Collages of String Diagrams

Dylan Braithwaite
University of Strathclyde
dylan.braithwaite@strath.ac.uk

Mario Román
Tallinn University of Technology
mroman@ttu.ee

We introduce collages of string diagrams as a diagrammatic syntax for glueing multiple monoidal categories. Collages of string diagrams are interpreted as pointed bimodular profunctors. As the main examples of this technique, we introduce string diagrams for bimodular categories, string diagrams for functor boxes, and string diagrams for internal diagrams.

1 Introduction

String diagrams are a convenient and intuitive, sound and complete syntax for monoidal categories [29]. Monoidal categories are algebras of processes composing in parallel and sequentially [32]; string diagrams formalize the process diagrams of engineering [6, 8]. Formalization is not only of conceptual interest: it means we can sharpen our reasoning, scale our diagrams, or explain them to a computer [40].

However, the formal syntax of monoidal categories is not enough for all applications and, sometimes, we need to extend it. Functor boxes allow us to reason about translations between theories of processes [15, 35], ownership [37], higher-order processes [1], or programming effects [41]. Quantum combs not only model some classes of supermaps [12, 16, 23], but they coincide with the monoidal lenses of functional programming [5, 13, 48] and compositional game theory [22, 7]. Premonoidal categories, which appear in Moggi's semantics of programming effects [36, 30, 49], are now within the realm of string diagrammatic reasoning [45]. Internal diagrams extend the syntax of monoidal categories allowing us to draw diagrams inside tubular cobordisms and reason about topological quantum field theories [3], but also coends [44] and traces [26].

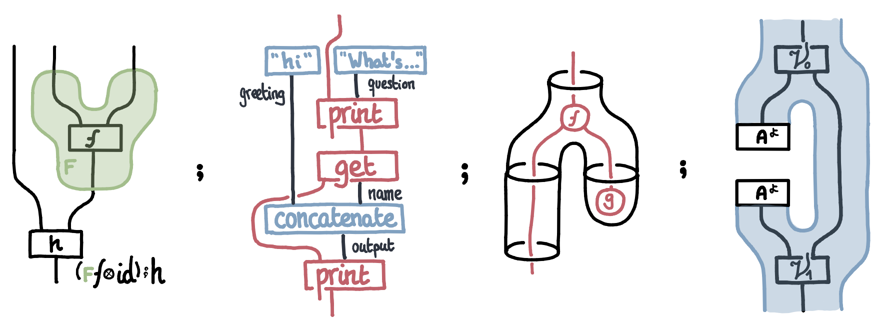


Figure 1: Examples from the literature. From left to right: functor boxes [35], premonoidal categories [45], internal diagrams [3], and combs or optics [12, 13, 23].

The extensions showcase the expressive power of string diagrams on surprisingly diverse application domains. At the same time, these different ideas could be regarded as separate ad-hoc extensions: they belong to different fields; they use different categorical formalisms. The overhead of learning and combining each one of them prevents the exchange of ideas between the different domains of application: e.g. an idea about topological quantum field diagrams does not transfer to premonoidal diagrams.

Submitted to:
ACT 2023

Collages. This manuscript claims that this division is only apparent and that all these extensions are particular instances of the same encompassing idea: that of glueing multiple string diagrams into what we call a *collage of string diagrams*. We introduce a formal notion of collage (Section 4.4) and employ string diagrammatic syntaxes for them, based on the calculus of bicategories (Sections 2.1, 3.1 and 5).

Even though collages of string diagrams are our novel contribution, collages are not yet another new concept to category theory. "Collage" was Bob Walters' term for a lax colimit in a module-like category [50]. This can be considered as a glueing of objects together along the action of a scalar. For example, given two sets A and B, with an action of a monoid M, we can construct their tensor product $A \otimes_M B$, where $(a \cdot m) \otimes b = a \otimes (m \cdot b)$ for any scalar $m \in M$. Categorifying this idea in a possible direction we obtain monoidal categories acting on *bimodular categories*. The following is the takeaway of this work.

> Collages of string diagrams consist of multiple string diagrams of different monoidal categories glued together. Collages can be interpreted as *pointed bimodular profunctors* between *bimodular categories*.

A bimodular category, sometimes referred to as a biactegory [10], is to a bimodule what a monoidal category is to a monoid. This is, a plain category $\mathbb{A}$ endowed with a left action of a monoidal category $(\triangleright) \colon \mathbb{M} \times \mathbb{A} \to \mathbb{A}$ and a right action of another, possibly different, monoidal category $(\triangleleft) \colon \mathbb{A} \times \mathbb{N} \to \mathbb{A}$. We can *collage* two bimodular categories along a common monoidal category that acts on both. Later on the paper, exploiting a second axis of categorification, we pass from bimodular categories to *bimodular profunctors*, which are a kind of 2-dimensional bimodule, and we define their collage. This structure facilitates glueing categories together in 2-dimensions: we can represent complexes of morphisms from different categories and glue them together. Collages of string diagrams are the syntactic representations of this glueing, in the same sense that ordinary string diagrams represent tensors in monoidal categories.

We observe that collages of bimodular categories embed into a tricategory of pointed bimodules. This provides a versatile setting where we can interpret many syntaxes already present in the literature.

Contributions. We introduce string diagrams of bimodular categories and we prove they construct the free bimodular category on a signature (Theorem 2.7). We introduce novel string diagrammatic syntax for *functor boxes* and we prove it constructs the free lax monoidal functor on a suitable signature (Theorem 3.4). We describe the tricategory of pointed bimodular profunctors (Definition 4.6) and, in terms of it, we explain the semantics of functor boxes (Proposition 4.9) and internal diagrams (Theorem 5.3), for which we also provide a novel explicit formal syntax (Definition 5.2).

2 String Diagrams of Bimodular Categories

This section introduces string diagrams for bimodular categories in terms of the better known string diagrams of bicategories. In algebra, a *bimodule* is a structure that has both a left and a right action such that they are compatible. *Bimodular categories are to bimodules what monoidal categories are to monoids*. This means that a bimodular category is a category, $\mathbb{C}$, acted on by two monoidal categories, $\mathbb{M}$ and $\mathbb{N}$ [51]. Bimodular categories are also known as "biactegories" [10, 34], while the name "bimodule category" has been reserved for actions of vector enriched categories with extra properties [19]. For our purposes, a bimodular category, $\mathbb{C}$, glues the string diagrams of their two monoidal categories, $\mathbb{M}$ and $\mathbb{N}$.

Definition 2.1. A *bimodular category* $(\mathbb{C}, \mathbb{M}, \mathbb{N})$ is a category $\mathbb{C}$ endowed with a left monoidal action $(\triangleright) \colon \mathbb{M} \times \mathbb{C} \to \mathbb{C}$, and a right monoidal action $(\triangleleft) \colon \mathbb{C} \times \mathbb{N} \to \mathbb{C}$. These two actions must be compatible, meaning that there exists a natural isomorphism, $\gamma_{M,N,X} : M \triangleright (X \triangleleft N) \longrightarrow (M \triangleright X) \triangleleft N$, such that all formal equations between these isomorphisms and the coherence isomorphisms of both monoidal categories and monoidal actions hold.

A bimodular category is a *strict bimodular category* whenever the two monoidal categories are strict, their two actions are strict and, moreover, the compatibility isomorphism is an identity. Every monoidal category $(\mathbb{C}, \otimes, I)$ is a $(\mathbb{C}, \mathbb{C})$-bimodular category with its own tensor product defining the two actions.

Proposition 2.2. *Strict bimodular categories over arbitrary strict monoidal categories form a category,* **sBimod**. *Morphisms* $(F,H,K)\colon (\mathbb{C}, \mathbb{M}, \mathbb{N}) \to (\mathbb{D}, \mathbb{P}, \mathbb{Q})$ *consist of two strict monoidal functors* $H\colon \mathbb{M} \to \mathbb{P}$ *and* $K\colon \mathbb{N} \to \mathbb{Q}$ *and a functor* $F\colon \mathbb{C} \to \mathbb{D}$ *that strictly preserves monoidal actions according to* H *and* K.

2.1 Signature of a Bimodular Category

The next sections prove that a variant of string diagrams is a sound and complete syntax for bimodular categories. String diagrams for bimodular categories consist of two monoidal regions glued by a bimodular wire. We first introduce a notion of bimodular signature (Definition 2.3) and then construct an adjunction (Theorem 2.8) using the notion of *collages*.

Definition 2.3. A *bimodular graph* $(\mathscr{A}, \mathscr{M}, \mathscr{N})$ (the bimodular analogue of a multigraph [46]) is given by three sets of objects $(\mathscr{A}_{obj}, \mathscr{M}_{obj}, \mathscr{N}_{obj})$ and three different types of edges:

- the left-acting edges, a set $\mathscr{M}(M_0, \dots, M_m; P_0, \dots, P_p)$ for each $M_0, \dots, M_m, P_0, \dots, P_p \in \mathscr{M}_{obj}$; and
- the right-acting edges, a set $\mathscr{N}(N_0, \dots, N_n; Q_0, \dots, Q_q)$ for each $N_0, \dots, N_n, Q_0, \dots, Q_q \in \mathscr{N}_{obj}$;
- the *central edges*, a set of edges $\mathscr{A}(M_0, \dots, M_m, A, N_0, \dots, N_n; O_0, \dots, P_p, B, Q_0, \dots, Q_q)$, for each $M_0, \dots, M_m, P_0, \dots, P_p \in \mathscr{M}_{obj}$; each $N_0, \dots, N_n, Q_0, \dots, Q_q \in \mathscr{N}_{obj}$ and each $A, B \in \mathscr{A}_{obj}$.

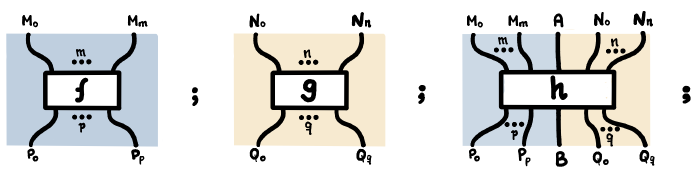


Figure 2: Left, right, and central edges of a bimodular graph.

Proposition 2.4. *Bimodular graphs form a category* **bmGraph**. *We define a morphism of bimodular graphs* $(l, f, g)\colon (\mathscr{A}, \mathscr{M}, \mathscr{N}) \to (\mathscr{A}', \mathscr{M}', \mathscr{N}')$ *to be a triple of functions on objects,* $(l_{obj}, f_{obj}, g_{obj})$*, that extend to the morphism sets. There exists a forgetful functor* $U\colon$ **sBimod** $\to$ **bmGraph**.

Proof. See Appendix, Proposition B.3. □

So far we have described a syntactic presentation of strict bimodular categories. We would like to, additionally, go the other way and construct a free model from a syntactic presentation. Our approach is to note that the central edges in a bimodular graph can be considered as dividing the graph into two regions: one containing the left-acting vertices and edges and one containing the right-acting vertices and edges. Diagrams of this sort with multiple labelled regions can naturally be considered as string diagrams for bicategories: explicitly, the diagrams of the *collage* of the bimodular category.

2.2 The Collage of a Bimodular Category

Each profunctor induces a *collage category*; in an analogous fashion, a bimodular category induces a *collage bicategory*. This section proves that constructing the collage of a bimodular category is left adjoint to considering the bimodular hom-category between any two cells of a 2-category.

Definition 2.5. The *collage* of an $(\mathbb{M},\mathbb{N})$-bimodular category $\mathbb{C}$ is a bicategory, $\mathsf{Coll}_{\mathbb{C}}$. This bicategory has two 0-cells, M and N, and it is defined by the following hom-categories. Endocells on M are given by the monoidal category $\mathsf{Coll}_{\mathbb{C}}(M,M) = \mathbb{M}$; likewise, endocells on N are given by the monoidal category, $\mathsf{Coll}_{\mathbb{C}}(N,N) = \mathbb{N}$. The 1-cells from M to N are given by the category $\mathsf{Coll}_{\mathbb{C}}(M,N) = \mathbb{C}$; and composition of 1-cells is given by the monoidal actions. Finally, $\mathsf{Coll}_{\mathbb{C}}(N,M)$ is the empty category.

Definition 2.6. The category of strict bipointed 2-categories, $\mathbf{2Cat}_2$, has objects, $(\mathbb{A},M,N)$, given by a strict 2-category $\mathbb{A}$ and two chosen 0-cells on it, $M \in \mathbb{A}$ and $N \in \mathbb{A}$. A morphism of bipointed 2-categories is a strict 2-functor preserving the two chosen 0-cells.

Theorem 2.7. *There exists an adjunction between strict bimodular categories and bipointed 2-categories given by the collage,* $\mathsf{Coll}_{\mathbb{C}} \colon \mathbf{sBimod} \to \mathbf{2Cat}_2$, *and picking the hom-category between the chosen 0-cells,* $\mathsf{Chosen} \colon \mathbf{2Cat}_2 \to \mathbf{sBimod}$. *Moreover, the unit of this adjunction is a natural isomorphism.*

Proof. See Appendix, Theorem B.7. □

2.3 String Diagrams of Bimodular Categories, via Collages

We have the two ingredients for string diagrams of bimodular categories: string diagrams for bicategories, and collages, a way of embedding a bimodular category into a bicategory. This section combines both results to provide an adjunction from bimodular graphs to bimodular categories.

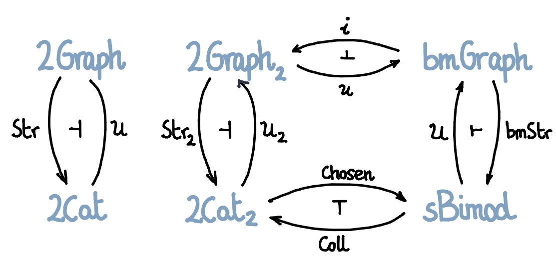


Figure 3: Summary of adjunctions for the string diagrams of bimodular categories.

Theorem 2.8. *There exists an adjunction between bimodular graphs and strict bimodular categories. The left side of this adjunction is given by finding the bimodular category whose collage is the free 2-category on the bimodular graph,* $\mathsf{bmStr} \colon \mathbf{bmGraph} \to \mathbf{sBimod}$. *The right side of the adjunction is the previously mentioned forgetful functor* $\mathsf{U} \colon \mathbf{sBimod} \to \mathbf{bmGraph}$.

Proof. See Appendix, Theorem B.8, the proof follows Figure 3. □

Remark 2.9. The string diagrams of bimodular categories particularize into the string diagrams of premonoidal and effectful categories. See the Appendix B.2 for details.

We have presented string diagrams for bimodular categories via the string diagrams of bicategories, and we will now give an example. We take inspiration from this first result to address now other syntaxes that depend on string diagrams of bicategories: the next section proposes string diagrams for functor boxes.

2.4 Example: Shared State

In the same way that premonoidal categories are particularly well-suited to describe stateful computations, bimodular categories are particularly well-suited to describe shared state between two processes. These two processes can be different and even live on different categories. As an example, consider the generators in Figure 4. They represent two different process theories (two different monoidal categories, $\mathbb{A}$ and $\mathbb{B}$) that access a common state with get and put operations.

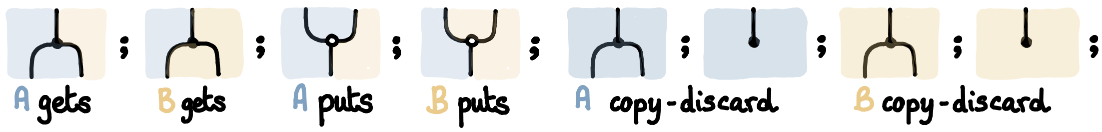


Figure 4: Signature generators for bimodular theory of shared state.

In the same way that monoidal categories are a good setting where to define monoids and comonoids, bimodular categories are a good setting where to define bimodules. In order to capture interacting shared state, the generators of Figure 4 are quotiented by the equations of a pair of semifrobenius modules with compatible comonoid actions and semimonoid actions (see Appendix, Figure 15, for details).

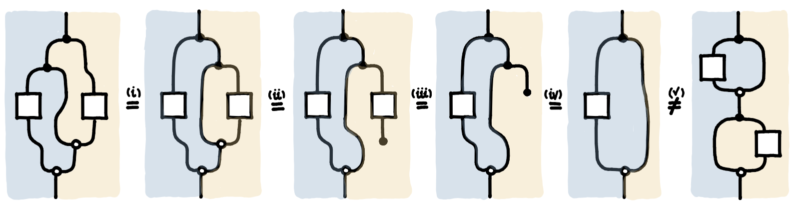


Figure 5: Race condition in bimodular string diagrams.

This setup is enough to exhibit one of the most salient features of shared state: *race conditions*. Race conditions were first studied by Huffman in 1954, who used diagrams to show how the behaviour of shared state is dependent on the relative timing of the actions of the parties [27]. We employ string diagrams of bimodular categories to show how two different timings of the actions – the leftmost and rightmost sides of the equation in Figure 5 – result in two different executions: even when the two get statements are compatible *(i)*, the two put statements interact causing the earlier of the two to be discarded *(ii,iii,iv)*; this causes the discrepancy with the intended protocol *(v)*.

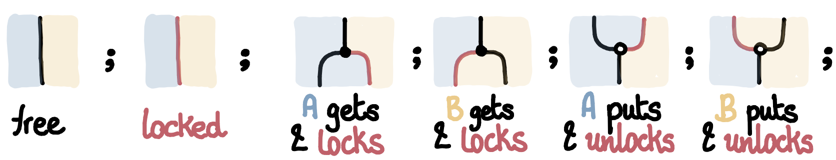


Figure 6: Binary semaphore in bimodular string diagrams.

Race conditions have a commonly accepted workaround: the *binary semaphore* [47]. Dijkstra described general semaphores with the aid of flow diagrams [18]; we use instead string diagrams of bimodular categories to implement a binary semaphore (Figure 6). We consider two different object generators for our bimodular category (free and locked): each operation must suitably lock or unlock the semaphore. This renders race conditions ill-typed, and renders most of the interaction equations unnecessary (in the Appendix, Figure 15).

Remark 2.10. String diagrams of bimodular categories model a pair of interacting monoidal categories. However, we can also model an arbitrary number of interacting monoidal categories via a general collage construction (see Appendix F). These collages can be subsumed into a 'universe of collages' that we will construct in Section 4: the tricategory of pointed bimodular profunctors. To argue for its need, we first introduce a second example: the syntax of functor boxes.

3 String Diagrams of Functor Boxes

Functor boxes are a extension of the string diagrammatic notation that represents plain functors, lax, oplax and strong monoidal functors. Functor boxes were introduced by Cockett and Seely [15] and later studied by Melliès [35]. We introduce here a syntactic presentation of (op)lax functor boxes that has the advantage of treating each piece of the box as a separate entity in a bicategory and apply the string diagrammatic calculus of bicategories.

3.1 Functor box signatures

Definition 3.1. A *functor box signature* $\mathscr{F} = (\mathscr{A}, \mathscr{X}, \mathscr{F}_\bullet, \mathscr{F}^\bullet)$ consists of a pair of sets, $\mathscr{A}_{obj}$ and $\mathscr{X}_{obj}$, and four different types of edges:

- the plain edges, $\mathscr{A}(A_0,\dots,A_n;B_0,\dots,B_m)$ for any objects $A_0,\dots,A_n,B_0,\dots,B_m \in \mathscr{A}_{obj}$;
- the functor box edges, $\mathscr{X}(X_0,...,X_n;Y_0,...,Y_m)$ for any objects $X_0,\dots,X_n,Y_0,\dots,Y_m \in \mathscr{X}_{obj}$;
- the in-box edges, $\mathscr{F}_\bullet(A_0,...,A_n;Y_0,...,Y_m)$ for any $A_0,...,A_n \in \mathscr{A}_{obj}$ and $Y_0,...,Y_m \in \mathscr{X}_{obj}$
- the out-box edges, $\mathscr{F}^\bullet(X_0,...,X_n;B_0,...,B_m)$ for any $B_0,...,B_m \in \mathscr{A}_{obj}$ and $X_0,...,X_n \in \mathscr{X}_{obj}$.

A *functor box signature morphism* $(h,k,l)\colon (\mathscr{A}, \mathscr{X}, \mathscr{F}) \to (\mathscr{B}, \mathscr{Y}, \mathscr{G})$ is a pair of functions between the object sets, $h_{obj}\colon \mathscr{A}_{obj} \to \mathscr{B}_{obj}$ and $k_{obj}\colon \mathscr{X}_{obj} \to \mathscr{Y}_{obj}$, that extend to a function between the edge sets;

- $h\colon \mathscr{A}(A_0,...,A_n;B_0,...,B_m) \to \mathscr{B}(h(A_0),...,h(A_n);h(B_0),...,h(B_m))$;
- $k\colon \mathscr{X}(X_0,...,X_n;Y_0,...,Y_m) \to \mathscr{Y}(k(X_0),...,k(X_n);k(Y_0),...,k(Y_m))$;
- $l_\bullet\colon \mathscr{F}_\bullet(A_0,...,A_n;Y_0,...,Y_m) \to \mathscr{G}_\bullet(h(A_0),...,h(A_n);k(Y_0),...,k(Y_m))$;
- $l^\bullet\colon \mathscr{F}^\bullet(X_0,...,X_n;B_0,...,B_m) \to \mathscr{G}^\bullet(k(X_0),...,k(X_n);h(B_0),...,h(B_m))$.

Functor box signatures and homomorphisms form a category, **Fbox**.

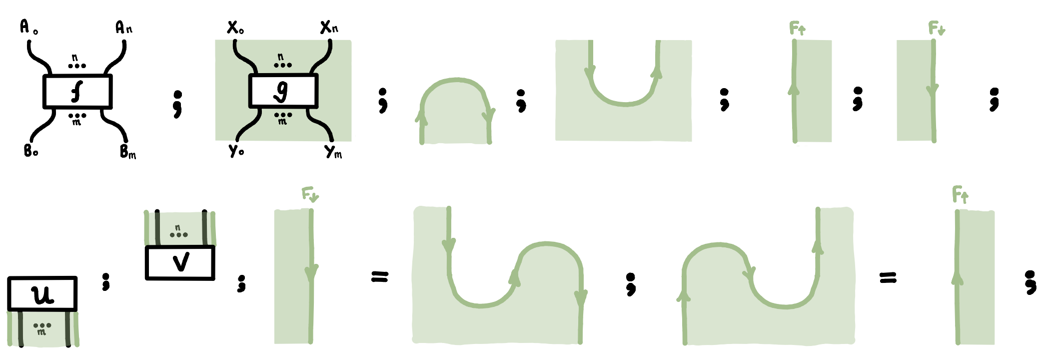


Figure 7: Syntactic bicategory of a lax monoidal functor box signature.

Definition 3.2. The syntactic bicategory of a functor box signature $\mathscr{F} = (\mathscr{A}, \mathscr{X}, \mathscr{F}_{\bullet}, \mathscr{F}^{\bullet})$ is the bicategory freely presented by Figure 7, which we call $\mathbb{S}_{\mathscr{A},\mathscr{X},\mathscr{F}}$.

In other words, the bicategory $\mathbb{S}_{\mathscr{A},\mathscr{X},\mathscr{F}}$ contains exactly two 0-cells, labelled $\mathscr{A}$ and $\mathscr{X}$; it contains a 1-cell $A \colon \mathscr{A} \to \mathscr{A}$ for each $A \in \mathscr{A}_{obj}$, a 1-cell $X \colon \mathscr{X} \to \mathscr{X}$ for each $X \in \mathscr{X}_{obj}$ and, moreover, a pair of adjoint 1-cells $F^{\uparrow} \colon \mathscr{A} \to \mathscr{X}$ and $F^{\downarrow} \colon \mathscr{X} \to \mathscr{A}$. Finally, it contains a pair of 2-cells witnessing the adjunction $F^{\uparrow} \dashv F^{\downarrow}$, given by $n \colon \mathrm{id} \to F^{\uparrow} ⨾ F^{\downarrow}$ and $e \colon F^{\downarrow} ⨾ F^{\uparrow} \to \mathrm{id}$ which additionally satisfy the snake equations; and it also contains

- a 2-cell, $f \in \mathbb{S}(\mathscr{A},\mathscr{A})(A_0 ⨾ \dots ⨾ A_n;\ B_0 ⨾ \dots ⨾ B_m)$, for each *plain edge*;
- a 2-cell, $g \in \mathbb{S}(\mathscr{X},\mathscr{X})(X_0 ⨾ \dots ⨾ X_n;\ Y_0 ⨾ \dots ⨾ Y_m)$, for each *functor box edge*;
- a 2-cell, $u \in \mathbb{S}(\mathscr{A},\mathscr{A})(A_0 ⨾ \dots ⨾ A_n;\ F^{\uparrow} ⨾ Y_0 ⨾ \dots ⨾ Y_m ⨾ F^{\downarrow})$ for each *in-box edge*; and
- a 2-cell, $v \in \mathbb{S}(\mathscr{A},\mathscr{A})(F^{\uparrow} ⨾ X_0 ⨾ \dots ⨾ X_n ⨾ F^{\downarrow};\ B_0 ⨾ \dots ⨾ B_m)$ for each *out-box edge*.

3.2 Lax Monoidal Functor Semantics

Definition 3.3 (Lax functors category)**.** An object of the *lax functors category*, **Lax**, is a pair of strict monoidal categories $(\mathbb{A}, \mathbb{X})$ together with a lax monoidal functor between them, (F, ε, μ); that is, a functor $F \colon \mathbb{X} \to \mathbb{A}$ endowed with two natural transformations $\varepsilon \colon I \to FI$, and $\mu \colon FX \otimes FY \to F(X \otimes Y)$, satisfying associativity $(\mu \otimes id) ⨾ \mu = (id \otimes \mu) ⨾ \mu$, left unitality $(\varepsilon \otimes id) ⨾ \mu = id$ and right unitality $(id \otimes \varepsilon) ⨾ \mu = id$.

A morphism of the *lax functors category*, from $(\mathbb{A}, \mathbb{X}, F, \varepsilon_F, \mu_F)$ to $(\mathbb{B}, \mathbb{Y}, G, \varepsilon_G, \mu_G)$ is a pair of strict monoidal functors $H \colon \mathbb{X} \to \mathbb{A}$ and $K \colon \mathbb{A} \to \mathbb{B}$ such that $F ⨾ K = H ⨾ G$ and such that $K(\varepsilon_F) = \varepsilon_G$ and $K(\mu_F) = \mu_G$.

Theorem 3.4. *There exists an adjunction between the category of functor box signatures,* **Fbox***, and the category of pairs of strict monoidal categories with a lax monoidal functor between them,* **Lax***. The free side of this adjunction is given by the syntax of Figure 7.*

Proof. See Appendix, Theorem C.3. □

Collages, by themselves, explained the 2-region diagrams of bimodular categories; collages will also explain the two-region diagrams of functor boxes in Section 4.5. However, as currently defined, collages are only sufficient to encode the vertical boundaries. In order to additionally represent boundaries along the horizontal axis we can make use of profunctors between bimodular categories and extend our notion of collage to these structures (described in Appendix F). Following this thread we find that collages embed into a tricategory of pointed bimodular profunctors, described in the next section, which we consider a universe of interpretation for all of the graphical theories described.

4 Bimodular Profunctors

Where can we interpret all these string diagrams and provide compositional semantics for them? In this section, we introduce a single structure where all the previous calculi take semantics.

We will need two different ingredients: *coends* and *bimodularity*. Coends and profunctors [31, 32], far from being obscure concepts from category theory, can be seen as the right tool to glue together morphisms from different categories [17, 44]; we follow a explicitly *pointed* version of coend calculus, which keeps track of the transformation between profunctors we are constructing (Section 4.3). In a similar sense, bimodular categories tensor together objects from different monoidal categories. Both ideas combine into the calculus of pointed bimodular profunctors.

4.1 Bimodular Profunctors

Consider $\mathbb{C}$ and $\mathbb{D}$, both $(\mathbb{M},\mathbb{N})$-bimodular categories. A natural notion of morphism between them is a functor $\mathbb{C} \to \mathbb{D}$ which preserves both actions. However, there is another notion of morphism between them, which is a generalisation of a profunctor between categories to this bimodular setting. Bimodular profunctors are a generalized reformulation of the Tambara modules of Pastro and Street [39].

Definition 4.1. Let $\mathbb{M}$ and $\mathbb{N}$ be two monoidal categories and let $\mathbb{C}$ and $\mathbb{D}$ be two $(\mathbb{M},\mathbb{N})$-bimodular categories. A *bimodular profunctor* from $\mathbb{C}$ to $\mathbb{D}$ is a profunctor $T : \mathbb{C}^{op} \times \mathbb{D} \to \mathrm{Set}$ with a natural family of strengths,

$$t_M : T(X,Y) \to T(M \triangleright X, M \triangleright Y), \quad \text{and} \quad t^N : T(X,Y) \to T(X \triangleleft N, Y \triangleleft N),$$

such that the actions are associative, $t_M \mathbin{⨟} t_{M'} = t_{M \otimes M'}$ and $t_N \mathbin{⨟} t_{N'} = t_{N \otimes N'}$, unital $t_I = id$ and $t^I = id$, and compatible, $t_M \mathbin{⨟} t^N = t^N \mathbin{⨟} t_M$, up to the coherence isomorphisms of the monoidal category. See Appendix B for details.

Proposition 4.2. *For any pair of monoidal categories, $\mathbb{M}$ and $\mathbb{N}$, there is a bicategory ${}_{\mathbb{M}}\mathbf{Mod}_{\mathbb{N}}$ of $(\mathbb{M},\mathbb{N})$-bimodular categories, bimodular profunctors, and natural transformations between them.*

These will form the hom-bicategories of the tricategory we later define. The other significant piece of data we require is a family of tensors $\otimes : {}_{\mathbb{M}}\mathbf{Mod}_{\mathbb{N}} \times {}_{\mathbb{N}}\mathbf{Mod}_{\mathbb{O}} \to {}_{\mathbb{M}}\mathbf{Mod}_{\mathbb{O}}$, which we now study.

4.2 Tensor of Bimodular Profunctors

The tensor of bimodular categories is similar to the tensor of modules over a monoid in classical algebra: we consider pairs of elements and we quotient out the action of a common scalar. In this case, the quotienting is substituted by an appropiate structural isomorphism: the *equilibrator*.

Definition 4.3 (Tensor of bimodular categories)**.** Let $\mathbb{C}$ be a $(\mathbb{M},\mathbb{N})$-bimodular category and let $\mathbb{D}$ be a $(\mathbb{N},\mathbb{O})$-bimodular category. Their tensor product, $\mathbb{C} \otimes_{\mathbb{N}} \mathbb{D}$, is a category with the same objects as $\mathbb{C} \times \mathbb{D}$: we write them as $X \otimes_{\mathbb{N}} Y$. The category is presented by the morphisms of $\mathbb{C} \times \mathbb{D}$ and a free family of natural isomorphisms, called the *equilibrators*,

$$\tau_{X,N,Y} \colon (X \triangleleft N) \otimes_N Y \to X \otimes_N (N \triangleright Y), \text{ for each } N \in \mathbb{N}, X \in \mathbb{C}, Y \in \mathbb{D},$$

which are additionally quotiented by the following equations up to the structure isomorphisms of the monoidal actions, $\tau_{X,M \otimes N,Y} = \tau_{X \triangleleft M,N,Y} \mathbin{⨟} \tau_{X,M,N \triangleright Y}$, and $\tau_{X,I,Y} = \mathrm{id}$.

Definition 4.4. Let $\mathbb{C}$ and $\mathbb{C}'$ be two $(\mathbb{M},\mathbb{N})$-bimodular categories and let $\mathbb{D}$ and $\mathbb{D}'$ be a $(\mathbb{N},\mathbb{O})$-bimodular categories. Given two bimodular profunctors, $T \colon \mathbb{C} \to \mathbb{C}'$ and $R \colon \mathbb{D} \to \mathbb{D}'$, their tensor is a bimodular profunctor, $T \otimes_{\mathbb{N}} R \colon \mathbb{C} \otimes_{\mathbb{N}} \mathbb{D} \to \mathbb{C}' \otimes_{\mathbb{N}} \mathbb{D}'$, defined by

$$T \otimes_{\mathbb{N}} R(X \otimes_N Y; X' \otimes_N Y') = T(X;X') \times R(Y,Y')/(\sim),$$

where $(\sim)$ is the equivalence relation generated by $(t_N(x), y) \sim (x, t_N(y))$.

4.3 Pointed Profunctors

Profunctors deal with families of morphisms, and their natural isomorphisms determine correspondences between these families. However, when we use profunctors for the semantics of string diagrams, we most often want to single out a particular morphism between a particular pair of objects. A simple technique to achieve this is to use *pointed profunctors* instead of simply profunctors: this technique was explicitly described by this second author [44] although it has implicit appearances in the literature [3, 26].

Definition 4.5. A pointed profunctor $(P,p)\colon (\mathbb{A},X) \to (\mathbb{B},Y)$ between two pointed categories with a chosen object $X \in \mathbb{A}_{obj}$ and $Y \in \mathbb{B}_{obj}$ is a profunctor $P\colon \mathbb{A} \to \mathbb{B}$ together with an element $p \in P(A,B)$ of the profunctor evaluated on the chosen object of the categories.

From now on, we work using pointed profunctors instead of plain profunctors, see the Appendix Appendix D.1 for a short reference on "pointed coend calculus".

4.4 The Tricategory of Pointed Bimodular Profunctors

We call *collages of string diagrams* to the diagrams of the tricategory of pointed bimodular profunctors.

Definition 4.6. The tricategory of pointed bimodular profunctors, $\mathbb{BmProf}_{\mathtt{pt}}$, has as 0-cells the monoidal categories, $\mathbb{M},\mathbb{N},\mathbb{O},\dots$. The 1-cells between two monoidal categories $\mathbb{M}$ and $\mathbb{N}$ are *pointed bimodular categories*, $(\mathbb{A},\triangleright,\triangleleft,A)$, consisting of a $(\mathbb{M},\mathbb{N})$-bimodular category with two actions $(\mathbb{A},\triangleright,\triangleleft)$ and some object of that category, $A \in \mathbb{A}$. Pointed bimodular categories compose by the tensor of bimodular categories,

$$(\mathbb{A},\triangleright,\triangleleft,A) \otimes_{\mathbb{N}} (\mathbb{B},\triangleright,\triangleleft,B) = (\mathbb{A} \otimes_{\mathbb{N}} \mathbb{B},\triangleright,\triangleleft,A \otimes_{\mathbb{N}} B).$$

The 2-cells between two pointed bimodular categories $(\mathbb{A},\triangleright,\triangleleft,A)$ and $(\mathbb{B},\triangleright,\triangleleft,B)$ are *pointed bimodular profunctors* (P,t,p), consisting of a profunctor $P\colon \mathbb{A} \to \mathbb{B}$ together with a point $p \in P(A,B)$ that are moreover bimodular with compatible natural transformations $t_M\colon P(A;B) \to P(M \triangleright A; M \triangleright B)$, and $t_N\colon P(A;B) \to P(A \triangleleft N; B \triangleleft N)$. These 2-cells compose by profunctor composition and by the tensor of bimodular profunctors.

Finally, the 3-cells between two pointed bimodular profunctors (P,t,p) and (Q,r,q) are bimodular natural transformations that preserve the point, consisting of a natural transformation $\alpha\colon P \to Q$ such that the $\alpha(p) = q$ and moreover $t_M \mathbin{⨟} \alpha = \alpha \mathbin{⨟} r_M$ and $t_N \mathbin{⨟} \alpha = \alpha \mathbin{⨟} r_N$.

Remark 4.7. At the moment of writing, it is unclear to the authors whether a string diagrammatic calculus for tricategories, described by transformations of the string diagrammatic calculus of bicategories, has been fully described and proved sound and complete. However, there seems to be consensus in that this would be the right language for tricategories: much literature assumes it. Let us close this section by tracking explicitly the assumptions we need to employ a diagrammatic syntax for bimodular profunctors.

Conjecture 4.8. *The previous data satisfies all coherence conditions of a tricategory. Moreover, we can reason with tricategories using the calculus of deformations of string diagrams, extending the string diagrams for quasistrict monoidal 2-categories of Bartlett [2].*

4.5 Functor Boxes via Collages of String Diagrams

The following Figure 8 details how to interpret functor boxes as collages of string diagrams. The colored region represents the domain of the lax monoidal functor; the white region represents the codomain. Morphisms of both categories are interpreted as elements of their respective hom-profunctors; and the laxators are used to merge colored regions. The only element that we will explicitly detail is the bimodular category that appears in the closing and opening wires of a functor box.

Proposition 4.9 (Bimodular categories of a lax monoidal functor)**.** *Let $\mathbb{X}$ and $\mathbb{A}$ be two monoidal categories and let $F\colon \mathbb{X} \to \mathbb{A}$ be a monoidal functor between them, endowed with natural transformations $\psi_0\colon J \to FI$ and $\psi_2\colon FX \otimes FY \to F(X \otimes Y)$. The following profunctors, $\mathbb{A} \rtimes_F \mathbb{X}\colon \mathbb{A} \times \mathbb{X} \to \mathbb{A} \times \mathbb{X}$ and*

$\mathbb{X} \ltimes_F \mathbb{A} \colon \mathbb{X} \times \mathbb{A} \to \mathbb{X} \times \mathbb{A}$ determine two promonads, and therefore two Kleisli categories.

$$\mathbb{A} \rtimes_F \mathbb{X}(A,X;B,Y) = \int^{M \in \mathbb{X}} \mathbb{A}(A;B \otimes FM) \times \mathbb{X}(M \otimes X;Y);$$
$$\mathbb{X} \ltimes_F \mathbb{A}(X,A;Y,B) = \int^{M \in \mathbb{X}} \mathbb{A}(A;FM \otimes B) \times \mathbb{X}(M \otimes A;B);$$

These two Kleisli categories are $(\mathbb{A},\mathbb{X})$ and $(\mathbb{X},\mathbb{A})$-bimodular, respectively.

Proof. See Appendix, Proposition D.4. The construction uses the laxity of the monoidal functor. □

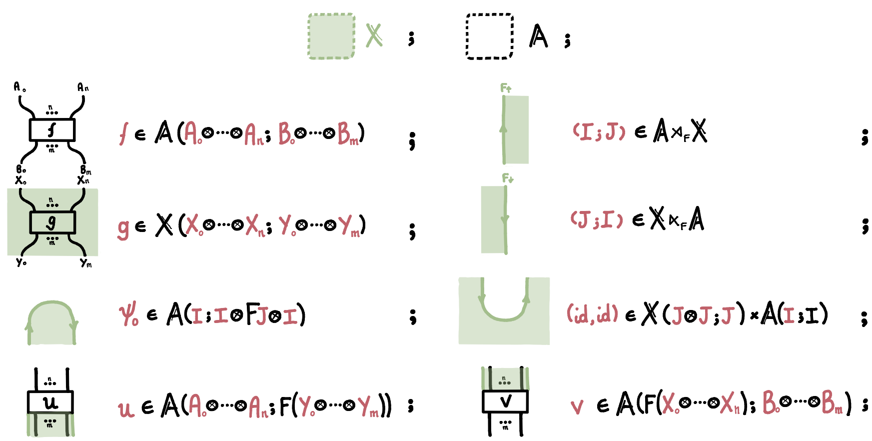


Figure 8: Semantics for functor boxes in terms of pointed bimodular profunctors.

5 String Diagrams of Internal Diagrams

The tubular 3-dimensional cobordisms of internal diagrams are first described as a Frobenius algebra by Bartlett, Douglas, Schommer-Pries and Vicary [3]. We are indebted to this first introduction, which made internal diagrams into a convenient graphical notation in topological quantum field theory [3]. Internal diagrams themselves were later given a expiclit semantics in a monoidal bicategory of pointed profunctors; this was the subject of this second author's contribution to *Applied Category Theory 2020* [43]. An important aspect of the syntax of internal diagrams is their 3-dimensional nature: the syntax not only contains string diagrams, but also reductions between them.

We introduce here a novel syntactic presentation of *internal diagrams* that has the advantage of treating each piece of an internal diagram (including the closing and opening of tubes) as a separate entity in a tricategory. That is, the identity tube or the multiplication and comultiplication tubes are constructed out of smaller pieces in Figure 9. As a consequence, we are later able to introduce for the first time a more refined semantics in terms of a tricategory of *pointed bimodular profunctors*.

Definition 5.1. A *polygraph*, $\mathscr{G}$, is the signature for the string diagrams of a monoidal category. It consists of a set of objects, $\mathscr{G}_{obj}$, and a set of morphisms $\mathscr{G}(A_0,...,A_n;B_0,...,B_m)$ between any two lists of objects, $A_0,...,A_n,B_0,...,B_m \in \mathscr{G}_{obj}$.

Definition 5.2. The *syntactic 3-category of internal diagrams* over a polygraph $\mathscr{G}$ is the 3-category **G** presented by the cells in Figure 9. In other words, it contains two 0-cells, $\mathscr{I}$ and $\mathscr{G}$, in white and blue

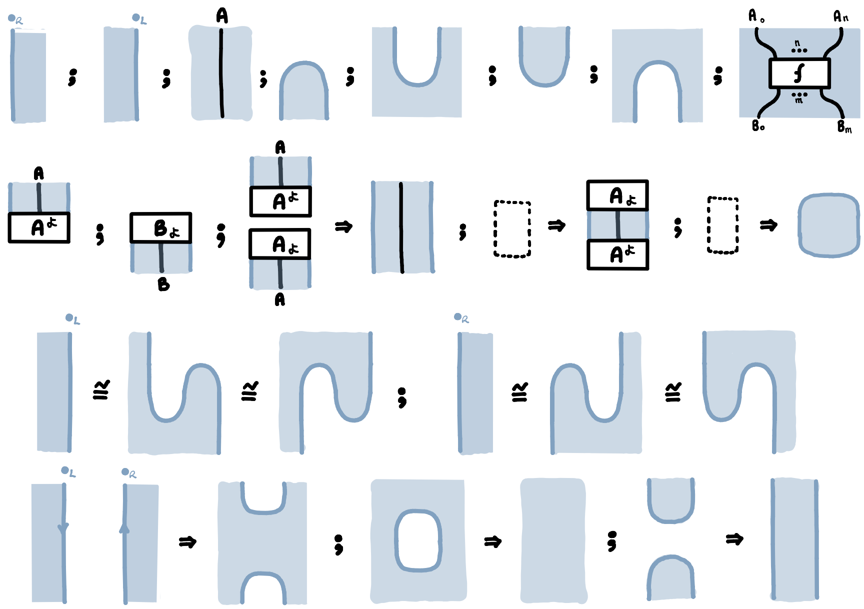


Figure 9: Syntax for open internal diagrams.

in the figure, respectively. It contains a 1-cell $A\colon \mathscr{G} \to \mathscr{G}$ for each object $A \in \mathscr{G}_{obj}$ and two 1-cells, $L_\bullet\colon \mathscr{I} \to \mathscr{G}$ and $R_\bullet\colon \mathscr{G} \to \mathscr{I}$ forming two 2-adjunctions $(L_\bullet) \dashv (R_\bullet)$ and $(R_\bullet) \dashv (L_\bullet)$ up to a 3-cell. It contains the following 2-cells,

- two 2-cells $n_1\colon \mathrm{id} \to L_\bullet \fatsemi R_\bullet$ and $e_1\colon R_\bullet \fatsemi L_\bullet \to \mathrm{id}$ witnessing the 2-adjunction $(L_\bullet) \dashv (R_\bullet)$ and two 2-cells $n_2\colon 1 \to R_\bullet \fatsemi L_\bullet$ and $e_2\colon L_\bullet \fatsemi R_\bullet \to \mathrm{id}$ witnessing the 2-adjunction $(R_\bullet) \dashv (L_\bullet)$ – see Vicary and Heunen [24] for a reference on 2-adjunctions and the swallowtail equations;
- two 2-cells, $A^{\dagger}\colon L_\bullet \fatsemi A \fatsemi R_\bullet \to \mathrm{id}$ and $A_{\dagger}\colon \mathrm{id} \to L_\bullet \fatsemi A \fatsemi R_\bullet$, forming an adjunction $A^{\dagger} \dashv A_{\dagger}$ for each object $A \in \mathscr{G}_{obj}$; and a 2-cell, $f\colon A_0 \fatsemi \dots \fatsemi A_n \to B_0 \fatsemi \dots \fatsemi B_m$, for each edge $f \in \mathscr{G}(A_0,\dots,A_n;B_0,\dots,B_m)$.

Finally, it contains the following 3-cells,

- two invertible 3-cells, $\alpha_1\colon (1 \otimes n_1) \fatsemi (e_1 \otimes 1) \to 1$ and $\beta_1\colon (n_1 \otimes 1) \fatsemi (1 \otimes e_1) \to 1$, witnessing the 2-adjunction $(L_\bullet) \dashv (R_\bullet)$ and satisfying the swallowtail equations; and two invertible 3-cells, $\alpha'_2\colon (1 \otimes n_2) \fatsemi (e_2 \otimes 1) \to 1$ and $\beta_2\colon (n_1 \otimes 1) \fatsemi (1 \otimes e_1) \to 1$, witnessing the 2-adjunction $(R_\bullet) \dashv (L_\bullet)$ and and satisfying the swallowtail equations;
- two 3-cells, $c\colon A^{\dagger} \fatsemi A_{\dagger} \to 1$ and $i\colon 1 \to A_{\dagger} \fatsemi A^{\dagger}$, witnessing the adjunction $A^{\dagger} \dashv A_{\dagger}$ and satisfying the snake equations;
- two 3-cells, $u_i\colon n_1 \fatsemi e_2 \to 1$ and $v_i\colon 1 \to e_2 \fatsemi n_1$ witnessing an adjunction $e_2 \dashv n_1$ and satisfying the snake equations; two 3-cells $u_j\colon 1 \to n_2 \fatsemi e_1$ and $v_i\colon e_1 \fatsemi n_2 \to 1$ witnessing an adjunction $n_2 \dashv e_1$ and satisfying the snake equations.

Theorem 5.3. *For any interpretation of a polygraph into a monoidal category, there exists a 3-functor from the syntactic tricategory of internal diagrams into pointed bimodular profunctors that preserves this interpretation.*

Proof. See Appendix, Theorem E.1. □

Remark 5.4. This syntax can be exemplified by evaluating a quantum comb [12], or a monoidal lens [42] with a morphism, in terms of internal string diagrams [26], see Figure 10. It has been used more generally to reason about coends in monoidal categories [44] and topological quantum field theory [3].

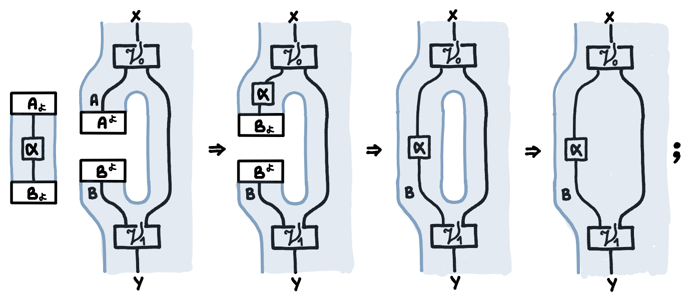


Figure 10: Evaluating a comb in terms of internal string diagrams.

6 Conclusions

Collages of string diagrams provide an abundant graphical calculus. Functor boxes, tensors of bimodular categories and internal diagrams all exist in the graphical calculus of collages. Their technical undepinning is complex: we characterized them as diagrams of pointed bimodular profunctors, but these arrange themselves into a tricategory, which may be difficult to reason about.

Apart from introducing the technique of collages and formalizing multiple extensions to string diagrams, we would like to call the attention to the techniques we use: most of our results on soundness and completeness of diagrams are arranged into adjunctions, which allows us to prove them by reusing the better known results on soundness and completenss for monoidal categories and bicategories.

Related work. An important line of research revolves around *module categories* and *fusion categories*, some specific enriched categories with actions with applications in topological quantum field theories [19, 20, 38]. Specially relevant and recent is Hoek's work, which constructs diagrams for a bimodule category [25, Theorem 3.5.2]. We follow the more elementary notion of bimodular category, called "*biactegory*" in the taxonomy of Capucci and Gavranović [10]. Cockett and Pastro [14] have used instead *linear actions* for concurrency, and even when we take inspiration from their work, their approach is more sophisticated and expressive than our toy example demonstrating bimodular categories (Figure 5).

Most work has been presented for some particular cases of collages: functor boxes have been extensively employed, but never reduced to string diagrams [15, 35]; internal diagrams have served both quantum theory and category theory [3, 26, **?**], and can be given semantics into pointed profunctors [43], but again a presentation as string diagrams was missing. A convenient algebra of lenses [42], a particular type of incomplete diagram, has been recently introduced [21], but this is still independent of the semantics of arbitrary internal diagrams.

Finally, the first author has published a blog post that accompanies this manuscript [9].

Further work. It should be possible to "destrictify" many of the results of this paper. We have only presented a 1-adjunction between strict bimodular categories and bipointed 2-categories; but a higher adjunction would allow us to reuse coherence for bicategories to automatically obtain coherence for bimodular categories. We indicated along the paper the conjectures where further work is warranted.

We conjecture that pointed bimodular profunctors form a compact closed tricategory, with the dual of each monoidal category being the *reverse monoidal category*, $A \otimes_{Rev} B = B \otimes A$. Even when it may be conceptually clear what a compact tricategory should be, it is technically challenging to come up with a concrete definition for it in terms of coherence equations.

Acknowledgements

The authors want to thank David A. Dalrymple for discussion on the string diagrammatic interpretation of functor boxes; and Matteo Capucci for several insightful conversations about notions of 2-dimensional profunctor, that helped us understand how to tie disparate aspects of this story together. The authors thank John Baez and the anonymous reviewers at ACT23 for multiple comments and suggestions that improved this manuscript.

Dylan Braithwaite was supported by an Industrial CASE studentship from the UK Engineering and Physical Sciences Research Council (EPSRC) and the National Physical Laboratory. Mario Román was supported by the European Union through the ESF Estonian IT Academy research measure (2014-2020.4.05.19-0001).

References

[1] Mario Alvarez-Picallo, Dan R. Ghica, David Sprunger & Fabio Zanasi (2021): *Functorial String Diagrams for Reverse-Mode Automatic Differentiation*. arXiv:2107.13433.

[2] Bruce Bartlett (2014): *Quasistrict symmetric monoidal 2-categories via wire diagrams*. arXiv:1409.2148.

[3] Bruce Bartlett, Christopher L. Douglas, Christopher J. Schommer-Pries & Jamie Vicary (2015): *Modular categories as representations of the 3-dimensional bordism 2-category*. arXiv:1509.06811.

[4] Jean Bénabou (1967): *Introduction to bicategories*. In: *Reports of the Midwest Category Seminar*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 1–77.

[5] Guillaume Boisseau & Jeremy Gibbons (2018): *What you needa know about Yoneda: profunctor optics and the Yoneda lemma (functional pearl)*. *Proc. ACM Program. Lang.* 2(ICFP), pp. 84:1–84:27, doi:10.1145/3236779. Available at https://doi.org/10.1145/3236779.

[6] Guillaume Boisseau & Pawel Sobocinski (2021): *String Diagrammatic Electrical Circuit Theory*. In Kohei Kishida, editor: *Proceedings of the Fourth International Conference on Applied Category Theory, ACT 2021, Cambridge, United Kingdom, 12-16th July 2021*, *EPTCS* 372, pp. 178–191, doi:10.4204/EPTCS.372.13. Available at https://doi.org/10.4204/EPTCS.372.13.

[7] Joe Bolt, Jules Hedges & Philipp Zahn (2019): *Bayesian open games*. *CoRR* abs/1910.03656. arXiv:1910.03656.

[8] Filippo Bonchi, Joshua Holland, Robin Piedeleu, Paweł Sobociński & Fabio Zanasi (2019): *Diagrammatic algebra: from linear to concurrent systems*. *Proc. ACM Program. Lang.* 3(POPL), pp. 25:1–25:28, doi:10.1145/3290338. Available at https://doi.org/10.1145/3290338.

[9] Dylan Braithwaite (2023): *Diagrams for Actegories*. Available at https://dylanbraithwaite.github.io/2023/01/31/diagrams-for-actegories.html.

[10] Matteo Capucci & Bruno Gavranović (2022): *Actegories for the working amthematician*. *arXiv preprint arXiv:2203.16351*.

[11] Dimitri Chikhladze (2015): *Lax formal theory of monads, monoidal approach to bicategorical structures and generalized operads*. arXiv:1412.4628.

[12] Giulio Chiribella, Giacomo Mauro D'Ariano & Paolo Perinotti (2009): *Theoretical framework for quantum networks*. *Phys. Rev. A* 80, p. 022339, doi:10.1103/PhysRevA.80.022339.

[13] Bryce Clarke, Derek Elkins, Jeremy Gibbons, Fosco Loregian, Bartosz Milewski, Emily Pillmore & Mario Román (2022): *Profunctor Optics, a Categorical Update*. arXiv:2001.07488.

[14] Robin B. Cockett & Craig A. Pastro (2009): *The logic of message-passing*. *Science of Computer Programming* 74(8), pp. 498–533, doi:10.1016/j.scico.2007.11.005.

[15] Robin B. Cockett & Robert A. G. Seely (1999): *Linearly distributive functors*. Journal of Pure and Applied Algebra 143(1-3), pp. 155–203.

[16] Bob Coecke, Tobias Fritz & Robert W. Spekkens (2016): *A mathematical theory of resources*. Information and Computation 250, pp. 59–86, doi:10.1016/j.ic.2016.02.008.

[17] Elena Di Lavore, Giovanni de Felice & Mario Román (2022): *Monoidal Streams for Dataflow Programming*. In: Proceedings of the 37th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS '22, Association for Computing Machinery, New York, NY, USA, p. 14, doi:10.1145/3531130.3533365.

[18] Edsger W. Dijkstra (1962): *Over de sequentialiteit van procesbeschrijvingen*. Unpublished. Transcribed by Gerrit Jan Veltink for the E.W. Dijkstra Archive, Center for American History. Available at `https://www.cs.utexas.edu/users/EWD/ewd00xx/EWD35.PDF`.

[19] Christopher L. Douglas, Christopher Schommer-Pries & Noah Snyder (2019): *The balanced tensor product of module categories*. Kyoto Journal of Mathematics 59(1), doi:10.1215/21562261-2018-0006.

[20] Vladimir Drinfeld, Shlomo Gelaki, Dmitri Nikshych & Victor Ostrik (2010): *On braided fusion categories I*. Selecta Mathematica 16(1), pp. 1–119, doi:10.1007/s00029-010-0017-z.

[21] Matt Earnshaw, James Hefford & Mario Román (2023): *The Produoidal Algebra of Process Decomposition*. arXiv:2301.11867.

[22] Neil Ghani, Jules Hedges, Viktor Winschel & Philipp Zahn (2018): *Compositional Game Theory*. In: Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS '18, Association for Computing Machinery, New York, NY, USA, p. 472–481, doi:10.1145/3209108.3209165.

[23] James Hefford & Cole Comfort (2022): *Coend Optics for Quantum Combs*. arXiv:2205.09027.

[24] Chris Heunen & Jamie Vicary (2019): *Categories for Quantum Theory: an introduction*. Oxford University Press.

[25] Keeley Hoek (2019): *Drinfeld centers for bimodule categories*. Ph.D. thesis, MSc. thesis, The Australian National University.

[26] Nick Hu & Jamie Vicary (2021): *Traced Monoidal Categories as Algebraic Structures in Prof*. In Ana Sokolova, editor: Proceedings 37th Conference on Mathematical Foundations of Programming Semantics, MFPS 2021, Hybrid: Salzburg, Austria and Online, 30th August - 2nd September, 2021, EPTCS 351, pp. 84–97, doi:10.4204/EPTCS.351.6.

[27] David A Huffman (1954): *The Synthesis of Sequential Switching Circuits*. Journal of the Franklin Institute 257(3), pp. 161–190.

[28] Niles Johnson & Donald Yau (2020): *2-Dimensional Categories*. arXiv:2002.06055.

[29] André Joyal & Ross Street (1991): *The geometry of tensor calculus, I*. Advances in Mathematics 88(1), pp. 55–112, doi:10.1016/0001-8708(91)90003-P.

[30] Paul Blain Levy (2022): *Call-by-Push-Value*. ACM SIGLOG News 9(2), p. 7–29, doi:10.1145/3537668.3537670.

[31] Fosco Loregian (2021): *(Co)end Calculus*. London Mathematical Society Lecture Note Series, Cambridge University Press, doi:10.1017/9781108778657.

[32] Saunders Mac Lane (1971): *Categories for the Working Mathematician*. Graduate Texts in Mathematics 5, Springer Verlag, doi:10.1007/978-1-4757-4721-8.

[33] Daniel Marsden (2014): *Category theory using string diagrams*. arXiv preprint arXiv:1401.7220.

[34] Paddy McCrudden (2000): *Categories of representations of coalgebroids*. Advances in Mathematics 154(2), pp. 299–332.

[35] Paul-André Melliès (2006): *Functorial Boxes in String Diagrams*. In Zoltán Ésik, editor: Computer Science Logic, 20th International Workshop, CSL 2006, 15th Annual Conference of the EACSL, Szeged, Hungary, September 25-29, 2006, Proceedings, Lecture Notes in Computer Science 4207, Springer, pp. 1–30, doi:10.1007/11874683_1.

[36] Eugenio Moggi (1991): *Notions of Computation and Monads*. *Inf. Comput.* 93(1), pp. 55–92, doi:10.1016/0890-5401(91)90052-4.

[37] Chad Nester (2020): *A Foundation for Ledger Structures*. In Emmanuelle Anceaume, Christophe Bisière, Matthieu Bouvard, Quentin Bramas & Catherine Casamatta, editors: *2nd International Conference on Blockchain Economics, Security and Protocols, Tokenomics 2020, October 26-27, 2020, Toulouse, France*, *OASIcs* 82, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, pp. 7:1–7:13, doi:10.4230/OASIcs.Tokenomics.2020.7. Available at https://doi.org/10.4230/OASIcs.Tokenomics.2020.7.

[38] Victor Ostrik (2003): *Module categories, weak Hopf algebras and modular invariants*. *Transformation groups* 8, pp. 177–206.

[39] Craig Pastro & Ross Street (2007): *Doubles for monoidal categories*. *arXiv preprint arXiv:0711.1859*.

[40] Evan Patterson, David I. Spivak & Dmitry Vagner (2021): *Wiring diagrams as normal forms for computing in symmetric monoidal categories*. *Electronic Proceedings in Theoretical Computer Science* 333, pp. 49–64, doi:10.4204/eptcs.333.4.

[41] Maciej Piróg & Nicolas Wu (2016): *String diagrams for free monads (functional pearl)*. In Jacques Garrigue, Gabriele Keller & Eijiro Sumii, editors: *Proceedings of the 21st ACM SIGPLAN International Conference on Functional Programming, ICFP 2016, Nara, Japan, September 18-22, 2016*, ACM, pp. 490–501, doi:10.1145/2951913.2951947.

[42] Mitchell Riley (2018): *Categories of optics*. *arXiv preprint arXiv:1809.00738*.

[43] Mario Román (2020): *Comb Diagrams for Discrete-Time Feedback*. *CoRR* abs/2003.06214. arXiv:2003.06214.

[44] Mario Román (2020): *Open Diagrams via Coend Calculus*. In David I. Spivak & Jamie Vicary, editors: *Proceedings of the 3rd Annual International Applied Category Theory Conference 2020, ACT 2020, Cambridge, USA, 6-10th July 2020*, *EPTCS* 333, pp. 65–78, doi:10.4204/EPTCS.333.5.

[45] Mario Román (2022): *Promonads and String Diagrams for Effectful Categories*. *CoRR* abs/2205.07664, doi:10.48550/arXiv.2205.07664. arXiv:2205.07664.

[46] Michael Shulman (2016): *Categorical logic from a categorical point of view*. *Available on the web*. Available at https://mikeshulman.github.io/catlog/catlog.pdf.

[47] Abraham Silberschatz, Peter Baer Galvin & Greg Gagne (2018): *Operating System Concepts, 10th Edition*. Wiley. Available at http://os-book.com/OS10/index.html.

[48] David I. Spivak (2022): *Generalized Lens Categories via functors* $\mathscr{C}^{\mathrm{op}} \to \mathsf{Cat}$. arXiv:1908.02202.

[49] Sam Staton & Paul Blain Levy (2013): *Universal properties of impure programming languages*. In Roberto Giacobazzi & Radhia Cousot, editors: *The 40th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL '13, Rome, Italy - January 23 - 25, 2013*, ACM, pp. 179–192, doi:10.1145/2429069.2429091.

[50] Ross Street (1981): *Cauchy characterization of enriched categories*. *Rendiconti del Seminario Matematico e Fisico di Milano* 51(1), pp. 217–233.

[51] Zoran Škoda (2009): *Some equivariant constructions in noncommutative algebraic geometry*. *Georgian Mathematical Journal* 16(1), pp. 183–202. arXiv:0811.4770.

A Preliminaries

Proposition A.1 (Reducing an adjunction)**.** *Let $F\colon \mathbb{A} \to \mathbb{C}$ and $H ⨾ U\colon \mathbb{C} \to \mathbb{A}$ determine an adjunction $(F, H ⨾ U, \eta, \varepsilon)$ and let $P\colon \mathbb{B} \to \mathbb{C}$ determine a second adjunction (P, H, u, c) such that the unit $u\colon I \to P ⨾ H$ is a natural isomorphism (as in Figure 11). Then, $F ⨾ H$ is left adjoint to U.*

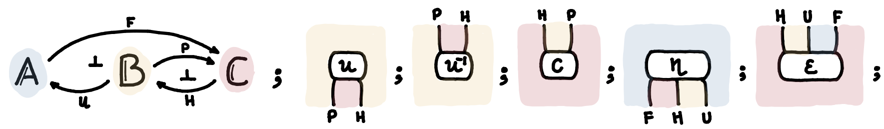


Figure 11: Setting for reducing an adjunction.

Proof. We employ the string diagrammatic calculus of bicategories to the bicategory of categories, functors and natural transformations [33]. We define the morphisms in Figure 12 to be the unit and the counit of the adjunction. We then prove that they satisfy the snake equations in Figures 12 and 13.

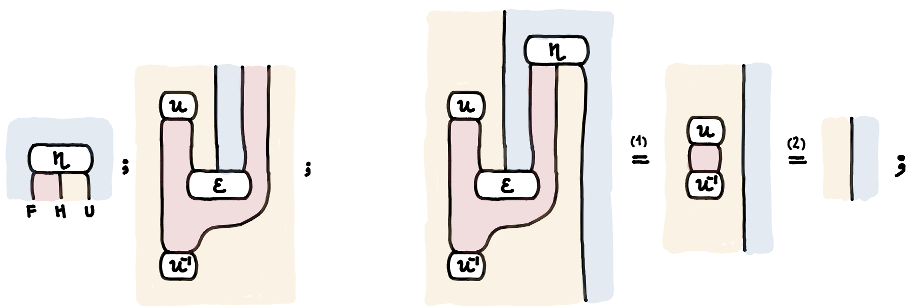


Figure 12: Unit and counit of the reduced adjunction (left). First snake equation (right).

In the first snake equation, in Figure 12, we use *(i)* that there is a duality (η, ε), and *(ii)* that u is invertible. In the second snake equation, in Figure 13, we use *(i)* that there is a duality (u, c), *(ii)* that u is invertible, *(iii)* that there is a duality (u, c), again; and *(iv)* that there is a duality (η, ε).

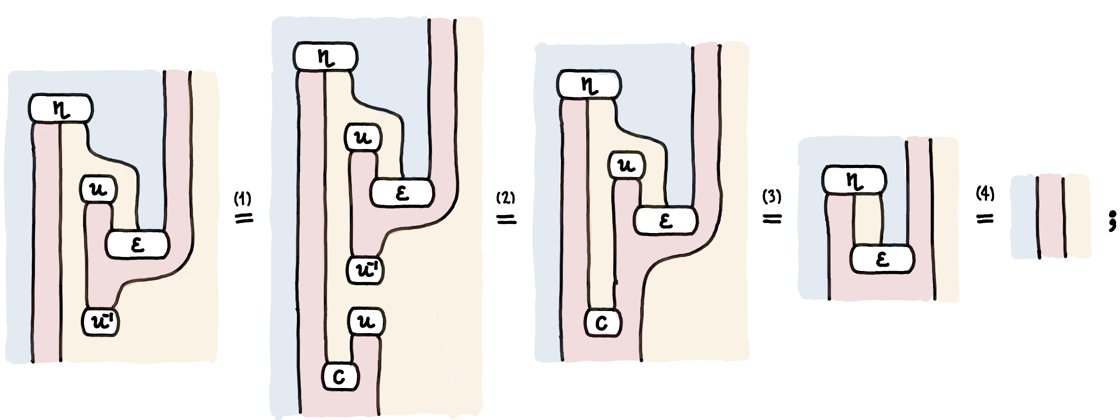


Figure 13: Second snake equation.

□

Proposition A.2. *Let $L\colon \mathbb{C} \to \mathbb{D}$ and $R\colon \mathbb{D} \to \mathbb{C}$ determine an adjunction (L,R,η,ε). For each object $A \in \mathbb{C}_{obj}$, this induces an adjunction between the coslice categories $\mathbb{C}\setminus A$ and $\mathbb{D}\setminus LA$.*

Proof. The functor $L_A\colon \mathbb{C}\setminus A \to \mathbb{D}\setminus LA$ is just an application of the functor L. The functor $R_A\colon \mathbb{D}\setminus LA \to \mathbb{C}\setminus A$ is defined using the unit of the adjunction as

$$R_A\left(LA \xrightarrow{f} B\right) = \left(A \xrightarrow{\eta} RLA \xrightarrow{Rf} B\right).$$

Note that a morphism $\alpha\colon LB \to C$ makes the first diagram in commute if and only if its adjunct, $\alpha^*\colon B \to RC$, makes the second diagram in commute.

$$\begin{array}{ccc} LB & \xrightarrow{\alpha} & C \\ {\scriptstyle Lf}\uparrow & \nearrow_{g} & \\ A & & \end{array} \qquad\qquad \begin{array}{ccc} B & \xrightarrow{\alpha^*} & RC \\ {\scriptstyle f}\uparrow & & \uparrow{\scriptstyle Rg} \\ A & \xrightarrow{\eta} & RLA \end{array}$$

This is because two morphisms are equal if and only if their adjuncts are. We have that $(Lf \mathbin{⨾} \alpha)^* = f \mathbin{⨾} \alpha^*$ and $g^* = \eta \mathbin{⨾} Rg$. This induces the hom-set isomorphism of the adjunction, where each morphism is again adjunct to the same adjunct it was before. □

A.1 Bicategories

In the same way that a polygraph represents the signature for a monoidal category, a *2-graph* is the signature that allows us to freely generate a 2-category.

Definition A.3. A 2-graph $\mathscr{G}$ is given by a set of vertices, $\mathscr{G}_{obj}$; a set of edges between any two vertices, $\mathscr{G}(X;Y)$ for $X,Y \in \mathscr{G}_{obj}$; and a set of 2-edges for each pair of paths of vertices with the same source and target. That is, there is a set of 2-edges $\mathscr{G}(X;Y)(A_0,\dots,A_n;B_0,\dots,B_m)$, for each path $A_0 \in \mathscr{G}(X;U_0),\dots,A_n \in \mathscr{G}(U_{n-1};Y)$ and each path $B_0 \in \mathscr{G}(X;V_0),\dots,B_m \in \mathscr{G}(V_{m-1};Y)$.

A homomorhpism of 2-graphs is a family of functions on vertices, edges, and 2-edges that preserve their sources and targets; these form a category **2Graph** endowed with a forgetful functor $\mathsf{U}\colon$ **2Cat** $\to$ **2Graph** from the category of 2-categories and 2-functors.

Correctness of string diagrams for 2-categories [4, 33] amounts the fact that free 2-category over a 2-graph is given by string diagrams. It is difficult to find a proof for this exact result in the literature, but the widespread use of string diagrams for bicategories suggests that it is commonly accepted.

Theorem A.4 (String diagrams for bicategories)**.** *There exists an adjunction between 2-graphs and 2-categories given by progressive string diagrams* $\mathsf{Str}\colon$ **2Graph** $\to$ **2Cat** *and the previously mentioned forgetful functor* $\mathsf{U}\colon$ **2Cat** $\to$ **2Graph**.

A.2 Profunctors

Definition A.5. A *profunctor* $(P,<,>)$ between two categories $\mathbb{A}$ and $\mathbb{B}$ is a family of sets $P(A,B)$ indexed by objects $\mathbb{A}$ and $\mathbb{B}$, and endowed with jointly functorial left and right actions of the morphisms of $\mathbb{A}$ and $\mathbb{B}$, respectively. Explicitly, types of these actions are $(>)\colon \hom(A',A) \times P(A',B) \to P(A,B)$, and $(<)\colon \hom(B,B') \times P(A,B) \to P(A,B')$. These must satisfy

- compatibility, $(f > p) < g = f > (p < g)$,
- preserve identities, $id > p = p$, and $p < id = p$,
- and preserve composition, $(p < f) < g = p < (f \mathbin{⨾} g)$ and $f > (g > p) = (f \mathbin{⨾} g) > p$.

More succintly, a profunctor $P\colon \mathbb{A} \to \mathbb{B}$ is a functor $P\colon \mathbb{A}^{op} \times \mathbb{B} \to \mathbf{Set}$. When presented as a family of sets with a pair of actions, profunctors have been sometimes called *bimodules*. We avoid this term, which we reserve for the classical algebraic notion, and its categorification.

B Bimodular categories

We provide here an alternative spelled-out definition of bimodular profunctors.

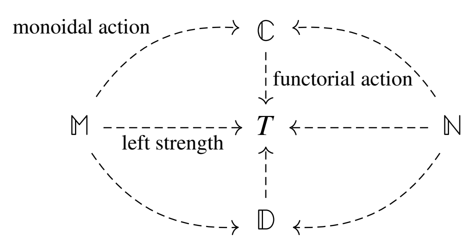
Definition B.1. A *bimodular profunctor* from $\mathbb{C}$ to $\mathbb{D}$, a pair of $(\mathbb{M}, \mathbb{N}, \triangleright, \triangleleft)$-bimodular categories, is a profunctor $(P, <, >)\colon \mathbb{C} \to \mathbb{D}$ endowed with a pair of natural whiskering operators,

$$(\succ)\colon \mathbb{M}(X;X') \times P(A;B) \to P(X \triangleright A; X' \triangleright B), \quad \text{and} \quad (\prec)\colon P(A;B) \times \mathbb{N}(Y;Y') \to P(A \triangleleft Y; B \triangleleft Y'),$$

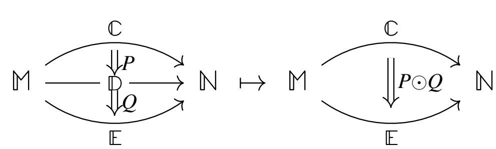
satisfying the following axioms up to coherence of the monoidal actions.

$$\begin{aligned}
&(m \succ f) < (m' \triangleright g) = (m \mathbin{⨾} m') \succ (f < g), && (f \prec n) < (g \triangleleft n') = (f < g) \prec (n \mathbin{⨾} n'),\\
&(m' \triangleright g) > (m \succ f) = (m' \mathbin{⨾} m) \succ (g > f), && (g \triangleleft n') > (f \prec n) = (g > f) \prec (n' \mathbin{⨾} n),\\
&\mathrm{id} \succ f = f = f \prec \mathrm{id}, && u \succ (f \prec v) = (u \succ f) \prec v,\\
&u \succ (v \succ f) = (u \otimes v) \succ f, && (f \prec u) \prec v = f \prec (u \otimes v).
\end{aligned}$$

Remark B.2. Not only is this definition a natural extension of profunctors as category-bimodules to bimodular categories, but the strengths of bimodular profunctors can be seen as an additional bimodule structure in the form of an action of the monoidal categories on the profunctor.

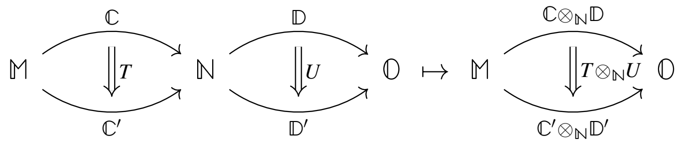


The vertical composition of bimodular profunctors is given by ordinary profunctor composition while the strength is obtained as the cartesian product of the two constituent strengths.



As with ordinary profunctors, this composition is not strictly associative or unital but indeed there is additionally a natural notion of morphism between bimodular profunctors with which structure maps can be defined. Namely these are natural transformations which commute with the strengths. This makes bimodular categories and the profunctors between them into a bicategory.

As noted in the main text, the strength of a bimodular profunctor could alternatively be considered as a horizontal action of a monoidal category on the profunctor. In fact such strengths are equivalent to the data of monoid actions for the lax monoidal functors $\mathbb{M}(-,-)$ and $\mathbb{N}(-,-)$, viewed as monoid objects in their respective functor categories. This gives a route for defining the horizontal composite of bimodular profunctors, just as we do for bimodular categories, by quotienting out the common action.



As in any category of modules we would like to define the composite by quotienting out the action of the common scalar, taking a coequaliser of the two action maps.

$$\begin{array}{c} P(C,C') \times Q(D,D') \\ \text{left strength} \downarrow\downarrow \text{right strength} \\ \int^M P(C < M, C' < M) \times Q(D,D') + P(C,C') \times Q(M > D, M > D') \\ \downarrow \\ P \otimes Q(C,C',D,D') \end{array}$$

The form of this diagram is slightly more complicated than the one we saw for bimodular categories. In the form we have seen them, the strengths of a Tambara module have different codomains for the two action maps. So to match the form of a coequaliser we embed both into their coproduct. This is alternatively the pushout of the two action maps, but we choose to present it as is to reinforce that this is just another instance of bimodule composition.

B.1 String Diagrams for Bimodular Categories

Proposition B.3 (From Proposition 2.4). *There exists a forgetful functor U :* **SBimod** $\rightarrow$ **Bgraph**.

Proof. Given $(\mathbb{A}, \mathbb{M}, \triangleright, \mathbb{N}, \triangleleft)$, we define the following bimodular graph.

$$\begin{aligned} &\mathscr{A}(M_0,...,M_m,A,N_0,...,N_n;P_0,...,P_p,B,Q_0,...,Q_q) = \\ &\qquad \mathbb{A}(M_0 \triangleright ... \triangleright M_m \triangleright A_n \triangleleft N_0 \triangleleft ... \triangleleft N_n; N_0 \triangleright ... \triangleright N_n \triangleright B \triangleleft Q_0 \triangleleft ... \triangleleft Q_q), \\ &\mathscr{M}(M_0,...,M_m;P_0,...,P_p) = \mathbb{M}(M_0 \otimes ... \otimes M_m; P_0 \otimes ... \otimes P_p), \\ &\mathscr{N}(N_0,...,N_n;Q_0,...,Q_q) = \mathbb{N}(N_0 \otimes ... \otimes N_n; Q_0 \otimes ... \otimes Q_q). \end{aligned}$$

We now check that this assignment is functorial: indeed, the functors (F,H,K) induce three functions between the edge sets of the bimodular graph. □

Lemma B.4. *Collages induce a functor* Coll: **sBimod** $\rightarrow$ **2Cat**$_2$ *from strict bimodular categories to bipointed 2-categories. Picking the hom-category between the chosen 0-cells of a bipointed 2-category induces a functor from bipointed 2-categories to strict bimodular categories* Chosen: **2Cat**$_2$ $\rightarrow$ **sBimod**.

Proof. See Appendix, Lemmas B.5 and B.6. □

Lemma B.5 (From Lemma B.4)**.** *Collages induce a functor* $\mathsf{Coll}\colon \mathbf{sBimod} \to \mathbf{2Cat}_2$ *from strict bimodular categories to bipointed 2-categories.*

Proof. On objects, the functor is defined by the collage with its two 0-cells, $\mathsf{Coll}(\mathbb{C},\mathbb{M},\mathbb{N}) = (\mathsf{Coll}_{\mathbb{C}}, M, N)$. Given a morphism of bimodular categories, $(F,H,K)\colon (\mathbb{C},\mathbb{M},\mathbb{N}) \to (\mathbb{D},\mathbb{P},\mathbb{Q})$, the functor takes it to the strict 2-functor defined by: sending $\mathsf{Coll}_{\mathbb{C}}(M,M) = \mathbb{M}$ to $\mathsf{Coll}_{\mathbb{D}}(P,P) = \mathbb{P}$ with H; sending $\mathsf{Coll}_{\mathbb{C}}(N,N) = \mathbb{N}$ to $\mathsf{Coll}_{\mathbb{D}}(Q,Q) = \mathbb{Q}$ with K; sending $\mathsf{Coll}_{\mathbb{C}}(M,N) = \mathbb{C}$ to $\mathsf{Coll}_{\mathbb{D}}(P,Q) = \mathbb{D}$ with F; and finally noticing that both $\mathsf{Coll}_{\mathbb{C}}(N,M)$ and $\mathsf{Coll}_{\mathbb{D}}(Q,P)$ are empty. This assignment defines a 2-functor preserving composition: this is thanks to the fact that composition has been defined in $\mathsf{Coll}_{\mathbb{C}}$ and $\mathsf{Coll}_{\mathbb{D}}$ using the strict monoidal actions, and the functors F, H and K do preserve the monoidal actions. □

Lemma B.6 (From Lemma B.4)**.** *Picking the hom-category between the chosen 0-cells of a bipointed 2-category induces a functor from bipointed 2-categories to strict bimodular categories* $\mathsf{Chosen}\colon \mathbf{2Cat}_2 \to \mathbf{sBimod}$.

Proof. On objects, the functor is defined by taking a bipointed bicategory the hom-category between the selected points, $\mathsf{Chosen}(\mathbb{A},M,N) = \mathbb{A}(M,N)$. This hom-category is a bimodular category acted on by the hom-categories $\mathbb{A}(M,M)$ and $\mathbb{A}(N,N)$; both of these are monoidal categories (bicategories with a single object) with the tensor defined by composition in the bicategory. The actions are also defined by pre and post-composition in the bicategory.

Consider now a strict 2-functor $S\colon (\mathbb{A},M,N) \to (\mathbb{B},P,Q)$ that sends $S(M) = P$ and $S(N) = Q$. It must induce strict monoidal functors $H = S(M,M)\colon \mathbb{A}(M,M) \to \mathbb{B}(P,P)$ and $K = S(N,N)\colon \mathbb{A}(N,N) \to \mathbb{B}(Q,Q)$ and a functor $F = S(M,N)\colon \mathbb{A}(M,N) \to \mathbb{B}(P,Q)$. All these functors must preserve composition in the original bicategory, so the triple $\mathsf{Chosen}(S) = (F,H,K)$ is a morphism of strict bimodular categories. □

Theorem B.7 (From Theorem 2.7)**.** *There exists an adjunction between strict bimodular categories and bipointed 2-categories given by the collage,* $\mathsf{Coll}_{\mathbb{C}}\colon \mathbf{sBimod} \to \mathbf{2Cat}_2$*, and picking the hom-category between the chosen 0-cells,* $\mathsf{Chosen}\colon \mathbf{2Cat}_2 \to \mathbf{sBimod}$*. Moreover, the unit of this adjunction is a natural isomorphism.*

Proof. We have already proven that both sides of the adjunction are indeed functors in Lemma B.4. Let us show that $\mathsf{Coll}_{\mathbb{C}}$ is the free bipointed 2-category on a bimodular category $\mathbb{C}$. We start by noting that there exists a homomorphism of bimodular categories

$$\mathscr{I}\colon (\mathbb{C},\mathbb{M},\mathbb{N}) \to (\mathsf{Coll}_{\mathbb{C}}(M,N), \mathsf{Coll}_{\mathbb{C}}(M,M), \mathsf{Coll}_{\mathbb{C}}(N,N)),$$

by construction of the collage; this determines the natural isomorphism of the unit of the adjunction we are constructing. Consider now a bipointed 2-category $(\mathbb{A},P,Q)$ and a homomorphism of bimodular categories

$$(F,H,K)\colon (\mathbb{C},\mathbb{M},\mathbb{N}) \to (\mathbb{A}(P,Q), \mathbb{A}(P,P), \mathbb{A}(Q,Q));$$

we will now prove that there exists a unique 2-functor $\mathscr{F}\colon \mathsf{Coll}_{\mathbb{C}} \to \mathbb{A}$ such that $\mathscr{I} ⨾ \mathscr{F} = (F,H,K)$. Because the 2-functor is bipointed, we know that $\mathscr{F}(M) = P$ and that $\mathscr{F}(N) = Q$, so it is determined on 0-cells. We know that its component on $\mathsf{Coll}_{\mathbb{C}}(M,N)$, $\mathsf{Coll}_{\mathbb{C}}(M,M)$ and $\mathsf{Coll}_{\mathbb{C}}(N,N)$ must be given by F, H and K; while its only possible component on the empty category $\mathsf{Coll}_{\mathbb{C}}(N,M)$ is trivial; this determines it on 1-cells, but also on 2-cells, because F, H, and K are a functor and a pair of monoidal functors. □

Theorem B.8 (From Theorem 2.8)**.** *There exists an adjunction between bimodular graphs and strict bimodular categories. The left side of this adjunction is given by finding the bimodular category whose collage is the free 2-category on the bimodular graph,* $\mathsf{bmStr}\colon \mathbf{bmGraph} \to \mathbf{sBimod}$*. The right side of the adjunction is the previously mentioned forgetful functor* $\mathsf{U}\colon \mathbf{sBimod} \to \mathbf{bmGraph}$*.*

Proof. String diagrams for bicategories are based on an adjunction between 2-graphs and 2-categories in Theorem A.4, whose left adjoint is $\mathbf{Str}\colon \mathbf{2Graph} \to \mathbf{2Cat}$. By Proposition A.2, this induces an adjunction between bipointed 2-graphs and bipointed 2-categories, $\mathbf{Str}_2\colon \mathbf{2Graph}_2 \to \mathbf{2Cat}_2$, both seen as slice categories of the discrete graph on two vertices and discrete category on two objects, respectively. We can compose this adjunction with the adjunction between bimodular graphs and bipointed 2-graphs in Proposition B.9 to obtain a left adjoint $\mathsf{bStr}_2\colon \mathbf{bmGraph} \to \mathbf{2Cat}_2$.

Finally, we employ the adjunction given by collages from strict bimodular categories to bipointed 2-categories in Theorem 2.7. This adjunction has an invertible unit, and thus, by a general principle (Proposition A.1), it induces an adjunction with left adjoint $\mathsf{bmStr}\colon \mathbf{bmGraph} \to \mathbf{sBimod}$, see Figure 14.

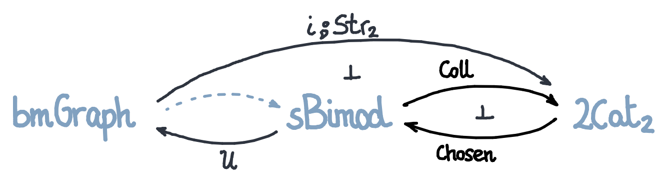


Figure 14: Induced adjunction between bimodular graphs and strict bimodular categories.

Here, we have used that the unit of the adjunction, $\mathbb{C} \to \mathsf{Chosen}(\mathsf{Coll}(\mathbb{C}))$, determines an isomorphism for any bimodular category $\mathbb{C}$. Note however that the same does not hold for the counit of the adjunction, $\mathsf{Coll}(\mathsf{Chosen}(\mathbb{B})) \to \mathbb{B}$, which fails to be an isomorphism and is merely an inclusion. □

Proposition B.9. *There exists an adjunction between bimodular graphs and bipointed 2-graphs, formed by functors* $i\colon \mathbf{bmGraph} \to \mathbf{2Graph}_2$ *and* $u\colon \mathbf{2Graph}_2 \to \mathbf{bmGraph}$*.*

Proof. The left adjoint $i\colon \mathbf{bmGraph} \to \mathbf{2Graph}_2$, given a bimodular graph $(\mathscr{A},\mathscr{M},\mathscr{N})$, constructs a bipointed 2-graph with precisely the two selected vertices, M and N; edges given by

$$\mathscr{G}(M,M) = \mathscr{M}_{obj}, \quad \mathscr{G}(N,N) = \mathscr{N}_{obj}, \quad \mathscr{G}(N,M) = 0, \quad \text{and} \quad \mathscr{G}(M,N) = \mathscr{A}_{obj};$$

and 2-edges given (and correctly typed) by the left, right, and central edges.

The right adjoint $u\colon \mathbf{2Graph}_2 \to \mathbf{bmGraph}$ takes a bipointed 2-graph $(\mathscr{G},X,Y)$ and constructs a bimodular graph, $(\mathscr{A},\mathscr{M},\mathscr{N})$, where

$$\mathscr{M}_{obj} = \mathscr{G}(X,X), \quad \mathscr{N}_{obj} = \mathscr{G}(Y,Y), \quad \text{and} \quad \mathscr{A}_{obj} = \mathscr{G}(X,Y),$$

forgetting about the rest of the vertices. The edges of the bimodular graph are given (and conveniently typed) by the 2-edges of the 2-graph. Note that this can be seen as the forgetful functor that removes all of the vertices of the 2-graph except for the selected ones and the arrows from one to the other, its free conterpart can be seen as an inclusion of a 2-graph that happens to have exactly two vertices and only edges going from one to the other or staying on the same vertex. □

Remark B.10 (Shared state). The equations in Figure 15 present the theory of shared state over the string diagrams of bimodular categories. Semantics can be given in two different theories of processes sharing the same state. For instance, the first category can allow for probabilistic processes, $\mathbb{A} = \mathbf{Stoch}$, while the second can be deterministic, $\mathbb{B} = \mathbf{Set}$. The bimodular category determined by the promonad $\mathbf{StochState}_S(A,B) = \mathbf{Set}(S \times A, \mathsf{Stoch}(S \times B))$ can give semantics to both and has suitable monoidal actions: the actions to the common wire become the get and put functions of the state promonad.

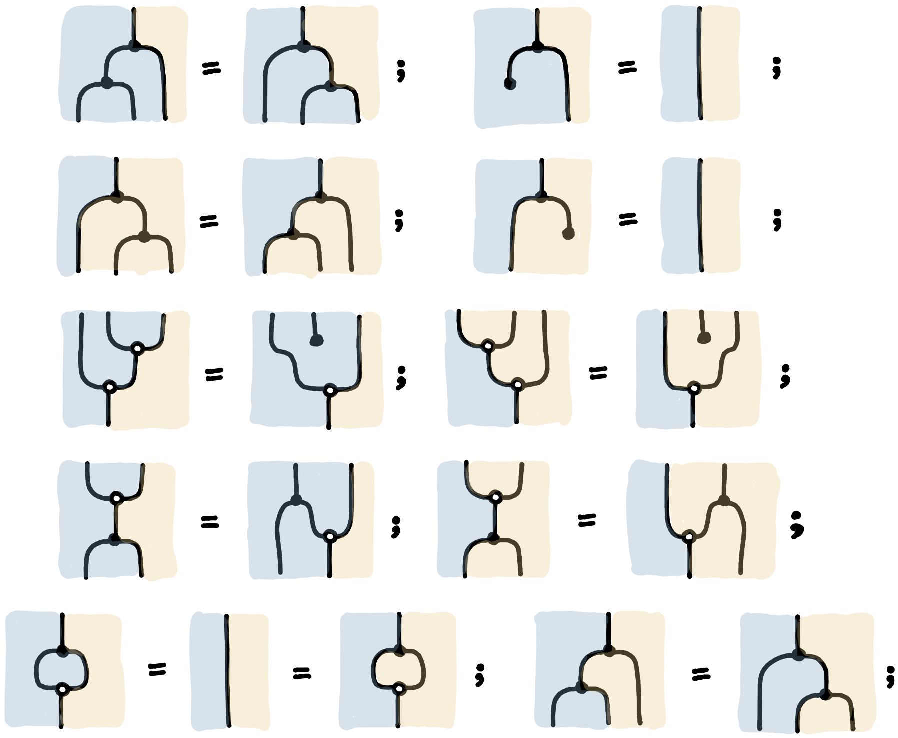


Figure 15: Theory of shared state.

B.2 String Diagrams of Premonoidal Categories

Definition B.11. A *premonoidal category* is a category $\mathbb{C}$ endowed with an object $I \in \mathbb{C}$ and an object $A \otimes B \in \mathbb{C}$ for each $A, B \in \mathbb{C}_{obj}$; and two functors $(A \otimes \bullet) \colon \mathbb{C} \to \mathbb{C}$ and $(\bullet \otimes B) \colon \mathbb{C} \to \mathbb{C}$ that coincide on $(A \otimes B)$, even if $(\bullet \otimes \bullet)$ is not itself a functor. Finally, it is endowed with the following cohernece isomorphisms, $\alpha_{A,B,C} \colon A \otimes (B \otimes C) \to (A \otimes B) \otimes C$, $\lambda_A \colon A \otimes I \to A$ and $\rho_A \colon I \otimes A \to A$, which interchange with any other morphism, are natural at each given component and satisfy the pentagon and triangle equations.

Definition B.12. An *effectful category* is an identity-on-objects functor, $\mathbb{V} \to \mathbb{C}$, from a monoidal category $\mathbb{V}$ to a premonoidal category $\mathbb{C}$ that strictly preserves all of the premonoidal structure and whose image is central.

Proposition B.13. *Effectful categories $\mathbb{V} \to \mathbb{C}$ are equivalent to $(\mathbb{V}, \mathbb{V})$-bimodular categories such that there exists an identity on objects functor that preserves the monoidal actions [30].*

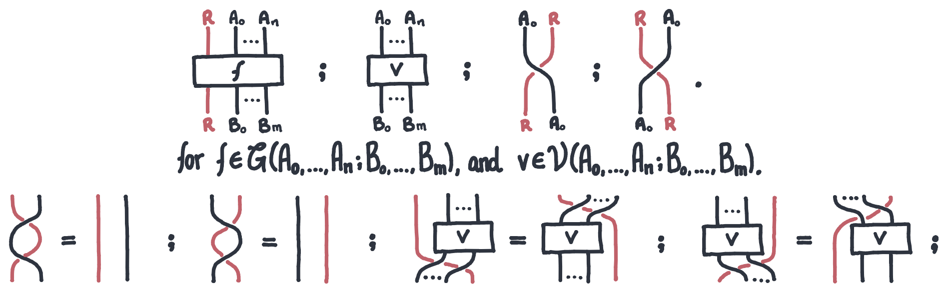


Figure 16: Syntax for the string diagrams of premonoidal and effectful categories (Román, 2020 [45]).

In this sense, the string diagrams of premonoidal categories and effectful categories are particular cases of the string diagrams for bimodular categories. The extra wire that appears in the string diagrams of an effectful category $\mathbb{V} \to \mathbb{C}$ is precisely the bimodular category $\mathbb{C}$, with its two actions on $\mathbb{V}$. Morphisms in $\mathbb{C}$ need this wire as an input and as an output, while morphisms in $\mathbb{V}$ do not. A detailed discussion of the string diagrams of premonoidal categories was presented to the last Applied Category Conference by this second author [45].

C Functor Boxes

Proposition C.1. *There exists a forgetful functor from the lax functors category to the category of functor box signatures,* $\mathsf{Ulax}\colon \mathbf{Lax} \to \mathbf{Fbox}$.

Proof. Any lax monoidal functor induces a functor box signature $\mathsf{Ulax}(\mathbb{A},\mathbb{X},F,\varepsilon,\mu) = (\mathscr{A},\mathscr{X},\mathscr{F}_\bullet,\mathscr{F}^\bullet)$ defined by $\mathscr{A}_{obj} = \mathbb{A}_{obj}$, by $\mathscr{X}_{obj} = \mathbb{X}_{obj}$ and taking edges to be morphisms,

- $\mathscr{A}(A_0,...,A_n;B_0,...,B_m) = \mathbb{A}(A_0 \otimes ... \otimes A_n; B_0 \otimes ... \otimes B_m)$,
- $\mathscr{X}(X_0,...,X_n;Y_0,...,Y_m) = \mathbb{X}(X_0 \otimes ... \otimes X_n; Y_0 \otimes ... \otimes Y_m)$,
- $\mathscr{F}_\bullet(A_0,...,A_n;Y_0,...,Y_m) = \mathbb{A}(A_0 \otimes ... \otimes A_n; F(Y_0 \otimes ... \otimes Y_m))$,
- $\mathscr{F}^\bullet(X_0,...,X_n;B_0,...,B_m) = \mathbb{A}(F(X_0 \otimes ... \otimes X_n); B_0 \otimes ... \otimes B_m)$.

Consider now a homomorphism of lax monoidal functors $(H,K)\colon (\mathbb{A},\mathbb{X},F) \to (\mathbb{B},\mathbb{Y},G)$. The pair of strict monoidal functors H and K extend to all the sets of edges. For instance, because of the condition $F \mathbin{⨾} K = H \mathbin{⨾} G$, the functor K induces a map

$$\mathbb{A}(A_0 \otimes ... \otimes A_n; F(Y_0 \otimes ... \otimes Y_m)) \to \mathbb{B}(K(A_0) \otimes ... \otimes K(A_n); G(H(Y_0) \otimes ... \otimes H(Y_m))),$$

and the rest of the maps are analogous. This assignment preserves the composition and identities of the lax monoidal functors category, which are precisely compositions and identities of functors. □

Lemma C.2. *The syntactic bicategory of a functor box signature* $(\mathscr{A},\mathscr{X},\mathscr{F})$ *induces a lax monoidal functor* $S_{\mathscr{A},\mathscr{X}}\colon \mathbf{F}_{mon}(\mathscr{X}) \to \mathbb{S}(\mathscr{A},\mathscr{A})$ *from the free monoidal category on* $\mathscr{X}$ *to the monoidal category of the endocells in* $\mathscr{A}$*. This assignment,* $S\colon \mathbf{Fbox} \to \mathbf{Lax}$*, is functorial.*

Proof. We begin by defining the assignment explicitly. We first consider $\mathbf{F}_{mon}(\mathscr{X})$, the free strict monoidal category on the polygraph $\mathscr{X}$. We then consider $\mathbb{S}(\mathscr{A},\mathscr{A})$, the strict monoidal category formed by the endocells of the syntactic 2-category on $\mathscr{A}$. A functor $F\colon \mathbf{F}_{mon}(\mathscr{X}) \to \mathbb{S}(\mathscr{A},\mathscr{A})$ is defined on objects by the composition $F(X_0) = F^{\uparrow} ⨟ X_0 ⨟ F_{\downarrow}$, and similarly on morphisms. It becomes a lax monoidal functor with thanks to the unit and counit maps provided by the adjunction $F^{\uparrow} \dashv F_{\downarrow}$.

We now prove that this assignment is functorial. Consider a functor box signature map determined by $(h,k)\colon (\mathscr{A},\mathscr{X}) \to (\mathscr{B},\mathscr{Y})$, inducing lax monoidal functors $F\colon \mathscr{F}_{mon}(\mathscr{X}) \to \mathbb{S}_{\mathscr{A},\mathscr{X}}(\mathscr{A},\mathscr{A})$ and $G\colon \mathbf{F}_{mon}(\mathscr{Y}) \to \mathbb{S}_{\mathscr{B},\mathscr{Y}}(\mathscr{B},\mathscr{B})$. Because of the adjunction determining free strict monoidal categories, the map $h\colon \mathscr{X} \to \mathscr{Y}$ determines a strict monoidal functor $H\colon \mathbf{F}_{mon}(\mathscr{X}) \to \mathbf{F}_{mon}(\mathscr{Y})$. Now, because the syntactic bicategory of a functor box is also freely generated, we can describe a map of 2-categories $\mathbb{S}_{\mathscr{A},\mathscr{X}} \to \mathbb{S}_{\mathscr{B},\mathscr{Y}}$ induced by the functions h,k and sending the pieces determining the adjunction on one side to the adjunction on the other side. This 2-categorical functor restricts to a strict monoidal functor $K\colon \mathbb{S}_{\mathscr{A},\mathscr{X}}(\mathscr{A},\mathscr{A}) \to \mathbb{S}_{\mathscr{B},\mathscr{Y}}(\mathscr{B},\mathscr{B})$.

Finally, by construction and because the 2-categorical functor sends $F^{\uparrow} \dashv F_{\downarrow}$ to $G^{\uparrow} \dashv G_{\downarrow}$, we have that the H and K here defined satisfy $F ⨟ K = H ⨟ G$ and preserve the structure maps of the lax monoidal functor. □

Theorem C.3 (From Theorem 3.4)**.** *There exists an adjunction between the category of functor box signatures,* **Fbox***, and the category of pairs of strict monoidal categories with a lax monoidal functor between them,* **Lax***. The free side of this adjunction is given by the syntax of Figure 7.*

Proof. Given a functor box signature $(\mathscr{A},\mathscr{X})$ we will prove that the lax monoidal functor induced by its syntactic bicategory, $F\colon \mathscr{F}_{mon}(\mathscr{X}) \to \mathbb{S}_{\mathscr{A},\mathscr{X}}(\mathscr{A},\mathscr{A})$, is the free one in the category **Lax**, where morphisms are pairs of functors instead of the usual natural transformations between functors.

Consider a lax monoidal functor $G\colon \mathbb{B} \to \mathbb{Y}$ endowed with a box signature morphism $(\mathscr{A},\mathscr{X}) \to (\mathbb{B},\mathbb{Y})$. Already by the universal property of the free strict monoidal category, we know that there exists a unique strict monoidal functor $H\colon \mathscr{F}_{mon}(\mathscr{X}) \to \mathscr{F}_{mon}(\mathscr{Y})$ that, under the forgetful functor, commutes with the box signature morphism.

We need to show that there exists a unique strict monoidal functor $K\colon \mathbb{S}_{\mathscr{A},\mathscr{X}}(\mathscr{A},\mathscr{A}) \to \mathbb{S}_{\mathscr{B},\mathscr{Y}}(\mathscr{B},\mathscr{B})$ that commutes with H and with the box signature morphism. We first define it on 1-cells. By structural induction, a 1-cell of $\mathbb{S}_{\mathscr{A},\mathscr{X}}(\mathscr{A},\mathscr{A})$ is: *(i)* an object of $\mathscr{A}$ followed by a 1-cell; or *(ii)* a functor box opening $F^{\uparrow}$, a 1-cell of $\mathscr{F}_{mon}(\mathscr{X})$ and a functor box closing $F_{\downarrow}$, followed by a 1-cell of $\mathbb{S}_{\mathscr{A},\mathscr{X}}(\mathscr{A},\mathscr{A})$. In the first case, the object in $\mathscr{A}$ must be sent to the object determined by the box signature morphism; in the second case, because the condition $F ⨟ K = H ⨟ G$ must be satisfied, we must send the object $F(X_0)$ to $G(H(X_0))$.

We now define it on 2-cells. The plain edges need to be mapped according to the functor box signature homomorphism; the functor box edges are already mapped according to H; the in-box and out-box edges are mapped according to the functor box signature homomorphism. The unit of the adjunction must be preserved because it is a structure map of the lax monoidal functor. Finally, the unit of the adjunction must always appear enclosed in between the cells $F^{\uparrow}$ and $F_{\downarrow}$, which means it always represents the μ_F structure map of the lax monoidal functor and must be mapped accordingly to μ_G. □

D Pointed Bimodular Profunctors

D.1 The Point of Coend Calculus

Coend calculus is the name given to the algebraic manipulations of coends that prove isomorphisms or construct natural transformations between profunctors. In the same way that regular logic links relations, a coend calculus expression is a list of profunctors linked by some objects that are bound to a coend.

Usually, the isomorphisms that we construct are never made explicit, and it is difficult for the reader to compute the precise map we constructed.

Fortunately, this has a straightforward solution. We propose to *point* the coends: to write an expression together with the *generic element* it computes. An expression of pointed coend calculus is a coend bounding some objects and a series of *pointed profunctors*. For instance,

$$\int^{M,N} f \in P(A;M,N) \times g \in Q(M;B) \times h \in \mathbb{C}(N;C), \quad \text{instead of} \quad \int^{M,N} P(A;M,N) \times Q(M;B) \times \mathbb{C}(N;C).$$

The coend quotients expressions by dinaturality, meaning that any action on the left of a coend can be also written as an action on the right. In terms of pointed profunctors, this means that

$$\int^{N} (f < h) \in P(A;N) \times g \in Q(N;B) = \int^{M} f \in P(A;M) \times (h > g) \in Q(M;B).$$

Proposition D.1. *Let $\mathbb{C}$ be a category and let $F\colon \mathbb{C}^{op} \to$ **Set** and $G\colon \mathbb{C} \to$ **Set** be a presheaf and a copresheaf, respectively. The following are natural isomorphisms of pointed profunctors,*

$$\int^{X} f \in \mathbb{C}(X;A) \times h \in F(A) \;\cong\; (f > h) \in F(X); \qquad \int^{X} f \in \mathbb{C}(A;X) \times h \in G(A) \;\cong\; (h < f) \in G(X).$$

We call these isomorphisms the "pointed" Yoneda reductions.

Remark D.2. Using pointed coends, any derivation does also include the computation of the isomorphism it induces. As an example, compare the following with the usual coend derivation of a cartesian lens [13],

Proposition D.3. *In a cartesian monoidal category, the pairs of morphisms $f \in \mathbb{C}(A;M \times X)$ and $g \in \mathbb{C}(M \times Y;B)$, quotiented by dinaturality, are in bijective correspondence with the pairs of morphisms $\mathbb{C}(A;M)$ and $\mathbb{C}(M \times Y;B)$.*

$$\begin{aligned}
&\int^{M} f \in \mathbb{C}(A;M \times X) \times g \in \mathbb{C}(M \times Y;B) && \cong (\textit{ by the adjunction } \Delta \dashv \times)\\
&\int^{M} (f \mathbin{⨾} \pi_1) \in \mathbb{C}(A;M) \times (f \mathbin{⨾} \pi_2) \in \mathbb{C}(A;M) \times g \in \mathbb{C}(M \times Y;B) && \cong (\textit{ by pointed Yoneda lemma })\\
&(f \mathbin{⨾} \pi_2) \in \mathbb{C}(A;M) \times ((f \mathbin{⨾} \pi_1) \otimes id) \mathbin{⨾} g \in \mathbb{C}(M \times Y;B).
\end{aligned}$$

In the first step, we have used that the adjunction $(\Delta \dashv \times)$ is given by postcomposition with projections and; in the second step, we use that the action on the last profunctor is defined as $h > g = (h \otimes id) \mathbin{⨾} g$. The bijection has been explicitly constructed as sending the pair $(f;g)$ to $(f \mathbin{⨾} \pi_2; ((f \mathbin{⨾} \pi_1) \otimes id) \mathbin{⨾} g)$.

D.2 Semantics of Functor Boxes

Proposition D.4 (Bimodular categories of a lax monoidal functor, from Proposition 4.9)**.** *Let $\mathbb{X}$ and $\mathbb{A}$ be two monoidal categories and let $F\colon \mathbb{X} \to \mathbb{A}$ be a monoidal functor between them, endowed with natural transformations $\psi_0\colon J \to FI$ and $\psi_2\colon FX \otimes FY \to F(X \otimes Y)$. The following profunctors, $\mathbb{A} \rtimes_F \mathbb{X}\colon \mathbb{A} \times \mathbb{X} \to \mathbb{A} \times \mathbb{X}$ and $\mathbb{X} \ltimes_F \mathbb{A}\colon \mathbb{X} \times \mathbb{A} \to \mathbb{X} \times \mathbb{A}$ determine two promonads, and therefore two Kleisli categories.*

$$\begin{aligned}
\mathbb{A} \rtimes_F \mathbb{X}(A,X;B,Y) &= \int^{M \in \mathbb{X}} \mathbb{A}(A;B \otimes FM) \times \mathbb{X}(M \otimes X;Y);\\
\mathbb{X} \ltimes_F \mathbb{A}(X,A;Y,B) &= \int^{M \in \mathbb{X}} \mathbb{A}(A;FM \otimes B) \times \mathbb{X}(M \otimes A;B);
\end{aligned}$$

These two Kleisli categories are $(\mathbb{A},\mathbb{X})$ and $(\mathbb{X},\mathbb{A})$-bimodular, respectively.

Proof. We prove that $\mathbb{A} \rtimes_F \mathbb{X}$ define a promonad and, in particular, the hom-sets of a category. We write the elements $\mathbb{A} \rtimes_F \mathbb{X}(A,X;B,Y)$ are given by pairs (f,α) where $f\colon A \to B \otimes FM$ and $\alpha\colon M \otimes X \to Y$ for some $M \in \mathbb{X}_{obj}$.

The unit of the promonad sends a pair of morphisms $u\colon A \to B$ and $r\colon X \to Y$ to the morphism $(u \otimes \psi_0, r)$, where $u \otimes \psi_0\colon A \to B \otimes FI$ and, modulo coherence, $r\colon I \otimes X \to Y$. The multiplication of the promonad sends $(f,\alpha)\colon (A,X) \to (Y,B)$ and $(g,\beta)\colon (Y,B) \to (Z,C)$, to the composite formed by $f ⨟ (g \otimes \mathrm{id}) ⨟ (\mathrm{id} \otimes \psi_2)\colon A \to Z \otimes F(M \otimes N)$ and $(\mathrm{id} \otimes \alpha) ⨟ \beta\colon N \otimes M \otimes A \to B$. Finally, from the axioms of lax monoidal functors, it follows that this composition is associative and unital. □

E Internal Diagrams

Theorem E.1 (From Theorem 5.3)**.** *For any interpretation of a polygraph into a monoidal category, there exists a 3-functor from the syntactic tricategory of internal diagrams into pointed bimodular profunctors that preserves this interpretation.*

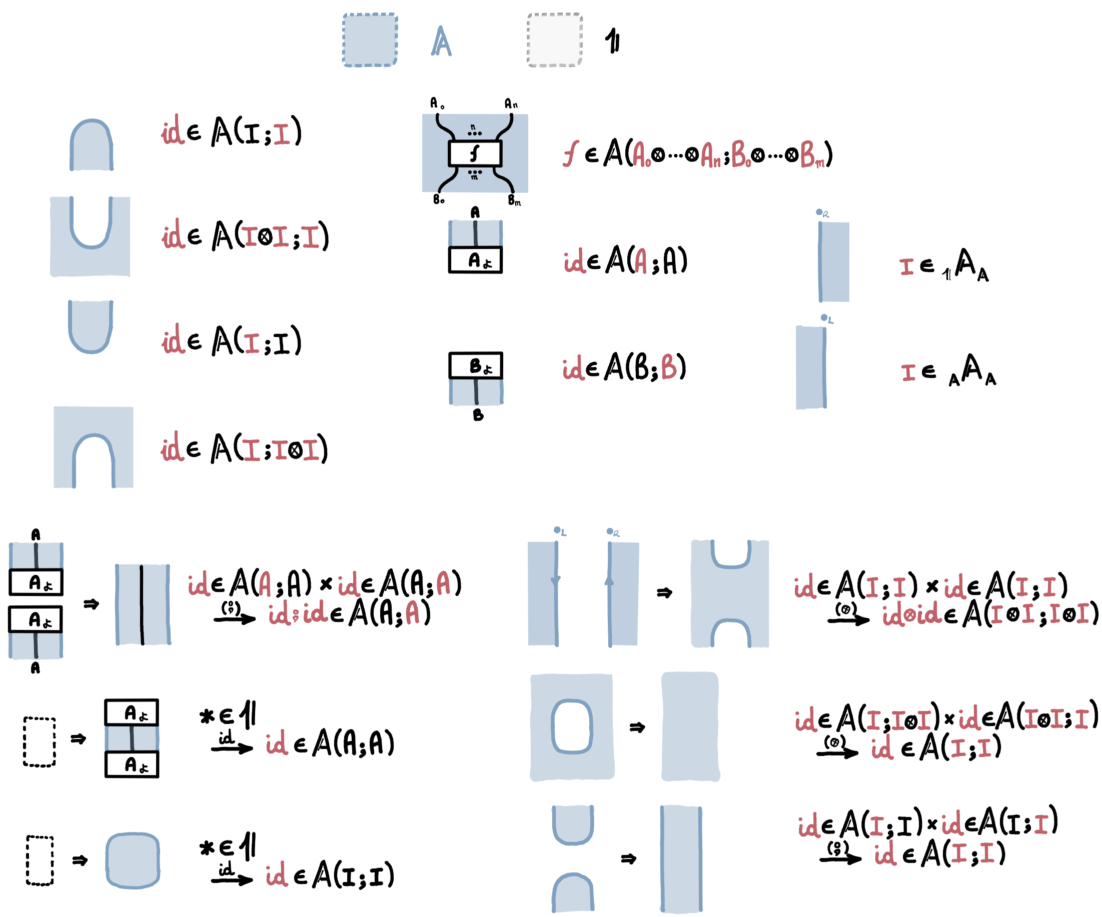


Figure 17: Semantics for open internal diagrams in terms of pointed bimodular profunctors.

Proof. The syntactic tricategory of internal diagrams has been constructed as a free tricategory, so it suffices to determine where the generators are sent. For this, we follow Figure 17. Let the square brackets, $[\![\bullet]\!]$, denote the interpretation of the polygraph into a monoidal category.

The region $\mathscr{G}$ is sent to the monoidal category $[\![\mathscr{G}]\!] = (\mathbb{A}, \otimes, I)$, while the region $\mathscr{I}$ is sent to the terminal monoidal category. The generator $L_\bullet$ is sent to the pointed bimodular category $({}_{\mathbb{1}}\mathbb{A}_{\mathbb{A}}, I)$, while the generator $R_\bullet$ is sent to the bimodular category $({}_{\mathbb{A}}\mathbb{A}_{\mathbb{1}}, I)$. The generator A is sent to the pointed bimodular category $({}_{\mathbb{A}}\mathbb{A}_{\mathbb{A}}, [\![A]\!])$.

Let us consider the 2-cells. The 2-cells n_1 and e_2 are sent to the profunctors $\mathbb{A}(\bullet \otimes \bullet; \bullet)$ and $\mathbb{A}(\bullet; \bullet \otimes \bullet)$, pointed in the identities of the monoidal unit. The 2-cells n_2 and e_1 are sent to the profunctors $\mathbb{A}(I; \bullet)$ and $\mathbb{A}(\bullet; I)$, pointed in the identities of the monoidal unit.

The 2-cells $A^{\pitchfork}$ and $A_{\pitchfork}$ are sent to the representable and corepresentable profunctors of the object $[\![A]\!]$, which are pointed in id_A, the identity on that object. Finally, any 2-cell arising from an edge $f \in \mathscr{G}(A_0, \dots, A_n; B_0, \dots, B_m)$ is sent to the hom-profunctor, pointed in the relevant morphism,

$$[\![f]\!] \in \mathbb{A}([\![A_0]\!] \otimes \dots \otimes [\![A_n]\!]; [\![B_0]\!] \otimes \dots \otimes [\![B_m]\!]).$$

It is well-known that every representable profunctor is adjoint to its corepresentable profunctor, which gives semantics to the syntactic adjunctions. The only adjunctions missing are that between ${}_{\mathbb{A}}\mathbb{A}_{\mathbb{1}}$ and ${}_{\mathbb{1}}\mathbb{A}_{\mathbb{A}}$; for these, we must note that there is a Yoneda isomorphism of the following form,

$$\int^{M} \mathbb{A}(I; M) \times \mathbb{A}(M \otimes X; Y) \cong \mathbb{A}(X; Y), \quad \int^{M} \mathbb{A}(X; Y \otimes M) \times \mathbb{A}(M; I) \cong \mathbb{A}(X; Y),$$

and analogous ones swapping the position of M in the tensor product. □

F The Collage of Bimodular Profunctors

To define collages in greater generality we use the notion of a *bimodular pasting diagram*: a composable arrangement of bimodular categories and profunctors. For a precise definition of 2-dimensional pasting diagrams, see the reference book by Johnson and Yau [28, Chapter 3].

Definition F.1 (Collages of Bimodular Profunctors)**.** Let $\mathscr{D}$ be a bimodular pasting diagram. We define a bicategory $\mathsf{Coll}(\mathscr{D})$ as follows:

- there is an object M for each vertex in $\mathscr{D}$, labelled by a monoidal category $\mathbb{M}$;
- endomorphism categories $\mathsf{Coll}(\mathscr{D})(M, M)$ are given by $\mathbb{M}$ with its monoidal structure;
- for each 1-edge labelled by an $(\mathbb{M}, \mathbb{N})$-bimodular category $\mathbb{C}$, and each object $C \in \mathbb{C}$, we have a 1-cell $(\mathbb{C}, C) : M \to N$;
- for each 2-edge labelled by an $(\mathbb{M}, \mathbb{N})$-bimodular profunctor $P : \mathbb{C} \to \mathbb{D}$, and each $p \in P(C, D)$ we have a 2-cell $(p, C, D) : (\mathbb{C}, C) \to (\mathbb{D}, D)$;
- compositions are given by monoidal actions, actions of morphisms on profunctors, or quotienting maps of tensor products, where relevant.

Example F.2. In the case where the pasting diagram $\mathscr{D}$ is a single edge labelled by a bimodular category $\mathbb{C}$, we recover our earlier notion of collage from Definition 2.5.

Example F.3. Each of the profunctors in Figure 8 have a collage whose 2-cells describe string diagrams for the section of functor box depicted. More complex composable arrangements of these profunctors can be assembled giving rise to bicategories modelling whole functors boxes or arrangements of them.

Remark F.4. The definition of a bimodular pasting diagram can be seen as that of a trifunctor from the free 2-category on a 2-graph to a tricategory of monoidal categories, bimodular categories, bimodular profunctors, and natural transformations. We conjecture that the collage described here realises a lax colimit of such a diagram, where the target category is enlarged to a tricategory of bicategories and "2-profunctors" [11, 31].

Conjecture F.5. *The collage of a diagram of bimodular profunctors is the lax 3-colimit of this diagram when viewed as a functor into a tricategory of 2-profunctors between 2-categories*

$$\mathsf{Coll}(\mathscr{D}) = \mathsf{Colim}_{\mathsf{lax}} \left(\mathbb{I} \xrightarrow{\mathscr{D}} \mathbb{BmProf} \hookrightarrow 2\mathbb{Prof} \right)$$

Moreover the tricategory of pointed bimodular profunctors is the universal collage, given by the identity diagram

$$\mathbb{BmProf}_{\mathsf{pt}} = \mathsf{Colim}_{\mathsf{lax}} \left(\mathbb{BmProf} = \mathbb{BmProf} \hookrightarrow 2\mathbb{Prof} \right).$$

This perspective unifies our construction with the typical notion of a *collage* of profunctors [50], which can be considered as a lax colimit of a functor into the bicategory of profunctors. Additionally, this elucidates the relationship between the various syntactic bicategories we have constructed, and tricategory of pointed bimodulars which we pronounced as a universe in which all such diagrams can live. Indeed, if $\mathbb{BmProf}_{\mathsf{pt}}$ is a colimit of the terminal diagram, then we should obtain inclusions of all collages into this tricategory. We leave the development of these notions for further work.