checkable in linear time [16,18]. Employing Trimble rewiring [33,3], we define two linkings as *similar* if they differ

E-mail address: dominic@theory.stanford.edu.

[1] This paper will not assume any familiarity with proof nets or linear logic.

0022-4049/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.jpaa.2012.03.020

by an edge from an I, e.g.



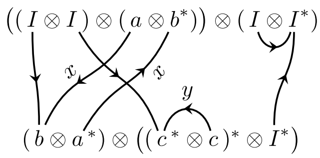
and define an $\mathbb{A}$-*net* as an $\mathbb{A}$-linking modulo similarity. The category of $\mathbb{A}$-nets is the free star-autonomous category generated by $\mathbb{A}$. To emphasise the simplicity:

(1) a morphism is a leaf function satisfying a standard criterion (checkable in linear time);
(2) composition is standard path composition;
(3) modulo a standard equivalence (Trimble rewiring).

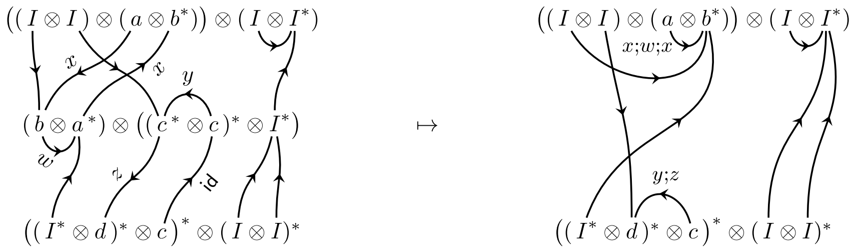
The key novelty is (2), the fact that composition is simply path composition. This preserves an elegant feature of Eilenberg–Kelly–MacLane graphs. In contrast, the composition in previous presentations of free star-autonomous categories [3,25,27] is more complex. (We return to this related work later in the Introduction.)

Abstractly, the underlying path composition can be understood as a forgetful functor from the category of $\mathbb{A}$-linkings (sketched above) to **Int**(Setp), the compact closed category obtained by applying the **Int** or geometry-of-interaction construction [12,21,1] to the traced monoidal category Setp of sets and partial functions (with coproduct as tensor). This ties in nicely with Eilenberg–Kelly–MacLane graphs and Kelly–Laplaza graphs [23] for compact closed categories, since each has a forgetful functor to **Int**(Setp).

Arbitrary base category. When $\mathbb{A}$ is not discrete, we simply label each edge between generators with a morphism of $\mathbb{A}$. For example, if $x : a \to b$ and $y : c \to c$ in $\mathbb{A}$, then



is an $\mathbb{A}$-linking. Composition collects labels along a path, and composes[2] them in $\mathbb{A}$, e.g.



The category of $\mathbb{A}$-nets ($\mathbb{A}$-linkings modulo Trimble rewiring, as before) is the free star-autonomous category generated by the category $\mathbb{A}$.

Full coherence. Equivalence modulo rewiring is decidable, by finiteness. Thus we have a full coherence theorem: we can decide the commutativity of diagrams of canonical maps in star-autonomous categories. Here are two short illustrative examples.

Example 1 (*Identity* $\neq$ *Twist on* $\perp = I^*$)**.** Let $\mathsf{tw}_{A\otimes B} : A \otimes B \to B \otimes A$ be the canonical twist (symmetry) isomorphism. The identity and twist $\mathsf{id}_{\perp\otimes\perp}, \mathsf{tw}_{\perp\otimes\perp} : \perp \otimes \perp \to \perp \otimes \perp$ determine respective linkings i and t:

They differ on two inputs, and there are no other [3] linkings $\perp \otimes \perp \to \perp \otimes \perp$, so i and t cannot be rewired into each other. Thus, in general, $\mathsf{id}_{\perp\otimes\perp} \neq \mathsf{tw}_{\perp\otimes\perp} : \perp \otimes \perp \to \perp \otimes \perp$.

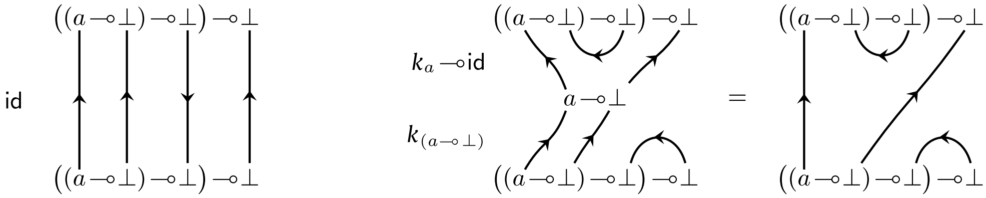
2 Throughout this paper we employ sequential notation $f ; g$ for the composite of $f : S \to T$ and $g : T \to U$ rather than functional notation gf, since it is more natural in a diagrammatic, path-following setting.

3 The other two functions from negative leaves to positive leaves fail the proof net criterion.

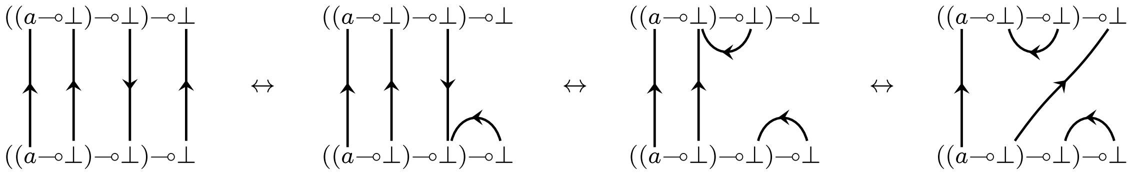
Example 2 (*Triple-dual Problem*)**.** We show that the following diagram commutes (a triple-dual problem [24, Section 1, diagram (1.4)])

$$\begin{tikzcd} ((a \multimap \perp) \multimap \perp) \multimap \perp \arrow[r, "k_a \multimap \mathsf{id}"] \arrow[dr, "\mathsf{id}"'] & a \multimap \perp \arrow[d, "k_{(a \multimap \perp)}"] \\ & ((a \multimap \perp) \multimap \perp) \multimap \perp \end{tikzcd} \tag{1}$$

where $A \multimap B = (A \otimes B^*)^*$ and $k_A : A \rightarrow (A \multimap \perp) \multimap \perp$ is the canonical map of its type. Each path in the triangle determines a corresponding linking



equivalent via three rewirings:



Thus we conclude that triangle (1) commutes in every star-autonomous category.[4]

Related work. This paper follows an approach which can be traced back to Todd Trimble's Ph.D. thesis [33].[5] We call this the *rewiring approach*:

(a) represent a morphism by a structure involving attachments ('wiring') of negative units[6];
(b) quotient by rewiring: identify correct structures which differ by just one such attachment.

This is the fourth paper to use the rewiring approach to construct free star-autonomous categories. The chronological sequence is detailed below, and is summarised in Table 1.

In [3] structures are circuit diagrams (in tensor calculus style [19]), attachments are (dotted) edges from negative units, called thinning links, correctness is the standard multiplicative proof net criterion,[7] and rewiring between correct structures is expressed in rules of surgery on circuits, which (by the empire rewiring Proposition 3.3 of [3]) permit an arbitrary re-targeting of a thinning link between correct circuits.[8] Equivalence classes yield the free linearly distributive category and free star-autonomous category (linearly distributive category with negation) generated by a polygraph (e.g., by a category), for full coherence.

In [25] structures are $\lambda\mu$-style terms [31] with explicit substitution $\{-/-\}$, for example $(\lambda\beta^{A\multimap\perp}.\langle\beta\rangle z^A)\{\mu\alpha^{A\multimap\perp}.\langle\gamma\rangle\alpha/z\}$, attachments are unit let constructs $\langle x/*\rangle(-)$, correctness is inductive (typability, i.e., sequentialisability) and rewiring is by an instance of the π-congruence rule, $\Gamma \vdash C[\langle x/*\rangle t] \sim \langle x/*\rangle C[t] : A$. Equivalence (congruence) classes yield an internal language for autonomous and star-autonomous categories, for full coherence.

In [27] the structure is a syntactic[9] proof net, a formula of multiplicative linear logic equipped with a leaf permutation, e.g. $\perp_1 \otimes ((\alpha_2 \otimes a_4^*) \parr (\perp_3 \otimes I_5)) \triangleright \perp \parr a, (\perp \otimes a^*) \parr I$ is a structure representing a morphism $I \otimes a^* \rightarrow (\perp \otimes a^*) \parr I$, the formula constructor $(-) \otimes \perp$ attaches negative units, correctness is again the standard multiplicative proof net criterion, and rewiring is by the invertible linear distributivity rewrite $Q \parr (R \otimes \perp) \leftrightarrow (Q \parr R) \otimes \perp$. Equivalence classes yield the free star-autonomous category with strict double involution[10] generated by a set, for full coherence.

4 Compare with similar arguments in [3, Section 4.2], [25, Section 2] and [30, Section 10]. A key advantage here is that, because of the simple path composition, the composite $k_a \multimap \mathsf{id}\ ;\ k_{(a\multimap\perp)}$ is immediate on inspection.

5 Copies of Trimble's thesis [33] are not particularly easy to come by. See [3, Section 1, Section 3.2] and [7, Section 3] for overviews of some of the content.

6 For history and development of the attachment of negative units in linear logic, see [8,32,15,13,14].

7 Sequentialisability/contractibility [8] is used to deal with the planar case; see Section 2.7 of [3].

8 Decomposing rewiring into shorter steps aided the freeness proofs in [3].

9 Axiom links $a \otimes a^\perp$ and attachments of negative units $(-) \otimes \perp$ are syntactic, enveloped in a formula sharing the leaves of the sequent. In conventional proof nets [14], axiom links and unit attachments ($\perp$-jumps) are edges.

10 The canonical map $A^{**} \rightarrow A$ is the identity. Up to equivalence this is a free star-autonomous category, in a strict sense [6].

Table 1
This paper is the fourth to construct free star-autonomous categories using the *'rewiring approach'*, which can be traced back to Trimble's thesis [33]: (a) represent a morphism by a structure involving attachments ('wiring') of negative units, (b) quotient by rewiring, that is, identify correct structures which differ by just one such attachment. In each case a morphism of the free category is a finite equivalence class, hence equality of morphisms is decidable.

	Structure	Attachment of negative unit	Correctness/allowability	Rewiring of attachments, between correct structures	Problem with composition/normalisation
[3]	Circuit diagram (proof net in tensor calculus style)	Thinning link (dotted edge)	Standard proof net criterion (sequentialisability/contractibility for planar case)	Surgery rules/re-target thinning link (by [3, Prop. 3.3])	Attachments can block cut redexes
[25]	$\lambda\mu$-style term, e.g. $(\lambda\beta^{A\multimap\perp}.\langle\beta\rangle z^A)$ $\{\mu\alpha^{A\multimap\perp}.\langle\gamma\rangle\alpha/z\}$	Unit let term constructor $\langle x/*\rangle(-)$	Typability (i.e., sequentialisability)	π-congruence rule, $\Gamma\vdash C[\langle x/*\rangle t]\sim\langle x/*\rangle C[t]:A$	Normalisation confluent only modulo rewiring/congruence
[27]	MLL formula sharing sequent leaves, with permutation, e.g. $\perp_1\otimes((\alpha_2\otimes a_4^*)\,⅋(\perp_3\otimes I_5))$ $\rhd\perp⅋\, a,\,(\perp\otimes a^*)\,⅋\, I$	Formula constructor $(-)\otimes\perp$	Standard proof net criterion	Formula rewrite $Q\,⅋(R\otimes\perp)\leftrightarrow(Q\,⅋\,R)\otimes\perp$	Attachments can block cut redexes
This paper	Leaf function	Edge from I	Standard proof net criterion	Re-target edge from I	

At first sight, it may seem repetitive and uninteresting to employ the rewiring approach for star-autonomous categories a fourth time. However, the simplicity of the end product relative to the previous approaches seems to justify the repetition. As we remarked earlier, we preserve an elegant feature of Eilenberg–Kelly–MacLane graphs:

- *Composition is simply path composition.*

Composition is more complex in the three previous approaches to free star-autonomous categories [3,25,27]. In each case, given normal forms s and t representing equivalence classes, one first forms a 'concatenation' $s;t$ (in [3], pasting the circuits at a cut wire, in [25] forming an explicit substitution, and in [27] forming a proof net with cuts), then normalises $s;t$ in a rewrite system. In [3,27] normalisation is defined only modulo equivalence, since unit attachments (thinning links in [3] and $(-)\otimes\perp$ in [27]) can block cut redexes,[11] and in [25] confluence is only modulo equivalence (congruence). In contrast, path composition, as in this paper and Eilenberg–Kelly–MacLane graphs, is simple and direct.

This paper is the sequel to [18][12] on multiplicative proof nets with units, which relate closely to $\mathbb{A}$-linkings, for $\mathbb{A}$ discrete. For comprehensive background and history on free star-autonomous categories and coherence, see the introductions of [3,25,27].

Potential future work. Perhaps the most direct redeployment of Trimble rewiring is [30], since it is for SMCCs (symmetric monoidal closed categories), the original case treated in [33].[13] Hybridising the present paper on star-autonomous categories with the extension of Lamarche's essential nets [26] in [30] might yield a simple presentation of free SMCCs: objects are shapes generated by $\otimes$, $\multimap$ and I (e.g. $(a\otimes I)\multimap\perp$), a morphism is a leaf function satisfying Lamarche's criterion, modulo Trimble rewiring, and composition is simply path composition.

Path composition would constitute a *direct* composition of generating functions, bypassing the complexities of strategies (O-orientation, shortsightedness, non-determinism, conditional exhaustion, etc.) for a more economical description of the free SMCC. Furthermore, the approach would extend immediately to the free SMCC generated by an arbitrary category $\mathbb{A}$, not just a set $\mathbb{A}$ (as in [30]), by labelling edges with morphisms of $\mathbb{A}$ (as in the present paper for star-autonomous categories). In summary, this path composition approach would abstract away from the intricacies of the strategies, extracting the essence: a geometry of interaction of generating functions in **Int**(Setp).

2. Split star-autonomous categories

The category of $\mathbb{A}$-linkings (sketched above, and defined in the next section) is almost, but not quite, star-autonomous. It becomes star-autonomous upon quotienting by Trimble rewiring. Below we axiomatise its raw structure, prior to

11 A problem also discussed by Girard, in the context of proof nets [14, Section A.2].

12 It was tempting to merge the two papers. However, some proof theorists and linear logicians may not be interested in star-autonomous categories and the emphasis on coherence over correctness criteria (combinatorial characterisations of *allowability*, in the parlance of [24]), and conversely, some categorists may not be interested in linear logic and its sequent calculus, and the emphasis on correctness criteria over coherence. The present paper targets categorists, and assumes no familiarity with linear logic.

13 The definition of $\sim$ in Section 9 of [30] is precisely Trimble rewiring: two (correct) structures (strategies/linkages/ generating functions) are identified if they differ by the attachment of just one negative unit (there called a joker move). Syntactically (see the discussion before Proposition 45 of [30]), Trimble rewiring here corresponds to that in [25]: π-congruence with the unit let construct, $\Gamma\vdash C[\langle x/*\rangle t]\sim\langle x/*\rangle C[t]:A$.

quotienting, as a *split star-autonomous category*, defined by relaxing the unit isomorphisms $A \to I \otimes A$ and $A \to A \otimes I$ of a star-autonomous category to be split monomorphisms (sections).

A ***star-autonomous category*** [2] is a category $\mathbb{C}$ equipped with the following structure:

(1) *Tensor*. A functor $- \otimes - : \mathbb{C} \times \mathbb{C} \to \mathbb{C}$.

(2) *Associativity*. A natural isomorphism $\alpha_{A,B,C} : (A \otimes B) \otimes C \to A \otimes (B \otimes C)$, natural in objects $A, B, C \in \mathbb{C}$, such that the following pentagon commutes:

$$\begin{tikzcd}
\big((A\otimes B)\otimes C\big)\otimes D \arrow[r, "\alpha"] \arrow[dr, "\alpha\otimes\mathsf{id}"'] & (A\otimes B)\otimes(C\otimes D) \arrow[r, "\alpha"] & A\otimes\big(B\otimes(C\otimes D)\big) \\
& \big(A\otimes(B\otimes C)\big)\otimes D \arrow[r, "\alpha"'] & A\otimes\big((B\otimes C)\otimes D\big) \arrow[u, "\mathsf{id}\otimes\alpha"']
\end{tikzcd}$$

(3) *Unit*. An object $I \in \mathbb{C}$.

(4) *Unit isomorphisms*. Natural isomorphisms[14]

$$\begin{aligned} \lambda_A : A &\to I \otimes A \\ \rho_A : A &\to A \otimes I \end{aligned}$$

natural in the object $A \in \mathbb{C}$, such that the following triangle commutes:

$$\begin{tikzcd}
& A\otimes B \arrow[dl, "\rho\otimes\mathsf{id}"'] \arrow[dr, "\mathsf{id}\otimes\lambda"] & \\
(A\otimes I)\otimes B \arrow[rr, "\alpha"'] & & A\otimes(I\otimes B)
\end{tikzcd}$$

(5) *Symmetry*. A natural isomorphism $\sigma_{A,B} : A \otimes B \to B \otimes A$, natural in objects $A, B \in \mathbb{C}$, such that the following diagrams commute:

$$\begin{tikzcd}
A\otimes B \arrow[rr, "\mathsf{id}"] \arrow[dr, "\sigma"'] & & A\otimes B \\
& B\otimes A \arrow[ur, "\sigma"'] &
\end{tikzcd}
\qquad
\begin{tikzcd}
(A\otimes B)\otimes C \arrow[r, "\alpha"] \arrow[d, "\sigma\otimes\mathsf{id}"'] & A\otimes(B\otimes C) \arrow[r, "\sigma"] & (B\otimes C)\otimes A \arrow[d, "\alpha"] \\
(B\otimes A)\otimes C \arrow[r, "\alpha"'] & B\otimes(A\otimes C) \arrow[r, "\mathsf{id}\otimes\sigma"'] & B\otimes(C\otimes A)
\end{tikzcd}$$

(6) *Involution*. A full and faithful functor $(-)^* : \mathbb{C}^{\mathsf{op}} \to \mathbb{C}$.

(7) *Closure*. A natural isomorphism $\mathbb{C}(A \otimes B, C^*) \to \mathbb{C}(A, (B \otimes C)^*)$, natural in objects $A, B, C \in \mathbb{C}$.

Axioms (1)–(4) define a monoidal category and (1)–(5) define a symmetric monoidal category [28].[15] The above is not the original definition of star-autonomous category, but (modulo our slightly different presentation of symmetric monoidal category) is equivalent [2].

Define a ***split star-autonomous category*** by relaxing (4): demand only that the natural transformations $\lambda_A : A \to I \otimes A$ and $\rho_A : A \to A \otimes I$ be split monomorphisms (sections), rather than isomorphisms. Thus we require for each A the existence[16] of retractions $\overline{\lambda}_A : I \otimes A \to A$ and $\overline{\rho}_A : A \otimes I \to A$ such that $\lambda_A; \overline{\lambda}_A = \mathsf{id}_A$ and $\rho_A; \overline{\rho}_A = \mathsf{id}_A$.

$$\begin{tikzcd}
A \arrow[rr, "\mathsf{id}"] \arrow[dr, "\lambda"'] & & A \\
& I\otimes A \arrow[ur, "\overline{\lambda}"'] &
\end{tikzcd}
\qquad
\begin{tikzcd}
A \arrow[rr, "\mathsf{id}"] \arrow[dr, "\rho"'] & & A \\
& A\otimes I \arrow[ur, "\overline{\rho}"'] &
\end{tikzcd}
\qquad \text{(required)}$$

We drop the requirement that $\overline{\lambda}_A; \lambda_A = \mathsf{id}_{I\otimes A}$ and $\overline{\rho}_A; \rho_A = \mathsf{id}_{A\otimes I}$.

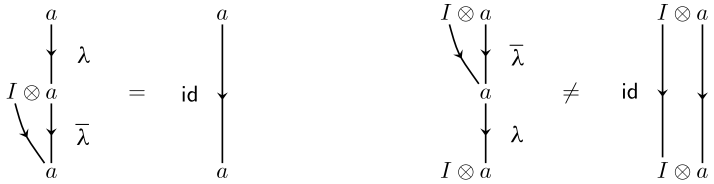
14 Conventionally the unit isomorphisms are typed $A \otimes I \to A$ and $I \otimes A \to A$ [28]. We reverse them to ease the definition of split star-autonomous category below.

15 In [28] Mac Lane demands $\rho_I = \lambda_I$ for a monoidal category and $\lambda_A; \sigma_{I,A} = \rho_A$ for a symmetric monoidal category, which are superfluous [22,20].

16 An anonymous referee noted that if we include a specific choice of retractions for each A in the definition of split star-autonomous category, the definition becomes monadic over the category of categories.

$$\begin{tikzcd} I \otimes A \arrow[rr, "\mathsf{id}"] \arrow[dr, "\overline{\lambda}"'] & & I \otimes A \\ & A \arrow[ur, "\lambda"'] & \end{tikzcd} \qquad \begin{tikzcd} A \otimes I \arrow[rr, "\mathsf{id}"] \arrow[dr, "\overline{\rho}"'] & & A \otimes I \\ & A \arrow[ur, "\rho"'] & \end{tikzcd} \qquad \text{(dropped)}$$

Although $\mathbb{A}$-linkings have yet to be defined formally, the following example should nonetheless help to motivate the definition.



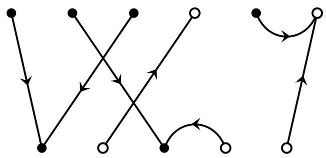
The category of $\mathbb{A}$-linkings is split star-autonomous, but not star-autonomous.[17] It becomes star-autonomous upon quotienting by Trimble rewiring.

3. Linkings

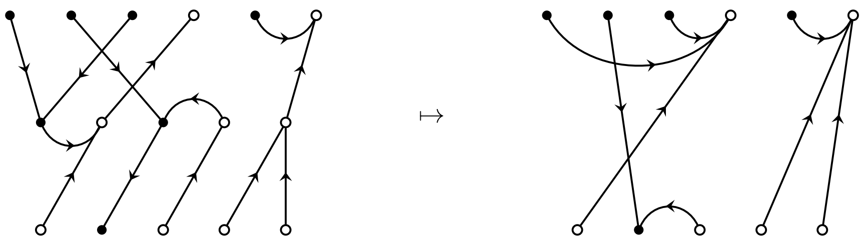
This section presents the split star-autonomous category $\mathsf{L}\mathbb{A}$ of $\mathbb{A}$-linkings over a category $\mathbb{A}$. Each linking is two-sided, being a morphism $S \to T$ between two star-autonomous shapes S and T, analogous to the original Eilenberg–Kelly–MacLane graphs [10,24]. Section 5 introduces one-sided linkings, more analogous to the graphs in [23] for compact closed categories. Auxiliary one-sided linkings will facilitate later proofs.

3.1. The category $\mathscr{L}$ of partial leaf functions between signed sets

Define the category $\mathscr{L}$ as follows. An object is a ***signed set*** X, whose elements we shall call ***leaves***, each signed either ***positive*** or ***negative***. (Thus a signed set is a set X equipped with a function $X \to \{+, -\}$.) A morphism $X \to Y$ is a ***partial leaf function***: a partial function from $X^+ + Y^-$ to $X^- + Y^+$, where $+$ is disjoint union and $(-)^+$ (resp. $(-)^-$) is the operation which restricts a signed set to its positive (resp. negative) leaves. For example,



is a (total) morphism from the upper signed set, 4 positive • and 2 negative ◦ leaves, to the lower one, 2 positive and 3 negative leaves. Composition is simply finite (directed) path composition:



Formally, given partial leaf functions $f : X \to Y$ and $g : Y \to Z$ define $f;g : X \to Z$ by $(f;g)(l) = l'$ iff there is a finite directed path from l to l' in the union of f and g, viewed as a directed graph on $X + Y + Z$. The following proposition guarantees that $f;g$ is well-defined (single-valued).

Proposition 1 (*Unique Path Property*)**.** *If $f;g$ is a composite partial leaf function containing the edge $\langle l, l'\rangle$ (i.e., $(f;g)(l) = l'$), then a unique path $ll_0 \dots l_n l'$ gave rise to $\langle l, l'\rangle$ during composition.*

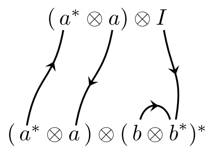
Proof. Suppose $ll'_0 \dots l'_m l'$ were an alternative path. Let i be minimal with $l_i \neq l'_i$. Then $f(l_{i-1}) = l_i$ and $f(l_{i-1}) = l'_i$, or $g(l_{i-1}) = l_i$ and $g(l_{i-1}) = l'_i$, contradicting (partial) functionality. □

[17] The category of $\mathbb{A}$-linkings is also *semi star-autonomous* in the sense of [17].

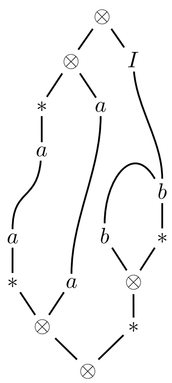
The category $\mathcal{L}$ is **Int**(Setp), the result of applying the feedback construction **Int** [21] (or geometry of interaction construction [1,12]) to the traced monoidal category Setp of sets and partial functions, with tensor as coproduct. Thus $\mathcal{L}$ is compact closed, with tensor as disjoint union and the dual of a signed set obtained by reversing signs. The subcategory of $\mathcal{L}$ whose objects are finite and morphisms are bijections is the category of involutions defined in [23, Section 3], a modified presentation of the fixed-point free involutions of [10,24].

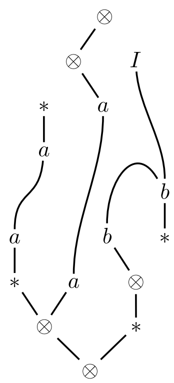
3.2. The category L$\mathbb{A}$ *of* $\mathbb{A}$*-linkings over a set* $\mathbb{A}$

Fix a set $\mathbb{A} = \{a, b, c, \ldots\}$ of generators. An ***atom*** is any generator in $\mathbb{A}$ or the constant I. An $\mathbb{A}$***-shape*** is an expression generated from atoms by binary tensor $\otimes$ and unary dual $(-)^*$, e.g. $\big(a \otimes (b^* \otimes a^{**})\big)^{***} \otimes I^*$. The ***sign*** of an atom or tensor in a shape is ***positive*** $+$ iff it is under an even number of duals $(-)^*$, otherwise ***negative*** $-$. Here is a shape with signs subscripted: $\big(\underset{-}{I}^{**} \underset{-}{\otimes} (\underset{+}{a}^{***} \underset{-}{\otimes} (\underset{+}{I} \underset{+}{\otimes} \underset{+}{b})^*)\big)^*$. Write $|S|$ for the underlying signed set of a shape S, obtained from its leaves, i.e., its occurrences of atoms. A ***leaf function*** $X \to Y$ between signed sets is a partial leaf function $X \to Y$ which is total (i.e., defined on the whole of $X^+ + Y^-$). A ***leaf function*** $f : S \to T$ between shapes is a leaf function $f : |S| \to |T|$ between the underlying signed sets. The ***graph of*** f is the disjoint union of the underlying parse trees of S and T (trees labelled with atoms at the leaves and $\otimes$ or $*$ at internal vertices) together with the edges of f, undirected. For example, if $f : S \to T$ is



then the graph of f is shown below-left:





A ***switching*** of a leaf function $f : S \to T$ between shapes is any subgraph of the graph of f obtained by deleting one of the two argument edges of each positive tensor in S and negative tensor in T. See above-right for an example. A leaf function $f : S \to T$ is an $\mathbb{A}$***-linking*** if it satisfies:

(1) Matching. Restricting f to a-labelled leaves (both in S and in T) yields a bijection for each generator $a \in \mathbb{A}$
(2) Switching. Every switching of f is a tree.

The leaf function above-left is an $\mathbb{A}$-linking: its switching shown above-right is a tree, as are its seven other switchings. Other examples of $\mathbb{A}$-linkings are depicted in the Introduction.

The linking criterion (conditions (1) and (2)) is the analogue of allowability in [10,24], presented combinatorially rather than inductively/syntactically. It is traditional in linear logic [11] to present allowability combinatorially. The criterion above derives from a standard one for multiplicative proof nets [9,18].

Proposition 2. *Verifying that a leaf function is a linking is linear time in the number of leaves.*

Thus the linking criterion (allowability) is very simple. This proposition is proved in Section 5.3, using one-sided linkings.

Given linkings $f : S \to T$ and $g : T \to U$ define $f; g : S \to U$ as their path composite (in the category $\mathcal{L}$ of partial leaf functions between signed sets, defined in Section 3.1).

Proposition 3. *The composite of two linkings is a linking.*

The proof is in Section 5.3, using one-sided linkings.

Write L$\mathbb{A}$ for the category of $\mathbb{A}$-linkings between $\mathbb{A}$-shapes. Identities are inherited from $\mathcal{L}$: the identity $S \to S$ has an edge between the ith leaf in the input shape and the ith leaf in the output shape. The identity is a well-defined linking (i.e., every switching is a tree) by a simple induction on the number of tensors in S.

Compatibility. Given leaf functions $f : X \to Y$ and $g : Y \to Z$ between signed sets, write $f + g$ for the disjoint union of f and g, viewed as a simple directed graph on $X + Y + Z$.[18] The following theorem will not be used in the sequel; we present it since it is the analogue of Theorem 2.1 in [24].

Theorem 1 (*Compatibility*)**.** *If $f : S \to T$ and $g : T \to U$ are linkings between shapes, then $f + g$ contains no cycle.*

By a ***cycle*** we mean an undirected graph [5] on a vertex set $\{v_1, \ldots, v_n\}$, all v_i distinct, $n \geq 3$, and with an edge $v_i v_j$ iff $j = i + 1 \bmod n$. The proof of the Compatibility Theorem is in Section 5.3, using one-sided linkings.

3.3. $\mathsf{L}\mathbb{A}$ *is split star-autonomous*

The category $\mathsf{L}\mathbb{A}$ of $\mathbb{A}$-linkings is split star-autonomous, as defined in Section 2.

Tensor and dual act symbolically on objects, i.e., the tensor of shapes S and T is the shape $S \otimes T$, and the dual of S is S^*. Tensor acts as disjoint union on morphisms, hence is functorial. Given linkings $f : S \to T$ and $f' : S' \to T'$, the leaf function $f \otimes f' : S \otimes S' \to T \otimes T'$ is a well-defined linking since every switching of $f \otimes f'$ is a disjoint union of switchings of f and f', connected at the tensor of $T \otimes T'$, together with one of the two argument edges of the tensor of $S \otimes S'$. The dual $f^* : T^* \to S^*$ of a linking $f : S \to T$ has the same underlying directed graph as f, hence $(-)^*$ is functorial, full and faithful.

Associativity and symmetry are the obvious bijective leaf functions (exactly the associativity and symmetry involutions of [24, Section 3]): associativity $(S \otimes T) \otimes U \to S \otimes (T \otimes U)$ has edges between the ith leaf of the input and the ith leaf of the output, and symmetry $S \otimes T \to T \otimes S$ has edges between the ith leaf of S in $S \otimes T$ and the ith leaf of S in $T \otimes S$, and similarly for T. Both are well-defined linkings by a simple induction (analogous to the well-definedness of the identity).

The natural isomorphism $\mathsf{L}\mathbb{A}(S \otimes T, U^*) \cong \mathsf{L}\mathbb{A}(S, (T \otimes U)^*)$ is the restriction of the corresponding natural isomorphism $\mathcal{L}(|S| + |T|, |U|^*) \cong \mathcal{L}(|S|, (|Y| + |Z|)^*)$ in the underlying compact closed category $\mathcal{L}$ of leaf functions between signed sets. This restriction is well-defined since switchings $S \otimes T \to U^*$ and $S \to (T \otimes U)^*$ are in bijection.

Define $\lambda_S : S \to I \otimes S$ as the identity $S \to S$ together with $I \otimes (-)$ added to the syntax of the output shape, and define $\rho_S : S \to S \otimes I$ similarly. Since the added I and $\otimes$ are positive, the switchings of λ_S and ρ_S are in bijection with those of the identity, hence λ_S and ρ_S are well-defined linkings. The requisite triangle commutes since the edge between the distinguished I's in associativity $\alpha : (S \otimes I) \otimes T \to S \otimes (I \otimes T)$ does not connect to an edge of $\rho \otimes \mathsf{id}$ during composition. Naturality of λ_S and ρ_S holds because there is no edge to the added I.

Define the retraction $\overline{\lambda}_S : I \otimes S \to S$ from the identity $S \to S$ by adding $I \otimes (-)$ to the input shape together with an edge from the added I to an arbitrary positive leaf of the S on the right of the arrow $I \otimes S \to S$ or a negative leaf of the S on the left of the arrow. This is a well-defined linking since every switching is a switching of the identity $S \to S$ together with two new edges and two new vertices, arranged so that the graph remains a tree, irrespective of whether the added tensor is switched left or right. We have $\lambda_S; \overline{\lambda}_S = \mathsf{id}_S$ since the edge of $\overline{\lambda}_S$ from the added I meets no edge of λ_S during the composition $\lambda_S; \overline{\lambda}_S$. The retraction $\overline{\rho}_S : S \otimes I \to S$ analogous.[19]

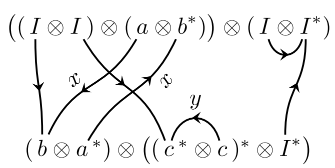
3.4. *The category* $\mathsf{L}\mathbb{A}$ *of* $\mathbb{A}$*-linkings over an arbitrary base category* $\mathbb{A}$

This section generalises $\mathbb{A}$-linkings from discrete $\mathbb{A}$ to an arbitrary category $\mathbb{A}$. Shapes are generated from the objects of $\mathbb{A}$ as before. A ***leaf function*** $S \to T$ between shapes is a partial leaf function $|S| \to |T|$ between the underlying signed sets which is total, equipped with a labelling: every edge from a leaf labelled by a generator (object of $\mathbb{A}$) is labelled with a morphism of $\mathbb{A}$. A leaf function $f : S \to T$ is an $\mathbb{A}$***-linking*** if it satisfies:

(1a) Bijection. Restricting f to $\mathbb{A}$-labelled leaves (both in S and in T) yields a bijection.
(1b) Labelling. If x is the label of an edge from a leaf labelled a to a leaf labelled b, then $x : a \to b$ is a morphism in $\mathbb{A}$.
(2) Switching. Every switching of f is a tree.

(In forming the switchings of f, ignore edge labels.) Conditions (1a) and (1b) reduced to (1) Matching (in Section 5) in the discrete case (since all labels are identities).

For example, if $x : a \to b$ and $y : c \to c$ in $\mathbb{A}$, then



is a linking from the upper to the lower shape.

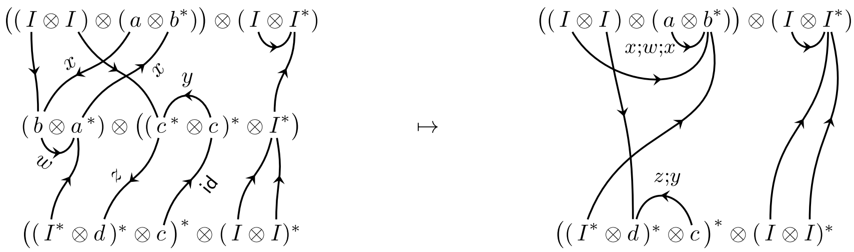
[18] A ***simple directed graph*** on a set V (cf. [5]) is a set of edges on V, where an ***edge on*** V is an ordered pair vw of distinct elements of V.

[19] Note that one cannot choose natural retractions: there are two candidates for $\overline{\lambda}_{a\otimes a}$, and in either case, naturality $\overline{\lambda}; \sigma = (\mathsf{id} \otimes \sigma); \overline{\lambda}$ fails for the symmetry map $\sigma : a \otimes a \to a \otimes a$.

Composition is path composition, as in the discrete case, but simultaneously collecting labels along each path and composing them in $\mathbb{A}$. More precisely, if $l_1 \dots l_n$ is a path traversed during (underlying discrete) composition, resulting in the edge $\langle l_0, l_n \rangle$ in the (underlying discrete) composite, then:

- if every edge $\langle l_{i-1}, l_i \rangle$ is labelled with a morphism x_i in $\mathbb{A}$, the composite edge $\langle l_0, l_n \rangle$ is labelled by the composite $x_n \dots x_1$ in $\mathbb{A}$,
- otherwise $\langle l_0, l_n \rangle$ is unlabelled.

Thus an edge in the composite is labelled iff every edge along the path giving rise to it is labelled. Here is an example of composition:

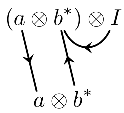


By the Unique Path Property (Proposition 1), composition is well-defined: there is no ambiguity in constructing the labels on the edges of a composite. Composition is associative since concatenation of paths is associative, and composition in $\mathbb{A}$ is associative. The identity linking has all labels identities in $\mathbb{A}$.

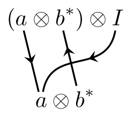
Split star-autonomy. The split star-autonomous structure carries over from the discrete case. Labels on linking edges do not interfere: every label of a canonical map (associativity, symmetry or unit map) is an identity.

4. Nets

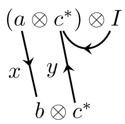
This section quotients the category $\mathsf{L}\mathbb{A}$ of $\mathbb{A}$-linkings by Trimble rewiring [33,3], yielding the free star-autonomous category generated by $\mathbb{A}$. Define two $\mathbb{A}$-linkings as ***similar*** if one can be obtained from the other by re-targeting an edge from an I, e.g.



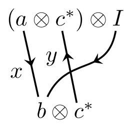
and



in the discrete case, or, if $x : a \to b$ and $y : c \to c$ are morphisms in $\mathbb{A}$,



and



An $\mathbb{A}$-***net*** is an equivalence class of $\mathbb{A}$-linkings modulo similarity (i.e., modulo the equivalence relation generated by similarity).

Theorem 2 (*Net Compositionality*)**.** *Composition of $\mathbb{A}$-linkings respects equivalence.*

In other words, if $f, f' : S \to T$ and $g, g' : T \to U$ are linkings with f equivalent to f' and g equivalent to g', then the composite linkings $f; g$ and $f'; g' : S \to U$ are equivalent. This theorem is proved in Section 6 via one-sided nets. By the theorem, composition of $\mathbb{A}$-nets is well-defined: given proof nets $[f] : S \to T$, and $[g] : T \to U$, where $[h]$ denotes the equivalence class of a linking h, define $[f]; [g] = [f; g] : S \to U$. Write $\mathsf{N}\mathbb{A}$ for the category of $\mathbb{A}$-nets. Typically we abbreviate a net $[f]$ to f, when it is clear from context that we are dealing with a net rather than a linking.

Star autonomy. The split star-autonomous structure of the category $\mathsf{L}\mathbb{A}$ of $\mathbb{A}$-linkings respects equivalence, yielding a star-autonomous structure on $\mathsf{N}\mathbb{A}$.

On morphisms, tensor in $\mathsf{L}\mathbb{A}$ is disjoint union, hence respects similarity in each argument, i.e., if linkings f and f' are similar, then $f \otimes g$ and $f' \otimes g$ are similar, as are $g \otimes f$ and $g \otimes f'$. Duality on $\mathsf{L}\mathbb{A}$ respects similarity (if f and f' are similar then f^* and f'^* are similar) since it acts trivially on the graph of a morphism (so re-targeting an edge from an I amounts to the same thing before and after dualising). Similarly, the natural isomorphism $\mathsf{L}\mathbb{A}(S \otimes T, U^*) \cong \mathsf{L}\mathbb{A}(S, (T \otimes U)^*)$ respects similarity, since a linking $S \otimes T \to U^*$ has the same underlying directed graph as its transpose $S \to (T \otimes U)^*$.

The split monomorphisms $\lambda_S : S \to I \otimes S$ and $\rho_S : S \to S \otimes I$ in $\mathsf{L}\mathbb{A}$ become isomorphisms upon quotienting. The composite

$$I \otimes S \xrightarrow{\bar{\lambda}_S} S \xrightarrow{\lambda_S} I \otimes S$$

in $\mathsf{L}\mathbb{A}$ differs from the identity $I \otimes S \to I \otimes S$ by just one edge, the edge from the distinguished I on the left of the arrow, hence is similar to the identity. Thus $\overline{\lambda}_S ; \lambda_S = \mathsf{id}_S$ in $\mathsf{N}\mathbb{A}$. Similarly, $\overline{\rho}_S ; \rho_S = \mathsf{id}_S$ in $\mathsf{N}\mathbb{A}$.

The associativity, symmetry and unit coherence diagrams commute in $\mathsf{L}\mathbb{A}$, hence also in $\mathsf{N}\mathbb{A}$.

4.1. Free star-autonomous category and full coherence

Theorem 3 (*Freeness*). *For any category $\mathbb{A}$, the category $\mathsf{N}\mathbb{A}$ of $\mathbb{A}$-nets is the free star-autonomous category generated by $\mathbb{A}$.*

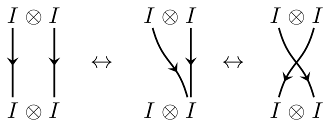
The proof is the subject of Section 7. By finiteness of equivalence classes, we have:

Theorem 4 (*Full Coherence*). *Equality of morphisms in the free star-autonomous category generated by a category is decidable.*

This theorem was first proved using the rewiring approach in [3] (in a more general form, over a polygraph with equations, and with *star-autonomous category* axiomatised as a symmetric linearly distributive category with negation).

Two examples were given in the Introduction (p. 2). Three more are provided below.

Example 3 (*Identity = twist on I.*). Example 1 (p. 2) proved $\mathsf{id}_{\perp\otimes\perp} \neq \mathsf{tw}_{\perp\otimes\perp} : \perp \otimes \perp \to \perp \otimes \perp$. The following pair of rewirings shows $\mathsf{id}_{I\otimes I} = \mathsf{tw}_{I\otimes I} : I \otimes I \to I \otimes I$.

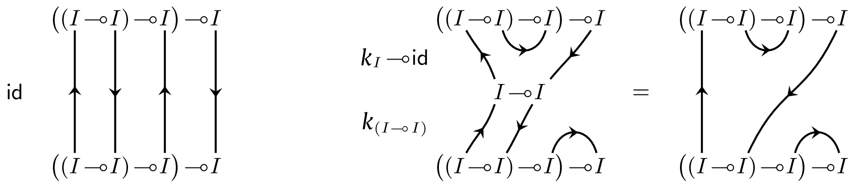


Compare this with [3, end of Section 3.1], which proves the dual result: $\mathsf{id}_{\perp\otimes\perp} = \mathsf{tw}_{\perp\otimes\perp}$.

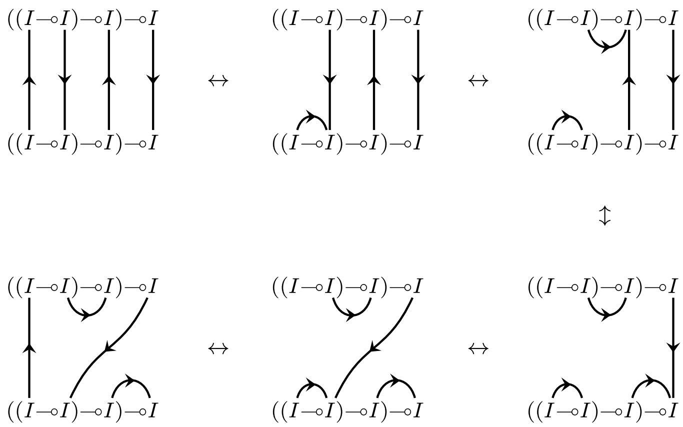
Example 4 (*Triple-dual Problem*). We show that the following diagram commutes (like Example 2, an instance of the triple-dual problem: see [24, Section 1, diagram (1.4)])

$$\begin{tikzcd} ((I \multimap I) \multimap I) \multimap I \arrow[r, "k_I \multimap \mathsf{id}"] \arrow[dr, "\mathsf{id}"'] & I \multimap I \arrow[d, "k_{(I\multimap I)}"] \\ & ((I \multimap I) \multimap I) \multimap I \end{tikzcd} \tag{2}$$

where $A \multimap B = (A \otimes B^*)^*$ and $k_A : A \to (A \multimap A) \multimap A$ is the canonical map of its type. Each path in the triangle determines a corresponding linking

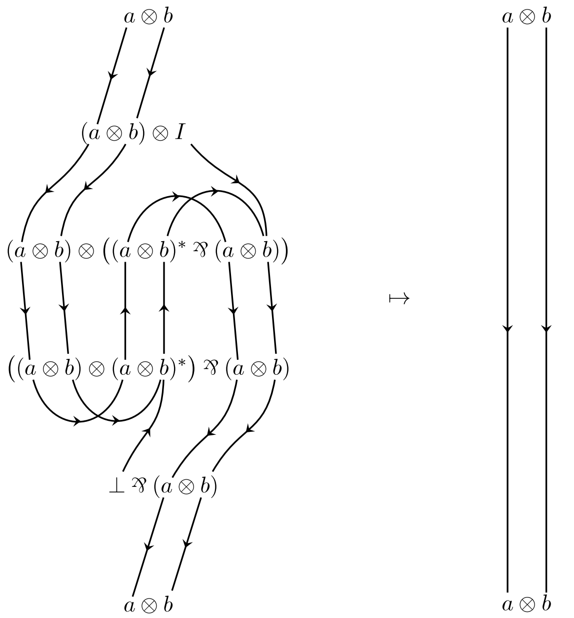


equivalent via five rewirings:



We conclude that triangle (2) commutes in every star-autonomous category. Compare this with [3, Fig. 3]. As with Example 2, a key advantage here is that the composite $k_I \multimap \mathsf{id};\ k_{(I \multimap I)}$ is immediate on inspection, since we have simple path composition.

Example 5 (*More Path Composition*)**.** The following example illustrates a larger path composition. Let $A \parr B = (A^* \otimes B^*)^*$ and $\perp = I^*$. The path composition of linkings



shows that the following diagram commutes in every star-autonomous category, where each map is canonical at its type:

$$\begin{tikzcd}
 & a \otimes b \arrow[dl] \arrow[dddd] \\
(a \otimes b) \otimes I \arrow[d] & \\
(a \otimes b) \otimes \big((a \otimes b)^* \parr (a \otimes b)\big) \arrow[d] & \\
\big((a \otimes b) \otimes (a \otimes b)^*\big) \parr (a \otimes b) \arrow[d] & \\
\perp \parr (a \otimes b) \arrow[r] & a \otimes b
\end{tikzcd}$$

This example is, of course, rather contrived. (What other canonical map $a \otimes b \to a \otimes b$ could there be?) The point is to give an example of path composition in a large diagram.

5. One-sided linkings

Eilenberg–Kelly–MacLane graphs [10,24] are *two-sided* in the sense that they are between a source and a target. Taking advantage of duality, Kelly–Laplaza graphs [23] for compact closed categories are *one-sided*: the authors define a (fixed-point free) involution on a single signed set, hence on a single shape.[20] They then define a two-sided involution, i.e., a morphism $S \to T$, as a (one-sided) involution on $S^* \otimes T$.

Since star-autonomous categories are more general than compact closed categories, to obtain a directly analogous one-sided representation one would define a two-sided linking $S \to T$ as a one-sided linking on the shape $(S \otimes T^*)^*$. However, to avoid the extra baggage of two auxiliary duals $(-)^*$ and a tensor $\otimes$, we shall instead define one-sided linkings on *sequents* $S_1, \ldots, S_n$ of shapes S_i, and define a two-sided linking $S \to T$ as a one-sided linking on the two-shape sequent S^*, T. Using

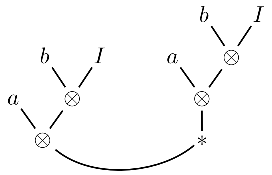
20 We refer to words built freely from generators and the constant I by binary $\otimes$ and unary $(-)^*$ as *shapes*, following the Eilenberg–Kelly–MacLane terminology for similar freely generated expressions. These shapes are exactly the words defined in [23, Section 3], objects of the free compact closed category.

the sequent calculus style, we can also define explicit cuts between shapes, facilitating an inductive proof that two-sided linkings compose, i.e., that the path composite of two (two-sided) linkings is a well-defined (two-sided) linking.

The material here on one-sided linkings amounts to the multiplicative proof nets with units in [18], with explicit negation $(-)^*$, and with axiom links between dual occurrences of generators in $\mathbb{A}$ generalised to morphisms of $\mathbb{A}$. However, we shall not require any familiarity on the part of the reader with proof nets or linear logic [11].

Sequents. The following definitions are a mild generalisation of those in Section 2 of [18].

Fix a set $\mathbb{A} = \{a, b, \ldots\}$ of generators. Henceforth identify a shape (generated from $\mathbb{A}$) with its parse tree, a tree labelled with atoms (generators or the constant I) at the leaves, and $\otimes$ and $*$ at internal vertices. (Examples of parse trees were shown in Section 3.2, in the example of a switching.) A ***sequent*** is a non-empty disjoint union of shapes. Thus a sequent is a particular kind of labelled forest. We take $S, T, \ldots$ to range over shapes, and $\Gamma, \Delta, \ldots$ to range over (possibly empty) disjoint unions of shapes. A ***cut pair*** $S \smile S^*$ is a disjoint union of a shape S with its dual S^* together with an undirected edge, a ***cut***, between the root of S and the root of S^*, e.g.



for $S = a \otimes (b \otimes I)$. A ***cut sequent*** is a disjoint union of a sequent and zero or more cut pairs. A ***switching*** of a cut sequent is any subgraph obtained by deleting one of the two argument edges of each negative $\otimes$ (cf. Section 3.2). We use comma to denote disjoint union, i.e. $S_1, \ldots, S_n$ is the disjoint union of the shapes S_i.

Linkings. We begin with the discrete case, $\mathbb{A}$ a set of generators. For the more general case of $\mathbb{A}$ an arbitrary category, see Section 5.2 below.

A ***leaf function*** on a cut sequent is a function from its negative leaves to its positive leaves. An $\mathbb{A}$***-linking*** on a cut sequent Γ is a leaf function f on Γ satisfying:

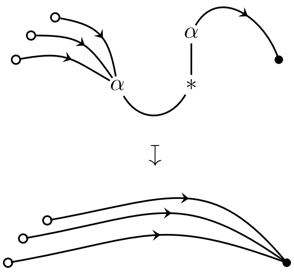
(1) Matching. For any generator $a \in \mathbb{A}$, the restriction of f to a-labelled leaves is a bijection between the negative a-labelled leaves of Γ and the positive a-labelled leaves of Γ.
(2) Switching. For any switching Γ' of Γ, the undirected graph obtained by adding the edges of f to Γ' is a tree.

Thus two-sided linkings $S \to T$ between shapes, as defined in Section 3, are in bijection with one-sided linkings on the two-shape sequent S^*, T.

5.1. Cut elimination

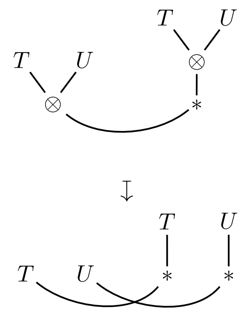
Let f be a linking on the cut sequent $\Gamma, S \smile S^*$. The result f' of ***eliminating*** the cut in $S \smile S^*$ is:

- *Atom.* Suppose S is an atom α (a generator $a \in \mathbb{A}$ or the constant I). Thus (in parse tree terms) the cut pair $S \smile S^*$ comprises leaves l^+, l^- labelled α, a vertex v labelled $*$, an argument edge from v to l^-, and a cut edge between v and l^+. Delete the cut pair (i.e., the vertices l^+, l^-, v and associated edges), and reset every f-edge to l^+ to target $f(l^-)$ instead, i.e., for all negative leaves n of the sequent such that $f(n) = l^+$, set $f(n) = f(l^-)$. Schematically,

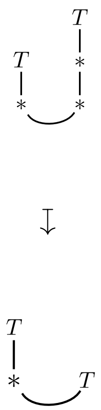


where the vertices ∘ represent all instances of n and • is $f(l^-)$. (The left-most α in the picture is the leaf l^+, the right-most α is the leaf l^-, and the $*$ is the vertex v.)

- *Tensor.* Suppose $S = T \otimes U$. Replace $\underset{\smile}{S\,S^*}$ by $\underset{\smile}{T\,T^*}$, $\underset{\smile}{U\,U^*}$. The leaves, and f, remain unchanged (identifying the leaves of T in $T \otimes U$ with T in $\underset{\smile}{T\,T^*}$, etc.). Schematically:



- *Dual.* Suppose $S = T^*$. Replace the cut pair $\underset{\smile}{T^*\,T^{**}}$ by $\underset{\smile}{T^*\,T}$. The leaves, and f, remain unchanged. Schematically,

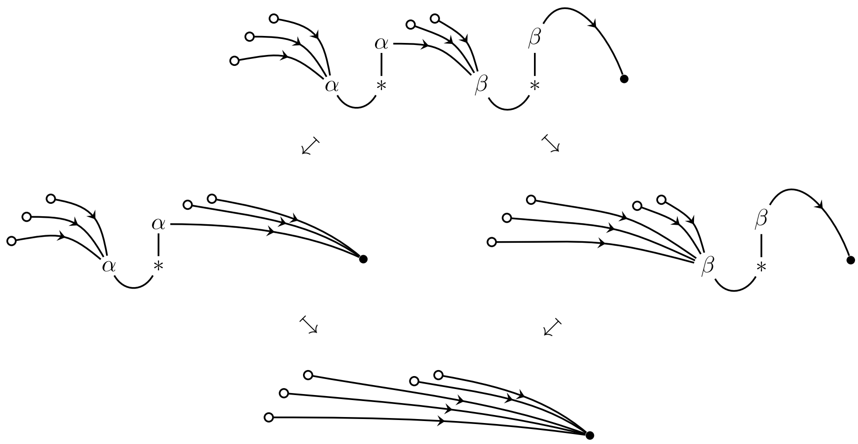


Theorem 5. *Cut elimination is well-defined: eliminating a cut from a linking on a cut sequent yields a linking on a cut sequent.*

Proof. The atomic and dual cases are trivial, since switchings correspond before and after the elimination. The tensor case is a simple combinatorial argument: any cycle in a switching after elimination induces a cycle in a switching before elimination. (This is a standard combinatorial argument for tensor elimination in multiplicative proof nets in linear logic. See [11,9,14] for details (cf. [18], Theorem 2)). □

Proposition 4. *Cut elimination is locally confluent.*

Proof. The only non-trivial case is a pair of atomic eliminations. This case is clear from the following schematic involving two interacting atomic cut redexes $\underset{\smile}{\alpha\,\alpha^*}$ and $\underset{\smile}{\beta\,\beta^*}$.



(This is adapted from the proof of Proposition 1 of [18].) □

Theorem 6. *Cut elimination is strongly normalising.*

Proof. It is locally confluent, and eliminating a cut reduces the number of vertices of the cut sequent. □

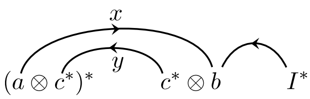
Turbo cut elimination. Analogous to [18], normalisation can be completed in a single step. For l the ith leaf of a shape S in a cut pair $S \smile S^*$, let l^* denote the ith leaf of S^*. The ***normal form*** of a cut sequent Γ is the sequent $\underline{\Gamma}$ obtained by deleting all cut pairs. Given a linking f on Γ, its ***normal form*** $\underline{f}$ is the linking on $\underline{\Gamma}$ obtained by replacing every set of directed edges $\langle l_0, l_1\rangle, \langle l_1^*, l_2\rangle, \langle l_2^*, l_3\rangle, \ldots, \langle l_{n-1}^*, l_n\rangle$ in f in which only l_0 and l_n occur in $\underline{\Gamma}$ by the single directed edge $\langle l_0, l_n\rangle$. By a simple induction on the number of vertices in cut sequents, $\underline{f}$ is precisely the normal form of f under one-step cut elimination. (In particular, this implies the turbo normal form $\underline{f}$ is indeed a linking, i.e., turbo cut elimination is well defined on linkings.)

5.2. Arbitrary base category

So far we have only presented one-sided linkings over a set $\mathbb{A}$ of generators. When $\mathbb{A}$ is a category, as in the two-sided case we add labels to every edge from a generator (object of $\mathbb{A}$), and define an $\mathbb{A}$-***linking*** on a cut sequent Γ as a leaf function f on Γ satisfying:

(1a) Bijection. Restricting f to generator-labelled leaves of Γ yields a bijection.
(1b) Labelling. If x is the label of an edge from a leaf labelled a to a leaf labelled b, then $x : a \to b$ is a morphism in $\mathbb{A}$.
(2) Switching. For any switching Γ' of Γ, the undirected graph obtained by adding the edges of f to Γ' is a tree.

For example, if $x : a \to b$ and $y : c \to c$ are morphisms in $\mathbb{A}$, then



is a (one-sided) $\mathbb{A}$-linking on the three-shape sequent $(a \otimes c^*)^*,\ c^* \otimes b,\ I^*$. As in the two-sided case, conditions (1a) and (1b) reduce to (1) Matching (in Section 5) in the discrete case, since every label is an identity.

Atomic cut elimination (in Section 5.1) incorporates labels as follows: when re-setting an f-edge $n \to l^+$ to target $f(l^-)$ instead, let the $\mathbb{A}$-morphism x be the label of the edge $n \to l^+$ (if any) and let the $\mathbb{A}$-morphism y be the label of the edge $l^- \to f(l^-)$ (if any); if both x and y are present, label the output edge $n \to f(l^-)$ by the composite $y; x$ in $\mathbb{A}$, otherwise leave $n \to f(l^-)$ unlabelled. Correspondingly, we adjust the definition of turbo cut elimination: the output edge $\langle l_0, l_n\rangle$ is labelled by the composite $\mathbb{A}$-morphism $x_1; \ldots; x_n$ iff every $\langle l_{i-1}, l_i\rangle$ is labelled by an $\mathbb{A}$-morphism x_i (cf. path composition of two-sided linkings defined in Section 3.4).

The properties of cut elimination (Theorem 5, Proposition 4 and Theorem 6) are unaffected by the presence of labels.

5.3. From one-sided linkings to two-sided linkings

By design, two-sided linkings $S \to T$ between shapes, as defined in Section 3, are in bijection with one-sided linkings on the two-shape sequent S^*, T. Via this correspondence, we can take care of the proof obligations remaining from Section 3.

Proof of Proposition 2 (*Linear Time Verification of the Linking Criterion*)**.** Every star-autonomous shape S induces a formula $\widehat{S}$ of multiplicative linear logic: replace negative tensors in S by pars $⅋$, replace negative generators a by $a^\perp$, replace negative I's by $\perp$, and delete all duals $(-)^*$. For example, $\big((1 \otimes a^*)^* \otimes 1\big)^*$ becomes $(1 \otimes a^\perp) ⅋ \perp$. Thus every shape sequent $\Gamma = S_1, \ldots, S_n$ induces a formula sequent $\widehat{\Gamma}$, namely $\widehat{S_1}, \ldots, \widehat{S_n}$. The formula $\widehat{S}$ has the same leaf vertices as the original shape S, and switchings of $\widehat{S}$ are in bijection with switchings of S. Thus a leaf function on a shape sequent Γ is a linking iff it constitutes a multiplicative proof net on the formula sequent $\widehat{\Gamma}$, in the sense of [18]. Via the translation $\Gamma \mapsto \widehat{\Gamma}$, Proposition 2 becomes a corollary of Theorem 4 of [18] (which, in turn, is not much more than a corollary of linear time verification of the proof net criterion for unit-free multiplicative nets [16,29]). □

Given linkings $f : S \to T$ and $g : T \to U$, since the definition of turbo cut elimination is precisely path composition, the composite $f; g : S \to U$ corresponds to the normal form of the one-sided linking $f \cup g$, the disjoint union of f and g, on the cut sequent $S^*, T \smile T^*, U$.

Proof of Proposition 3 (*Two-sided Linkings Compose*)**.** Via the correspondence just described, this is a corollary of Theorem 5 (and the correspondence between turbo cut elimination of one-sided linkings and normalisation by one-step cut elimination). □

Proof of Theorem 1 (*Compatibility: Cycles do not Arise During Composition*)**.** Suppose $f : S \to T$ and $g : T \to U$ are linkings such that the disjoint union $f + g$, a directed graph on $|S| + |T| + |U|$, contains a cycle. Let $f \cup g$ be the corresponding one-sided linking on the cut sequent $S^*, T \smile T^*, U$, and let $\overline{f \cup g}$ be the result of eliminating all tensor $\otimes$ and dual $(-)^*$ cuts from $f \cup g$, a one-sided linking on the cut sequent $S,\ \alpha_1 \smile \alpha_1^*, \ldots,\ \alpha_n \smile \alpha_n^*, U$, where α_i is the atom labelling the ith leaf of T. Had the directed graph $f + g$ a cycle C, then $\overline{f \cup g}$ would contain a cycle in every switching (the edges of the cycle C alternating with cut edges between the α_i), and therefore fail to be a linking, contradicting Theorem 5. □

See [4] for more on the relationship between compatibility and the multiplicative proof net criterion, in the unit-free case.

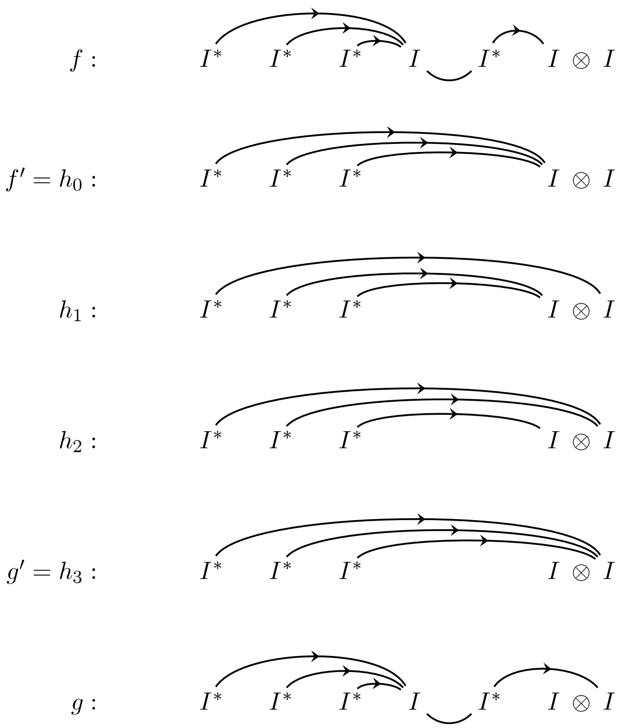
6. One-sided nets

We define a one-sided net as a one-sided linking modulo Trimble rewiring [33,3].

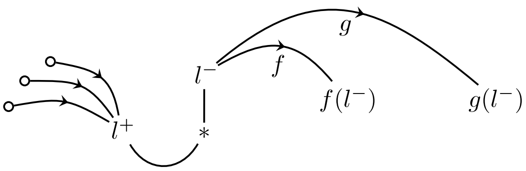
One-sided $\mathbb{A}$-linkings f and g on a cut sequent Γ are ***similar*** if they differ on a negative I, i.e., one can be obtained from the other by re-targeting one edge from an I. A one-sided $\mathbb{A}$***-net*** on a cut sequent Γ is an equivalence class of $\mathbb{A}$-linkings on Γ modulo similarity (i.e., modulo the transitive closure of similarity). Similarity here coincides with the earlier two-sided case: two-sided $\mathbb{A}$-linkings $f, g : S \to T$ are similar iff the corresponding one-sided linkings on the two-shape sequent S^*, T are similar in the sense just defined.

Theorem 7. *Cut elimination respects equivalence, i.e., cut elimination is well-defined on $\mathbb{A}$-nets. More precisely: if f and g are equivalent $\mathbb{A}$-linkings on the cut sequent Γ containing a cut pair $S \smile S^*$, and the $\mathbb{A}$-linkings f' and g' result from eliminating $S \smile S^*$ from f and g, respectively, then f' and g' are equivalent.*

Before proving theorem (below), we illustrate it with an example. (The notation in the example corresponds to the notation in the proof.) Similar (hence equivalent) linkings f and g are shown top and bottom, with adjacent normal forms f' and g'; the sequence h_0, h_1, h_2, h_3 of pairwise similar linkings witnesses the equivalence of f' and g'.



Proof. If $S \smile S^*$ is a tensor or dual cut, the result is trivial, since leaves and linkings are untouched by eliminating the cut. Suppose the cut is atomic. Let l^- be the negative leaf of $S \smile S^*$. If l^- is not the leaf l on which f and g differ, then f' and g' are similar or equal (hence equivalent, as desired) since they differ on at most l after eliminating the cut. So assume $l^- = l$. Thus we have the situation



where the leaf vertices l^- and l^+ are labelled by I, the unlabelled directed edges are present in both f and g, and the edge labelled f (resp. g) is present in f (resp. g) only. The vertices $\circ$ schematically represent the negative leaves $l_1 \ldots l_n$ of Γ whose edge targets l^+ (in both f and g), i.e., such that $f(l_i) = g(l_i) = l^+$. Note that, since l^+, l^- and the l_i are labelled I, none of the edges from them (i.e., the directed edges depicted above) is labelled by an $\mathbb{A}$-morphism, so the case of non-discrete $\mathbb{A}$ coincides with the case of $\mathbb{A}$ a set.

Let h be the (partial) leaf function obtained from f (or equivalently g) by deleting the edge from l^-. Since f and g are linkings, every switching σ of h is a disjoint union of two trees, with all l_i in one tree and $f(l^-)$ and $g(l^-)$ in the other (otherwise the corresponding switching of one of f or g would contain a cycle, via the edge between l^- and $f(l^-)$ or $g(l^-)$, respectively). Let h^- be the result of deleting from h all the edges from the l_i, and for $0 \leq j \leq n$ construct h_j from h^- by adding an edge from l_i to $g(l^-)$ for $1 \leq i \leq j$, and from l_i to $f(l^-)$ for $j < i \leq n$, and also an edge from l^- to $f(l^-)$ (or $g(l^-)$; the choice is arbitrary). Since for any switching of h, the l_i are in one tree and $f(l^-)$ and $g(l^-)$ are in the other, h_i is a well-defined linking. Let h'_i be result of eliminating the atomic cut from h_i. By design, $f' = h'_0$ and $g' = h'_n$. By Theorem 5, each h'_i is a linking. Since h'_i and h'_{i-1} differ on just one I (namely l_i), they are similar, so f' and g' are equivalent, via the h'_i. □

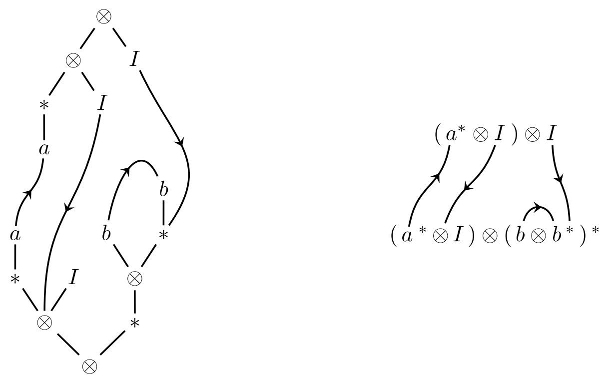
By the correspondence between (turbo) cut elimination of one-sided linkings and path composition of two-sided linkings, we obtain Theorem 2 (compositionality of two-sided nets).

7. Proof that the category $\mathsf{N}\mathbb{A}$ of $\mathbb{A}$-nets is free star-autonomous

This section proves the Freeness Theorem (Theorem 3): *For any category $\mathbb{A}$, the category $\mathsf{N}\mathbb{A}$ of $\mathbb{A}$-nets is the free star-autonomous category generated by $\mathbb{A}$.*

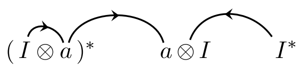
7.1. Lax linkings

Suppose $\mathbb{A}$ is a set. Given $\mathbb{A}$-shapes S and T, define a ***lax leaf function*** $S \to T$ as the relaxation of a leaf function (as defined in Section 5) obtained by permitting edges from I's to target any vertex of S or T (viewed as a parse trees). For example, here is a lax leaf function from $(a^* \otimes I) \otimes I$ to $(a^* \otimes I) \otimes (b \otimes b^*)^*$, drawn in parse-tree form on the left, and compact in-line form on the right:



Define the graph of a lax leaf function $f : S \to T$ by analogy with the original non-lax case: the undirected graph which is the disjoint union of the two parse trees S and T, together with the edges of f, undirected. Define a switching as before. A ***lax linking*** is a lax leaf function satisfying the linking criterion, i.e., conditions (1) Matching and (2) Switching in Section 3.2. For example, the lax leaf function depicted below-left is a lax linking, since both its switchings are trees, but the lax leaf function below-right is not, since the switching in which the upper tensor chooses its right argument has a cycle.

We make the corresponding lax definitions in the one-sided case. Given a cut sequent Γ over $\mathbb{A}$, define a ***lax leaf function*** on Γ as the variant of a leaf function obtained by permitted edges from I's to target any vertex of Γ. Switchings again generalise to lax leaf functions in the obvious way. Define a ***lax linking*** on Γ as a lax leaf function on Γ which satisfies the linking criterion, i.e., conditions (1) Matching and (2) Switching in Section 5 For example, here is a lax linking on the three-shape sequent $(I \otimes a)^*,\ a \otimes I,\ I^*$:



When $\mathbb{A}$ is an arbitrary category, generalise the definitions of lax linking exactly as in the original non-lax case: add $\mathbb{A}$-morphisms as labels on edges between generators (objects of $\mathbb{A}$), then replace condition (1) Matching by conditions (1a) Bijection and (1b) Labelling (given in Section 3.4 in the two-sided case, and those given in Section 5.2 in the one-sided case). Extending the non-lax case, lax linkings $S \to T$ correspond to lax linkings on the two-shape sequent S^*, T such that no edge targets the distinguished (i.e. outermost) $*$-vertex of S^*.

7.2. Lax equivalence

Define two lax linkings as ***similar*** if they differ by a single edge from an I, and ***lax equivalent*** if they are equivalent modulo similarity on lax linkings (i.e., modulo the reflexive–transitive closure of similarity). To help avoid ambiguity, while dealing with lax linkings in this section we shall refer to the original non-lax notion of a linking as a ***standard linking***, and the original non-lax notion of equivalence between standard linkings as ***standard equivalence***. The following Lemma, proved in Section 7.2.1, is the key technical step in the proof that the category $\mathsf{N}\mathbb{A}$ is free.

Lemma 1 (*Lax Rewiring*)**.** *Standard linkings are standard equivalent iff they are lax equivalent.*

Thus if the standard linking f can be rewired to the standard linking g along a sequence $f = h_1 \dots h_n = g$ of lax linkings with h_i similar to h_{i-1} for $1 < i \leq n$, then f can be rewired to g along a sequence of *standard* linkings: there exists a sequence $f = k_1 \dots k_m = g$ of standard linkings with k_i similar to k_{i-1} for $1 < i \leq n$. In other words, adding lax linkings has no impact on equivalence of standard linkings: no additional standard linkings are identified when we permit 'lax rewiring', via lax linkings.

7.2.1. Proof of the Lax Rewiring Lemma

Since two-sided lax linkings $S \to T$ are in bijection with certain one-sided lax linkings on the two-formula sequent S^*, T, henceforth we shall assume all lax linkings are one-sided.

An ***atomic linking*** is a lax linking whose every edge targets an atom. (Note that an atomic linking need not be a standard linking since an edge from an I may target a negative leaf.) Define atomic linkings f and g as ***atomic equivalent*** if they are lax equivalent via atomic linkings: there exists a sequence $f = h_1 \dots h_n = g$ of atomic linkings with h_i similar to h_{i-1} for $1 < i \leq n$. The following lemma reduces atomic equivalence to standard equivalence.

Lemma 2. *Standard linkings are standard equivalent iff they are atomic equivalent.*

Proof. Define the ***depth*** of an atomic linking as the number of its edges which target negative leaves. Thus an atomic linking is a standard linking iff it has depth 0. Define the depth of a sequence of lax linkings $f_1 \dots f_m$ as the maximum of the depths of the f_i. Suppose $f_1 \dots f_m$ has depth d. The ***size*** of $f_1 \dots f_m$ is the number of f_i of depth d.

Suppose f and g are standard linkings which are atomic equivalent via a sequence $f = h_1 \dots h_n = g$ of atomic linkings with h_i similar to h_{i-1} for $1 < i \leq n$. We proceed by a primary induction on the depth of $h_1 \dots h_n$, and a secondary induction on its size.

- Primary induction base: $h_1 \dots h_n$ has depth 0. Then all h_i are standard linkings, so f and g are already standard equivalent.
- Primary induction step: $h_1 \dots h_n$ has depth $d > 0$. Let pqr be a three-element subsequence of $h_1 \dots h_n$ with q of depth d (i.e., q is of maximum depth in the sequence) and r of depth $d - 1$. (Such a subsequence exists since $h_n = g$ has depth 0, and consecutive elements in the sequence differ by at most one in depth.)

  Let l be the negative leaf such that $q(l)$ is a negative leaf l^-, and $r(l)$ is a positive leaf l^+, and otherwise q and r are identical. (The leaf l must exist, since q has depth d while r has depth $d-1$.) Let l' be the negative leaf such that $p(l') \neq q(l')$, and let $l'' = p(l')$. If $l \neq l'$ then p, q and r are:

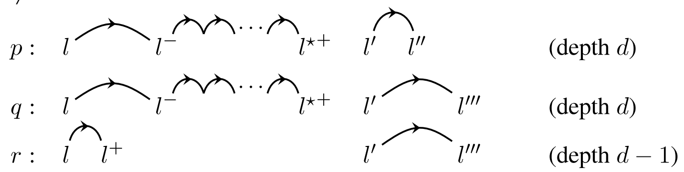
$$\begin{tikzcd}
p: & l \arrow[r, bend left] & l^- & l' \arrow[r, bend left] & l'' \\
q: & l \arrow[r, bend left] & l^- & l' \arrow[r, bend left] & l''' \\
r: & l \arrow[r, bend left] & l^+ & l' \arrow[r, bend left] & l'''
\end{tikzcd} \tag{3}$$

  where $l''' = q(l') = r(l')$, and if $l = l'$ then p, q and r are:

$$\begin{tikzcd}
p: & l \arrow[r, bend left] & l'' \\
q: & l \arrow[r, bend left] & l^- \\
r: & l \arrow[r, bend left] & l^+
\end{tikzcd} \tag{4}$$

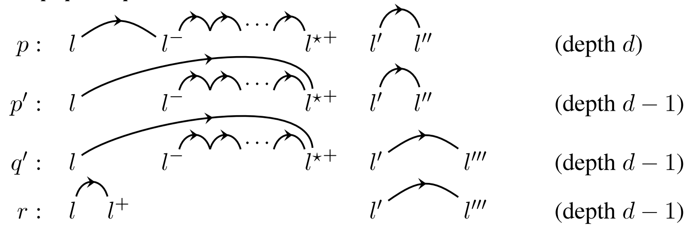
  (a) Secondary induction base: $h_1 \dots h_n$ has size 1. Thus p has depth $d - 1$. Therefore the rewirings $p \mapsto q$ and $q \mapsto r$ each re-target the same negative leaf, i.e., $l = l'$, as in diagram (4) above, so p and r are similar, without need for q as an intermediate. Delete q from $h_1 \dots h_n$, and appeal to the primary induction hypothesis with this new sequence of depth $d - 1$.
  (b) Secondary induction step: $h_1 \dots h_n$ has size $s > 1$. If p has depth $d - 1$, we can delete q exactly as in the previous case, reducing the size of $h_1 \dots h_n$ by 1, and appeal to the secondary induction hypothesis. So assume p has depth d. Let $l^{\star +}$ be the leaf at the end of the path in (the directed graph) p which begins at l^-. Thus $l^{\star +}$ is positive.

(i) Case $l^- \neq l'$. We have:



(5)

Define p' as p but for $p'(l) = l^{\star +}$ (versus $p(l) = l^-$) and similarly, define q' as q but for $q'(l) = l^{\star +}$ (versus $q(l) = l^-$). Substitute $p'q'$ for q:



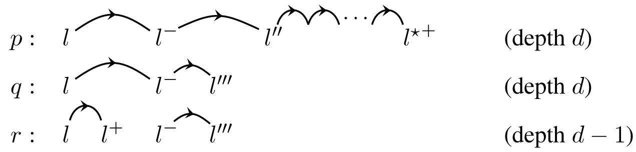
(6)

Note that p' has depth $d-1$ since p has depth d and $l^{\star +}$ is positive whereas l^- is negative; similarly, q' has depth $d-1$.

The lax leaf functions p' and q' are lax linkings (i.e., satisfy the linking correctness criterion), since p and q are lax linkings: the edge from l targets l^- in p and q, and $l^{\star +}$ in p' and q'; since l and $l^{\star +}$ are connected along the edges of the lax leaf function, a switching of p (resp. q) is a tree iff the corresponding switching of p' (resp. q') is a tree.

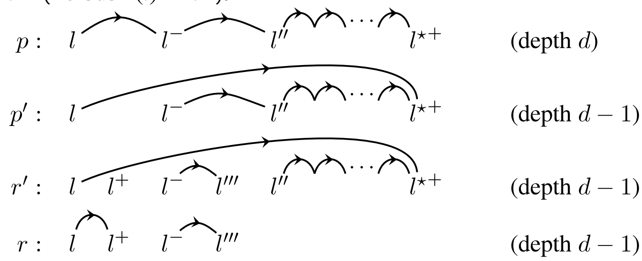
By construction, the pairs $p \leftrightarrow p'$, $p' \leftrightarrow q'$ and $q' \leftrightarrow r$ are similar. Thus we can appeal to the inductive hypothesis with the original sequence $h_1 \dots h_n$ with $p'q'$ substituted for q, which has strictly smaller size than the original (since p' and q' each have depth $d-1$, whereas q has depth d).

(ii) Case: $l^- = l'$. We have:



(7)

(Note that $l'' = l^{\star +}$ is possible.) Define p' as p but for $p'(l) = l^{\star +}$ (versus $p(l) = l^-$) and define r' as r but for $r'(l) = l^{\star +}$ (versus $r(l) = l^+$):

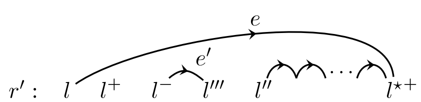


(8)

Note that p' has depth $d-1$ since p has depth d and $l^{\star +}$ is positive whereas l^- is negative, and that r' has depth $d-1$ since r has depth $d-1$ and both $l^{\star +}$ and l^+ are positive.

The lax leaf function p' is a lax linking (i.e., satisfies the linking correctness criterion), since the targets l^- and $l^{\star +}$ of the edge from l, in p and p' respectively, are connected along the edges of the lax leaf function.

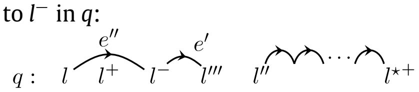
Claim: r' is a lax linking. *Proof.* Let e be the edge of r' from l to $l^{\star +}$, and let e' be the edge of r' from l^- to l'''.



(9)

Suppose C is a cycle in a switching σ of r'. The cycle C must traverse both e and e', for if it did not traverse e' it would be contained in the corresponding switching of p' (already proved to be a lax linking, and differing from r' only on l^-) whilst if it did not traverse e it would be contained in the corresponding switching of r (differing from r' only on l).

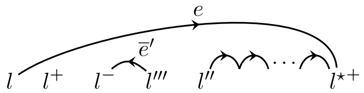
- Case: C (oriented one way or the other) has the form $e\pi e'\pi'$ for sequences of edges π and π'. Let e'' be the edge from l to l^- in q:



(10)

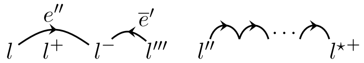
We obtain a cycle of the form $e''e'\pi'$ in the corresponding switching of q, contradicting the fact that q is a lax linking.

- Case: C (oriented one way or the other) has the form $e\pi\overline{e}'\pi'$ for sequences of edges π and π', where $\overline{e}'$ is e' traversed in the direction opposite to its given orientation:



(11)

Again, let e'' be the edge from l to l^- in q:



(12)

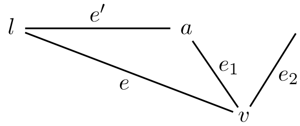
We obtain a cycle $e''\pi'$ in the corresponding switching of q, contradicting the fact that q is a lax linking.

QED Claim.

By construction, the pairs $p \leftrightarrow p'$, $p' \leftrightarrow r'$ and $r' \leftrightarrow r$ are similar, and p' and r' are lax linkings. Thus we can appeal to the inductive hypothesis with the original sequence $h_1 \dots h_n$ with $p'r'$ substituted for q, which has strictly smaller size than the original (since p' and r' each have depth $d - 1$, whereas q has depth d). □

Lemma 3. *Let f be a lax linking, let l be a negative I-labelled leaf whose f-edge targets a vertex v with an argument-vertex a (thus v is either $\otimes$ or $*$). Let f' be the result of retargeting the edge from l to point to a instead of v (i.e., the lax leaf function f' is f but with $f'(l) = a$ versus $f(l) = v$). Then f' is a lax linking.*

Proof. If v is a $*$-vertex, the result is immediate since switchings of f correspond to switchings of f'. Thus assume v is a $\otimes$-vertex. Without loss of generality, a is the left argument of v. If v is positive (hence retains both argument edges in every switching), then the result is immediate. So assume v is negative.[21] Towards a contradiction, suppose σ is a switching of f' containing a cycle C. Let e be the edge in f from l to v, let e' be the edge in f' from l to a, let e_1 be the edge between v and its left argument, and let e_2 be the edge between v and its right argument.



We may assume C contains e' (otherwise C is a cycle in the corresponding switching σ_f of f) and e_2 (otherwise we can assume e_1 is in σ (by substituting e_1 for e_2 in σ if necessary), and we obtain a cycle in σ_f by replacing e' in C by e and e_1). Thus, oriented one way or the other, C has the form $e'\pi e_2\pi'$, where e' is traversed from l to a, so $e\pi'$ (resp. $ee_2\pi'$) is a cycle if e_2 is oriented towards (resp. away from) v. □

Corollary 1. *Let f be a lax linking, let l be a negative I-labelled leaf whose f-edge targets a vertex v, and let l' be one of the leaves above v (i.e., a leaf which is a hereditary argument of v). The result of retargeting the f-edge $l \to v$ to $l \to l'$ is a lax linking.*

Proof. Iterate Lemma 3. □

Lemma 4. *Standard linkings are atomic equivalent iff they are lax equivalent.*

Proof. A three-level induction. Define the ***volume*** of a shape as its number of vertices. The volume of a vertex v in a shape is the volume of the (sub)shape rooted at v. For example, the volumes of vertices have been subscripted on the following shape: $(\underset{1}{I}\underset{2}{{}^*}\underset{3}{{}^*} \underset{13}{\otimes} ((\underset{1}{a} \underset{4}{\otimes} \underset{1}{a}\underset{2}{{}^*}) \underset{9}{\otimes} (\underset{1}{I} \underset{3}{\otimes} \underset{1}{b})\underset{4}{{}^*}))\underset{14}{{}^*}$. Define the volume of a negative I in a lax linking f as the volume of its target under f, and the volume of f as the maximum of the volumes of its negative I's. Thus f has volume 1 iff it is an atomic lax linking (i.e., every f-edge from a negative I targets a leaf). The volume of a sequence $f_1 \dots f_m$ of lax linkings is the maximum of the volumes of the f_i. Let V be the volume of $f_1 \dots f_m$. The ***depth*** of f_i (with respect to $f_1 \dots f_m$) is the number of negative I's in f_i which have volume V. Define the depth of a sequence of linkings $f_1 \dots f_m$ as the maximum of the depths of the f_i. Suppose $f_1 \dots f_m$ has depth d. The ***size*** of $f_1 \dots f_m$ is the number of f_i of depth d.

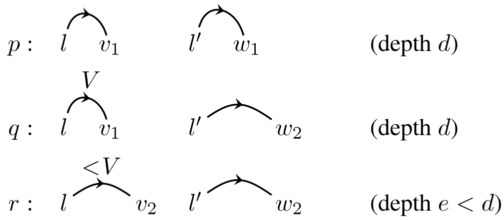
Let f and g be standard linkings, lax equivalent via a sequence $f = h_1 \dots h_n = g$ of lax linkings with h_i similar to h_{i-1} for $1 < i \leq n$. We proceed by a primary induction on the volume V of $h_1 \dots h_n$, a secondary induction on its depth d, and a tertiary induction on its size s.

If $V = 1$, then all the h_i are already atomic lax linkings.

Suppose $V > 1$. Let pqr be a subsequence of $h_1 \dots h_n$ in which q has depth $d > 0$ (hence volume V) and r has depth $e < d$. (Such a subsequence exists since $h_n = g$ has volume $1 < V$.) Let c be the depth of p. There are two cases:

[21] Note for readers familiar with linear distributivity in linear logic: the remainder of this proof is simply the usual argument (in disguise) that applying linear distributivity $A \otimes (B \parr C) \to (A \otimes B) \parr C$ preserves MLL proof net correctness. The correspondence is as follows. Let S be the (sub)shape rooted at the negative $\otimes$-vertex v in the proof in the main text; since v is negative, think of it as a par. Thus we think of S as $S_1 \parr S_2$. Substitute $I \otimes S$ for S, and retarget the f-edge from l to point to the new I; call this the lax linking $\widehat{f}$. Now apply linear distributivity (and ignore $*$-vertices, which are irrelevant for cycles in switchings), yielding a lax linking $\widehat{f}'$ on $(I \otimes S_1) \parr S_2$. This lax linking $\widehat{f}'$ corresponds to f' just as $\widehat{f}$ corresponded to f; hence f' is a well-defined lax linking.

1. *Case* $c < d$. Since $c < d > e$ the rewirings $p \leftrightarrow q$ and $q \leftrightarrow r$ both rewire an edge from the same negative I. Delete q from $h_1 \ldots h_n$. This reduces at least one of the volume, depth or size of $h_1 \ldots h_n$.
2. *Case* $c = d$. If $l = l'$, then we can simply delete q, as in the previous case. Thus assume $l \neq l'$. We have



(13)

where l and l' are negative I-labelled leaves, the v_i and w_j are vertices of unspecified type, and the volume of l is V in p and q and $<V$ in r.

Let a be a leaf above v_1. Define p' and q' from p and q by re-targeting the edge from l to target a instead of v_1. Each of p' and q' is a well-defined lax linking by Corollary 1. Substitute $p'q'$ for q in $h_1 \ldots h_n$. This reduces at least one of the volume, depth or size of $h_1 \ldots h_n$, since the volume of l in p' and q' is strictly less than V. □

Proof of Lemma 1. (The Lax Rewiring Lemma: standard linkings are standard equivalent iff they are lax equivalent) By Lemma 2 standard linkings are standard equivalent iff they are atomic equivalent, and by Lemma 4 they are atomic equivalent iff they are lax equivalent. □

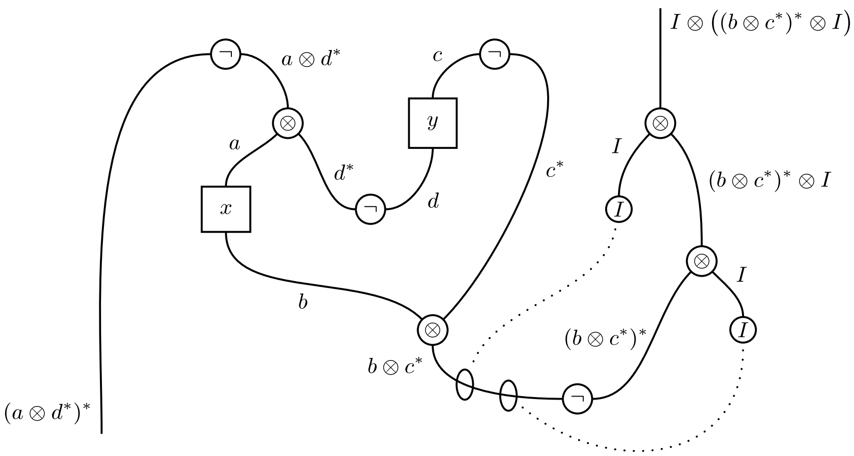
7.3. Main freeness proof

We are now ready to prove the Freeness Theorem (Theorem 3): for any category $\mathbb{A}$, the category $\mathsf{N}\mathbb{A}$ of $\mathbb{A}$-nets is the free star-autonomous category generated by $\mathbb{A}$. Rather than prove the theorem from scratch, we show that $\mathsf{N}\mathbb{A}$ is isomorphic to a full subcategory of the circuit category $\mathrm{Net}^{*}_{\mathcal{C}_{\mathbb{A}}}(E_{\mathbb{A}})$ of [3], where $(\mathcal{C}_{\mathbb{A}}, E_{\mathbb{A}})$ is the polygraph representing $\mathbb{A}$, with typed components $\mathcal{C}_{\mathbb{A}}$ and equations $E_{\mathbb{A}}$. By Theorem 5.1 of [3], $\mathrm{Net}^{*}_{\mathcal{C}_{\mathbb{A}}}(E_{\mathbb{A}})$ is the free symmetric linearly (=weakly) distributive category with negation generated by $\mathbb{A}$. Write $\mathsf{CircNet}\mathbb{A}$ for the full subcategory of $\mathrm{Net}^{*}_{\mathcal{C}_{\mathbb{A}}}(E_{\mathbb{A}})$ whose circuits are cotensor-free and cotensor-unit-free (i.e. par-free and $\perp$-free, in linear logic terminology [11]). Thus the objects of $\mathsf{CircNet}\mathbb{A}$ are in bijection with $\mathbb{A}$-shapes. By the equivalence between symmetric linearly distributive categories with negation and star-autonomous categories in [7], $\mathsf{CircNet}\mathbb{A}$ is the free star-autonomous category generated by $\mathbb{A}$, as a corollary of Theorem 5.1 of [3]. Thus our Freeness Theorem (Theorem 3) follows from:

Proposition 5. $\mathsf{N}\mathbb{A}$ *is isomorphic to* $\mathsf{CircNet}\mathbb{A}$.

7.4. Proof of Proposition 5

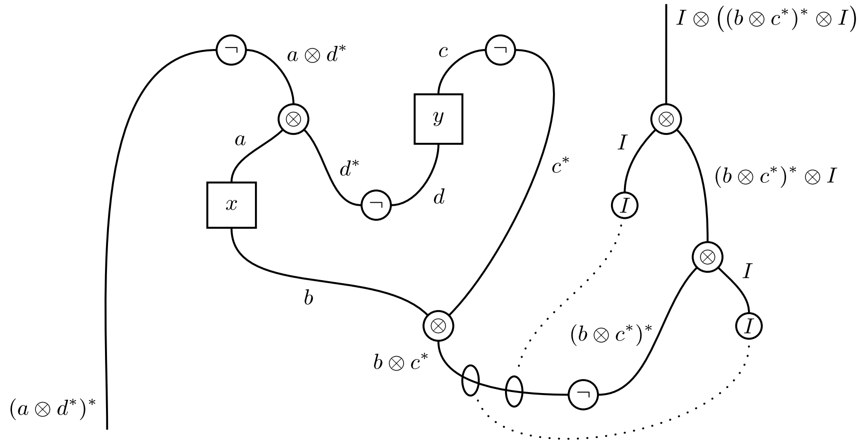
In this paper we have defined a *net* as an equivalence class of *linkings*. For $\mathsf{CircNet}\mathbb{A}$ we shall use a similar two-level convention: henceforth ***circuit-net*** refers to an equivalence class (a morphism of $\mathsf{CircNet}\mathbb{A}$) and ***circuit*** refers to a representative.[22] For example, given morphisms $x : a \to b$ and $y : c \to d$ in $\mathbb{A}$, here is a normal (i.e., redex-free[23]) circuit $I \otimes \big((b \otimes c^{*})^{*} \otimes I\big) \to (a \otimes d^{*})^{*}$:



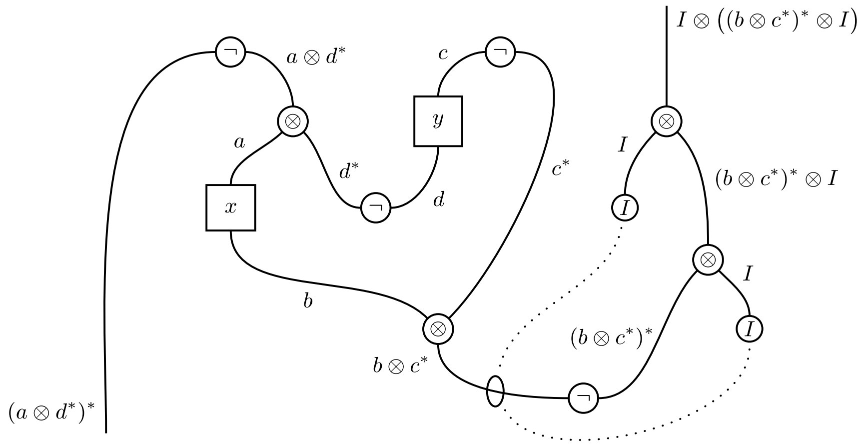
22 Thus, in particular, we shall always assume a circuit satisfies the correctness criterion.

23 [3] was forced to work with equivalence classes including un-normalised circuits since thinning links could block redexes.

(We write I for the tensor unit, denoted $\top$ in [3].) Define a ***canonical circuit*** as a normal circuit modulo the ordering of thinning links attached along each wire. For example, the following normal circuit denotes the same canonical circuit as the normal circuit above:

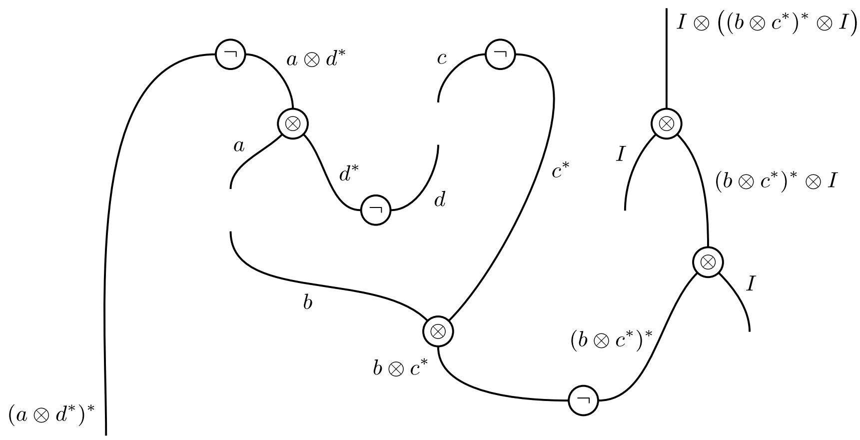


We render a canonical circuit uniquely by superimposing the attachment points of thinning links, for example, drawing the above canonical circuit as

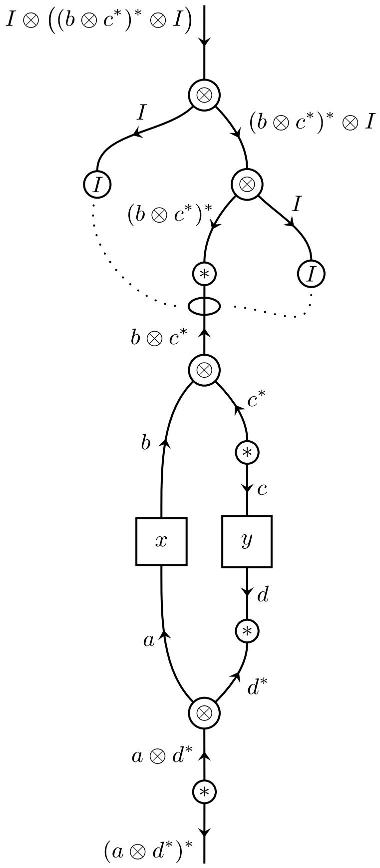


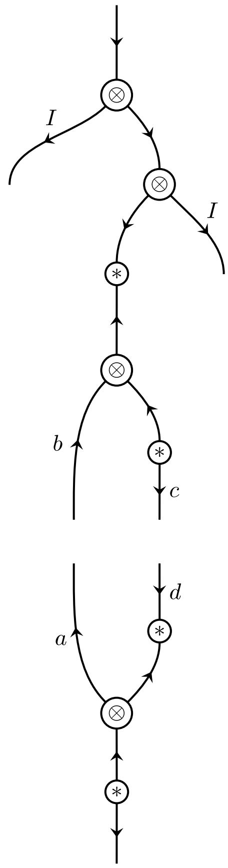
Lemma 5. *Lax linkings $S \to T$ are in bijection with canonical circuits $S \to T$.*

Proof. Deleting the thinning links and generators from a canonical circuit $S \to T$ leaves the parse tree structures of S and T. For example, the circuit above leaves

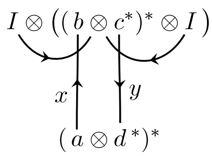


The parse tree relationship is clearer if we (1) abstract away from the dependency of the distinction between input and output wires/ports[24] on the up/down direction in the page, by explicitly orienting the wires from output to input, and (2) change the $\neg$ label to $*$, to match the shape syntax. For example, the canonical circuit above (prior to deleting the generators and thinning links) becomes the circuit below-left,





and deleting the generators x and y, the thinning links, and non-atomic labels more obviously leaves the parse trees of the shapes $S = I \otimes \big((b \otimes c^*)^* \otimes I\big)$ and $T = (a \otimes d^*)^*$, as shown above-right. Generators and thinning links can then be viewed as the labelled edges and unlabelled edges of a lax linking $S \to T$. For example, the canonical circuit above-left becomes the lax linking



□

Equivalence between canonical circuits is well-defined since re-ordering of thinning links along a wire is a sub-relation of circuit equivalence.

Lemma 6. *Lax linkings are lax equivalent iff the corresponding canonical circuits are equivalent.*

Proof. By the Empire Rewiring Proposition [3, Prop. 3.3], a thinning link can be moved to any wire in its empire. Since (by definition) we are only dealing with circuits satisfying the correctness criterion, such moves correspond to arbitrary retargeting of edges from negative Is, between (correct) circuits. □

Lemma 7. *Every lax linking $S \to T$ is lax equivalent to a standard linking $S \to T$.*

Proof. Using Corollary 1 we can re-target all the edges from negative Is to target leaves. Then, suppose an edge from a leaf l targets a negative leaf l', and suppose the edge from l' targets l''. Shift the edge from l to target l'' instead. Iterating this procedure leads to all edges targeting positive leaves, yielding a standard linking. □

---

24 Lambek's *covariables* and *variables*: [3, Section 2.1].

For any lax linking f, write f^{c} for the corresponding canonical circuit (via the bijection of Lemma 5). This induces a bijection between nets and circuit-nets:

Proposition 6. *There is a bijection* $(\ )^{n} : \mathsf{N}\mathbb{A}(S, T) \to \mathsf{CircNet}\mathbb{A}(S, T)$*, for all shapes* S, T.

Proof. Define $[f]^{n} = [f^{c}]$, the equivalence class of the canonical circuit f^{c}. This is independent of the choice of f by Lemma 6, injective by Lemma 1, and surjective by Lemma 7. $\square$

To obtain an isomorphism of categories $\mathsf{N}\mathbb{A} \cong \mathsf{CircNet}\mathbb{A}$, completing the proof of Proposition 5, we must prove that $(\ )^{n}$ is functorial.

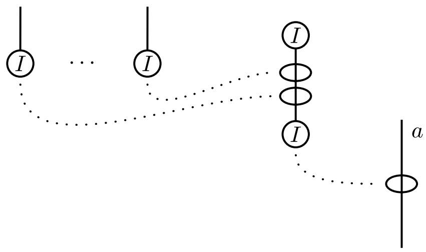
Lemma 8. *Suppose* $f : S \to T$ *and* $g : T \to U$ *are standard* $\mathbb{A}$*-linkings. Let* $\widehat{f}$ *and* $\widehat{g}$ *be normal circuits representing the canonical circuits* f^{c} *and* g^{c}*. Let* $\widehat{f} \cup \widehat{g}$ *be the circuit obtained by pasting* $\widehat{f}$ *and* $\widehat{g}$ *at the* T *wire. There is a strategy of reduction and rewiring of thinning links for* $\widehat{f} \cup \widehat{g}$ *leading to a normal circuit* $\widehat{f} \diamond \widehat{g} : S \to U$ *whose canonical circuit is* $(f; g)^{c}$ *(that of the composite of* f *and* g *in* $\mathsf{L}\mathbb{A}$*).*

Proof. Let f' be the one-sided linking on the two-shape sequent S^*, T corresponding to f, and let g' be the one-sided linking on T^*, U corresponding to g. Let $f' \cup g'$ be the disjoint union of f' and g' on the cut sequent $S^*, T \smile T^*, U$. Let h' be the one-sided linking on S^*, U corresponding to the composite two-sided linking $h = f; g$. Thus h' is the normal form resulting from cut elimination on $f' \cup g'$. We shall mimic the cut elimination steps on $\widehat{f} \cup \widehat{g}$ as reductions mixed with rewiring of thinning links.

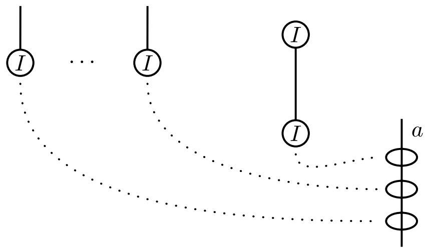
First, perform all $\otimes$ and $(-)^*$ eliminations on $f' \cup g'$. This leaves the same leaf function $f' \cup g'$ on the cut sequent $S^*, a_1 \smile a_1{}^*, \ldots, a_n \smile a_n{}^*, U$ where $a_1, \ldots, a_n$ are the labels of the leaves of T. Since these eliminations affect only the parse trees of the shapes (and not the edges of the leaf function), they can be mimicked directly on $\widehat{f} \cup \widehat{g}$, as the tensor and tensor-unit reductions [3, Section 3.1.1]. Let $\widehat{f_0} \cup \widehat{g_0}$ denote the end result of these reductions.

The normalisation of $f' \cup g'$ on the cut sequent $S^*, a_1 \smile a_1{}^*, \ldots, a_n \smile a_n{}^*, U$ finishes with atomic eliminations. (See the definition of atomic elimination in Section 5.1 for discrete $\mathbb{A}$, and its generalisation to an arbitrary category $\mathbb{A}$ towards the end of Section 5.2.) These atomic eliminations have one of two forms: reduction of (a) a cut pair $I \smile I^*$ or (b) a cut pair $a \smile a^*$ for a an object of $\mathbb{A}$.

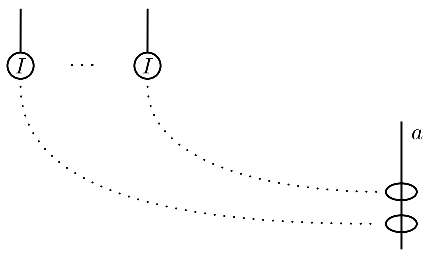
Consider (a). Let $l_1, \ldots, l_n$ be the leaves having an edge to the left I of $I \smile I^*$ and let l be the target of the edge from the right I of $I \smile I^*$. Thus elimination deletes $I \smile I^*$ and moves the edges from l_i to target l. In the circuit we have the corresponding redex:



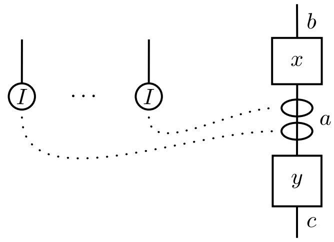
where the two left I-nodes tied by the ellipsis represent n I-nodes corresponding to the leaves $l_1, \ldots, l_n$, the highest I-node corresponds to the left I of $I \smile I^*$, the lowest I-node corresponds to the right I of $I \smile I^*$, and a is the label of l (either $a = I$ or a is an object of $\mathbb{A}$). Rewire the n thinning links on the left to target the a-wire:



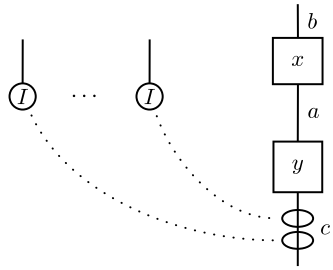
then reduce the I-node redex:



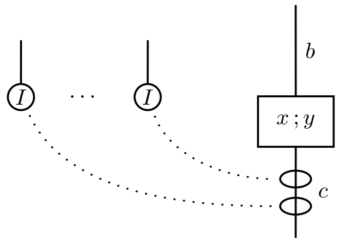
Case (b) is similar. The circuit has a corresponding redex



where $x : b \to a$ and $y : a \to c$ are morphisms in $\mathbb{A}$. Shift the thinning links,



then reduce:



where $x ; y : b \to c$ is the composite of the morphisms $x : b \to a$ and $y : a \to c$ in $\mathbb{A}$.

Since the elimination steps are mimicked precisely, the resulting normal circuit $\widehat{f} \diamond \widehat{g}$, modulo the order of attachments of thinning links along the same wire (i.e., the canonical circuit represented by $\widehat{f} \diamond \widehat{g}$), corresponds to the normal one-sided linking h', hence the composite two-sided linking $h = f ; g$. $\square$

Corollary 2. *Let $f : S \to T$ and $g : T \to U$ be nets in $\mathsf{N}\mathbb{A}$. Then $f^n ; g^n = (f ; g)^n : S \to U$ in $\mathsf{CircNet}\mathbb{A}$.*

Thus the bijection $(\ )^n$ of Proposition 6 preserves composition. It preserves identities because the identity linking and the identity circuit $S \to S$ each amount to a dual pair of copies of the parse tree of S. Therefore $(\ )^n : \mathsf{N}\mathbb{A} \to \mathsf{CircNet}\mathbb{A}$ is functorial, providing an isomorphism of categories $\mathsf{N}\mathbb{A} \cong \mathsf{CircNet}\mathbb{A}$. This completes the proof of Proposition 5, whence the Freeness Theorem: $\mathsf{N}\mathbb{A}$ is the free star-autonomous category generated by $\mathbb{A}$.

Acknowledgements

Many thanks to Robin Cockett and Robert Seely for helping me understand the construction of free star-autonomous categories in [3], an important precursor to this paper. I am extremely grateful to Robin Houston for insightful feedback, in particular for improvements to the definition of *split star-autonomous category*. Thanks to Peter Selinger for corrections. I am indebted to an anonymous referee for excellent suggestions.

References

[1] S. Abramsky, Retracing some paths in process algebra, in: Proc. CONCUR '96, 1996, pp. 1–17.
[2] M. Barr, ∗-Autonomous categories, in: Lecture Notes in Mathematics, vol. 752, Springer-Verlag, 1979.
[3] R.F. Blute, J.R.B. Cockett, R.A.G. Seely, T.H. Trimble, Natural deduction and coherence for weakly distributive categories, J. Pure Appl. Algebra 113 (1996) 229–296.
[4] R.F. Blute, Linear logic, coherence and dinaturality, Theoret. Comput. Sci. 115 (1993) 3–41.
[5] B. Bollobás, Modern Graph Theory, Springer-Verlag, 2002.
[6] J.R.B. Cockett, M. Hasegawa, R.A.G. Seely, Coherence of the double involution on *-autonomous categories, Theory Appl. Categ. 17 (2005) 17–29.
[7] J.R.B. Cockett, R.A.G. Seely, Proof theory for full intuitionistic linear logic, bilinear logic, and mix categories, Theory Appl. Categ. 3 (5) (1997) 85–131.
[8] V. Danos, La logique linéaire appliquée à l'étude de divers processus de normalisation et principalement du lambda calcul, Ph.D. thesis, Univ. de Paris, 1990.

[9] V. Danos, L. Regnier, The structure of multiplicatives, Arch. Math. Logic 28 (1989) 181–203.
[10] S. Eilenberg, G.M. Kelly, A generalization of the functorial calculus, J. Algebra 3 (1966) 366–375.
[11] J.-Y. Girard, Linear logic, Theoret. Comput. Sci. 50 (1987) 1–102.
[12] J.-Y. Girard, Towards a geometry of interaction, in: Categories in Computer Science and Logic, in: Contemporary Mathematics, vol. 92, 1989, pp. 69–108. Proc. of June 1987 meeting in Boulder, Colorado.
[13] J.-Y. Girard, Linear logic: a survey, in: Proceedings of the International Summer School of Marktoberdorf, NATO Advanced Science Institutes, series F94, Also in P. De Groote editor, The Curry-Howard Isomorphism, 193–255, Département de Philosophie, Université Catholique de Louvain, Cahiers du Centre de Logique 8, Academia Press, 1993, pp. 63–112.
[14] J.-Y. Girard, Proof-nets: the parallel syntax for proof theory, in: Logic and Algebra, in: Lecture Notes In Pure and Applied Mathematics, vol. 180, Marcel Dekker, New York, 1996.
[15] J.-Y. Girard, A. Scedrov, P.J. Scott, Bounded linear logic: a modular approach to polynomial time computability, Theoret. Comput. Sci. 97 (1992) 1–66.
[16] S. Guerrini, Correctness of multiplicative proof nets is linear, in: Proc. Logic in Computer Science '99, 1999.
[17] R. Houston, D.J.D. Hughes, A. Schalk, Modelling linear logic without units (draft). http://arxiv.org/abs/math.CT/0504037, April 2005.
[18] D.J.D. Hughes, Simple multiplicative proof nets with units, Ann. Pure Appl. Logic (2012) http://arxiv.org/abs/math.CT/0507003.
[19] A. Joyal, R. Street, The geometry of tensor calculus I, Adv. Math. 88 (1991) 55–112.
[20] A. Joyal, R. Street, Braided tensor categories, Adv. Math. 102 (1) (1993) 20–78.
[21] A. Joyal, R. Street, D. Verity, Traced monoidal categories, Math. Proc. Cambridge Philos. Soc. 119 (1996) 447–468.
[22] G.M. Kelly, On Mac Lane's conditions for coherence of natural associativities, commutativities, etc., J. Algebra 1 (1964) 397–402.
[23] G.M. Kelly, M.L. Laplaza, Coherence for compact closed categories, J. Pure Appl. Algebra 19 (1980) 193–213.
[24] G.M. Kelly, S. Mac Lane, Coherence in closed categories, J. Pure Appl. Algebra 1 (1971) 97–140.
[25] T.W. Koh, C.-H.L. Ong, Explicit substitution internal languages for autonomous and *-autonomous categories (preliminary version), in: Proceedings of the 8th Conference on Category Theory and Computer Science, Edinburgh, September 1999, in: Electronic Notes in Theoretical Computer Science, vol. 29, 1999.
[26] F. Lamarche, Proof nets for intuitionistic linear logic I: essential nets. Unpublished note, 1994.
[27] F. Lamarche, L. Straßburger, On proof nets for multiplicative linear logic with units, in: Proc. CSL'04, in: Lecture Notes in Computer Science, vol. 3210, 2004, pp. 145–159.
[28] S. Mac Lane, Categories for the Working Mathematician, Springer-Verlag, 1971.
[29] A.S. Murawski, C.-H.L. Ong, Dominator trees and fast verification of proof nets, in: Proc. Logic in Computer Science '00, 2000.
[30] A.S. Murawski, C.-H.L. Ong, Exhausting strategies, joker games and full completeness for IMLL with unit, Theoret. Comput. Sci. 294 (2003) 269–305.
[31] M. Parigot, $\lambda\mu$-calculus: an algorithmic interpretation of classical natural deduction, in: Proc. Int. Conf. Logic Prog. Automated Reasoning, in: Lecture Notes in Computer Science, vol. 624, Springer, 1992, pp. 190–201.
[32] L. Regnier, Lambda-Calcul et Rèseaux, Ph.D. thesis, Univ. Paris VII, 1992.
[33] T.H. Trimble, Linear logic, bimodules, and full coherence for autonomous categories, Ph.D. thesis, Rutgers University, 1994.