Categories for computation in context and unified logic: the "intuitionist" case

R.F. Blute[1] J.R.B. Cockett[2] R.A.G. Seely[3]

Presented to Peter Freyd
to mark the occasion of
his 60^{th} birthday

Abstract

In this paper we introduce the notion of *contextual categories*. These provide a categorical semantics for the modelling of computation in context, based on the idea of separating logical sequents into two zones, one representing the context over which the computation is occurring, the other the computation itself. The separation into zones is achieved *via* a bifunctor equipped with a tensorial strength. We show that a category with such a functor can be viewed as having an action on itself. With this interpretation, we obtain a fibration in which the base category consists of contexts, and the reindexing functors are used to change the context.

We further observe that this structure also provides a framework for developing categorical semantics for Girard's Unified Logic, a key feature of which is to separate logical sequents into two zones, one in which formulas behave classically and one in which they behave linearly. This separation is analogous to the context/computation separation above, and is handled by our semantics in a similar fashion. Furthermore, our approach allows an analysis of the exponential structure of linear logic using a tensorially strong action as the primitive notion. We demonstrate that from such a structure one can recover a model of the linear storage operator.

Finally, we introduce a sequent calculus for the fragment of Unified Logic modeled by contextual categories. We show cut elimination for this fragment, and we introduce a simple notion of proof circuit, which provides a description of free contextual categories.

1 Introduction

THIS document has its roots in the attempt to elucidate the structure of linear logic using weakly distributive categories [CS91]. In that programme we started by investigating the categorical proof theory of the (two-sided) linear cut rules, giving rise to the notion of weakly distributive categories. Our rationale was that by so doing we could better

[1]Department of Mathematics, University of Ottawa, 585 King Edward St., Ottawa, ON, K1N 6N5, Canada. rblute@mathstat.uottawa.ca Research partially supported by Le Fonds FCAR, Québec, and NSERC, Canada.

[2]Department of Computer Science, University of Calgary, 2500 University Drive, Calgary, AL, T2N 1N4, Canada. robin@cpsc.ucalgary.ca Research partially supported by NSERC, Canada.

[3]Department of Mathematics, McGill University, 805 Sherbrooke St., Montréal, PQ, H3A 2K6, Canada. rags@math.mcgill.ca Research partially supported by Le Fonds FCAR, Québec, and NSERC, Canada.

modularize the structure of linear logic which would facilitate the establishment of coherence theorems [BCST]. It was a natural step to ask whether the exponentials, **!** and **?**, could be added in a modular fashion to this basic setting [BCS92]. When we started to study this question, we were struck by the prominent role of tensorial strength (and costrength) in the formulation. This led us to consider whether it might be possible to use strength as a more primitive notion in the modular decomposition of these settings and resulted in the development of our notion of contextual categories[1].

The coherence conditions (over twenty commuting diagrams) for a setting in which strength is taken as primitive are quite daunting. Indeed, initially, as we lacked any motivating models, we were concerned that these "contextual categories" would simply foist a needless and rather complex abstraction on the community. However, two connections persuaded us that this basic setting was worthwhile.

The first connection, which we begin to explore in this paper, is the similarity of the system to Girard's approach to amalgamate classical and linear logic into one setting. In a linear sequent calculus it is natural to model context by dividing the terms to the left of the turnstile into a "classical" portion followed by a "linear" portion. Contextual categories are the proof theory of this fragment of Girard's "unified logic" [G93] and, indeed, we are convinced this is the basic building block of the categorical proof theory for this logic.

The second connection, which we do not explore in this paper but rather leave to its sequel, is to the "Action Calculi" of Robin Milner. Strength is of course a pervasive notion in computer science. The view of a function (or program) as a map between two objects in a given context is absolutely fundamental to computing. The categorical machinery for handling context in this sense is, specifically, strength and, in generality, fibration. That the former gives rise to the latter in the classical setting is known (this is detailed in [Co91]). However, less well-explored is the notion of strength as applied to linear settings and this becomes quite important when reasoning about communicating processes. The fact that some processes are limited-resource or "threaded" leads naturally to treating them in a linear fashion. It was partly to accommodate these features that Milner developed his "Action Calculi" [M93]. These provide another source of examples for contextual categories.

The development actually starts by describing the general notion of strength and how it gives rise to a fibration. The aim is to remind the reader of the correspondence that the various notions of strong functor, strong natural transformation *etc.* have to their fibrational counterparts. Formally, there is a full 2–embedding of these strong categories into (structured) fibrations. We briefly discuss datatypes and the properties one should expect of them in the presence of context.

Next we study contextual categories: these are categories with a natural contextual self-action. Much of the discussion is concerned with establishing the coherence conditions both for contextual categories and for their actions (which we call contextual modules). Of some interest is the fact that the empty context $\top$ induces a "storage" cotriple which turns an object X into a context for $\top$: $X \mapsto X \oslash \top$.

Next we describe the fragment of unified logic, called the context calculus, for which these contextual categories are the proof theory. The categorical coherence diagrams may be seen as arising from those manipulations which are necessary to establish the cut elimination theorem. We close by describing the proof circuits of this logic which provide a usable and very economical view of the whole calculus.

[1] We must point out that this term has been used in a completely different sense by J. Cartmell [Ca86].

In a sequel we plan to extend this study to include the dual notion of cocontext. In a weakly distributive setting the underlying duality between the tensor ($\otimes$) and the par ($\oplus$) should extend to the situation where one includes context. Thus, there should be a dual notion of cocontext. One way to view this is as the requirement that the corresponding sequent calculus not only has a classical and linear portion to the left of the turnstile but also to the right. Categorically this gives a cofibration over the cocontexts and gives rise to a *fibrational fork* in which context and cocontext interact. In the setting of a fibrational fork the datatypes must, of course, preserve both structures. In addition, to be able to handle at least the tensor-par fragment of linear logic, we must add those connectives to the context calculus as well. The interaction between the classical and linear portions of sequents then begins to develop some of the complexity we expect from linear logic and from action calculi.

2 Strong contexts and fibrations

This section starts by introducing the notion of strength through the notion of a functorial action. Next strong contexts are described. These, it is then shown, can alternatively be formulated as a fibration.

Throughout this development we shall talk about "computations in context." This terminology is not meant to prejudice the generality of the theory but rather to lend a certain intuition to the proceedings.

2.1 Functorial actions and strength

Let $\mathbf{X}$ be a category then $\mathbf{X}$ is said to have an **action** on a category $\mathbf{Y}$ if there is a bifunctor[2]

$$\oslash : \mathbf{X} \times \mathbf{Y} \longrightarrow \mathbf{Y}$$

A **strong functor** $F : \mathbf{Y} \longrightarrow \mathbf{Y}'$ between two categories with an $\mathbf{X}$–action is a functor together with a natural transformation

$$\theta : X \oslash F(Y) \longrightarrow F(X \oslash Y)$$

called a **strength**. A **strong natural transformation** between strong functors is an ordinary natural transformation $\alpha : F_1 \longrightarrow F_2$ which satisfies the additional property:

$$\begin{tikzcd}
X \oslash F_1(Y) \arrow[r, "1 \oslash \alpha"] \arrow[d, "\theta_1"'] & X \oslash F_2(Y) \arrow[d, "\theta_2"] \\
F_1(X \oslash Y) \arrow[r, "\alpha"'] & F_2(X \oslash Y)
\end{tikzcd}$$

We shall often call these strong transformations. It is almost a formality to observe that:

Proposition 2.1 *Categories with an $\mathbf{X}$–action, strong functors, and strong natural transformations form a 2–category $\mathbf{Act}(\mathbf{X})$.*

The only ingredient in this which needs a comment is the manner of composing the strengths of strong functors:

$$(F, \theta_F) \circ (G, \theta_G) = (F \circ G, \theta_G; G(\theta_F)).$$

[2]In this paper, by "bifunctor" we just mean a functor whose domain is a product of categories.

2.2 Structural actions

An action $\_ \oslash \_$ is **structural** if the functor $X \oslash \_$ is a cotriple for each object X; explicitly this means that there are in addition the following natural transformations, called **duplication** and **elimination**:

$$
\begin{aligned}
d &: X \oslash Y \longrightarrow X \oslash (X \oslash Y) \\
e &: X \oslash Y \longrightarrow Y
\end{aligned}
$$

where these data satisfy:

$$
\begin{tikzcd}
X \oslash Y & X \oslash (X \oslash Y) \arrow[l, "e"'] \arrow[r, "1 \oslash e"] & X \oslash Y \\
& X \oslash Y \arrow[ul, equal] \arrow[u, "d"] \arrow[ur, equal] &
\end{tikzcd}
$$

$$
\begin{tikzcd}
X \oslash Y \arrow[r, "d"] \arrow[d, "d"'] & X \oslash (X \oslash Y) \arrow[d, "d"] \\
X \oslash (X \oslash Y) \arrow[r, "1 \oslash d"'] & X \oslash (X \oslash (X \oslash Y))
\end{tikzcd}
\qquad
\begin{tikzcd}
X \oslash (Y \oslash Z) \arrow[r, "1 \oslash e"] \arrow[d, "e"'] & X \oslash Z \arrow[d, "e"] \\
Y \oslash Z \arrow[r, "e"'] & Z
\end{tikzcd}
$$

or equationally:

$$
\begin{aligned}
d; e &= 1 && (1) \\
d; 1 \oslash e &= 1 && (2) \\
d; d &= d; 1 \oslash d && (3) \\
e; e &= 1 \oslash e; e && (4)
\end{aligned}
$$

This last condition is a consequence of naturality, but it is worth making it explicit here.

Next we extend the definitions of strong functors and transformations to the structural case. In fact, the strong transformations are unchanged from above, but strong structural functors must also preserve the duplication and elimination structure. Specifically this means that the following diagrams must commute:

Elimination strength:

$$
\begin{tikzcd}
X \oslash F(Y) \arrow[r, "\theta"] \arrow[dr, "e"'] & F(X \oslash Y) \arrow[d, "F(e)"] \\
& F(Y)
\end{tikzcd}
$$

Duplication strength:

$$\begin{tikzcd} X \oslash F(Y) \arrow[r, "d"] \arrow[d, "\theta"'] & X \oslash (X \oslash F(Y)) \arrow[d, "1 \oslash \theta;\theta"] \\ F(X \oslash Y) \arrow[r, "F(d)"'] & F(X \oslash (X \oslash Y)) \end{tikzcd}$$

Once again it is a formality to show that

Proposition 2.2 *Categories with an* $\mathbf{X}$*–structural action, strong structural functors, and strong natural transformations form a 2–category* $\mathsf{StrAct}(\mathbf{X})$.

2.3 Fibrations and structural actions

We suppose the reader is familiar with the basic notions concerning fibrations and indexed categories. See for example [Bo94].

An $\mathbf{X}$–structural action on $\mathbf{Y}$ gives rise to an indexed category over $\mathbf{X}$ and thus a fibration over $\mathbf{X}$. The fiber over an object X is to be thought of as the $\mathbf{Y}$–computations in the context X. Thus, maps in the fiber are

$$f : X \oslash Y \longrightarrow Y' \text{ and } \ g : X \oslash Y' \longrightarrow Y''$$

with composition given by

$$d; 1 \oslash f; g : X \oslash Y \longrightarrow Y''$$

and identities given by

$$e : X \oslash Y \longrightarrow Y.$$

This is of course the Kleisli category of the cotriple $X \oslash \_$. Functors between these categories of computations in context are provided by precomposing with the change in context. The functorial nature of this change in context follows immediately from the naturality of the duplication and the elimination transformation.

Alternatively we may form the total category $\mathcal{C}_{\mathbf{X}}(\mathbf{Y})$ of computations and contexts and show that the functor to $\mathbf{X}$ is a fibration. The objects of the total category are (following the Grothendieck construction) pairs (X, Y) where $X \in \mathbf{X}$ and $Y \in \mathbf{Y}$, the maps are pairs $(h, f) : (X, Y) \longrightarrow (X', Y')$ where $h : X \longrightarrow X'$ and $f : X \oslash Y \longrightarrow Y'$ is a computation in the context X.

Slightly more surprising is the fact that strong functors give rise to morphisms of fibrations and strong transformations to transformations of fibrations. The passage between the two is as follows:

- *Strong functor to morphism of fibration:* Given a strong functor $F : \mathbf{Y} \longrightarrow \mathbf{Y}'$ we may define a family of functors between $\mathbf{Y}$–computations in context X and $\mathbf{Y}'$–computations in context X. Let $f : X \oslash Y \longrightarrow Y'$ then define $F_X(f) : X \oslash F(Y) \longrightarrow F(Y')$ to be

  $$X \oslash F(Y) \xrightarrow{\theta} F(X \oslash Y) \xrightarrow{F(f)} F(Y').$$

  It is straightforward to check that these are functors and they commute with the functors which change context.

- *Strong transformation to fibrational transformation:* Given a strong transformation $\alpha : F \longrightarrow G$ we have a transformation $\alpha_X : F_X \longrightarrow G_X$ defined in the obvious way by $e; \alpha : X \oslash F(Y) \longrightarrow G(Y)$.

It is clear that this gives a faithful 2-functor from the 2–category $\mathsf{StrAct}(\mathbf{X})$ to the 2–category of fibrations over $\mathbf{X}$. Our aim is now to supply a full and faithful 2–functor to a 2–category which we know to have all weighted limits. (We shall see in the next section that $\mathsf{StrAct}(\mathbf{X})$ does not have this property.) In order to achieve this we construct the comma 2–category between the following 2–functor and the identity:

$$\mathrm{Const} : \mathsf{Cat} \longrightarrow \mathsf{Fib}(\mathbf{X}); \mathbf{Y} \mapsto [\mathrm{Pr}_1 : \mathbf{Y} \times \mathbf{X} \longrightarrow \mathbf{X}].$$

An object of this 2-category $\mathsf{ConstFib}(\mathbf{X})$ is a triple:

$$(\mathbf{Y}, M : \mathrm{Pr}_1 \longrightarrow F, F)$$

where F is a fibration over $\mathbf{X}$ and M is a morphism of fibrations from the constant fibration at $\mathbf{Y}$.

Theorem 2.3 *For any category* $\mathbf{X}$ *there is a full faithful 2–embedding of* $\mathcal{V} : \mathsf{StrAct}(\mathbf{X}) \longrightarrow \mathsf{ConstFib}(\mathbf{X})$.

First we note that the functor

$$K : \mathbf{Y} \times \mathbf{X} \longrightarrow \mathcal{C}_{\mathbf{X}}(\mathbf{Y}) : (f, x) \mapsto (e; f, x)$$

which is the identity on objects but sends maps to those which do not use context is a morphism of fibrations. This is a 2–functorial assignment. We must show that it is full and faithful. To this end we show how from morphism of fibration and transformations of fibrations we can recapture their strong counterparts:

- *Morphism of fibration to strong functor:* Given a 1–cell in $\mathsf{ConstFib}(\mathbf{X})$ between morphisms of fibrations given by contextual actions, part of this data is a functor $F : \mathbf{Y} \longrightarrow \mathbf{Y}'$. It suffices to show that this is a strong functor. The strength at X for this functor is provided by the identity map on $X \oslash Y$ as seen in the fiber over X. Here the map $1 : X \oslash Y \longrightarrow (X \oslash Y)$ is not the identity and under F turns into a map $\theta : X \oslash F(Y) \longrightarrow F(X \oslash Y)$ which is the strength of F.

- *Fibrational transformation to strong transformation:* A transformation in $\mathsf{ConstFib}(\mathbf{X})$ has as part of its data an ordinary transformation. It suffices to show that it is strong which is a consequence of the fact that the transformation of the total category is natural at the particular maps which provide the strength.

2.4 2–limits and datatypes

Given two $\mathbf{X}$–structural actions $\mathbf{Y}$ and $\mathbf{Y}'$ their product is an $\mathbf{X}$–structural action given by:

$$X \oslash (Y, Y') = (X \oslash Y, X \oslash Y').$$

Thus the 2–category of $\mathsf{StrAct}(\mathbf{X})$ has finite products.

Also $\mathsf{StrAct}(\mathbf{X})$ has arrow categories, and so is cartesian closed. It is the ordinary arrow category with the tensor action

$$X \oslash (Y \xrightarrow{f} Y') = X \oslash Y \xrightarrow{1 \oslash f} X \oslash Y'.$$

The strength natural transformation for a functor into this arrow category gives a commutative square which expresses the strength of the natural transformation.

$\mathsf{StrAct}(\mathbf{X})$ does not have equalizers. If H is the equalizing functor of two functors of this 2–category $F, G : \mathbf{Y} \longrightarrow \mathbf{Y}'$ then certainly $H \circ F = H \circ G$ as functors. However, taking the equalizer as functors will not do as this equalizer category need not be closed to the $\mathbf{X}$–action.

Not only must H equalize the functors but also the composite must agree on the strengths and it is possible that H have a non-trivial strength (consider the pullback of the identity functor with an arbitrary functor expressed as an equalizer). This means that the equation

$$\theta_F; F(\theta_H) = \theta_G; G(\theta_H)$$

must hold. This does not hold in general with a structural action.

Thus, the 2–category of $\mathsf{StrAct}(\mathbf{X})$ does not have all weighted limits.

By contrast, of course, the 2–category of fibrations (with morphisms preserving cleavage) certainly has equalizers and preserves the products and arrow categories of contextual actions under the embedding. This is also true of the underlying functor to Cat. This means the full 2-embedding to $\mathsf{ConstFib}(\mathbf{X})$ preserves these limits. This allows us to regard this 2–category as a "completion" of the latter in which equalizers exist.

For the discussion of datatypes, inserters are needed [Co91] and so it is pragmatic to work in the 2–category $\mathsf{ConstFib}(\mathbf{X})$ to determine the form datatypes take in this setting. When one unwinds the various diagrams involved for a contextual action, the universal diagram which must be satisfied by a linear natural number object, for example, takes the following form:

$$\begin{tikzcd}
X \oslash 1 \arrow[r, "1 \oslash 0"] \arrow[dr, "f"'] & X \oslash \mathcal{N} \arrow[d, dotted, "\lambda"] & X \oslash \mathcal{N} \arrow[l, "1 \oslash s"'] \arrow[d, dotted, "d; 1 \oslash \lambda"] \\
 & C & X \oslash C \arrow[l, "g"]
\end{tikzcd}$$

Note how it differs from the form suggested by Paré and Román [PR89] as the context appears not only on the top line but also on the bottom where the properties of being a context become crucial.

Perhaps the most important single inductive datatype is the list datatype. To construct lists a strong tensor product in $\mathbf{Y}$ is needed: the universal diagram for lists will then be:

$$\begin{tikzcd}
X \oslash \top \arrow[r, "1 \oslash \mathrm{nil}"] \arrow[dr, "f"'] & X \oslash \mathrm{list}(A) \arrow[d, dotted, "k"'] & X \oslash (A \otimes \mathrm{list}(A)) \arrow[l, "1 \oslash \mathrm{cons}"'] \arrow[d, dotted, "d; 1 \oslash (\theta_{A \otimes _}; 1 \otimes k)"] \\
 & C & X \oslash C \arrow[l, "g"]
\end{tikzcd}$$

Similarly one may define diagrams for other inductive datatypes. As datatypes have hardly been studied in these settings it is of no small interest to wonder what properties they satisfy: an area we barely broach here!

3 Contextual Categories

A situation of particular interest arises when a category $\mathbf{X}$ has a structural action on itself which is, furthermore, strong with respect to itself. Such categories are essentially the subject matter of this section. To be reasonable, the strength transformations must satisfy various coherence conditions. When these are written out they are very similar to the conditions governing a symmetric tensor product. There are, however, two major differences: first the associativity map is not an isomorphism, and second the unit (called an empty context) does not have an identity action on either side. This, of course makes it necessary to write down explicitly many coherence diagrams which would otherwise be implied.

A category with a strong contextual action on itself has a natural transformation:

$$\theta_{\_\oslash\_} : X \oslash (Y \oslash Z) \longrightarrow (X \oslash Y) \oslash (X \oslash Z)$$

such that $e = \theta_{\_\oslash\_}; e \oslash e$ and $d; 1 \oslash \theta_{\_\oslash\_}; \theta_{\_\oslash\_} = \theta_{\_\oslash\_}; d \oslash d$. Rather than dealing with the strength of $\theta_{\_\oslash\_}$ we may work with the *linear* strengths in each argument:

$$a_\oslash = \theta_{\_\oslash\_}; 1 \oslash e : X \oslash (Y \oslash Z) \longrightarrow (X \oslash Y) \oslash Z$$

to be thought of as an associativity map and

$$c_\oslash = \theta_{\_\oslash\_}; e \oslash 1 : X \oslash (Y \oslash Z) \longrightarrow Y \oslash (X \oslash Z)$$

to be thought of as a symmetry map. We note that $\theta_{\_\oslash\_}$ can then be reconstituted as:

$$\theta_{\_\oslash\_} = d; 1 \oslash a_\oslash; c_\oslash = d; 1 \oslash c_\oslash; a_\oslash.$$

Thus, in axiomatizing such a setting we may organize the axioms around the linear strengths of associativity and symmetry. Our first major test of the axiomatization will be to recover the strength of $\theta_{\_\oslash\_}$ from its linear strengths (see 3.4).

3.1 The definition

A **contextual category** is a category equipped with a structural action

$$\_ \oslash \_ : \mathbf{X} \times \mathbf{X} \longrightarrow \mathbf{X}$$

and an **empty context** $\top$ and natural transformations:

$$\begin{array}{rcl} a_\oslash & : & X \oslash (Y \oslash Z) \longrightarrow (X \oslash Y) \oslash Z \\ c_\oslash & : & X \oslash (Y \oslash Z) \longrightarrow Y \oslash (X \oslash Z) \\ \text{lift} & : & X \longrightarrow \top \oslash X \\ \text{read} & : & X \oslash \top \longrightarrow X \end{array}$$

Satisfying a number of coherence diagrams. These we shall organize according to their source:

Empty context: (Written so as to suggest the tensorial counterparts)

$$\begin{tikzcd}
 & \top \oslash \top \arrow[dr, "\mathrm{read}"] & \\
\top \arrow[ur, "\mathrm{lift}"] \arrow[rr, equal] & & \top
\end{tikzcd}
\qquad
\begin{tikzcd}
X \oslash (\top \oslash Y) \arrow[rrr, "a_{\oslash}"] & & & (X \oslash \top) \oslash Y \arrow[dl, "\mathrm{read} \oslash 1"] \\
 & X \oslash Y \arrow[ul, "1 \oslash \mathrm{lift}"'] \arrow[r, equal] & X \oslash Y &
\end{tikzcd}$$

$$\mathrm{lift}; \mathrm{read} = 1 \tag{5}$$

$$1 \oslash \mathrm{lift}; a_{\oslash}; \mathrm{read} \oslash 1 = 1 \tag{6}$$

Symmetry:

$$\begin{tikzcd}
X \oslash (Y \oslash Z) \arrow[rr, equal] \arrow[dr, "c_{\oslash}"'] & & X \oslash (Y \oslash Z) \\
 & Y \oslash (X \oslash Z) \arrow[ur, "c_{\oslash}"'] &
\end{tikzcd}
\qquad
\begin{tikzcd}
 & X \oslash Z \arrow[dl, "d"'] \arrow[dr, "d"] & \\
X \oslash (X \oslash Z) \arrow[rr, "c_{\oslash}"'] & & X \oslash (X \oslash Z)
\end{tikzcd}$$

$$c_{\oslash}; c_{\oslash} = 1 \tag{7}$$

$$d; c_{\oslash} = d \tag{8}$$

Elimination, duplication and lifting:

$$\begin{tikzcd}
X \arrow[r, "\mathrm{lift}"] \arrow[d, "\mathrm{lift}"'] & \top \oslash X \arrow[d, "\mathrm{lift}"] \\
\top \oslash X \arrow[r, "d"'] & \top \oslash (\top \oslash X)
\end{tikzcd}
\qquad
\begin{tikzcd}
X \arrow[r, "\mathrm{lift}"] \arrow[dr, equal] & \top \oslash X \arrow[d, "e"] \\
 & X
\end{tikzcd}$$

$$\mathrm{lift}; d = \mathrm{lift}; \mathrm{lift} \tag{9}$$

$$1 = \mathrm{lift}; e \tag{10}$$

Transpositions:

$$\begin{tikzcd}
X \oslash (Y \oslash Z) \arrow[r, "1 \oslash d; d"] \arrow[d, "a_\oslash"'] & X \oslash (X \oslash (Y \oslash (Y \oslash Z))) \arrow[r, "1 \oslash c_\oslash"] & X \oslash (Y \oslash (X \oslash (Y \oslash Z))) \arrow[d, "a_\oslash; 1 \oslash a_\oslash"] \\
(X \oslash Y) \oslash Z \arrow[rr, "d"'] & & (X \oslash Y) \oslash ((X \oslash Y) \oslash Z)
\end{tikzcd}$$

$$\begin{tikzcd}
A \oslash (B \oslash Z) \arrow[r, "1 \oslash d; d"] \arrow[d, "c_\oslash"'] & A \oslash (A \oslash (B \oslash (B \oslash Z))) \arrow[r, "1 \oslash c_\oslash"] & A \oslash (B \oslash (A \oslash (B \oslash Z))) \arrow[d, "c_\oslash; 1 \oslash (1 \oslash c_\oslash)"] \\
B \oslash (A \oslash Z) \arrow[r, "1 \oslash d; d"'] & B \oslash (B \oslash (A \oslash (A \oslash Z))) \arrow[r, "1 \oslash c_\oslash"'] & B \oslash (A \oslash (B \oslash (A \oslash Z)))
\end{tikzcd}$$

$$a_\oslash; d = 1 \oslash d; d; 1 \oslash c_\oslash; a_\oslash; 1 \oslash a_\oslash \tag{11}$$

$$c_\oslash; 1 \oslash d; d; 1 \oslash c_\oslash = 1 \oslash d; d; 1 \oslash c_\oslash; c_\oslash; 1 \oslash (1 \oslash c_\oslash) \tag{12}$$

Bistrengths of the action:

$$\begin{tikzcd}
X \oslash (Y \oslash (A \oslash B)) \arrow[rr, "1 \oslash c_\oslash"] \arrow[d, "c_\oslash"'] & & X \oslash (A \oslash (Y \oslash B)) \arrow[d, "a_\oslash"] \\
Y \oslash (X \oslash (A \oslash B)) \arrow[r, "1 \oslash a_\oslash"'] & Y \oslash ((X \oslash A) \oslash B) \arrow[r, "c_\oslash"'] & (X \oslash A) \oslash (Y \oslash B)
\end{tikzcd}$$

This ensures that $\theta_{\_\oslash\_}$ is a contextual strength (see 3.4).

$$c_\oslash; 1 \oslash a_\oslash; c_\oslash = 1 \oslash c_\oslash; a_\oslash \tag{13}$$

Strong transformations:

$$\begin{tikzcd} X \oslash Y \arrow[r, "1\oslash \mathrm{lift}"] \arrow[dr, "\mathrm{lift}"'] & X \oslash (\top \oslash Y) \arrow[d, "c_\oslash"] \\ & \top \oslash (X \oslash Y) \end{tikzcd} \qquad \begin{tikzcd} X \oslash (Y \oslash \top) \arrow[r, "1\oslash \mathrm{read}"] \arrow[d, "a_\oslash"'] & X \oslash Y \\ (X \oslash Y) \oslash \top \arrow[ur, "\mathrm{read}"'] & \end{tikzcd}$$

$$\begin{aligned} 1 \oslash \mathrm{lift}; c_\oslash &= \mathrm{lift} && (14) \\ a_\oslash; \mathrm{read} &= 1 \oslash \mathrm{read} && (15) \end{aligned}$$

Elimination of strengths:

$$\begin{tikzcd} X \oslash (Y \oslash Z) \arrow[r, "a_\oslash"] \arrow[dr, "e"'] & (X \oslash Y) \oslash Z \arrow[d, "e \oslash 1"] \\ & Y \oslash Z \end{tikzcd} \qquad \begin{tikzcd} X \oslash (Y \oslash Z) \arrow[r, "c_\oslash"] \arrow[dr, "e"'] & Y \oslash (X \oslash Z) \arrow[d, "1 \oslash e"] \\ & Y \oslash Z \end{tikzcd}$$

$$\begin{aligned} a_\oslash; e \oslash 1 &= e && (16) \\ c_\oslash; 1 \oslash e &= e && (17) \end{aligned}$$

Duplication of strengths:

$$\begin{tikzcd} X \oslash (Y \oslash Z) \arrow[r, "d"] \arrow[d, "a_\oslash"'] & X \oslash (X \oslash (Y \oslash Z)) \arrow[d, "1 \oslash a_\oslash; a_\oslash"] \\ (X \oslash Y) \oslash Z \arrow[r, "d \oslash 1"'] & (X \oslash (X \oslash Y)) \oslash Z \end{tikzcd} \qquad \begin{tikzcd} X \oslash (Y \oslash Z) \arrow[r, "d"] \arrow[d, "c_\oslash"'] & X \oslash (X \oslash (Y \oslash Z)) \arrow[d, "1 \oslash c_\oslash; c_\oslash"] \\ Y \oslash (X \oslash Z) \arrow[r, "1 \oslash d"'] & Y \oslash (X \oslash (X \oslash Z)) \end{tikzcd}$$

$$\begin{aligned} d; 1 \oslash a_\oslash; a_\oslash &= a_\oslash; d \oslash 1 && (18) \\ d; 1 \oslash c_\oslash; c_\oslash &= c_\oslash; 1 \oslash d && (19) \end{aligned}$$

Associativity of strengths:

$$\begin{tikzcd}
X \oslash (Y \oslash (Z \oslash W)) \arrow[rr, "a_\oslash"] \arrow[d, "1 \oslash a_\oslash"'] & & (X \oslash Y) \oslash (Z \oslash W) \arrow[d, "a_\oslash"] \\
X \oslash ((Y \oslash Z) \oslash W) \arrow[r, "a_\oslash"'] & (X \oslash (Y \oslash Z)) \oslash W \arrow[r, "a_\oslash \oslash 1"'] & ((X \oslash Y) \oslash Z) \oslash W
\end{tikzcd}$$

Making $a_\oslash$ "associative" (although not invertible).

$$\begin{tikzcd}
X \oslash (Y \oslash (Z \oslash W)) \arrow[r, "c_\oslash"] \arrow[d, "1 \oslash a_\oslash"'] & Y \oslash (X \oslash (Z \oslash W)) \arrow[r, "1 \oslash a_\oslash"] & Y \oslash ((X \oslash Z) \oslash W)) \arrow[d, "a_\oslash"] \\
X \oslash ((Y \oslash Z) \oslash W) \arrow[r, "a_\oslash"'] & (X \oslash (Y \oslash Z)) \oslash W \arrow[r, "c_\oslash \oslash 1"'] & (Y \oslash (X \oslash Z)) \oslash W
\end{tikzcd}$$

$$\begin{aligned}
1 \oslash a_\oslash; a_\oslash; a_\oslash \oslash 1 &= a_\oslash; a_\oslash && (20)\\
1 \oslash a_\oslash; a_\oslash; c_\oslash \oslash 1 &= c_\oslash; 1 \oslash a_\oslash; a_\oslash && (21)
\end{aligned}$$

Symmetry of strengths:

$$\begin{tikzcd}
X \oslash (Y \oslash (Z \oslash W)) \arrow[r, "c_\oslash"] \arrow[d, "1 \oslash a_\oslash"'] & Y \oslash (X \oslash (Z \oslash W)) \arrow[r, "1 \oslash c_\oslash"] & Y \oslash (Z \oslash (X \oslash W)) \arrow[d, "a_\oslash"] \\
X \oslash ((Y \oslash Z) \oslash W) \arrow[rr, "c_\oslash"'] & & (Y \oslash Z) \oslash (X \oslash W)
\end{tikzcd}$$

$$\begin{tikzcd}
X \oslash (Y \oslash (V \oslash Z)) \arrow[r, "1 \oslash c_\oslash"] \arrow[d, "c_\oslash"'] & X \oslash (V \oslash (Y \oslash Z)) \arrow[r, "c_\oslash"] & V \oslash (X \oslash (Y \oslash Z)) \arrow[d, "1 \oslash c_\oslash"] \\
Y \oslash (X \oslash (V \oslash Z)) \arrow[r, "1 \oslash c_\oslash"'] & Y \oslash (V \oslash (X \oslash Z)) \arrow[r, "c_\oslash"'] & V \oslash (Y \oslash (X \oslash Z))
\end{tikzcd}$$

$$1 \oslash a_\oslash ; c_\oslash \quad = \quad c_\oslash ; 1 \oslash c_\oslash ; a_\oslash \tag{22}$$
$$1 \oslash c_\oslash ; c_\oslash ; 1 \oslash c_\oslash \quad = \quad c_\oslash ; 1 \oslash c_\oslash ; c_\oslash \tag{23}$$

Lifting of strengths:

$$\begin{tikzcd} X \oslash Y \arrow[r, "\mathrm{lift}"] \arrow[dr, "\mathrm{lift} \oslash 1"'] & \top \oslash (X \oslash Y) \arrow[d, "a_\oslash"] \\ & (\top \oslash X) \oslash Y \end{tikzcd} \qquad \begin{tikzcd} X \oslash Y \arrow[r, "\mathrm{lift}"] \arrow[dr, "1 \oslash \mathrm{lift}"'] & \top \oslash (X \oslash Y) \arrow[d, "c_\oslash"] \\ & X \oslash (\top \oslash Y) \end{tikzcd}$$

Note that the second of these diagrams has already occurred to ensure the strong naturality for lift.

$$\mathrm{lift} ; a_\oslash \quad = \quad \mathrm{lift} \oslash 1 \tag{24}$$

3.2 Contextual modules

It is reasonable to ask what a contextual action of $\mathbf{X}$ must look like. We shall call such actions **X–modules**: a module is an action, as before, but is equipped not only with elimination and duplication but also a lifting, associativity, and symmetry map. The diagrams which must be satisfied are all the diagrams above less (5) and (15) (which would demand that $\top$ is in the module).

The strong functors between modules must now preserve the additional structure we have introduced. This gives three further diagrams (that is including elimination and duplication strength) to be satisfied by the strength transformation of the functors:

Associative strength:

$$\begin{tikzcd} X \oslash (Y \oslash F(Z)) \arrow[r, "1 \oslash \theta_F ; \theta_F"] \arrow[d, "a_\oslash"'] & F(X \oslash (Y \oslash Z)) \arrow[d, "F(a_\oslash)"] \\ (X \oslash Y) \oslash F(Z) \arrow[r, "\theta_F"'] & F((X \oslash Y) \oslash Z) \end{tikzcd}$$

Symmetric strength:

$$\begin{tikzcd}
X \oslash (Y \oslash F(Z)) \arrow[r, "1 \oslash \theta_F; \theta_F"] \arrow[d, "c_\oslash"'] & F(X \oslash (Y \oslash Z)) \arrow[d, "F(c_\oslash)"'] \\
(Y \oslash (X \oslash F(Z)) \arrow[r, "1 \oslash \theta_F; \theta_F"'] & F(Y \oslash (X \oslash Z))
\end{tikzcd}$$

Lifting strength:

$$\begin{tikzcd}
F(X) \arrow[r, "\mathrm{lift}"] \arrow[dr, "F(\mathrm{lift})"'] & \top \oslash F(X) \arrow[d, "\theta_F"] \\
 & F(\top \oslash X)
\end{tikzcd}$$

A functor (with strength) which satisfies the normal strength conditions together with the above will be called **contextually strong**.

It is clear that we may now form a 2–category of $\mathbf{X}$–modules, contextually strong functors, and strong transformations:

Proposition 3.1 *Contextual actions, contextually strong functors, and strong transformations form a 2-category* $\mathsf{Context}(\mathbf{X})$.

Again the only point that needs comment concerns the composition of contextually strong functors. Clearly we may define

$$(F, \theta_F) \circ (G, \theta_G) = (F \circ G, \theta_G; G(\theta_F))$$

and we leave it as an exercise for the reader to check that the five strength requirements are satisfied by the composite strength.

Strength has the interesting property of transferring onto the Eilenberg-Moore category of strong cotriples. Suppose (S, ϵ, δ) is a contextually strong cotriple on an $\mathbf{X}$–module $\mathbf{Y}$. Then an arbitrary object of the Eilenberg–Moore category is $\nu : Y \longrightarrow S(Y)$ satisfying the usual diagrams. However, this gives

$$X \oslash Y \xrightarrow{\nu_X} S(X \oslash Y) = X \oslash Y \xrightarrow{1 \oslash \nu} X \oslash S(Y) \xrightarrow{\theta_S} S(X \oslash Y)$$

which is also a coalgebra, since the following commute:

$$
\begin{tikzcd}
X \oslash Y \arrow[d, "1 \oslash \nu"'] \arrow[dr, equal] & \\
X \oslash S(Y) \arrow[r, "1 \oslash \epsilon"] \arrow[d, "\theta_S"'] & X \oslash Y \\
S(X \oslash Y) \arrow[ur, "\epsilon"'] &
\end{tikzcd}
\qquad
\begin{tikzcd}
X \oslash Y \arrow[r, "1 \oslash \nu"] \arrow[d, "1 \oslash \nu"'] & X \oslash S(Y) \arrow[r, "\theta_S"] \arrow[d, "1 \oslash S(\nu)"'] & S(X \oslash Y) \arrow[d, "S(1 \oslash \nu)"] \\
X \oslash S(Y) \arrow[r, "1 \oslash \delta"'] \arrow[d, "\theta_S"'] & X \oslash S(S(Y)) \arrow[r, "\theta_S"'] & S(X \oslash S(Y)) \arrow[d, "S(\theta_S)"] \\
S(X \oslash Y) \arrow[rr, "\delta"'] & & S(S(X \oslash Y))
\end{tikzcd}
$$

In each case, the bottom diagrams commute by the strength of ϵ, δ, and the top diagrams commute because ν is a coalgebra (and θ is natural, in the case of the right-hand diagram).

Further this is actually a contextual action:

Lemma 3.2 $\mathsf{Context}(\mathbf{X})$ *has the Eilenberg-Moore construction for cotriples.*

The requirements (16)–(24) can retrospectively now be seen as arising out of the demand that $a_\oslash$ and $c_\oslash$ be the linear contextual strengths for the functor $\_ \oslash \_$. Certainly, we have:

Lemma 3.3 *In a contextual category* $\mathbf{X}$ *for all objects* Y *and* Z *the following functors with strengths are contextually strong:* $(\_ \oslash Z, a_\oslash)$ *and* $(Y \oslash \_, c_\oslash)$.

We now wish to establish that $\_ \oslash \_$ has a contextual strength. We shall accomplish this by proving the following proposition:

Proposition 3.4 *A bifunctor* $F : \mathbf{Y}_1 \times \mathbf{Y}_2 \longrightarrow \mathbf{Y}$ *between* $\mathbf{X}$*–contextual categories has a contextual strength* θ_F *if and only if*

- *F has linear contextual strengths* $\mathrm{fst} : X \oslash F(Y_1, Y_2) \longrightarrow F(X \oslash Y_1, Y_2)$ *and* $\mathrm{snd} : X \oslash F(Y_1, Y_2) \longrightarrow F(Y_1, X \oslash Y_2)$,
- *The linear strengths commute:*

$$
\begin{tikzcd}
X \oslash (Y \oslash F(A,B)) \arrow[rr, "1 \oslash \mathrm{snd}"] \arrow[d, "c_\oslash"'] & & X \oslash F(A, (Y \oslash B)) \arrow[d, "\mathrm{fst}"] \\
Y \oslash (X \oslash F(A,B)) \arrow[r, "1 \oslash \mathrm{fst}"'] & Y \oslash F(X \oslash A, B) \arrow[r, "\mathrm{snd}"'] & F(X \oslash A, Y \oslash B)
\end{tikzcd}
$$

Furthermore a natural transformation between bifunctors $\alpha : F(X,Y) \longrightarrow G(X,Y)$ *is strong if and only if it is strong with respect to the linear strengths.*

This immediately explains the requirement (13) as this will give

Corollary 3.5 *In a contextual category* $\theta_{\_\oslash\_} = d; 1 \oslash c_\oslash; a_\oslash$ *is a contextual strength for the contextual action.*

Proof (of 3.4). We start by assuming that θ is a contextual strength for F: our task is to show that

$$X \oslash F(A,B) \xrightarrow{\text{fst}} F(X \oslash A, B) = X \oslash F(A,B) \xrightarrow{\theta_F} F(X \oslash A, X \oslash B) \xrightarrow{F(1,e)} F(X \oslash A, B)$$

will be a contextual strength. By symmetry this will allow us to conclude that $\text{snd} : X \oslash F(A,B) \longrightarrow F(A, X \oslash B)$ will be a contextual strength. To establish the first direction of the equivalence it will then only remain to show that these linear strengths commute.

We need to check the five conditions for strength:

Elimination strength:

$$\text{fst}; F(e,1) = \theta_F; F(1,e); F(e,1) = \theta_F; F(e,e) = e.$$

Duplication strength:

$$\begin{aligned}
\text{fst}; F(d,1) &= \theta_F; F(1,e); F(d,1) = \theta_F; F(d,e) \\
&= \theta_F; F(d, d; 1 \oslash e; e) \\
&= \theta_F; F(d,d); F(1, 1 \oslash e; e) \\
&= d; 1 \oslash \theta_F; \theta_F; F(1, 1 \oslash e); F(1,e) \\
&= d; 1 \oslash \theta_F; 1 \oslash F(1,e); \theta_F; F(1,e) \\
&= d; 1 \oslash \text{fst}; \text{fst}
\end{aligned}$$

Associative strength:

$$\begin{aligned}
1 \oslash \text{fst}; \text{fst}; F(a_\oslash, 1) &= 1 \oslash (\theta_F; F(1,e)); \theta_F; F(1,e); F(a_\oslash, 1) \\
&= 1 \oslash \theta_F; \theta_F; F(1, 1 \oslash e); F(1,e); F(a_\oslash, 1) \\
&= 1 \oslash \theta_F; \theta_F; F(a_\oslash, 1 \oslash e; e) \\
&= 1 \oslash \theta_F; \theta_F; F(a_\oslash, e; e) \\
&= 1 \oslash \theta_F; \theta_F; F(a_\oslash, a_\oslash; e \oslash 1; e) \\
&= 1 \oslash \theta_F; \theta_F; F(a_\oslash, a_\oslash); F(1, e \oslash 1; e) \\
&= a_\oslash; \theta_F; F(1,e) \\
&= a_\oslash; \text{fst}
\end{aligned}$$

Symmetric strength:

$$\begin{aligned}
1 \oslash \text{fst}; \text{fst}; F(c_\oslash, 1) &= 1 \oslash \theta_F; \theta_F; F(c_\oslash, 1); F(1, 1 \oslash e; e) \\
&= c_\oslash; 1 \oslash \theta_F; \theta_F; F(1, 1 \oslash e); F(1,e) \\
&= c_\oslash; 1 \oslash \theta_F; 1 \oslash F(1,e); \theta_F; F(1,e) \\
&= c_\oslash; 1 \oslash \text{fst}; \text{fst}
\end{aligned}$$

Lifting strength:

$$\begin{aligned}
\text{lift}; \text{fst} &= \text{lift}; \theta_F; F(1,e) \\
&= F(\text{lift}, \text{lift}; e) \\
&= F(\text{lift}, 1)
\end{aligned}$$

Note the use of (10) in the last step.

It remains (for the first direction) to show that the linear strengths so defined commute:

$$
\begin{array}{rcl}
1 \oslash \mathrm{snd}; \mathrm{fst} & = & 1 \oslash (\theta_F; F(e,1)); \theta_F; F(1,e) \\
& = & 1 \oslash \theta_F; \theta_F; F(1 \oslash e, e) \\
& = & 1 \oslash \theta_F; \theta_F; F(c_\oslash; e, c_\oslash; 1 \oslash e) \\
& = & c_\oslash; 1 \oslash \theta_F; \theta_F; F(e, 1 \oslash e) \\
& = & c_\oslash; 1 \oslash (\theta_F; F(1,e)); \theta_F; F(e,1) \\
& = & c_\oslash; 1 \oslash \mathrm{fst}; \mathrm{snd}\,.
\end{array}
$$

For the converse we now assume that we have the linear strengths which commute and will establish that

$$
\begin{array}{l}
X \oslash F(A,B) \xrightarrow{\theta_F} F(X \oslash A, X \oslash B) \\
\quad = \quad X \oslash F(A,B) \xrightarrow{d} X \oslash (X \oslash F(A,B)) \xrightarrow{1 \oslash \mathrm{snd}} X \oslash F(A, X \oslash B) \xrightarrow{\mathrm{fst}} F(X \oslash A, X \oslash B)
\end{array}
$$

is a contextual strength. Again we must check the five conditions governing strength. Before we do this we record a manipulation which will be used repeatedly:

$$
\begin{array}{rcl}
1 \oslash \theta_F; \theta_F & = & 1 \oslash (d; 1 \oslash \mathrm{snd}; \mathrm{fst}); d; 1 \oslash \mathrm{snd}; \mathrm{fst} \\
& = & d; 1 \oslash (1 \oslash (d; 1 \oslash \mathrm{snd}; \mathrm{fst})); 1 \oslash \mathrm{snd}; \mathrm{fst} \\
& = & d; 1 \oslash (1 \oslash d); 1 \oslash (1 \oslash (1 \oslash \mathrm{snd})); 1 \oslash (1 \oslash \mathrm{fst}); 1 \oslash \mathrm{snd}; \mathrm{fst} \\
& = & 1 \oslash d; d; 1 \oslash (1 \oslash (1 \oslash \mathrm{snd})); 1 \oslash c_\oslash; 1 \oslash (1 \oslash \mathrm{snd}); 1 \oslash \mathrm{fst}; \mathrm{fst} \\
& = & 1 \oslash d; d; 1 \oslash c_\oslash; 1 \oslash (1 \oslash (1 \oslash \mathrm{snd}; \mathrm{snd})); 1 \oslash \mathrm{fst}; \mathrm{fst}
\end{array}
$$

which uses the fact that the linear strengths commute to reexpress the process of moving contexts inside .

Elimination strength:

$$
\begin{array}{rcl}
\theta_F; F(e,e) & = & d; 1 \oslash \mathrm{snd}; \mathrm{fst}; F(e,e) \\
& = & d; 1 \oslash \mathrm{snd}; e; F(1,e) \\
& = & d; e; \mathrm{snd}; F(1,e) \\
& = & e
\end{array}
$$

Duplication strength:

$$
\begin{array}{rcl}
d; 1 \oslash \theta_F; \theta_F & = & d; 1 \oslash d; d; 1 \oslash c_\oslash; 1 \oslash (1 \oslash (1 \oslash \mathrm{snd}; \mathrm{snd})); 1 \oslash \mathrm{fst}; \mathrm{fst} \\
& = & d; d; 1 \oslash d; 1 \oslash c_\oslash; 1 \oslash (1 \oslash (1 \oslash \mathrm{snd}; \mathrm{snd})); 1 \oslash \mathrm{fst}; \mathrm{fst} \\
& = & d; d; 1 \oslash d; 1 \oslash (1 \oslash (1 \oslash \mathrm{snd}; \mathrm{snd})); 1 \oslash \mathrm{fst}; \mathrm{fst} \\
& = & d; d; 1 \oslash (1 \oslash d); 1 \oslash (1 \oslash (1 \oslash \mathrm{snd}; \mathrm{snd})); 1 \oslash \mathrm{fst}; \mathrm{fst} \\
& = & d; d; 1 \oslash (1 \oslash \mathrm{snd}); 1 \oslash (1 \oslash F(1,d)); 1 \oslash \mathrm{fst}; \mathrm{fst} \\
& = & d; 1 \oslash \mathrm{snd}; 1 \oslash F(1,d); d; 1 \oslash \mathrm{fst}; \mathrm{fst} \\
& = & d; 1 \oslash \mathrm{snd}; 1 \oslash F(1,d); \mathrm{fst}; F(d,1) \\
& = & d; 1 \oslash \mathrm{snd}; \mathrm{fst}; F(d,d)
\end{array}
$$

Associative strength:

$$\begin{array}{rcl} 1 \oslash \theta_F; \theta_F; F(a_\oslash, a_\oslash) & = & 1 \oslash d; d; 1 \oslash c_\oslash; 1 \oslash (1 \oslash (1 \oslash \mathrm{snd}; \mathrm{snd})); 1 \oslash \mathrm{fst}; \mathrm{fst}; F(a_\oslash, a_\oslash) \\ & = & 1 \oslash d; d; 1 \oslash c_\oslash; 1 \oslash (1 \oslash (1 \oslash \mathrm{snd}; \mathrm{snd})); a_\oslash; 1 \oslash F(1, a_\oslash); \mathrm{fst} \\ & = & 1 \oslash d; d; 1 \oslash c_\oslash; a_\oslash; 1 \oslash (1 \oslash \mathrm{snd}; \mathrm{snd}); 1 \oslash F(1, a_\oslash); \mathrm{fst} \\ & = & 1 \oslash d; d; 1 \oslash c_\oslash; a_\oslash; 1 \oslash a_\oslash; 1 \oslash \mathrm{snd}; \mathrm{fst} \\ & = & a_\oslash; d; 1 \oslash \mathrm{snd}; \mathrm{fst} \\ & = & a_\oslash; \theta_F \end{array}$$

Notice the use of the transposition identity (11) to bring the associative map out in the penultimate step.

Commutative strength:

$$\begin{array}{rcl} 1 \oslash \theta_F; \theta_F; F(c_\oslash, c_\oslash) & = & 1 \oslash d; d; 1 \oslash c_\oslash; 1 \oslash (1 \oslash (1 \oslash \mathrm{snd}; \mathrm{snd})); 1 \oslash \mathrm{fst}; \mathrm{fst}; F(c_\oslash, c_\oslash) \\ & = & 1 \oslash d; d; 1 \oslash c_\oslash; 1 \oslash (1 \oslash (1 \oslash \mathrm{snd}; \mathrm{snd})); c_\oslash; 1 \oslash (1 \oslash F(1, c_\oslash)); \\ & & 1 \oslash \mathrm{fst}; \mathrm{fst} \\ & = & 1 \oslash d; d; 1 \oslash c_\oslash; c_\oslash; 1 \oslash (1 \oslash c_\oslash); 1 \oslash (1 \oslash (1 \oslash \mathrm{snd}; \mathrm{snd})); \\ & & 1 \oslash \mathrm{fst}; \mathrm{fst} \\ & = & c_\oslash; 1 \oslash d; d; 1 \oslash c_\oslash; 1 \oslash (1 \oslash (1 \oslash \mathrm{snd}; \mathrm{snd})); 1 \oslash \mathrm{fst}; \mathrm{fst} \\ & = & c_\oslash; 1 \oslash \theta_F; \theta_F \end{array}$$

Notice the use of the transposition identity (12) to bring the symmetry map out in the penultimate step.

Lifting strength:

$$\begin{array}{rcl} \mathrm{lift}; \theta_F & = & \mathrm{lift}; d; 1 \oslash \mathrm{snd}; \mathrm{fst} \\ & = & \mathrm{lift}; \mathrm{lift}; 1 \oslash \mathrm{snd}; \mathrm{fst} \\ & = & \mathrm{lift}; \mathrm{snd}; \mathrm{lift}; \mathrm{fst} \\ & = & F(1, \mathrm{lift}); F(\mathrm{lift}, 1) \\ & = & F(\mathrm{lift}, \mathrm{lift}) \end{array}$$

Note the use of (9) in the first step.

The last statement of the proposition is straightforward to verify. $\square$

It would be strange indeed if the re-expression of a bistrength in terms of its linear strengths (and vice-versa) did not yield the bistrength itself. We now verify that these transitions are indeed inverses of each other. First we assume we are given the bistrength and reconstitute it from its components:

$$\begin{array}{rcl} \theta_F & = & d; 1 \oslash \mathrm{snd}; \mathrm{fst} \\ & = & d; 1 \oslash (\theta_F; F(e, 1)); \theta_F; F(1, e) \\ & = & d; 1 \oslash \theta_F; \theta_F; F(1 \oslash e, e) \\ & = & \theta_F; F(d; 1 \oslash e, d; e) \\ & = & \theta_F \end{array}$$

Next we assume we have the linear strengths and verify that extracting the linear strengths from the bistrength that we build does give back the original linear strengths:

$$
\begin{aligned}
\mathrm{snd} &= \theta_F; F(e,1) \\
&= d; 1 \oslash \mathrm{snd}; \mathrm{fst}; F(e,1) \\
&= d; 1 \oslash \mathrm{snd}; e \\
&= d; e; \mathrm{snd} \\
&= \mathrm{snd} \\
\mathrm{fst} &= \theta_F; F(1,e) \\
&= d; 1 \oslash \mathrm{fst}; \mathrm{snd}; F(1,e) \\
&= d; 1 \oslash \mathrm{fst}; e \\
&= d; e; \mathrm{fst} \\
&= \mathrm{fst}\,.
\end{aligned}
$$

For example, in the 2–category $\mathsf{Context}(\mathbf{X})$ the trivial category $\mathbf{1}$ of one object and one map (the identity) with the trivial action is a final object. Functors from it pick out objects as usual and have strength

$$e : X \oslash F(1) \longrightarrow F(X \oslash 1) = F(1),$$

thus the above result is just the analogue of the classical result concerning bifunctors.

As another example, if $\mathbf{Y}$ is an $\mathbf{X}$–module, that is a category with an $\mathbf{X}$-contextual action, then for each object $Y \in \mathbf{Y}$ there is a strong contextual functor

$$[Y] : \mathbf{X} \longrightarrow \mathbf{Y}; X \longrightarrow X \oslash Y$$

with the strength given by

$$a_\oslash : X' \oslash [Y](X) = X' \oslash (X \oslash Y) \longrightarrow [Y](X' \oslash X) = (X' \oslash X) \oslash Y.$$

This is called a (contextual) **line**.

3.3 Actional functors and transformations

A functor between two contextual categories $F : \mathbf{X}_1 \longrightarrow \mathbf{X}_2$ is **actional** in case it is equipped with two maps:

$$\tau : \top \longrightarrow F(\top) \quad \text{and} \quad \tau_\oslash : F(X) \oslash F(Y) \longrightarrow F(X \oslash Y)$$

such that the following diagrams commute:

$$
\begin{tikzcd}
F(X) \oslash F(Y) \arrow[r, "e"] \arrow[d, "\tau_\oslash"'] & F(Y) \\
F(X \oslash Y) \arrow[ur, "F(e)"'] &
\end{tikzcd}
\qquad
\begin{tikzcd}
F(X) \oslash F(Y) \arrow[r, "d"] \arrow[d, "\tau_\oslash"'] & F(X) \oslash (F(X) \oslash F(Y)) \arrow[d, "1 \oslash \tau_\oslash; \tau_\oslash"] \\
F(X \oslash Y) \arrow[r, "F(d)"'] & F(X \oslash (X \oslash Y))
\end{tikzcd}
$$

$$
\begin{tikzcd}
F(X) \arrow[r, "\mathrm{lift}"] \arrow[d, "F(\mathrm{lift})"'] & \top \oslash F(X) \arrow[d, "\tau"] \\
F(\top \oslash X) & F(\top) \oslash F(X) \arrow[l, "\tau_{\oslash}"]
\end{tikzcd}
\qquad
\begin{tikzcd}
F(X) \oslash \top \arrow[r, "\mathrm{read}"] \arrow[d, "1 \oslash \tau"'] & F(X) \\
F(X) \oslash F(\top) \arrow[r, "\tau_{\oslash}"'] & F(X \oslash \top) \arrow[u, "F(\mathrm{read})"']
\end{tikzcd}
$$

$$
\begin{tikzcd}
F(X) \oslash (F(Y) \oslash F(Z)) \arrow[r, "a_{\oslash}"] \arrow[d, "1 \oslash \tau_{\oslash}; \tau_{\oslash}"'] & (F(X) \oslash F(Y)) \oslash F(Z) \arrow[d, "1 \oslash \tau_{\oslash}; \tau_{\oslash}"] \\
F(X \oslash (Y \oslash Z)) \arrow[r, "F(c_{\oslash})"'] & F((X \oslash Y) \oslash Z)
\end{tikzcd}
$$

$$
\begin{tikzcd}
F(X) \oslash (F(Y) \oslash F(Z)) \arrow[r, "c_{\oslash}"] \arrow[d, "1 \oslash \tau_{\oslash}; \tau_{\oslash}"'] & F(Y) \oslash (F(X) \oslash F(Z)) \arrow[d, "1 \oslash \tau_{\oslash}; \tau_{\oslash}"] \\
F(X \oslash (Y \oslash Z)) \arrow[r, "F(a_{\oslash})"'] & F(Y \oslash (X \oslash Z))
\end{tikzcd}
$$

A natural transformation $\alpha : F \longrightarrow G$ between actional functors is actional in case the following diagrams commute

$$
\begin{tikzcd}
 & \top \arrow[dl, "\tau^F"'] \arrow[dr, "\tau^G"] & \\
F(\top) \arrow[rr, "\alpha"'] & & G(\top)
\end{tikzcd}
\qquad
\begin{tikzcd}
F(A) \oslash F(B) \arrow[r, "\alpha \oslash \alpha"] \arrow[d, "\tau^F_{\oslash}"'] & G(A) \oslash G(B) \arrow[d, "\tau^G_{\oslash}"] \\
F(A \oslash B) \arrow[r, "\alpha"'] & G(A \oslash B)
\end{tikzcd}
$$

Proposition 3.6 *Contextual categories, actional functors, and actional transformations form a 2–category* $\mathsf{Context}$.

The only difficulty concerns composition. The actional maps for a composition are defined exactly as expected:

$$
\begin{aligned}
& \top \xrightarrow{\tau^{G \circ F}} G(F(\top)) \\
& \quad = \quad \top \xrightarrow{\tau^{G}} G(\top) \xrightarrow{G(\tau^F)} G(F(\top)) \\
& G(F(X)) \oslash G(F(Y)) \xrightarrow{\tau^{G \circ F}_{\oslash}} G(F(X \oslash Y)) \\
& \quad = \quad G(F(X)) \oslash G(F(Y)) \xrightarrow{\tau^{G}_{\oslash}} G(F(X) \oslash F(Y)) \xrightarrow{G(\tau^F_{\oslash})} G(F(X \oslash Y))
\end{aligned}
$$

It is then necessary to check the above diagrams for these maps. This is a straightforward calculation which utilizes the naturality of the actional maps.

Of course, actional functors are the analogue of monoidal functors. The proof that this composition works is analogous to the proof for monoidal functors.

The importance of actional maps between contextual categories is precisely that they allow the transfer of the modules. This means that the modules sit above the 2–category $\mathsf{Context}$ as a 2-fibration and $\mathsf{Context}(\mathbf{X})$ is the fiber above the contextual category $\mathbf{X}$. We shall be content to prove:

Proposition 3.7 *There is a 2-functor* $(\_)^* : \mathsf{Context} \longrightarrow 2-\mathbf{Cat}$ *(the 3–category of 2–categories) which takes*

$$\mathbf{X} \mapsto \mathsf{Context}(\mathbf{X})$$

giving a 2–indexed 2–category.

Proof (partial sketch!). We shall restrict ourselves to defining the way the actional functors induce 2–functors.

Let $F : \mathbf{X}_0 \longrightarrow \mathbf{X}_1$ be a 1–cell of $\mathsf{Context}$. To define

$$F^* : \mathsf{Context}(\mathbf{X}_1) \longrightarrow \mathsf{Context}(\mathbf{X}_0)$$

Define the $\mathbf{X}_0$–action on an $\mathbf{X}_1$–module by $X_0 \oslash Y = F(X_0) \oslash Y$. We need the lift, associativity, and symmetry for this action. The elimination, duplication, and symmetry map are unchanged. However associativity and lifting must be modified:

$$\begin{aligned}
& F(X_1) \oslash (F(X_1') \oslash Y) \xrightarrow{a'_\oslash} F(X_1 \oslash X_1') \oslash Y \\
& \quad = \; F(X_1) \oslash (F(X_1') \oslash Y) \xrightarrow{a_\oslash} (F(X_1) \oslash F(X_1')) \oslash Y \xrightarrow{\tau_\oslash \oslash 1} F(X_1 \oslash X_1') \oslash Y \\
& Y \xrightarrow{\mathrm{lift}'} F(\top) \oslash Y \\
& \quad = \; Y \xrightarrow{\mathrm{lift}} \top \oslash Y \xrightarrow{\tau \oslash 1} F(\top) \oslash Y.
\end{aligned}$$

In order to check that this is a contextual action we need to check that the twenty two applicable diagrams still commute! However, those that we have not modified (that is those only involving elimination, duplication, and symmetry) will automatically still commute. This leaves (6), (9), (10), (11), (13), (14), (16), (18), (21), (22), and (24). These are straightforward to check. We shall do (6) to illustrate:

$$\begin{tikzcd}
 & F(X) \oslash (F(\top) \oslash Z) \arrow[r, "a_\oslash"] & (F(X) \oslash F(\top)) \oslash Z \arrow[r, "\tau_\oslash \oslash 1"] & F(X \oslash \top) \oslash Z \arrow[d, "F(\mathrm{read}) \oslash 1"] \\
F(X) \oslash Z \arrow[r, "1 \oslash \mathrm{lift}"] & F(X) \oslash (\top \oslash Z) \arrow[u, "1 \oslash (\tau \oslash 1)"] \arrow[r, "a_\oslash"] & (F(X) \oslash \top) \oslash Z \arrow[u, "(1 \oslash \tau) \oslash 1"] \arrow[r, "\mathrm{read} \oslash 1"] & F(X) \oslash Y
\end{tikzcd}$$

We must show that this assignment preserves the composition. Further, we must show how actional transformations give rise to 2–cells and preserve composition. This is left as an exercise for the reader. □

Now $\mathsf{Context}$ does not have all weighted limits and colimits but it does have some:

Proposition 3.8 $\mathsf{Context}$ *has Eilenberg-Moore and Kleisli constructions for cotriples.*

The Kleisli construction is of course not the standard one as we must close up (the standard construction) by the action and ensure the presence of the required actional transformations. To get a precise feel of this category is beyond the scope of the current article.

To show that the Eilenberg-Moore category exists is more straightforward. It suffices to indicate why the contextual structure, action and transformations, carries over to the coalgebras.

Note that if

$$X \xrightarrow{\nu} S(X) \quad \text{and} \quad X' \xrightarrow{\nu'} S(X')$$

are coalgebras then

$$X \oslash X' \xrightarrow{\nu \oslash \nu'} S(X) \oslash S(X') \xrightarrow{\tau^S_\oslash} S(X \oslash X')$$

is easily seen to be a coalgebra too. It is then easily checked that, as the contextual transformations can be pulled through the actional maps, the category is contextual.

3.4 Storage in contextual categories

Suggestively we shall adopt the convention, in any contextual category, of writing $\overline{!}(X) = X \oslash \top$. Notice that we have:

$$\overline{!}(X) \xrightarrow{\delta} \overline{!}(\overline{!}(X)) = X \oslash \top \xrightarrow{1 \oslash \mathrm{lift}} X \oslash (\top \oslash \top) \xrightarrow{a_\oslash} (X \oslash \top) \oslash \top$$

and

$$!(X) \xrightarrow{\epsilon} X = X \oslash \top \xrightarrow{\mathrm{read}} X.$$

Lemma 3.9 *In any contextual category* $(\overline{!}(\_), \epsilon, \delta)$ *is a cotriple.*

Proof. We must show that $\delta; \overline{!}(\epsilon) = \delta; \epsilon = 1$ and $\delta; \delta = \delta; \overline{!}(\delta)$.

The first counit identity $\delta; \overline{!}(\epsilon) = 1$ is (when translated) (6) above.

For the second counit identity we have:

$$\begin{tikzcd}
X \oslash \top \arrow[r, "1 \oslash \mathrm{lift}"] \arrow[dr, equal] & X \oslash (\top \oslash \top) \arrow[r, "a_\oslash"] \arrow[d, "1 \oslash \mathrm{read}"] & (X \oslash \top) \oslash \top \arrow[dl, "\mathrm{read}"] \\
 & X \oslash \top &
\end{tikzcd}$$

where the left triangle is (5) and the right triangle commutes as read is strong – (15) above.

Finally for the associativity of comultiplication we have:

$$
\begin{tikzcd}
X \oslash \top \arrow[rr, "1 \oslash \mathrm{lift}"] \arrow[dd, "1 \oslash \mathrm{lift}"'] \arrow[ddr, phantom, "(A)"] & & X \oslash \overline{!}(\top) \arrow[r, "a_\oslash"] \arrow[d, "1 \oslash (1 \oslash \mathrm{lift})"'] \arrow[dr, phantom, "(B)"] & \overline{!}(\overline{!}(X)) \arrow[d, "1 \oslash \mathrm{lift}"] \\
& & X \oslash (\top \oslash \overline{!}(\top)) \arrow[r, "a_\oslash"] \arrow[dl, "1 \oslash a_\oslash"'] \arrow[ddr, phantom, "(D)"] & \overline{!}(X) \oslash \overline{!}(\top) \arrow[dd, "a_\oslash"] \\
X \oslash \overline{!}(\top) \arrow[r, "1 \oslash (\mathrm{lift} \oslash 1)"'] \arrow[d, "a_\oslash"'] \arrow[dr, phantom, "(C)"] & X \oslash \overline{!}(\overline{!}(\top)) \arrow[d, "a_\oslash \oslash 1"] & & \\
\overline{!}(\overline{!}(X)) \arrow[r, "(1 \oslash \mathrm{lift}) \oslash 1"'] & \overline{!}(X \oslash \overline{!}(\top)) \arrow[rr, "a_\oslash \oslash 1"'] & & \overline{!}(\overline{!}(\overline{!}(X)))
\end{tikzcd}
$$

where the cell (A) commutes using the naturality of lift to obtain the equality:

$$\mathrm{lift}; \mathrm{lift} = \mathrm{lift}; 1 \oslash \mathrm{lift}$$

and the fact that lift is strong (14) which says $\mathrm{lift}; a_\oslash = \mathrm{lift} \oslash 1$; these allow

$$
\begin{aligned}
\mathrm{lift}; 1 \oslash \mathrm{lift}; a_\oslash &= \mathrm{lift}; \mathrm{lift}; a_\oslash \\
&= \mathrm{lift}; \mathrm{lift} \oslash 1
\end{aligned}
$$

Squares (B) and (C) commute by naturality, (D) uses the associativity pentagon (20). □

So, in particular, the storage cotriple gives rise to a module. It would be particularly satisfying if the storage cotriple were actional. This would mean that the Eilenberg-Moore category would also be a contextual category. Unfortunately this is not the case in general (a counterexample is provided by the free linear category on a set of types). However, it is actional whenever the storage cotriple has its comultiplication an isomorphism. (This condition has been studied by Wadler [W94].) This makes the "stored types" a full reflexive subcategory:

Proposition 3.10 *Any contextual category, in which the storage comultiplication is an isomorphism, has an actional storage cotriple.*

Proof. Define

$$
\begin{aligned}
\top \xrightarrow{\tau} \overline{!}(\top) &= \top \xrightarrow{\mathrm{lift}} \top \oslash \top \\
\overline{!}(X) \oslash \overline{!}(Y) \xrightarrow{\tau_\oslash} \overline{!}(X \oslash Y) &= (X \oslash \top) \oslash (Y \oslash \top) \xrightarrow{\mathrm{read} \oslash 1} X \oslash (Y \oslash \top) \xrightarrow{a_\oslash} (X \oslash Y) \oslash \top.
\end{aligned}
$$

The only difficulty in the proof is the requirement concerned with reading:

$$1 \oslash \tau; \tau_{\oslash}; \overline{!}(\mathrm{read}) = \mathrm{read}$$

We may perform the following manipulation:

$$\begin{aligned} 1 \oslash \tau; \tau_{\oslash}; \overline{!}(\mathrm{read}) &= 1 \oslash \mathrm{lift}; \mathrm{read} \oslash 1; a_{\oslash}; \mathrm{read} \oslash 1 \\ &= \mathrm{read} \oslash 1; 1 \oslash \mathrm{lift}; a_{\oslash}; \mathrm{read} \oslash 1 \\ &= \mathrm{read} \oslash 1 \end{aligned}$$

where the last step uses (6). However, we are left to prove that $\mathrm{read} \oslash 1 = \mathrm{read} : (X \oslash \top) \oslash \top \longrightarrow X \oslash \top$ which in general is not the case. However, we can now use the assumption of the proposition to force this for prefixing these maps by the comultiplication in each case is the identity. Thus, as the comultiplication is assumed to be an isomorphism these maps are, by assumption, equal. □

Remark 3.11 ()
In developing the theory above, we have had one model always in mind, which is in a sense the converse of this section, *viz.* in a monoidal category with a "storage" cotriple $!$, we may define an action *via* $X \oslash Y = \,!X \otimes Y$. The reader might like to verify that this is indeed a contextual category. (For definiteness, one could take the setup of either [BCS92] or [BBPH].) As we shall not need this here, we shall leave the details of this verification to such readers. □

Remark 3.12 (Modules and fibrations)
As before it is the case that $\mathsf{Context}(\mathbf{X})$ does not have all weighted limits. As before we therefore would like to be assured of a completion in which all these limits do exist. Again it is reasonable to look for a fibrational completion for these modules. However, for the moment we shall leave this as an exercise for the reader! Instead we shall now consider coherence for contextual categories, *via* a study of proof circuits for these categories. □

4 The context calculus

It must be obvious by now that the structures we have been studying are very similar to Girard's approach to unifying classical and linear logic [G93]: context variables are "classical" and general variables are "linear". This is represented by a morphism $C \oslash G \longrightarrow A$; the position C before the $\oslash$ is "classical", while the position G after $\oslash$ is "linear". In the remainder of this paper we shall develop this idea, representing it by sequents $\Gamma; \Pi \vdash A$, where Γ is a finite (including empty) sequence of formulas, and Π is either empty or a single formula. Note that in a sense the role of $\oslash$ is taken by the semicolon in the sequent. We shall present the sequent calculus, outline the cut elimination theorem for it, and sketch the interpretation of the calculus in contextual categories.

We begin with the sequent calculus—as suggested above, this is a fragment of Unified Logic, and is intended to illustrate the way in which contextual categories handle the features of $!$ and $?$, and how they handle the interaction of classical and linear behaviour. The sequent rules are listed in Table 1. There are some derived rules that one might otherwise expect in this system. These are listed at the bottom of the Table. We note that it is clear that the context rules amount to making the $\oslash$ carry the structure of the semicolon; the $(funct)$ rule essentially says that $\oslash$ is functorial. In the $(funct)$ rule it is understood that if either Π_i is empty, it will be replaced with

Axioms:

$$\frac{}{;A \vdash A}\ (id)$$

Structure Rules:

$$\frac{\Gamma, A, A, \Gamma'; \Pi \vdash B}{\Gamma, A, \Gamma'; \Pi \vdash B}\ (contr) \qquad \frac{\Gamma, \Gamma'; \Pi \vdash B}{\Gamma, A, \Gamma'; \Pi \vdash B}\ (thin)$$

$$\frac{\Gamma; A \vdash B}{\Gamma, A;\ \ \vdash B}\ (der) \qquad \frac{\Gamma, A, B, \Gamma'; \Pi \vdash C}{\Gamma, B, A, \Gamma'; \Pi \vdash C}\ (exch)$$

Cut rules:

$$\frac{\Delta; \Pi \vdash A \quad \Gamma; A \vdash B}{\Gamma, \Delta; \Pi \vdash B}\ (lcut) \qquad \frac{\Delta; \Sigma \vdash A \quad \Gamma, A, \Gamma'; \Pi \vdash B}{\Gamma, \Delta, \Sigma, \Gamma'; \Pi \vdash B}\ (ccut)$$

Unit Rules:

$$\frac{\Gamma;\ \ \vdash A}{\Gamma; \top \vdash A}\ (\top L) \qquad \frac{}{;\ \ \vdash \top}\ (\top R)$$

Context Rules:

$$\frac{\Gamma, A; B \vdash C}{\Gamma; A \oslash B \vdash C}\ (\oslash L) \qquad \frac{\Gamma; \Sigma \vdash A \quad \Delta; \Pi \vdash B}{\Gamma, \Sigma, \Delta; \Pi \vdash A \oslash B}\ (\oslash R)$$

Derived Rules:

$$\frac{}{A;\ \ \vdash A}\ (der - ax)$$

$$\frac{\Gamma, \top; \Pi \vdash A}{\Gamma; \Pi \vdash A}\ (\top E) \qquad \frac{\Gamma; \top \vdash A}{\Gamma;\ \ \vdash A}\ (\top E')$$

$$\frac{}{A; B \vdash A \oslash B}\ (\oslash I) \qquad \frac{\Gamma_0; \Pi_0 \vdash A \quad \Gamma_1; \Pi_1 \vdash B}{\Gamma_0, \Gamma_1; \Pi_0 \oslash \Pi_1 \vdash A \oslash B}\ (funct)$$

Note that $\Pi, \Sigma,$ or Π_i is either empty or a single formula.

Table 1: Sequent rules for the context calculus

$\top$ in $\Pi_0 \oslash \Pi_1$. The unit rules essentially amount to making the "unit" $\top$ carry the structure of an empty linear formula, that is, an empty formula after the semicolon. Note that $\top$ is *not* the unit for the $\oslash$, which would require a constant carrying the structure of an empty classical formula, that is, an empty formula before the semicolon.

5 Interpretation

We shall interpret a sequent $A_1, A_2, \ldots, A_n; B \vdash C$ as a morphism $A_1 \oslash (A_2 \oslash (\ldots (A_n \oslash B) \ldots)) \longrightarrow C$. In this way then, the (id) axiom is just the identity morphism $A \longrightarrow A$. The structure rules are given by the defining natural transformations for contextual categories: contraction is induced by the duplication natural transformation $d: X \oslash Y \longrightarrow X \oslash (X \oslash Y)$; thinning is induced by the elimination natural transformation $e: X \oslash Y \longrightarrow Y$; exchange is induced by $c_\oslash: X \oslash (Y \oslash Z) \longrightarrow Y \oslash (X \oslash Z)$; and dereliction is induced by read: $X \oslash \top \longrightarrow X$. The "linear" cut rule $(lcut)$ is simple composition (and functoriality of $\oslash$): given $Y \oslash Z \xrightarrow{g} B$ and $X \oslash B \xrightarrow{f} C$, the $lcut$ of f, g is $X \oslash (Y \oslash Z) \xrightarrow{X \oslash g} X \oslash B \xrightarrow{f} C$. The "classical" cut rule $(ccut)$ is given by the context natural transformation $a_\oslash$ and composition: for $Y \oslash S \xrightarrow{g} B$ and $X \oslash (B \oslash Z) \xrightarrow{f} C$, the $ccut$ of f, g is $X \oslash (Y \oslash (S \oslash Z)) \xrightarrow{X \oslash a_\oslash} X \oslash ((Y \oslash S) \oslash Z) \xrightarrow{X \oslash (g \oslash Z)} X \oslash (B \oslash Z) \xrightarrow{f} C$. $(\top L)$ is an identity interpretation, in the sense that we interpret the bottom formula by the interpretation of the top formula (we use $\top$ to interpret an empty linear formula, and we use the absence of $\oslash$ to interpret an empty classical sequence of formulas). $(\top R)$ is the identity on $\top$. The $(\oslash L)$ rule is also an identity interpretation; $(\oslash I)$ is interpreted as the identity morphism. $(\oslash R)$ is equivalent to $(\oslash I)$, but one can easily check that its interpretation may be given as follows. For $f: G \oslash S \longrightarrow A, g: D \oslash P \longrightarrow B$, $(\oslash R)$ yields $G \oslash (S \oslash (D \oslash P)) \xrightarrow{G \oslash (S \oslash g)} G \oslash (S \oslash B) \xrightarrow{a} (G \oslash S) \oslash B \xrightarrow{f \oslash B} A \oslash B$. The functoriality rule $(funct)$ is given by functoriality of $\oslash$, together with c, a to group the brackets according to the sequent syntax. Given $f_i: G_i \oslash P_i \longrightarrow A_i$, for $i = 0, 1$, $(funct)$ yields $G_0 \oslash (G_1 \oslash (P_0 \oslash P_1)) \xrightarrow{c_\oslash} G_0 \oslash (P_0 \oslash (G_1 \oslash P_1)) \xrightarrow{a_\oslash} (G_0 \oslash P_0) \oslash (G_1 \oslash P_1)) \xrightarrow{f_0 \oslash f_1} A_0 \oslash A_1$.

In a weakly distributive category with storage we may also interpret the context calculus, essentially $X \oslash Y = \mathbf{!}X \otimes Y$, as we shall see in a sequel.

6 Cut elimination

The cut elimination process for the context calculus is quite straightforward, first eliminating all linear cuts. There are a few points to illustrate.

- (Permuting $(lcut)$ and $(contr)$): If the contraction occurs on the left of the cut, it cannot be the cut formula that is contracted, so permuting the contraction past the cut is simple:

$$\dfrac{\dfrac{\Delta, A, A, \Delta'; \Pi \vdash B}{\Delta, A, \Delta'; \Pi \vdash B} \quad \Gamma; B \vdash C}{\Gamma, \Delta, A, \Delta'; \Pi \vdash C} \quad \Longrightarrow \quad \dfrac{\dfrac{\Delta, A, A, \Delta'; \Pi \vdash B \quad \Gamma; B \vdash C}{\Gamma, \Delta, A, A, \Delta'; \Pi \vdash C}}{\Gamma, \Delta, A, \Delta'; \Pi \vdash C}$$

  If the contraction occurs on the right of the cut, again the cut formula is not contracted:

$$\dfrac{\Delta; \Pi \vdash B \quad \dfrac{\Gamma, A, A, \Gamma'; B \vdash C}{\Gamma, A, \Gamma'; B \vdash C}}{\Gamma, A, \Gamma', \Delta; \Pi \vdash C} \quad \Longrightarrow \quad \dfrac{\dfrac{\Delta; \Pi \vdash B \quad \Gamma, A, A, \Gamma'; B \vdash C}{\Gamma, A, A, \Gamma', \Delta; \Pi \vdash C}}{\Gamma, A, \Gamma', \Delta; \Pi \vdash C}$$

- (Permuting $(ccut)$ and $(contr)$): Again, permuting an instance of contraction on the left of the cut is simple:

$$\frac{\dfrac{\Delta, A, A, \Delta'; \Sigma \vdash B}{\Delta, A, \Delta'; \Sigma \vdash B} \quad \Gamma, B, \Gamma'; \Pi \vdash C}{\Gamma, \Delta, A, \Delta', \Sigma, \Gamma'; \Pi \vdash C} \quad \Longrightarrow \quad \frac{\dfrac{\Delta, A, A, \Delta'; \Sigma \vdash B \quad \Gamma, B, \Gamma'; \Pi \vdash C}{\Gamma, \Delta, A, A, \Delta', \Sigma, \Gamma'; \Pi \vdash C}}{\Gamma, \Delta, A, \Delta', \Sigma, \Gamma'; \Pi \vdash C}$$

If the contraction appears on the right and the cut formula is not the formula contracted, the permutation is similar to the case above with $(lcut)$. If the contraction appears on the right and involves the cut formula, then permuting the contraction introduces a second cut, as illustrated below. Note the use of numerous instances of exchange and contraction indicated by the double lines.

$$\frac{\Delta; \Sigma \vdash A \quad \dfrac{\Gamma, A, A, \Gamma'; \Pi \vdash B}{\Gamma, A, \Gamma'; \Pi \vdash B}}{\Gamma, \Delta, \Sigma, \Gamma'; \Pi \vdash B} \quad \Longrightarrow \quad \frac{\dfrac{\Delta; \Sigma \vdash A \quad \dfrac{\Delta; \Sigma \vdash A \quad \Gamma, A, A, \Gamma'; \Pi \vdash B}{\Gamma, A, \Delta, \Sigma, \Gamma'; \Pi \vdash B}}{\Gamma, \Delta, \Sigma, \Delta, \Sigma, \Gamma'; \Pi \vdash B}}{\Gamma, \Delta, \Sigma, \Gamma'; \Pi \vdash B}$$

- (Permuting $(lcut)$ and (der)): An instance of dereliction cannot occur on the right of a linear cut; if it occurs on the left, permuting the cut is straightforward:

$$\frac{\dfrac{\Delta; A \vdash B}{\Delta, A; \quad \vdash B} \quad \Gamma; B \vdash C}{\Gamma, \Delta, A; \quad \vdash C} \quad \Longrightarrow \quad \frac{\dfrac{\Delta; A \vdash B \quad \Gamma; B \vdash C}{\Gamma, \Delta; A \vdash C}}{\Gamma, \Delta, A; \quad \vdash C}$$

- (Permuting $(ccut)$ and (der)): If the instance of dereliction occurs on the left side of a classical cut, the dereliction may be removed (notice that the $(ccut)$ rule has an instance of dereliction built in):

$$\frac{\dfrac{\Delta; A \vdash B}{\Delta, A; \quad \vdash B} \quad \Gamma, B, \Gamma'; \Pi \vdash C}{\Gamma, \Delta, A, \Gamma'; \Pi \vdash C} \quad \Longrightarrow \quad \frac{\Delta; A \vdash B \quad \Gamma, B, \Gamma'; \Pi \vdash C}{\Gamma, \Delta, A, \Gamma'; \Pi \vdash C}$$

If the dereliction occurs on the right, the permutation is straightforward:

$$\frac{\Delta, \Sigma \vdash B \quad \dfrac{\Gamma, B, \Gamma'; \Pi \vdash C}{\Gamma, B, \Gamma', \Pi; \quad \vdash C}}{\Gamma, \Delta, \Sigma, \Gamma', \Pi; \quad \vdash C} \quad \Longrightarrow \quad \frac{\dfrac{\Delta, \Sigma \vdash B \quad \Gamma, B, \Gamma'; \Pi \vdash C}{\Gamma, \Delta, \Sigma, \Gamma'; \Pi \vdash C}}{\Gamma, \Delta, \Sigma, \Gamma', \Pi; \quad \vdash C}$$

- (Permuting $(ccut)$ and $(thin)$): The only interesting case is when the cut formula is introduced by thinning: note the use of numerous instances of thinning indicated by the double line.

$$\frac{\Delta; \Sigma \vdash A \quad \dfrac{\Gamma, \Gamma'; \Pi \vdash B}{\Gamma, A, \Gamma'; \Pi \vdash B}}{\Gamma, \Delta, \Sigma, \Gamma'; \Pi \vdash A} \quad \Longrightarrow \quad \frac{\Gamma, \Gamma'; \Pi \vdash B}{\Gamma, \Delta, \Sigma, \Gamma'; \Pi \vdash B}$$

- (Permuting cuts and $(exch)$): These cases are completely analogous to the cases above.

- (Permuting (*lcut*) and (*ccut*)): There are two cases, depending on where the cut formula appears.

$$\dfrac{\dfrac{\Delta;\Sigma\vdash A \quad \Gamma,A,\Gamma';B\vdash C}{\Gamma,\Delta,\Sigma,\Gamma';B\vdash C} \quad \Phi;C\vdash D}{\Phi,\Gamma,\Delta,\Sigma,\Gamma';B\vdash D} \quad\Longrightarrow\quad \dfrac{\Delta;\Sigma\vdash A \quad \dfrac{\Gamma,A,\Gamma';B\vdash C \quad \Phi;C\vdash D}{\Phi,\Gamma,A,\Gamma';B\vdash D}}{\Phi,\Gamma,\Delta,\Sigma,\Gamma';B\vdash D}$$

$$\dfrac{\Phi;\Pi\vdash B \quad \dfrac{\Delta;\Sigma\vdash A \quad \Gamma,A,\Gamma';B\vdash C}{\Gamma,\Delta,\Sigma,\Gamma';B\vdash C}}{\Gamma,\Delta,\Sigma,\Gamma',\Phi;\Pi\vdash C} \quad\Longrightarrow\quad \dfrac{\Delta;\Sigma\vdash A \quad \dfrac{\Phi;\Pi\vdash B \quad \Gamma,A,\Gamma';B\vdash C}{\Gamma,A,\Gamma',\Phi;\Pi\vdash C}}{\Gamma,\Delta,\Sigma,\Gamma',\Phi;\Pi\vdash C}$$

- (Permuting (*lcut*) and $\oslash$): Note that if $A\oslash B$ is introduced by the $\oslash$ rules and is the cut formula, the cut must be linear. We illustrate such a cut, and also the permutation step if $A\oslash B$ is not the cut formula.

$$\dfrac{\dfrac{\Delta';\Sigma\vdash A \quad \Delta;\Pi\vdash B}{\Delta',\Sigma,\Delta;\Pi\vdash A\oslash B} \quad \dfrac{\Gamma,A;B\vdash C}{\Gamma;A\oslash B\vdash C}}{\Gamma,\Delta',\Sigma,\Delta;\Pi\vdash C} \quad\Longrightarrow\quad \dfrac{\Delta;\Pi\vdash B \quad \dfrac{\Delta';\Sigma\vdash A \quad \Gamma,A;B\vdash C}{\Gamma,\Delta',\Sigma;B\vdash C}}{\Gamma,\Delta',\Sigma,\Delta;\Pi\vdash C}$$

$$\dfrac{\Delta;\Pi\vdash B \quad \dfrac{\Gamma;\Sigma\vdash A \quad \Gamma';B\vdash D}{\Gamma,\Sigma,\Gamma';B\vdash A\oslash D}}{\Gamma,\Sigma,\Gamma',\Delta;\Pi\vdash A\oslash D} \quad\Longrightarrow\quad \dfrac{\Gamma;\Sigma\vdash A \quad \dfrac{\Delta;\Pi\vdash B \quad \Gamma';B\vdash D}{\Gamma',\Delta;\Pi\vdash D}}{\Gamma,\Sigma,\Gamma',\Delta;\Pi\vdash A\oslash D}$$

- (Permuting (*ccut*) and $\oslash$): Here $A\oslash B$ cannot be the cut formula.

$$\dfrac{\Delta;\Sigma\vdash B \quad \dfrac{\Gamma,B,A;D\vdash C}{\Gamma,B;A\oslash D\vdash C}}{\Gamma,\Delta,\Sigma;A\oslash D\vdash C} \quad\Longrightarrow\quad \dfrac{\dfrac{\Delta;\Sigma\vdash B \quad \Gamma,B,A;D\vdash C}{\Gamma,\Delta,\Sigma,A;D\vdash C}}{\Gamma,\Delta,\Sigma;A\oslash D\vdash C}$$

To show that these cut elimination steps are valid in contextual categories, we have some commutative diagrams to verify. For example, permuting (*ccut*) and (*thin*) amounts to the commutativity of the outer rectangle of the following diagram; the inner cells prove its commutativity.

$$\begin{tikzcd}
X\oslash(A\oslash B) \arrow[r, "1\oslash e"] & X\oslash B \\
X\oslash((Y\oslash Z)\oslash B) \arrow[u, "1\oslash(g\oslash 1)"] \arrow[r, "1\oslash e"] \arrow[dr, "1\oslash(e\oslash 1)"] & X\oslash B \arrow[u, "1"] \\
X\oslash(Y\oslash(Z\oslash B)) \arrow[u, "1\oslash a"] \arrow[r, "1\oslash e"] & X\oslash(Z\oslash B) \arrow[u, "1\oslash e"]
\end{tikzcd}$$

Similarly, permuting $(ccut)$ and $(contr)$ amounts to the commutativity of the outer rectangle of the following diagram; again, the inner cells prove its commutativity. The essential condition is the commutative diagram corresponding to equation (11) in the definition of a contextual category.

$$
\begin{tikzcd}
D \oslash (S \oslash P) \arrow[r, "d"] \arrow[dd, "a"] & D \oslash (D \oslash (S \oslash P)) \arrow[r, "1 \oslash (1 \oslash d)"] & D \oslash (D \oslash (S \oslash (S \oslash P))) \arrow[r, "1 \oslash c"] & D \oslash (S \oslash (D \oslash (S \oslash P))) \arrow[d, "a"] \\
 & (11) & & (D \oslash S) \oslash (D \oslash (S \oslash P)) \arrow[d, "g \oslash 1"] \arrow[dll, "1 \oslash a"'] \\
(D \oslash S) \oslash P \arrow[r, "d"] \arrow[dd, "g \oslash 1"] & (D \oslash S) \oslash ((D \oslash S) \oslash P) \arrow[drr, "g \oslash 1"] \arrow[ddr, "g \oslash (g \oslash 1)"'] & & A \oslash (D \oslash (S \oslash P)) \arrow[d, "1 \oslash a"] \\
 & & & A \oslash ((D \oslash S) \oslash P) \arrow[dl, "1 \oslash (g \oslash 1)"] \\
A \oslash P \arrow[rr, "d"'] & & A \oslash (A \oslash P) &
\end{tikzcd}
$$

The remaining diagrams will be left as an exercise.

7 Proof circuits

We now introduce proof circuits for the context calculus. The basic links are given in Table 2, with corresponding rewrites in Table 3. The following comments ought to be taken in conjunction with the figures in these Tables. First, as in [BCS92] we have a set of n-ary "duplication" nodes, which take a single wire (which may not be in "linear position", which is to say that it may not be the right-most wire in the graph) as input, and produce an *unordered* set of n output wires, all bearing the same label as the input. In Table 2 we illustrate only the binary case of this duplication node. In Table 3 we have a number of rewrites involving duplication; although we illustrate only the binary cases, the reader ought to keep in mind that there are similar rewrites for the n-ary versions. (Actually, one could make do with a binary node only, and use the binary rewrites to simulate the n-ary nodes, but it is simpler to use n-ary nodes.) Note that one rewrite is written as an equality: this emphasizes that the outputs are considered as unordered.

We shall not give a net criterion for graphs created from these links; rather, we shall suppose that nets are created inductively, following the sequentialization steps illustrated in Table 4. We have no boxes for the circuit links, though this might be a simple way to keep track of the linear and classical wires. Apart from this—admittedly crucial—matter, these circuits are quite similar

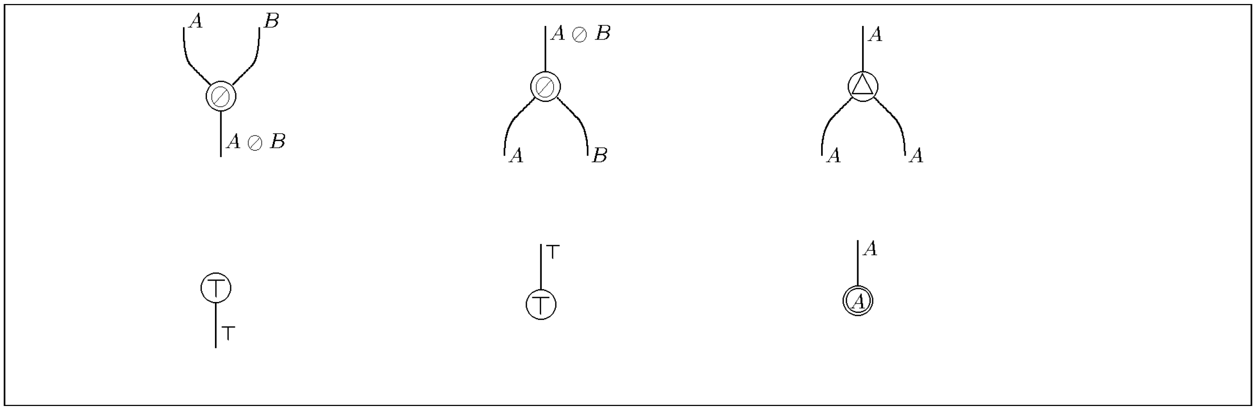


Table 2: Links for proof circuits

to ordinary proof nets, and so the usual acyclic and connectedness criterion would apply. But it is vital to ensure that the distinction between classical and linear wires is maintained. This can be done by giving the wires "weights", but we have not done this in order to keep the graphs simple. Rather than box every rule to guarantee correctness, we prefer to box none. Curiously, if we were to add linear connectives, such as $\otimes$, it seems we would need boxes for these, dual to the usual situation with linear logic proof nets. The resulting situation has a close connection with Milner's action structures, a connection we hope to explore in a sequel. Unlike [BCS92] we have no need for thinning links, since we have just the one "tensor" operator. In a full calculus that allowed a dual context "tensor" $\obslash$ (corresponding to **?** in linear logic), we would need to introduce thinning links.

Note that in these circuits we can have no "empty wires"—if the linear position in a sequent is indeed empty, we use a $\top$ wire to represent this. In this way, the $(\top L)$ rule is embedded in the notation, and so does not appear in the sequentialization steps. Wires can be "grounded" (*i.e.* disappear): this is indicated by a node that has only one wire attached to it. Note that we have no node or link for the exchange rule; instead we shall let wires cross each other where appropriate. These crossings may be moved about, treating the circuits as string diagrams. Such a crossing may not involve the rightmost wire, which is likewise blocked from other "classical" operations such as duplication and thinning. This should be clear from the sequentialization steps of Table 4.

It is of interest to note that dereliction is formally very similar in this circuit presentation to thinning. Indeed, it is a form of "linear thinning" for $\top$, corresponding to the "classical thinning" for any formula, in that it allows a $\top$ wire to be grounded in linear position, just as thinning allows any wire to be grounded in classical position.

To illustrate these circuits and the rewrites, we show that in the category of circuits the commutative diagrams in the definition of contextual categories do indeed commute. This shows that this circuit category is indeed contextual. In fact it is the free such contextual category. This can be generalized as in [BCST], presenting the free contextual category generated by a given set of components (essentially a given graph) together with a set of equivalences, as a suitable category of circuits. Note that in several instances the relevant morphisms correspond to equal circuits, because of our handling of exchange as wire crossing and of the circuits as string diagrams (see Figure 3 for diagrams 12 and 13 for example).

Then, as in [BCST] we can derive a decision procedure for equality of maps in the free category;

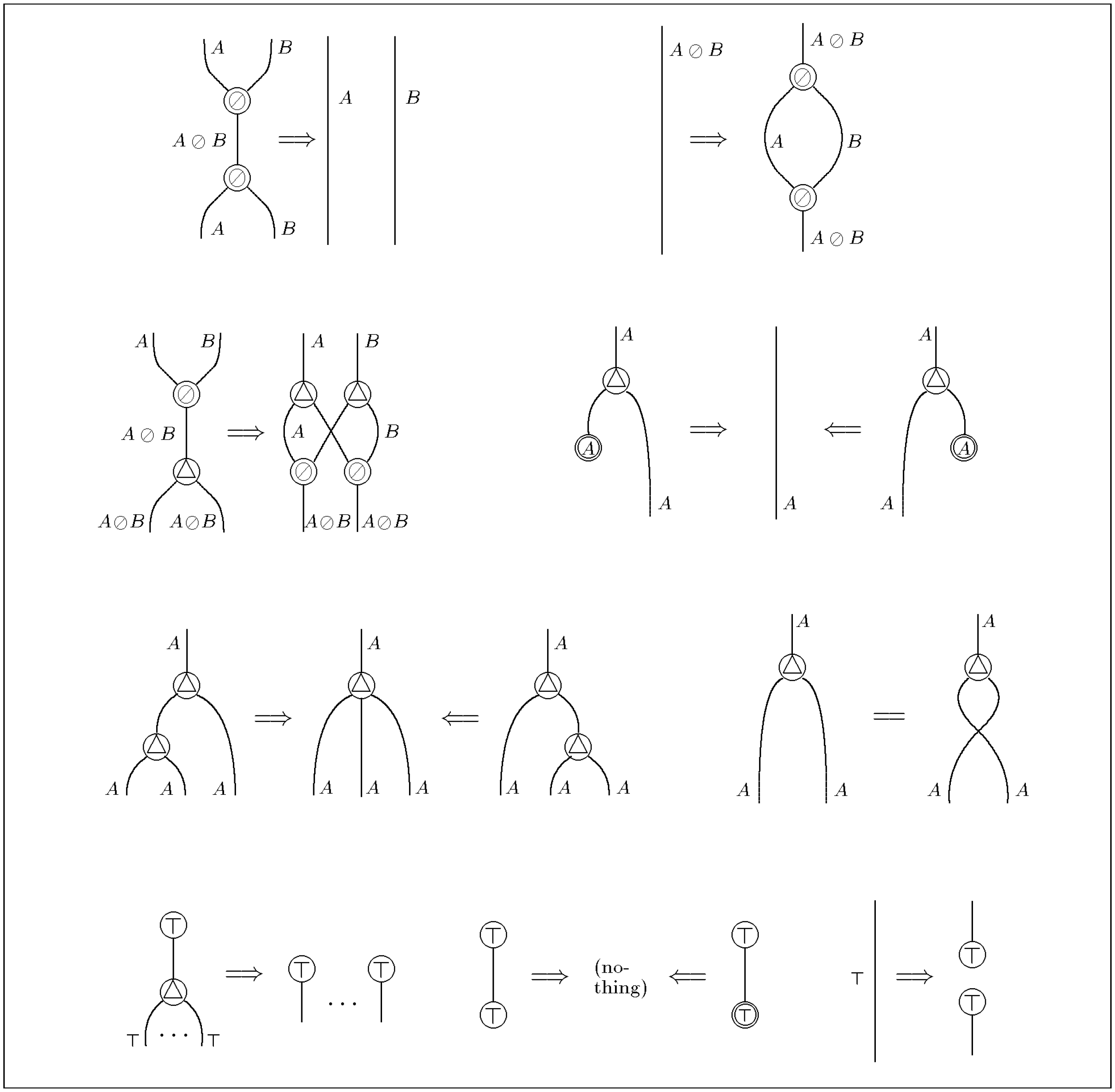


Table 3: Reductions and expansions for proof circuits

two morphisms are equal if their expanded normal forms are equal. In fact, in this case this is quite trivial due to not having thinning links, and essentially amounts to the traditional technique of Kelly–Mac Lane graphs, as shown in [B92], where the coherence question for various theories of monoidal categories was solved using this approach.

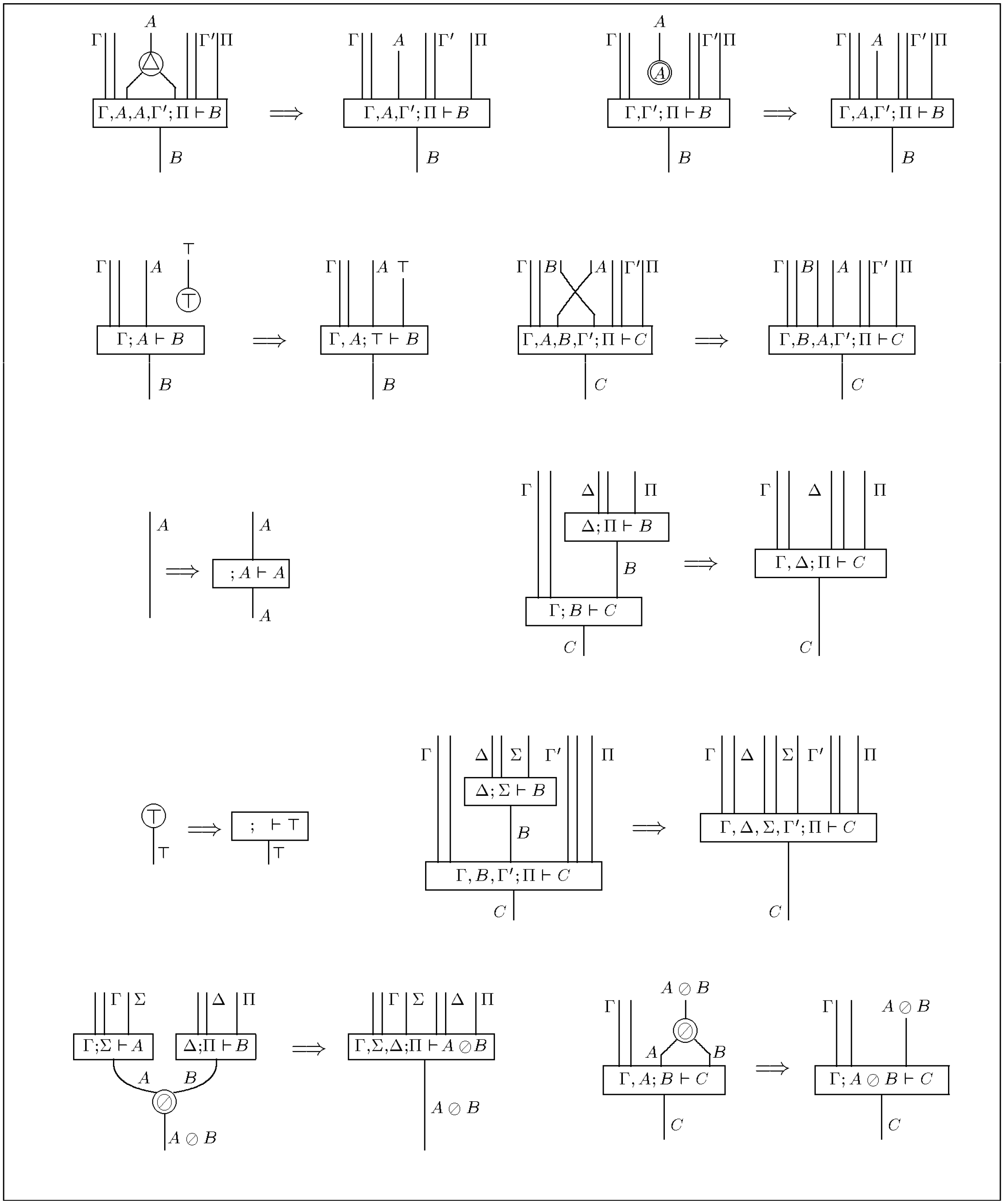
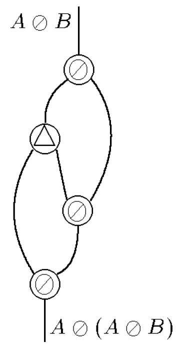
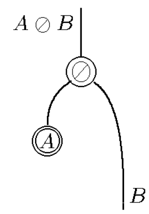
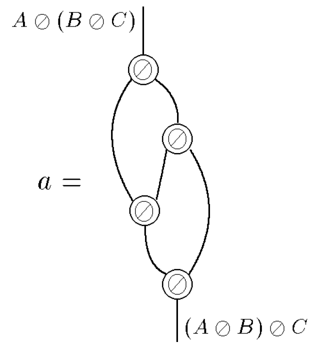


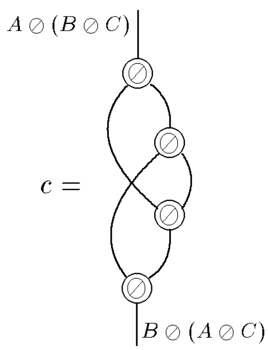
Table 4: Sequentialization

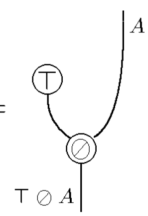
We begin with the circuits for the morphisms $d, e, a, c,$ lift and read:

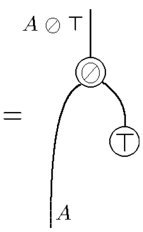
$d =$ 

$e =$ 

$a =$ 

$c =$ 

lift $=$ 

read $=$ 

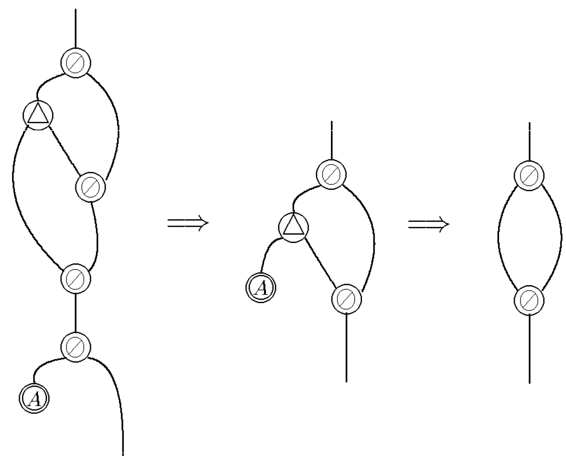


Diagram 1: $d; e = 1$

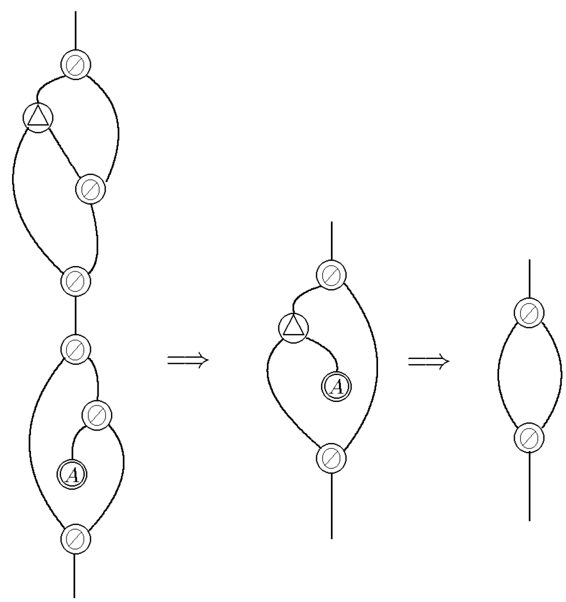


Diagram 2: $d; 1 \oslash e = 1$

Figure 1: Commutative diagrams

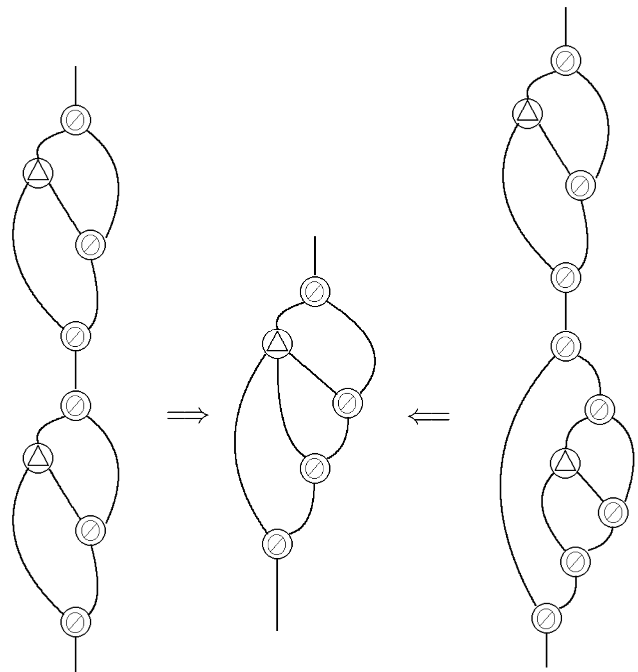


Diagram 3: $d; d = d; 1 \oslash d$

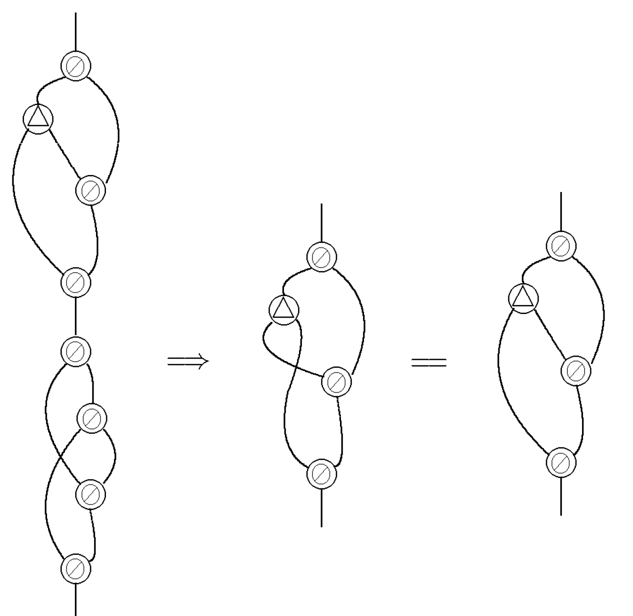


Diagram 8: $d; c = d$

Figure 2: Commutative diagrams

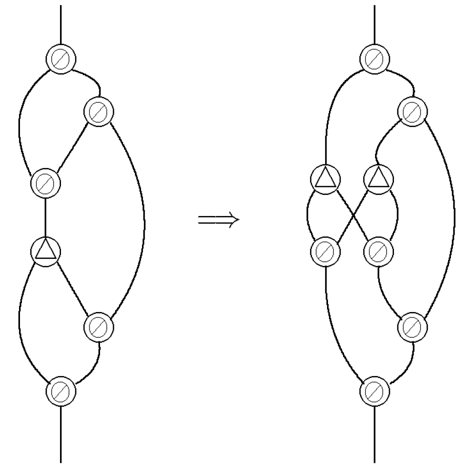


Diagram 11: $a; d = 1 \oslash d; d; 1 \oslash c; a; 1 \oslash a$

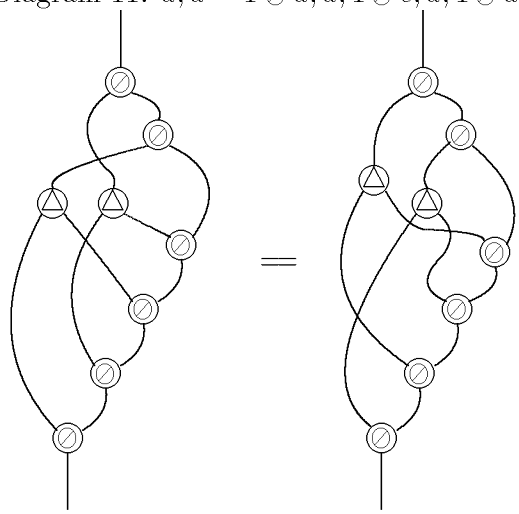


Diagram 12: $c; 1 \oslash d; d; 1 \oslash c = 1 \oslash d; d; 1 \oslash c; c; 1 \oslash (1 \oslash c)$

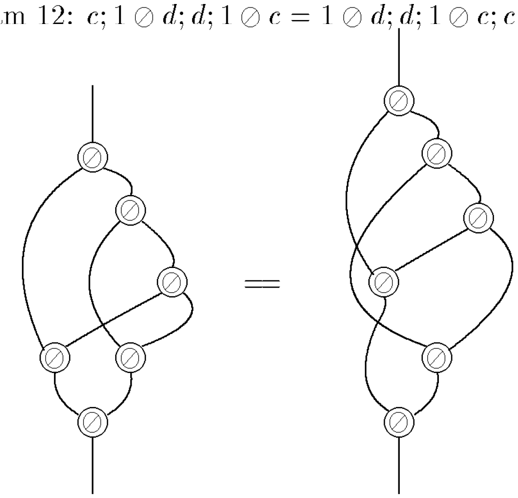


Diagram 13: $1 \oslash c; a = c; 1 \oslash a; c$

Figure 3: Commutative diagrams

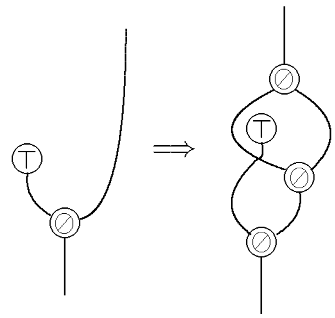


Diagram 14: $\mathrm{lift} = 1 \oslash \mathrm{lift}; c$

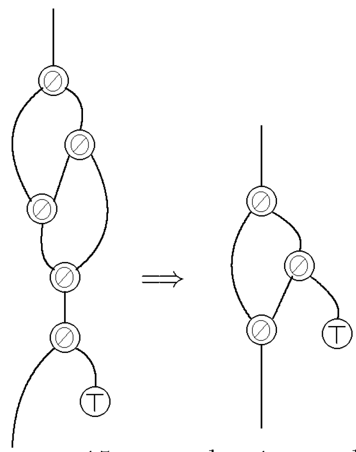


Diagram 15: $a; \mathrm{read} = 1 \oslash \mathrm{read}$

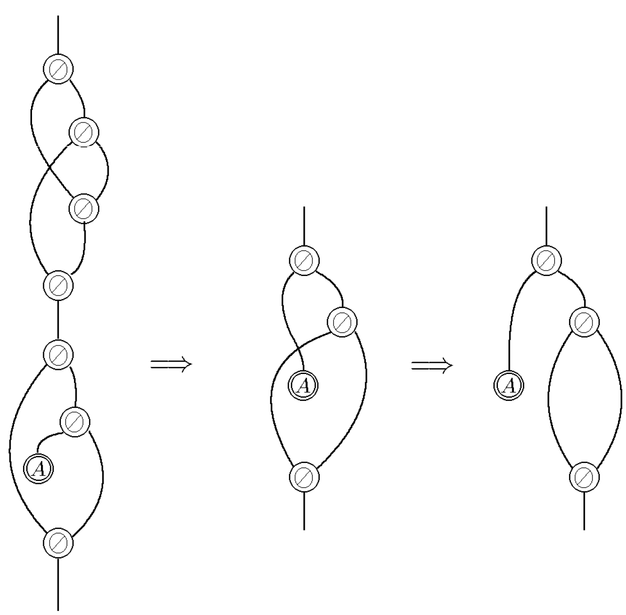


Diagram 17: $c; 1 \oslash e = e$

Figure 4: Commutative diagrams

8 Future work

In a sequel to this paper we shall extend the structure of context categories to include a tensor product, and then dualize this to obtain categories with context and cocontext (and tensor and par). We can sketch a part of this here, but full details must await the sequel.

If we were to add a tensor product to a contextual category at the very least we should like it to be a contextually strong bifunctor . This introduces the strength maps:

$$\theta_{\top} : X \oslash \top \longrightarrow \top$$

$$\theta_{\_\otimes\_} : X \oslash (Y \otimes Z) \longrightarrow (X \oslash Y) \otimes (X \oslash Z)$$

The former map has to coincide with elimination (to satisfy the elimination diagram). For the latter we may again break this down into two components:

$$\mathrm{fst} = \theta_{\_\otimes Z} : X \oslash (Y \otimes Z) \longrightarrow (X \oslash Y) \otimes Z$$

$$\mathrm{snd} = \theta_{Y\otimes\_} : X \oslash (Y \otimes Z) \longrightarrow Y \otimes (X \oslash Z)$$

In addition, we shall see that it is desirable that these two components are isomorphisms. For example, we can recapture the categorical semantics of tensor and $\mathbf{!}$: contextual categories with a contextually strong tensor product whose linear strengths are isomorphisms are very nearly "linear categories" in the sense of being monoidal categories with a storage operator $\mathbf{!}$ (as defined more or less in [BBPH] or [BCS92]).

The diagrams which arise from the assumption of contextual strength include those which make these natural transformations strong, for example:

$$\begin{tikzcd} X \oslash Y \arrow[r, "1 \oslash u^R_\otimes"] \arrow[dr, "u^R_\otimes"'] & X \oslash (Y \otimes \top) \arrow[d, "\mathrm{fst}"] \\ & (X \oslash Y) \otimes \top \end{tikzcd} \qquad \begin{tikzcd} X \oslash Y \arrow[r, "1 \oslash u^L_\otimes"] \arrow[dr, "u^L_\otimes"'] & X \oslash (\top \otimes Y) \arrow[d, "\mathrm{snd}"] \\ & \top \otimes (X \oslash Y) \end{tikzcd}$$

There is a similar diagram to show that the associative isomorphism of the tensor is strong. These diagrams are used in the proof of the next lemma. Observe that $\top$ is trivially a comonoid so that $X \oslash \top$ is a comonoid by strengthening the comultiplication of $\top$ and using $e : X \oslash \top \longrightarrow \top$ as the counit.

Proposition 8.1 *In any contextual category with a contextually strong tensor product if Y is a comonoid then $X \oslash Y$ is a comonoid.*

In other words the category of comonoids of a such contextual category is a module. This allows the observation:

Corollary 8.2 *In any contextual category with a contextually strong tensor product in which $\top$ is a (commutative) comonoid, the storage functor $\mathbf{!}(\_)$ carries objects onto (commutative) comonoids.*

To dualize these notions, it becomes necessary to consider the structure of a *fibrational fork* (this notion is due to Benabou—we learned of it from Bart Jacobs). The point then is that a category $\mathbf{X}$

is bicontextual if it is a bimodule over itself, where here a bimodule in general must have both an $\mathbf{X}$–contextual action and a $\mathbf{X}'$–cocontextual action (with $\mathbf{X}'$ being a cocontextual category). The strength and costrength maps $X \oslash (Y \otimes X') \longrightarrow (X \oslash Y) \otimes X'$ are identified (analogous to the weakly distributive case, where the weak distributions are simultaneously strength and costrength). The result gives a fibrational fork. Once tensor and cotensor ("par") are added we recover the notion of a weakly distributive category with storage [BCS92].

References

[BBPH] Benton B.N., G. Bierman, V. de Paiva, M. Hyland "Term assignment for intuitionistic linear logic", M. Bezem and J. F. Groote, eds, *Proceedings of the International Conference on Typed Lambda Calculi and Applications*, 1992, Springer Lecture Notes in Computer Science **664**, 75–90.

[B92] Blute, R.F. "Linear Logic, Coherence and Dinaturality", Theoretical Computer Science **115** (1992) 3–42.

[BCS92] Blute, R.F., J.R.B. Cockett, and R.A.G. Seely "**!** and **?**: Storage as tensorial strength", to appear in Mathematical Structures in Computer Science, (Preprint, McGill University, 1992, revised 1994).

[BCST] Blute R.F., J.R.B. Cockett, R.A.G. Seely, and T.H. Trimble "Natural deduction and coherence for weakly distributive categories", to appear in Journal of Pure and Applied Algebra, (Preprint, McGill University, 1992, revised 1994).

[Bo94] Borceux, F. *Handbook of Categorical Algebra*, Cambridge University Press, 1994.

[Ca86] Cartmell, J. "Generalized Algebraic Theories and Contextual Categories", Annals of Pure and Applied Logic, **32** (1986), 209–243

[Co91] Cockett J.R.B., D.L. Spencer "Strong categorical datatypes I", in R.A.G. Seely, ed., *Category Theory 1991, Montreal*, CMS Conference Proceedings, **13** (1991), 141–169.

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

[G93] Girard, J.-Y. "On the unity of logic", Annals of Pure and Applied Logic **59** (1993) 201–217.

[M93] Milner, R. "Action calculi, or concrete action structures", Mathematical Foundations of Computer Science, Springer Lecture Notes in Computer Science **711** (1993), 105–121.

[PR89] Paré, R. and Román, L. "Monoidal categories with natural numbers object", Studia Logica XLVIII (1989) 361–376.

[Se89] Seely, R.A.G. "Linear logic, $*$-autonomous categories and cofree coalgebras", in J. Gray and A. Scedrov (eds.), *Categories in Computer Science and Logic*, Contemporary Mathematics **92** (Am. Math. Soc. 1989).

[W94] Wadler, P. "A Syntax for Linear Logic", Springer Lecture Notes in Computer Science **802** (1994), 513–528.