Theory and Applications of Categories, Vol. 35, No. 48, 2020, pp. 1771–1822.

STRUCTURED COSPANS

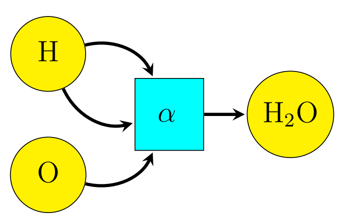
JOHN C. BAEZ AND KENNY COURSER

Abstract. One goal of applied category theory is to better understand networks appearing throughout science and engineering. Here we introduce 'structured cospans' as a way to study networks with inputs and outputs. Given a functor $L\colon \mathsf{A} \to \mathsf{X}$, a structured cospan is a diagram in X of the form $L(a) \to x \leftarrow L(b)$. If A and X have finite colimits and L is a left adjoint, we obtain a symmetric monoidal category whose objects are those of A and whose morphisms are isomorphism classes of structured cospans. This is a hypergraph category. However, it arises from a more fundamental structure: a symmetric monoidal double category where the horizontal 1-cells are structured cospans. We show how structured cospans solve certain problems in the closely related formalism of 'decorated cospans', and explain how they work in some examples: electrical circuits, Petri nets, and chemical reaction networks.

1. Introduction

Structured cospans are a framework for dealing with open networks: that is, networks with inputs and outputs. Networks arise in many areas of science and engineering and come in many kinds, but a companion paper illustrates the general framework developed here with the example of open Petri nets [5], so let us consider those.

Petri nets are important in computer science, chemistry and other subjects. For example, the chemical reaction that takes two atoms of hydrogen and one atom of oxygen and produces a molecule of water can be represented by this very simple Petri net:

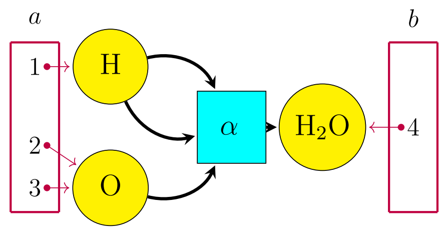


Here we have a set of 'places' (or in chemistry, 'species') drawn in yellow and a set of 'transitions' (or 'reactions') drawn in blue. The disjoint union of these two sets then forms the vertex set of a directed bipartite graph, which is one description of a Petri net.

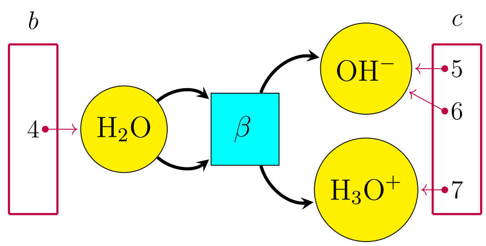
Networks can often be seen as pieces of larger networks. This naturally leads to the idea of an *open* Petri net, meaning that the set of places is equipped with 'inputs' and 'outputs'. We can do this by prescribing two functions into the set of places that pick out

Received by the editors 2020-01-04 and, in final form, 2020-10-27.
Transmitted by Joachim Kock. Published on 2020-10-29.
2020 Mathematics Subject Classification: 18B10, 18M35, 18N10.
Key words and phrases: bicategory, cospan, double category, monoidal category, network.
© John C. Baez and Kenny Courser, 2020. Permission to copy for private use granted.

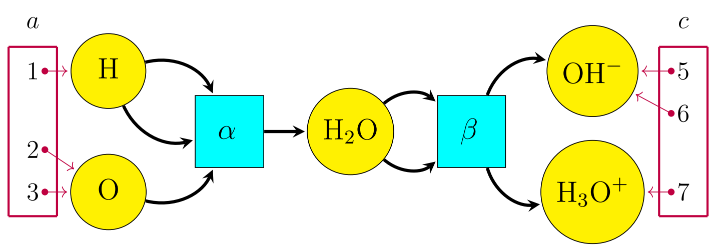
these inputs and outputs. For example:



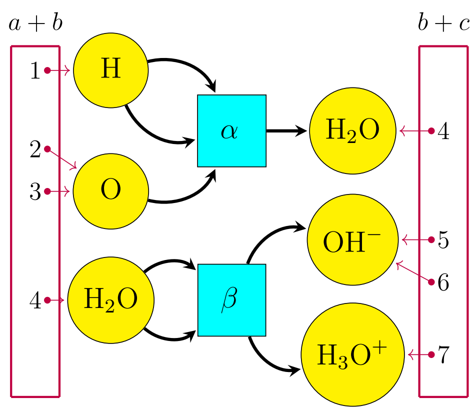
The inputs and outputs let us compose open Petri nets. For example, suppose we have another open Petri net that represents the chemical reaction of two molecules of water turning into hydronium and hydroxide:



Since the outputs of the first open Petri net coincide with the inputs of the second, we can compose them by identifying the outputs of the first with the inputs of the second:



Similarly we can 'tensor' two open Petri nets by placing them side by side:



We can formalize this example using 'structured cospans'. Given a functor $L\colon \mathsf{A} \to \mathsf{X}$, a **structured cospan** is a diagram in X of the form

$$\begin{array}{ccccc} & & x & & \\ & \overset{i}{\nearrow} & & \overset{o}{\nwarrow} & \\ L(a) & & & & L(b). \end{array}$$

The objects a and b are called the **input** and **output**, respectively, while x is called the **apex**. The morphisms i and o are called the **legs** of the cospan.

Typically the input and output of a structured cospan are simpler in nature than the apex. For example, an open Petri net is a structured cospan where a and b are sets while x is a Petri net. As explained in Section 6.6, there is a category Petri with Petri nets as objects and a functor $L\colon \mathsf{Set} \to \mathsf{Petri}$ sending any set to the Petri net with that set of places and no transitions. Furthermore, L is a left adjoint, so it preserves colimits. This occurs in many examples.

Given a functor $L\colon \mathsf{A} \to \mathsf{X}$, we can compose structured cospans whenever X has pushouts. In Corollary 2.5 we show this gives a category ${}_L\mathsf{Csp}(\mathsf{X})$ with:

- objects of A as objects,
- isomorphism classes of structured cospans as morphisms.

Here we say two structured cospans $L(a) \to x \leftarrow L(b)$ and $L(a) \to y \leftarrow L(b)$ are **isomorphic** if there is an isomorphism $f\colon x \to y$ such that the diagram

$$\begin{array}{ccccc} & \nearrow & x & \nwarrow & \\ L(a) & & \downarrow f & & L(b) \\ & \searrow & y & \swarrow & \end{array}$$

commutes. In Corollary 3.11 we show this category ${}_L\mathsf{Csp}(\mathsf{X})$ becomes symmetric monoidal when A and X have finite colimits and L preserves them. Under these assumptions, in Theorem 3.12 we prove that ${}_L\mathsf{Csp}(\mathsf{X})$ is actually a special sort of symmetric monoidal category called a 'hypergraph category' [17]. These are important in the theory of networks [13, 14].

Sometimes it is inconvenient to work with isomorphism classes of structured cospans. For example, in an open Petri net we can refer to a particular place or transition; in an isomorphism class of open Petri nets we cannot. To use actual structured cospans as morphisms we need a higher categorical structure, because composing them is associative only up to isomorphism. Indeed, in Corollary 2.4 we show that for any functor $L\colon \mathsf{A} \to \mathsf{X}$, if X has pushouts there is a bicategory ${}_L\mathbf{Csp}(\mathsf{X})$ with:

- objects of A as objects,
- structured cospans as 1-morphisms,

- commutative diagrams

$$\begin{array}{ccccc} & \nearrow & x & \nwarrow & \\ L(a) & & \Big\downarrow f & & L(b) \\ & \searrow & y & \swarrow & \end{array}$$

as 2-morphisms.

In Corollary 3.10 we show that the bicategory ${}_L\mathbf{Csp}(\mathsf{X})$ is symmetric monoidal when A and X have finite colimits and L preserves them. However, the coherence laws for a symmetric monoidal bicategory are rather complicated [35]. As noted by Ehresmann [12], and then by Grandis and Paré [19, 20], double categories are sometimes more convenient than bicategories. This is especially true in the symmetric monoidal case [22, 34]. Thus we show in Theorem 2.3 that for any functor $L\colon \mathsf{A} \to \mathsf{X}$, if X has pushouts there is a double category ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ with:

- objects of A as objects,
- morphisms of A as vertical 1-morphisms,
- structured cospans as horizontal 1-cells,
- commutative diagrams

$$\begin{array}{ccccc} L(a) & \xrightarrow{i} & x & \xleftarrow{o} & L(b) \\ {\scriptstyle L(\alpha)}\Big\downarrow & & \Big\downarrow{\scriptstyle f} & & \Big\downarrow{\scriptstyle L(\beta)} \\ L(a') & \xrightarrow[i']{} & x' & \xleftarrow[o']{} & L(b') \end{array}$$

as 2-morphisms.

Note that vertical composition in this double category is strictly associative, while horizontal composition is not. In Theorem 3.9 we show that ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ is a symmetric monoidal double category when A and X have finite colimits and L preserves them. Using Shulman's work [34], we conclude in Corollary 3.10 that the bicategory ${}_L\mathbf{Csp}(\mathsf{X})$ is a symmetric monoidal bicategory under the same conditions.

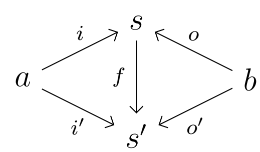
The reader familiar with decorated cospans may wonder why we need structured cospans. Recall that Fong [13] constructed a category of 'decorated cospans' $F\mathsf{Cospan}$ from any category A with finite colimits together with a symmetric lax monoidal functor $F\colon (\mathsf{A}, +) \to (\mathsf{Set}, \times)$. The objects of $F\mathsf{Cospan}$ are those of A, while the morphisms are equivalence classes of F-decorated cospans. Here an ***F*-decorated cospan** is a pair

$$(a \xrightarrow{\;i\;} s \xleftarrow{\;o\;} b,\ d \in F(s)).$$

The element d, called the **decoration**, serves as a way to equip the apex s with extra structure. The above decorated cospan is equivalent to

$$(a \xrightarrow{i'} s' \xleftarrow{o'} b, d' \in F(s'))$$

iff there an isomorphism $f\colon s \to s'$ in A making this diagram commute:



and such that $F(f)(d) = d'$.

Both decorated and structured cospans are ways to describe a cospan whose apex is equipped with extra structure. Since the theory of decorated cospans is already well-developed, what is the point of another formalism? One reason is that structured cospans are a bit simpler: instead of a symmetric lax monoidal functor $F\colon \mathsf{A} \to \mathsf{Set}$ assigning to each object of A the set of possible structures we can put on it, we can simply use a left adjoint L from A to any category X. Another reason is that structured cospans solve some problems that prevent decorated cospans from being applied as originally intended, and indeed led to errors in a number of published papers. We discuss these problems, and how structured cospans get around them, in Section 5. In Section 6 we study applications of structured cospans to electrical circuits, Petri nets and chemical reaction networks.

Conventions. In this paper, 'double category' means 'pseudo double category', as in Definition A.1. Following Shulman [34], vertical composition in our double categories is strictly associative, while horizontal composition need not be. We use sans-serif font like C for categories, boldface like $\mathbf{B}$ for bicategories or 2-categories, and blackboard bold like $\mathbb{D}$ for double categories. We also use blackboard bold for weak category objects in any 2-category. For double categories with names having more than one letter, like $\mathbb{C}\mathbf{sp}(\mathsf{X})$, only the first letter is in blackboard bold. A double category $\mathbb{D}$ has a category of objects and a category of arrows, and we call these $\mathbb{D}_0$ and $\mathbb{D}_1$ despite the fact that they are categories.

Acknowledgements. The authors would like to thank Christina Vasilakopoulou for the clever idea of replacing the category of objects of some double category by some other category. We would also like to thank Marco Grandis and Robert Paré for pointing out the importance of double categories with double colimits, and Joachim Kock and Mike Shulman for catching errors.

2. Structured cospans

Given a functor $L\colon \mathsf{A} \to \mathsf{X}$, a **structured cospan** is a cospan in X whose feet come from a pair of objects in A:

$$\begin{array}{ccc} & x & \\ \nearrow & & \nwarrow \\ L(a) & & L(b). \end{array}$$

When L has a right adjoint $R\colon \mathsf{X} \to \mathsf{A}$ we can also think of this as a cospan in A,

$$\begin{array}{ccc} & R(x) & \\ \nearrow & & \nwarrow \\ a & & b, \end{array}$$

where the apex is equipped with extra structure, namely an object $x \in \mathsf{X}$ that it comes from. However, treating structured cospans as living in X is technically more convenient, since then we only need X to have pushouts to compose them. In Theorem 2.3 we show that when X has pushouts, structured cospans are the horizontal 1-cells of a double category ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$. To prove this we begin by recalling the double category of cospans in X. For the definition of double category see Appendix A.

2.1. Lemma. *Given a category X with chosen pushouts, there is a double category $\mathbb{C}\mathbf{sp}(\mathsf{X})$ in which:*

- *an object is an object of X,*
- *a vertical 1-morphism is a morphism of X,*
- *a horizontal 1-cell from x_1 to x_2 is a cospan in X:*

$$x_1 \xrightarrow{\;i\;} y \xleftarrow{\;o\;} x_2$$

- *a 2-morphism is a commutative diagram in X of this form:*

$$\begin{array}{ccccc} x_1 & \xrightarrow{\;i\;} & y & \xleftarrow{\;o\;} & x_2 \\ {\scriptstyle f_1}\downarrow & & \downarrow{\scriptstyle g} & & \downarrow{\scriptstyle f_2} \\ x_1' & \xrightarrow[\;i'\;]{} & y' & \xleftarrow[\;o'\;]{} & x_2', \end{array}$$

- *composition of vertical 1-morphisms is composition in X,*

- *composition of horizontal 1-cells is done using the chosen pushouts in* X*:*

$$
\begin{array}{ccccccc}
 & & & y +_{x_2} z & & & \\
 & & \overset{j_y}{\nearrow} & & \overset{j_z}{\nwarrow} & & \\
 & y & & & & z & \\
 \overset{i_1}{\nearrow} & & \overset{o_1}{\nwarrow} & & \overset{i_2}{\nearrow} & & \overset{o_2}{\nwarrow} \\
x_1 & & & x_2 & & & x_3
\end{array}
$$

where j_y *and* j_z *are the canonical morphisms from* y *and* z *to the pushout object,*

- *the horizontal composite of two 2-morphisms:*

$$
\begin{array}{ccccc}
x_1 & \xrightarrow{i_1} & y & \xleftarrow{o_1} & x_2 \\
{\scriptstyle f_1}\downarrow & & {\scriptstyle g}\downarrow & & \downarrow{\scriptstyle f_2} \\
x_1' & \xrightarrow{i_1'} & y' & \xleftarrow{o_1'} & x_2'
\end{array}
\qquad
\begin{array}{ccccc}
x_2 & \xrightarrow{i_2} & z & \xleftarrow{o_2} & x_3 \\
{\scriptstyle f_2}\downarrow & & {\scriptstyle h}\downarrow & & \downarrow{\scriptstyle f_3} \\
x_2' & \xrightarrow{i_2'} & z' & \xleftarrow{o_2'} & x_3'
\end{array}
$$

is given by

$$
\begin{array}{ccccc}
x_1 & \xrightarrow{j_y i_1} & y +_{x_2} z & \xleftarrow{j_z o_2} & x_3 \\
{\scriptstyle f_1}\downarrow & & {\scriptstyle g +_{f_2} h}\downarrow & & \downarrow{\scriptstyle f_3} \\
x_1' & \xrightarrow[j_{y'} i_1']{} & y' +_{x_2'} z' & \xleftarrow[j_{z'} o_2']{} & x_3'.
\end{array}
$$

- *the vertical composite of two 2-morphisms:*

$$
\begin{array}{ccccc}
x_1 & \xrightarrow{i} & y & \xleftarrow{o} & x_2 \\
{\scriptstyle f_1}\downarrow & & {\scriptstyle g}\downarrow & & \downarrow{\scriptstyle f_2} \\
x_1' & \xrightarrow[i']{} & y' & \xleftarrow[o']{} & x_2'
\end{array}
$$

$$
\begin{array}{ccccc}
x_1' & \xrightarrow{i'} & y' & \xleftarrow{o'} & x_2' \\
{\scriptstyle f_1'}\downarrow & & {\scriptstyle g'}\downarrow & & \downarrow{\scriptstyle f_2'} \\
x_1'' & \xrightarrow[i'']{} & y'' & \xleftarrow[o'']{} & x_2''
\end{array}
$$

is given by

$$
\begin{array}{ccccc}
x_1 & \xrightarrow{i} & y & \xleftarrow{o} & x_2 \\
{\scriptstyle f_1' f_1}\downarrow & & {\scriptstyle g' g}\downarrow & & \downarrow{\scriptstyle f_2' f_2} \\
x_1'' & \xrightarrow[i'']{} & y'' & \xleftarrow[o'']{} & x_2''
\end{array}
$$

- *the associator and unitors are defined using the universal property of pushouts.*

PROOF. This is well known [9, 31]. ∎

We expect that a different choice of pushouts in X will give an equivalent double category $\mathbb{C}\mathbf{sp}(\mathsf{X})$, since pushouts are unique up to canonical isomorphism.

To build structured cospan double categories, we use a method we learned from Christina Vasilakopoulou for taking a double category $\mathbb{X}$ and replacing its objects and vertical 1-morphisms with the objects and morphisms of some category A. In Appendix A, we recall that any double category $\mathbb{X}$ has a category $\mathbb{X}_0$ called its **category of objects**, whose objects are those of $\mathbb{X}$ and whose morphisms are the vertical 1-morphisms of $\mathbb{X}$. We can replace the category of objects by A using a functor $L\colon \mathsf{A} \to \mathbb{X}_0$.

2.2. LEMMA. *Given a double category $\mathbb{X}$, a category A and a functor $L\colon \mathsf{A} \to \mathbb{X}_0$, there is a double category ${}_L\mathbb{X}$ in which:*

- *an object is an object of A,*
- *a vertical 1-morphism is a morphism of A,*
- *a horizontal 1-cell from a to b is a horizontal 1-cell $L(a) \xrightarrow{M} L(b)$ of $\mathbb{X}$,*
- *a 2-morphism is a 2-morphism in $\mathbb{X}$ of the form:*

$$\begin{array}{ccc} L(a) & \xrightarrow{M} & L(b) \\ {\scriptstyle L(f)}\downarrow & \Downarrow \alpha & \downarrow{\scriptstyle L(g)} \\ L(a') & \xrightarrow{N} & L(b'), \end{array}$$

- *composition of vertical 1-morphisms is composition in A*
- *composition of horizontal 1-morphisms is defined as in $\mathbb{X}$,*
- *vertical and horizontal composition of 2-morphisms are defined as in $\mathbb{X}$,*
- *the associator and unitors are defined as in $\mathbb{X}$.*

PROOF. It is easy to check the double category axioms using the fact that $\mathbb{X}$ is a double category and L is a functor. ∎

Putting the above lemmas together, we obtain our double category of structured cospans. We describe it quite explicitly for reference purposes:

2.3. THEOREM. *Let $L\colon \mathsf{A} \to \mathsf{X}$ be a functor where X is a category with chosen pushouts. Then there is a double category ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ in which:*

- *an object is an object of A,*
- *a vertical 1-morphism is a morphism of A,*

- *a horizontal 1-cell from a to b is a diagram in* X *of this form:*

$$L(a) \xrightarrow{i} x \xleftarrow{o} L(b)$$

- *a 2-morphism is a commutative diagram in* X *of this form:*

$$\begin{array}{ccccc} L(a) & \xrightarrow{i} & x & \xleftarrow{o} & L(b) \\ {\scriptstyle L(\alpha)}\downarrow & & {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle L(\beta)} \\ L(a') & \xrightarrow[i']{} & x' & \xleftarrow[o']{} & L(b') \end{array}$$

- *composition of horizontal 1-cells is done using the chosen pushouts in* X*:*

$$\begin{array}{ccccccc} & & & x +_{L(b)} y & & & \\ & & {\scriptstyle j_x}\nearrow & & \nwarrow{\scriptstyle j_y} & & \\ & & x & & y & & \\ & {\scriptstyle i_1}\nearrow & & \nwarrow{\scriptstyle o_1} \quad {\scriptstyle i_2}\nearrow & & \nwarrow{\scriptstyle o_2} & \\ L(a) & & & L(b) & & & L(c) \end{array}$$

where j_x and j_y are the canonical morphisms from x and y to the pushout object,

- *identity horizontal 1-cells are diagrams of this form:*

$$L(a) \xrightarrow{1} L(a) \xleftarrow{1} L(a)$$

- *the horizontal composite of two 2-morphisms:*

$$\begin{array}{ccccc} L(a) & \xrightarrow{i_1} & x & \xleftarrow{o_1} & L(b) \\ {\scriptstyle L(\alpha)}\downarrow & & {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle L(\beta)} \\ L(a') & \xrightarrow[i_1']{} & x' & \xleftarrow[o_1']{} & L(b') \end{array} \qquad \begin{array}{ccccc} L(b) & \xrightarrow{i_2} & y & \xleftarrow{o_2} & L(c) \\ {\scriptstyle L(\beta)}\downarrow & & {\scriptstyle g}\downarrow & & \downarrow{\scriptstyle L(\gamma)} \\ L(b') & \xrightarrow[i_2']{} & y' & \xleftarrow[o_2']{} & L(c') \end{array}$$

is given by

$$\begin{array}{ccccc} L(a) & \xrightarrow{j_x i_1} & x +_{L(b)} y & \xleftarrow{j_y o_2} & L(c) \\ {\scriptstyle L(\alpha)}\downarrow & & {\scriptstyle f +_{L(\beta)} g}\downarrow & & \downarrow{\scriptstyle L(\gamma)} \\ L(a') & \xrightarrow[j_{x'} i_1']{} & x' +_{L(b')} y' & \xleftarrow[j_{y'} o_2']{} & L(c') \end{array}$$

- *the identities for horizontal composition of 2-morphisms are diagrams of this form:*

$$\begin{array}{ccccc} L(a) & \xrightarrow{1} & L(a) & \xleftarrow{1} & L(a) \\ {\scriptstyle L(\alpha)}\downarrow & & {\scriptstyle L(\alpha)}\downarrow & & \downarrow{\scriptstyle L(\alpha)} \\ L(a') & \xrightarrow[1]{} & L(a') & \xleftarrow[1]{} & L(a') \end{array}$$

- *the vertical composite of two 2-morphisms:*

$$\begin{array}{ccccc} L(a) & \xrightarrow{i} & y & \xleftarrow{o} & L(b) \\ {\scriptstyle L(\alpha)}\downarrow & & {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle L(\beta)} \\ L(a') & \xrightarrow[i']{} & y' & \xleftarrow[o']{} & L(b') \end{array}$$

$$\begin{array}{ccccc} L(a') & \xrightarrow{i'} & y' & \xleftarrow{o'} & L(b') \\ {\scriptstyle L(\alpha')}\downarrow & & {\scriptstyle f'}\downarrow & & \downarrow{\scriptstyle L(\beta')} \\ L(a'') & \xrightarrow[i'']{} & y'' & \xleftarrow[o'']{} & L(b'') \end{array}$$

is given by

$$\begin{array}{ccccc} L(a) & \xrightarrow{i} & y & \xleftarrow{o} & L(b) \\ {\scriptstyle L(\alpha'\alpha)}\downarrow & & {\scriptstyle f'f}\downarrow & & \downarrow{\scriptstyle L(\beta'\beta)} \\ L(a'') & \xrightarrow[i'']{} & y'' & \xleftarrow[o'']{} & L(b'') \end{array}$$

- *the associator and unitors are defined using the universal property of pushouts.*

PROOF. We apply Lemma 2.2 to the double category $\mathbb{C}\mathbf{sp}(\mathsf{X})$ of Lemma 2.1. ■

From the double category ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ we can extract a bicategory ${}_L\mathbf{Csp}(\mathsf{X})$ and then a category ${}_L\mathsf{Csp}(\mathsf{X})$. In many applications all we need is a bicategory or even a mere category of structured cospans, so the reader should not get the misimpression that working with structured cospans *requires* using double categories. We begin with the bicategory:

2.4. COROLLARY. *Let $L\colon \mathsf{A} \to \mathsf{X}$ be a functor where X is a category with chosen pushouts. Then there is a bicategory ${}_L\mathbf{Csp}(\mathsf{X})$ in which:*

- *an object is an object of A,*
- *a morphism from a to b is a diagram in X of this form:*

$$L(a) \xrightarrow{i} x \xleftarrow{o} L(b)$$

- *a 2-morphism is a commutative diagram in* X *of this form:*

$$\begin{array}{ccccc} & & x & & \\ & \overset{i}{\nearrow} & & \overset{o}{\nwarrow} & \\ L(a) & & f\downarrow & & L(b) \\ & \underset{i'}{\searrow} & & \underset{o'}{\swarrow} & \\ & & x' & & \end{array}$$

- *composition of morphisms is done using the chosen pushouts in* X*,*
- *identity morphisms are of this form:*

$$L(a) \xrightarrow{1} L(a) \xleftarrow{1} L(a)$$

- *the horizontal composite of 2-morphisms:*

$$\begin{array}{ccccc} & & x & & \\ & \overset{i_1}{\nearrow} & & \overset{o_1}{\nwarrow} & \\ L(a) & & f\downarrow & & L(b) \\ & \underset{i'_1}{\searrow} & & \underset{o'_1}{\swarrow} & \\ & & x' & & \end{array} \qquad \begin{array}{ccccc} & & y & & \\ & \overset{i_2}{\nearrow} & & \overset{o_2}{\nwarrow} & \\ L(b) & & g\downarrow & & L(c) \\ & \underset{i'_2}{\searrow} & & \underset{o'_2}{\swarrow} & \\ & & y' & & \end{array}$$

  is given by

$$\begin{array}{ccccc} & & x +_{L(b)} y & & \\ & \overset{j_x i_1}{\nearrow} & & \overset{j_y o_2}{\nwarrow} & \\ L(a) & & f +_{L(1_b)} g \downarrow & & L(c) \\ & \underset{j_{x'} i'_1}{\searrow} & & \underset{j_{y'} o'_2}{\swarrow} & \\ & & x' +_{L(b)} y' & & \end{array}$$

  where j_x *and* j_y *are the canonical morphisms from* x *and* y *to the pushout object* $x +_{L(b)} y$*, and similarly for* $j_{x'}$ *and* $j_{y'}$*,*
- *the vertical composite of 2-morphisms:*

$$\begin{array}{ccccc} & & y & & \\ & \overset{i}{\nearrow} & & \overset{o}{\nwarrow} & \\ L(a) & & f\downarrow & & L(b) \\ & \underset{i'}{\searrow} & & \underset{o'}{\swarrow} & \\ & & y' & & \end{array}$$

$$\begin{array}{ccccc} & & y' & & \\ & \overset{i'}{\nearrow} & & \overset{o'}{\nwarrow} & \\ L(a) & & f'\downarrow & & L(b) \\ & \underset{i''}{\searrow} & & \underset{o''}{\swarrow} & \\ & & y'' & & \end{array}$$

  is given by

$$\begin{array}{ccccc} & & y & & \\ & \overset{i}{\nearrow} & & \overset{o}{\nwarrow} & \\ L(a) & & f'f\downarrow & & L(b) \\ & \underset{i''}{\searrow} & & \underset{o''}{\swarrow} & \\ & & y'' & & \end{array}$$

- *the associator and unitors are defined using the universal property of pushouts.*

Proof. As noted for example by Shulman [34], any double category $\mathbb{X}$ gives rise to a bicategory $\mathbf{X}$ with

- objects given by objects of $\mathbb{X}$,
- morphisms given by horizontal 1-cells of $\mathbb{X}$,
- 2-morphisms given by **globular** 2-morphisms of $\mathbb{X}$, meaning 2-morphisms whose source and target vertical 1-morphisms are identities,
- composition of morphisms given by horizontal composition of horizontal 1-cells in $\mathbb{X}$,
- vertical and horizontal composition of 2-morphisms given by vertical and horizontal composition of 2-morphisms in $\mathbb{X}$.

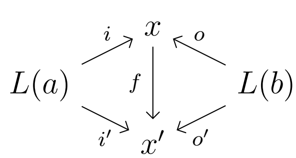
Applying this to ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ we obtain ${}_L\mathbf{Csp}(\mathsf{X})$. ■

2.5. Corollary. *Let $L\colon \mathsf{A} \to \mathsf{X}$ be a functor where X is a category with pushouts. Then there is a category ${}_L\mathsf{Csp}(\mathsf{X})$ in which:*

- *an object is an object of A,*
- *a morphism from a to b is an isomorphism class of diagrams in X of this form:*

$$L(a) \xrightarrow{i} x \xleftarrow{o} L(b)$$

where $L(a) \overset{i}{\to} x \overset{o}{\leftarrow} L(b)$ and $L(a) \overset{i'}{\to} x' \overset{o'}{\leftarrow} L(b)$ are isomorphic iff there is an isomorphism $f\colon x \to x'$ making this diagram commute:



- *composition of morphisms is done using pushouts in X.*

Proof. By decategorifying a bicategory $\mathbf{B}$ we obtain a category B with the same objects, whose morphisms are isomorphism classes of 1-morphisms in $\mathbf{B}$. Applying this to ${}_L\mathbf{Csp}(\mathsf{X})$ we obtain ${}_L\mathsf{Csp}(\mathsf{X})$. Note that this category is independent of our choice of pushouts in X, since pushouts are unique up to isomorphism. ■

3. Symmetric monoidal double categories of structured cospans

Now we give simple conditions under which the double category ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$, the bicategory ${}_L\mathbf{Csp}(\mathsf{X})$ and the category ${}_L\mathsf{Csp}(\mathsf{X})$ all become symmetric monoidal. We have seen that if X has pushouts and $L\colon \mathsf{A} \to \mathsf{X}$ is any functor then there is a double category of structured cospans ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$. In Theorem 3.9 we show that ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ becomes symmetric monoidal when A and X have finite colimits and L preserves these. The monoidal structure describes our ability to take two structured cospans:

$$L(a) \xrightarrow{i} x \xleftarrow{o} L(b) \qquad\qquad L(a') \xrightarrow{i'} x' \xleftarrow{o'} L(b')$$

and form a new one via coproduct:

$$L(a+a') \xrightarrow{\cong} L(a)+L(a') \xrightarrow{i+i'} x+x' \xleftarrow{o+o'} L(b)+L(b') \xleftarrow{\cong} L(b+b')$$

One can check that this operation makes ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ into a monoidal double category simply by verifying that a rather large number of diagrams commute. This is the approach taken in [10]. There is nothing tricky about it. Indeed, requiring that L preserve finite colimits is overkill: it suffices for L to preserve finite coproducts. Thus, for most readers the best thing to do at this point would be to review the definition of 'symmetric monoidal double category' in Appendix A, look at the statement of Theorem 3.9, and move on to the next section.

However, it is a bit irksome to check that all the necessary diagrams commute, especially since one gets the feeling that there must be a simple underlying reason. So, we decided to give a more conceptual proof. While perhaps harder to digest, this gives us more—at least when F preserves finite colimits. In this case we can do much more than take binary coproducts of structured cospans: we can take finite colimits of them! This means that we can glue together structured cospans in more interesting ways than merely composing them end to end or setting them side by side. Thus, we prove Theorem 3.9 as a consequence of a stronger result, Theorem 3.7, which captures the full range of ways we can take finite colimits of structured cospans.

The key concept we need is that of a 'weak category' or 'pseudocategory' [26] in a 2-category. This is a slight generalization of the concept of double category.

3.1. Definition. *Given a 2-category* $\mathbf{C}$, *a* **weak category** $\mathbb{D}$ *in* $\mathbf{C}$ *consists of:*

- *an* **object of objects** $\mathbb{D}_0 \in \mathbf{C}$ *and an* **object of arrows** $\mathbb{D}_1 \in \mathbf{C}$,

- **source** *and* **target** *morphisms*

$$S, T\colon \mathbb{D}_1 \to \mathbb{D}_0,$$

- *an* **identity-assigning** *morphism*

$$U\colon \mathbb{D}_0 \to \mathbb{D}_1,$$

- *and a* **composition** *morphism*

$$\odot\colon \mathbb{D}_1 \times_{\mathbb{D}_0} \mathbb{D}_1 \to \mathbb{D}_1$$

  where the pullback is taken over $\mathbb{D}_1 \xrightarrow{T} \mathbb{D}_0 \xleftarrow{S} \mathbb{D}_1$,

such that:

- *the source and target morphisms behave as expected for identities:*

$$S \circ U = 1_{\mathbb{D}_0} = T \circ U$$

  and for composition:

$$S \circ \odot = S \circ p_1, \qquad T \circ \odot = T \circ p_2$$

  where $p_1, p_2\colon \mathbb{D}_1 \times_{\mathbb{D}_0} \mathbb{D}_1 \to \mathbb{D}_1$ *are projections to the two factors;*

- *composition is associative up to a 2-isomorphism called the* **associator***:*

$$\begin{array}{ccc} \mathbb{D}_1 \times_{\mathbb{D}_0} \mathbb{D}_1 \times_{\mathbb{D}_0} \mathbb{D}_1 & \xrightarrow{1 \times \odot} & \mathbb{D}_1 \times_{\mathbb{D}_0} \mathbb{D}_1 \\ {\scriptstyle \odot \times 1}\downarrow & \alpha \nearrow & \downarrow{\scriptstyle \odot} \\ \mathbb{D}_1 \times_{\mathbb{D}_0} \mathbb{D}_1 & \xrightarrow[\odot]{} & \mathbb{D}_1 \end{array}$$

- *composition obeys the left and right unit laws up to 2-isomorphisms called the* **left** *and* **right unitors***:*

$$\begin{array}{ccccc} \mathbb{D}_0 \times_{\mathbb{D}_0} \mathbb{D}_1 & \xrightarrow{U \times_{\mathbb{D}_0} 1} & \mathbb{D}_1 \times_{\mathbb{D}_0} \mathbb{D}_1 & \xleftarrow{1 \times_{\mathbb{D}_0} U} & \mathbb{D}_1 \times_{\mathbb{D}_0} \mathbb{D}_0 \\ & \lambda \swarrow & \downarrow{\scriptstyle \odot} & \rho \searrow & \\ & & \mathbb{D}_1 & & \end{array}$$
$$\text{(with } p_2\colon \mathbb{D}_0 \times_{\mathbb{D}_0} \mathbb{D}_1 \to \mathbb{D}_1 \text{ and } p_1\colon \mathbb{D}_1 \times_{\mathbb{D}_0} \mathbb{D}_0 \to \mathbb{D}_1\text{)}$$

- α, λ *and* ρ *obey the pentagon identity and triangle identity.*

In this definition we assume that the necessary pullbacks exist; if **C** has pullbacks this is automatic.

Consulting Appendix A, the reader can check that a weak category in **Cat** is the same as a double category. We need weak categories in the following 2-categories as well:

3.2. Definition. *Let* **Rex** *be the 2-category with:*

- *categories with chosen finite colimits as objects,*
- *right exact functors as morphisms,*
- *natural transformations as 2-morphisms.*

3.3. Definition. *Let* **SymMonCat** *be the 2-category with:*

- *symmetric monoidal categories as objects,*
- *(strong) symmetric monoidal functors as morphisms,*
- *monoidal natural transformations as 2-morphisms.*

The word 'rex' is an abbreviation of 'right exact', which is another term for 'preserving finite colimits'. Note that a right exact functor need not preserve a given *choice* of finite colimits. Thus, our 2-category **Rex** is 2-equivalent to one where no choices of finite colimits were made. One reason for making these choices is that they give us an unambiguously defined 2-functor

$$\Phi\colon \mathbf{Rex} \to \mathbf{SymMonCat}$$

as follows. Given an object $\mathsf{C} \in \mathbf{Rex}$, $\Phi(\mathsf{C})$ is the symmetric monoidal category $(\mathsf{C}, +, 0)$ where $+$ is the chosen binary coproduct and 0 is the chosen initial object. Each right exact functor $F\colon \mathsf{C} \to \mathsf{C}'$ between categories $\mathsf{C}, \mathsf{C}' \in \mathbf{Rex}$ then becomes symmetric monoidal in a canonical way, and each natural transformation between right exact functors becomes monoidal.

Our plan now proceeds as follows. First, in Theorem 3.7, we show that when $L\colon \mathsf{A} \to \mathsf{X}$ is a morphism in **Rex**, the double category ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ is not merely a weak category in **Cat**, but actually a weak category in **Rex**. In Theorem 3.8 we use the 2-functor Φ to convert ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ into a weak category in **SymMonCat**.

Finally, from this weak category in **SymMonCat**, we wish to get a symmetric monoidal double category. Here we need the concept of a 'symmetric pseudomonoid' [36]. To understand the following definitions the reader should keep in mind the example where **B** is **Cat** made into a symmetric monoidal bicategory using cartesian products. Then a pseudomonoid in **B** is a monoidal category, a braided pseudomonoid is a braided monoidal category, and a symmetric pseudomonoid is a symmetric monoidal category.

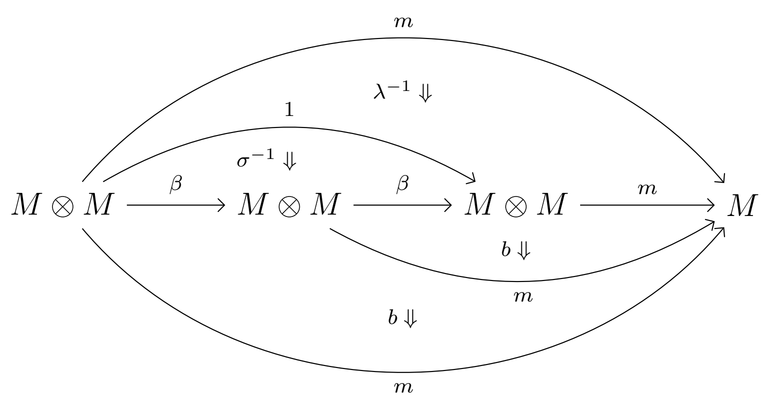
3.4. Definition. *A* **pseudomonoid** *in a monoidal bicategory* **B** *is an object* $M \in \mathbf{B}$ *equipped with 1-morphisms called the* **multiplication** $m\colon M \otimes M \to M$ *and* **unit** $i\colon I \to M$ *that obey associativity and the left and right unit laws up to 2-isomorphisms called the* **associator** *and left and right* **unitors**, *that in turn obey the pentagon identity and triangle identity.*

3.5. Definition. *A pseudomonoid M in a braided monoidal bicategory* **B** *is* **braided** *if it is equipped with a 2-isomorphism*

$$b\colon m \circ \beta \overset{\sim}{\Rightarrow} m$$

where $\beta\colon M \otimes M \to M \otimes M$ is the braiding in **B**, *and b obeys the hexagon identities [24].*

3.6. Definition. *A braided pseudomonoid M in a symmetric monoidal bicategory* **B** *is called* **symmetric** *if*



is the identity 2-morphism from m to m. Here λ is the left unitor for composition of 1-morphisms in **B** *and $\sigma\colon \beta^2 \Rightarrow 1$ is the syllepsis for* **B**.

Readers unfamiliar with these concepts may be relieved to learn that the syllepsis in **Cat** is the identity; in a general symmetric monoidal bicategory the square of the braiding may be only *isomorphic* to the identity, and this isomorphism is called the syllepsis [11].

The plan continues as follows. Having shown that ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ is a weak category in **SymMonCat**, we notice that such a thing is

a weak category in [symmetric pseudomonoids in **Cat**].

By 'commutativity of internalization' we could hope that this is the same as

a symmetric pseudomonoid in [weak categories in **Cat**].

But the latter is precisely a symmetric double category. So, ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ should be a symmetric monoidal double category.

Unfortunately, this hope is a bit naive. Shulman explains the reason [34]:

> The general yoga of internalization says that an X internal to Ys internal to Zs is equivalent to a Y internal to Xs internal to Zs, but this is only strictly true when the internalizations are all strict. We have defined a symmetric monoidal double category to be a (pseudo) symmetric monoid internal to (pseudo) categories internal to categories, but one could also consider a (pseudo) category

internal to (pseudo) symmetric monoids internal to categories, i.e. a pseudo internal category in the 2-category **SymMonCat** of symmetric monoidal categories and strong symmetric monoidal functors. This would give *almost* the same definition, except that S and T would only be strong monoidal (preserving $\otimes$ up to isomorphism) rather than strict monoidal.

Luckily, the difference between the two definitions is quite small, so with a bit of care we can arrange for ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ to be a symmetric monoidal double category.

We begin as follows:

3.7. Theorem. *Given a morphism $L\colon \mathsf{A} \to \mathsf{X}$ in* **Rex**, *the double category ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ is a weak category object in* **Rex**.

Proof. In the double category ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$,

- the category of objects ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_0$ is A, while
- the category of arrows ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1$ has structured cospans

$$L(a) \xrightarrow{i} x \xleftarrow{o} L(b)$$

as objects and commutative diagrams of this form:

$$\begin{array}{ccccc} L(a) & \xrightarrow{i} & x & \xleftarrow{o} & L(b) \\ {\scriptstyle L(\alpha)}\downarrow & & {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle L(\beta)} \\ L(a') & \xrightarrow[i']{} & x' & \xleftarrow[o']{} & L(b') \end{array}$$

as morphisms.

We need to choose finite colimits for ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_0$ and ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1$ and show the source and target functors

$$S, T\colon {}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1 \to {}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_0,$$

the identity-assigning functor

$$U\colon {}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_0 \to {}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1,$$

and the composition functor

$$\circ\colon {}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1 \times_{{}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_0} {}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1 \to {}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1$$

are right exact. We also need to check that all the pullbacks in **Cat** used to define the double category ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ are also pullbacks in **Rex**.

The category of objects ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_0 = \mathsf{A}$ has chosen finite colimits by hypothesis. The category of arrows ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1$ has finite colimits because L preserves finite colimits and these colimits are computed pointwise in X. We give ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1$ *chosen* finite colimits using the chosen finite colimits in A and X. The functors S, T and U are right exact, again because colimits in ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1$ are computed pointwise in X. The functor $\circ$ sends a composable pair of structured cospans to their composite, which is defined using a pushout. This functor is right exact as a consequence of colimits commuting with other colimits.

We also need to check that the category

$$\mathsf{Z} = {}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1 \times_{{}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_0} {}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1,$$

defined as a pullback in **Cat**, is also a pullback in **Rex**. Note that objects of Z are composable pairs of structured cospans:

$$L(a) \to x \leftarrow L(b) \to y \leftarrow L(c),$$

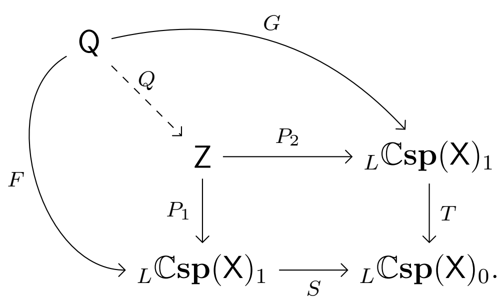
while morphisms are commuting diagrams of the form

$$\begin{array}{ccccccccc}
L(a) & \longrightarrow & x & \longleftarrow & L(b) & \longrightarrow & y & \longleftarrow & L(c) \\
\downarrow{\scriptstyle L(\alpha)} & & \downarrow{\scriptstyle f} & & \downarrow{\scriptstyle L(\beta)} & & \downarrow{\scriptstyle g} & & \downarrow{\scriptstyle L(\gamma)} \\
L(a') & \longrightarrow & x' & \longleftarrow & L(b') & \longrightarrow & y' & \longleftarrow & L(c').
\end{array}$$

Because A and X have finite colimits and L preserves them, Z has finite colimits computed pointwise. Consider the pullback square in **Cat** defining Z:

$$\begin{array}{ccc}
\mathsf{Z} & \xrightarrow{P_2} & {}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1 \\
\downarrow{\scriptstyle P_1} & & \downarrow{\scriptstyle T} \\
{}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1 & \xrightarrow[S]{} & {}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_0
\end{array}$$

where P_1 projects to the first structured cospan of an object in Z, and P_2 projects to the second. All the arrows here are right exact because colimits are computed pointwise. Suppose next that F and G below are right exact:



Then there exists a unique functor Q making the diagram commute. This functor Q is right exact because its composites with P_1 and P_2 are: since colimits in a diagram category are computed pointwise, a cocone in Z is a colimit of $F\colon \mathsf{D} \to \mathsf{Z}$ if and only if the 'pieces' obtained by applying P_1 and P_2 to this cocone are colimits of $P_1 \circ F$ and $P_2 \circ F$, respectively.

The other pullbacks used in defining the double category ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$, such as the pullback ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1 \times_{{}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_0} {}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1 \times_{{}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_0} {}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1$ used in defining the associator, are also pullbacks in **Rex** for the same sort of reason. ■

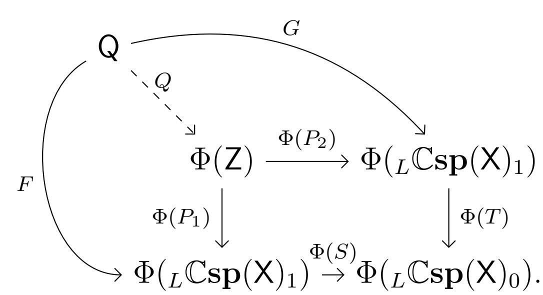
Next we make ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ into a weak category in **SymMonCat**. We do this by applying the 2-functor $\Phi\colon \mathbf{Rex} \to \mathbf{SymMonCat}$.

3.8. THEOREM. *Given a morphism $L\colon \mathsf{A} \to \mathsf{X}$ in* **Rex**, *the functor $\Phi\colon \mathbf{Rex} \to$* **SymMonCat** *maps the weak category ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ in* **Rex** *to a weak category in* **SymMonCat**.

PROOF. We need to show that the various pullbacks in **Rex** used to make ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ into a weak category in **Rex** are mapped by Φ to pullbacks in **SymMonCat**. We do this only for the pullback $\mathsf{Z} = {}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1 \times_{{}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_0} {}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1$, since the others are similar. To show that $\Phi(\mathsf{Z})$ is the pullback of the following square in **SymMonCat**:

$$\begin{array}{ccc} \Phi(\mathsf{Z}) & \xrightarrow{\Phi(P_2)} & \Phi({}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1) \\ {\scriptstyle \Phi(P_1)}\downarrow & & \downarrow{\scriptstyle \Phi(T)} \\ \Phi({}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1) & \xrightarrow{\Phi(S)} & \Phi({}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_0) \end{array}$$

we need to show that for any symmetric monoidal category Q and symmetric monoidal functors $F, G\colon \mathsf{Q} \to \Phi({}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1)$ with $\Phi(S)F = \Phi(T)G$, there exists a unique symmetric monoidal functor Q making this diagram commute:



By Theorem 3.7 there exists a unique right exact functor Q making the underlying diagram of functors commute. We now show that this Q can be made symmetric monoidal in such a way that the diagram commutes in **SymMonCat**.

First, let 0_{Q} be the monoidal unit of Q. Since $F\colon \mathsf{Q} \to \Phi({}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1)$ is symmetric monoidal, we have an isomorphism between monoidal units:

$$F_0\colon 0_{\Phi({}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1)} \xrightarrow{\sim} F(0_{\mathsf{Q}})$$

where $0_{\Phi({}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1)}$ is initial in $\Phi({}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1)$. Similarly we have an isomorphism

$$G_0 \colon 0_{\Phi({}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1)} \xrightarrow{\sim} G(0_{\mathsf{Q}}).$$

It follows that $Q(0_{\mathsf{Q}})$ is a pair of composable initial cospans in X so there is a unique isomorphism

$$Q_0 \colon 0_{\mathsf{Z}} \xrightarrow{\sim} Q(0_{\mathsf{Q}}).$$

Next, given two objects a_1 and a_2 in Q, we have a natural isomorphism

$$F_{a_1,a_2} \colon F(a_1) + F(a_2) \xrightarrow{\sim} F(a_1 \otimes a_2)$$

as F is symmetric monoidal, and similarly for G. We know that as objects, $F(a_1)$ and $F(a_2)$ are simply cospans in X with $F(a_1) + F(a_2)$ their chosen coproduct. We also know that $Q(a_1)$ is a pair of composable cospans $(F(a_1), G(a_1))$ and likewise $Q(a_2)$ is a pair of composable cospans $(F(a_2), G(a_2))$. This results in a natural isomorphism

$$Q_{a_1,a_2} \colon Q(a_1) + Q(a_2) \to Q(a_1 \otimes a_2)$$

given by the composite

$$(F(a_1), G(a_1)) + (F(a_2), G(a_2)) \xrightarrow{\sim} (F(a_1) + F(a_2), G(a_1) + G(a_2))$$

$$\xrightarrow{\sim} (F(a_1 \otimes a_2), G(a_1 \otimes a_2)).$$

One can check that this family of natural isomorphisms Q_{a_1,a_2} together with the natural isomorphism Q_0 give Q the structure of a symmetric monoidal functor, and that the above diagram then commutes in **SymMonCat**. It follows that $\Phi(\mathsf{Z})$ is a pullback square in **SymMonCat**, as was to be shown. ■

In Theorem 3.8 we made ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ into a weak category in **SymMonCat**. Now we make ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ into a symmetric monoidal double category.

3.9. THEOREM. *Suppose A and X have finite colimits and $L \colon \mathsf{A} \to \mathsf{X}$ preserves them. Choose finite colimits in A and X. Then the double category ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ becomes symmetric monoidal where:*

- *the tensor product of objects a_1, a_2 is their chosen coproduct $a_1 + a_2$ in A,*
- *the unit object is the chosen initial object 0_{A} in A,*
- *the tensor product of two vertical 1-morphisms is given by*

$$\begin{array}{c} a_1 \\ {\scriptstyle f_1}\big\downarrow \\ b_1 \end{array} \otimes \begin{array}{c} a_2 \\ {\scriptstyle f_2}\big\downarrow \\ b_2 \end{array} = \begin{array}{c} a_1 + a_2 \\ {\scriptstyle f_1 + f_2}\big\downarrow \\ b_1 + b_2 \end{array}$$

- *the tensor product of horizontal 1-cells is given by*

$$\begin{array}{ccc} & x & \\ {}^{i}\nearrow & & \nwarrow^{o} \\ L(a) & & L(b) \end{array} \otimes \begin{array}{ccc} & x' & \\ {}^{i'}\nearrow & & \nwarrow^{o'} \\ L(a') & & L(b') \end{array} = \begin{array}{ccc} & x+x' & \\ {}^{i+i'}\nearrow & & \nwarrow^{o+o'} \\ L(a+a') & & L(b+b') \end{array}$$

  where $i+i'$ and $o+o'$ are defined using the fact that L preserves coproducts,

- *the unit horizontal 1-cell is given by*

$$L(0_{\mathsf{A}}) \xrightarrow{i} 0_{\mathsf{X}} \xleftarrow{o} L(0_{\mathsf{A}})$$

  where 0_{X} is the chosen initial object in X,

- *the tensor product of two 2-morphisms is given by:*

$$\begin{array}{ccccc} L(a_1) & \xrightarrow{i_1} & x_1 & \xleftarrow{o_1} & L(b_1) \\ {\scriptstyle L(f)}\downarrow & & \downarrow{\scriptstyle \alpha} & & \downarrow{\scriptstyle L(g)} \\ L(a_2) & \xrightarrow[i_2]{} & x_2 & \xleftarrow[o_2]{} & L(b_2) \end{array} \otimes \begin{array}{ccccc} L(a'_1) & \xrightarrow{i'_1} & x'_1 & \xleftarrow{o'_1} & L(b'_1) \\ {\scriptstyle L(f')}\downarrow & & \downarrow{\scriptstyle \alpha'} & & \downarrow{\scriptstyle L(g')} \\ L(a'_2) & \xrightarrow[i'_2]{} & x'_2 & \xleftarrow[o'_2]{} & L(b'_2) \end{array}$$

$$= \begin{array}{ccccc} L(a_1+a'_1) & \xrightarrow{i_1+i'_1} & x_1+x'_1 & \xleftarrow{o_1+o'_1} & L(b_1+b'_1) \\ {\scriptstyle L(f+f')}\downarrow & & \downarrow{\scriptstyle \alpha+\alpha'} & & \downarrow{\scriptstyle L(g+g')} \\ L(a_2+a'_2) & \xrightarrow[i_2+i'_2]{} & x_2+x'_2 & \xleftarrow[o_2+o'_2]{} & L(b_2+b'_2), \end{array}$$

and the associators, left and right unitors, and braidings are defined using the universal properties of binary coproducts and unit objects.

Proof. By Theorem 3.8, ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ is a weak category object in **SymMonCat**, so both its category of objects and category of arrows are symmetric monoidal. To show that it is a symmetric monoidal double category, we need only show that the source and target functors

$$S, T \colon {}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1 \to {}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_0$$

are *strict* symmetric monoidal [34, Remark 2.12]. This follows because S and T simply pick out the input and output of a structured cospan, and we are using the same chosen binary coproducts and initial object in A in defining the monoidal structures on both ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_0$ and ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})_1$. ■

In fact, to make ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ into a symmetric monoidal double category it suffices for A to have finite coproducts, X to have finite colimits, and L to preserve finite coproducts [10, Theorem 3.2.3]. But in the examples we have studied, A and X have finite colimits, and L, being a left adjoint, preserves all of these.

Next we take the symmetric monoidal double category ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ and water it down, obtaining first a symmetric monoidal bicategory and then a symmetric monoidal category. The definition of symmetric monoidal bicategory is nicely presented by Stay [35], who recalls how this definition was gradually discovered by a series of authors. Shulman [34] provides a convenient way to construct symmetric monoidal bicategories from symmetric monoidal double categories. He defines a double category $\mathbb{D}$ to be isofibrant if every vertical 1-isomorphism has a 'companion' and a 'conjoint' [20], and proves that if $\mathbb{D}$ is symmetric monoidal and isofibrant, then $\mathbf{D}$ becomes symmetric monoidal in a canonical way.

A **companion** of a vertical 1-morphism $f\colon a \to b$ is a horizontal 1-cell $\hat{f}\colon a \to b$ equipped with 2-morphisms

$$\begin{array}{ccc} a & \xrightarrow{\hat{f}} & b \\ {\scriptstyle f}\downarrow & \alpha\Downarrow & \downarrow{\scriptstyle 1} \\ b & \xrightarrow[U_b]{} & b \end{array} \qquad \text{and} \qquad \begin{array}{ccc} a & \xrightarrow{U_a} & a \\ {\scriptstyle 1}\downarrow & \beta\Downarrow & \downarrow{\scriptstyle f} \\ a & \xrightarrow[\hat{f}]{} & b \end{array}$$

that obey these equations:

$$\begin{array}{ccc} a & \xrightarrow{U_a} & a \\ {\scriptstyle 1}\downarrow & \beta\Downarrow & \downarrow{\scriptstyle f} \\ a & \xrightarrow{\hat{f}} & b \\ {\scriptstyle f}\downarrow & \alpha\Downarrow & \downarrow{\scriptstyle 1} \\ b & \xrightarrow[U_b]{} & b \end{array} = \begin{array}{ccc} a & \xrightarrow{U_a} & a \\ {\scriptstyle f}\downarrow & \Downarrow U_f & \downarrow{\scriptstyle f} \\ b & \xrightarrow[U_b]{} & b \end{array} \quad \text{and} \quad \begin{array}{ccccc} a & \xrightarrow{U_a} & a & \xrightarrow{\hat{f}} & b \\ {\scriptstyle 1}\downarrow & \beta\Downarrow & {\scriptstyle f}\downarrow & \alpha\Downarrow & \downarrow{\scriptstyle 1} \\ a & \xrightarrow{\hat{f}} & b & \xrightarrow{U_b} & b \\ {\scriptstyle 1}\downarrow & & \lambda_{\hat{f}}\Downarrow & & \downarrow{\scriptstyle 1} \\ a & & \xrightarrow[\hat{f}]{} & & b \end{array} = \begin{array}{ccccc} a & \xrightarrow{U_a} & a & \xrightarrow{\hat{f}} & b \\ {\scriptstyle 1}\downarrow & & \rho_f\Downarrow & & \downarrow{\scriptstyle 1} \\ a & & \xrightarrow[\hat{f}]{} & & b \end{array} \tag{1}$$

A **conjoint** of f is a horizontal 1-cell $\check{f}\colon b \to a$ that is a companion of f in the 'horizontal opposite' of the double category in question. Since ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ is its own horizontal opposite, we only need to check the existence of companions.

3.10. Corollary. *If A and X have finite colimits, $L\colon \mathsf{A} \to \mathsf{X}$ preserves them, and we choose finite colimits in both A and X, then the bicategory ${}_L\mathbf{Csp}(\mathsf{X})$ of Corollary 2.4 becomes symmetric monoidal as follows:*

- *the tensor product of objects a_1 and a_2 is their chosen coproduct $a_1 + a_2$ in A,*
- *the unit for the tensor product is the chosen initial object 0_A in A,*

- *the tensor product of 1-morphisms is given by*

$$\begin{array}{ccc} & x & \\ & {\scriptstyle i}\nearrow \quad \nwarrow{\scriptstyle o} & \\ L(a) & & L(b) \end{array} \otimes \begin{array}{ccc} & x' & \\ & {\scriptstyle i'}\nearrow \quad \nwarrow{\scriptstyle o'} & \\ L(a') & & L(b') \end{array} = \begin{array}{ccc} & x+x' & \\ & {\scriptstyle i+i'}\nearrow \quad \nwarrow{\scriptstyle o+o'} & \\ L(a+a') & & L(b+b') \end{array}$$

- *the tensor product of 2-morphisms is given by*

$$\begin{array}{ccc} & x_1 & \\ {\scriptstyle i_1}\nearrow & & \nwarrow{\scriptstyle o_1} \\ L(a_1) & \quad {\scriptstyle \alpha_1}\downarrow & L(a_1') \\ {\scriptstyle i_1'}\searrow & & \swarrow{\scriptstyle o_1'} \\ & x_1' & \end{array} \otimes \begin{array}{ccc} & x_2 & \\ {\scriptstyle i_2}\nearrow & & \nwarrow{\scriptstyle o_2} \\ L(a_2) & \quad {\scriptstyle \alpha_2}\downarrow & L(a_2') \\ {\scriptstyle i_2'}\searrow & & \swarrow{\scriptstyle o_2'} \\ & x_2' & \end{array} = \begin{array}{ccc} & x_1+x_2 & \\ {\scriptstyle i_1+i_2}\nearrow & & \nwarrow{\scriptstyle o_1+o_2} \\ L(a_1+a_2) & \quad {\scriptstyle \alpha_1+\alpha_2}\downarrow & L(a_1'+a_2') \\ {\scriptstyle i_1'+i_2'}\searrow & & \swarrow{\scriptstyle o_1'+o_2'} \\ & x_1'+x_2' & \end{array}$$

- *the associators, unitors, symmetries, and other structures of a symmetric monoidal bicategory are constructed using the universal properties of binary coproducts and initial objects.*

PROOF. A vertical 1-isomorphism in ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ is a isomorphism $f\colon a \to b$ in A. We take its companion $\hat{f}$ to be the structured cospan

$$L(a) \xrightarrow{L(f)} L(b) \xleftarrow{1} L(b).$$

The unit horizontal 1-cells U_a and U_b are given respectively by

$$L(a) \xrightarrow{1} L(a) \xleftarrow{1} L(a) \qquad \text{and} \qquad L(b) \xrightarrow{1} L(b) \xleftarrow{1} L(b)$$

and the accompanying 2-morphisms α and β are given by

$$\begin{array}{ccccc} L(a) & \xrightarrow{L(f)} & L(b) & \xleftarrow{1} & L(b) \\ {\scriptstyle L(f)}\downarrow & & {\scriptstyle 1}\downarrow & & \downarrow{\scriptstyle 1} \\ L(b) & \xrightarrow[1]{} & L(b) & \xleftarrow[1]{} & L(b) \end{array} \qquad \text{and} \qquad \begin{array}{ccccc} L(a) & \xrightarrow{1} & L(a) & \xleftarrow{1} & L(a) \\ {\scriptstyle 1}\downarrow & & {\scriptstyle L(f)}\downarrow & & \downarrow{\scriptstyle L(f)} \\ L(a) & \xrightarrow[L(f)]{} & L(b) & \xleftarrow[1]{} & L(b) \end{array}$$

respectively. An easy calculation verifies Equation (1). ∎

3.11. Corollary. *If* A *and* X *have finite colimits,* $L\colon \mathsf{A} \to \mathsf{X}$ *preserves them, and we choose binary coproducts and an initial object in* A*, then the category* ${}_L\mathsf{Csp}(\mathsf{X})$ *of Corollary 2.5 becomes symmetric monoidal as follows:*

- *the tensor product of objects* a_1 *and* a_2 *is their chosen coproduct* $a_1 + a_2$ *in* A*,*
- *the unit for the tensor product is the chosen initial object* 0_{A} *in* A*,*
- *the tensor product of morphisms is given by*

$$\begin{array}{ccc} & x & \\ {}^{i}\nearrow & & \nwarrow^{o} \\ L(a) & & L(b) \end{array} \otimes \begin{array}{ccc} & x' & \\ {}^{i'}\nearrow & & \nwarrow^{o'} \\ L(a') & & L(b') \end{array} = \begin{array}{ccc} & x+x' & \\ {}^{i+i'}\nearrow & & \nwarrow^{o+o'} \\ L(a+a') & & L(b+b') \end{array}$$

  where in each case the cospan actually denotes an isomorphism class of cospans,
- *the associator, left and right unitors, and symmetry are constructed using the universal properties of binary coproducts and initial objects.*

Proof. It can be checked by inspecting the definitions that any symmetric monoidal bicategory $\mathbf{B}$ gives rise to a symmetric monoidal category B where:

- the objects of B are those of $\mathbf{B}$,
- the morphisms of B are isomorphism classes of morphisms of $\mathbf{B}$,
- the unit object and the tensor product of objects are those of $\mathbf{B}$,
- the tensor product of morphisms, the associator, the left and right unitor, and the symmetry of B arise from those of $\mathbf{B}$ by taking isomorphism classes.

Applying this 'decategorification' construction to the symmetric monoidal bicategory ${}_L\mathbf{Csp}(\mathsf{X})$ gives the symmetric monoidal category ${}_L\mathsf{Csp}(\mathsf{X})$. ∎

The symmetric monoidal category ${}_L\mathsf{Csp}(\mathsf{X})$ is determined up to equality by the choice of binary coproducts and initial object in A, but different choices of this data give isomorphic symmetric monoidal categories.

Readers interested in hypergraph categories may be pleased to learn that structured cospan categories tend to be of this type. A 'hypergraph category' is a symmetric monoidal category where each object has the structure of a special commutative Frobenius monoid in a way that is compatible with tensor products but not necessarily preserved by morphisms [13]. Such categories are ubiquitous in network theory, where Frobenius structure allows us to split, join, start and terminate strings in string diagrams [14]. While the definition of hypergraph category may seem awkward at first, Fong and Spivak have clarified this concept using operads [17].

3.12. Theorem. *If A and X have finite colimits, $L\colon \mathsf{A} \to \mathsf{X}$ preserves them, and we choose binary coproducts and an initial object in A, then the symmetric monoidal category ${}_L\mathsf{Csp}(\mathsf{X})$ is a hypergraph category where each object $a \in \mathsf{A}$ is a special commutative Frobenius monoid as follows:*

- *The multiplication is given by the structured cospan*

$$L(a+a) \xrightarrow{L(\nabla)} L(a) \xleftarrow{1} L(a).$$

where $\nabla\colon a+a \to a$ is the fold map.

- *The unit is given by*

$$L(0) \xrightarrow{L(!)} L(a) \xleftarrow{1} L(a).$$

where $!\colon 0 \to a$ is the unique morphism.

- *The comultiplication is given by*

$$L(a) \xrightarrow{1} L(a) \xleftarrow{L(\nabla)} L(a+a).$$

- *The counit is given by*

$$L(a) \xrightarrow{1} L(a) \xleftarrow{L(!)} L(0).$$

Proof. Whenever $F\colon \mathsf{C} \to \mathsf{D}$ is a symmetric monoidal functor bijective on objects and C is a hypergraph category, there is a unique way to make D into a hypergraph category such that F is a hypergraph functor. To see this, first note that F equips each object of D with the structure of a special commutative Frobenius monoid, coming from its structure in C. These Frobenius structures are compatible with tensor product because they were in C and F is symmetric monoidal. Thus, D becomes a hypergraph category. By construction $F\colon \mathsf{C} \to \mathsf{D}$ preserves the Frobenius structures on objects, so F is a hypergraph functor. Moreover, the Frobenius structures on objects of D are uniquely determined by this requirement.

Let $\mathsf{Csp}(\mathsf{A})$ be the symmetric monoidal category whose morphisms are isomorphism classes of cospans in A. Since L preserves finite colimits, there is a symmetric monoidal functor $F\colon \mathsf{Csp}(\mathsf{A}) \to {}_L\mathsf{Csp}(\mathsf{X})$ given as follows:

$$\begin{array}{ccccccc} & c & & & & L(c) & \\ {}^{i}\nearrow & & \nwarrow^{o} & \mapsto & {}^{L(i)}\nearrow & & \nwarrow^{L(o)} \\ a & & b & & L(a) & & L(b). \end{array}$$

This is bijective on objects, and $\mathsf{Csp}(\mathsf{A})$ is a hypergraph category [13], so ${}_L\mathsf{Csp}(\mathsf{X})$ has a unique hypergraph category structure making F into a hypergraph functor. This is given as in the statement of the theorem. ∎

4. Maps between structured cospan double categories

In this section we show how to construct maps between structured cospan categories, or bicategories, or double categories. As before, it is best to start with double categories and work our way down. A map between double categories is called a 'double functor', and these are defined in Definition A.3. Suppose that we have structured cospan double categories coming from functors $L\colon \mathsf{A} \to \mathsf{X}$ and $L'\colon \mathsf{A}' \to X'$, where X and X' have chosen pushouts. Then we get a double functor between these double categories from a diagram of this form:

$$\begin{array}{ccc} \mathsf{A} & \xrightarrow{L} & \mathsf{X} \\ {\scriptstyle F_0}\downarrow & \alpha \nearrow & \downarrow{\scriptstyle F_1} \\ \mathsf{A}' & \xrightarrow[L']{} & \mathsf{X}' \end{array}$$

where F_0 is a functor, F_1 is a functor preserving pushouts, and α is a natural isomorphism. We prove this in Theorem 4.2. Furthermore, if all four categories involved have finite colimits and all four functors preserve these, then this double functor is symmetric monoidal—a concept defined in Definition A.7. We prove this in Theorem 4.3.

4.1. Definition. *Given a 2-category* $\mathbf{C}$ *and two weak categories* $\mathbb{D}$ *and* $\mathbb{D}'$ *in* $\mathbf{C}$*, a* **weak functor** $\mathbb{F}\colon \mathbb{D} \to \mathbb{D}'$ *in* $\mathbf{C}$ *consists of:*

- *a morphism of objects* $\mathbb{F}_0\colon \mathbb{D}_0 \to \mathbb{D}'_0$,
- *a morphism of arrows* $\mathbb{F}_1\colon \mathbb{D}_1 \to \mathbb{D}'_1$,

such that:

- $\mathbb{F}$ *preserves the source and target morphisms:* $S' \circ \mathbb{F}_1 = \mathbb{F}_0 \circ S$ *and* $T' \circ \mathbb{F}_1 = \mathbb{F}_0 \circ T$,
- *composition and the identity-assigning morphism are preserved up to 2-isomorphisms* $\mathbb{F}_\odot$ *and* $\mathbb{F}_U$*, respectively:*

$$\begin{array}{ccc} \mathbb{D}_1 \times_{\mathbb{D}_0} \mathbb{D}_1 & \xrightarrow{\circ} & \mathbb{D}_1 \\ {\scriptstyle \mathbb{F}_1 \times_{\mathbb{F}_0} \mathbb{F}_1}\downarrow & \mathbb{F}_\odot \nearrow & \downarrow{\scriptstyle \mathbb{F}_1} \\ \mathbb{D}'_1 \times_{\mathbb{D}'_0} \mathbb{D}'_1 & \xrightarrow{\circ'} & \mathbb{D}'_1 \end{array} \qquad \begin{array}{ccc} \mathbb{D}_0 & \xrightarrow{U} & \mathbb{D}_1 \\ {\scriptstyle \mathbb{F}_0}\downarrow & \mathbb{F}_U \nearrow & \downarrow{\scriptstyle \mathbb{F}_1} \\ \mathbb{D}'_0 & \xrightarrow{U'} & \mathbb{D}'_1 \end{array}$$

- *the 2-isomorphisms* $\mathbb{F}_\odot$ *and* $\mathbb{F}_U$ *satisfy the hexagon and square identities familiar from the definition of a monoidal functor.*

A weak functor in **Cat** is the same as a double functor, and one can consult Definition A.3 to see the hexagon and square identities in this case. We will also need weak functors in **Rex** and **SymMonCat**.

We begin by getting double functors between structured cospan double categories.

4.2. Theorem. *Suppose we have a square in* **Cat***:*

$$\begin{array}{ccc} \mathsf{A} & \xrightarrow{L} & \mathsf{X} \\ {\scriptstyle F_0}\downarrow & \alpha \nearrow & \downarrow{\scriptstyle F_1} \\ \mathsf{A}' & \xrightarrow[L']{} & \mathsf{X}' \end{array}$$

where X *and* X' *have chosen pushouts,* F_1 *preserves pushouts and* α *is a natural isomorphism. Then there is a double functor* $\mathbb{F}\colon {}_L\mathbb{C}\mathbf{sp}(\mathsf{X}) \to {}_{L'}\mathbb{C}\mathbf{sp}(\mathsf{X}')$ *such that:*

- $\mathbb{F}_0 = F_0$.
- $\mathbb{F}_1$ *acts as follows on objects:*

$$L(a) \xrightarrow{i} x \xleftarrow{o} L(b) \quad \mapsto \quad L'(F_0(a)) \xrightarrow{F_1(i)\alpha_a} F_1(x) \xleftarrow{F_1(o)\alpha_b} L'(F_0(b))$$

and as follows on morphisms:

$$\begin{array}{ccccc} L(a) & \xrightarrow{i} & x & \xleftarrow{o} & L(b) \\ {\scriptstyle L(f)}\downarrow & & \downarrow{\scriptstyle \gamma} & & \downarrow{\scriptstyle L(g)} \\ L(a') & \xrightarrow{i'} & x' & \xleftarrow{o'} & L(b') \end{array} \quad \mapsto \quad \begin{array}{ccccc} L'(F_0(a)) & \xrightarrow{F_1(i)\alpha_a} & F_1(x) & \xleftarrow{F_1(o)\alpha_b} & L'(F_0(b)) \\ {\scriptstyle L'(F_0(f))}\downarrow & & \downarrow{\scriptstyle F_1(\gamma)} & & \downarrow{\scriptstyle L'(F_0(g))} \\ L'(F_0(a')) & \xrightarrow{F_1(i')\alpha_{a'}} & F_1(x') & \xleftarrow{F_1(o')\alpha_{b'}} & L'(F_0(b')) \end{array}$$

- *Given composable structured cospans in* ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$*:*

$$L(a) \xrightarrow{i} x \xleftarrow{o} L(b) \qquad L(b) \xrightarrow{i'} y \xleftarrow{o'} L(c)$$

the natural isomorphism $\mathbb{F}_\odot\colon \mathbb{F}_1(M) \odot \mathbb{F}_1(N) \to \mathbb{F}_1(M \odot N)$ *is given by this map of cospans:*

$$\begin{array}{ccccc} L'(F_0(a)) & \xrightarrow{\Psi j_{F_1(x)} F_1(i)\alpha_a} & F_1(x) +_{L'(F_0(b))} F_1(y) & \xleftarrow{\Psi j_{F_1(y)} F_1(o')\alpha_a} & L'(F_0(c)) \\ {\scriptstyle 1}\downarrow & & \downarrow{\scriptstyle \phi_{M,N}} & & \downarrow{\scriptstyle 1} \\ L'(F_0(a)) & \xrightarrow{F_1(\psi j_x i)\alpha_c} & F_1(x +_{L(b)} y) & \xleftarrow{F_1(\psi j_y o')\alpha_c} & L'(F_0(c)) \end{array}$$

Here $j_x\colon x \to x + y$ *is the natural map into a coproduct, and likewise for* $j_y, j_{F_1(x)}, j_{F_1(y)}$, $\psi\colon x + y \to x +_{L(b)} y$ *is the natural map from a coproduct to a pushout and likewise for* Ψ*, and* $\phi_{M,N}\colon F_1(x) +_{L'(F_0(b))} F_1(y) \to F_1(x +_{L(b)} y)$ *is given by the composite*

$$F_1(x) +_{L'(F_0(b))} F_1(y) \xrightarrow{\mathrm{id} +_{\alpha_b} \mathrm{id}} F_1(x) +_{F_1(L(b))} F_1(y) \xrightarrow{\kappa} F_1(x +_{L(b)} y)$$

where κ *is the natural isomorphism arising from* F_1 *preserving pushouts.*

- *Given an object $a \in A$, the natural isomorphism $\mathbb{F}_U \colon U'(\mathbb{F}_0(a)) \to \mathbb{F}_1(U(a))$ is given by this map of cospans:*

$$\begin{array}{ccccc} L'(F_0(a)) & \xrightarrow{1} & L'(F_0(a)) & \xleftarrow{1} & L'(F_0(a)) \\ {\scriptstyle 1}\downarrow & & {\scriptstyle \alpha_a}\downarrow & & \downarrow{\scriptstyle 1} \\ L'(F_0(a)) & \xrightarrow{\alpha_a} & F_1(L(a)) & \xleftarrow{\alpha_a} & L'(F_0(a)) \end{array}$$

Proof. The diagram in the definition of $\mathbb{F}_\odot$ commutes as

$$F_1(\psi j_x i)\alpha_a = F_1(\psi) F_{1_{x,y}} j_{F_1(x)} F_1(i)\alpha_a = \phi_{M,N} \Psi j_{F_1(x)} F_1(i)\alpha_a$$

where $F_{1_{x,y}} \colon F_1(x) + F_1(y) \to F_1(x+y)$ is the natural isomorphism arising from F_1 preserving binary coproducts. One can check that the natural isomorphisms $\mathbb{F}_\odot$ and $\mathbb{F}_U$ satisfy the left and right unit squares and laxator hexagon of a monoidal functor. ■

4.3. Theorem. *Suppose we have a square commuting up to isomorphism in* **Rex***:*

$$\begin{array}{ccc} \mathsf{A} & \xrightarrow{L} & \mathsf{X} \\ {\scriptstyle F_0}\downarrow & \alpha \nearrow & \downarrow{\scriptstyle F_1} \\ \mathsf{A}' & \xrightarrow{L'} & \mathsf{X}' \end{array}$$

Then the double functor $\mathbb{F} \colon {}_L\mathbb{C}\mathbf{sp}(\mathsf{X}) \to {}_{L'}\mathbb{C}\mathbf{sp}(\mathsf{X}')$ is a weak functor between weak category objects in **Rex***. Moreover, if we make ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ and ${}_{L'}\mathbb{C}\mathbf{sp}(\mathsf{X}')$ into symmetric monoidal double categories as in Theorem 3.9, then $\mathbb{F} \colon {}_L\mathbb{C}\mathbf{sp}(\mathsf{X}) \to {}_{L'}\mathbb{C}\mathbf{sp}(\mathsf{X}')$ can be given the structure of a symmetric monoidal double functor.*

Proof. This is a straightforward but lengthy verification. ■

We can then water down this result, obtaining maps between symmetric monoidal bicategories or categories:

4.4. Theorem. *A symmetric monoidal double functor $\mathbb{F} \colon {}_L\mathbb{C}\mathbf{sp}(\mathsf{X}) \to {}_{L'}\mathbb{C}\mathbf{sp}(\mathsf{X}')$ induces a symmetric monoidal functor $\mathbf{F} \colon {}_L\mathbf{Csp}(\mathsf{X}) \to {}_{L'}\mathbf{Csp}(\mathsf{X}')$.*

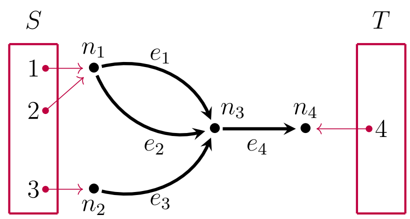
Proof. See Hansen and Shulman [22] for details of how this works, and a proof. ■

4.5. Theorem. *A symmetric monoidal functor between bicategories $\mathbf{F} \colon {}_L\mathbf{Csp}(\mathsf{X}) \to {}_{L'}\mathbf{Csp}(\mathsf{X}')$ induces a symmetric monoidal functor between categories $F \colon {}_L\mathsf{Csp}(\mathsf{X}) \to {}_{L'}\mathsf{Csp}(\mathsf{X}')$.*

Proof. This is a straightforward decategorification process. ■

5. Structured versus decorated cospans

We can illustrate some of the advantages of structured over decorated categories with an example that is fundamental in the study of networks: the double category with open graphs as morphisms. An 'open graph' consists of a graph together with maps from two sets into its set of nodes:



As usual in category theory, by 'graph' we mean a directed multigraph or quiver. In what follows we restrict attention to finite graphs because these are the most important in applications.

5.1. Definition. *A* **graph** *is a pair of functions* $s, t\colon E \to N$ *where* E *and* N *are finite sets. We call elements of* E **edges** *and elements of* N **nodes**. *We say that the edge* $e \in E$ *has* **source** $s(e)$ *and* **target** $t(e)$, *and say that* e *is an edge* **from** $s(e)$ **to** $t(e)$. *A* **morphism** *from the graph* $s, t\colon E \to N$ *to the graph* $s', t'\colon E' \to N'$ *is a pair of functions* $f\colon E \to E', g\colon N \to N'$ *such that these diagrams commute:*

$$\begin{array}{ccc} E & \xrightarrow{s} & N \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g} \\ E' & \xrightarrow{s'} & N' \end{array} \qquad \begin{array}{ccc} E & \xrightarrow{t} & N \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g} \\ E' & \xrightarrow{t'} & N'. \end{array}$$

5.2. Definition. *Let* Graph *be the category of graphs and morphisms between them, with composition defined by*

$$(f, g) \circ (f', g') = (f \circ f', g \circ g').$$

There is a functor $U\colon \mathsf{Graph} \to \mathsf{FinSet}$ that takes a graph $s, t\colon E \to N$ to its underlying set of nodes N. This has a left adjoint $L\colon \mathsf{FinSet} \to \mathsf{Graph}$ sending any set to the graph with that set of nodes and no edges. Both FinSet and Graph have finite colimits, and L, being a left adjoint, preserves them. Thus Theorem 3.9 gives us a symmetric monoidal double category ${}_L\mathbb{C}\mathbf{sp}(\mathsf{Graph})$ where:

- an object is a finite set,
- a vertical 1-morphism is a function between finite sets,
- a horizontal 1-cell from S to T is an **open graph**, meaning a cospan in Graph of this form:

$$L(S) \longrightarrow G \longleftarrow L(T),$$

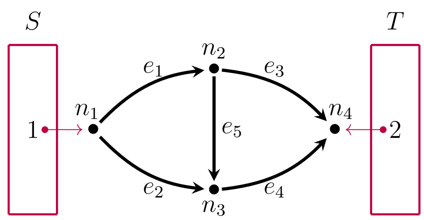
- a 2-morphism is a **map of open graphs**, meaning a commutative diagram in Graph of this form:

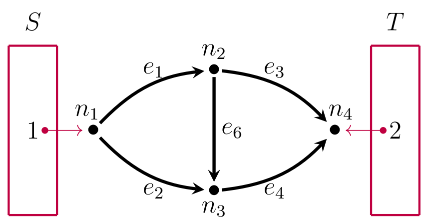
$$\begin{array}{ccccc} L(S) & \longrightarrow & G & \longleftarrow & L(T) \\ {\scriptstyle L(f)}\downarrow & & {\scriptstyle h}\downarrow & & \downarrow{\scriptstyle L(g)} \\ L(S') & \longrightarrow & G' & \longleftarrow & L(T') \end{array}$$

Applying Corollary 3.10 we obtain a symmetric monoidal bicategory ${}_L\mathbf{Csp}(\mathsf{Graph})$ where the objects are finite sets, the morphisms are open graphs, and the 2-morphisms are commutative diagrams in Graph of this form:

$$\begin{array}{ccccc} & \overset{i}{\nearrow} & G & \overset{o}{\nwarrow} & \\ L(S) & & {\scriptstyle h}\downarrow & & L(T) \\ & \underset{i'}{\searrow} & G' & \underset{o'}{\swarrow} & \end{array}$$

We can go further and apply Corollary 3.11 to obtain a symmetric monoidal category ${}_L\mathsf{Csp}(\mathsf{Graph})$ where the objects are finite sets and the morphisms are *isomorphism classes* of open graphs. An isomorphism of open graphs is a diagram as above where h is an isomorphism. Below is a pair of isomorphic open graphs.





These differ only in that the edge e_5 has been renamed e_6. We could also rename nodes, but we chose this example for a specific reason. We can define a similar category of open graphs using the machinery of decorated cospans. The morphisms in this other category are again equivalence classes of open graphs—but with a finer equivalence relation, for which the above open graphs are *not* equivalent! Indeed, this other notion of equivalence between open graphs only allows us to rename nodes, not edges.

Now let us compare the decorated cospan category of open graphs. We shall go into some detail here, since the problems we meet afflict a number of attempted applications of decorated cospans in the published literature [3, 4, 6, 13]. We start with a functor $F\colon \mathsf{FinSet} \to \mathsf{Set}$ that assigns to any finite set N the collection of all **graph structures** on N, meaning graphs whose set of nodes is N. A small issue immediately presents itself: as described, $F(N)$ is actually a proper class. We can get around this in various ways. For example, we can replace FinSet by an equivalent small category, and define a finite graph to be a diagram $s, t \colon E \to N$ in this category. Henceforth we consider this done.

The functor F acts on morphisms as follows: given any function $f \colon N \to N'$, we say that $F(f) \colon F(N) \to F(N')$ maps the graph structure $s, t \colon E \to N$ to the graph structure

$$f \circ s, f \circ t \colon E \to N'.$$

Thus, we use f to rename the nodes and let the edges 'go along for the ride'.

To obtain a symmetric monoidal category $F\mathsf{Cospan}$ as described in Section 1, we need to make F into a symmetric lax monoidal functor from $(\mathsf{FinSet}, +)$ to $(\mathsf{Set}, \times)$. There is an obvious choice of laxator

$$\phi_{N,N'} \colon F(N) \times F(N') \to F(N + N')$$

since there is a natural graph structure on $N + N'$ built from graph structures $s, t \colon E \to N$ and $s', t' \colon E' \to N'$: namely, $s + s', t + t' \colon E + E' \to N + N'$. However, as pointed out by an anonymous referee in a paper by Moeller and Vasilakopoulou [29], this diagram in the definition of lax monoidal functor may fail to commute:

$$\begin{array}{ccc}
(F(N) \times F(N')) \times F(N'') & \longrightarrow & F(N) \times (F(N') \times F(N'')) \\
{\scriptstyle \phi_{N,N'} \times 1} \downarrow & & \downarrow {\scriptstyle 1 \times \phi_{N',N''}} \\
F(N + N') \times F(N'') & & F(N) \times F(N' + N'') \\
{\scriptstyle \phi_{N+N',N''}} \downarrow & & \downarrow {\scriptstyle \phi_{N,N'+N''}} \\
F((N + N') + N'') & \longrightarrow & F(N + (N' + N''))
\end{array}$$

where the horizontal arrows are the associator in $(\mathsf{Set}, \times)$ and F of the associator in $(\mathsf{FinSet}, +)$, respectively. Suppose we start at upper left with a triple of graph structures $s, t \colon E \to N$, $s', t' \colon E' \to N'$ and $s'', t'' \colon E'' \to N''$. If we follow the arrows first down and then across, we obtain a graph structure on $N + (N' + N'')$ where the set of edges is $(E + E') + E''$. If instead we follow the arrows first across and then down, we obtain a graph structure where the set of edges is $E + (E' + E'')$. These graph structures are different if $(E + E') + E'' \neq E + (E' + E'')$. The problem is that $(\mathsf{FinSet}, +)$ may not be a strict monoidal category. We say "may not" because we have replaced the original $(\mathsf{FinSet}, +)$ by an equivalent small category.

Of course we can use Mac Lane's coherence theorem to choose an equivalent monoidal category that is both small and strict. One can then prove F becomes lax monoidal with ϕ as its laxator—but still not *symmetric* lax monoidal. The problem is that this diagram fails to commute:

$$\begin{array}{ccc}
F(N) \times F(N') & \longrightarrow & F(N') \times F(N) \\
{\scriptstyle \phi_{N,N'}} \downarrow & & \downarrow {\scriptstyle \phi_{N',N}} \\
F(N + N') & \longrightarrow & F(N' + N)
\end{array}$$

where the horizontal arrows are the braiding in $(\mathsf{Set}, \times)$ and F of the braiding in $(\mathsf{FinSet}, +)$, respectively. Suppose we start at upper left with a pair of graph structures $s, t\colon E \to N$ and $s', t'\colon E' \to N'$. If we follow the arrows first down and then across we obtain a graph structure on $N' + N$ where the set of edges is $E + E'$, but if we follow the arrows first across and then down we obtain a graph structure where the set of edges is $E' + E$. These graph structures are different in general, and we cannot cure this problem with further strictification: $(\mathsf{FinSet}, +)$ is not equivalent as a symmetric monoidal category to one that where the braiding is the identity.

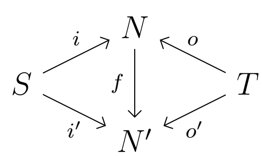
As a result, the theory of decorated cospans only gives a monoidal category $F\mathsf{Cospan}$ [10, Thm. 2.1.3]. An object of $F\mathsf{Cospan}$ is a finite set, while a morphism is an equivalence class of F-decorated cospans

$$S \xrightarrow{\ i\ } N \xleftarrow{\ o\ } T, \quad G \in F(N).$$

Such an F-decorated cospan is a way of describing an open graph from S to T. However, two such F-decorated cospans, say the above one and this:

$$S \xrightarrow{\ i\ } N' \xleftarrow{\ o\ } T, \ G' \in F'(N),$$

are equivalent iff there is a bijection $f\colon N \to N'$ making this diagram commute:



and such that $F(f)(G) = G'$. It follows that the graphs $G = (s, t\colon E \to N)$ and $G' = (s', t'\colon E' \to N')$ are isomorphic, but in a specific way: we must have $E' = E$, $s' = f \circ s$, and $t' = f \circ t$. Thus, two open graphs with different edge sets cannot be equivalent!

In short, the decorated cospan category of open graphs resembles the structured cospan category, but it is merely monoidal, not symmetric monoidal, and it has many morphisms for each morphism in the structured cospan category, for no particularly useful reason. This 'redundancy' is eliminated by the functor $J\colon F\mathsf{Cospan} \to {}_L\mathsf{Csp}(\mathsf{Graph})$ that is the identity on objects and identifies isomorphic open graphs.

In attempted applications so far, one often uses a decorated cospan category as the 'syntax' for open systems of a particular kind, with the 'semantics' given by a monoidal functor out of this category [14]. Often this functor factors through a structured cospan category that eliminates the redundancy in the morphisms of the structured cospan category. We give some examples in the next section.

On the other hand, there are also useful decorated cospan categories that do not suffer from the problems we have described. Some appear not to be structured cospan categories. An example is the category of open dynamical systems described in Section

6.16. Furthermore, the theory of decorated cospans plays an important role in the more general theory of decorated corelations [15, 16]. So, it also interesting to see if we can improve the theory of decorated cospans a bit to eliminate the problems we have seen.

In the case of open graphs, one cheap solution is to use a different symmetric lax monoidal functor, say $F' \colon (\mathsf{FinSet}, +) \to (\mathsf{Set}, \times)$, that sends any finite set N to the set of *isomorphism classes* of graph structures on N. Here given two graph structures $s, t \colon E \to N$ and $s', t' \colon E' \to N$ on N, we define a **morphism** from the first to the second to be a function $f \colon E \to E'$ such that these diagrams commute:

$$\begin{array}{ccc} E & \xrightarrow{s} & N \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle 1} \\ E' & \xrightarrow{s'} & N \end{array} \qquad \begin{array}{ccc} E & \xrightarrow{t} & N \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle 1} \\ E' & \xrightarrow{t'} & N \end{array}$$

We obtain a category of graph structures on N in this way, allowing us to define isomorphism classes of these. One can check that using the theory of decorated cospans we obtain a symmetric monoidal category $F'\mathsf{Cospan}$ that is equivalent to ${}_L\mathsf{Csp}(\mathsf{Graph})$.

However, working with isomorphism classes of graph structures does not give a double category of decorated cospans that is equivalent to ${}_L\mathbb{C}\mathbf{sp}(\mathsf{Graph})$. We should really work with the *category* of graph structures, not isomorphism classes of graph structures! A clue to a better approach is to note that the forgetful functor $U \colon \mathsf{Graph} \to \mathsf{FinSet}$ is an opfibration, and the category of graph structures on a finite set N is the fiber of this opfibration over N. Thus, the inverse Grothendieck construction gives a pseudofunctor $\tilde{F} \colon \mathsf{FinSet} \to \mathbf{Cat}$ sending each finite set N to the category of graph structures on N. Moreover, $\tilde{F}$ is symmetric lax monoidal from $(\mathsf{FinSet}, +)$ to $(\mathbf{Cat}, \times)$.

In a forthcoming paper with Vasilakopoulou [1], we extend the theory of decorated cospans to handle this sort of data. That is, given a category A with finite colimits and a symmetric lax monoidal pseudofunctor $\tilde{F} \colon (\mathsf{A}, +) \to (\mathbf{Cat}, \times)$, we construct a symmetric monoidal double category $\tilde{F}\mathbb{C}\mathbf{ospan}$ with decorated cospans as horizontal 1-cells. Any such pseudofunctor also gives an opfibration $R \colon \mathsf{X} \to \mathsf{A}$ where $\mathsf{X} = \int \tilde{F}$ is defined by the Grothendieck construction. We show that if the symmetric lax monoidal pseudofunctor $\tilde{F} \colon (\mathsf{A}, +) \to (\mathbf{Cat}, \times)$ factors through $(\mathbf{Rex}, \times)$, the resulting opfibration $R \colon \mathsf{X} \to \mathsf{A}$ is also a right adjoint. From the accompanying left adjoint $L \colon \mathsf{A} \to \mathsf{X}$, we construct a symmetric monoidal double category ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ of structured cospans. Finally, we prove that this structured cospan double category ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$ is equivalent to the decorated cospan double category $\tilde{F}\mathbb{C}\mathbf{ospan}$. Thus, we reconcile the theory of structured cospans and the theory of decorated cospan categories by enhancing the latter.

6. Applications

Decorated cospans have already been used to study electrical circuits [3], Markov processes [4], and chemical reaction networks [6], while structured cospans have been used to study electrical circuits [2] and Petri nets [5]. Here we revisit this work and show that structured cospans can take the place of decorated cospans in many of these applications. For structured cospans in graph rewriting, see Cicala's thesis [8].

6.1. Circuits. Building on work with Fong [3], Coya, Rebro and the first author have used structured cospans to describe electrical circuits with inputs and outputs [2]. The key idea is to use graphs with labeled edges. The edge labels can stand for resistors with any chosen resistance, capacitors with any chosen capacitance, inductors with any chosen inductance, or other circuit elements such as voltage sources, current sources, diodes, and so on. To study such circuits quite generally we start by fixing any set $\mathcal{L}$ to serve as edge labels.

6.2. Definition. *Given a set $\mathcal{L}$ of* **labels**, *an* **$\mathcal{L}$-graph** *is a graph $s, t \colon E \to N$ equipped with a function $\ell \colon E \to \mathcal{L}$. A* **morphism** *from the $\mathcal{L}$-graph*

$$\mathcal{L} \xleftarrow{\ \ell\ } E \overset{s}{\underset{t}{\rightrightarrows}} N$$

to the $\mathcal{L}$-graph

$$\mathcal{L} \xleftarrow{\ \ell'\ } E' \overset{s'}{\underset{t'}{\rightrightarrows}} N'$$

is a pair of functions $f \colon E \to E', g \colon N \to N'$ such that these diagrams commute:

$$\begin{array}{ccc} E & \xrightarrow{s} & N \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g} \\ E' & \xrightarrow{s'} & N' \end{array} \qquad \begin{array}{ccc} E & \xrightarrow{t} & N \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g} \\ E' & \xrightarrow{t'} & N' \end{array} \qquad \begin{array}{ccc} & \overset{\ell}{\swarrow} & E \\ \mathcal{L} & & \downarrow{\scriptstyle f} \\ & \underset{\ell'}{\nwarrow} & E'. \end{array}$$

We say such a morphism is **determined by its action on nodes** *if $E' = E$ and $f = 1_E$.*

6.3. Definition. *We define $\mathsf{Graph}_{\mathcal{L}}$ to be the category of $\mathcal{L}$-graphs and morphisms between them, with composition given by*

$$(f, g) \circ (f', g') = (f \circ f', g \circ g').$$

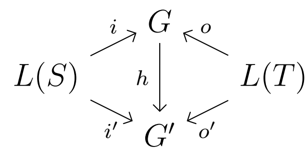
When $\mathcal{L} = 1$, an $\mathcal{L}$-graph reduces to a graph and $\mathsf{Graph}_{\mathcal{L}}$ reduces to the category Graph discussed in Section 5. We now generalize the key ideas of that section from graphs to $\mathcal{L}$-graphs. Everything works the same way, but following previous work [2] we call an open $\mathcal{L}$-graph an '$\mathcal{L}$-circuit'.

There is a functor $U \colon \mathsf{Graph}_{\mathcal{L}} \to \mathsf{FinSet}$ that takes an $\mathcal{L}$-graph to its underlying set of nodes. This has a left adjoint $L \colon \mathsf{FinSet} \to \mathsf{Graph}_{\mathcal{L}}$ sending any set to the $\mathcal{L}$-graph with that set of nodes and no edges. Both FinSet and $\mathsf{Graph}_{\mathcal{L}}$ have colimits, and L preserves them. Thus Theorem 3.9 gives us a symmetric monoidal double category ${}_L\mathbb{C}\mathbf{sp}(\mathsf{Graph}_{\mathcal{L}})$. Alternatively, we can use Corollary 3.11 to create a symmetric monoidal category ${}_L\mathsf{Csp}(\mathsf{Graph}_{\mathcal{L}})$ where:

- an object is a finite set,
- a morphism is an isomorphism class of $\mathcal{L}$-circuits, where an **$\mathcal{L}$-circuit** is a cospan in $\mathsf{Graph}_{\mathcal{L}}$ of this form:

$$L(S) \longrightarrow G \longleftarrow L(T),$$

  and an **isomorphism** of $\mathcal{L}$-circuits is a commutative diagram in $\mathsf{Graph}_{\mathcal{L}}$ of this form:



  where h is an isomorphism.

This category has a nice universal property, found by Rosebrugh, Sabadini and Walters [32]. To state this, it is convenient to use the language of props. Recall that a **prop** is a symmetric strict monoidal category whose objects are natural numbers, with tensor product of objects given by addition. An **algebra** of a prop T in a symmetric strict monoidal category C is a symmetric strict monoidal functor $A\colon \mathsf{T} \to \mathsf{C}$. A **morphism** from the algebra $A\colon \mathsf{T} \to \mathsf{C}$ to the algebra $A'\colon \mathsf{T} \to \mathsf{C}$ is a monoidal natural transformation $\alpha\colon A \Rightarrow A'$.

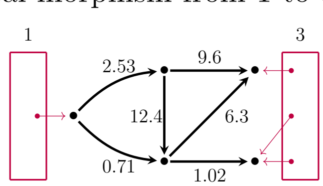
6.4. Lemma. *As a symmetric monoidal category, ${}_L\mathsf{Csp}(\mathsf{Graph}_{\mathcal{L}})$ is equivalent to a prop $\mathsf{Circ}_{\mathcal{L}}$.*

Proof. This is [2, Proposition 4.3]. ■

6.5. Proposition. *An algebra of $\mathsf{Circ}_{\mathcal{L}}$ in a symmetric strict monoidal category C is a special commutative Frobenius monoid in C whose underlying object x is equipped with an endomorphism $\ell\colon x \to x$ for each element $\ell \in \mathcal{L}$. A morphism of algebras of $\mathsf{Circ}_{\mathcal{L}}$ in C is a morphism of special commutative Frobenius monoids that also preserves all these endomorphisms.*

Proof. This was proved by Rosebrugh, Sabadini and Walters [32], and appears in the above form in [2, Proposition 7.2]. ■

In applications to circuits, the morphisms $\ell\colon x \to x$ describe different circuit elements, while the special commutative Frobenius monoid structure is used to split and join wires. This framework is used to study a wide variety of electrical circuits in a paper with Coya and Rebro [2], so the reader can turn there for details. To illustrate the ideas let us consider circuits of resistors, where a label in $\mathcal{L} = (0, \infty)$ serves to indicate the resistance of a resistor. In this case a typical morphism from 1 to 3 in $\mathsf{Circ}_{\mathcal{L}}$ looks like this:



The edges here represent wires, and the positive real numbers labeling them describe the resistance of the resistor on each wire. The points in the boxes represent 'terminals': that is, points where we allow ourselves to attach a wire from another circuit. The points in the left box are called 'inputs' and the points in the right box are called 'outputs'. In electrical engineering we associate two real numbers to each terminal, called 'potential' and 'current'. Any circuit of resistors imposes a specific relation between the potentials and currents at its inputs and those at its outputs. All these relations, for all circuits of resistors, can be described using a single functor as follows.

There is a symmetric monoidal category $\mathsf{FinRel}_{\mathbb{R}}$ where the objects are finite-dimensional real vector spaces and a morphism from V to W is a **linear relation** from V to W: that is, a relation $L \subseteq V \times W$ that is a linear subspace of $V \times W$. Composition in $\mathsf{FinRel}_{\mathbb{R}}$ is the usual composition of relations, and the symmetric monoidal structure is provided by direct sum.

There is a symmetric monoidal functor

$$\blacksquare \colon \mathsf{Circ}_{\mathcal{L}} \to \mathsf{FinRel}_{\mathbb{R}}$$

sending any finite set S to the vector space $\mathbb{R}^S \oplus \mathbb{R}^S$ and sending any circuit of resistors to the relation it imposes between the potentials and currents at its inputs and those at its outputs [2, Section 9]. We can construct this using Proposition 6.5, by choosing a special commutative Frobenius monoid in $\mathsf{FinRel}_{\mathbb{R}}$ whose underlying object is equipped with an endomorphism for each resistance $R \in (0, \infty)$. The object $\mathbb{R}^2 \in \mathsf{FinRel}_{\mathbb{R}}$ is a special commutative Frobenius monoid in a standard way [2, Section 8], so we choose this one. To define $\blacksquare \colon \mathsf{Circ}_{\mathcal{L}} \to \mathsf{FinRel}_{\mathbb{R}}$ it then suffices to choose for each $R \in (0, \infty)$ a linear relation from $\mathbb{R}^2$ to itself. We use this:

$$\{(\phi_1, I_1, \phi_2, I_2) :\ I_1 = I_2,\ \phi_2 - \phi_1 = RI_1\} \subseteq \mathbb{R}^2 \oplus \mathbb{R}^2.$$

This expresses two laws of electrical engineering. Kirchhoff's current law says that the current flowing into a wire equals the current flowing out: $I_1 = I_2$. Ohm's law says that the voltage across a wire with a resistor on it, $\phi_2 - \phi_1$, is equal to the current flowing through the wire times the resistance R of that resistor.

Earlier work with Fong studied circuits using decorated rather than structured cospans [3], and it fell afoul of the problems discussed in Section 5. We make no attempt to explain the results here, but we can quickly explain a corrected version of this decorated cospan category of circuits. For any set $\mathcal{L}$, define an **$\mathcal{L}$-graph structure** on a finite set N to be an $\mathcal{L}$-graph whose set of nodes is N. If we use a small strict monoidal version of $(\mathsf{FinSet}, +)$, there is a lax monoidal functor

$$F_{\mathcal{L}} \colon (\mathsf{FinSet}, +) \to (\mathsf{Set}, \times)$$

assigning to each finite set N the collection of all $\mathcal{L}$-graph structures on N. The theory of decorated cospans [10, Thm. 2.1.3] thus gives a monoidal category $F_{\mathcal{L}}\mathsf{Cospan}$ where:

- an object is a finite set,

- a morphism is an equivalence class of $\mathcal{L}$-circuits

$$L(S) \longrightarrow G \longleftarrow L(T)$$

where two are equivalent if there is a commutative diagram in $\mathsf{Graph}_{\mathcal{L}}$ of this form:

$$\begin{array}{ccccc} & \overset{i}{\nearrow} & G & \overset{o}{\nwarrow} & \\ L(S) & & \downarrow h & & L(T) \\ & \underset{i'}{\searrow} & G' & \underset{o'}{\swarrow} & \end{array}$$

with h an isomorphism that is determined by its action on nodes in the sense of Definition 6.3.

The restriction that h be determined by its action on nodes means that isomorphic $\mathcal{L}$-circuits can give different morphisms in $F_{\mathcal{L}}\mathsf{Cospan}$. However, there is a functor

$$J\colon F_{\mathcal{L}}\mathsf{Cospan} \to \mathsf{Circ}_{\mathcal{L}}$$

that eliminates this redundancy: it is the identity on objects, and it maps each open circuit to its isomorphism class. Furthermore, $\mathsf{Circ}_{\mathcal{L}}$ is symmetric monoidal, while $F_{\mathcal{L}}\mathsf{Cospan}$ is merely monoidal, due to the problem discussed in Section 5.

6.6. PETRI NETS. Petri nets are widely used by computer scientists as a simple model of distributed, concurrent computation [18, 30]. From the viewpoint of a category theorist, a Petri net is a convenient way to present a simple sort of symmetric monoidal category: namely, a *commutative* monoidal category—a commutative monoid object in **Cat**—that is free on some objects and morphisms [28]. Recently Master and the first author studied 'open' Petri nets using structured cospans [5]. By composing and tensoring open Petri nets, we can build complicated Petri nets out of smaller pieces. As we shall see, the semantics of open Petri nets is a nice illustration of our main method of describing maps between structured cospan categories, Theorem 4.3.

To define Petri nets [28] we start with the monad for commutative monoids, $\mathbb{N}\colon \mathsf{Set} \to \mathsf{Set}$. Concretely, $\mathbb{N}[X]$ is the set of formal finite linear combinations of elements of X with natural number coefficients. The set X naturally includes in $\mathbb{N}[X]$, and for any function $f\colon X \to Y$, there is a unique monoid homomorphism $\mathbb{N}[f]\colon \mathbb{N}[X] \to \mathbb{N}[Y]$ extending f.

6.7. DEFINITION. *We define a* **Petri net** *to be a pair of functions of the following form:*

$$T \overset{s}{\underset{t}{\rightrightarrows}} \mathbb{N}[S].$$

We call T the set of **transitions***, S the set of* **places***, s the* **source** *function and t the* **target** *function. A* **morphism** *from the Petri net $s, t\colon T \to \mathbb{N}[S]$ to the Petri net*

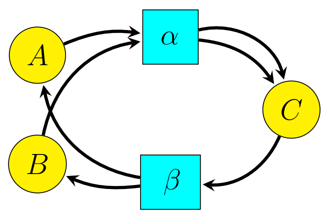
$s', t' \colon T' \to \mathbb{N}[S']$ is a pair of functions $f \colon T \to T'$, $g \colon S \to S'$ such that the following diagrams commute:

$$\begin{array}{ccc} T & \xrightarrow{s} & \mathbb{N}[S] \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle \mathbb{N}[g]} \\ T' & \xrightarrow{s'} & \mathbb{N}[S'] \end{array} \qquad \begin{array}{ccc} T & \xrightarrow{t} & \mathbb{N}[S] \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle \mathbb{N}[g]} \\ T' & \xrightarrow{t'} & \mathbb{N}[S']. \end{array}$$

Let Petri *be the category of Petri nets and Petri net morphisms, with composition defined by*

$$(f, g) \circ (f', g') = (f \circ f', g \circ g').$$

It is commmon to draw a Petri net as a bipartite graph with the places as circles and the transitions as squares:



However, we must bear in mind that the edges in this graph are merely a device for describing the source and target of each transition: there is not really a set of edges from a place to a transition or a transition to a place, but merely a *number*. For example, α above is a transition with $s(\alpha) = A + B$ and $t(\alpha) = 2C$.

Any Petri net has an underlying set of places. Indeed there is a functor $R \colon \mathsf{Petri} \to \mathsf{Set}$ that acts as follows on Petri nets and Petri net morphisms:

$$\begin{array}{ccc} T & \overset{s}{\underset{t}{\rightrightarrows}} & \mathbb{N}[S] \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle \mathbb{N}[g]} \\ T' & \overset{s'}{\underset{t'}{\rightrightarrows}} & \mathbb{N}[S'] \end{array} \quad \mapsto \quad \begin{array}{c} S \\ \downarrow{\scriptstyle g} \\ S'. \end{array}$$

To build a structured cospan category we need the left adjoint of R, and we need Petri to have finite colimits.

6.8. Lemma. *The functor R has a left adjoint $L \colon \mathsf{Set} \to \mathsf{Petri}$ defined on sets and functions as follows:*

$$\begin{array}{c} X \\ {\scriptstyle g}\downarrow \\ Y \end{array} \quad \mapsto \quad \begin{array}{ccc} \emptyset & \rightrightarrows & \mathbb{N}[X] \\ \downarrow & & \downarrow{\scriptstyle \mathbb{N}[g]} \\ \emptyset & \rightrightarrows & \mathbb{N}[Y] \end{array}$$

where the unlabeled maps are the unique maps of that type.

Proof. This is [5, Lemma 11]. ∎

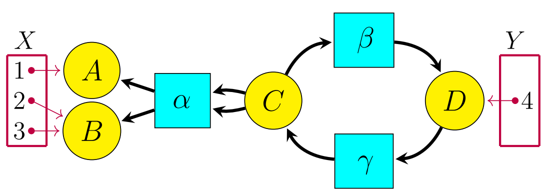
6.9. Lemma. *The category* Petri *has small colimits.*

Proof. This is [5, Lemma 15]. ∎

Thanks to these lemmas, Theorem 3.9 gives us a symmetric monoidal double category ${}_L\mathbb{C}\mathbf{sp}(\mathsf{Petri})$, or $\mathbb{O}\mathbf{pen}(\mathsf{Petri})$ for short, in which:

- an object is a set,
- a vertical 1-morphism is a function,
- a horizontal 1-cell from X to Y is an **open Petri net**, meaning a cospan in Petri of this form:
$$L(X) \longrightarrow P \longleftarrow L(Y),$$
- a 2-morphism is a **map of open Petri nets**, meaning a commutative diagram in Petri of this form:
$$\begin{array}{ccccc} L(X) & \longrightarrow & P & \longleftarrow & L(Y) \\ {\scriptstyle L(f)}\downarrow & & {\scriptstyle h}\downarrow & & \downarrow{\scriptstyle L(g)} \\ L(X') & \longrightarrow & P' & \longleftarrow & L(Y'). \end{array}$$

We can draw an open Petri net as a Petri net with maps from sets X and Y into its set of places:



We explained composition and tensoring of open Petri nets using pictures in Section 1.

Now we construct a structured cospan category $\mathbb{O}\mathbf{pen}(\mathsf{CMC})$ of 'open commutative monoidal categories' and a map

$$\mathbb{O}\mathbf{pen}(F)\colon \mathbb{O}\mathbf{pen}(\mathsf{Petri}) \to \mathbb{O}\mathbf{pen}(\mathsf{CMC}).$$

6.10. Definition. *A* **commutative monoidal category** *is a symmetric strict monoidal category where all the braidings $a \otimes b \to b \otimes a$ are identities. A* **morphism** *of commutative monoidal categories is a symmetric strict monoidal functor.*

6.11. Definition. *Let* CMC *be the category of commutative monoidal categories and morphisms between them.*

Any commutative monoidal category has an underlying set of objects. Let $R'\colon \mathsf{CMC} \to \mathsf{Set}$ be the functor sending any commutative monoidal category to its underlying set of objects and any morphism to its underlying function on objects. To build a structured cospan category of open commutative monoidal categories we use a left adjoint of R', and we need CMC to have finite colimits.

6.12. Lemma. *The functor R' has a left adjoint $L'\colon \mathsf{Set} \to \mathsf{CMC}$ sending any set S to the commutative monoidal category with $\mathbb{N}[S]$ as its commutative monoid of objects and with only identity morphisms.*

Proof. This is [5, Lemma 9]. ∎

6.13. Lemma. *The category CMC has small colimits.*

Proof. This can be shown in various ways; see [5, Theorem 16] for two. ∎

Thanks to these lemmas, Theorem 3.9 gives us a symmetric monoidal double category ${}_{L'}\mathbb{C}\mathbf{sp}(\mathsf{CMC})$, or $\mathbb{O}\mathbf{pen}(\mathsf{CMC})$ for short, in which:

- an object is a set,
- a vertical 1-morphism is a function,
- a horizontal 1-cell from X to Y is an **open commutative monoidal category**, meaning a cospan in CMC of this form:

$$L'(X) \longrightarrow \mathsf{C} \longleftarrow L'(Y),$$

- a 2-morphism is a **map of open commutative monoidal categories**, meaning a commutative diagram in CMC of this form:

$$\begin{array}{ccccc} L'(X) & \longrightarrow & \mathsf{C} & \longleftarrow & L'(Y) \\ {\scriptstyle L'(f)}\downarrow & & {\scriptstyle h}\downarrow & & \downarrow{\scriptstyle L'(g)} \\ L'(X') & \longrightarrow & \mathsf{C}' & \longleftarrow & L(Y'). \end{array}$$

We can turn a Petri net $P = (s, t\colon T \to \mathbb{N}[S])$ into a commutative monoidal category FP as follows. We take the commutative monoid of objects $\mathrm{Ob}(FP)$ to be the free commutative monoid on S. We construct the commutative monoid of morphisms $\mathrm{Mor}(FP)$ as follows. First we generate morphisms recursively:

- for every transition $\tau \in T$ we include a morphism $\tau\colon s(\tau) \to t(\tau)$;
- for any object a we include a morphism $1_a\colon a \to a$;
- for any morphisms $f\colon a \to b$ and $g\colon a' \to b'$ we include a morphism denoted $f + g\colon a + a' \to b + b'$ to serve as their tensor product;
- for any morphisms $f\colon a \to b$ and $g\colon b \to c$ we include a morphism $g \circ f\colon a \to c$ to serve as their composite.

Then we mod out by an equivalence relation on morphisms that imposes the laws of a commutative monoidal category, obtaining the commutative monoid $\mathrm{Mor}(FP)$.

Let $F\colon \mathsf{Petri} \to \mathsf{CMC}$ be the functor that makes the following assignments on Petri nets and morphisms:

$$\begin{array}{ccccc} T & \overset{s}{\underset{t}{\rightrightarrows}} & \mathbb{N}[S] & & FP \\ {\scriptstyle f}\big\downarrow & & \big\downarrow{\scriptstyle \mathbb{N}[g]} & \mapsto & \big\downarrow{\scriptstyle F(f,g)} \\ T' & \overset{s'}{\underset{t'}{\rightrightarrows}} & \mathbb{N}[S'] & & FP'. \end{array}$$

Here $F(f,g)\colon FP \to FP'$ is defined on objects by $\mathbb{N}[g]$. On morphisms, $F(f,g)$ is the unique map extending f that preserves identities, composition, and the tensor product.

6.14. Lemma. *The functor*

$$F\colon \mathsf{Petri} \to \mathsf{CMC}$$

is a left adjoint.

Proof. This is a special case of [27, Theorem 5.1]. ∎

We thus obtain a triangle of left adjoint functors, which commutes up to natural isomorphism:

$$\begin{array}{ccc} \mathsf{Set} & \xrightarrow{L} & \mathsf{Petri} \\ & {\scriptstyle L'}\searrow \;\; \alpha \nearrow\!\!\!\nearrow & \big\downarrow{\scriptstyle F} \\ & & \mathsf{CMC} \end{array}$$

As a result we obtain:

6.15. Theorem. *There is a symmetric monoidal double functor*

$$\mathbb{O}\mathbf{pen}(F)\colon \mathbb{O}\mathbf{pen}(\mathsf{Petri}) \to \mathbb{O}\mathbf{pen}(\mathsf{CMC})$$

that is the identity on objects and vertical 1-morphisms and makes the following assignments on horizontal 1-cells and 2-morphisms:

$$\begin{array}{ccccccccccc} LX & \xrightarrow{i} & P & \xleftarrow{o} & LY & & L'X & \xrightarrow{F(i)\alpha_X} & FP & \xleftarrow{F(o)\alpha_Y} & L'Y \\ {\scriptstyle Lf}\big\downarrow & & \big\downarrow{\scriptstyle h} & & \big\downarrow{\scriptstyle Lg} & \mapsto & {\scriptstyle L'f}\big\downarrow & & \big\downarrow{\scriptstyle Fh} & & \big\downarrow{\scriptstyle L'g} \\ LX' & \xrightarrow{i'} & P' & \xleftarrow{o'} & LY' & & L'X' & \xrightarrow{F(i')\alpha_{X'}} & FP' & \xleftarrow{F(o')\alpha_{Y'}} & L'Y'. \end{array}$$

Proof. The triangle above is a degenerate case of the square studied in Theorem 4.2:

$$\begin{array}{ccc} \mathsf{Set} & \xrightarrow{L} & \mathsf{Petri} \\ {\scriptstyle 1}\big\downarrow & \alpha \nearrow\!\!\!\nearrow & \big\downarrow{\scriptstyle F} \\ \mathsf{Set} & \xrightarrow{L'} & \mathsf{CMC} \end{array}$$

and applying that theorem we obtain the desired result. ∎

In the language of computer science, the commutative monoidal category FP provides an 'operational semantics' for the Petri net P: morphisms in this category are processes allowed by the Petri net. The above theorem says that this semantics is compositional. That is, if we write P as a composite (or tensor product) of smaller open Petri nets, FP will be the composite (or tensor product) of the corresponding open commutative monoidal categories.

6.16. Petri nets with rates. Chemists often describe collections of chemical reactions using 'reaction networks'. They have a standard formalism for obtaining a dynamical system from any reaction network where each reaction is labeled by a positive real number called its 'rate constant' [23]. Reaction networks equipped with rate constants are equivalent to Petri nets where every transition is labeled by a positive real number. These are sometimes called 'stochastic' Petri nets, and they are used not only in chemistry but also biology and other fields [21, 25].

Pollard and the first author studied 'open' reaction networks using decorated cospans [6]. Here we show how to translate some of that work into the language of structured cospans. We need a finiteness condition in many applications, so we include that from the start.

6.17. Definition. *A* **Petri net with rates** *is a Petri net* $s, t\colon T \to \mathbb{N}[S]$ *where* S *and* T *are finite sets, together with a function* $r\colon T \to (0,\infty)$. *We call* $r(\tau)$ *the* **rate constant** *of the transition* $\tau \in T$. *A* **morphism** *from the Petri net with rates*

$$(0,\infty) \xleftarrow{r} T \overset{s}{\underset{t}{\rightrightarrows}} \mathbb{N}[S]$$

to the Petri net with rates

$$(0,\infty) \xleftarrow{r'} T' \overset{s'}{\underset{t'}{\rightrightarrows}} \mathbb{N}[S']$$

is a morphism $f\colon T \to T', g\colon S \to S'$ *of the underlying Petri nets such that the following diagram also commutes:*

$$\begin{array}{ccc} & & T \\ & \swarrow^{r} & \\ (0,\infty) & & \downarrow f \\ & \nwarrow_{r'} & \\ & & T' \end{array}$$

Let Petri_r *be the category of Petri nets with rates and morphisms between them, with composition defined by*

$$(f,g)\circ(f',g') = (f\circ f', g\circ g').$$

There is a functor $R\colon \mathsf{Petri}_r \to \mathsf{Set}$ that sends any Petri net with rates to its underlying set of places

$$\begin{array}{ccccccc} (0,\infty) & \xleftarrow{r} & T & \overset{s}{\underset{t}{\rightrightarrows}} & \mathbb{N}[S] & & S \\ \downarrow 1 & & \downarrow f & & \downarrow \mathbb{N}[g] & \mapsto & \downarrow g \\ (0,\infty) & \xleftarrow{r'} & T' & \overset{s'}{\underset{t'}{\rightrightarrows}} & \mathbb{N}[S'] & & S'. \end{array}$$

To build a structured cospan category we use the left adjoint of R, and we need Petri_r to have finite colimits.

6.18. Lemma. *The functor R has a left adjoint $L\colon \mathsf{Set} \to \mathsf{Petri}_r$ defined on sets and functions as follows:*

$$
\begin{array}{ccc}
X & & (0,\infty) \longleftarrow \emptyset \rightrightarrows \mathbb{N}[X] \\
{\scriptstyle f}\big\downarrow & \mapsto & {\scriptstyle 1}\big\downarrow \qquad\quad \big\downarrow \qquad\quad \big\downarrow{\scriptstyle \mathbb{N}[f]} \\
Y & & (0,\infty) \longleftarrow \emptyset \rightrightarrows \mathbb{N}[Y]
\end{array}
$$

where the unlabeled maps are the unique maps of that type.

Proof. This is easily checked from the definitions. ■

6.19. Lemma. *The category Petri_r has finite colimits.*

Proof. Note that Petri_r is equivalent to the comma category f/g where $f\colon \mathsf{FinSet} \to \mathsf{FinSet}$ is the identity and $g\colon \mathsf{FinSet} \to \mathsf{FinSet}$ is $(0,\infty) \times \mathbb{N}[-]^2$. Whenever A and B are have finite colimits, $f\colon A \to C$ preserves finite colimits and $g\colon B \to C$ is any functor, then f/g has finite colimits [7, Section 5.2, Theorem 3]. ■

As a consequence of these lemmas, Corollary 3.11 gives a symmetric monoidal category ${}_L\mathsf{Csp}(\mathsf{Petri}_r)$, or $\mathsf{Open}(\mathsf{Petri}_r)$ for short, in which:

- an object is a finite set,
- a morphism is an isomorphism class of open Petri nets with rates, where an **open Petri net with rates** is a cospan in Petri_r of this form:

  $$L(X) \longrightarrow P \longleftarrow L(Y),$$

  and an **isomorphism** of such is a commutative diagram in Petri_r of this form:

  $$
  \begin{array}{ccccc}
  & \overset{i}{\nearrow} & P & \overset{o}{\nwarrow} & \\
  L(X) & & {\scriptstyle h}\big\downarrow & & L(Y) \\
  & \underset{i'}{\searrow} & P' & \underset{o'}{\swarrow} &
  \end{array}
  $$

  where h is an isomorphism.

Pollard and the first author [6] used decorated cospans to construct a symmetric monoidal category RxNet equivalent to $\mathsf{Open}(\mathsf{Petri}_r)$. They avoided the 'redundancy problem' using a trick explained in Section 5. Namely, they used a symmetric lax monoidal functor $F'\colon (\mathsf{FinSet}, +) \to (\mathsf{Set}, \times)$ sending any finite set S to the set of *isomorphism classes* of Petri nets with rates having S as their set of places.

Pollard and the first author then constructed a symmetric monoidal functor from RxNet to a category Dynam of 'open dynamical systems', and a further symmetric monoidal functor from Dynam assigning to each open dynamical system the relation between its inputs and outputs that holds in steady state. Thanks to the equivalence between RxNet and $\mathsf{Open}(\mathsf{Petri}_r)$, these functors can also be construed as functors out of the structured cospan category $\mathsf{Open}(\mathsf{Petri}_r)$. Thus, structured cospans can be used to study both the dynamics and the steady states of open systems of chemical reactions.

A. Double Categories

What follows is a brief review of double categories. A more detailed exposition can be found in the work of Grandis and Paré [19, 20], and for monoidal double categories the work of Hansen and Shulman [22, 33, 34]. We use 'double category' to mean what earlier authors called a 'pseudo double category'.

A.1. Definition. *A **double category** is a weak category in* **Cat**. *More explicitly, a double category* $\mathbb{D}$ *consists of:*

- *a* **category of objects** $\mathbb{D}_0$ *and a* **category of arrows** $\mathbb{D}_1$*,*
- **source** *and* **target** *functors*
  $$S, T\colon \mathbb{D}_1 \to \mathbb{D}_0,$$
  an **identity-assigning** *functor*
  $$U\colon \mathbb{D}_0 \to \mathbb{D}_1,$$
  and a **composition** *functor*
  $$\odot\colon \mathbb{D}_1 \times_{\mathbb{D}_0} \mathbb{D}_1 \to \mathbb{D}_1$$
  where the pullback is taken over $\mathbb{D}_1 \xrightarrow{T} \mathbb{D}_0 \xleftarrow{S} \mathbb{D}_1$*, such that*
  $$S(U_A) = A = T(U_A), \quad S(M \odot N) = SN, \quad T(M \odot N) = TM,$$
- *natural isomorphisms called the* **associator**
  $$\alpha_{N,N',N''}\colon (N \odot N') \odot N'' \xrightarrow{\sim} N \odot (N' \odot N''),$$
  the **left unitor**
  $$\lambda_N\colon U_{T(N)} \odot N \xrightarrow{\sim} N,$$
  and the **right unitor**
  $$\rho_N\colon N \odot U_{S(N)} \xrightarrow{\sim} N$$
  such that $S(\alpha), S(\lambda), S(\rho), T(\alpha), T(\lambda)$ *and* $T(\rho)$ *are all identities, and such that the standard coherence axioms hold: the pentagon identity for the associator and the triangle identity for the left and right unitor.*

If α, λ and ρ are identities, we call $\mathbb{D}$ a **strict** *double category.*

Objects of $\mathbb{D}_0$ are called **objects** and morphisms in $\mathbb{D}_0$ are called **vertical 1-morphisms**. Objects of $\mathbb{D}_1$ are called **horizontal 1-cells** of $\mathbb{D}$ and morphisms in $\mathbb{D}_1$ are called **2-morphisms**. A morphism $\alpha\colon M \to N$ in $\mathbb{D}_1$ can be drawn as a square:

$$\begin{array}{ccc} a & \xrightarrow{M} & b \\ {\scriptstyle f}\big\downarrow & \Downarrow \alpha & \big\downarrow{\scriptstyle g} \\ c & \xrightarrow[N]{} & d \end{array}$$

where $f = S\alpha$ and $g = T\alpha$. If f and g are identities we call α a **globular 2-morphism**. These give rise to a bicategory:

A.2. Definition. *Let $\mathbb{D}$ be a double category. Then the* **horizontal bicategory** *of $\mathbb{D}$, denoted $H(\mathbb{D})$, is the bicategory consisting of objects, horizontal 1-cells and globular 2-morphisms of $\mathbb{D}$.*

We have maps between double categories, and also transformations between maps:

A.3. Definition. *Let $\mathbb{A}$ and $\mathbb{B}$ be double categories. A* **double functor** $\mathbb{F}\colon \mathbb{A} \to \mathbb{B}$ *consists of:*

- *functors $\mathbb{F}_0\colon \mathbb{A}_0 \to \mathbb{B}_0$ and $\mathbb{F}_1\colon \mathbb{A}_1 \to \mathbb{B}_1$ obeying the following equations:*

$$S \circ \mathbb{F}_1 = \mathbb{F}_0 \circ S, \qquad T \circ \mathbb{F}_1 = \mathbb{F}_0 \circ T,$$

- *natural isomorphisms called the* **composition comparison***:*

$$\phi(N, N')\colon \mathbb{F}_1(N) \odot \mathbb{F}_1(N') \xrightarrow{\sim} \mathbb{F}_1(N \odot N')$$

  and the **unit comparison***:*

$$\phi_A\colon U_{\mathbb{F}_0(A)} \xrightarrow{\sim} \mathbb{F}_1(U_A)$$

  whose components are globular 2-morphisms,

such that the following diagrams commmute:

- *a diagram expressing compatibility with the associator:*

$$\begin{array}{ccc} (\mathbb{F}_1(N) \odot \mathbb{F}_1(N')) \odot \mathbb{F}_1(N'') & \xrightarrow{\alpha} & \mathbb{F}_1(N) \odot (\mathbb{F}_1(N') \odot \mathbb{F}_1(N'')) \\ {\scriptstyle \phi(N,N') \odot 1}\big\downarrow & & \big\downarrow{\scriptstyle 1 \odot \phi(N',N'')} \\ \mathbb{F}_1(N \odot N') \odot \mathbb{F}_1(N'') & & \mathbb{F}_1(N) \odot \mathbb{F}_1(N' \odot N'') \\ {\scriptstyle \phi(N \odot N', N'')}\big\downarrow & & \big\downarrow{\scriptstyle \phi(N, N' \odot N'')} \\ \mathbb{F}_1((N \odot N') \odot N'') & \xrightarrow{\mathbb{F}_1(\alpha)} & \mathbb{F}_1(N \odot (N' \odot N'')) \end{array}$$

- *two diagrams expressing compatibility with the left and right unitors:*

$$\begin{array}{ccc} \mathbb{F}_1(N) \odot U_{\mathbb{F}_0(A)} & \xrightarrow{\rho_{\mathbb{F}_1(N)}} & \mathbb{F}_1(N) \\ {\scriptstyle 1\odot\phi_A}\downarrow & & \uparrow{\scriptstyle \mathbb{F}_1(\rho_N)} \\ \mathbb{F}_1(N) \odot \mathbb{F}_1(U_A) & \xrightarrow{\phi(N,U_A)} & \mathbb{F}_1(N \odot U_A) \end{array}$$

$$\begin{array}{ccc} U_{\mathbb{F}_0(B)} \odot \mathbb{F}_1(N) & \xrightarrow{\lambda_{\mathbb{F}_1(N)}} & \mathbb{F}_1(N) \\ {\scriptstyle \phi_B\odot 1}\downarrow & & \uparrow{\scriptstyle F_1(\lambda_N)} \\ \mathbb{F}_1(U_B) \odot \mathbb{F}_1(N) & \xrightarrow{\phi(U_B,N)} & \mathbb{F}_1(U_B \odot N). \end{array}$$

If the 2-morphisms $\phi(N, N')$ and ϕ_A are identities for all $N, N' \in \mathbb{A}_1$ and $A \in \mathbb{A}_0$, we say $\mathbb{F}\colon \mathbb{A} \to \mathbb{B}$ is a **strict** *double functor. If on the other hand we drop the requirement that these 2-morphisms be invertible, we call $\mathbb{F}$ a* **lax** *double functor.*

A.4. DEFINITION. *Let $\mathbb{F}\colon \mathbb{A} \to \mathbb{B}$ and $\mathbb{G}\colon \mathbb{A} \to \mathbb{B}$ be lax double functors. A* **transformation** *$\beta\colon \mathbb{F} \Rightarrow \mathbb{G}$ consists of natural transformations $\beta_0\colon \mathbb{F}_0 \Rightarrow \mathbb{G}_0$ and $\beta_1\colon \mathbb{F}_1 \Rightarrow \mathbb{G}_1$ (both usually written as β) such that*

- *$S(\beta_M) = \beta_{SM}$ and $T(\beta_M) = \beta_{TM}$ for any $M \in \mathbb{A}_1$,*
- *β preserves the composition comparison, and*
- *β preserves the unit comparison.*

Grandis and Paré define a 2-category **Dbl** of double categories, double functors, and transformations [20]. This has finite products. In any 2-category with finite products we can define a pseudomonoid [11].

A.5. DEFINITION. *A* **monoidal double category** *is a pseudomonoid in* **Dbl**. *Explicitly, a monoidal double category is a double category equipped with double functors $\otimes\colon \mathbb{D} \times \mathbb{D} \to \mathbb{D}$ and $I\colon 1 \to \mathbb{D}$ where 1 is the terminal double category, along with invertible transformations called the* **associator***:*

$$\alpha\colon \otimes \circ (1_{\mathbb{D}} \times \otimes) \Rightarrow \otimes \circ (\otimes \times 1_{\mathbb{D}}),$$

left unitor*:*

$$\ell\colon \otimes \circ (1_{\mathbb{D}} \times I) \Rightarrow 1_{\mathbb{D}},$$

and **right unitor***:*

$$r\colon \otimes \circ (I \times 1_{\mathbb{D}}) \Rightarrow 1_{\mathbb{D}}$$

satisfying the pentagon axiom and triangle axioms.

This definition neatly packages a large quantity of information. In detail, a monoidal double category $\mathbb{D}$ is a double category with:

- monoidal structures on both categories $\mathbb{D}_0$ and $\mathbb{D}_1$ (with tensor product denoted $\otimes$, associator a, left unitor ℓ and right unitor r), and
- the structure of a double functor on $\otimes$: that is, invertible globular 2-morphisms

$$\chi\colon (M_1 \otimes N_1) \odot (M_2 \otimes N_2) \xrightarrow{\sim} (M_1 \odot M_2) \otimes (N_1 \odot N_2)$$

$$\mu\colon U_{A\otimes B} \xrightarrow{\sim} (U_A \otimes U_B)$$

making these diagrams commute:

$$\begin{array}{ccc}
((M_1 \otimes N_1) \odot (M_2 \otimes N_2)) \odot (M_3 \otimes N_3) & \xrightarrow{\chi\odot 1} & ((M_1 \odot M_2) \otimes (N_1 \odot N_2)) \odot (M_3 \otimes N_3) \\
\downarrow{\scriptstyle \alpha} & & \downarrow{\scriptstyle \chi} \\
(M_1 \otimes N_1) \odot ((M_2 \otimes N_2) \odot (M_3 \otimes N_3)) & & ((M_1 \odot M_2) \odot M_3) \otimes ((N_1 \odot N_2) \odot N_3) \\
\downarrow{\scriptstyle 1\odot\chi} & & \downarrow{\scriptstyle \alpha\otimes\alpha} \\
(M_1 \otimes N_1) \odot ((M_2 \odot M_3) \otimes (N_2 \odot N_3)) & \xrightarrow{\chi} & (M_1 \odot (M_2 \odot M_3)) \otimes (N_1 \odot (N_2 \odot N_3))
\end{array}$$

$$\begin{array}{ccc}
(M \otimes N) \odot U_{C\otimes D} & \xrightarrow{1\odot\mu} & (M \otimes N) \odot (U_C \otimes U_D) \\
\downarrow{\scriptstyle \rho} & & \downarrow{\scriptstyle \chi} \\
M \otimes N & \xleftarrow{\rho\otimes\rho} & (M \odot U_C) \otimes (N \odot U_D)
\end{array}$$

$$\begin{array}{ccc}
U_{A\otimes B} \odot (M \otimes N) & \xrightarrow{\mu\odot 1} & (U_A \otimes U_B) \odot (M \otimes N) \\
\downarrow{\scriptstyle \lambda} & & \downarrow{\scriptstyle \chi} \\
M \otimes N & \xleftarrow{\lambda\otimes\lambda} & (U_A \odot M) \otimes (U_B \odot N)
\end{array}$$

We also demand the following properties:

- If I is the monoidal unit of $\mathbb{D}_0$ then U_I the monoidal unit of $\mathbb{D}_1$.
- The functors S and T are strict monoidal.
- The associator and left and right unitors for the tensor product in $\mathbb{D}$ are transformations between double functors. In other words, the following six diagrams commute:

$$\begin{array}{ccc}
((M_1 \otimes N_1) \otimes P_1) \odot ((M_2 \otimes N_2) \otimes P_2) & \xrightarrow{a\odot a} & (M_1 \otimes (N_1 \otimes P_1)) \odot (M_2 \otimes (N_2 \otimes P_2)) \\
\downarrow{\scriptstyle \chi} & & \downarrow{\scriptstyle \chi} \\
((M_1 \otimes N_1) \odot (M_2 \otimes N_2)) \otimes (P_1 \odot P_2) & & (M_1 \odot M_2) \otimes ((N_1 \otimes P_1) \odot (N_2 \otimes P_2)) \\
\downarrow{\scriptstyle \chi\otimes 1} & & \downarrow{\scriptstyle 1\otimes\chi} \\
((M_1 \odot M_2) \otimes (N_1 \odot N_2)) \otimes (P_1 \odot P_2) & \xrightarrow{a} & (M_1 \odot M_2) \otimes ((N_1 \odot N_2) \otimes (P_1 \odot P_2))
\end{array}$$

$$
\begin{array}{ccc}
U_{(A\otimes B)\otimes C} & \xrightarrow{U_a} & U_{A\otimes(B\otimes C)} \\
\downarrow{\scriptstyle \mu} & & \downarrow{\scriptstyle \mu} \\
U_{A\otimes B}\otimes U_C & & U_A\otimes U_{B\otimes C} \\
\downarrow{\scriptstyle \mu\otimes 1} & & \downarrow{\scriptstyle 1\otimes\mu} \\
(U_A\otimes U_B)\otimes U_C & \xrightarrow{a} & U_A\otimes(U_B\otimes U_C)
\end{array}
$$

$$
\begin{array}{ccc}
(U_I\otimes M)\odot(U_I\otimes N) & \xrightarrow{\chi} & (U_I\odot U_I)\otimes(M\odot N) \\
\downarrow{\scriptstyle \ell\odot\ell} & & \downarrow{\scriptstyle \lambda\otimes 1} \\
M\odot N & \xleftarrow{\ell} & U_I\otimes(M\odot N)
\end{array}
$$

$$
\begin{array}{ccc}
U_{I\otimes A} & \xrightarrow{\mu} & U_I\otimes U_A \\
& {\scriptstyle U_\ell}\searrow & \downarrow{\scriptstyle \ell} \\
& & U_A
\end{array}
$$

$$
\begin{array}{ccc}
(M\otimes U_I)\odot(N\otimes U_I) & \xrightarrow{\chi} & (M\odot N)\otimes(U_I\odot U_I) \\
\downarrow{\scriptstyle r\odot r} & & \downarrow{\scriptstyle 1\otimes\rho} \\
M\odot N & \xleftarrow{r} & (M\odot N)\otimes U_I
\end{array}
$$

$$
\begin{array}{ccc}
U_{A\otimes I} & \xrightarrow{\mu} & U_A\otimes U_I \\
& {\scriptstyle U_r}\searrow & \downarrow{\scriptstyle r} \\
& & U_A.
\end{array}
$$

A.6. Definition. *A* **braided monoidal double category** *is a braided pseudomonoid in* **Dbl**. *Explicitly, it is a monoidal double category equipped with an invertible transformation*

$$\beta\colon \otimes \Rightarrow \otimes\circ\tau$$

called the **braiding**, *where* $\tau\colon \mathbb{D}\times\mathbb{D}\to\mathbb{D}\times\mathbb{D}$ *is the twist double functor sending pairs in the object and arrow categories to the same pairs in the opposite order. The braiding is required to satisfy the usual two hexagon identities [24]. If the braiding is self-inverse we say that* $\mathbb{D}$ *is a* **symmetric monoidal double category**.

A.7. Definition. *A* **monoidal lax double functor** $\mathbb{F}\colon\mathbb{C}\to\mathbb{C}'$ *between monoidal double categories* $\mathbb{C}$ *and* $\mathbb{C}'$ *is a lax double functor* $\mathbb{F}\colon\mathbb{C}\to\mathbb{C}'$ *such that*

- $\mathbb{F}_0$ *and* $\mathbb{F}_1$ *are monoidal functors,*
- $S'\mathbb{F}_1=\mathbb{F}_0S$ *and* $T'\mathbb{F}_1=\mathbb{F}_0T$ *as monoidal functors, and*

- *the composition and unit comparisons* $\phi(N_1, N_2)\colon \mathbb{F}_1(N_1) \odot \mathbb{F}_1(N_2) \to \mathbb{F}_1(N_1 \odot N_2)$ *and* $\phi_A\colon U_{\mathbb{F}_0(A)} \to \mathbb{F}_1(U_A)$ *are monoidal natural transformations.*

The monoidal lax double functor is **braided** *if* $\mathbb{F}_0$ *and* $\mathbb{F}_1$ *are braided monoidal functors and* **symmetric** *if they are symmetric monoidal functors.*

We also have transformations between double functors:

A.8. Definition. *A* **double transformation** $\Phi\colon \mathbb{F} \Rightarrow \mathbb{G}$ *between two double functors* $\mathbb{F}\colon \mathbb{X} \to \mathbb{X}'$ *and* $\mathbb{G}\colon \mathbb{X} \to \mathbb{X}'$ *consists of two natural transformations* $\Phi_0\colon \mathbb{F}_0 \Rightarrow \mathbb{G}_0$ *and* $\Phi_1\colon \mathbb{F}_1 \Rightarrow \mathbb{G}_1$ *such that for all horizontal 1-cells* M *we have that* $S(\Phi_{1M}) = \Phi_{0S(M)}$ *and* $T(\Phi_{1M}) = \Phi_{0T(M)}$ *and for composable horizontal 1-cells* M *and* N*, we have*

$$
\begin{array}{ccccc}
\mathbb{F}(x) & \xrightarrow{\mathbb{F}(M)} & \mathbb{F}(y) & \xrightarrow{\mathbb{F}(N)} & \mathbb{F}(z) \\
{\scriptstyle 1}\downarrow & & \Downarrow \mathbb{F}_{M,N} & & \downarrow{\scriptstyle 1} \\
\mathbb{F}(x) & & \xrightarrow{\mathbb{F}(M \odot N)} & & \mathbb{F}(z) \\
{\scriptstyle \Phi_{0x}}\downarrow & & \Downarrow \Phi_{1M\odot N} & & \downarrow{\scriptstyle \Phi_{0z}} \\
\mathbb{G}(x) & & \xrightarrow{\mathbb{G}(M \odot N)} & & \mathbb{G}(z)
\end{array}
\quad = \quad
\begin{array}{ccccc}
\mathbb{F}(x) & \xrightarrow{\mathbb{F}(M)} & \mathbb{F}(y) & \xrightarrow{\mathbb{F}(N)} & \mathbb{F}(z) \\
{\scriptstyle \Phi_{0x}}\downarrow & \Downarrow \Phi_{1M} & {\scriptstyle \Phi_{0y}}\downarrow & \Downarrow \Phi_{1N} & \downarrow{\scriptstyle \Phi_{0z}} \\
\mathbb{G}(x) & \xrightarrow{\mathbb{G}(M)} & \mathbb{G}(y) & \xrightarrow{\mathbb{G}(N)} & \mathbb{G}(z) \\
{\scriptstyle 1}\downarrow & & \Downarrow \mathbb{G}_{M,N} & & \downarrow{\scriptstyle 1} \\
\mathbb{G}(x) & & \xrightarrow{\mathbb{G}(M \odot N)} & & \mathbb{G}(z)
\end{array}
$$

$$
\begin{array}{ccc}
\mathbb{F}(x) & \xrightarrow{U_{\mathbb{F}(x)}} & \mathbb{F}(x) \\
{\scriptstyle 1}\downarrow & \Downarrow \mathbb{F}_U & \downarrow{\scriptstyle 1} \\
\mathbb{F}(x) & \xrightarrow{\mathbb{F}(U_x)} & \mathbb{F}(x) \\
{\scriptstyle \Phi_{0x}}\downarrow & \Downarrow \Phi_{1U_x} & \downarrow{\scriptstyle \Phi_{0x}} \\
\mathbb{G}(x) & \xrightarrow{\mathbb{G}(U_x)} & \mathbb{G}(x)
\end{array}
\quad = \quad
\begin{array}{ccc}
\mathbb{F}(x) & \xrightarrow{U_{\mathbb{F}(x)}} & \mathbb{F}(x) \\
{\scriptstyle \Phi_{0x}}\downarrow & \Downarrow U_{\Phi_{0x}} & \downarrow{\scriptstyle \Phi_{0x}} \\
\mathbb{G}(x) & \xrightarrow{U_{\mathbb{G}(x)}} & \mathbb{G}(x) \\
{\scriptstyle 1}\downarrow & \Downarrow \mathbb{G}_U & \downarrow{\scriptstyle 1} \\
\mathbb{G}(x) & \xrightarrow{\mathbb{G}(U_x)} & \mathbb{G}(x)
\end{array}
$$

We call Φ_0 *the* **object component** *and* Φ_1 *the* **arrow component** *of the double transformation* Φ.

One can also define monoidal, braided monoidal and symmetric monoidal double transformations, but since we do not use these, we refer the reader to Hansen and Shulman for the details [22, Definition 2.15].

References

[1] J. C. Baez, K. Courser and C. Vasilakopoulou, Structured versus decorated cospans. Manuscript in preparation.

[2] J. C. Baez, B. Coya and F. Rebro, Props in circuit theory, *Theory Appl. Categ.* **33** (2018), 727–783. Available as arXiv:1707.08321.

[3] J. C. Baez and B. Fong, A compositional framework for passive linear networks, *Theory Appl. Categ.* **33** (2018), 1158–1222. Available as arXiv:1504.05625.

[4] J. C. Baez, B. Fong and B. Pollard, A compositional framework for Markov processes, *J. Math. Phys.* **57** (2016), 033301. Available as arXiv:1508.06448.

[5] J. C. Baez and J. Master, Open Petri nets, *Math. Struct. Comput. Sci.* **30** (2020), 314–341. Available as arXiv:1808.05415.

[6] J. C. Baez and B. Pollard, A compositional framework for chemical reaction networks, *Rev. Math. Phys.* **29** (2017), 1750028.

[7] R. M. Burstall and D. E. Rydeheard, *Computational Category Theory*, Prentice Hall, Englewood Cliffs, 1988.

[8] D. Cicala, *Rewriting Structured Cospans: A Syntax For Open Systems*, Ph.D. thesis, Department of Mathematics, U. C. Riverside, 2019. Available at arXiv:1906.05443.

[9] K. Courser, A bicategory of decorated cospans, *Theory Appl. Categ.* **32** (2017), 995–1027. Available as arXiv:1605.08100.

[10] K. Courser, *Open Systems: a Double Categorical Perspective*, Ph.D. thesis, Department of Mathematics, U. C. Riverside, 2020. Available as arxiv:2008.02394.

[11] B. Day and R. Street, Monoidal bicategories and Hopf algebroids, *Adv. Math.* **129** (1997), 99–157.

[12] C. Ehresmann, *Catégories et Structures*, Dunod, Paris, 1965.

[13] B. Fong, Decorated cospans, *Theory Appl. Categ.* **30** (2015), 1096–1120. Available as arXiv:1502.00872.

[14] B. Fong, *The Algebra of Open and Interconnected Systems*, Ph.D. thesis, Computer Science Department, University of Oxford, 2016. Available as arXiv:1609.05382.

[15] B. Fong, Decorated corelations, *Theory Appl. Categ.* **33** (2018), 608–643. Available as arXiv:1703.09888.

[16] B. Fong and M. Sarazola, A recipe for black box functors, *Theory Appl. Categ.* **35** (2020), 979–1011. Available as arXiv:1812.03601.

[17] B. Fong and D. Spivak, Hypergraph categories. Available as arXiv:1806.08304.

[18] C. Girault and R. Valk, *Petri Nets for Systems Engineering: a Guide to Modeling, Verification, and Applications*, Springer, Berlin, 2013.

[19] M. Grandis and R. Paré, Limits in double categories, *Cah. Top. Géom. Differ. Catég.* **40** (1999), 162–220.

[20] M. Grandis and R. Paré, Adjoints for double categories, *Cah. Top. Géom. Differ. Catég.* **45** (2004), 193–240.

[21] P. J. Haas, *Stochastic Petri Nets: Modelling, Stability, Simulation*, Springer, Berlin, 2002.

[22] L. W. Hansen and M. Shulman, Constructing symmetric monoidal bicategories functorially. Available as arXiv:1910.09240.

[23] F. Horn and R. Jackson, General mass action kinetics, *Arch. Ration. Mech. An.* **47** (1972), 81–116.

[24] S. Mac Lane, Natural associativity and commutativity, *Rice Univ. Stud.* **49** (1963), 28–46.

[25] M. A. Marsan, G. Balbo, G. Conte, S. Donatelli and G. Franceschinis, *Modelling with Generalized Stochastic Petri Nets*, Wiley, New York, 1995.

[26] N. Martins-Ferreira, Pseudo-categories, *J. Homotopy Relat. Struct.* **1** (2006), 47–78. Available as arXiv:0604549.

[27] J. Master, Petri nets based on Lawvere theories. Available as arXiv:1904.09091.

[28] J. Meseguer and U. Montanari, Petri nets are monoids, *Inform. and Comput.* **88** (1990), 105–155.

[29] J. Moeller and C. Vasilakopoulou, Monoidal Grothendieck construction, *Theory Appl. Categ.* **35** (2020), 1159–1207. Available as arXiv:1809.00727.

[30] J. L. Peterson, *Petri Net Theory and the Modeling of Systems*, Prentice–Hall, New Jersey, 1981.

[31] S. Niefield, Span, cospan, and other double categories, *Theory Appl. Categ.* **26** (2012), 729–742. Available as arXiv:1201.3789.

[32] R. Rosebrugh, N. Sabadini and R. F. C. Walters, Generic commutative separable algebras and cospans of graphs, *Theory Appl. Categ.* **15** (2005), 164–177.

[33] M. Shulman, Framed bicategories and monoidal fibrations, *Theory Appl. Categ.* **20** (2008), 650–738. Available as arXiv:0706.1286.

[34] M. Shulman, Constructing symmetric monoidal bicategories. Available as arXiv:1004.0993.

[35] M. Stay, Compact closed bicategories, *Theory Appl. Categ.* **31** (2016), 755–798. Available as arXiv:1301.1053.

[36] D. Verdon, Coherence for braided and symmetric pseudomonoids. Available as arXiv:1705.09354.

Department of Mathematics, University of California, Riverside, CA 92521, USA
Email: `baez@math.ucr.edu`
`courser@math.ucr.edu`

This article may be accessed at `http://www.tac.mta.ca/tac/`

THEORY AND APPLICATIONS OF CATEGORIES will disseminate articles that significantly advance the study of categorical algebra or methods, or that make significant new contributions to mathematical science using categorical methods. The scope of the journal includes: all areas of pure category theory, including higher dimensional categories; applications of category theory to algebra, geometry and topology and other areas of mathematics; applications of category theory to computer science, physics and other mathematical sciences; contributions to scientific knowledge that make use of categorical methods.
Articles appearing in the journal have been carefully and critically refereed under the responsibility of members of the Editorial Board. Only papers judged to be both significant and excellent are accepted for publication.

SUBSCRIPTION INFORMATION Individual subscribers receive abstracts of articles by e-mail as they are published. To subscribe, send e-mail to tac@mta.ca including a full name and postal address. Full text of the journal is freely available at http://www.tac.mta.ca/tac/.

INFORMATION FOR AUTHORS LaTeX2e is required. Articles may be submitted in PDF by email directly to a Transmitting Editor following the author instructions at
http://www.tac.mta.ca/tac/authinfo.html.

MANAGING EDITOR. Geoff Cruttwell, Mount Allison University: gcruttwell@mta.ca

TEXNICAL EDITOR. Michael Barr, McGill University: michael.barr@mcgill.ca

ASSISTANT TEX EDITOR. Gavin Seal, Ecole Polytechnique Fédérale de Lausanne:
gavin_seal@fastmail.fm

TRANSMITTING EDITORS.
Clemens Berger, Université de Nice-Sophia Antipolis: cberger@math.unice.fr
Julie Bergner, University of Virginia: jeb2md (at) virginia.edu
Richard Blute, Université d' Ottawa: rblute@uottawa.ca
Gabriella Böhm, Wigner Research Centre for Physics: bohm.gabriella (at) wigner.mta.hu
Valeria de Paiva: Nuance Communications Inc: valeria.depaiva@gmail.com
Richard Garner, Macquarie University: richard.garner@mq.edu.au
Ezra Getzler, Northwestern University: getzler (at) northwestern(dot)edu
Kathryn Hess, Ecole Polytechnique Fédérale de Lausanne: kathryn.hess@epfl.ch
Dirk Hofmann, Universidade de Aveiro: dirk@ua.pt
Pieter Hofstra, Université d' Ottawa: phofstra (at) uottawa.ca
Anders Kock, University of Aarhus: kock@math.au.dk
Joachim Kock, Universitat Autònoma de Barcelona: kock (at) mat.uab.cat
Stephen Lack, Macquarie University: steve.lack@mq.edu.au
Tom Leinster, University of Edinburgh: Tom.Leinster@ed.ac.uk
Matias Menni, Conicet and Universidad Nacional de La Plata, Argentina: matias.menni@gmail.com
Ieke Moerdijk, Utrecht University: i.moerdijk@uu.nl
Susan Niefield, Union College: niefiels@union.edu
Kate Ponto, University of Kentucky: kate.ponto (at) uky.edu
Robert Rosebrugh, Mount Allison University: rrosebrugh@mta.ca
Jiri Rosicky, Masaryk University: rosicky@math.muni.cz
Giuseppe Rosolini, Università di Genova: rosolini@disi.unige.it
Alex Simpson, University of Ljubljana: Alex.Simpson@fmf.uni-lj.si
James Stasheff, University of North Carolina: jds@math.upenn.edu
Ross Street, Macquarie University: ross.street@mq.edu.au
Tim Van der Linden, Université catholique de Louvain: tim.vanderlinden@uclouvain.be