STRING DIAGRAMS FOR PHYSICAL DUOIDAL CATEGORIES
TECHNICAL REPORT
June 28, 2024

MARIO ROMÁN

Department of Computer Science, University of Oxford.

Abstract. We introduce string diagrams for physical duoidal categories (normal symmetric duoidal categories): they consist of string diagrams with wires forming a zigzag-free partial order and order-preserving nodes whose inputs and outputs form intervals. We derive string diagrams for symmetric monoidal categories as a particular case.

1. Introduction

Physical duoidal categories (or normal $\otimes$-symmetric duoidal categories) have been applied to the study of process dependencies [SS22, EHR24]. We take this intuition seriously to develop a string diagrammatic calculus of physical duoidal categories. String diagrams for physical duoidal categories particularize both to the hypergraph-based diagrams of symmetric monoidal categories [JS91] and to string diagrams for, at least, some spacial monoidal categories [Sel10]: essentially, they are string diagrams where wires form a poset and nodes must take intervals of the poset as inputs and outputs.

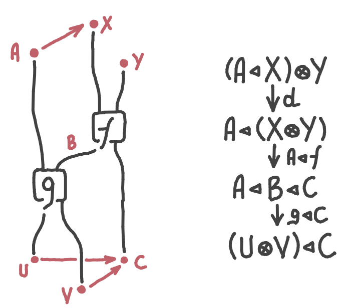


Figure 1. Example translation: from string diagrams to text.

Remark 1.1. Assume we want to compose two generators, $f\colon X \otimes Y \to C$ and $g\colon A \triangleleft B \to U \otimes V$ into a morphism $(A \triangleleft X) \otimes Y \to (U \otimes V) \triangleleft C$. We may first use

Key words and phrases. Physical duoidal category, normal duoidal category, string diagrams.
Originally published on June 18, 2024.

the physical duoidal distributors to get a map $(A \triangleleft X) \otimes Y \to A \triangleleft (X \otimes Y)$, and then apply $(A \triangleleft f) \colon A \triangleleft (X \otimes Y) \to A \triangleleft B \triangleleft C$ and $(g \triangleleft C) \colon A \triangleleft B \triangleleft C \to (U \otimes V) \triangleleft C$.

Our string diagram (in Figure 1) starts with three wires: a, x and y, assuming $a \leq x$. The subposet $\{x, y\}$ forms an interval, so we can apply f to get the wires b and c, which now must satisfy $b \leq c$ (because of the type of f); while also satisfying $a \leq b$ and $a \leq c$ (because of $a \leq x$). Again, $\{a, b\}$ form an interval, and we can apply g to get the wires u and v, which are independent but both below c, leaving us with $u \leq c$ and $v \leq c$. We can either choose to keep track of these dependencies in our head or write them explicitly in the string diagrams: if we follow the typing rules, we need to declare no explicit dependencies, except for those of the input and output types.

The main result of this paper (Theorem 7.21) is the construction of the free physical duoidal category over a physical duoidal signature; the morphisms of this free physical duoidal category consist of string diagrams with strings ordered by a poset without zigzags. In other words, we construct an adjunction between the category of strict physical duoidal categories, **PhyDuo**, and the category of physical duoidal signatures, **PhySig**.

Physical duoidal categories, because of their relation to posets, have multiple potential applications to causality, concurrency, and formal category theory. However, without an appropriate syntax, reasoning with them can be tedious and unenlightening. The string diagrammatic syntax may dramatically simplify proofs regarding duoidal structures.

2. Physical Duoidal Categories

Physical duoidal category is the term Shapiro and Spivak [SS22] give to normal $\otimes$-symmetric duoidal categories [AM10, GF16].

Definition 2.1 (Strict physical duoidal category)**.** A *strict physical duoidal category* is a category with a strict monoidal structure and a strict symmetric monoidal structure sharing the same unit, $(\mathbb{V}, \triangleleft, \otimes, N)$, and such that the first monoidal structure distributes over the second; that is, there exist maps

$$\partial_{X,Y,Z,W} \colon (X \triangleleft Z) \otimes (Y \triangleleft W) \to (X \otimes Y) \triangleleft (Z \otimes W);$$
$$\sigma_{X,Y} \colon X \otimes Y \to Y \otimes X.$$

Strict physical duoidal categories are defined to be coherent structures, meaning that any formally distinctly typed equation of morphisms on the free strict physical duoidal category holds true.

Definition 2.2 (Strict physical duoidal functor)**.** A *strict physical duoidal functor* between two strict physical duoidal categories,

$$(\mathbb{V}, \otimes_V, \triangleleft_V, N_V, \partial_V, \sigma_V) \text{ and } (\mathbb{W}, \otimes_W, \triangleleft_W, N_W, \partial_W, \sigma_W),$$

is a functor that is strict symmetric monoidal for the parallel monoidal structures, $(\otimes_V)$ and $(\otimes_W)$, and strict monoidal for the sequential monoidal structures, $(\triangleleft_V)$ and $(\otimes_W)$; and that, moreover, strictly preserves the structure maps, $F(\partial_V) = \partial_W$ and $F(\sigma_V) = \sigma_W$.

Proposition 2.3. *Strict physical duoidal categories and strict physical duoidal functors between them form a category,* **PhyDuo***.*

Proof. See Appendix, Proposition A.1. □

3. Duoidal expressions

Let us start by studying duoidal expressions – the objects of the free physical duoidal category over a set of objects. These form a "boring" construction of the objects of the free physical duoidal category: if we care about string diagrams, the interesting bit of mathematics will be in the construction of the morphisms of the free physical duoidal category.

Definition 3.1 (Duoidal expression)**.** The set of *duoidal expressions*, $\mathsf{expr}(A)$, over some set of objects A is inductively built as

- the empty expression, N;
- a singleton a, for each $a \in A$;
- a sequence of expressions, $E = E_1 \triangleleft ... \triangleleft E_n$, each not empty nor sequencing;
- a tensoring of expressions, $E = E_1 \otimes ... \otimes E_n$, each not empty nor tensoring.

In other words, by definition, we forbid unncessary nested expressions, such as $(A \otimes B) \otimes C$; we only allow the correspoding reduced expressions, such as $A \otimes B \otimes C$. This allows us to avoid redundancy and construct a strict physical duoidal category; however, this also forces us to define non-trivial operations for substitution, parallel, and tensoring compositions (see Proposition 3.2).

Proposition 3.2 (Sequencing and tensoring duoidal expressions)**.** *There exist two binary operations in duoidal expressions,* $(\triangleleft_e) \colon \mathsf{expr}(A) \times \mathsf{expr}(A) \to \mathsf{expr}(A)$ *and* $(\otimes_e) \colon \mathsf{expr}(A) \times \mathsf{expr}(A) \to \mathsf{expr}(A)$*, defined by sequencing and tensoring after reducing by associativity and unitality.*

Proposition 3.3. *Duoidal expressions induce a monad* $\mathsf{expr} \colon \mathbf{Set} \to \mathbf{Set}$*. The objects of any strict physical duoidal category form an algebra for the monad,*

$$[\![\bullet]\!] \colon \mathsf{expr}(\mathbb{V}_{obj}) \to \mathbb{V}_{obj}.$$

Definition 3.4 (Equality up to symmetry)**.** Two duoidal expressions are *equal up to $\otimes$-symmetry* (or, simply, *up to symmetry*) if they are related by the following inductively defined relation $(\approx) \subseteq \mathsf{expr}(A) \times \mathsf{expr}(A)$.

- $N \approx N$;
- $a \approx a$, for each $a \in A$;
- $E_1 \triangleleft ... \triangleleft E_n \approx E'_1 \triangleleft ... \triangleleft E'_n$, for $E_i \approx E'_i$;
- $E_1 \otimes ... \otimes E_n \approx E'_{\sigma(1)} \otimes ... \otimes E'_{\sigma(n)}$, for $E_i \approx E'_i$ permuted by $\sigma \in \mathrm{Perm}(n)$.

4. Posets and Zetless Posets

Let us give some basic definitions on posets. The rest of the text will revolve around those posets that contain no zigzags (the *zetless posets*), but most operations and definitions apply to arbitrary posets.

Definition 4.1 (Poset)**.** A (finite) *poset* is a finite set endowed with a reflexive, antisymmetric, and transitive relation, $(\leq)$, on its elements.

Remark 4.2. Given any relation on a finite set, $(\to)$, its reflexive, antisymmetric, and transitive closure forms a poset. When we draw posets, we use arrows $(\to)$ instead of the less-or-equal-than symbols $(\leq)$ to represent the generators of the poset: the poset is the closure under transitivity and reflexivity (quotiented by antisymmetry if necessary) of these generators.

Definition 4.3 (Incomparability)**.** Two elements on a poset, $x, y \in P$, are *incomparable*, $x \parallel y$, if neither $x \leq y$ nor $y \leq x$ are true.

Definition 4.4 (Incomparable connectedness)**.** Two elements, $x, y \in P$, are *incomparable connected* if there exists a path of pairwise incomparable elements between them, $x \parallel p_1$, $p_1 \parallel p_2$, ..., $p_n \parallel y$. An *incomparable connected component* is a full subposet such that all objects are incomparable connected between them.

Definition 4.5 (Sequencing of posets)**.** The *sequencing* of two posets, P and Q, is the poset that contains the disjoint union of the objects of both posets, all the edges of both P and Q, and an edge from every element of P to every element of Q. That is,

$$P \triangleleft Q = (P_{obj} + Q_{obj}; \leq_P + \leq_Q + \{p \leq q\}_{p \in P, q \in Q}).$$

Definition 4.6 (Tensoring of posets)**.** The tensoring of two posets, P and Q, is the poset containing the disjoint union of objects from both, and the disjoint union of edges from both.

$$P \otimes Q = (P_{obj} + Q_{obj}; \leq_P + \leq_Q).$$

4.1. **Zetless posets.** Zetless posets are posets without zigzags inside them. We note that both the empty poset and the singleton poset are zetless posets; the tensoring and sequencing of zetless posets also form zetless posets.

Definition 4.7 (Zetless poset)**.** A *zetless poset* is a poset that does not admit a fully faithful embedding of the Z-poset, $x \to u \leftarrow y \to v$.

Proposition 4.8. *The sequencing and tensoring of two zetless posets is again a zetless poset.*

Proof. See Appendix, Proposition A.13. □

Proposition 4.9. *In a zetless poset, any two connected elements must be connected by either a span or a cospan. That is, if there is a path between two elements, $x_0 \to x_1 \leftarrow x_2 \to ... \leftarrow x_n$, there must exist either a cospan between them, $x_0 \to u \leftarrow x_n$, or a span between them, $x_0 \leftarrow v \to x_n$.*

Proof. See Appendix, Proposition A.2. □

4.2. **Prime posets.** Primality for sequencing and primality for tensoring can be characterized by more familiar notions: connectedness and incomparable connectedness. We comment on this characterization, as it will simplify the rest of our proofs.

Definition 4.10 (Prime posets)**.** A non-empty poset P is parallel prime (or $\otimes$-prime) if $P = Q_1 \otimes ... \otimes Q_n$ with $Q_i \neq N$ implies $n = 1$ and $Q_1 = P$; it is sequential prime (or $\triangleleft$-prime) if $P = Q_1 \triangleleft ... \triangleleft Q_n$ with $Q_i \neq N$ implies $n = 1$ and $Q_1 = P$.

Proposition 4.11. *A poset is $\otimes$-prime if and only if it is connected. A poset P is $\triangleleft$-prime if and only if it is incomparable connected. Any $\triangleleft$-prime and $\otimes$-prime poset must be a singleton poset.*

Proof. See Appendix, Propositions A.9 to A.11. □

5. Posets versus Duoidal Expressions

Definition 5.1 (Typed zetless poset)**.** A *zetless poset typed by a set T* is a zetless poset structure whose set is finite, $\{1, ..., n\}$, endowed with a type-labelling function $t\colon \{1, ..., n\} \to T$, and considered up to type-preserving bijection. The set of zetless posets labelled by a set A is written as $\mathsf{zetless}(A)$.

For instance, there are 7 posets of cardinality 2 typed over $\{A, B\}$ (Figure 2).

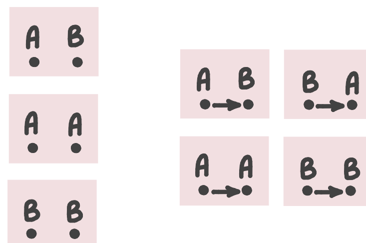


FIGURE 2. Zetless posets typed over the set $\{A, B\}$ with cardinality 2.

Proposition 5.2 (Posets versus duoidal expressions)**.** *Zetless posets labelled over a set are in correspondence with duoidal expressions on that set, up to symmetries of the tensored components,*

$$\mathsf{zetless}(A) \cong \mathsf{expr}(A)/(\approx).$$

In particular, there is a surjective function $\mathsf{encode}\colon \mathsf{expr}(A) \to \mathsf{zetless}(A)$*.*

Proof. See Appendix, Proposition A.3. □

Remark 5.3. In other words, zetless posets are a symmetry-aware encoding of duoidal expressions. Every time we write a duoidal expression we must choose an order in which we write the poset; however, $P \otimes Q$ and $Q \otimes P$ represent the same poset.

Proposition 5.4 (Shapiro and Spivak [SS22])**.** *The existence of an inclusion of zetless posets corresponds to the existence of a structure map between their corresponding duoidal expressions in a physical duoidal category.*

Proof. See Appendix, Proposition A.4. □

Definition 5.5. The category of *zetless maps* over a set, $\mathsf{Zetless}(A)$, has, as objects, the duoidal expressions. The morphisms between two duoidal expressions are type-preserving bijective-on-objects inclusions between their corresponding zetless posets.

Proposition 5.6. *The construction of the category of zetless maps over a set induces a functor* $\mathsf{Zetless}\colon \mathbf{Set} \to \mathbf{PhyDuo}$*. The functor carries a function $f\colon A \to B$ into the functor that changes the types of the duoidal expressions on objects and transports the bijective-on-objects inclusions.*

Theorem 5.7. *Zetless maps construct the free physical duoidal category over a set of objects. In other words, the functor* $\mathsf{Zetless}\colon \mathbf{Set} \to \mathbf{PhyDuo}$ *is left adjoint to the forgetful functor that picks the objects of a physical duoidal category,* $\mathsf{Obj}\colon \mathbf{PhyDuo} \to \mathbf{Set}$.

Proof. See Appendix, Theorem A.5. □

Corollary 5.8. *Unlabelled duoidal expressions and bijective-on-objects inclusions of unlabelled zetless posets form the free physical duoidal category over the singleton set.*

6. Intervals

This first section has recalled the construction of the free physical duoidal category on a set as a category of zetless posets and bijective-on-object inclusions between them. String diagrams will need more than that: instead of constructing the free physical duoidal category *over a set*, we must construct it over an arbitrary signature of physical duoidal operations. This is what we will pursue for the rest of this text.

For this purpose, it becomes relevant to analyze the notion of subterm, and to characterize it in terms of zetless posets. This is what we do now.

Definition 6.1 (Interval)**.** An *interval* is a subset of a poset that is closed under intermediate elements. That is, a subset of a poset $I \subseteq P$, is an interval if any element $y \in P$ in between two elements of the subset, $x_0 \leq y \leq x_1$ for $x_0, x_1 \in I$, belongs to the interval, $y \in I$.

Definition 6.2 (Substitution)**.** The *substitution* of a poset P into a poset Q at an element $x \in Q$ is a poset, $Q[x \backslash P]$, containing the objects of P and Q but excluding $x \in Q$, and containing all the edges of P, of Q, and from and to x.

$$Q[x\backslash P] = (P_{obj} + Q_{obj} - \{x\}; \leq_{Q-\{x\}} + \leq_P + \{q \leq p \mid q \in Q, q \leq x\} + \{p \leq q \mid q \in Q, x \leq q\}).$$

Proposition 6.3 (Posets form an operad)**.** *Substituting two posets in two different elements yields the same result independently of the order of substitution,*

$$Q[x\backslash P_1][y\backslash P_2] = Q[y\backslash P_2][x\backslash P_1].$$

Substitution is associative, meaning that, substituting into a poset used for substitution at some element $y \in P_1$ is the same as substituting into that element on the resulting poset,

$$Q[x\backslash P_1][y\backslash P_2] = Q[x\backslash P_1[y\backslash P_2]].$$

Zetless posets with substitution form an operad.

Definition 6.4 (Bracketed)**.** A full subposet $P \subseteq Q$ is *bracketed* if any element $q \in Q$ above an element of the subposet, $p_0 \leq q$ for $p_0 \in P$, is above all elements of the poset, $p \leq q$ for any $p \in P$; and if any element $q \in Q$ below an element of the subposet, $q \leq p_0$ for $p_0 \in P$, is above all elements of the poset, $q \leq p$ for any $p \in P$.

Proposition 6.5 (Bracketed only if substituted)**.** *A poset R arises as a substitution $R = Q[x\backslash P]$ of any of its full subposets, $P \subseteq R$, if and only if it is bracketed.*

Proof. See Appendix, Proposition A.6. □

Proposition 6.6 (Bracketed implies interval)**.** *Any bracketed poset is an interval.*

Proposition 6.7 (Interval if and only if bracketed in a saturation)**.** *A subset of a zetless poset is an interval if and only if it appears as a bracketed poset in some saturation of the poset.*

Proof. See Appendix, Proposition A.7. □

7. String Diagrams

7.1. **Signatures.** The signature for a string diagram for physical duoidal categories consists of a set of basic types and some generators that are typed by duoidal expressions on both the input and the outputs.

Definition 7.1 (Physical duoidal signature)**.** A *physical duoidal signature* $\mathcal{G}$ is given by a set of basic types, $\mathcal{G}_t$, and a set of generators, $\mathcal{G}(E_i; E_o)$, for each two duoidal expressions over the types, $E_i, E_o \in \mathsf{expr}(\mathcal{G}_t)$.

We write $\mathcal{G}$ for the set of all generators: there exist functions $\mathsf{source}\colon \mathcal{G} \to \mathsf{expr}(\mathcal{G}_t)$ and $\mathsf{target}\colon \mathcal{G} \to \mathsf{expr}(\mathcal{G}_t)$ picking the source and target duoidal expressions of the generator.

Definition 7.2 (Homomorphism of physical duoidal signatures)**.** Let $\mathcal{G}$ and $\mathcal{H}$ be two physical duoidal signatures. A *homomorphism of physical duoidal signatures* (or, more succintly, a *signature homomorphism*), $f\colon \mathcal{G} \to \mathcal{H}$, consists of a function between basic types, $f_t\colon \mathcal{G}_t \to \mathcal{H}_t$, and a function of generators,

$$f\colon \mathcal{G}(U; V) \to \mathcal{H}(\mathsf{expr}(f_t)(U); \mathsf{expr}(f_t)(V)).$$

Proposition 7.3 (Category of physical duoidal signatures)**.** *Physical duoidal signatures and signature homomorphisms between them form a category,* **PhySig**.

Proposition 7.4 (Forgetful functor to physical duoidal signatures)**.** *There is a forgetful functor from the category of strict physical duoidal categories to the category of physical duoidal signatures,*

$$\mathsf{Forget}\colon \mathbf{PhyDuo} \to \mathbf{PhySig}.$$

The forgetful functor picks all of the objects of a category as basic types, $\mathsf{Forget}(\mathbb{V})_t = \mathbb{V}_{obj}$*; it picks all of the morphisms of a given type as generators,*

$$\mathsf{Forget}(\mathbb{V})(U; V) = \mathbb{V}([\![U]\!]; [\![V]\!]).$$

Proof. See Appendix, Proposition A.12. □

7.2. **Hypergraphs and String Diagrams.** The combinatorial structure behind physical string diagrams is that of linear and acyclic hypergraphs: the hypergraph contains wires and nodes, and each wire connects the output of exactly one node to the input of exactly one node. The main result of this section will show that linear acyclic hypergraphs whose wires are ordered following certain rules form the free physical duoidal category over a physical duoidal signature.

Of course, there exists a more obvious construction of the free physical duoidal category: we can inductively write all possible terms arising from composition and tensoring and then quotient by the appropriate equations. However, the interesting bit of mathematics is to show that physical string diagrams are just as good as terms for that purpose.

Finally, note that the structure we can extract from a diagram is the connectivity – the hypergraph – and not how this connectivity has been drawn. The situation is similar (if not identical) to the hypergraph representation for strict symmetric monoidal categories [BHP+19].

Definition 7.5 (Hypergraph)**.** A *hypergraph* H consists of a finite set of wires (or hypervertices), $\mathsf{wires}(H)$, and a finite set of nodes (or posetal edges), $\mathsf{nodes}(H)$, with a source and target functions,

$$\mathsf{input}\colon \mathsf{nodes}(H) \to \mathrm{List}(\mathsf{wires}(H)) \quad \text{and} \quad \mathsf{output}\colon \mathsf{nodes}(H) \to \mathrm{List}(\mathsf{wires}(H)).$$

Hypergraphs are considered equal up to source-and-target-preserving isomorphism of their sets of wires and nodes.

Definition 7.6 (Wire-linear hypergraph)**.** A hypergraph H is *wire-linear* if every wire $w \in H_W$ appears exactly once as a source and once as a target. That is, there uniquely exist two nodes $s^*(w)$ and $t^*(w)$ such that $t(s^*(w)) = \Gamma, w, \Gamma'$ and $s(t^*(w)) = \Delta, w, \Delta'$, with $w \notin \Gamma, \Gamma', \Delta, \Delta'$. These induce functions $s^*\colon H_W \to H_N$ and $t^*\colon H_W \to H_N$.

Definition 7.7 (Acyclic hypergraph)**.** A hypergraph H is *acyclic* if it contains no closed paths of non-zero length. A path of length $n \in \mathbb{N}$ is a sequence of wires $w_0, ..., w_n \in H_W$ such that the target of a wire is the source of the next wire, $t^*(w_i) = s^*(w_{i+1})$ for each $i = 0, ..., n-1$. A path is closed whenever $w_0 = w_n$.

Definition 7.8 (List of types)**.** The *list of types* of a duoidal expression is defined inductively to match the list of types we would obtain traversing the expression from left to right. Explicitly, it is defined inductively as

- $\mathsf{listType}(N) = []$;
- $\mathsf{listType}(a) = [a]$;
- $\mathsf{listType}(E_1 \triangleleft ... \triangleleft E_n) = \mathsf{listType}(E_1), ..., \mathsf{listType}(E_n)$;
- $\mathsf{listType}(E_1 \otimes ... \otimes E_n) = \mathsf{listType}(E_1), ..., \mathsf{listType}(E_n)$.

Definition 7.9 (Physical hypergraph)**.** A *physical hypergraph* H labelled over a physical duoidal signature $\mathcal{G}$ is a wire-linear acyclic hypergraph where wires form a poset,

$$(\sqsubseteq) \subseteq \mathsf{wires}(H) \times \mathsf{wires}(H),$$

and the following properties hold.

(1) Wires, $w \in \mathsf{wires}(H)$, are labelled by basic types, $\mathsf{label}(w) \in \mathcal{G}_{obj}$.
(2) Nodes, $n \in \mathsf{nodes}(H)$, are labelled by generators, $\mathsf{label}(n) \in \mathcal{G}$.
(3) For each node $n \in H_N$, input wires are typed by the input expression; output wires are typed by the output expression,

$$\mathsf{label}(\mathsf{input}(n)) = \mathsf{listType}(\mathsf{source}(\mathsf{label}(n))),$$
$$\mathsf{label}(\mathsf{output}(n)) = \mathsf{listType}(\mathsf{target}(\mathsf{label}(n))).$$

(4) For each node $n \in H_N$, the induced poset over the input wires, $\mathsf{input}(n)$, forms an interval. There must exist an identity-on-objects inclusion of the input wires into the poset of inputs of the generator,

$$\mathsf{input}(n) \stackrel{\sim}{\to} \mathsf{source}(\mathsf{label}(n)).$$

(5) The induced poset over the output wires coincides with the target zetless poset of the generator, $\mathsf{output}(n) = \mathsf{target}(\mathsf{label}(n))$.
(6) Input wires induce order to the output wires. That is, for any node $n \in \mathsf{nodes}(H)$ and any of its outputs $o \in \mathsf{output}(n)$, any wire is above it $x \sqsubseteq o$ if and only if there exists an input $i \in \mathsf{input}(n)$ above the wire, $x \sqsubseteq i$; any output is below a wire, $o \sqsubseteq x$, if and only if there exists an input $i \in \mathsf{input}(n)$ below the wire, $i \sqsubseteq x$.

Note that this last condition never breaks asymmetry because the input is an interval: if there were wire below and above the input, then it should belong to the input.

Definition 7.10 (Physical string diagram)**.** A *physical string diagram*, $\alpha\colon E_1 \to E_2$, labelled on a physical duoidal signature $\mathcal{G}$ and from a duoidal expression $E_1 \in \mathsf{expr}(\mathcal{G}_t)$ to a duoidal expression $E_2 \in \mathsf{expr}(\mathcal{G}_t)$, is a physical hypergraph, α, endowed with special unlabelled input and output nodes, $i, o \in \mathsf{nodes}(\alpha)$, satisfying $\mathsf{label}(\mathsf{output}(i)) = E_1$ and $\mathsf{label}(\mathsf{input}(o)) = E_2$, while $\mathsf{label}(\mathsf{input}(i)) = \mathsf{label}(\mathsf{output}(o)) = N$.

Remark 7.11. Every node is labelled by a duoidal expression for its input and its output. The type list of that expression fixes a linear ordering to the inputs and the outputs. When we draw, the inputs and outputs are ordered from left to right, and thus they are identified.

Remark 7.12. We decide that the wires will always keep the minimal possible poset that is compatible with the diagram. This forces an asymmetry between rules (4) and (5); however, this is merely a convention: we could have also decided that the wires always keep the maximal possible poset that is compatible with the diagram.

Theorem 7.13 (Diagrams form a physical duoidal category)**.** *Physical duoidal string diagrams over a physical duoidal signature, $\mathcal{G}$ form a physical duoidal category,* $\mathsf{PhyString}(\mathcal{G})$*.*

Objects are duoidal expressions over the types of the physical duoidal signature; morphisms are hypergraphs with input and output given by the zetless posets corresponding to the duoidal expressions.

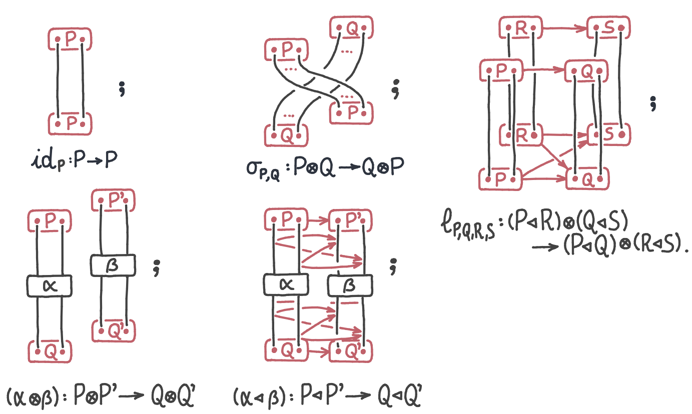


FIGURE 3. Physical string diagrams form a physical duoidal category.

Composing two string diagrams, α and β, concatenates them, so that the output wires of α and the input wires of β get merged into single wires. Parallel tensoring juxtaposes diagrams; sequential tensoring juxtaposes but also links every wire from the first diagram to the second diagram (see Figure 3).

Proof. See Appendix, Theorem A.14. □

Proposition 7.14 (String diagrams functor)**.** *The construction of physical duoidal string diagrams over a physical duoidal signature extends to a functor*

$$\mathsf{PhyString}\colon \mathbf{PhySig} \to \mathbf{PhyDuo}.$$

Proof. See Appendix, Proposition A.8. □

7.3. **Freeness.** We conclude this text by proving that string diagrams over a physical duoidal category form the free physical duoidal category over a physical duoidal signature. The idea of the proof is that every hypergraph can be decomposed in multiple *atomic hypergraphs*, consisting of a single node and wires. The interpretation of each one of these atomic hypergraphs is completely determined; thus, we are forced to conclude that the interpretation of each hypergraph is completely determined. This decomposition into atomic hypergraphs will not be unique, but the interpretation will still be well-defined thanks to the interchange law for duoidal categories.

Definition 7.15 (Nodes connected by a wire)**.** Two nodes of a hypergraph, $n_1, n_2 \in H_N$ are *connected by a wire*, $n_1 \preceq_0 n_2$, when there exists a wire, $w \in H_W$, going from the output of the first node, $w \in \mathsf{output}(n_1)$ to the input of the second node, $w \in \mathsf{input}(n_2)$. This defines a relation, $(\preceq_0)\colon H_N \times H_N$.

Definition 7.16 (Ordering the nodes)**.** Let H be a physical string diagram. An *ordering* on the nodes is any total order, $(\preceq)\colon H_N \times H_N$, that extends the relation being connected by a wire, $(\preceq_0)\colon H_N \times H_N$.

Note that this ordering always exist: physical hypergraphs are acyclic. However, it is not unique: $(\preceq_0)$ is typically not a total order.

Lemma 7.17 (Parallel wires)**.** *For any string diagram H and any ordering on its nodes $(\preceq)$, we can define the set of wires* parallel to a node, $\mathrm{par}(n)$, *to contain exactly those wires $w \in W$ that are neither on the input below the node – $w \notin \mathsf{input}(m)$ for each $m \preceq n$ – nor on the output above the node – $w \notin \mathsf{output}(m)$ for each $n \preceq m$. Parallel wires,* $\mathrm{par}(n)$*, together with the node, n, form a poset,* $\mathrm{Level}(n)$*.*

Definition 7.18 (Atomic hypergraph)**.** The *atomic hypergraph* of a node n has a single node and three types of wires, $H_W = \mathsf{input}(n) + \mathsf{output}(n) + \mathrm{par}(n)$: (1) those on the input of the node; (2) those on the output of the node; and (3) those parallel to the node.

$$\mathrm{Atomic}(n)\colon \mathrm{Level}(n)[n \backslash \mathsf{input}(n)] \to \mathrm{Level}(n)[n \backslash \mathsf{output}(n)]$$

Proposition 7.19 (Decomposition into atomic hypergraphs)**.** *Every physical hypergraph can be written as the composition of atomic hypergraphs.*

Proof. See Appendix, Proposition A.19. □

Lemma 7.20 (String diagram universal map)**.** *For each physical duoidal signature, $\mathcal{G}$, there exists a signature homomorphism, $u_{\mathcal{G}}\colon \mathcal{G} \to \mathsf{Forget}(\mathsf{PhyString}(\mathcal{G}))$.*

Proof. See Appendix, Lemma A.20. □

Theorem 7.21 (String diagram adjunction)**.** *There exists an adjunction from the category of physical duoidal signatures to the category of strict physical duoidal categories given by physical duoidal string diagrams $\mathsf{PhyString}\colon \mathbf{PhySig} \to \mathbf{PhyDuo}$ and the forgetful functor, $\mathsf{forget}\colon \mathbf{PhyDuo} \to \mathbf{PhySig}$.*

Proof. See Appendix, Theorem A.21. ☐

8. Examples

Let us study multiple examples of constructions in physical duoidal categories using the string diagrams that we formalize in the rest of the text.

8.1. Duoids. In the same way that the *microcosm principle* prescribes that monoids are more generally defined in a multicategory and that Frobenius monoids are more generally definable on a polycategory, duoids are more generally defined in a duoidal category.

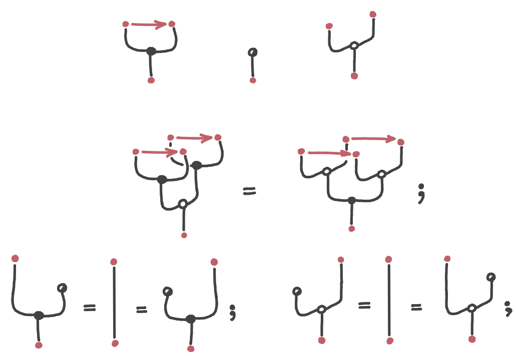


Figure 4. Generators and axioms for a duoid in a physical duoidal.

Definition 8.1 (Normal duoid)**.** A *normal duoid* in a physical duoidal category is a $\triangleleft$-monoid in the category of $\otimes$-monoids that additionally shares its unit with that of the base monoid.

$$m\colon X \otimes X \to X, \qquad s\colon X \triangleleft X \to X, \qquad u\colon N \to X.$$

This means that the following diagrams must all commute.

$$\begin{array}{ccc} X \otimes X \otimes X & \xrightarrow{m \otimes \mathrm{id}} & X \otimes X \\ {\scriptstyle \mathrm{id}\otimes m}\downarrow & & \downarrow{\scriptstyle m} \\ X \otimes X & \xrightarrow[m]{} & X \end{array} \qquad \begin{array}{ccc} X & \xrightarrow{\mathrm{id} \otimes u} & X \otimes X \\ {\scriptstyle u\otimes \mathrm{id}}\downarrow & \searrow^{\mathrm{id}} & \downarrow{\scriptstyle m} \\ X \otimes X & \xrightarrow[m]{} & X \end{array}$$

$$\begin{array}{ccc} X \triangleleft X \triangleleft X & \xrightarrow{m \triangleleft \mathrm{id}} & X \triangleleft X \\ {\scriptstyle \mathrm{id}\triangleleft m}\downarrow & & \downarrow{\scriptstyle m} \\ X \triangleleft X & \xrightarrow[m]{} & X \end{array} \qquad \begin{array}{ccc} X & \xrightarrow{\mathrm{id} \triangleleft u} & X \triangleleft X \\ {\scriptstyle u\triangleleft \mathrm{id}}\downarrow & \searrow^{\mathrm{id}} & \downarrow{\scriptstyle m} \\ X \triangleleft X & \xrightarrow[m]{} & X \end{array}$$

$$\begin{array}{ccccc} (X \triangleleft X) \otimes (X \triangleleft X) & \xrightarrow{s \otimes s} & X \otimes X & & \\ & & & \searrow^{m} & \\ {\scriptstyle \partial}\downarrow & & & & X \\ & & & \nearrow_{s} & \\ (X \otimes X) \triangleleft (X \otimes X) & \xrightarrow[m \triangleleft m]{} & X \triangleleft X & & \end{array}$$

Alternatively, a normal duoidal is given by the string diagrams of Figure 4.

Remark 8.2. A normal duoid in the normal duoidal category of bistrong profunctors [GF16] (or *Tambara modules*) is a monoidal promonad: an identity on objects strict monoidal functor.

8.2. **Enriched multicategories.** Multicategories can be enriched over any symmetric monoidal category. However, a more subtle and general enrichment for multicategories is that over physical duoidal categories, due to Rajesh in recent work [Raj13].

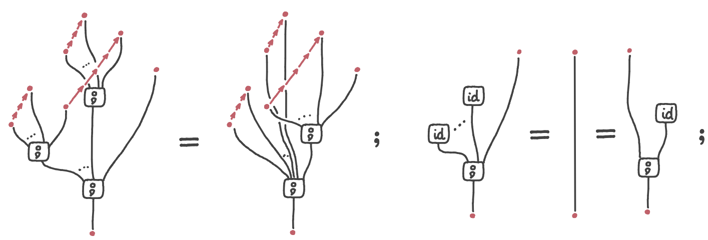


FIGURE 5. Axioms for physical-duoidally enriched multicategories.

Definition 8.3 (Duoidally enriched multicategory, Rajesh [Raj13])**.** An *enriched multicategory*, $\mathbb{C}$, over a physical duoidal category consists of *(i)* a set of objects, $\mathbb{C}_{obj}$; *(ii)* a set of multimorphisms from each list of objects to each single object, $\mathbb{C}(X_1, ..., X_n; Y)$ for each $X_1, ..., X_n, Y \in \mathbb{C}_{obj}$; *(iii)* an identity operation, $\mathrm{i}\colon N \to \mathbb{C}(X; X)$ for each $X \in \mathbb{C}_{obj}$; and *(iv)* a composition operation,

$$(\fatsemi)\colon \left(\triangleright_{i=1}^{k} \mathbb{C}(X_1^i, ..., X_{n_i}^i; Y_i)\right) \otimes \mathbb{C}(Y_1, ..., Y_m; Z) \to \mathbb{C}(X_1^1, ..., X_{n_1}^1, ..., X_1^k, ..., X_{n_k}^k; Y).$$

These must make all the following diagrams commute.

$$\begin{array}{ccc} \mathbb{C}(X_1, ..., X_n; Y) \otimes \mathbb{C}(Y; Y) & \xrightarrow{(\fatsemi)} & \mathbb{C}(X_1, ..., X_n; Y) \\ \uparrow{\scriptstyle \mathrm{id} \triangleright (\mathrm{i})} & & \\ \mathbb{C}(X_1, ..., X_n; Y) & \xrightarrow{\quad \mathrm{id} \quad} & \mathbb{C}(X_1, ..., X_n; Y) \end{array}$$

$$\begin{array}{ccc} \left(\triangleright_{i=1}^{k} \mathbb{C}(X_i; X_i)\right) \otimes \mathbb{C}(X_1, ..., X_n; Y) & \xrightarrow{(\fatsemi)} & \mathbb{C}(X_1, ..., X_n; Y) \\ \uparrow{\scriptstyle \left(\triangleright_{i=1}^{k} \mathrm{i}\right) \otimes \mathrm{id}} & & \\ \mathbb{C}(X_1, ..., X_n; Y) & \xrightarrow{\quad \mathrm{id} \quad} & \mathbb{C}(X_1, ..., X_n; Y) \end{array}$$

$$\begin{array}{ccc}
\begin{array}{c}\big(\triangleright_{j=1}^{p}\big(\triangleright_{i=1}^{m_j}\mathbb{C}(X_1^{i,j},...,X_{n_{i,j}}^{i,j};Y_i^j)\big)\\ \otimes\mathbb{C}(Y_1^j,...,Y_{m_j}^j;Z_j)\big)\\ \otimes\mathbb{C}(Z_1,...,Z_p;U)\end{array} & \xrightarrow{(\fatsemi)\otimes\mathrm{id}} & \begin{array}{c}\big(\triangleright_{j=1}^{p}\mathbb{C}(X_1^{1,1},...,X_{n_{m_j,j}}^{m_j,j};Z_j)\big)\\ \otimes\mathbb{C}(Z_1,...,Z_p;U)\end{array} \\
\downarrow{\scriptstyle \eth} & & \\
\begin{array}{c}\big(\triangleright_{j=1}^{p}\triangleright_{i=1}^{m_p}\mathbb{C}(X_1^{i,j},...,X_{n_{i,j}}^{i,j};Y_i^j)\big)\\ \otimes\big(\triangleright_{i=1}^{m_p}\mathbb{C}(Y_1^j,...,Y_{m_p}^j;Z_j)\big)\\ \otimes\mathbb{C}(Z_1,...,Z_p;U)\end{array} & & \downarrow{\scriptstyle (\fatsemi)} \\
\downarrow{\scriptstyle \mathrm{id}\otimes(\fatsemi)} & & \\
\begin{array}{c}\big(\triangleright_{j=1}^{p}\triangleright_{i=1}^{m_p}\mathbb{C}(X_1^{i,j},...,X_{n_{i,j}}^{i,j};Y_i^j)\big)\\ \otimes\mathbb{C}(Y_1^1,...,Y_{m_p}^p;U)\end{array} & \xrightarrow[(\fatsemi)]{} & \mathbb{C}(X_1^{1,1},...,X_{n_{m_p,p}}^{m_p,p};U)
\end{array}$$

Alternatively, a duoidally-enriched multicategory is given by the string diagrams of Figure 5 (c.f. Definition A.22).

8.3. **Bimonoids, Frobenius monoids and Dualities.** We can also define most of the concepts we can define in braided monoidal categories, with the twist of using two different tensors: note that this is not less nor more general than defining these concepts in braided monoidal categories; we assume symmetry of one of the tensors, but we gain a new tensor that does not need to be braided.

Definition 8.4 (Bimonoid in a physical duoidal category)**.** A *bimonoid* in a physical duoidal category is a $\triangleleft$-comonoid in the category of $\otimes$-monoids.

$$m\colon X\otimes X\to X,\qquad u\colon N\to X,\qquad d\colon X\to X\triangleleft X,\qquad e\colon X\to N.$$

This means that the following diagrams must all commute.

$$\begin{array}{ccc}
X\otimes X\otimes X & \xrightarrow{m\otimes\mathrm{id}} & X\otimes X\\
\downarrow{\scriptstyle \mathrm{id}\otimes m} & & \downarrow{\scriptstyle m}\\
X\otimes X & \xrightarrow[m]{} & X
\end{array}
\qquad
\begin{array}{ccc}
X & \xrightarrow{\mathrm{id}\otimes u} & X\otimes X\\
\downarrow{\scriptstyle u\otimes\mathrm{id}} & \searrow^{\mathrm{id}} & \downarrow{\scriptstyle m}\\
X\otimes X & \xrightarrow[m]{} & X
\end{array}$$

$$\begin{array}{ccc}
X & \xrightarrow{d} & X\triangleleft X\\
\downarrow{\scriptstyle d} & & \downarrow{\scriptstyle d\triangleleft\mathrm{id}}\\
X\triangleleft X & \xrightarrow[\mathrm{id}\triangleleft d]{} & X\triangleleft X\triangleleft X
\end{array}
\qquad
\begin{array}{ccc}
X & \xrightarrow{d} & X\triangleleft X\\
\downarrow{\scriptstyle d} & & \downarrow{\scriptstyle \mathrm{id}\triangleleft e}\\
X\triangleleft X & \xrightarrow[e\triangleleft\mathrm{id}]{} & X
\end{array}$$

$$\begin{array}{ccccc}
X\otimes X & \xrightarrow{m} & X & \xrightarrow{d} & X\triangleleft X\\
\downarrow{\scriptstyle d\otimes d} & & & & \|{\scriptstyle \mathrm{id}}\\
(X\triangleleft X)\otimes(X\triangleleft X) & \xrightarrow[\eth]{} & (X\otimes X)\triangleleft(X\otimes X) & \xrightarrow[m\triangleleft m]{} & X\triangleleft X
\end{array}$$

$$\begin{array}{ccc}
N & \xrightarrow{u\triangleleft u} & X\triangleleft X\\
\downarrow{\scriptstyle u} & & \|{\scriptstyle \mathrm{id}}\\
X & \xrightarrow[d]{} & X\triangleleft X
\end{array}
\qquad
\begin{array}{ccc}
X\otimes X & \xrightarrow{e\otimes e} & N\\
\downarrow{\scriptstyle m} & & \|{\scriptstyle \mathrm{id}}\\
X & \xrightarrow{e} & N
\end{array}$$

Alternatively, a bimonoid is given by the string diagrams of Figure 6.

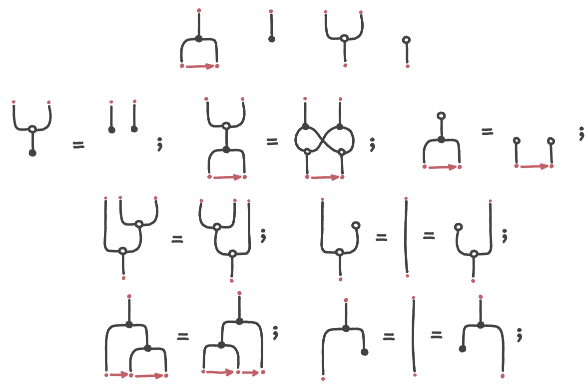


Figure 6. Generators and axioms for a bimonoid in a physical duoidal category.

Definition 8.5 (Frobenius monoid in a physical duoidal category)**.** A *Frobenius monoid* in a physical duoidal category is both a $\triangleleft$-comonoid and a $\otimes$-monoid structure over the same object, interacting by the Frobenius axiom.

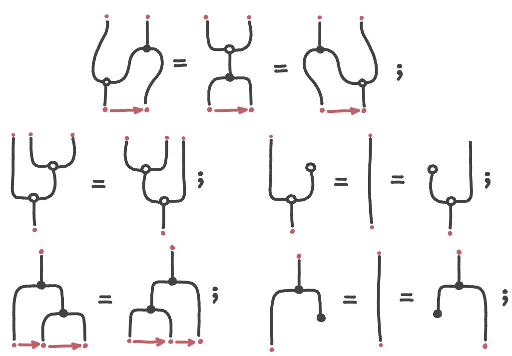


Figure 7. Generators and axioms for a Frobenius monoid in a physical duoidal category.

Definition 8.6 (Duoidal duality)**.** A *duoidal duality*, $A \dashv A^*$, is a pair of morphisms, $\eta\colon N \to A \triangleleft A^*$ and $\varepsilon\colon A^* \otimes A \to N$, satisfying the snake equations up to the duoidal distributor.

Proposition 8.7. *A duoidal duality induces a duoidal Frobenius monoid structure on $A \triangleleft A^*$ (see Figure 8).*

Definition 8.8. Traditionally, we can only define commutative monoids in symmetric (or braided) monoidal categories. A $\otimes$-commutative $\triangleleft$-monoid in a physical duoidal category is such that the following equation holds.

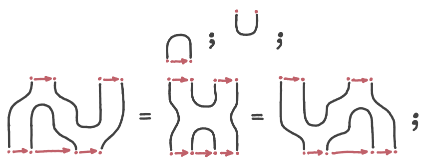


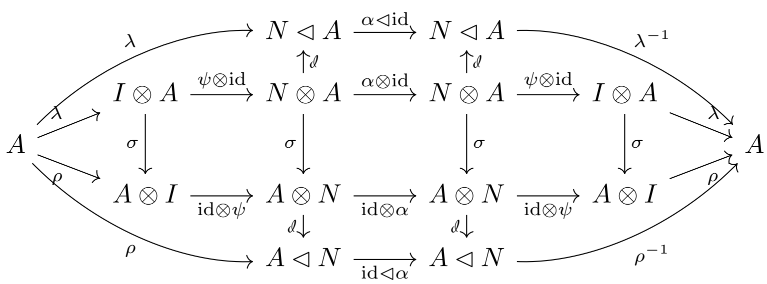
FIGURE 8. A duoidal duality induces a Frobenius monoid.

8.4. **Physical Duoidal Categories are Spacial.**

Proposition 8.9. *Every physical duoidal category is* ($\triangleleft$)*-spacial, in the sense of Selinger* [Sel10]*. Given any object* A *and any scalar* $\alpha \colon I \to I$*, we have that*

$$\mathrm{id}_A \triangleleft \alpha = \alpha \triangleleft \mathrm{id}_A.$$

Proof. The following is a commutative diagram. We use naturality of the symmetry (σ) and of the distributors (∂), and we use coherence for normal duoidal categories.



Alternatively, in terms of string diagrams, the equation in Figure 9 holds. □

FIGURE 9. Spacial equation.

9. OPPHYSICAL DUOIDAL CATEGORIES

While the opposite of a normal duoidal category, $(\mathbb{V}, \otimes, \triangleleft, N)$, is again a normal duoidal category, $(\mathbb{V}^{op}, \triangleleft, \otimes, N)$, it is not true that the opposite of a physical duoidal category is again a physical duoidal category: there is no reason to expect that both tensors will be symmetric. Instead, the opposite of a physical duoidal category is an *opphysical duoidal category.*

Definition 9.1. A *strict ophysical duoidal category* is a category with a strict monoidal structure and a strict symmetric monoidal structure sharing the same unit, $(\mathbb{V}, \boxtimes, \triangleleft, N)$, and such that the first monoidal structure distributes over the second; that is, there exist maps

$$\partial_{X,Y,Z,W} \colon (X \boxtimes Z) \triangleleft (Y \boxtimes W) \to (X \triangleleft Y) \boxtimes (Z \triangleleft W);$$

$$\sigma_{X,Y} : X \boxtimes Y \to Y \boxtimes X.$$

Opphysical duoidal categories are defined to be coherent structures, meaning that any formally distinctly typed equation of morphisms on the free strict opphysical duoidal category holds true.

Corollary 9.2. *String diagrams for opphysical duoidal categories are exactly the bottom-top string diagrams for physical duoidal categories.*

10. Conclusions

10.1. **Related Work.** Our main reference text is the monograph on duoidal categories by Aguiar and Mahajan [AM10], where various results on coherence for duoidal categories and normal duoidal categories are discussed. Our presentation of physical duoidal categories follows that of Shapiro and Spivak [SS22], who also introduced the name *physical* for normal and ⊗-symmetric duoidal categories. The study of physical duoidal expressions and their correspondence to zetless posets is due to Grabowski and Gischer.

On the applied side, we follow the idea of Garner and López Franco [GF16] of using normal duoidal categories to study commutativity on algebraic structures; but also previous work by Earnshaw, Hefford, and this author on the interpretation of causality on monoidal string diagrams using duoidal structures [EHR24].

10.2. **Further work.** We can easily conjecture that the string diagrams for *dependence categories* [SS22] are similar to those presented here, with the only difference of removing the restriction to zetless posets. Although much more natural from this point of view, dependence categories are less frequent than physical duoidal categories, and we leave their study to further work.

String diagrams for physical duoidal categories particularize into the hypergraph string diagrams for symmetric monoidal categories: in fact, any symmetric monoidal category is automatically duoidal with itself. A question remains on whether string diagrams for physical duoidal categories also particularize to certain planar monoidal categories: it can be shown that a monoidal category can be part of a duoidal structure only if it is spacial in the sense of Selinger [Sel10].

After our characterization, it becomes obvious that a do-notation where a poset of variables is automatically tracked by the type-checker could constitute a good programming-like internal language for physical duoidal categories. We leave this development for further work.

10.3. **Acknowledgements.** The author wants to thank Matt Earnshaw, Nayan Rajesh, Philip Saville, and Sam Staton, for helpful discussion on physical duoidal categories.

References

[AM10] Marcelo Aguiar and Swapneel Arvind Mahajan. *Monoidal functors, species and Hopf algebras*, volume 29. American Mathematical Society Providence, RI, 2010.

[BHP+19] Filippo Bonchi, Joshua Holland, Robin Piedeleu, Paweł Sobociński, and Fabio Zanasi. Diagrammatic algebra: from linear to concurrent systems. *Proc. ACM Program. Lang.*, 3(POPL):25:1–25:28, 2019.

[EHR24] Matt Earnshaw, James Hefford, and Mario Román. The Produoidal Algebra of Process Decomposition. In Aniello Murano and Alexandra Silva, editors, *32nd EACSL Annual Conference on Computer Science Logic, CSL 2024, February 19-23, 2024, Naples,* *Italy*, volume 288 of *LIPIcs*, pages 25:1–25:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024.

[GF16] Richard Garner and Ignacio López Franco. Commutativity. *Journal of Pure and Applied Algebra*, 220(5):1707–1751, 2016.

[JS91] André Joyal and Ross Street. The geometry of tensor calculus, I. *Advances in Mathematics*, 88(1):55–112, 1991.

[Raj13] Nayan Rajesh. Universal Algebra and Effectful Computation. *Master's Thesis, University of Oxford.*, 2013.

[Sel10] Peter Selinger. A survey of graphical languages for monoidal categories. In *New structures for physics*, pages 289–355. Springer, 2010.

[SS22] Brandon T. Shapiro and David I. Spivak. Duoidal structures for compositional dependence. *arXiv preprint arXiv:2210.01962*, 2022.

Appendix A. Omitted Proofs

Proposition A.1 (From Proposition 2.3)**.** *Strict physical duoidal categories and strict physical duoidal functors between them form a category,* **PhyDuo***.*

Proof. Let us show that any identity functor is a strict physical duoidal functor: it is strict monoidal for the two structures, respectively, and it also strictly preserves structure maps, $\mathrm{Id}(\eth_V) = \eth_V$ and $\mathrm{Id}(\sigma_V) = \sigma_V$.

Let us show that any composition of two strict physical duoidal functors,

$$F\colon (\mathbb{V}, \otimes_V, \lhd_V, \eth_V, \sigma_V) \to (\mathbb{W}, \otimes_W, \lhd_W, \eth_W, \sigma_W), \text{ and}$$
$$G\colon (\mathbb{W}, \otimes_W, \lhd_W, \eth_W, \sigma_W) \to (\mathbb{X}, \otimes_X, \lhd_X, \eth_X, \sigma_X)$$

is again a strict physical duoidal functor. In fact, the composition of strict symmetric monoidal functors is again a strict symmetric monoidal functor; the composition of strict monoidal functors is again a strict monoidal functor; and we can see that $G(F(\eth_V)) = G(\eth_W) = \eth_X$ and $G(F(\sigma_V)) = G(\sigma_W) = \sigma_X$. □

Proposition A.2 (From Proposition 4.9)**.** *In a zetless poset, any two connected elements must be connected by either a span or a cospan. That is, if there is a path between two elements, $x_0 \to x_1 \leftarrow x_2 \to \dots \leftarrow x_n$, there must exist either a cospan between them, $x_0 \to u \leftarrow x_n$, or a span between them, $x_0 \leftarrow v \to x_n$.*

Proof. Assume x is connected to y. If the shortest path connecting them has less than three steps, we are done. Otherwise, the shortest path connecting them must start, without loss of generality, by $x_0 \to x_1 \leftarrow x_2 \to x_3$.

We know that $x_0 \not\leq x_2$, because $x_0 \leq x_2$ would break the minimality of the path. We also know that $x_2 \not\leq x_0$, because $x_2 \leq x_0$ would build a cospan, breaking the minimality again. By an analogous reasoning, $x_1 \not\leq x_3$ and $x_3 \not\leq x_1$. Finally, we know that $x_0 \not\leq x_3$ and $x_3 \not\leq x_0$, again because of minimality of the path.

However, all this means that we have a fully faithfull embedding of the Z-poset: we have reached a contradiction. □

Proposition A.3 (From Proposition 5.2)**.** *Zetless posets labelled over a set are in correspondence with duoidal expressions on that set, up to symmetries of the tensored components,*

$$\mathsf{zetless}(A) \cong \mathsf{expr}(A)/(\approx).$$

Proof. Any zetless poset is exactly in one of these four cases: (1) it is empty; (2) it is a singleton; (3) it is $\lhd$-composite, connected and thus $\lhd$-prime; or (4) it is $\otimes$-composite, disconnected and thus $\otimes$-prime. Note that, if any zetless poset was to be both $\otimes$-prime and $\lhd$-prime, it must be a singleton (Proposition 4.11).

As a consequence, every zetless poset can be built, exclusively, either as (1) the empty expression, N; (2) a singleton, X; (3) a sequencing of zetless posets, $P_1 \lhd \dots \lhd P_n$; or (4) a tensoring of zetless posets, $P_1 \otimes \dots \otimes P_n$: in this last case, because the tensoring operation is commutative, it can be built not uniquely, but up to a permutation, which is accounted for by the quotienting $(\approx)$. □

Proposition A.4 (From Proposition 5.4)**.** *The existence of an inclusion of zetless posets corresponds to the existence of a structure map between their corresponding duoidal expressions in a physical duoidal category.*

Proof. Adapted from Shapiro and Spivak [SS22]. Any structure map in the category of zetless posets induces an inclusion. Let us prove that the existence of an inclusion of zetless posets implies the existence of a structure map between their corresponding expressions. We will employ induction on the size of the zetless posets forming the inclusion, $P \leftrightarrows Q$.

The posets P and Q must be in any of the previous four cases. Note that, if P is a sequence of posets, then it must be connected and Q must also be connected and factor as a sequence of posets. Note that, if Q is a tensoring of posets, then it must be incomparable connected and P must also be incomparable connected and factor as a tensoring of posets.

1. (1) Both are empty; $\mathrm{id}_N : N \to N$ is a structure map.
2. (2) Both are a singleton; $\mathrm{id}_A : N \to N$ is a structure map.
3. (3) The first one is a sequencing of posets, $P = P_1 \triangleleft ... \triangleleft P_n$, thus we can write the second one (which has more edges) as a sequencing of the posets containing the same objects, $Q = Q_1 \triangleleft ... \triangleleft Q_n$, where the objects of P_i are the objects of Q_i. Any inclusion of posets between the sequencing of posets must factor as a sequencing of inclusions. We proceed by induction.
4. (4) The second one is a tensoring of posets, $Q = Q_1 \otimes ... \otimes Q_n$, thus we can write the first one (which has less edges) as a tensoring of of the posets containing the same objects, $P = P_1 \otimes ... \otimes P_n$, where the objects of P_i are the objects of Q_i. Any inclusion of posets between the tensoring of posets must factor as a tensoring of inclusions. We proceed by induction.
5. (5) Finally, assume that the first one is a tensoring of posets, $P = P_1 \otimes ... \otimes P_n$, and that the second one is a sequencing of posets, $Q = Q_1 \triangleleft ... \triangleleft Q_m$. Let us define P_{ij} to be the full subposet of P whose objects are the intersection of the objects in P_i and Q_j; let us define Q_{ij} to be the full subposet of Q whose objects are the intersection of the objects in P_i and Q_j.

   If there is an inclusion, then the following is a structure map.

$$\begin{aligned} P_1 \otimes ... \otimes P_n &\longrightarrow (P_{11} \triangleleft ... \triangleleft P_{1m}) \otimes ... \otimes (P_{n1} \triangleleft ... \triangleleft P_{nm}), \\ &\longrightarrow (P_{11} \otimes ... \otimes P_{n1}) \triangleleft ... \triangleleft (P_{1m} \triangleleft ... \triangleleft P_{nm}), \\ &\longrightarrow Q_1 \triangleleft ... \triangleleft Q_m. \end{aligned}$$

These cover all possible cases: any inclusion of zetless posets corresponds to an structure map. □

Theorem A.5 (From Theorem 5.7)**.** *Zetless posets construct the free physical duoidal category over a set of objects. In other words, the functor* Zetless: **Set** → **PhyDuo** *is left adjoint to the forgetful functor that picks the objects of a physical duoidal category,* Obj: **PhyDuo** → **Set**.

Proof. Let us first construct the unit, $u_A : A \to \mathsf{Obj}(\mathsf{Zetless}(A))$: every element appears as a singleton. We will show that this is a universal arrow. Let $\mathbb{V}$ be a physical duoidal category and let $f : A \to \mathsf{Obj}(\mathbb{V})$ pick some objects of the category. We will prove that there exists a unique strict physical duoidal functor, $f^* : \mathsf{Zetless}(A) \to \mathbb{V}$, factoring $f = u \mathbin{⨾} \mathsf{Obj}(f^*)$.

The factoring forces $f^*(a) = f(a)$ for any singleton poset for an element $a \in A$. This is enough to force the inductive definition of the functor on objects.

1. (1) $f^*(N) = N$, because of duoidality;
2. (2) $f^*(a) = f(a)$, because of universality;

(3) $f^*(E_1 \lhd ... \lhd E_n) = f^*(E_1) \lhd ... \lhd f^*(E_n)$, because of duoidality; and
(4) $f^*(E_1 \otimes ... \otimes E_n) = f^*(E_1) \otimes ... \otimes f^*(E_n)$, because of duoidality.

Any zetless poset can be writen uniquely as a duoidal expression on the singleton posets, uniquely up to symmetry (Proposition 5.2); because the strict physical duoidal functor must preserve these expressions, it is determined on objects. Finally, an bijective-on-objects inclusion between posets correspond to structure maps (Proposition 5.4); because the functor must preserve structure maps, the image of these is determined.

We have built then an assignment on objects and morphisms. The assignment must be functorial because there exists at most a unique morphism between any two distinctly typed objects in the free physical duoidal category. This creates the only possible functor, $f^* \colon \mathsf{Zetless}(A) \to \mathbb{V}$, satisfying the factorization property. □

Proposition A.6 (From Proposition 6.5)**.** *A poset R arises as a substitution $R = Q[x \backslash P]$ of any of its full subposets, $P \subseteq R$, if and only if it is bracketed.*

Proof. The subposet $P \subseteq Q[x \backslash P]$ is bracketed by construction. Let us show that any poset with a bracketed subposet is the result of a substitution. Indeed, if $P \subseteq R$ is bracketed, then we can construct a poset Q that contains all of the objects of R but a single object substituting these on P,

$$Q = (R_{obj} - P_{obj} + \{x\}; \leq_R + \{r \leq x \mid r \leq p, p \in P\}) + \{x \leq r \mid p \leq r, p \in P\}).$$

We can see now that $R = Q[x \backslash P]$. Indeed, if $r \leq p_0$ in the original poset then $r \leq p$ for each $p \in P$; but then $r \leq x$ and we indeed recover $r \leq p$. Analogously, if $p_0 \leq r$ in the original poset then $p \leq r$ for each $p \in P$; but then $x \leq r$ and we indeed recover $p \leq r$. □

Proposition A.7 (From Proposition 6.7)**.** *A subset of a zetless poset is an interval if and only if it appears as a bracketed poset in some saturation of the poset.*

Proof. Any bracketed poset must be an interval; and any interval remains an interval in a less saturated poset. Let us prove that any interval $I \subseteq P$ appears as a bracketed poset after some saturation. Indeed, for each $u \in P$ such that there exists $x_0 \in I$ with $u \leq x_0$, we impose that $u \leq x$ for every $x \in I$; for each $u \in P$ such that there exists $x_0 \in I$ with $x_0 \leq u$, we impose that $x \leq u$ for every $x \in I$. Because I is an interval, this constitutes a saturation that does not identify any two elements of the poset; this saturation makes the interval bracketed. □

Proposition A.8 (From Proposition 7.14)**.** *The construction of physical duoidal string diagrams over a physical duoidal signature extends to a functor*

$$\mathsf{PhyString} \colon \mathbf{PhySig} \to \mathbf{PhyDuo}.$$

Proof. The functor will take a signature homomorphism $f \colon \mathcal{G} \to \mathcal{H}$ into a strict physical duoidal functor $\mathsf{PhyString}(f) \colon \mathsf{PhyString}(\mathcal{G}) \to \mathsf{PhyString}(\mathcal{H})$. Let us first define what $\mathsf{PhyString}(f)$ does: it takes any string diagram into the same string diagram but relabelling the nodes and wires using f and f_t, respectively: this is a compatible relabelling because a signature homomorphism must preserve the source and target of its generators, $\mathcal{G}(U; V)$ is mapped into $\mathcal{H}(fU; fV)$.

We need to show that $\mathsf{PhyString}(f)$, as defined here, forms a strict physical duoidal functor. Because the relabelling of a composition, sequencing or tensoring of diagrams is equal to the composition, sequencing or tensoring of its relabeled components, we know that it must form a strict physical duoidal functor.

Finally, we can check that relabelling by a composition of functions is the same as relabelling twice by each of the functions; relabelling by an identity yields the identity on string diagrams. This makes our original construction functorial. □

Proposition A.9 (From Proposition 4.11)**.** *A poset is ⊗-prime if and only if it is connected.*

Proof. If a poset is disconnected, then it can be written as the tensor if its connected components, contradicting primality. If a poset is not ⊗-prime, it must be disconnected because vertices on its different factors cannot be connected. □

Proposition A.10 (From Proposition 4.11)**.** *A poset P is $\triangleleft$-prime if and only if it is incomparable connected.*

Proof. Assume the poset P has more than one incomparable connected component: $P_1, ..., P_n$; with all of them being non-empty. Picking any two components and any two elements on them, assume without loss of generality that $p_i \in P_i$ and $p_j \in P_j$ give $p_i \leq p_j$.

Now, if $p_i' \leq p_i$, it cannot be that $p_j \leq p_i'$ because that would imply $p_j \leq p_i$; it must be that $p_i' \leq p_j$: thus, all incomparable connected elements must be below p_j, and we conclude $P_i < p_j$. By an analogous reansoning, $P_i < P_j$. This imposes a total order on the components. If, without loss of generality, we assume that $P_1 < ... < P_n$, then we can conclude $P = P_1 \triangleleft ... \triangleleft P_n$, contradicting primality.

Finally, if a poset is written as a sequencing of posets, it must be incomparable-disconnected, as each factor is a different component. □

Proposition A.11. *Any $\triangleleft$-prime and ⊗-prime poset is a singleton.*

Proof. Primality means that the poset must be connected (Proposition 4.11) and incomparable-connected (Proposition 4.11). Let us assume the poset is not a singleton and arrive to a contradiction. If it is not a singleton, it must contain a minimal and maximal element that are distinct but, because of connectedness, related as $i < m$.

Now, they must be incomparable connected, and we are going to prove that any element ($\|$)-connect to o cannot be ($\|$)-connected to m: it must be below m. There are two cases here.

- Connected by a single step: let $o \parallel u$. They must be connected by a span, $o \to v \leftarrow u$; but then we require $u \leq m$ to break the zet.
- Connected by multiple steps: let it be $i \parallel u_1 \parallel u_2 \parallel ... \parallel u_n$. We will prove by induction on the length n that u_n is below m. We know that it cannot be that $u_{n-2} \parallel u$, because that would contradict minimality: that means that either $u_n < u_{n-2}$ (and, in that case, $u < m$) or $u_{n-2} < u_n$ (and, to break the zet, $u_n < m$). In any case, $u_n < m$ is forced.

Because anything incomparable connected to n is below m, it will never ($\|$)-connect to m; this reaches a contradiction. □

Proposition A.12 (From Proposition 7.4)**.** *Forgetting about the composition, sequencing and tensoring of a physical duoidal category extends to a functor*

$$\mathsf{Forget}\colon \mathbf{PhyDuo} \to \mathbf{PhySig}.$$

Proof. Let $\mathbb{V}$ be a physical duoidal category. The basic types of the associated physical duoidal signature are all of the objects of the category, $\mathsf{Forget}(\mathcal{G})_t = \mathbb{V}_{obj}$. For each pair of duoidal expressions, $E_i, E_o \in \mathsf{expr}(\mathcal{G}_t)$, we pick the set of morphisms $\mathbb{V}(⟦E_i⟧, ⟦E_o⟧)$ as our generators. □

Proposition A.13 (From Proposition 4.8)**.** *The sequencing and tensoring of two zetless posets is again a zetless poset.*

Proof. Because Z is connected, if it embeds into the disconnected poset $P \otimes Q$, the embedding must be contained into any of the connected components. Because Z is incomparable connected, if it embeds into the incomparable-disconnected poset $P \triangleleft Q$, the embedding must be contained into any of the incomparable connected components. □

Theorem A.14 (From Theorem 7.13)**.** *Physical duoidal string diagrams over a physical duoidal signature, $\mathcal{G}$ form a physical duoidal category,* $\mathsf{PhyString}(\mathcal{G})$.

Objects are duoidal expressions over the types of the physical duoidal signature; morphisms are hypergraphs with input and output given by the zetless posets corresponding to the duoidal expressions.

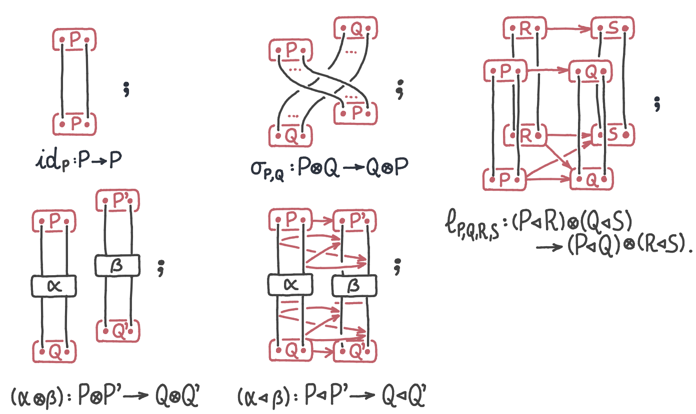


FIGURE 10. Physical string diagrams form a physical duoidal category.

Composing two string diagrams, α and β, concatenates them, so that the output wires of α and the input wires of β get merged into single wires. Parallel tensoring juxtapos two diagrams; sequential tensoring juxtaposes but also links every wire from the first diagram to the second diagram (see Figure 3).

Proof. The category structure is given by composition of string diagrams (Definition A.15), tensoring (Definition A.17), and sequencing (Definition A.18). Composition is associative (Proposition A.16), and we can check it is unital; tensoring and sequencing are unital and associative. The interchange equation for duoidal categories holds. We can construct the laxators by a string diagram consisting only of wires. □

Definition A.15 (Composition of string diagrams)**.** Let $\alpha\colon E_1 \to E_2$ and $\beta\colon E_2 \to E_3$ be two string diagrams. Their composition, $(\alpha \mathbin{⨾} \beta)\colon E_1 \to E_3$, has all the nodes of both string diagrams,

$$\mathsf{nodes}(\alpha \mathbin{⨾} \beta) = \mathsf{nodes}(\alpha) + \mathsf{nodes}(\beta);$$

and the pushout of wires of string diagrams along the output and input boundaries; in other words, it is the pushout of $\mathsf{input}(o_\alpha)\colon E_2 \to \mathsf{wires}(\alpha)$ and $\mathsf{output}(i_\beta)\colon E_2 \to \mathsf{wires}(\beta)$, written as

$$\mathsf{wires}(\alpha \mathbin{⨾} \beta) = \mathsf{wires}(\alpha) +_{\mathsf{input}(o_\alpha)}^{\mathsf{output}(i_\beta)} \mathsf{wires}(\beta).$$

The source and target functions of the nodes are preserved. The composition of wire-linear and acyclic hypergraphs is again a wire-linear and acyclic hypergraph: by definition, there does not exist any wire with source in β and target in α; the only wires that change source and target were output and input wires that got merged, going from a single node in α to a single node in β.

Proposition A.16 (Composition of string diagrams is associative)**.** *Let $\alpha\colon E_1 \to E_2$, $\beta\colon E_2 \to E_3$, and $\gamma\colon E_3 \to E_4$ be three string diagrams. Their composition is associative, $(\alpha \mathbin{⨾} \beta) \mathbin{⨾} \gamma = \alpha \mathbin{⨾} (\beta \mathbin{⨾} \gamma)$.*

Proof. The only observation we need is that the construction of the pushout is associative,

$$\big(\mathsf{wires}(\alpha) +_{\mathsf{input}(o_\alpha)}^{\mathsf{output}(i_\beta)} \mathsf{wires}(\beta)\big) +_{\mathsf{input}(o_\beta)}^{\mathsf{output}(i_\gamma)} \mathsf{wires}(\gamma) \cong$$
$$\mathsf{wires}(\alpha) +_{\mathsf{input}(o_\alpha)}^{\mathsf{output}(i_\beta)} \big(\mathsf{wires}(\beta) +_{\mathsf{input}(o_\beta)}^{\mathsf{output}(i_\gamma)} \mathsf{wires}(\gamma)\big).$$

The source and target functions of the nodes are preserved in any case. □

Definition A.17 (Tensoring of string diagrams)**.** Let $\alpha\colon E_1 \to E_3$ and $\beta\colon E_2 \to E_4$ be two string diagrams. Their tensoring, $(\alpha \otimes \beta)\colon E_1 \otimes_e E_2 \to E_3 \otimes_e E_4$, has all the nodes and wires of both string diagrams,

$$\mathsf{nodes}(\alpha \otimes \beta) = \mathsf{nodes}(\alpha) + \mathsf{nodes}(\beta);$$
$$\mathsf{wires}(\alpha \otimes \beta) = \mathsf{wires}(\alpha) + \mathsf{wires}(\beta).$$

The source and target functions of the nodes are preserved. The poset structure is preserved. The tensoring of wire-linear and acyclic hypergraphs is again a wire-linear and acyclic hypergraph: by definition, there does not exist any wire with source in β and target in α; no wires change source or target.

Definition A.18 (Sequencing of string diagrams)**.** Let $\alpha\colon E_1 \to E_3$ and $\beta\colon E_2 \to E_4$ be two string diagrams. Their sequencing, $(\alpha \triangleleft \beta)\colon E_1 \triangleleft_e E_2 \to E_3 \triangleleft_e E_4$, has all the nodes and wires of both string diagrams,

$$\mathsf{nodes}(\alpha \triangleleft \beta) = \mathsf{nodes}(\alpha) + \mathsf{nodes}(\beta);$$
$$\mathsf{wires}(\alpha \triangleleft \beta) = \mathsf{wires}(\alpha) + \mathsf{wires}(\beta).$$

The source and target functions of the nodes are preserved. However, the poset structure changes: we construct a new poset with all the previous inequalities but additionally imposing that $x \sqsubseteq y$ whenever $x \in \alpha$ and $y \in \beta$. This is still a poset because elements from both diagrams were previously incomparable. The sequencing of wire-linear and acyclic hypergraphs is again a wire-linear and acyclic hypergraph: by definition, there does not exist any wire with source in β and target in α; no wires change source or target.

Proposition A.19 (From Proposition 7.19)**.** *Every physical hypergraph can be written as the composition of atomic hypergraphs.*

Proof. Every physical hypergraph is acyclic, and nodes can be given a total ordering, $(\preceq)$, even if not uniquely (Definition 7.16). If nodes are ordered as

$$n_0 \preceq n_1 \preceq n_2 \preceq ... \preceq n_k,$$

then we claim that the following composition recovers the hypergraph,

$$\text{Atomic}(n_0) \fatsemi \text{Atomic}(n_1) \fatsemi \text{Atomic}(n_2) \fatsemi ... \fatsemi \text{Atomic}(n_k).$$

We can proceed by induction on the number of nodes. If a physical hypergraph has zero nodes, then it consists of only wires and it is the identity for composition.

If a physical hypergraph has $k+1$ nodes, we can pick the first of them, n_0, and realize that all of its input wires must be part of the input to the hypergraph: if they were connected to the output of any node, that would contradict the properties of the ordering of nodes, $(\preceq)$. We can thus form a new hypergraph by removing the first node and all its inputs, and by taking the outputs of the node to be inputs of the hypergraph: this is still acyclic and we have not removed any wire connected to another node. By induction hypothesis, this new hypergraph can be written as follows,

$$\text{Atomic}(n_1) \fatsemi \text{Atomic}(n_2) \fatsemi ... \fatsemi \text{Atomic}(n_k).$$

It suffices to check that we constructed it precisely so that composing with $\text{Atomic}(n_0)$ recovers the initial graph: it adds the node n_0, reconnects the output wires to the outputs of n_0, and substitutes them by the inputs of n_1 in the inputs to the hypergraph. □

Lemma A.20 (From Lemma 7.20)**.** *For each physical duoidal signature, $\mathcal{G}$, there exists a signature homomorphism, $u_{\mathcal{G}} \colon \mathcal{G} \to \mathsf{Forget}(\mathsf{PhyString}(\mathcal{G}))$.*

Proof. Each generator, $f \in \mathcal{G}(E_i; E_o)$, gets sent to the string diagram consisting only of a single node labelled by the generator, going from the zetless poset given by $\mathsf{encode}(E_i)$ to the zetless poset given by $\mathsf{encode}(E_o)$. □

Theorem A.21 (From Theorem 7.21)**.** *There exists an adjunction from the category of physical duoidal signatures to the category of strict physical duoidal categories given by physical duoidal string diagrams $\mathsf{PhyString} \colon \mathbf{PhySig} \to \mathbf{PhyDuo}$ and the forgetful functor, $\mathsf{forget} \colon \mathbf{PhyDuo} \to \mathbf{PhySig}$.*

Proof. We will show that the morphisms $u_{\mathcal{G}} \colon \mathcal{G} \to \mathsf{Forget}(\mathsf{PhyString}(\mathcal{G}))$ defined in Lemma 7.20 are universal. Let $f \colon \mathcal{G} \to \mathsf{Forget}(\mathbb{V})$ be any signature homomorphism. We will show that there exists a unique strict physical duoidal functor $F \colon \mathsf{PhyString}(\mathcal{G}) \to \mathbb{V}$ such that $f = u_{\mathcal{G}} \fatsemi \mathsf{Forget}(F)$.

Let us first define the functor on objects. The objects of $\mathsf{PhyString}(\mathcal{G})$ are duoidal expressions labelled by the basic types of the signature (by virtue of Proposition 5.2). As a consequence, $F(E)$ is determined uniquely for each $E \in \mathsf{expr}(\mathcal{G}_t)$.

Let us now define the functor on morphisms. The morphisms of $\mathsf{PhyString}(\mathcal{G})$ are physical hypergraphs, which can be decomposed into atomic hypergraphs (Proposition 7.19). Any atomic hypergraph, $\text{Level}(n)$, consists of a generator, $\mathsf{label}(n)$, tensored and sequenced with parallel wires: it must be mapped to $F(\mathsf{label}(n))$ tensored and sequenced with parallel wires. Finally, because the functor preserves composition, the value on any physical hypergraph is determined by its decomposition into atomic hypergraphs.

It suffices to check that this assignment is well-defined and that it forms a strict physical duoidal functor. Firstly, let us note that, if we were to pick a different ordering of the nodes, then the only nodes that would change ordering are these that are parallel: they cannot share any wire. In that case, by the interchange law for duoidal categories, we know that the value of both interpretations in the target category is the same.

Finally, we need to prove that this forms a strict physical duoidal functor. Note that it must preserve compositions, for the concatenation of decompositions into atomic graphs is a possible decomposition of the composition. It must preserve tensoring and sequencing, for there is always an ordering that gets all of the nodes of one of the hypergraphs first, and there, again, the concatenation of decompositions into atomic hypergraphs (this time with the extra input or output wires of the other hypergraph) is a possible decomposition of the tensoring or sequencing. □

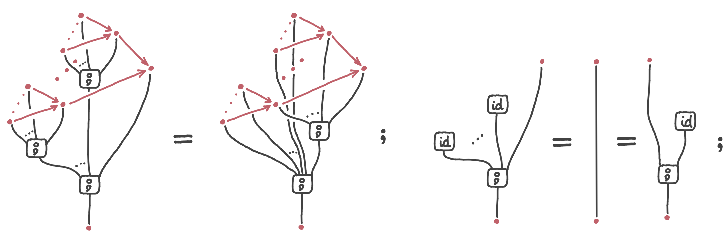


FIGURE 11. Axioms for physical-duoidally enriched multicategories.

Definition A.22 (Opduoidally enriched multicategory, Rajesh [Raj13])**.** An *openriched multicategory*, $\mathbb{C}$, over a physical duoidal category consists of *(i)* a set of objects, $\mathbb{C}_{obj}$; *(ii)* a set of multimorphisms from each list of objects to each single object, $\mathbb{C}(X_1, ..., X_n; Y)$ for each $X_1, ..., X_n, Y \in \mathbb{C}_{obj}$; *(iii)* an identity operation, $\mathrm{i}\colon N \to \mathbb{C}(X; X)$ for each $X \in \mathbb{C}_{obj}$; and *(iv)* a composition operation,

$$(\fatsemi)\colon \Big(\bigotimes_{i=1}^{k} \mathbb{C}(X^i_1, ..., X^i_{n_i}; Y_i)\Big) \triangleright \mathbb{C}(Y_1, ..., Y_m; Z) \to \mathbb{C}(X^1_1, ..., X^1_{n_1}, ..., X^k_1, ..., X^k_{n_k}; Y).$$

These must make all the following diagrams commute.

$$\begin{array}{ccc} \mathbb{C}(X_1, ..., X_n; Y) \triangleright \mathbb{C}(Y; Y) & \xrightarrow{(\fatsemi)} & \mathbb{C}(X_1, ..., X_n; Y) \\ {\scriptstyle \mathrm{id} \triangleright (\mathrm{i})} \uparrow & \nearrow_{\mathrm{id}} & \\ \mathbb{C}(X_1, ..., X_n; Y) & & \end{array}$$

$$\begin{array}{ccc} \big(\bigotimes_{i=1}^{k} \mathbb{C}(X_i; X_i)\big) \triangleright \mathbb{C}(X_1, ..., X_n; Y) & \xrightarrow{(\fatsemi)} & \mathbb{C}(X_1, ..., X_n; Y) \\ {\scriptstyle \big(\bigotimes_{i=1}^{k} \mathrm{i}\big) \triangleright \mathrm{id}} \uparrow & \nearrow_{\mathrm{id}} & \\ \mathbb{C}(X_1, ..., X_n; Y) & & \end{array}$$

$$
\begin{array}{ccc}
\begin{array}{c}\big(\bigotimes_{j=1}^{p}\big(\bigotimes_{i=1}^{m_j}\mathbb{C}(X_1^{i,j},...,X_{n_{i,j}}^{i,j};Y_i^j)\big)\\ \triangleright\mathbb{C}(Y_1^j,...,Y_{m_j}^j;Z_j)\big)\\ \triangleright\mathbb{C}(Z_1,...,Z_p;U)\end{array} & \xrightarrow{(\fatsemi)\triangleright\mathrm{id}} & \begin{array}{c}\big(\bigotimes_{j=1}^{p}\mathbb{C}(X_1^{1,1},...,X_{n_{m_j,j}}^{m_j,j};Z_j)\big)\\ \triangleright\mathbb{C}(Z_1,...,Z_p;U)\end{array}\\
\Big\downarrow{\scriptstyle \ell} & & \Big\downarrow{\scriptstyle (\fatsemi)}\\
\begin{array}{c}\big(\bigotimes_{j=1}^{p}\bigotimes_{i=1}^{m_p}\mathbb{C}(X_1^{i,j},...,X_{n_{i,j}}^{i,j};Y_i^j)\big)\\ \triangleright\big(\bigotimes_{i=1}^{m_p}\mathbb{C}(Y_1^j,...,Y_{m_p}^j;Z_j)\big)\\ \triangleright\mathbb{C}(Z_1,...,Z_p;U)\end{array} & & \\
\Big\downarrow{\scriptstyle \mathrm{id}\triangleright(\fatsemi)} & & \\
\begin{array}{c}\big(\bigotimes_{j=1}^{p}\bigotimes_{i=1}^{m_p}\mathbb{C}(X_1^{i,j},...,X_{n_{i,j}}^{i,j};Y_i^j)\big)\\ \triangleright\mathbb{C}(Y_1^1,...,Y_{m_p}^p;U)\end{array} & \xrightarrow[(\fatsemi)]{} & \mathbb{C}(X_1^{1,1},...,X_{n_{m_p,p}}^{m_p,p};U)
\end{array}
$$

Alternatively, a duoidally-enriched multicategory is given by the string diagrams of Figure 11.