COMPLETENESS OF NOMINAL PROPS

SAMUEL BALCO AND ALEXANDER KURZ

Department of Informatics, University of Leicester, United Kingdom and `https://gdlyrttnap.pl`
e-mail address: sam@definitelynotspam.email

Chapman University, Orange California, USA
e-mail address: akurz@chapman.edu

Abstract. We introduce nominal string diagrams as string diagrams internal in the category of nominal sets. This leads us to define nominal PROPs and nominal monoidal theories. We show that the categories of ordinary PROPs and nominal PROPs are equivalent. This equivalence is then extended to symmetric monoidal theories and nominal monoidal theories, which allows us to transfer completeness results between ordinary and nominal calculi for string diagrams.

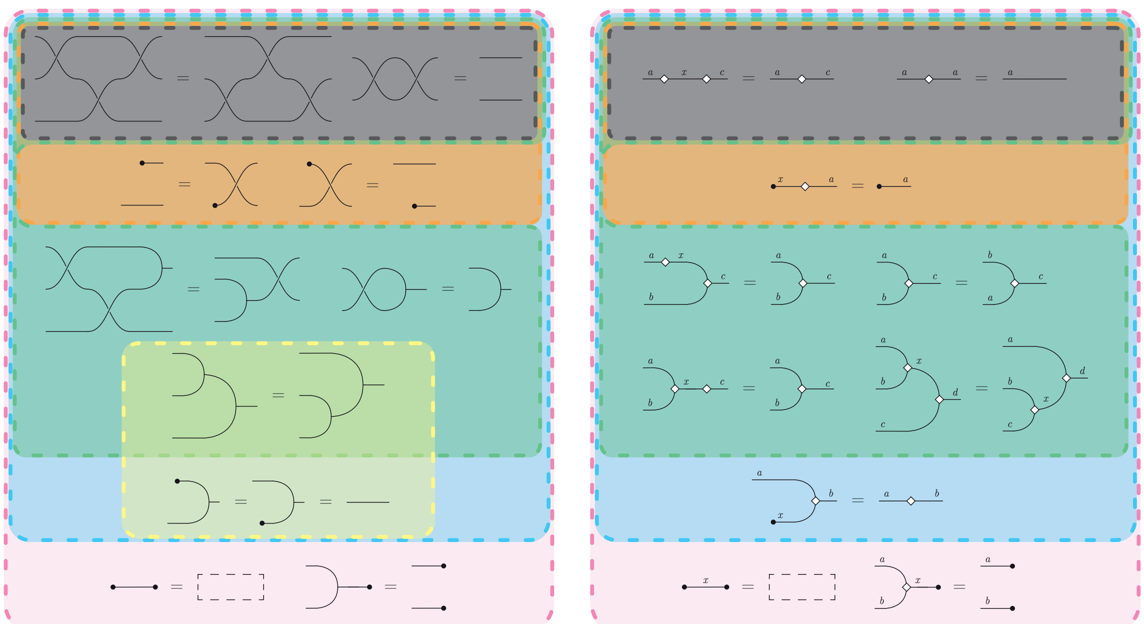
Contents

Preprint submitted to
Logical Methods in Computer Science

1. Introduction

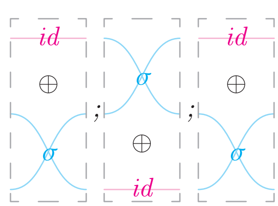
The purpose of this work is to give rigorous foundations to string diagrams with named wires. To achieve this, we follow the wide-spread practice of modeling string diagrams as **pro**duct and **p**ermuntation categories, or PROPs for short. The novel feature of our approach is to internalise PROPs in the category Nom of nominal sets. We then show that the category of ordinary PROPs is equivalent to the category of PROPs internal in Nom, or, briefly, that ordinary PROPs are equivalent to nominal PROPs.

This formalises an observation familiar to users of string diagrams, namely that we can afford ourselves the practical convenience of named wires without departing from the mathematical convenience of canonically ordered wires. To illustrate this consider the following picture.

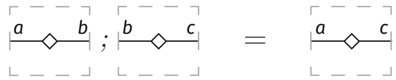


On the left, we see equations between ordinary string diagrams. Sequential composition of two diagrams connects the wires respecting their order.

For example, the top left most diagram is actually a composition of several smaller diagrams:



Mathematically, a collection of wires is an ordinal. On the right we see nominal string diagrams. A collection of wires is a set of 'names'. Sequential composition connects wires that have the same name.



Let us look at the uppermost equations (on grey background). For ordinary string diagrams on the left, we need axioms that equate all diagrams that represent the same permutations, since the order of wires is important. On the right, we see that we do not have any equations with wire crossings. This reflects that invariance under permutations for nominal string diagrams is inherited from nominal sets. Intuitively, while in the ordinary case wires are lists, in the nominal case wires are sets.

The third row (on green background) contains the equations stating that the binary operation in question (we may call it a "cup") is commutative and associative and interacts in the expected way with wire-crossings (on the left) and with renamings (on the right).

Apart from proving that the PROPs defined by the equations on the left are equivalent to the nominal PROPs axiomatised by the equations on the right, we are also interested in a procedure translating ordinary equations into nominal ones, and vice versa. This needs some care as the equations involving wire-crossings (on the left) and renamings (on the right) look quite different.

Our main results are transfer theorems that allow us not only to translate a complete symmetric monoidal theory (as illustrated on the left) to a nominal monoidal theory (as shown on the right), but also to transfer completeness. We also show the other direction from the nominal to the ordinary side.

As a result of our study we obtain the equivalence of symmetric and nominal monoidal theories and can now use named wires in a completely rigorous way, transerring results back and forth. We will remark on possible applications of nominal PROPs in the conclusions.

This is an extended version of [BK19]. In particular, Sections 6.5-6.8 are new.

Hyper-links. Due to its syntactic nature, this paper introduces a large number of technical definitions and notations, for example: PROP, nPROP, SMT, NMT, $\mathbb{F}$, n$\mathbb{F}$, Trm, nTrm, Prop, $\mathcal{Th}$, NOM, ORD, box, dia, **NOM**, **ORD**, nfNmt, nfSmt. To make the paper easier to read, these notions are hyper-linked in the electronic version.

Acknowledgments. We started to work on this project inspired by Pawel Sobocinski's course on Graphical Linear Algebra at MGS in Leicester 2016. Over the years we profited from discussions with Fredrik Dahlqvist, Giuseppe Greco, Bart Jacobs, Peter Jipsen, Samuel Mimram, Drew Moshier, Alessandra Palmigiano, David Pym, Mike Shulman, Georg Struth, Apostolos Tzimoulis and Fabio Zanasi who all influenced the paper in some way. Special thanks go to Chad Nester for pointing out the important paper by Blute et al [BCST96], which we had missed, and to Thomas Streicher who helped us with questions on fibrations and internal categories.

2. Related Work

We divide discussion of related work into string diagrams and nominal sets.

String Diagrams. Whilst somewhat difficult to tell with certainty, arguably the first formal definition of string diagrams appears in the habilitation thesis of Günter Hotz [Hot65]. However, forms of diagrammatic reasoning in areas such as knot theory have much earlier origins (see [Prz98] for a nice historical summary). Definitions of string diagrams have also been introduced, amongst others, by Penrose [Pen71], Joyal & Street [JS93, JS91] and have cropped up in presentations of sequent calculi [CS18], linear logic as proof nets [Gir87, Mel06], bigraphs [Mil06], signal flow diagrams in control theory [Mas53] and network theory [BSZ15] as well as in areas such as quantum physics and computing [CK17].

All of these formalisms are underpinned by the same category theory, namely that of (symmetric) monoidal categories, specifically **pro**duct and **p**ermuntation categories called PROPs for short, introduced by MacLane [ML65]. For an overview of classic/single sorted string diagrams see [Sel10].

Whilst our work is novel in its presentation of nominal string diagrams as monoidal categories internal in Nom, we are by no means the first to generalise PROPs to a multi-sorted or nominal settings. Indeed, even (one of) the earliest papers on string diagrams, namely that of Roger Penrose [Pen71], already introduces "nominal" string diagrams where the wires of his pictures are given labels. Amongst later works, a commonly seen variation to ordinary string diagrams is the notion of colored props [HR14, Zan17]. This generalisation from one-sorted to many-sorted PROPs is orthogonal to our generalisation to nominal PROPs.

Finally, we must mention the work of Blute et al. [BCST96], which is similar in many aspects to our work, especially in the use of the diamond notation $\langle -] - [- \rangle$, which we arrived at independently from the authors. We also add the converse $[- \rangle - \langle -]$ and work out the axioms under which they are inverse to each other.

Another paper in similar spirit, by Ghica and Lopez [GL17], introduces a version of nominal string diagrams by explicitly introducing names and binders for ordinary string diagrams.

Nominal Sets. Nominal sets were introduced by Gabbay and Pitts in [GP] and are are based on Fraenkel-Mostowski set theory in which sets are equipped with an additional permutation action. Equivalent formulations based on presheaves [Hof99, FPT99] and named sets [MP98] were introduced around same time.

All three approaches have their own advantages. The presheaf approach explicitly types terms by their sets of free names and recognises quantifiers as adjoints as in Lawvere's hyperdoctrines [Law69]. Named sets give finite presentations to infinite orbits and have proved to be at suitable basis for implementing generalisations of algorithms known from automata such as partition refinement [FMT05] or Angluin's algorithm [MSS$^+$17].

Nominal sets are designed to make minimal modifications to ordinary set theory and are thus well suited to internalising established areas of mathematics. Gabbay and Pitts showed that λ-calculus terms up to α-equivalence form a term-algebra not in sets but in nominal sets. Since names and name binding play a fundamental role in programming languages, several areas of theoretical computer science have been internalised in nominal sets as witnessed for example by work on universal algebra [GM09, Pet11], domain theory [TW09, LP14], Stone duality [GLP11, Pet11] and automata theory [BKL11].

This paper adds new methodology to this line of research by internalising monoidal categories in nominal sets, where nominal sets are themselves taken to be a monoidal category with respect to the so-called separating tensor. This allows us to recast categories with a partial monoidal operation as internal categories with a total monoidal operation.

3. Preliminaries

From a technical point of view, this paper can be understood as bringing together existing work on string diagrams, or, more specifically, PROPs and nominal sets. In fact, we will be developing the beginnings of a theory of PROPs internal in the category of nominal sets. In this section we review preliminaries on string diagrams and nominal sets.

3.1. **String Diagrams, SMTs and PROPs.**

String diagrams are a 2-(or higher)-dimensional notation for monoidal categories [JS91]. Their algebraic theory can be formalised by PROPs as defined by MacLane [ML78]. There is also the weaker notion by Lack [Lac04], see Remark 2.9 of Zanasi [Zan15] for a discussion.

A PROP (**pro**ducts and **p**ermutation category) is a symmetric strict monoidal category, with natural numbers as objects, where the monoidal tensor $\oplus$ is addition. Moreover, PROPs, along with strict symmetric monoidal identity-on-objects functors form the category PROP. A PROP contains all bijections between numbers as they can be be generated from the symmetry (twist) $\sigma : 1 \oplus 1 \to 1 \oplus 1$ and from the parallel composition $\oplus$ and sequential composition ; (which we write in diagrammatic order). We denote by $\sigma_{n,m}$ the canonical symmetry $n \oplus m \to m \oplus n$. Functors between PROPs preserve bijections.

PROPs can be presented in algebraic form by operations and equations as *symmetric monoidal theories* (SMTs) [Zan15].

An SMT (Σ, E) has a set Σ of generators, where each generator $\gamma \in \Sigma$ is given an arity m and co-arity n, usually written as $\gamma : m \to n$ and a set E of equations, which are pairs of Σ-terms. The set of all Σ-Terms is denoted by $\mathsf{Trm}(\Sigma)$. Σ-terms can be obtained by composing generators in Σ with the unit $id : 1 \to 1$ and symmetry $\sigma : 2 \to 2$, using either the parallel or sequential composition (see Figure 1). Equations E are pairs of Σ-terms with the same arity and co-arity.

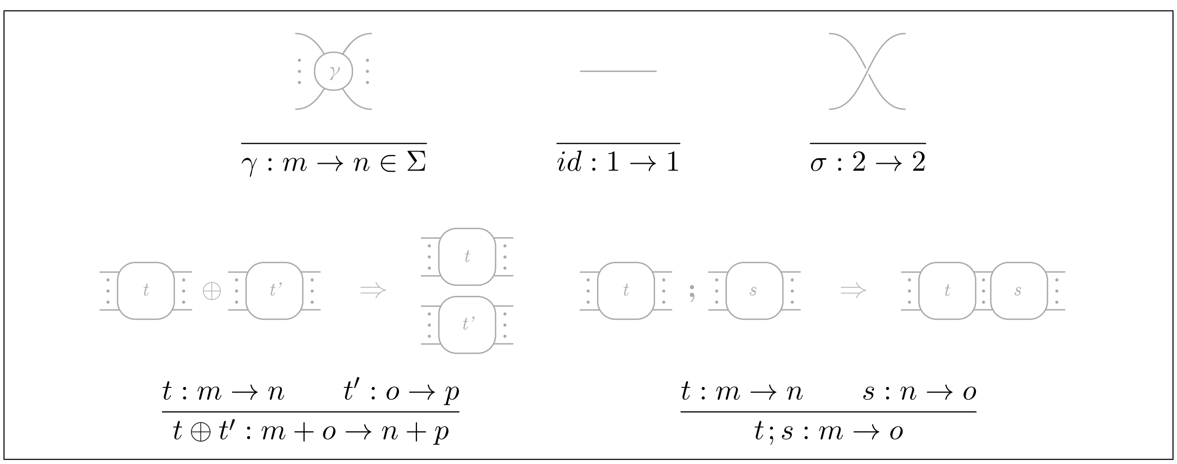


Figure 1: SMT Terms (Trm)

Given an SMT $\langle \Sigma, E \rangle$, we can freely generate a PROP, by taking Σ-terms as arrows, modulo the equations **SMT**, which are:

- the equations stating that compositions ; and $\oplus$ form monoids

- the equations of Figure 2
- the equations E

$$\sigma_{1,1} \,;\, \sigma_{1,1} = id_2 \tag{SMT-sym}$$

$$(s\,;t) \oplus (u\,;v) = (s \oplus u)\,;(t \oplus v) \tag{SMT-ch}$$

$$\frac{s : m \to n \qquad t : o \to p}{(s \oplus t)\,;\sigma_{n,p} = \sigma_{m,o}\,;(t \oplus s)} \tag{SMT-nat}$$

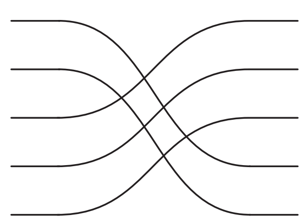
Figure 2: Equations of symmetric monoidal categories

PROPs have a nice 2-dimensional notation, where sequential composition is horizontal composition of diagrams, and parallel/tensor composition is vertical stacking of diagrams (see Figure 1).

We now present the SMTs of bijections B, injections I, surjections S, functions F, partial functions P, relations R and monotone maps M.[1] The diagram in Figure 3 shows the corresponding operations and equations.

To ease comparison with the corresponding nominal monoidal theories in Figure 8, we also added a striped background to the equations with wire-crossings, since they are already implied by the naturality of symmetries (SMT-nat). These are equations that are part of the definition of a PROP in the sense of MacLane [ML78] but not in the sense of Lack [Lac04]. The right-hand equation for bijections B is (SMT-sym) and holds in all symmetric monoidal theories. We list it here to emphasise the difference with Figure 8.

Example 3.1. The SMT of functions F presents the category $\mathbb{F}$ which has natural numbers as objects, all functions as arrows and coproducts as $\oplus$. it is important to note that $\oplus$ is symmetric, but not commutative, with, for example, the symmetry $\sigma_{2,3}$ being depicted as



We will see later that the category n$\mathbb{F}$ of nominal finite functions, see Examples 3.2, 4.2 and 5.6, has a commutative tensor.

[1]The theory of monotone maps M does not include equations involving the symmetry σ and is in fact presented by a so-called PRO rather than a PROP. However, in this paper we will only be dealing with theories presented by PROPs (the reason why this is the case is illustrated in the proof of Proposition 6.6).

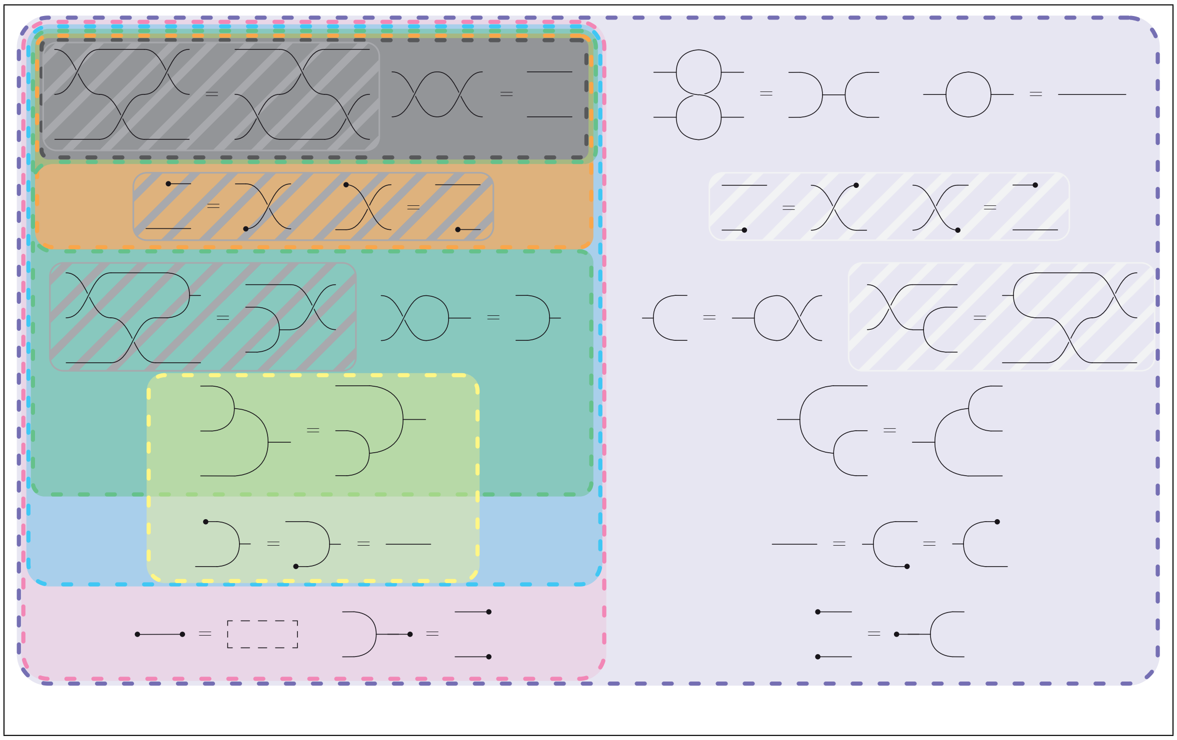


Figure 3: Symmetric monoidal theories (compiled from [Laf03])

3.2. **Nominal Sets.**

Let $\mathcal{N}$ be a countably infinite set of 'names' or 'atoms'. Let $\mathfrak{S}$ be the group of finite[2] permutations $\mathcal{N} \to \mathcal{N}$. An element $x \in X$ of a group action $\mathfrak{S} \times X \to X$ is *supported* by $S \subseteq \mathcal{N}$ if $\pi \cdot x = x$ for all $\pi \in \mathfrak{S}$ and $x \in S$ (i.e. π restricted to S is the identity). A group action $\mathfrak{S} \times X \to X$ where all elements of X have finite support is called a *nominal set.*

We write $\mathsf{supp}(x)$ for the minimal support of x and Nom for the category of nominal sets, which has as maps the *equivariant* functions, that is, those functions that respect the permutation action. As our running example, we choose the category of simultaneous substitutions:

Example 3.2. We denote by $\mathsf{n}\mathbb{F}$ the category of finite subsets of $\mathcal{N}$ as objects and with all functions as arrows. While $\mathsf{n}\mathbb{F}$ is a category, it also carries additional nominal structure. In particular, both the set of objects and the set of arrows are nominal sets with $\mathsf{supp}(A) = A$ and $\mathsf{supp}(f) = A \cup B$ for $f : A \to B$. The categories of injections $\mathsf{n}\mathbb{I}$, surjections $\mathsf{n}\mathbb{S}$, bijections $\mathsf{n}\mathbb{B}$, partial functions $\mathsf{n}\mathbb{P}$ and relations $\mathsf{n}\mathbb{R}$ are further examples along the same lines.

One of the aims of this paper is to exhibit and analyse further structure of this example. For example, all bijections in $\mathsf{n}\mathbb{F}$ can be built from basic functions

$$[a \mapsto b] : \{a\} \to \{b\}$$

and the monoidal operations of sequential and parallel composition as for example in

$$[a{\mapsto}b] \,;\, [b{\mapsto}c] = [a{\mapsto}c] \qquad\qquad [a{\mapsto}b] \uplus [c{\mapsto}d] = [a{\mapsto}b, c{\mapsto}d].$$

[2]A permutation is called finite if it is generated by finitely many transpositions.

We call $\uplus$ the tensor, or the monoidal or vertical or parallel composition. Semantically, the simultaneous substitution on the right-hand side above, will correspond to the function $f : \{a, c\} \to \{b, d\}$ satisfying $f(a) = b$ and $f(c) = d$.

Importantly, parallel composition of simultaneous substitutions is partial. For example, $[a{\mapsto}b] \uplus [a{\mapsto}c]$ is undefined, since there is no function $\{a\} \to \{b, c\}$ that maps a simultaneously to both b and c.

Remark 3.3. Let us make some remarks about the advantages of a 2-dimensional calculus for simultaneous substitutions over a 1-dimensional calculus one. A calculus of substitutions can be understood as an algebraic representation of the category $\mathsf{n}\mathbb{F}$ of finite subsets of $\mathcal{N}$. In a 1-dimensional calculus, operations $[a{\mapsto}b]$ have to be indexed by finite sets S

$$[a{\mapsto}b]_S : S \cup \{a\} \to S \cup \{b\}$$

for sets S with $a, b \notin S$. On the other hand, in a 2-dimensional calculus with an explicit operation $\uplus$ for set union, indexing with subsets S is unnecessary. Moreover, while the swapping

$$\{a, b\} \to \{a, b\}$$

in the 1-dimensional calculus needs an auxiliary name such as c in $[a{\mapsto}c]_{\{b\}} \,;\, [b{\mapsto}a]_{\{c\}} \,;\, [c{\mapsto}a]_{\{b\}}$ it is represented in the 2-dimensional calculus directly by

$$[a{\mapsto}b] \uplus [b{\mapsto}a]$$

Finally, while it is possible to write down the equations and rewrite rules for the 1-dimensional calculus, it does not appear as particularly natural. In particular, only in the 2-dimensional calculus, will the swapping have a simple normal form such as $[a{\mapsto}b] \uplus [b{\mapsto}a]$ (unique up to commutativity of $\uplus$).

4. Internal monoidal categories

We introduce the, to our knowledge, novel notion of an internal monoidal category. Given a symmetric monoidal category $(\mathcal{V}, I, \otimes)$ with finite limits, we are interested in categories $\mathbb{C}$, internal in $\mathcal{V}$, that carry a monoidal structure not of type $\mathbb{C} \times \mathbb{C} \to \mathbb{C}$ but of type $\mathbb{C} \circledast \mathbb{C} \to \mathbb{C}$, where $\circledast$ is a certain lifting, defined below, of $\otimes$ to categories internal in $\mathcal{V}$.

Before going into the technicalities, let us resume Example 3.2 from the previous section in order to explain why we want to work with the lifted tensor product $\circledast$. First, we give the example of the monoidal category in which we will internalise.

Example 4.1. The symmetric monoidal (closed) category $(\mathsf{Nom}, 1, *)$ of nominal sets with the separated product $*$ is defined as follows [Pit13]. 1 is the terminal object, i.e. a singleton with empty support. The separated product of two nominal sets is defined as $A * B = \{(a, b) \in A \times B \mid \mathsf{supp}(a) \cap \mathsf{supp}(b) = \varnothing\}$. Nom is also a symmetric monoidal (closed) category wrt the cartesian product. The two monoidal structures are related by injections $j_{A,B} : A * B \to A \times B$ natural in A and B.

Next, let us go back to the category $\mathsf{n}\mathbb{F}$ of Example 3.2, which is the category we want to internalise in $(\mathsf{Nom}, 1, *)$. Recall that while parallel composition in the category $\mathbb{F}$ of Example 3.1 is the coproduct

$$\oplus : \mathbb{F} \times \mathbb{F} \to \mathbb{F}$$

we have seen in Example 3.2 that the parallel composition in $\mathsf{n}\mathbb{F}$ is a partial operation

$$\uplus : \mathsf{n}\mathbb{F} \times \mathsf{n}\mathbb{F} \to \mathsf{n}\mathbb{F}.$$

One way to proceed would be to develop a theory of partial monoidal categories. But in our case, it seems more elegant to notice that $\uplus$ is a total operation

$$\uplus : \mathsf{n}\mathbb{F} * \mathsf{n}\mathbb{F} \to \mathsf{n}\mathbb{F}$$

since the separated product $*$ accounts for the constraint that $f \uplus g$ is defined iff the domain and codomain of f and g are disjoint. The purpose of this section is to define the notion of internal monoidal category and to show that $(\mathsf{n}\mathbb{F}, \varnothing, \uplus)$ is a monoidal category internal in $(\mathsf{Nom}, 1, *)$ with $\uplus$ of type

$$\uplus : \mathsf{n}\mathbb{F} \circledast \mathsf{n}\mathbb{F} \to \mathsf{n}\mathbb{F}.$$

where $\circledast$ is the lifting of $*$ to categories internal in Nom.

The task, then, of this section is to extend $* : \mathsf{Nom} \times \mathsf{Nom} \to \mathsf{Nom}$ to

$$\circledast : \mathsf{Cat}(\mathsf{Nom}) \times \mathsf{Cat}(\mathsf{Nom}) \to \mathsf{Cat}(\mathsf{Nom})$$

where we denote by $\mathsf{Cat}(\mathsf{Nom})$, the category of small internal categories[3] in Nom. See also Remark 4.10 for a summary of notation. At the end of the section, we will have justified our main example:

Example 4.2. The category $\mathsf{n}\mathbb{F}$ is an internal monoidal category with monoidal operation given by $A \uplus B = A \cup B$ if A, B are disjoint and $f \uplus f' = f \cup f'$ if A, A' and B, B' are disjoint where $f : A \to B$ and $f' : A' \to B'$. The unit is the empty set $\varnothing$.

Since $\otimes$ does not need to preserve finite limits, we cannot expect that defining $(\mathbb{C} \boldsymbol{\otimes} \mathbb{C})_0 = \mathbb{C}_0 \otimes \mathbb{C}_0$ and $(\mathbb{C} \boldsymbol{\otimes} \mathbb{C})_1 = \mathbb{C}_1 \otimes \mathbb{C}_1$ results in $\mathbb{C} \boldsymbol{\otimes} \mathbb{C}$ being an internal category. The purpose of the next example is to show what goes wrong in a concrete instance.

Example 4.3. Following on from the previous example, given $(\mathsf{Nom}, 1, *)$, we define, just for the purposes of this example, a binary operation $\mathsf{n}\mathbb{F} \circledast \mathsf{n}\mathbb{F}$ as $(\mathsf{n}\mathbb{F} \circledast \mathsf{n}\mathbb{F})_0 = \mathsf{n}\mathbb{F}_0 * \mathsf{n}\mathbb{F}_0$ and $(\mathsf{n}\mathbb{F} \circledast \mathsf{n}\mathbb{F})_1 = \mathsf{n}\mathbb{F}_1 * \mathsf{n}\mathbb{F}_1$. Then $\mathsf{n}\mathbb{F} \circledast \mathsf{n}\mathbb{F}$ cannot be equipped with the structure of an internal category. Indeed, assume for a contradiction that there was an appropriate pullback $(\mathsf{n}\mathbb{F} \circledast \mathsf{n}\mathbb{F})_2$ and arrow $\mathbf{comp}$ such that the two diagrams commute:

$$\begin{tikzcd}
(\mathsf{n}\mathbb{F} \circledast \mathsf{n}\mathbb{F})_2 \arrow[rr, "\mathbf{comp}"] \arrow[d, shift right, "\pi_1"'] \arrow[d, shift left, "\pi_2"] & & \mathsf{n}\mathbb{F}_1 * \mathsf{n}\mathbb{F}_1 \arrow[d, shift right, "\mathbf{dom}"'] \arrow[d, shift left, "\mathbf{cod}"] \\
\mathsf{n}\mathbb{F}_1 * \mathsf{n}\mathbb{F}_1 \arrow[rr, shift left, "\mathbf{dom}"] \arrow[rr, shift right, "\mathbf{cod}"'] & & \mathsf{n}\mathbb{F}_0 * \mathsf{n}\mathbb{F}_0
\end{tikzcd}$$

Let $\delta_{xy} : \{x\} \to \{y\}$ be the unique function in $\mathsf{n}\mathbb{F}$ of type $\{x\} \to \{y\}$. Then $((\delta_{ac}, \delta_{bd}), (\delta_{cb}, \delta_{da}))$, which can be depicted as

$$\begin{array}{c}
\{a\} \xrightarrow{\delta_{ac}} \{c\} \xrightarrow{\delta_{cb}} \{b\} \\
\{b\} \xrightarrow[\delta_{bd}]{} \{d\} \xrightarrow[\delta_{da}]{} \{a\}
\end{array}$$

[3]The necessary notation from internal categories is reviewed in Appendix A.

is in the pullback $(\mathsf{n}\mathbb{F} \circledast \mathsf{n}\mathbb{F})_2$, but there is no $\mathbf{comp}$ such that the two squares above commute, since $\mathbf{comp}((\delta_{ac}, \delta_{bd}), (\delta_{cb}, \delta_{da}))$ would have to be $(\delta_{ab}, \delta_{ba})$. But since δ_{ab} and δ_{ba} do not have disjoint support (since $\mathsf{supp}(\delta_{ab}) = \mathsf{supp}(\delta_{ba}) = \{a, b\}$), this set cannot be in $\mathsf{n}\mathbb{F}_1 * \mathsf{n}\mathbb{F}_1$. $\square$

In the example, the attempt to define a tensor on the category $\mathbb{C} = \mathsf{n}\mathbb{F}$ internal in $(\mathcal{V}, I, \otimes) = (\mathsf{Nom}, 1*)$ via $(\mathbb{C} \otimes \mathbb{C})_1 = \mathbb{C}_1 \otimes \mathbb{C}_1$ fails. To ask for pairs of arrows in $(\mathbb{C} \otimes \mathbb{C})_1$ to have disjoint support is too much. Instead we should be looking for a general categorical definition that restricts to those pairs of arrows in $(\mathbb{C} \otimes \mathbb{C})_1$ that have disjoint domains and disjoint codomains.

The solution to the problem consists in assuming that the given symmetric monoidal category with finite limits $(\mathcal{V}, 1, \otimes)$ is semi-cartesian (aka affine), that is, the unit 1 is the terminal object. In such a category there are canonical arrows natural in A and B (dropping the subscripts of $j_{A,B}$)

$$j : A \otimes B \to A \times B$$

and we can use them to define arrows $j_1 : (\mathbb{C} \circledast \mathbb{C})_1 \to \mathbb{C}_1 \times \mathbb{C}_1$ that give us the right notion of tensor on arrows. From our example $\mathsf{n}\mathbb{F}$ above, we know that we want arrows (f, g) to be in $(\mathbb{C} \circledast \mathbb{C})_1$ if $\mathbf{dom}(f) \cap \mathbf{dom}(g) = \varnothing$ and $\mathbf{cod}(f) \cap \mathbf{cod}(g) = \varnothing$. We now turn this into a category theoretic definition, which is in fact an instance of the general and well-known construction of pulling back an internal category $\mathbb{C}$ along an arrow $j : X \to \mathbb{C}_0$. This construction yields an internal category $\mathbb{X}$ with $\mathbb{X}_0 = X$ and $\mathbb{X}_1$ the pullback of $\langle \mathbf{dom}_{\mathbb{C}}, \mathbf{cod}_{\mathbb{C}} \rangle$ along $j \times j$, or, equivalently, the limit in the following diagram

$$\begin{tikzcd}
& \mathbb{X}_1 \arrow[rr, "j_1"] \arrow[ddl, "\mathbf{dom}_{\mathbb{X}}"'] \arrow[dr, dotted, "\mathbf{cod}_{\mathbb{X}}"] & & \mathbb{C}_1 \arrow[ddl, "\mathbf{dom}_{\mathbb{C}}"'] \arrow[dr, dotted, "\mathbf{cod}_{\mathbb{C}}"] & \\
& & \mathbb{X}_0 \arrow[rr, dotted, "j"] & & \mathbb{C}_0 \\
\mathbb{X}_0 \arrow[rr, "j"'] & & \mathbb{C}_0 & &
\end{tikzcd}$$

which we abbreviate to

$$\begin{tikzcd}
\mathbb{X}_1 \arrow[r, "j_1"] \arrow[d, shift right, "\mathbf{dom}_{\mathbb{X}}"'] \arrow[d, shift left, "\mathbf{cod}_{\mathbb{X}}"] & \mathbb{C}_1 \arrow[d, shift right, "\mathbf{dom}_{\mathbb{C}}"'] \arrow[d, shift left, "\mathbf{cod}_{\mathbb{C}}"] \\
\mathbb{X}_0 \arrow[r, "j"'] & \mathbb{C}_0
\end{tikzcd} \tag{4.1}$$

Next we define $i : \mathbb{X}_0 \to \mathbb{X}_1$ as the arrow into the limit $\mathbb{X}_1$ given by

$$\begin{tikzcd}
\mathbb{X}_0 \arrow[dr, dotted, "i_{\mathbb{X}}"] \arrow[drr, bend left, "i_{\mathbb{C}} \circ j"] \arrow[ddr, bend right, "id"] \arrow[ddr, bend right=40, "id"'] & & \\
& \mathbb{X}_1 \arrow[r, "j_1"] \arrow[d, shift right, "\mathbf{dom}_{\mathbb{X}}"'] \arrow[d, shift left, "\mathbf{cod}_{\mathbb{X}}"] & \mathbb{C}_1 \arrow[d, shift right, "\mathbf{dom}_{\mathbb{C}}"'] \arrow[d, shift left, "\mathbf{cod}_{\mathbb{C}}"] \\
& \mathbb{X}_0 \arrow[r, "j"'] & \mathbb{C}_0
\end{tikzcd} \tag{4.2}$$

from which one reads off

$$\mathbf{dom}_{\mathbb{X}} \circ i_{\mathbb{X}} = id_{\mathbb{X}_0} = \mathbf{cod}_{\mathbb{X}} \circ i_{\mathbb{X}}$$

Next, $\mathbb{X}_2$ is the pullback

$$\begin{tikzcd}
& \mathbb{X}_2 \arrow[dl, "\pi_{\mathbb{X}1}"'] \arrow[dr, "\pi_{\mathbb{X}2}"] & \\
\mathbb{X}_1 \arrow[dr, "\mathbf{cod}_{\mathbb{X}}"'] & & \mathbb{X}_1 \arrow[dl, "\mathbf{dom}_{\mathbb{X}}"] \\
& \mathbb{X}_0 &
\end{tikzcd}$$

Recalling the definition of j_1 from (4.1), there is also a corresponding $j_2 : \mathbb{X}_2 \to \mathbb{C}_2$ due to the fact that the product of pullbacks is a pullback of products.

$$\begin{tikzcd}
& \mathbb{X}_2 \arrow[rrr, "j_2"] \arrow[dl, dashed, "\pi_{\mathbb{X}1}"'] \arrow[dr, dashed, "\pi_{\mathbb{X}2}"] & & & \mathbb{C}_2 \arrow[dl, dashed, "\pi_{\mathbb{X}1}"'] \arrow[dr, dashed, "\pi_{\mathbb{C}2}"] & \\
\mathbb{X}_1 \arrow[rrr, dotted, bend left, "j_1"] \arrow[dr, "\mathbf{cod}_{\mathbb{X}}"'] & & \mathbb{X}_1 \arrow[rrr, dotted, bend right, "j_1"'] \arrow[dl, "\mathbf{dom}_{\mathbb{X}}"] & \mathbb{C}_1 \arrow[dr, "\mathbf{cod}_{\mathbb{C}}"'] & & \mathbb{C}_1 \arrow[dl, "\mathbf{dom}_{\mathbb{C}}"] \\
& \mathbb{X}_0 \arrow[rrr, "j"'] & & & \mathbb{C}_0 &
\end{tikzcd} \tag{4.3}$$

Recall the definition of the limit $\mathbb{X}_1$ from (4.1). Then $\mathbf{comp}_{\mathbb{X}} : \mathbb{X}_2 \to \mathbb{X}_1$ is the arrow into $\mathbb{X}_1$

$$\begin{tikzcd}
\mathbb{X}_2 \arrow[dr, dotted, "\mathbf{comp}_{\mathbb{X}}"] \arrow[drr, bend left, "\mathbf{comp}_{\mathbb{C}} \circ j_2"] \arrow[ddr, bend right=40, "\mathbf{dom}_{\mathbb{X}} \circ \pi_{\mathbb{X}1}"'] \arrow[ddr, bend right=25, "\mathbf{cod}_{\mathbb{X}} \circ \pi_{\mathbb{X}2}"] & & \\
& \mathbb{X}_1 \arrow[r, "j_1"] \arrow[d, shift right, "\mathbf{dom}_{\mathbb{X}}"'] \arrow[d, shift left, "\mathbf{cod}_{\mathbb{X}}"] & \mathbb{C}_1 \arrow[d, shift right, "\mathbf{dom}_{\mathbb{C}}"'] \arrow[d, shift left, "\mathbf{cod}_{\mathbb{C}}"] \\
& \mathbb{X}_0 \arrow[r, "j"'] & \mathbb{C}_0
\end{tikzcd} \tag{4.4}$$

from which one reads off

$$\mathbf{dom}_{\mathbb{X}} \circ \mathbf{comp}_{\mathbb{X}} = \mathbf{dom}_{\mathbb{X}} \circ \pi_{\mathbb{X}1} \qquad \mathbf{cod}_{\mathbb{X}} \circ \mathbf{comp}_{\mathbb{X}} = \mathbf{cod}_{\mathbb{X}} \circ \pi_{\mathbb{X}2} \qquad j_1 \circ \mathbf{comp}_{\mathbb{X}} = \mathbf{comp}_{\mathbb{C}} \circ j_2$$

and the remaining equations $\mathbf{comp}_{\mathbb{X}} \circ \langle i_{\mathbb{X}} \circ \mathbf{dom}_{\mathbb{X}}, id_{\mathbb{X}_1} \rangle = id_{\mathbb{X}_1} = \mathbf{comp}_{\mathbb{X}} \circ \langle id_{\mathbb{X}_1}, i_{\mathbb{X}} \circ \mathbf{cod}_{\mathbb{X}} \rangle$ are also not difficult to prove.

Finally, in analogy with the definition of j_2 in (4.3), j_3 is defined as the unique arrow into the pullback $\mathbb{C}_3$, where $\mathbb{X}_3$ is defined in the expected way:

$$\begin{tikzcd}
& \mathbb{X}_3 \arrow[rrr, "j_3"] \arrow[dl, dashed, "\mathbf{left}_{\mathbb{X}}"'] \arrow[dr, dashed, "\mathbf{right}_{\mathbb{X}}"] & & & \mathbb{C}_3 \arrow[dl, dashed, "\mathbf{left}_{\mathbb{C}}"'] \arrow[dr, dashed, "\mathbf{right}_{\mathbb{C}}"] & \\
\mathbb{X}_2 \arrow[rrr, dotted, bend left, "j_2"] \arrow[dr, "\pi_{\mathbb{X}2}"'] & & \mathbb{X}_2 \arrow[rrr, dotted, bend right, "j_2"'] \arrow[dl, "\pi_{\mathbb{X}1}"] & \mathbb{C}_2 \arrow[dr, "\pi_{\mathbb{C}2}"'] & & \mathbb{C}_2 \arrow[dl, "\pi_{\mathbb{C}1}"] \\
& \mathbb{X}_1 \arrow[rrr, "j_1"'] & & & \mathbb{C}_1 &
\end{tikzcd} \tag{4.5}$$

This ends the definition of $\mathbb{X}$. Since pulling back internal categories is a well known construction, we skip the verification that $\mathbb{X}$ is an internal category. The reader interested in the technical details may find them in the thesis [Bal20] of the first author. We summarise what we have done so far in

Proposition 4.4. *Given an internal category $\mathbb{C}$ and an arrow $j : X \to \mathbb{C}_0$ there is an internal category $\mathbb{X}$ and an internal functor $\mathbb{j} : \mathbb{X} \to \mathbb{C}$ such that $\mathbb{X}_0 = X$ and $\mathbb{j}_0 = j$.*

To continue our development, we specialise to $\mathbb{j} : \mathbb{C} \otimes \mathbb{C} \to \mathbb{C} \times \mathbb{C}$ available in all semi-cartesian monoidal categories. To make progress, we need to show that $\mathbb{C} \otimes \mathbb{C}$ extends to a functor $\boldsymbol{\otimes}$: $\mathsf{Cat}(\mathcal{V}) \times \mathsf{Cat}(\mathcal{V}) \to \mathsf{Cat}(\mathcal{V})$ and that the $\mathbb{j} : \mathbb{C} \boldsymbol{\otimes} \mathbb{C} \to \mathbb{C} \times \mathbb{C}$ are natural in $\mathbb{C}$. As usual, the easiest way of proving functoriality and naturality is to exhibit the construction in question, in our case $\boldsymbol{\otimes}$, as a universal one. Thus, the next step is to exhibit the $\mathbb{j} : \mathbb{X} \to \mathbb{C}$ from the proposition as a so-called cartesian lifting, a notion from the theory of fibrations [Jac99, Str99].

Definition 4.5 (Fibration)**.** If $P : \mathcal{W} \to \mathcal{V}$ is a functor, then $\mathbb{j} : \mathbb{X} \to \mathbb{Y}$ is a *cartesian lifting* of $j : X \to P\mathbb{Y}$ if for all $\mathbb{k} : \mathbb{W} \to \mathbb{Y}$ and all $h : P\mathbb{W} \to X$ with $P\mathbb{k} = j \circ h$ there is a unique $\mathbb{h} : \mathbb{W} \to \mathbb{X}$ such that $\mathbb{j} \circ \mathbb{h} = \mathbb{k}$ and $P\mathbb{h} = h$.

$$\begin{tikzcd}
\mathbb{W} \arrow[dr, dotted, "!\mathbb{h}"'] \arrow[drr, "\mathbb{k}"] & & \\
& \mathbb{X} \arrow[r, "\mathbb{j}"'] & \mathbb{Y}
\end{tikzcd}$$

$$\begin{tikzcd}
P\mathbb{W} \arrow[dr, "h"'] \arrow[drr, "P\mathbb{k}=j\circ h"] & & \\
& X \arrow[r, "j"'] & P\mathbb{Y}
\end{tikzcd}$$

Moreover, $P : \mathcal{W} \to \mathcal{V}$ is called a (Grothendieck) *fibration* if all $j : X \to P\mathbb{Y}$ have a cartesian lifting for all $\mathbb{Y}$ in $\mathcal{W}$. If $P : \mathcal{W} \to \mathcal{V}$ is a fibration, the subcategory of $\mathcal{W}$ that has as arrows the arrows $\mathbb{f}$ such that $P\mathbb{f} = id_Y$ is called the *fibre* over Y.

The next lemma is a strengthening of Proposition 4.4.

Lemma 4.6. *Let $\mathcal{V}$ be a category with finite limits. The forgetful functor $\mathsf{Cat}(\mathcal{V}) \to \mathcal{V}$ is a fibration.*

Proof. We have already shown how to lift $j : X \to \mathbb{C}_0$ to $\mathbb{j} : \mathbb{X} \to \mathbb{C}$. One can show that this is a cartesian lifting by drawing out the appropriate diagram. Namely, we have the forgetful functor $(-)_0 : \mathsf{Cat}(\mathcal{V}) \to \mathcal{V}$, which sends an internal category to its "object of objects", an internal category $\mathbb{X}$, $\mathbb{Y}$ and an internal functor $\mathbb{j}$ between them. Given another internal category $\mathbb{W}$ and an internal functor $\mathbb{k} : \mathbb{W} \to \mathbb{Y}$ and an arrow $h : \mathbb{W}_0 \to \mathbb{Y}_0$, s.t. $\mathbb{k}_0 = \mathbb{j}_0 \circ h$, we show there is a unique $\mathbb{h}$, s.t. $\mathbb{k} = \mathbb{j} \circ \mathbb{h}$. This essentially means we need to fill in the following diagram, such that all sub-diagrams commute:

$$\begin{tikzcd}
\mathbb{W}_2 \arrow[d, "\mathbf{comp}"] \arrow[d, bend right, "\pi_1"'] \arrow[d, bend left, "\pi_2"] \arrow[rr, bend left, "\mathbb{k}_2"] & \mathbb{X}_2 \arrow[r, "\mathbb{j}_2"] \arrow[d, "\mathbf{comp}"] \arrow[d, bend right, "\pi_1"'] \arrow[d, bend left, "\pi_2"] & \mathbb{Y}_2 \arrow[d, "\mathbf{comp}"] \arrow[d, bend right, "\pi_1"'] \arrow[d, bend left, "\pi_2"] \\
\mathbb{W}_1 \arrow[d, shift right=2, "\mathbf{dom}_{\mathbb{W}}"'] \arrow[d, shift left=2, "\mathbf{cod}_{\mathbb{W}}"] \arrow[rr, bend right=15, "\mathbb{k}_1"] & \mathbb{X}_1 \arrow[r, "\mathbb{j}_1"] \arrow[d, shift right=2, "\mathbf{dom}_{\mathbb{X}}"'] \arrow[d, shift left=2, "\mathbf{cod}_{\mathbb{X}}"] & \mathbb{Y}_1 \arrow[d, shift right=2, "\mathbf{dom}_{\mathbb{Y}}"'] \arrow[d, shift left=2, "\mathbf{cod}_{\mathbb{Y}}"] \\
\mathbb{W}_0 \arrow[u, "i_{\mathbb{W}}"] \arrow[r, "h"'] \arrow[rr, bend right, "\mathbb{k}_0"'] & \mathbb{X}_0 \arrow[u, "i_{\mathbb{X}}"] \arrow[r, "\mathbb{j}_0"'] & \mathbb{Y}_0 \arrow[u, "i_{\mathbb{Y}}"]
\end{tikzcd}$$

Since our category has all finite limits, we can define $\mathbb{h}_1$ as an arrow into the limit $\mathbb{X}_1$:

$$\begin{tikzcd}
\mathbb{W}_1 \arrow[dr, dotted, "\mathbb{h}_1"] \arrow[drr, bend left, "\mathbb{k}_1"] \arrow[ddr, bend right, shift right, "\mathbf{dom}_{\mathbb{W}} \circ h"'] \arrow[ddr, bend right, shift left, "\mathbf{cod}_{\mathbb{W}} \circ h"] & & \\
& \mathbb{X}_1 \arrow[r, "\mathbb{j}_1"] \arrow[d, shift right, "\mathbf{dom}_{\mathbb{X}}"'] \arrow[d, shift left, "\mathbf{cod}_{\mathbb{X}}"] & \mathbb{Y}_1 \arrow[d, shift right, "\mathbf{dom}_{\mathbb{Y}}"'] \arrow[d, shift left, "\mathbf{cod}_{\mathbb{Y}}"] \\
& \mathbb{X}_0 \arrow[r, "\mathbb{j}_0"'] & \mathbb{Y}_0
\end{tikzcd}$$

We obtain $\mathbb{h}_2$ in a similar fashion, thus getting a unique $\mathbb{h} = (\mathbb{h}_2, \mathbb{h}_1, h)$, for which we have $\mathbb{k} = \mathbb{j} \circ \mathbb{h}$. ☐

While both Proposition 4.4 and Lemma 4.6 allow us to conclude that $\otimes$ on $\mathcal{V}$ can be lifted to an operation $\circledast$ on $\mathsf{Cat}(\mathcal{V})$, we rely on the universal property of Lemma 4.6 to argue that $\circledast$ is functorial and that $\mathbb{j}$ is natural. To show that $\circledast$ is functorial we use that $\otimes : \mathcal{V} \times \mathcal{V} \to \mathcal{V}$ is functorial and that $j : \mathbb{C}_0 \otimes \mathbb{D}_0 \to \mathbb{C}_0 \times \mathbb{D}_0$ is natural in $\mathbb{C}$ and $\mathbb{D}$. In order to lift such natural transformations, which are arrows in the functor category $\mathcal{V}^{\mathsf{Cat}(\mathcal{V}) \times \mathsf{Cat}(\mathcal{V})}$, we use

Lemma 4.7. *If $P : \mathcal{W} \to \mathcal{V}$ is a fibration and $\mathcal{A}$ is a category, then $P^{\mathcal{A}} : \mathcal{W}^{\mathcal{A}} \to \mathcal{V}^{\mathcal{A}}$ is a fibration.*

Proof. $P^{\mathcal{A}}$ is defined via post-composition with P, that is, $P^{\mathcal{A}}(\mathbb{G}) = P \circ \mathbb{G} = P\mathbb{G}$ and $P^{\mathcal{A}}(\eta : \mathbb{G} \to \mathbb{H}) = P\eta$. To show that $P^{\mathcal{A}}$ is a fibration, i.e. that all $j : G \to P\mathbb{H}$ have a *cartesian lifting* $\mathbb{j} : \mathbb{G} \to \mathbb{H}$, we lift j point-wise, using the fact that for all $j_A : GA \to P\mathbb{H}A$ we have $\mathbb{j}_A : \mathbb{G}A \to \mathbb{H}A$ due to P being a fibration. It remains to check that $\mathbb{j}$ is a *cartesian lifting*, that is, given natural transformations $\mathbb{k} : \mathbb{F} \to \mathbb{H}$ and $h : P\mathbb{F} \to G$, such that

$P\mathbb{k} = P\mathbb{j} \circ h$, there is a unique $\mathbb{h}$, s.t. the following diagrams commute

$$
\begin{tikzcd}
\mathbb{F} \arrow[dr, dotted, Rightarrow, "!\mathbb{h}"'] \arrow[drr, Rightarrow, bend left, "\mathbb{k}"] & & \\
& \mathbb{G} \arrow[r, Rightarrow, "\mathbb{j}"'] & \mathbb{H}
\end{tikzcd}
$$

$$
\begin{tikzcd}
P\mathbb{F} \arrow[dr, Rightarrow, "P\mathbb{h}=h"'] \arrow[drr, Rightarrow, bend left, "P\mathbb{k}=P\mathbb{j}\circ h"] & & \\
& G \arrow[r, Rightarrow, "j=P\mathbb{j}"'] & P\mathbb{H}
\end{tikzcd}
$$

Since $\mathbb{k}$, $\mathbb{j}$ and h are natural transformations we have for all $f : A \to B$

$$
\begin{tikzcd}
\mathbb{F}A \arrow[r, "\mathbb{F}f"] \arrow[d, "\mathbb{k}_A"] & \mathbb{F}B \arrow[d, "\mathbb{k}_B"'] \\
\mathbb{H}A \arrow[r, "\mathbb{H}f"] & \mathbb{H}B \\
\mathbb{G}A \arrow[r, "\mathbb{G}f"'] \arrow[u, "\mathbb{j}_A"] & \mathbb{G}B \arrow[u, "\mathbb{j}_B"']
\end{tikzcd}
\qquad
\begin{tikzcd}
P\mathbb{F}A \arrow[r, "P\mathbb{F}f"] \arrow[d, "P\mathbb{k}_A"] \arrow[dd, bend right, "h_A"'] & PFB \arrow[d, "P\mathbb{k}_B"'] \arrow[dd, bend left, "h_B"] \\
P\mathbb{H}A \arrow[r, "P\mathbb{H}f"] & P\mathbb{H}B \\
P\mathbb{G}A \arrow[r, "P\mathbb{G}f"'] \arrow[u, "P\mathbb{j}_A"] & P\mathbb{G}B \arrow[u, "P\mathbb{j}_B"']
\end{tikzcd}
$$

As P is a fibration, we obtain unique $\mathbb{h}_A$ and $\mathbb{h}_B$ for the diagram on the left above, s.t. $P\mathbb{h}_A = h_A$ and $P\mathbb{h}_B = h_B$, thus obtaining a unique natural transformation $\mathbb{h}$, for which $\mathbb{k} = \mathbb{h} \circ \mathbb{j}$. □

Instantiating the lemma with $P = (-)_0 : \mathsf{Cat}(\mathcal{V}) \to \mathcal{V}$ and $\mathcal{A} = \mathsf{Cat}(\mathcal{V}) \times \mathsf{Cat}(\mathcal{V})$, we obtain as a corollary that lifting the tensor $\otimes : \mathcal{V} \times \mathcal{V} \to \mathcal{V}$ to $\boldsymbol{\otimes}$: $\mathsf{Cat}(\mathcal{V}) \times \mathsf{Cat}(\mathcal{V}) \to \mathsf{Cat}(\mathcal{V})$ is functorial:

Theorem 4.8. *Let $(\mathcal{V}, 1, \otimes)$ be a (symmetric) monoidal category with finite limits in which the monoidal unit is the terminal object. Let $P = (-)_0 : \mathsf{Cat}(\mathcal{V}) \to \mathcal{V}$ be the forgetful functor from categories internal in $\mathcal{V}$. Then the canonical arrow $j : \mathbb{C}_0 \otimes \mathbb{D}_0 \to \mathbb{C}_0 \times \mathbb{D}_0$ lifts uniquely to a natural transformation $\mathbb{j} : \mathbb{C} \boldsymbol{\otimes} \mathbb{D} \to \mathbb{C} \times \mathbb{D}$ and the terminal object $1 \in \mathcal{V}$ lifts uniquely to a monoidal unit $\mathbb{I}$ of $\boldsymbol{\otimes}$. Moreover, $(\mathsf{Cat}(\mathcal{V}), \mathbb{I}, \boldsymbol{\otimes})$ inherits from $(\mathcal{V}, 1, \otimes)$ the structure of a (symmetric) monoidal category with finite limits in which the monoidal unit is the terminal object.*

Proof. We continue to use the notation of the proof of Lemma 4.7. Let

$$
\begin{array}{lll}
\mathbb{H} : \mathsf{Cat}(\mathcal{V}) \times \mathsf{Cat}(\mathcal{V}) \to \mathsf{Cat}(\mathcal{V}) & G : \mathsf{Cat}(\mathcal{V}) \times \mathsf{Cat}(\mathcal{V}) \to \mathcal{V} & \mathbb{G} : \mathsf{Cat}(\mathcal{V}) \times \mathsf{Cat}(\mathcal{V}) \to \mathsf{Cat}(\mathcal{V}) \\
\mathbb{H}(\mathbb{A}, \mathbb{B}) = \mathbb{A} \times \mathbb{B} & G(\mathbb{A}, \mathbb{B}) = P\mathbb{A} \otimes P\mathbb{B} & \mathbb{G}(\mathbb{A}, \mathbb{B}) = \mathbb{A} \boldsymbol{\otimes} \mathbb{B}
\end{array}
$$

and $j : G \to P\mathbb{H}$ be the associated natural transformation. We have by definition that $G = P\mathbb{G}$, that is, $\mathbb{A}_0 \otimes \mathbb{B}_0 = (\mathbb{A} \boldsymbol{\otimes} \mathbb{B})_0$. Therefore, j has a cartesian lifting $\mathbb{j} : \mathbb{G} \to \mathbb{H}$ by Lemma 4.7. In particular, $\mathbb{G}$ is a functor. □

In this work we only need internal monoidal categories that are strict. In the same way as a strict monoidal category is a monoid in $(\mathsf{Cat}, \mathbb{1}, \times)$, an internal strict monoidal category is a monoid in $(\mathsf{Cat}(\mathcal{V}), \mathbb{I}, \otimes)$:

Definition 4.9 (Internal monoidal category)**.** Let $(\mathcal{V}, 1, \otimes)$ be a monoidal category with finite limits in which the monoidal unit is the terminal object and let $(\mathsf{Cat}(\mathcal{V}), \mathbb{I}, \otimes)$ be the induced symmetric monoidal category of internal categories in $\mathcal{V}$. A strict internal monoidal category $\mathbb{C}$ is a monoid $(\mathbb{C}, \varnothing, \odot)$ in $(\mathsf{Cat}(\mathcal{V}), \mathbb{I}, \otimes)$.

Remark 4.10. It may be useful to catalogue the different tensors. The first one is the cartesian product $\times$ of categories, with the help of which we define a monoidal product $\otimes$ on a particular categery $\mathcal{V}$ and then lift it to a monoidal product $\otimes$ on the category of categories internal in $\mathcal{V}$. This then allows us to define on an internal category $\mathbb{C}$ a tensor $\odot$, which we also call an *internal tensor*:

$$\otimes : \mathcal{V} \times \mathcal{V} \to \mathcal{V}$$
$$\otimes \colon \mathsf{Cat}(\mathcal{V}) \times \mathsf{Cat}(\mathcal{V}) \to \mathsf{Cat}(\mathcal{V})$$
$$\odot : \mathbb{C} \otimes \mathbb{C} \to \mathbb{C}$$

Example 4.11. Picking up Example 4.2 again, for the category $\mathsf{n}\mathbb{F}$ of finite sets of names and functions we choose for $\varnothing$ the empty set and for the internal tensor $\uplus = \odot$ the union of disjoint sets on objects and, on arrows, the union of functions with both disjoint domains and disjoint codomains.

Remark 4.12. In the classical case where $\mathcal{V} = \mathsf{Cat}$ and both $\otimes$ and $\otimes$ are the cartesian product, the interchange law for $\odot$ follows from $\odot$ being a functor. In the same way, in our more general situation, the interchange law for $\odot$ states that $\odot$ is an internal functor (A.2)

$$\begin{tikzcd}
(\mathbb{C} \otimes \mathbb{C})_2 \arrow[r, "\mathbf{comp}_{\mathbb{C}\otimes\mathbb{C}}"] \arrow[d, "\odot_2"'] & (\mathbb{C} \otimes \mathbb{C})_1 \arrow[d, "\odot_1"] \\
\mathbb{C}_2 \arrow[r, "\mathbf{comp}_{\mathbb{C}}"'] & \mathbb{C}_1
\end{tikzcd}$$

Example 4.13. In the category $(\mathsf{n}\mathbb{F}, \varnothing, \uplus)$ of finite sets of names and functions, see Example 4.2, we have the interchange law

$$(\mathbb{f} \uplus \mathbb{g});(\mathbb{f}' \uplus \mathbb{g}') = (\mathbb{f};\mathbb{f}') \uplus (\mathbb{g};\mathbb{g}')$$

with the right-hand side being defined whenever the left-hand side is.

5. Nominal PROPs and Nominal Monoidal Theories

We introduce nominal PROPs as particular instances of the internal categories of Section 4, taking care to explicate the definition in enough detail so that the reader can follow the rest of the paper without working through the details of the previous section. Nominal PROPs in turn are presented by nominal monoidal theories (NMTs) and it is with this notion that we start. We also take the time to point out that sequential composition in NMTs is a binding operation and induces what we call diagrammatic α-equivalence.

5.1. **Nominal monoidal theories.**

In analogy with symmetric monoidal theories (Section 3.1), a *nominal monoidal theory* (Σ, E) is given by a set Σ of generators and a nominal set E of equations. Each ordinary generator γ gives rise to a set of nominal generators: The set $\mathsf{n}\Sigma$ of nominal generators consists of all $[\boldsymbol{a}\rangle\gamma\langle\boldsymbol{b}] : A \to B$ where $\boldsymbol{a}, \boldsymbol{b}$ are lists of size n, m enumerating the elements of A, B, respectively. The set of nominal generators $\mathsf{n}\Sigma$ is closed under permutations:

$$\pi \cdot [\boldsymbol{a}\rangle\gamma\langle\boldsymbol{b}] : \pi \cdot A \to \pi \cdot B = [\pi(\boldsymbol{a})\rangle\gamma\langle\pi(\boldsymbol{b})]. \qquad (\pi\text{-def})$$

The set of nominal terms or s is given by closing under the operations of Figure 4, which should be compared with Figure 1.

$$\frac{\gamma : m \to n \in \Sigma}{[\boldsymbol{a}\rangle\gamma\langle\boldsymbol{b}] : A \to B} \qquad \frac{}{id_a : \{a\} \to \{a\}} \qquad \frac{}{\delta_{ab} : \{a\} \to \{b\}}$$

$$\frac{t : A \to B \qquad t' : A' \to B'}{t \uplus t' : A \uplus A' \to B \uplus B'} \qquad \frac{t : A \to B \qquad s : B \to C}{t ; s : A \to C} \qquad \frac{t : A \to B}{(a\ b)\, t : (a\ b) \cdot A \to (a\ b) \cdot B}$$

Figure 4: NMT Terms (nTrm)

Every NMT freely generates a monoidal category internal in nominal sets by quotienting the generated terms by equations in E, together with the set **NMT** of equations containing:

- the equations that state that id and ; obey the laws of a category
- the equations stating that $id_\varnothing$ and $\uplus$ are a monoid
- the equations of an internal monoidal category of Figure 5
- the equations of permutation actions of Figure 6
- the equations on the interaction of generators with bijections δ of Figure 7

Comparing **SMT** and **NMT**, we find that the main difference between the equations in Figure 2 and in Figure 7 is that the interchange law for $\uplus$ is required to hold only if both sides are defined and that the two laws involving symmetries are replaced by the commutativity of $\uplus$.

$$t \uplus s = s \uplus t \qquad \text{(NMT-comm)}$$

$$(s\,; t) \uplus (u\,; v) = (s \uplus u)\,; (t \uplus v) \qquad \text{(NMT-ch)}$$

Figure 5: NMT Equations of $\uplus$

$$(a\ b)id_x = id_{(a\ b)\cdot x} \qquad (a\ b)\delta_{xy} = \delta_{(a\ b)\cdot x\ (a\ b)\cdot y} \qquad (a\ b)\gamma = (a\ b) \cdot \gamma$$

$$(a\ b)(x \uplus y) = (a\ b)x \uplus (a\ b)y \qquad (a\ b)(x\,; y) = (a\ b)x\,; (a\ b)y$$

Figure 6: NMT Equations of the permutation actions

$$\delta_{aa} = id_a \qquad\qquad \delta_{ab}\,;\delta_{bc} = \delta_{ac}$$

$$\frac{[a_1,\ldots,a_i,\ldots,a_m\rangle\gamma\langle\boldsymbol{b}] : \{a_i\} \uplus A \to B}{(\delta_{xa_i} \uplus id_A)\,;[a_1,\ldots,a_i,\ldots,a_m\rangle\gamma\langle\boldsymbol{b}] = [a_1,\ldots,x,\ldots,a_m\rangle\gamma\langle\boldsymbol{b}]} \quad \text{(NMT-left)}$$

$$\frac{[\boldsymbol{a}\rangle\gamma\langle b_1,\ldots,b_i,\ldots,b_n] : A \to B \uplus \{b_i\}}{[\boldsymbol{a}\rangle\gamma\langle b_1,\ldots,b_i,\ldots,b_n]\,;(id_B \uplus \delta_{b_i x}) = [\boldsymbol{a}\rangle\gamma\langle b_1,\ldots,x,\ldots,b_n]} \quad \text{(NMT-right)}$$

Figure 7: NMT Equations of δ

For terms to form a nominal set, we need equations between permutations to hold, along with the equations of Figure 6 that specify how permutations act on terms.
All the equations presented in the figures above are routine, with the exception of the last two, specifying the interaction of renamings δ with the generators $[\boldsymbol{a}\rangle\gamma\langle\boldsymbol{b}] \in \Sigma$, which we also depict in diagrammatic form:

Instances of these rules can be seen in Figure 8, where they are distinguished by a striped background.

5.2. Diagrammatic α-equivalence.

The equations of Figure 6 and Figure 7 introduce a notion of *diagrammatic α-equivalence*, which allows us to rename 'internal' names and to contract renamings.

Definition 5.1. Two terms of a nominal monoidal theory are α-equivalent if their equality follows from the equations in Figure 6 and Figure 7.

Every permutation π of names gives rise to bijective functions $\pi_A : A \to \pi[A] = \{\pi(a) \mid a \in A\} = \pi \cdot A$. Any such π_A, as well as the inverse π_A^{-1}, are parallel compositions of δ_{ab} for suitable $a, b \in \mathcal{N}$. In fact, we have

$$\pi_A = \biguplus_{a\in A} \delta_{a\,\pi(a)} \qquad \text{and} \qquad \pi_A^{-1} = \biguplus_{a\in A} \delta_{\pi(a)\,a}$$

We may therefore use the π_A as abbreviations in terms. The following proposition is proved by induction on the structure of terms, see [Bal20] for the details.

Proposition 5.2. *Let $t : A \to B$ be a term of a nominal monoidal theory. The equations in Figure 6 and Figure 7 entail that $\pi \cdot t = \pi_A^{-1}\,; t\,; \pi_B$.*

$$\begin{tikzcd} A \arrow[r, "t"] \arrow[d, "\pi_A"'] & B \arrow[d, "\pi_B"] \\ \pi[A] \arrow[r, "\pi\cdot t"'] & \pi[B] \end{tikzcd}$$

Corollary 5.3. *Let $t : A \to B$ be a term of a nominal monoidal theory. Modulo the equations of Figure 6 and Figure 7, the support of t is $A \cup B$.*

Proof. It follows from the proposition that $\mathsf{supp}\, t \subseteq A \cup B$. For the converse, suppose that there is $x \in A \cup B$ and a support S of t with $x \notin S \subseteq A \cup B$. Choose a permutation π that fixes S and maps x to some $\pi(x) \notin A \cup B$. Then either $\pi \cdot A \neq A$ or $\pi \cdot B \neq B$, hence $\pi \cdot t \neq t$, contradicting that S is a support of t. □

The corollary shows that internal names are bound by sequential composition. Indeed, in a composition $A \xrightarrow{t} C \xrightarrow{s} B$, the names in $C \setminus (A \cup B)$ do not appear in the support of $t; s$.

5.3. **Nominal PROPs.**

From the point of view of Section 4, a nominal PROP, or nPROP for short, is an internal strict commutative monoidal category in $(\mathsf{Nom}, 1, *)$ where
- objects are finite sets of names,
- tensor is given by union of disjoint sets,
- arrows contain all bijections,
- permutations act on objects and bijections point-wise.

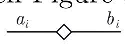
A functor between nPROPs is an internal strict monoidal functor that is the identity on objects and bijections. For readers who skipped Section 4, we spell out the definition of nPROP explicitly:

Definition 5.4. Given a countably infinite set $\mathcal{N}$ of 'names', an nPROP $(\mathbb{C}, \mathbb{I}, \uplus)$ consists of a set $\mathbb{C}_0$ of 'objects' and a set $\mathbb{C}_1$ of 'arrows' satisfying the following conditions.
- $\mathbb{C}_0$ is the set of finite subsets of $\mathcal{N}$.
- $\mathbb{C}_0$ is equipped with a permutation action given by $\pi \cdot A = \pi[A] = \{\pi(a) \mid a \in A\}$ for all finite permutations $\pi : \mathcal{N} \to \mathcal{N}$.
- $\mathbb{C}$ is a category. We write ; for its 'sequential' composition (in the diagrammatic order).
- $\mathbb{C}_1$ contains at least all bijections ('renamings') $\pi_A : A \to \pi \cdot A$, see Definition 5.1, and is closed under the operation mapping an arrow $f : A \to B$ to $\pi \cdot f : \pi \cdot A \to \pi \cdot B$ defined as $\pi \cdot f = (\pi_A)^{-1}; f; \pi_B$.
- $\mathbf{dom}, \mathbf{cod} : \mathbb{C}_1 \to \mathbb{C}_0$ preserve the permutation action.
- $\uplus : \mathbb{C}_0 \times \mathbb{C}_0 \to \mathbb{C}_0$ is the partial operation 'union of disjoint sets'.
- $\uplus : \mathbb{C}_1 \times \mathbb{C}_1 \to \mathbb{C}_1$ is a (partial) commutative and associative operation defined on $\{(f, g) \in \mathbb{C}_1 \times \mathbb{C}_1 \mid \mathbf{dom}\, f \cap \mathbf{dom}\, g = \varnothing = \mathbf{cod}\, f \cap \mathbf{cod}\, g\}$ with the empty function as a unit.
- $(f; f') \uplus (g; g')$ is defined whenever $(f \uplus g); (f' \uplus g')$ is and then both are equal.

Remark 5.5.
- A nominal PROP has a nominal set of objects and a nominal set of arrows.
- The support $\mathsf{supp}(A)$ of an object A is A and the support of an arrow $f : A \to B$ is $A \cup B$. In particular, $\mathsf{supp}\,(f;g) = \mathbf{dom}\, f \cup \mathbf{cod}\, g$. In other words, nominal PROPs have diagrammatic α equivalence.
- There is a category nPROP that consists of nominal PROPs together with functors that are the identity on objects and strict monoidal and equivariant.

- Every NMT presents a nPROP. Conversely, every nPROP is presented by at least one NMT given by all terms as generators and all equations as relations.

5.4. **Examples.**

We present as examples those NMTs that correspond to the SMTs of Figure 3. The significant differences between Figure 3 and Figure 8 are that wires now carry labels and that there is a new generator  which allows us to change the label of a wire. Moreover, in the nominal setting, rules for wire crossings are not needed.

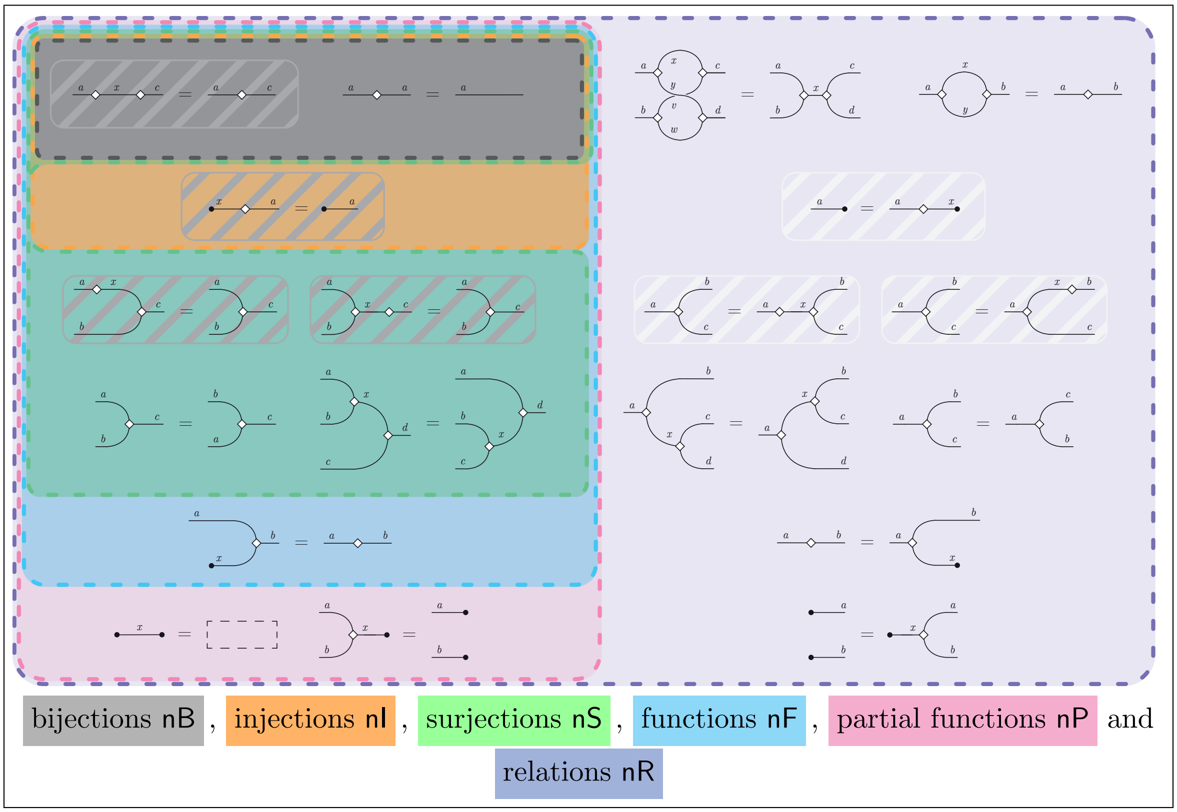
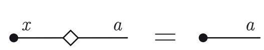


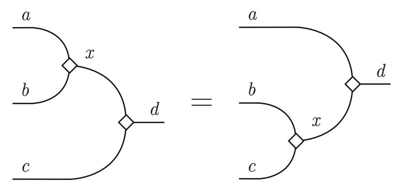
Figure 8: Nominal monoidal theories

Example 5.6. We spell out the details of Figure 8. The category of finite nominals sets and

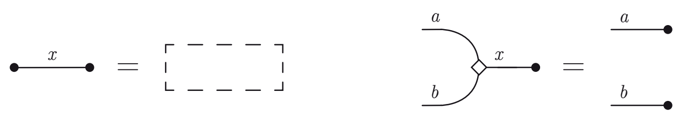
- bijections is presented by the empty signature and equations.
- injections is presented by $\Sigma_i = \{\eta_a : \varnothing \to \{a\} \mid a \in \mathcal{N}\}$ and $E_i = \varnothing$. The equations



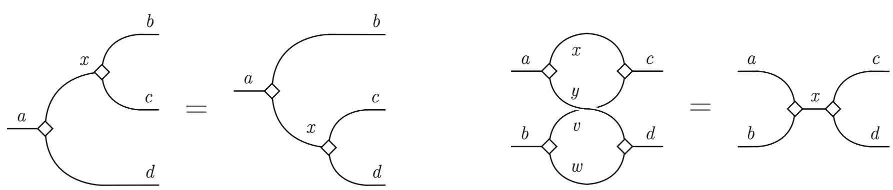
  follow from those of Figure 7.
- surjections is presented by $\Sigma_s = \{\mu_{abc} : \{a, b\} \to \{c\} \mid a, b, c \in \mathcal{N}\}$ and equations E_s are $(\mu_{abx} \uplus id_c) \circ \mu_{cdx} = (\mu_{bcx} \uplus id_a) \circ \mu_{adx}$, presented graphically as

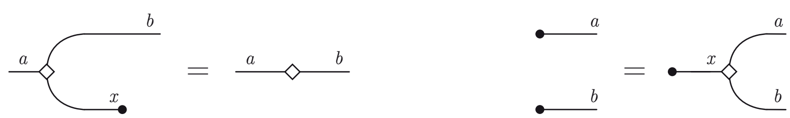


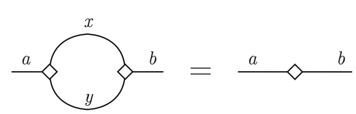
- functions has $\Sigma_f = \Sigma_i \cup \Sigma_s$ and equations E_f are $E_i \cup E_s$ plus $(id_a \uplus \eta_x) \circ \mu_{abx} = \delta_{ab}$
- partial functions has $\Sigma_{pf} = \Sigma_f \cup \{\hat{\eta}_a : \{a\} \to \varnothing \mid a \in \mathcal{N}\}$ and equations E_{pf} are E_f plus $\eta_x \circ \hat{\eta}_x = \varepsilon$ and $\mu_{abx} \circ \hat{\eta}_x = \hat{\eta}_a \uplus \hat{\eta}_b$, shown below



- relations has $\Sigma_r = \Sigma_{pf} \cup \{\hat{\mu}_{abc} : \{a\} \to \{b, c\} \mid a, b, c \in \mathcal{N}\}$, and equations E_r are E_{pf} plus the following







6. Equivalence of Nominal and Symmetric Monoidal Theories

We should be able to switch easily between a notion of ordered names on the one hand and a notion of unordered abstract names on the other. This intuition is reinforced by putting Figures 3 and 8 next to each other. A careful investigation suggests that there is a general procedure to automatically translate one into the other. This section will give such translations and prove that these translations are inverse to each other and preserve completeness. This yields a tool to derive completeness of an NMT from the completeness of the corresponding SMT and vice versa.

In Section 6.1 we define operations Prop : SMT $\to$ PROP and nProp : NMT $\to$ nPROP quotienting theories to their represented categories. Sections 6.2-6.4 are devoted to showing that the categories PROP and nPROP are equivalent. Section 6.5 defines the translation nfNmt : SMT $\to$ NMT and Section 6.6 shows that nfNmt($\mathcal{S}$) is complete if $\mathcal{S}$ is complete. Sections 6.7-6.8 establish the analogous result in the other direction.

6.1. Interpreting (nominal) monoidal theories as (nominal) PROPs.

We start by formalising the operation that maps a theory to the category it represents. Given a theory $\langle \Sigma, E\rangle$ of string diagrams, where Σ is the set of generators and $E \subseteq \mathsf{Trm}(\Sigma) \times \mathsf{Trm}(\Sigma)$ is the set of equations, the operation $\mathsf{Prop} : \mathsf{SMT} \to \mathsf{PROP}$ takes the signature $\langle \Sigma, E\rangle$ to the category of SMT terms, quotiented by the equations of E.

Remark 6.1. The operation $\mathsf{Prop} : \mathsf{SMT} \to \mathsf{PROP}$ is defined as

$$\mathsf{Prop}\,\langle \Sigma, E\rangle = \mathsf{Trm}(\Sigma)/\mathcal{Th}(E \cup \mathbf{SMT})$$

$$\frac{}{s = s \in \mathcal{Th}(E)} \qquad \frac{s = t \in \mathcal{Th}(E)}{t = s \in \mathcal{Th}(E)}$$

$$\frac{s = t \in \mathcal{Th}(E) \qquad t = u \in \mathcal{Th}(E)}{s = u \in \mathcal{Th}(E)} \qquad \frac{s = s' \in \mathcal{Th}(E) \qquad t = t' \in \mathcal{Th}(E)}{s * t = s' * t' \in \mathcal{Th}(E)}$$

Figure 9: Closure operator

This definition uses the closure operator $\mathcal{Th}$, defined in Figure 9, which is the usual deductive closure of equational logic. We have $* = \{\,;\,,\,\oplus\,\}$ for $\mathcal{Th}$ over equations on Trms and for equations over nTrms we have $* = \{\,;\,,\,\uplus\,\}$ along with an additional rule for permutations:

$$\frac{s = t \in \mathcal{Th}(E)}{\pi \cdot s = \pi \cdot t \in \mathcal{Th}(E)}$$

We have a similar construction for NMTs, where we define a functor $\mathsf{nProp} : \mathsf{NMT} \to \mathsf{nPROP}$:

Remark 6.2. $\mathsf{nProp} : \mathsf{NMT} \to \mathsf{nPROP}$ is defined as

$$\mathsf{nProp}\,\langle \Sigma, E\rangle = \mathsf{nTrm}(\Sigma)/\mathcal{Th}(E \cup \mathbf{NMT})$$

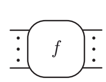
Finally, we prove the following property of the closure operator, which we will use in a later lemma.

Lemma 6.3. *Given a set of equations $X \subseteq \mathsf{nTrm}(A) \times \mathsf{nTrm}(A)$ (or $X \subseteq \mathsf{Trm}(A) \times \mathsf{Trm}(A)$), and a homomorphism $f : \mathsf{nTrm}(A) \to \mathsf{nTrm}(B)$ (or $f : \mathsf{Trm}(A) \to \mathsf{Trm}(B)$), we have:*

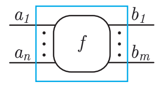
$$f[\mathcal{Th}(X)] \subseteq \mathcal{Th}(f[X])$$

6.2. Embedding PROPs into nominal PROPs.

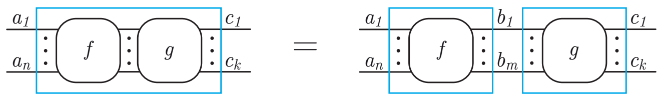
In this section, we start proving the equivalence of the categories PROP and nPROP by embedding ordinary PROPs into nPROPs. This is achieved in the following manner. Given an ordinary diagram $f : n \to m$,

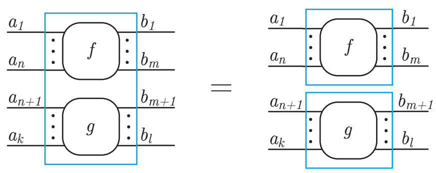


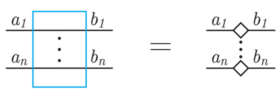
we create “boxed” nominal versions $[\boldsymbol{a}\rangle f \langle \boldsymbol{b}]$, where $\boldsymbol{a} = [a_1, \ldots, a_n]$ and $\boldsymbol{b} = [b_1, \ldots, b_m]$ are lists of pairwise distinct names:



For "boxing" to preserve the relevant structure, we have to ensure, in particular, that the symmetric monoidal tensor of a PROPs is mapped to the commutative tensor of nPROPs, and that sequential composition and identities are preserved:







We give the details of this construction in the proposition below, using the following notational conventions. If $\boldsymbol{a} = [a_1, \dots a_n]$ is a list, then $\underline{\boldsymbol{a}} = \{a_1, \dots a_n\}$. Given lists $\boldsymbol{a}$ and $\boldsymbol{a'}$ with $\underline{\boldsymbol{a}} = \underline{\boldsymbol{a'}}$ we abbreviate the bijections in a PROP (also called symmetries) mapping $i \mapsto j$ whenever $a_i = a'_j$ as

$$\langle \boldsymbol{a} | \boldsymbol{a'} \rangle$$

Similarly, in an nPROP, given lists $\boldsymbol{a}$ and $\boldsymbol{b}$ of the same length, we write

$$[\boldsymbol{a} | \boldsymbol{b}]$$

for the bijection $\biguplus \delta_{a_i b_i}$ mapping $a_i \mapsto b_i$.

Proposition 6.4. *For any PROP $\mathcal{S}$, there is an nPROP*

$$\mathsf{NOM}(\mathcal{S})$$

that has an arrow $[\boldsymbol{a}\rangle f \langle \boldsymbol{b}] : \underline{\boldsymbol{a}} \to \underline{\boldsymbol{b}}$ for all arrows $f : \underline{n} \to \underline{m}$ of $\mathcal{S}$ and for all lists $\boldsymbol{a} = [a_1, \dots a_n]$ and $\boldsymbol{b} = [b_1, \dots b_m]$ of pairwise distinct names. These arrows are subject to the set

$$\mathbf{NOM}$$

consisting of the following equations:

$$[\boldsymbol{a}\rangle f \,;g \langle \boldsymbol{c}] = [\boldsymbol{a}\rangle f \langle \boldsymbol{b}] \,; [\boldsymbol{b}\rangle g \langle \boldsymbol{c}] \tag{NOM-1}$$

$$[\boldsymbol{a} +\!\!+ \boldsymbol{c}\rangle f \oplus g \langle \boldsymbol{b} +\!\!+ \boldsymbol{d}] = [\boldsymbol{a}\rangle f \langle \boldsymbol{b}] \uplus [\boldsymbol{c}\rangle g \langle \boldsymbol{d}] \tag{NOM-2}$$

$$[\boldsymbol{a}\rangle id \langle \boldsymbol{b}] = [\boldsymbol{a} | \boldsymbol{b}] \tag{NOM-3}$$

$$[\boldsymbol{a}\rangle \, \langle \boldsymbol{b} | \boldsymbol{b'} \rangle \,; f \, \langle \boldsymbol{c}] \; = [\boldsymbol{a} | \boldsymbol{b}] \,; [\boldsymbol{b'}\rangle f \langle \boldsymbol{c}] \tag{NOM-4}$$

$$[\boldsymbol{a}\rangle \, f \,; \langle \boldsymbol{b} | \boldsymbol{b'} \rangle \, \langle \boldsymbol{c}] \; = [\boldsymbol{a}\rangle f \langle \boldsymbol{b}] \,; [\boldsymbol{b'} | \boldsymbol{c}] \tag{NOM-5}$$

Then, $\mathsf{NOM} : \mathsf{PROP} \to \mathsf{nPROP}$ *is a functor mapping an arrow of* PROP*s* $F : \mathcal{S} \to \mathcal{S}'$ *to an arrow of* nPROP*s* $\mathsf{NOM}(F) : \mathsf{NOM}(\mathcal{S}) \to \mathsf{NOM}(\mathcal{S}')$ *defined by* $\mathsf{NOM}(F)([\boldsymbol{a}\rangle\, g\, \langle\boldsymbol{b}]) = [\boldsymbol{a}\rangle\, Fg\, \langle\boldsymbol{b}]$.

In the proposition above, $[\boldsymbol{a}\rangle f \langle\boldsymbol{b}]$ is a purely formal generator. Of course, the inutition is that $[\boldsymbol{a}\rangle f \langle\boldsymbol{b}]$ represents a composition of maps $[\boldsymbol{a}\rangle ; f ; \langle\boldsymbol{b}]$. This is made precise in the following example.

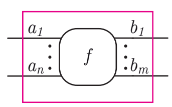
Example 6.5. $\mathsf{n}\mathbb{F}$ is isomorphic to $\mathsf{NOM}(\mathbb{F})$ with the isomorphism $G : \mathsf{NOM}(\mathbb{F}) \to \mathsf{n}\mathbb{F}$ being defined as

$$G([\boldsymbol{a}\rangle f \langle\boldsymbol{b}]) = [\![\boldsymbol{a}\rangle\!\rangle f \langle\!\langle\boldsymbol{b}]\!]$$

where the semantic box $[\![-\rangle\!\rangle - \langle\!\langle-]\!]$ translates an arrow $f : n \to m$ by pre-composing with $\overrightarrow{\boldsymbol{a}} : A \to n$ and post-composing with $\overrightarrow{\boldsymbol{b}}^{-1} : m \to B$, where $\overrightarrow{\boldsymbol{a}}$ is the bijection between $\underline{\boldsymbol{a}}$ and n given by the ordering of the list $\boldsymbol{a}$. In other words, we have $[\![\boldsymbol{a}\rangle\!\rangle f \langle\!\langle\boldsymbol{b}]\!] \stackrel{\text{def}}{=} \overrightarrow{\boldsymbol{a}} ; f ; \overrightarrow{\boldsymbol{b}}^{-1}$.

6.3. **Embedding nominal PROPs into PROPs.**

Interestingly enough we can embed nPROPs into PROPs in much the same fashion as above. We write this translation in one dimension as $\langle\boldsymbol{a}]\, f\, [\boldsymbol{b}\rangle$ and in two-dimensions as



Proposition 6.6. For any nPROP $\mathcal{T}$ there is a PROP

$$\mathsf{ORD}(\mathcal{T})$$

that has an arrow $\langle\boldsymbol{a}]f[\boldsymbol{b}\rangle \in \mathsf{ORD}(\mathcal{T})$ for all arrows $f : A \to B$ of $\mathcal{T}$ and for all lists $\boldsymbol{a} = [a_1, \ldots a_n]$ and $\boldsymbol{b} = [b_1, \ldots b_m]$ of pairwise distinct names. These arrows are subject to the equations below:

$$\langle\boldsymbol{a}]\, f\, ; g\, [\boldsymbol{c}\rangle = \langle\boldsymbol{a}]\, f\, [\boldsymbol{b}\rangle ; \langle\boldsymbol{b}]\, g\, [\boldsymbol{c}\rangle \tag{ORD-1}$$

$$\langle\boldsymbol{a}_f +\!\!\!+\, \boldsymbol{a}_g]\, f \uplus g\, [\boldsymbol{b}_f +\!\!\!+\, \boldsymbol{b}_g\rangle = \langle\boldsymbol{a}_f]\, f\, [\boldsymbol{b}_f\rangle \oplus \langle\boldsymbol{a}_g]\, g\, [\boldsymbol{b}_g\rangle \tag{ORD-2}$$

$$\langle\boldsymbol{a}]\, id\, [\boldsymbol{a}\rangle = id \tag{ORD-3}$$

$$\langle\boldsymbol{a}]\, [\boldsymbol{a}'|\boldsymbol{b}] ; f\, [\boldsymbol{c}\rangle = \langle\boldsymbol{a}|\boldsymbol{a}'\rangle ; \langle\boldsymbol{b}]\, f\, [\boldsymbol{c}\rangle \tag{ORD-4}$$

$$\langle\boldsymbol{a}]\, f ; [\boldsymbol{b}|\boldsymbol{c}]\, [\boldsymbol{c}'\rangle = \langle\boldsymbol{a}]\, f\, [\boldsymbol{b}\rangle ; \langle\boldsymbol{c}|\boldsymbol{c}'\rangle \tag{ORD-5}$$

$$\langle\boldsymbol{a}]\, f\, [\boldsymbol{b}\rangle = \langle\pi \cdot \boldsymbol{a}]\, \pi \cdot f\, [\pi \cdot \boldsymbol{b}\rangle \tag{ORD-6}$$

Then, ORD is a functor mapping an arrow of nPROPs $F : \mathcal{T} \to \mathcal{T}'$ to an arrow of PROPs $\mathsf{ORD}(F) : \mathsf{ORD}(\mathcal{T}) \to \mathsf{ORD}(\mathcal{T}')$ defined by

$$\mathsf{ORD}(F)(\langle\boldsymbol{a}]\, f\, [\boldsymbol{b}\rangle) = \langle\boldsymbol{a}]\, Ff\, [\boldsymbol{b}\rangle$$

Example 6.7. The category $\mathbb{F}$ is isomorphic to $\mathsf{ORD}(\mathsf{n}\mathbb{F})$, with the isomorphism $G : \mathsf{ORD}(\mathsf{n}\mathbb{F}) \to \mathbb{F}$ given, for all $f : A \to B$, as

$$G(\langle\boldsymbol{a}]f[\boldsymbol{b}\rangle) = \langle\!\langle\boldsymbol{a}]\!]f[\![\boldsymbol{b}\rangle\!\rangle.$$

The semantic brackets $\langle\!\langle -]\!] - [\![- \rangle\!\rangle$ translate the arrow $f \in \mathsf{n}\mathbb{F}$ into an arrow in $\mathbb{F}$, defined as $\langle\!\langle \boldsymbol{a}]\!] f [\![\boldsymbol{b} \rangle\!\rangle \stackrel{\text{def}}{=} \overrightarrow{\boldsymbol{a}}^{-1} \,;\, f \,;\, \overrightarrow{\boldsymbol{b}}$.
The detailed verifications are straightforward and can be found in [Bal20, Example 7.26].

6.4. Equivalence of PROPs and nominal PROPs.

We show that PROP and nPROP are equivalent. More precisely, the functors ORD from Proposition 6.6 and NOM from Proposition 6.4 form an equivalence of categories.

Proposition 6.8. For each PROP $\mathcal{S}$, there is an isomorphism of PROPs, natural in $\mathcal{S}$,

$$\Delta : \mathcal{S} \to \mathsf{ORD}(\mathsf{NOM}(\mathcal{S}))$$

mapping $f \in \mathcal{S}$ to $\langle \boldsymbol{a}] [\boldsymbol{a} \rangle \, f \, \langle \boldsymbol{b}] [\boldsymbol{b} \rangle$ for some choice of $\boldsymbol{a}, \boldsymbol{b}$.

Proof. For a proof see [Bal20, Proposition 7.33]. □

Proposition 6.9. For each nPROP $\mathcal{T}$, there is an isomorphism of nPROPs, natural in $\mathcal{T}$,

$$\mathsf{NOM}(\mathsf{ORD}(\mathcal{T})) \to \mathcal{T}$$

mapping the $[\boldsymbol{c}\rangle\langle\boldsymbol{a}] \, f \, [\boldsymbol{b}\rangle\langle\boldsymbol{d}]$ generated by an $f : \underline{\boldsymbol{a}} \to \underline{\boldsymbol{b}}$ in $\mathcal{T}$ to $[\boldsymbol{c}|\boldsymbol{a}] \,;\, f \,;\, [\boldsymbol{b}|\boldsymbol{d}]$.

Proof. For a proof see [Bal20, Proposition 7.34]. □

Since the last two propositions provide an isomorphic unit and counit of an adjunction, we obtain

Theorem 6.10. The categories PROP and nPROP are equivalent.

Remark 6.11. If we generalise the notion of PROP from MacLane [ML78] to Lack [Lac04], in other words, if we drop the last equation of Figure 2 expressing the naturality of symmetries, we still obtain an adjunction, in which NOM is left-adjoint to ORD. Nominal PROPs then are a full reflective subcategory of ordinary PROPs. In other words, the (generalised) PROPs $\mathcal{S}$ that satisfy naturality of symmetries are exactly those for which $\mathcal{S} \cong \mathsf{ORD}(\mathsf{NOM}(\mathcal{S}))$. Example 6.13 shows in more detail how naturality of symmetries is a consequence of the commutativity of the nominal tensor.

6.5. Translating SMTs into NMTs.

We give a formal definition of the translation of the ordinary theories of Figure 3 to the nominal theories of Figure 8.

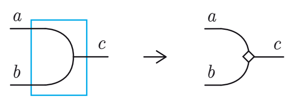
First let us recall that the composition of Prop from Remark 6.1 with the functor NOM from Proposition 6.4, gives us the following interpretation of a PROP as an nPROP: Given an SMT $\langle \Sigma, E \rangle$, we can generate an nPROP, by simply taking all the SMT-terms over Σ, as generators (taking $\mathsf{Trm}(\Sigma)$ to $\mathsf{nTrm}(\mathsf{Trm}(\Sigma))$) and taking $\mathrm{box}(E) \cup \mathbf{NOM}$ as equations, where:

- $\mathrm{box}(E) = \{[\boldsymbol{a}\rangle f \langle \boldsymbol{b}] = [\boldsymbol{a}\rangle g \langle \boldsymbol{b}] \mid f = g \in E\}$
- $\mathbf{NOM}$ are the equations from Proposition 6.4

Whilst this construction gives us a way of embedding equivalence classes of ordinary string diagrams into equivalence classes of nominal ones, it does not answer the question of how to translate the axioms defining an SMT into the axioms of the corresponding NMT.

If we recall the definition of an NMT, we see that the signature of a nominal theory consists of a set of ordinary generators Σ and set of equations over $\mathsf{nTrm}(\Sigma)$. Thus, given the ordinary signature of an SMT, with generators Σ and the set of equations $E \subseteq \mathsf{Trm}(\Sigma) \times \mathsf{Trm}(\Sigma)$, we need to obtain an $E' \subseteq \mathsf{nTrm}(\Sigma) \times \mathsf{nTrm}(\Sigma)$ such that any equivalence class induced by E' and the equations of NOM (due to the ordinary diagrams being embedded in nominal ones) are mirrored by E.

Intuitively, we translate equations of E, by first embedding them inside a 'nominal box' as a whole and then use the rules of NOM to recursively normalise all "sub-diagrams" into nominal ones (see Example 6.13). When we hit the "base case", i.e. a "boxed" generator from Σ, we simply replace it with a corresponding nominal generator:



Below, we use the notation $\underline{\gamma}$ to highlight the difference between an element γ of Σ and the string diagram $\underline{\gamma} \in \mathsf{Trm}(\Sigma)$ as in the blue box above.

The definition

$$\mathsf{nfNmt} : \mathsf{nTrm}(\mathsf{Trm}(\Sigma)) \to \mathsf{nTrm}(\Sigma)$$

is thus straightforward:

$$\begin{aligned}
&\mathsf{nfNmt}\,([\boldsymbol{a}\rangle \underline{\gamma} \langle \boldsymbol{b}]) = [\boldsymbol{a}\rangle \gamma \langle \boldsymbol{b}] \text{ where } \gamma \in \Sigma\\
&\mathsf{nfNmt}\,([a\rangle id \langle b]) = \delta_{ab}\\
&\mathsf{nfNmt}\,([ab\rangle \sigma \langle cd]) = [ab|dc]\\
&\mathsf{nfNmt}\,([\boldsymbol{a}\rangle f\,; g \langle \boldsymbol{c}]) = \mathsf{nfNmt}\,([\boldsymbol{a}\rangle f \langle \boldsymbol{b}])\,; \mathsf{nfNmt}\,([\boldsymbol{b}\rangle g \langle \boldsymbol{c}])\\
&\mathsf{nfNmt}\,([\boldsymbol{a} + \boldsymbol{b}\rangle f \oplus g \langle \boldsymbol{c} + \boldsymbol{d}]) = \mathsf{nfNmt}\,([\boldsymbol{a}\rangle f \langle \boldsymbol{c}]) \uplus \mathsf{nfNmt}\,([\boldsymbol{b}\rangle g \langle \boldsymbol{d}])\\
&\mathsf{nfNmt}\,(id_a) = id_a\\
&\mathsf{nfNmt}\,(\delta_{ab}) = \delta_{ab}\\
&\mathsf{nfNmt}\,(f\,; g) = \mathsf{nfNmt}\,(f)\,; \mathsf{nfNmt}\,(g)\\
&\mathsf{nfNmt}\,(f \uplus g) = \mathsf{nfNmt}\,(f) \uplus \mathsf{nfNmt}\,(g)\\
&\mathsf{nfNmt}\,(\pi \cdot f) = \pi \cdot \mathsf{nfNmt}\,(f)
\end{aligned}$$

Definition 6.12. We define $\mathsf{Nmt} : \mathsf{SMT} \to \mathsf{NMT}$ as

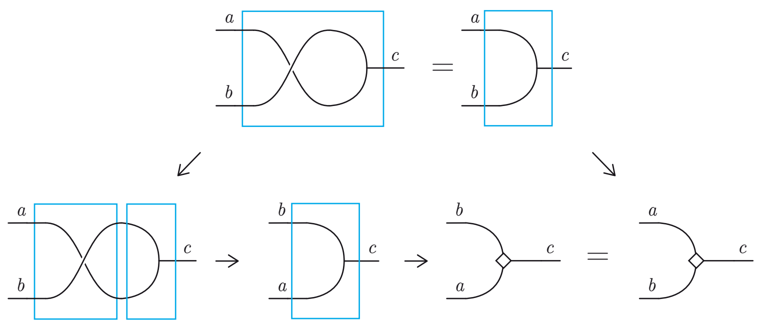
$$\mathsf{Nmt}\,\langle \Sigma, E\rangle = \langle \Sigma, \mathsf{nfNmt}(\mathsf{box}(E))\rangle$$

where we extend the function nfNmt on a set of equations in the obvious way:

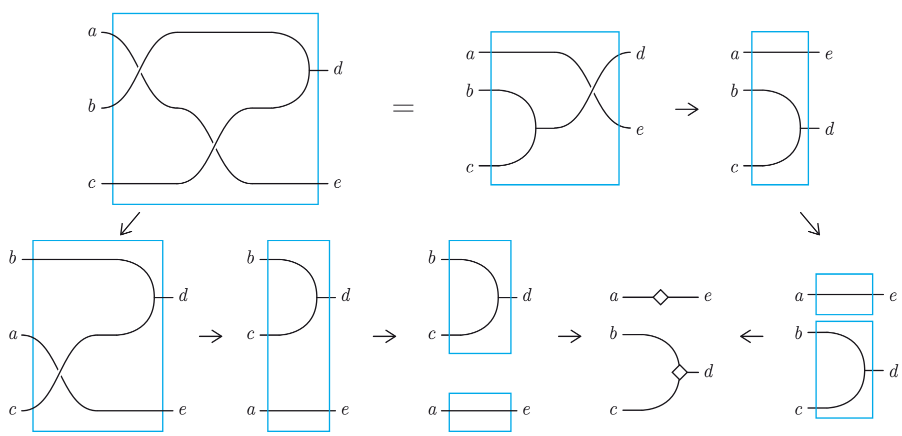
$$\mathsf{nfNmt}\,(E) = \{\mathsf{nfNmt}(f) = \mathsf{nfNmt}(g) \mid f = g \in E\}$$

We now return to Figure 3 and Figure 8 and show at the hand of an example how, by applying Nmt to the equations in Figure 3, we obtain the equations in Figure 8.

Example 6.13. In this example, we illustrate the translation of a rule of an SMT into the corresponding rule of an NMT via nfNmt. The diagram below shows the application of nfNmt to both sides of an equation in the SMT theory of surjections.



The next diagram illustrates the fact that the equation (SMT-nat) get subsumed by the equations of NMT, namely by (NMT-comm).



6.6. **Completeness of NMTs.**

We show how to prove the completeness of $\mathsf{Nmt}(\Sigma, E)$ from the completeness of $\langle \Sigma, E\rangle$. The central observation is that the diagram in Figure 10 commutes up to isomorphism.
We set up some preliminaries. First, we define $\iota : \mathsf{nTrm}(\Sigma) \to \mathsf{nTrm}(\mathsf{Trm}(\Sigma))$:

$$
\begin{aligned}
&\iota\,([\boldsymbol{a}\rangle\gamma\langle\boldsymbol{b}]) = [\boldsymbol{a}\rangle\underline{\gamma}\langle\boldsymbol{b}] \text{ where } \gamma \in \Sigma\\
&\iota\,(id_a) = id_a\\
&\iota\,(\delta_{ab}) = \delta_{ab}\\
&\iota\,(f\,;g) = \iota\,(f)\,;\,\iota\,(g)\\
&\iota\,(f \uplus g) = \iota\,(f) \uplus \iota\,(g)\\
&\iota\,(\pi \cdot f) = \pi \cdot \iota\,(f)
\end{aligned}
$$

The only interesting case is the one of a nominal generator $[\boldsymbol{a}\rangle\gamma\langle\boldsymbol{b}]$, which gets turned into an ordinary string diagram $\underline{\gamma}$, embedded in a nominal diagram $[\boldsymbol{a}\rangle\underline{\gamma}\langle\boldsymbol{b}]$.

$$\begin{tikzcd}
\mathsf{SMT} \arrow[rrr, "\mathsf{Prop}"] \arrow[dddd, "\mathsf{Nmt}"'] & & & \mathsf{PROP} \arrow[dddd, "\mathsf{NOM}"] \\
& \langle \Sigma, E \rangle \arrow[r, mapsto] \arrow[ddr, mapsto, rounded corners] & \mathsf{Prop}\,\langle \Sigma, E \rangle \arrow[d, mapsto] & \\
& & \mathsf{NOM}(\mathsf{Prop}\,\langle \Sigma, E \rangle) \arrow[d, bend left, "\mathsf{nfNmt}"] & \\
& & \mathsf{nProp}(\mathsf{Nmt}\,\langle \Sigma, E \rangle) \arrow[u, bend left, "\iota"] & \\
\mathsf{NMT} \arrow[rrr, "\mathsf{nProp}"] & & & \mathsf{nPROP}
\end{tikzcd}$$

Figure 10: Completing the square

Lemma 6.14. *The diagram in Figure 10 commutes up to isomorphism.*

Proof. We want to show that the two maps nfNmt and ι are isomorphisms. By definition, both nfNmt and ι are homomorphisms between the term algebras. To verify that they are well-defined, that is, that they respect equivalence classes of nTrms, we need to check that they preserve the equations:

- For the map ι we have to show

$$\iota\,[\mathcal{T}\!h(\mathsf{nfNmt}[\mathrm{box}[E]] \cup \mathbf{NMT})] \subseteq \mathcal{T}\!h(\mathcal{T}\!h(\mathrm{box}[E \cup \mathbf{SMT}]) \cup \mathbf{NOM} \cup \mathbf{NMT})$$

  In fact, by Lemma 6.3, it suffices to check that $\iota\,[\mathsf{nfNmt}[\mathrm{box}[E]]] \subseteq \mathcal{T}\!h(\mathrm{box}[E] \cup \mathbf{NMT})$ and $\iota\,[\mathbf{NMT}] \subseteq \mathcal{T}\!h(\mathrm{box}[E] \cup \mathbf{NMT})$. The first inequality follows immediately from the fact that $\iota\,[\mathsf{nfNmt}[\mathrm{box}[E]] = \mathrm{box}[E]$. The second inequality follows straightforwardly.
- For the map nfNmt we have to show the other direction

$$\mathsf{nfNmt}[\mathcal{T}\!h(\mathcal{T}\!h(\mathrm{box}[E \cup \mathbf{SMT}]) \cup \mathbf{NOM} \cup \mathbf{NMT})] \subseteq \mathcal{T}\!h(\mathsf{nfNmt}[\mathrm{box}[E]] \cup \mathbf{NMT})$$

  For this it suffices:

$$\begin{aligned}
&\mathsf{nfNmt}[\mathcal{T}\!h(\mathcal{T}\!h(\mathrm{box}[E \cup \mathbf{SMT}]) \cup \mathbf{NOM} \cup \mathbf{NMT})] \\
\subseteq\ &\mathcal{T}\!h(\mathsf{nfNmt}[\mathrm{box}[E]] \cup \mathsf{nfNmt}[\mathbf{NOM}] \cup \mathsf{nfNmt}[\mathrm{box}[\mathbf{SMT}]] \cup \mathsf{nfNmt}[\mathbf{NMT}]) \\
\subseteq\ &\mathcal{T}\!h(\mathsf{nfNmt}[\mathrm{box}[E]] \cup \mathbf{NMT})
\end{aligned}$$

  To justify the second inequality, we only need to prove:
  – $\mathsf{nfNmt}[\mathrm{box}[E]] \cup \mathsf{nfNmt}[\mathbf{NMT}] \subseteq \mathcal{T}\!h(\mathsf{nfNmt}[\mathrm{box}[E]] \cup \mathbf{NMT})$, which is immediate.
  – $\mathsf{nfNmt}[\mathrm{box}[\mathbf{SMT}]] \subseteq \mathcal{T}\!h(\mathsf{nfNmt}[\mathrm{box}[E]] \cup \mathbf{NMT})$, which follows since the equations in $\mathbf{SMT}$ get subsumed by $\mathbf{NMT}$ when box-ed and normalised via nfNmt.
  – $\mathsf{nfNmt}[\mathbf{NOM}] \subseteq \mathcal{T}\!h(\mathsf{nfNmt}[\mathrm{box}[E]] \cup \mathbf{NMT})$. The only two equations which require any serious verification are (NOM-4) and (NOM-5). The proofs of both are essentially

the same, so we will only consider the first one here:

$$\begin{aligned}
\mathsf{nfNmt}([\boldsymbol{a}\rangle\langle\boldsymbol{b}|\boldsymbol{b}'\rangle\,;f\langle\boldsymbol{c}]) &= \mathsf{nfNmt}([\boldsymbol{a}\rangle\langle\boldsymbol{b}|\boldsymbol{b}'\rangle\langle\boldsymbol{x}])\,;\mathsf{nfNmt}([\boldsymbol{x}\rangle f\langle\boldsymbol{c}])\\
&\overset{\mathrm{NMT}}{=} \mathsf{nfNmt}([\boldsymbol{a}\rangle\langle\boldsymbol{b}|\boldsymbol{b}'\rangle\langle\boldsymbol{b}'])\,;\mathsf{nfNmt}([\boldsymbol{b}'\rangle f\langle\boldsymbol{c}])\\
&\overset{\mathrm{NMT}}{=} [\boldsymbol{a}|\boldsymbol{b}]\,;[\boldsymbol{b}|\boldsymbol{a}]\,;\mathsf{nfNmt}([\boldsymbol{a}\rangle\langle\boldsymbol{b}|\boldsymbol{b}'\rangle\langle\boldsymbol{b}'])\,;\mathsf{nfNmt}([\boldsymbol{b}'\rangle f\langle\boldsymbol{c}])\\
&\overset{\mathrm{NMT}}{=} [\boldsymbol{a}|\boldsymbol{b}]\,;\mathsf{nfNmt}([\boldsymbol{b}\rangle\langle\boldsymbol{b}|\boldsymbol{b}'\rangle\langle\boldsymbol{b}'])\,;\mathsf{nfNmt}([\boldsymbol{b}'\rangle f\langle\boldsymbol{c}])\\
&\overset{\mathrm{NMT}}{=} [\boldsymbol{a}|\boldsymbol{b}]\,;\mathsf{nfNmt}([\boldsymbol{b}'\rangle f\langle\boldsymbol{c}])
\end{aligned}$$

Finally, we show that the maps nfNmt and ι are inverses of each other. We have $\mathsf{nfNmt} \circ \iota\,(f) = f$ for any $f \in \mathsf{nTrm}(\Sigma)$, by induction on f. We have $\iota \circ \mathsf{nfNmt}\,(f) \overset{\mathrm{NOM}}{=} f$ for any $f \in \mathsf{nTrm}(\mathsf{Trm}(\Sigma))$, where

$$\overset{\mathrm{NOM}}{=}$$

is equality up to the equations $\mathbf{NOM} \cup \mathbf{NMT} \cup \mathsf{box}[\mathbf{SMT}]$, also by induction on f. The interesting case is $f = [ab\rangle\sigma\langle cd]$ for the twist σ:

$$\begin{aligned}
\iota \circ \mathsf{nfNmt}\,([ab\rangle\Sigma\langle cd]) &= \iota\,([ab|dc])\\
&= [ab|dc] \overset{\mathrm{NOM}}{=} [ba|cd] \overset{\mathrm{NOM}}{=} [ba\rangle id\langle cd]\\
&\overset{\mathrm{NOM}}{=} [ab|ab]\,;[ba\rangle id\langle cd]\\
&\overset{\mathrm{NOM}}{=} [ab\rangle\langle ab|ba\rangle\,;id\langle cd]\\
&\overset{\mathrm{NOM}}{=} [ab\rangle\langle ab|ba\rangle\langle cd]\\
&= [ab\rangle\Sigma\langle cd]
\end{aligned}$$

This finishes the proof. □

In Section 5.4, we introduced the nominal monoidal theories theories for the categories of bijections, injections, surjections and functions (amongst others), see Figure 8. We show now how to prove the completeness of these theories from the completeness of the corresponding SMTs of Figure 3.

Recall that completeness of an SMT $\langle\Sigma, E\rangle$ with regards to some category $\mathbb{C}$ means that the PROP presented by $\langle\Sigma, E\rangle$ is isomorphic to $\mathbb{C}$

$$\mathsf{Prop}\,\langle\Sigma, E\rangle \cong \mathbb{C}.$$

Likewise, the completeness of an NMT with regards to some category $\mathsf{n}\mathbb{C}$ is the existence of an isomorphism

$$\mathsf{nProp}\,\langle\Sigma, E\rangle \cong \mathsf{n}\mathbb{C}$$

Theorem 6.15 (Completeness of NMTs)**.** *The calculi of Figure 8 are sound and complete, that is, the categories presented by these calculi are isomorphic to the categories of finite sets of names with the respective maps.*

Proof. We show the result for the category of finite functions $\mathsf{n}\mathbb{F}$. Similar arguments apply to the other theories presented in Figure 8. In order to show completeness of the nominal theory of functions w.r.t. $\mathsf{n}\mathbb{F}$, we start with the SMT $\langle\Sigma_{\mathbb{F}}, E_{\mathbb{F}}\rangle$ of Figure 3. We know

$$\mathsf{NOM}(\mathsf{Prop}\,\langle\Sigma_{\mathbb{F}}, E_{\mathbb{F}}\rangle) \cong \mathsf{nProp}(\mathsf{Nmt}\,\langle\Sigma_{\mathbb{F}}, E_{\mathbb{F}}\rangle)$$

from Lemma 6.14,

$$\mathsf{NOM}(\mathbb{F}) \cong \mathsf{n}\mathbb{F}$$

from Example 6.5 and

$$\mathsf{Prop}\,\langle \Sigma_{\mathbb{F}}, E_{\mathbb{F}} \rangle \cong \mathbb{F}$$

from completeness of $\langle \Sigma_{\mathbb{F}}, E_{\mathbb{F}} \rangle$ for $\mathbb{F}$. Putting these together, we obtain

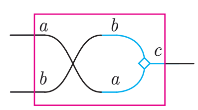
$$\mathsf{nProp}(\mathsf{Nmt}\,\langle \Sigma_{\mathbb{F}}, E_{\mathbb{F}} \rangle) \cong \mathsf{n}\mathbb{F}$$

that is, $\mathsf{Nmt}\,\langle \Sigma_{\mathbb{F}}, E_{\mathbb{F}} \rangle$ is complete for $\mathsf{n}\mathbb{F}$. □

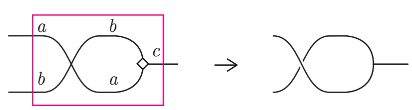
6.7. **Translating NMTs into SMTs.**

This section follows the same lines as Section 6.5, but now translating nominal monoidal theories into symmetric monoidal theories. Indeed, we can convert an NMT into an SMT by first embedding nominal equations into ordinary string diagrams and then normalising the diagrams via a function nfSmt, which we are going to define now.

Compared to what we have seen when we defined nfNmt, normalising embedded nominal string diagrams into ordinary string diagrams is slightly more tricky. This is due to the fact that in nominal sequential composition, we are allowed to compose two diagrams which share the same set of output and input labels, disregarding the order of the named ports. For example, in the picture below, we see a wire crossing inside the purple box, introduced by the fact that the ports of the box interface and the ports of the generator inside the box have to be lined up.



However, no such crossing is (directly) visible in the linear syntax $\langle a, b][b, a\rangle\mu\langle c][c\rangle$.
Thus, when translating such a diagram back into ordinary string diagrams, we might need to insert some symmetries, i.e. the diagram $\langle a, b][b, a\rangle\mu\langle c][c\rangle$ should normalise to $\sigma\,;\mu$:



After these preliminary considerations, we now define

$$\mathsf{nfSmt} : \mathsf{Trm}(\mathsf{dia}(\mathsf{nTrm}(\Sigma))) \to \mathsf{Trm}(\Sigma)$$

as:

$$\begin{aligned}
&\mathsf{nfSmt}\,(\langle \boldsymbol{a}][\boldsymbol{a'}\rangle\gamma\langle \boldsymbol{b'}][\boldsymbol{b}\rangle) = \langle \boldsymbol{a}|\boldsymbol{a'}\rangle\,;\underline{\gamma}\,;\langle \boldsymbol{b'}|\boldsymbol{b}\rangle \text{ where } \gamma\in\Sigma\\
&\mathsf{nfSmt}\,(\langle a]id_a[a\rangle) = id\\
&\mathsf{nfSmt}\,(\langle a]\delta_{ab}[b\rangle) = id\\
&\mathsf{nfSmt}\,(\langle \boldsymbol{a}]f\,;g[\boldsymbol{c}\rangle) = \mathsf{nfSmt}\,(\langle \boldsymbol{a}]f[\boldsymbol{b}\rangle)\,;\mathsf{nfSmt}\,(\langle \boldsymbol{b}]g[\boldsymbol{c}\rangle)\\
&\mathsf{nfSmt}\,(\langle \boldsymbol{a}]f\uplus g[\boldsymbol{b}\rangle) = \langle \boldsymbol{a}|\boldsymbol{a}_1 +\!\!+\, \boldsymbol{a}_2\rangle\,;(\mathsf{nfSmt}\,(\langle \boldsymbol{a}_1]f[\boldsymbol{b}_1\rangle)\oplus\mathsf{nfSmt}\,(\langle \boldsymbol{a}_2]g[\boldsymbol{b}_2\rangle))\,;\langle \boldsymbol{b}_1 +\!\!+\, \boldsymbol{b}_2|\boldsymbol{b}\rangle\\
&\mathsf{nfSmt}\,(\langle \boldsymbol{a}]\pi\cdot f[\boldsymbol{b}\rangle) = \mathsf{nfSmt}\,(\langle \pi^{-1}\cdot \boldsymbol{a}]\,f\,[\pi^{-1}\cdot \boldsymbol{b}\rangle)\\
&\mathsf{nfSmt}\,(\gamma) = \gamma \text{ where } \gamma\in\Sigma\\
&\mathsf{nfSmt}\,(id) = id\\
&\mathsf{nfSmt}\,(\sigma) = \sigma\\
&\mathsf{nfSmt}\,(f\,;g) = \mathsf{nfSmt}\,(f)\,;\mathsf{nfSmt}\,(g)\\
&\mathsf{nfSmt}\,(f\oplus g) = \mathsf{nfSmt}\,(f)\oplus\mathsf{nfSmt}\,(g)
\end{aligned}$$

Definition 6.16. We define $\mathsf{Smt} : \mathsf{NMT} \to \mathsf{SMT}$ as

$$\mathsf{Smt}\,\langle\Sigma, E\rangle = \langle\Sigma, \mathsf{nfSmt}(\mathrm{dia}(E))\rangle.$$

In terms of Figure 11, the definition above corresponds to going around the square down-right. To prepare us for the next subsection, we also recall how to around right-down. This is the converse to the construction at the start of Section 6.2, now taking an NMT $\langle\Sigma, E\rangle$ to $\langle\mathsf{Trm}(\mathrm{dia}(\mathsf{nTrm}(\Sigma))), \mathrm{dia}(E)\cup\mathbf{ORD}\rangle$, where:

- $\mathrm{dia}(t : A \to B) = \langle \boldsymbol{a}]\,t\,[\boldsymbol{b}\rangle$ where $set(\boldsymbol{a}) = A$ and $set(\boldsymbol{b}) = B$, which is extended to a set of equations in the obvious way $\mathrm{dia}(E) = \{\langle \boldsymbol{a}]\,s\,[\boldsymbol{b}\rangle = \langle \boldsymbol{a}]\,t\,[\boldsymbol{b}\rangle \mid s = t \in E\}$
- $\mathbf{ORD}$ are the equations from Proposition 6.6

In the next section, we show that this construction and the one in Definition 6.16 give rise to the same PROPs.

6.8. Completeness of SMTs.

While Section 6.6 showed how to transfer completeness of SMTs to completeness of NMTs, we now go into the opposite direction. Indeed, we show how to prove the completeness of the $\mathsf{Smt}(\Sigma, E)$ from the completeness of a nominal monoidal theory $\langle\Sigma, E\rangle$. The central observation is that the diagram in Figure 11 commutes up to isomorphism.

We set up some preliminaries. First, we define $\iota : \mathsf{Trm}(\Sigma) \to \mathsf{Trm}(\mathrm{dia}(\mathsf{nTrm}(\Sigma)))$:

$$\begin{aligned}
&\iota\,(\underline{\gamma}) = \langle \boldsymbol{a}][\boldsymbol{a}\rangle\gamma\langle \boldsymbol{b}][\boldsymbol{b}\rangle \text{ where } \gamma\in\Sigma\\
&\iota\,(id) = id\\
&\iota\,(\Sigma) = \Sigma\\
&\iota\,(f\,;g) = \iota\,(f)\,;\,\iota\,(g)\\
&\iota\,(f\oplus g) = \iota\,(f)\oplus\iota\,(g)
\end{aligned}$$

Next, we show that the maps nfSmt and ι are inverses of each other.

Lemma 6.17. *The diagram in Figure 11 commutes up to isomorphism.*

$$\begin{tikzcd}
\mathsf{NMT} \arrow[rrr, "\mathsf{nProp}"] \arrow[dddd, "\mathsf{Smt}"'] & & & \mathsf{nPROP} \arrow[dddd, "\mathsf{ORD}"] \\
& \langle \Sigma, E \rangle \arrow[r, mapsto] \arrow[ddr, mapsto] & \mathsf{nProp}\,\langle \Sigma, E \rangle \arrow[d, mapsto] & \\
& & \mathsf{ORD}(\mathsf{nProp}\,\langle \Sigma, E \rangle) \arrow[d, bend left, "\mathsf{nfSmt}"] & \\
& & \mathsf{Prop}(\mathsf{Smt}\,\langle \Sigma, E \rangle) \arrow[u, bend left, "\iota"] & \\
\mathsf{SMT} \arrow[rrr, "\mathsf{Prop}"] & & & \mathsf{PROP}
\end{tikzcd}$$

Figure 11: Completing the square

Proof. We want to show that the two maps nfSmt and ι are isomorphisms. By definition, both nfSmt and ι are homomorphisms between the term algebras. To verify that these maps are well-defined, that is, that they respect equivalence of Trms, we need to check that they preserve the following equations:

- For the map ι, we have to show

$$\iota\,[\mathcal{Th}(\mathsf{nfSmt}[\mathrm{dia}[E]] \cup \mathbf{SMT})] \subseteq \mathcal{Th}(\mathcal{Th}(\mathrm{dia}[E \cup \mathbf{NMT}]) \cup \mathbf{ORD} \cup \mathbf{SMT})$$

  In fact, by Lemma 6.3, it suffices to check that $\iota\,[\mathsf{nfSmt}[\mathrm{dia}[E]]] \subseteq \mathcal{Th}(\mathrm{dia}[E \cup \mathrm{dia}[\mathbf{NMT}] \cup \mathbf{ORD} \cup \mathbf{SMT})$ and $\iota\,[\mathbf{SMT}] \subseteq \mathcal{Th}(\mathbf{SMT})$. The first inequality follows from $\iota\,[\mathsf{nfSmt}[\mathrm{dia}[E]] \overset{\mathrm{ORD}}{=} \mathrm{dia}[E]$, where $\overset{\mathrm{ORD}}{=}$ is equality up to the equations $\mathbf{ORD} \cup \mathbf{SMT} \cup \mathrm{dia}[\mathbf{NMT}]$. The second is immediate.
- For the map nfSmt, we have to show the other direction

$$\mathsf{nfSmt}[\mathcal{Th}(\mathcal{Th}(\mathrm{dia}[E \cup \mathbf{NMT}]) \cup \mathbf{ORD} \cup \mathbf{SMT})] \subseteq \mathcal{Th}(\mathsf{nfSmt}[\mathrm{dia}[E]] \cup \mathbf{SMT})$$

  for which it suffices:

$$\begin{aligned}
&\mathsf{nfSmt}[\mathcal{Th}(\mathcal{Th}(\mathrm{dia}[E \cup \mathbf{NMT}]) \cup \mathbf{ORD} \cup \mathbf{SMT})] \\
&\subseteq \mathcal{Th}(\mathsf{nfNmt}[\mathrm{dia}[E]] \cup \mathsf{nfSmt}[\mathbf{SMT}] \cup \mathsf{nfSmt}[\mathrm{dia}[\mathbf{NMT}]] \cup \mathsf{nfSmt}[\mathbf{ORD}]) \\
&\subseteq \mathcal{Th}(\mathsf{nfSmt}[\mathrm{dia}[E]] \cup \mathbf{SMT})
\end{aligned}$$

  To justify the last inequality, we show:
  - $\mathsf{nfSmt}[\mathrm{dia}[E]] \cup \mathsf{nfSmt}[\mathbf{SMT}] \subseteq \mathcal{Th}(\mathsf{nfSmt}[\mathrm{dia}[E]] \cup \mathbf{SMT})$, which is immediate.
  - $\mathsf{nfSmt}[\mathrm{dia}[\mathbf{NMT}]] \subseteq \mathcal{Th}(\mathsf{nfSmt}[\mathrm{dia}[E]] \cup \mathbf{SMT})$ It is easy enough to see that most equations of $\mathsf{nfSmt}[\mathrm{dia}[\mathbf{NMT}]]$ are in $\mathcal{Th}(\mathbf{SMT})$. For the interesting case of (NMT-comm) being preserved by $\mathsf{nfSmt} \circ \mathrm{dia}$, see the proof of [Bal20, Proposition 7.25].
  - $\mathsf{nfSmt}[\mathbf{ORD}] \subseteq \mathcal{Th}(\mathsf{nfSmt}[\mathrm{dia}[E]] \cup \mathbf{SMT})$. The only two equations which require any serious verification are (ORD-4) and (ORD-5). The proofs of both are essentially the

same, so we will only consider the first one here:

$$\begin{aligned}
\mathsf{nfSmt}(\langle \boldsymbol{a}][\boldsymbol{a}'|\boldsymbol{b}]; f[\boldsymbol{c}\rangle) &= \mathsf{nfSmt}(\langle \boldsymbol{a}][\boldsymbol{a}'|\boldsymbol{b}][\boldsymbol{b}'\rangle) \,;\, \mathsf{nfSmt}(\langle \boldsymbol{b}'] f[\boldsymbol{c}\rangle) \\
&\overset{\mathrm{SMT}}{=} \mathsf{nfSmt}(\langle \boldsymbol{a}][\boldsymbol{a}'|\boldsymbol{b}][\boldsymbol{b}\rangle) \,;\, \mathsf{nfSmt}(\langle \boldsymbol{b}] f[\boldsymbol{c}\rangle) \\
&\overset{\mathrm{SMT}}{=} \langle \boldsymbol{a}|\boldsymbol{a}'\rangle \,;\, \mathsf{nfSmt}(\langle \boldsymbol{b}] f[\boldsymbol{c}\rangle) \\
&= \mathsf{nfSmt}(\langle \boldsymbol{a}|\boldsymbol{a}'\rangle) \,;\, \mathsf{nfSmt}(\langle \boldsymbol{b}] f[\boldsymbol{c}\rangle) \\
&= \mathsf{nfSmt}(\langle \boldsymbol{a}|\boldsymbol{a}'\rangle \,;\, \langle \boldsymbol{b}] f[\boldsymbol{c}\rangle)
\end{aligned}$$

For these equalities to hold, we need to show

$$\mathsf{nfSmt}(\langle \boldsymbol{a}]t[\boldsymbol{b}'\rangle) \,;\, \langle \boldsymbol{b}'|\boldsymbol{b}\rangle \overset{\mathrm{SMT}}{=} \mathsf{nfSmt}(\langle \boldsymbol{a}]t[\boldsymbol{b}\rangle)$$

which follows by induction on t.

To show that nfSmt and ι are inverses, we have $\mathsf{nfSmt} \circ \iota\ (f) \overset{\mathrm{SMT}}{=} f$ for any $f \in \mathsf{Trm}(\Sigma)$, by induction on f. The only case of interest is $f = \underline{\gamma}$ where $\gamma \in \Sigma$:

$$\mathsf{nfSmt} \circ \iota\ (\underline{\gamma}) = \mathsf{nfSmt}\,(\langle \boldsymbol{a}][\boldsymbol{a}\rangle \gamma \langle \boldsymbol{b}][\boldsymbol{b}\rangle) = \langle \boldsymbol{a}|\boldsymbol{a}\rangle \,;\, \underline{\gamma} \,;\, \langle \boldsymbol{b}|\boldsymbol{b}\rangle \overset{\mathrm{SMT}}{=} \underline{\gamma}$$

Finally, we have $\iota \circ \mathsf{nfSmt}\,(f) \overset{\mathrm{ORD}}{=} f$ for any $f \in \mathsf{Trm}(\mathsf{dia}(\mathsf{nTrm}(\Sigma)))$. □

To conclude this section, we give an analogous result to Theorem 6.15 below.

Theorem 6.18 (Completeness of SMTs)**.** *Given an* NMT $\langle \Sigma, E\rangle$, *which is complete for some* $\mathsf{n}\mathbb{C}$, *s.t.* $\mathsf{ORD}(\mathsf{n}\mathbb{C}) \cong \mathbb{C}$, *we show that* $\mathsf{Smt}\,\langle \Sigma_{\mathbb{F}}, E_{\mathbb{F}}\rangle$ *is complete for* $\mathbb{C}$.

Proof. From Lemma 6.17 we know that

$$\mathsf{ORD}(\mathsf{nProp}\,\langle \Sigma, E\rangle) \cong \mathsf{Prop}(\mathsf{Smt}\,\langle \Sigma, E\rangle)$$

From completeness of $\langle \Sigma, E\rangle$ for $\mathsf{n}\mathbb{C}$ we know

$$\mathsf{nProp}\,\langle \Sigma, E\rangle \cong \mathsf{n}\mathbb{C}$$

Putting these together, we obtain

$$\mathsf{Prop}(\mathsf{Smt}\,\langle \Sigma, E\rangle) \cong \mathbb{C}$$

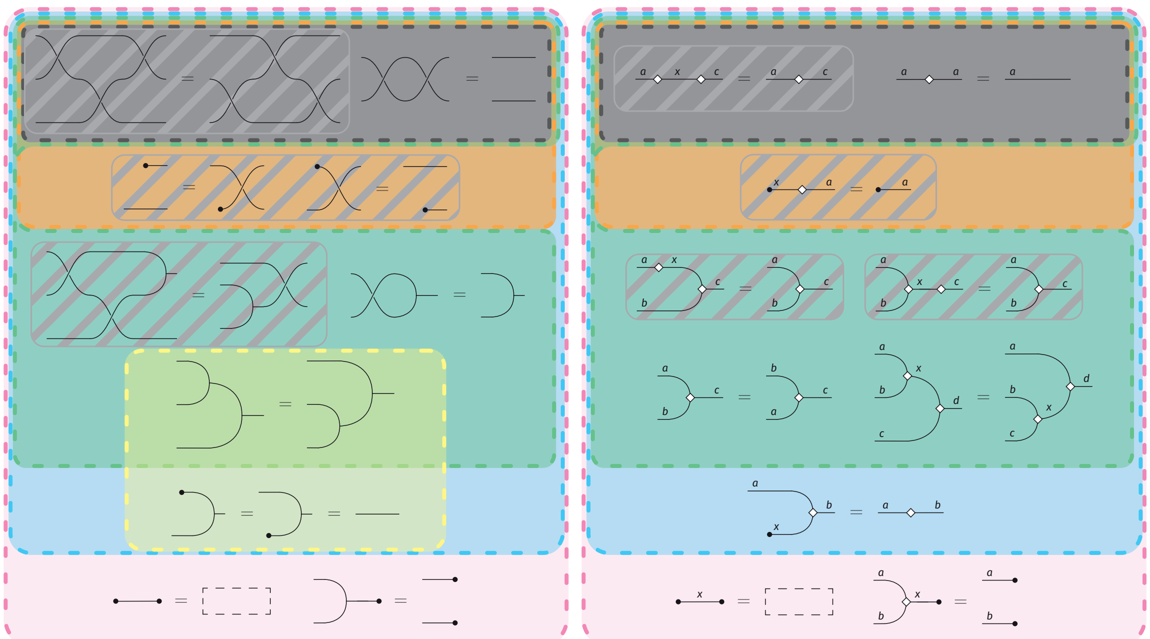
□

The theorem above can be useful when trying to prove soundness and completeness of an ordinary SMT, where the presented NMT is easier to prove sound and complete, like in the case of bijections, discussed previously at the end of Section 5.4.

Example 6.19. The completeness proof for ordinary string diagrams given in [Laf03] is already quite complex in the case of bijections because it requires a complete calculus for permutations. On the other hand, the corresponding nominal theory of bijections has a trivial completeness proof because all nominal string diagrams in the theory of bijections normalise to a disjoint union of renamings. We can now apply the theorem above to obtain an easy completeness proof for the corresponding symmetric monoidal theorry of bijections.

7. Conclusion

The equivalence of nominal and ordinary PROPs (Theorem 6.10), as well as the equivalence of nominal and symmetric monoidal theories (Theorems 6.18 and 6.15) has a satisfactory graphical interpretation. Indeed, comparing Figure 3 and Figure 8, truncated and shown side by side below, we see that both share, modulo different labellings of wires mediated by the functors ORD and NOM, the same core of generators and equations. The difference lies only in the equations expressing, on the one hand, that $\oplus$ has natural symmetries and, on the other hand, that generators are a nominal set and that $\uplus$ is commutative. In fact, this can be taken as a justification of the importance of the naturality of symmetries, which, informally speaking, compensates for the irrelevant detail induced by ordering names.



There are several directions for future research. First, the notion of an internal monoidal category has been developed because it is easier to prove the basic results in general rather than only in the special case of nominal sets. Nevertheless, it would be interesting to explore whether there are more interesting instances of internal monoidal categories in other semi-cartesian closed categories.

Second, internal monoidal categories are a principled way to build monoidal categories with a partial tensor. By working internally in the category of nominal sets with the separated product we can capture in a natural way constraints such as the tensor $f \oplus g$ for two partial maps $f, g : \mathcal{N} \to V$ being defined only if the domains of f and g are disjoint. This reminds us of the work initiated by O'Hearn and Pym on categorical and algebraic models for separation logic and other resource logics, see e.g. [OP99, GMP05, DGS15]. It seems promising to investigate how to build categorical models for resource logics based on internal monoidal theories. In one direction, one could extend the work of Curien and Mimram [CM17] to partial monoidal categories.

Third, there has been substantial progress in exploiting Lack's work on composing PROPs [Lac04] in order to develop novel string diagrammatic calculi for a wide range of applications, see e.g. [BGK+16, BSZ17]. It will be interesting to explore how much of this technology can be transferred from PROPs to nominal PROPs.

Fourth, various applications of nominal string diagrams could be of interest. The original motivation for our work was to obtain a convenient calculus for simultaneous substitutions that can be integrated with multi-type display calculi [FGK+16] and, in particular, with the multi-type display calculus for first-order logic of Tzimoulis [Tzi18]. Another direction for applications comes from the work of Ghica and Lopez [GL17] on a nominal syntax for string diagrams. In particular, it would be of interest to add various binding operations to nominal PROPs.

More generally, we expect nominal PROPs to play a role as an intermediate level of abstraction for the implementation of programming languages that have a denotational semantics as string diagrams. As case study, it would be interesting to look at the Bayesian Networks of [JZ18, JKZ19].

Appendix A. A review of internal category theory

We review the notation that we use in Section 4. We consulted Borceux, Handbook of Categorical Algebra, Volume 1, Chapter 8 and the nLab, adapting the notation to our needs.

Definition A.1 (internal category)**.** In a category with finite limits an *internal category* is a diagram

$$\begin{tikzcd} A_3 \arrow[r, shift left=4.5, "right"] \arrow[r, shift left=1.5, "compr"] \arrow[r, shift right=1.5, "compl"] \arrow[r, shift right=4.5, "left"] & A_2 \arrow[r, shift left=3, "\pi_2"] \arrow[r, "\mathbf{comp}"] \arrow[r, shift right=3, "\pi_1"] & A_1 \arrow[r, shift left=3, "\mathbf{dom}"] \arrow[r, shift right=3, "\mathbf{cod}"] & A_0 \arrow[l, "i"] \end{tikzcd} \tag{A.1}$$

where

(1) the diagram $\begin{tikzcd} A_2 \arrow[r, "\pi_2"] \arrow[d, "\pi_1"'] & A_1 \arrow[d, "\mathbf{dom}"] \\ A_1 \arrow[r, "\mathbf{cod}"] & A_0 \end{tikzcd}$ is a pullback,

(2) $\mathbf{dom} \circ \mathbf{comp} = \mathbf{dom} \circ \pi_1$ and $\mathbf{cod} \circ \mathbf{comp} = \mathbf{cod} \circ \pi_2$,

(3) $\mathbf{dom} \circ i = id_{A_0} = \mathbf{cod} \circ i$,

(4) $\mathbf{comp} \circ \langle i \circ \mathbf{dom}, id_{A_1} \rangle = id_{A_1} = \mathbf{comp} \circ \langle id_{A_1}, i \circ \mathbf{cod} \rangle$

(5) $\mathbf{comp} \circ \mathbf{compl} = \mathbf{comp} \circ \mathbf{compr}$

where

- $\langle i \circ \mathbf{dom}, id_{A_1} \rangle : A_1 \to A_2$ and $\langle id_{A_1}, i \circ \mathbf{cod} \rangle : A_1 \to A_2$ are the arrows into the pullback A_2 pairing $i \circ \mathbf{dom}, id_{A_1} : A_1 \to A_1$ and $id_{A_1}, i \circ \mathbf{cod} : A_1 \to A_1$, respectively;
- the "triple of arrows"-object A_3 is the pullback

$$\begin{tikzcd} A_3 \arrow[r, "right"] \arrow[d, "left"'] & A_2 \arrow[d, "\pi_1"] \\ A_2 \arrow[r, "\pi_2"] & A_1 \end{tikzcd}$$

  where, intuitively, *left* "projects out the left two arrows" and *right* "projects out the right two arrows";

- **compl** is the arrow composing the "left two arrows"

$$\begin{tikzcd}
A_3 \arrow[dr, dotted, "\mathbf{compl}"] \arrow[drr, bend left, "\pi_2 \circ \mathit{right}"] \arrow[ddr, bend right, "\mathbf{comp} \circ \mathit{left}"'] & & \\
& A_2 \arrow[r, "\pi_2"] \arrow[d, "\pi_1"'] & A_1 \arrow[d, "\mathbf{dom}"] \\
& A_1 \arrow[r, "\mathbf{cod}"] & A_0
\end{tikzcd}$$

- **compr** is the arrow composing the "right two arrows"

$$\begin{tikzcd}
A_3 \arrow[dr, dotted, "\mathbf{compr}"] \arrow[drr, bend left, "\mathbf{comp} \circ \mathit{right}"] \arrow[ddr, bend right, "\pi_1 \circ \mathit{left}"'] & & \\
& A_2 \arrow[r, "\pi_2"] \arrow[d, "\pi_1"'] & A_1 \arrow[d, "\mathbf{dom}"] \\
& A_1 \arrow[r, "\mathbf{cod}"] & A_0
\end{tikzcd}$$

Remark A.1. 1. and 2. define A_2 as the 'object of composable pairs of arrows' while 3. and 4. express that the 'object of arrows' A_1 has identities and 5. formalises associativity of composition. Since A_2 and A_3 are pullbacks, the structure is defined completely by $(A_0, A_1, \mathbf{dom}, \mathbf{cod}, i, \mathbf{comp})$ only, but including A_3 as well as $\mathbf{compr}, \mathbf{compl}, \mathit{right}, \mathit{left}, \pi_2, \pi_1$ helps writing out the equations.

Definition A.2. A morphism $f : A \to B$ between internal categories, an *internal functor*, is a pair (f_0, f_1) of arrows such that the six squares (one for each of π_2, **comp**, π_1, **dom**, **cod**, i)

$$\begin{tikzcd}
A_2 \arrow[r, shift left=2, "\pi_2"] \arrow[r, "\mathbf{comp}" description] \arrow[r, shift right=2, "\pi_1"'] \arrow[d, "f_2"'] & A_1 \arrow[r, shift left=2, "\mathbf{dom}"] \arrow[r, shift right=2, "\mathbf{cod}"'] \arrow[d, "f_1"'] & A_0 \arrow[l, "i" description] \arrow[d, "f_0"'] \\
B_2 \arrow[r, shift left=2, "\pi_2"] \arrow[r, "\mathbf{comp}" description] \arrow[r, shift right=2, "\pi_1"'] & B_1 \arrow[r, shift left=2, "\mathbf{dom}"] \arrow[r, shift right=2, "\mathbf{cod}"'] & B_0 \arrow[l, "i" description]
\end{tikzcd} \tag{A.2}$$

commute.

Remark A.2. • Because B_2 is a pullback f_2 is uniquely determined by f_1. In more detail, if $\Gamma \to B_2$ is any arrow then, because B_2 is a pullback, it can be written as a pair

$$\langle l, r \rangle : \Gamma \to B_2 \tag{A.3}$$

of arrows $l, r : \Gamma \to B_1$ and f_2 is determined by f_1 via

$$f_2 \circ \langle l, r \rangle = \langle f_1 \circ l, f_1 \circ r \rangle \tag{A.4}$$

- Even if f_2 is not needed as part of the structure in the above definition, including f_2 makes it easier to state that f_1 preserves composition.
- Similarly, B_3 is a pullback, and there is a unique arrow f_3 such that (f_0, f_1, f_2, f_3) together make further 4 squares commute, one for each of *right*, **compr**, **compl**, *left*, see (A.1). We may include f_3 in the structure whenever convenient.

Definition A.3. A natural transformation $\alpha : f \to g$ between internal functors $f, g : A \to B$, an *internal natural transformation*, is an arrow $\alpha : A_0 \to B_1$ such that, recalling (A.3),

$$\mathbf{dom} \circ \alpha = f_0 \qquad \mathbf{cod} \circ \alpha = g_0 \qquad \mathbf{comp} \circ \langle f_1, \alpha \circ \mathbf{cod} \rangle = \mathbf{comp} \circ \langle \alpha \circ \mathbf{dom}, g_1 \rangle$$

Remark A.3. Internal categories with functors and natural transformations form a 2-category. We denote by $\mathsf{Cat}(\mathcal{V})$ the category or 2-category of categories internal in $\mathcal{V}$. The forgetful functor $\mathsf{Cat}(\mathcal{V}) \to \mathcal{V}$ mapping an internal category A to its object of objects A_0 has both left and right adjoints and, therefore, preserves limits and colimits. Moreover, a limit of internal categories is computed component-wise as $(\lim D)_j = \lim(D_j)$ for $j = 0, 1, 2$.

Remark A.4. A monoidal category can be thought of both as a monoid in the category of categories and as an internal category in the category of monoids. To understand this in more detail, note that both definitions give rise to the diagram

$$\begin{tikzcd}
A_2 \times A_2 \arrow[r, "\mathbf{comp} \times \mathbf{comp}"] \arrow[d, "m_2"'] & A_1 \times A_1 \arrow[r, shift left, "\mathbf{dom} \times \mathbf{dom}"] \arrow[r, shift right, "\mathbf{cod} \times \mathbf{cod}"'] \arrow[d, "m_1"'] & A_0 \times A_0 \arrow[d, "m_0"'] \\
A_2 \arrow[r, "\mathbf{comp}"] & A_1 \arrow[r, shift left, "\mathbf{dom}"] \arrow[r, shift right, "\mathbf{cod}"'] & A_0
\end{tikzcd}$$

where

- in the case of a monoid A in the category of internal categories, $m = (m_0, m_1, m_2)$ is an internal functor $A \times A \to A$ and, using that products of internal categories are computed component-wise, we have $\mathbf{comp} \circ m_2 = m_1 \circ (\mathbf{comp} \times \mathbf{comp})$, which gives us the interchange law
$$(f; g) \cdot (f'; g') = (f \cdot f') \,;(g \cdot g')$$
by using (A.4) with m for f and writing ; for **comp** and $\cdot$ for m_1;
- in the case of a category internal in monoids we have monoids A_0, A_1, A_2 and monoid homomorphisms $i, \mathbf{dom}, \mathbf{cod}, \mathbf{comp}$ which, if spelled out, leads to the same commuting diagrams as the previous item.

Remark A.5. In Section 4 we answer the question of how to internalise the cartesian product $A \times A$ in a monoidal category $(\mathcal{V}, I, \otimes)$. In other words, where above we use $A_0 \times A_0$ we now want $A_0 \otimes A_0$. This rules out the second item above, leaving only the definition of a monoidal category as a monoid in the category of internal categories.

References

[BK19] Samuel Balco, Alexander Kurz: Nominal String Diagrams. *CALCO'19*.

[Bal20] Samuel Balco. *Display calculi and nominal string diagrams*. Phd thesis, University of Leicester, 2020.

[BCST96] R.F. Blute, J.R.B. Cockett, R.A.G. Seely, and T.H. Trimble. Natural deduction and coherence for weakly distributive categories. *Journal of Pure and Applied Algebra*, 113(3), 1996.

[BGK+16] Filippo Bonchi, Fabio Gadducci, Aleks Kissinger, Paweł Sobociński, and Fabio Zanasi. Rewriting modulo symmetric monoidal structure. In *LICS '16*.

[BKL11] Mikołaj Bojańczyk, Bartek Klin, and Slawomir Lasota. Automata with group actions. In *LICS'11*.

[BSZ15] Filippo Bonchi, Paweł Sobociński, and Fabio Zanasi. Full Abstraction for Signal Flow Graphs. In *POPL '15*.

[BSZ17] Filippo Bonchi, Paweł Sobociński, and Fabio Zanasi. The Calculus of Signal Flow Diagrams I: Linear relations on streams. *Information and Computation*, 252(C), 2017.

[CK17] Bob Coecke and Aleks Kissinger. *Picturing Quantum Processes: A First Course in Quantum Theory and Diagrammatic Reasoning*. Cambridge University Press, 2017.

[CM17] Pierre-Louis Curien and Samuel Mimram. Coherent Presentations of Monoidal Categories. *Logical Methods in Computer Science*, 13(3), 2017.

[CS18] J. R. B. Cockett and R. A. G. Seely. Proof theory of the cut rule. In *Categories for the Working Philosopher*. Oxford University Press, 2018.

[DGS15] Brijesh Dongol, Victor B. F. Gomes, and Georg Struth. A program construction and verification tool for separation logic. In *MPC'15*.

[FGK+16] Sabine Frittella, Giuseppe Greco, Alexander Kurz, Alessandra Palmigiano, and Vlasta Sikimic. Multi-type display calculus for dynamic epistemic logic. *Journal of Logic and Computation*, 26(6), 2016.

[FMT05] Gian Luigi Ferrari, Ugo Montanari, and Emilio Tuosto. Model checking for nominal calculi. In *FOSSACS'05*.

[FPT99] Marcelo P. Fiore, Gordon D. Plotkin, and Daniele Turi. Abstract syntax and variable binding. In *LICS'99*.

[Gir87] Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50(1), 1987.

[GL17] Dan R. Ghica and Aliaume Lopez. A Structural and Nominal Syntax for Diagrams. In *QPL'17*, 2017.

[GLP11] Murdoch James Gabbay, Tadeusz Litak, and Daniela Petrisan. Stone duality for nominal boolean algebras. In *CALCO'11*.

[GM09] Murdoch James Gabbay and Aad Mathijssen. Nominal (universal) algebra: Equational logic with names and binding. *J. Log. Comput.*, 19(6), 2009.

[GMP05] D. Galmiche, D. Méry, and D. Pym. The semantics of bi and resource tableaux. *Mathematical. Structures in Comp. Sci.*, 15(6), 2005.

[GP] M. Gabbay and A. Pitts. A new approach to abstract syntax involving binders. In *LICS'99*.

[Hof99] Martin Hofmann. Semantical analysis of higher-order abstract syntax. In *LICS'99*.

[Hot65] Günter Hotz. Eine Algebraisierung des Syntheseproblems von Schaltkreisen I. *Elektronische Informationsverarbeitung und Kybernetik*, 1:185–205, 1965.

[HR14] Philip Hackney and Marcy Robertson. On the Category of Props. *Applied Categorical Structures*, 23(4), March 2014.

[Jac99] Bart Jacobs, Fabio Zanasi: The Logical Essentials of Bayesian Reasoning. CoRR abs/1804.01193 (2018).

[JKZ19] Bart Jacobs, Aleks Kissinger, Fabio Zanasi: Causal Inference by String Diagram Surgery. *FoSSaCS'19*.

[JZ18] Bart Jacobs. *Categorical Logic and Type Theory*. Elsevier, 1999.

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

[JS93] André Joyal and Ross Street. Braided Tensor Categories. *Advances in Mathematics*, 102(1), November 1993.

[Lac04] Steve Lack. Composing PROPs. *Theory and Applications of Categories*, 13, 2004.

[Laf03] Yves Lafont. Towards an algebraic theory of Boolean circuits. *Journal of Pure and Applied Algebra*, 184, 2003.

[Law69] Francis William Lawvere. Adjointness in Foundations. *Dialectica*, 23, 1969.

[LP14] Steffen Lösch and Andrew M. Pitts. Denotational semantics with nominal scott domains. *J. ACM*, 61(4), 2014.

[Mas53] Samuel Mason. Feedback Theory-Some Properties of Signal Flow Graphs. *Proceedings of the IRE*, 41(9), September 1953.

[Mel06] Paul-André Melliès. Functorial Boxes in String Diagrams. In *CSL*'06.

[Mil06] Robin Milner. Pure bigraphs: Structure and dynamics. *Information and Computation*, 204(1), 2006.

[ML65] Saunders Mac Lane. Categorical algebra. *Bulletin of the American Mathematical Society*, 71(1), 1965.

[ML78] Saunders Mac Lane. *Categories for the Working Mathematician*. Springer, 1978.

[MP98] U. Montanari and M. Pistore. History dependent automata. Technical Report TR-11-98, Università di Pisa, Dipartimento di Informatica, 1998.

[MSS+17] Joshua Moerman, Matteo Sammartino, Alexandra Silva, Bartek Klin, and Michal Szynwelski. Learning nominal automata. In *POPL'17*.

[OP99] Peter W. O'Hearn and David J. Pym. The logic of bunched implications. *The Bulletin of Symbolic Logic*, 5(2), 1999.

[Pen71] Roger Penrose. Applications of Negative Dimensional Tensors. *Combinatorial Mathematics and its Applications.* page 221-244. Academic Press, 1971.

[Pet11] Daniela Petrisan. *Investigations into Algebra and Topology over Nominal Sets*. PhD thesis, 2011.

[Pit13] Andrew M. Pitts. *Nominal Sets: Names and Symmetry in Computer Science*. Cambridge University Press, 2013.

[Prz98] Józef H. Przytycki. Classical Roots of Knot Theory. *Chaos, Solitons & Fractals*, 9(4-5), 1998.

[Sel10] P. Selinger. A Survey of Graphical Languages for Monoidal Categories. In *New Structures for Physics*. Springer, 2010.

[Str99] Thomas Streicher. Fibred Categories à la Jean Bénabou, 1999.

[TW09] David Turner and Glynn Winskel. Nominal domain theory for concurrency. In *CSL'09*.

[Tzi18] Apostolos Tzimoulis. *Algebraic and Proof-Theoretic Foundations of the Logics for Social Behaviour*. Phd thesis, Delft University of Technology, 2018.

[Zan15] Fabio Zanasi. *Interacting Hopf Algebras- the Theory of Linear Systems*. Phd thesis, Ecole Normale Supérieure de Lyon - ENS LYON, October 2015.

[Zan17] Fabio Zanasi. Rewriting in Free Hypegraph Categories. In *Third Workshop on Graphs as Models*, 2017.