1

Generalised Proof-Nets for Compact Categories with Biproducts

Abstract

Just as conventional functional programs may be understood as proofs in an intuitionistic logic, so quantum processes can also be viewed as proofs in a suitable logic. We describe such a logic, the logic of compact closed categories and biproducts, presented both as a sequent calculus and as a system of proof-nets. This logic captures much of the necessary structure needed to represent quantum processes under classical control, while remaining agnostic to the fine details. We demonstrate how to represent quantum processes as proof-nets, and show that the dynamic behaviour of a quantum process is captured by the cut-elimination procedure for the logic. We show that the cut elimination procedure is strongly normalising: that is, that every legal way of simplifying a proof-net leads to the same, unique, normal form. Finally, taking some initial set of operations as non-logical axioms, we show that that the resulting category of proof-nets is a representation of the free compact closed category with biproducts generated by those operations.

1.1 Introduction

1.1.1 Logic, Processes, and Categories

Birkhoff and von Neumann initiated the logical study of quantum mechanics in their 1936 paper [BvN36]. They constructed a logic by assigning a proposition letter to each observable property of a given quantum system, and studied negations, conjunctions, and disjunctions of these properties. The resulting lattice is non-distributive, and so the heart of what is called "quantum logic"is the study of various kinds of non-distributive lattices. These traditional quantum logics suffer from a number of defects. Firstly, they are monolithic: there is no way to derive the properties of a composite system from the properties of its parts. Each system has its own associated lattice which can only rarely be related to those of other systems. Further, these systems are seen statically: a system that undergoes some dynamical change is a new system. Finally, the failure of compositionality is connected with the fact that quantum logic has no decent notion of implication [Sme01]; hence we have a logic which has a notion of validity, but no concept of *inference* or *proof*.

These limitations form a serious obstacle to the use of Birkhoff-von Neumann-style quantum logic to study *interacting* quantum systems. If by "system" we understand the ubiquitous qubit, then it is precisely such interactions which form the basis of quantum informatics. Indeed, the principal concern of the computer scientist†—how to soundly construct large systems from smaller ones—is exactly where quantum logic is weakest.

In this article I will describe a very different kind of logic which can address these questions. This logic is called *tensor-sum logic* and it shares many features with linear logic [Gir87a], which has been widely studied in computer science and structural proof theory.

We proceed in accordance with an old tradition in computer science, that of linking computational systems and logics, of claiming that a certain logic "is the same as" some formal computing machine. The archetype of this approach is the Curry-Howard correspondence between intuitionistic natural deduction [Gen35, Pra65] and the simply-typed λ-calculus [CF58, How80]. The basic insight is that the inference rules of the logic are essentially the same as the constructors used to form a λ-term. The valid types of the λ-calculus are nothing more than the theorems of the logic, and more importantly every proof represents a λ-term. We can thus view the logic itself as a computational system, with the important proviso that the objects of interest are the *proofs* and not the theorems. This correspondence is not just skin deep. Recall that the β-reduction relation between λ-terms expresses the *execution* behaviour of the calculus; we consider terms to be computationally equivalent when they reduce to the same β-normal form. This dynamical aspect of the λ-calculus corresponds exactly to the normalisation procedure for natural deduction proofs or, in the sequent calculus setting, to the *cut-elimination* procedure [Gen35]. In our presentation of

† At least: the principal concern of *many* computer scientists.

tensor-sum logic we will take this correspondence as given, and treat the proof theoretic presentation as a computational system in its own right. The normalisation procedure gives the computational dynamics.

There is a further correspondence that we engage with rather more seriously: that between the proofs in a given logic and the arrows in a particular class of categories [Lam68, Lam69]. We can equate the propositions of a logic L with the objects of a category $\mathcal{C}$; for each proof of, say, a proposition B from a premise A, we define a corresponding arrow $f : A \to B$ in the category. Stated so blandly, we have little structure to work with, so we go further, and demand that the logical connectives are represented by functors on $\mathcal{C}$. The natural transformations between these functors then give rise to the inference rules of the logic. In this way an arrow in $\mathcal{C}$ may be constructed for each proof in L; we say that $\mathcal{C}$ is a *denotational model* of L if, whenever two proofs share the same normal form, they have equal denotations in $\mathcal{C}$. The categorical model gives an extensional account of the computational dynamics represented in the term language. The general schema of this tripartite relation is shown in Table 1.1.

Computation	**Logic**	**Category**
types	formulae	objects
terms	proofs	arrows
type formers	connectives	functors
term constructors	inference rules	natural transformations

Table 1.1. *Curry-Howard-Lambek correspondences*

To return to our earlier example, the simply-typed λ-calculus forms such a triple with intuitionistic natural deduction and the class of Cartesian closed categories [LS86]. Another example is provided by intuitionistic multiplicative linear logic [Gir87a]; this logical system corresponds to the *linear* λ-calculus and to the class of *-autonomous categories [Bar79, Bar91]. The pattern is quite general, and more examples can easily be found.

This relationship between a logic and its categorical models can be made exact. In the above description we embedded the connectives and inference rules of L into $\mathcal{C}$ using only functors and natural transformations: the logical structure is agnostic with respect to the concrete elements of the category. Hence any category with the requisite functors can provide a model, up to some assignment of the basic proposition letters. Of course such a category may well contain other objects or arrows which do not correspond to anything in the logic we are trying to model. To make the correspondence exact we must be able to translate from the *free category* (with appropriate structure) faithfully back to our logic. That is, we must find an injective translation from the arrows of $\mathcal{C}$ onto the cut-free proofs of L. In the case of our generalised proof-nets, that is indeed possible, but for a simpler example, consider the simply-typed λ-calculus with just one ground type; then the category of its terms is the free Cartesian closed category generated by the category **1**, which has only one object, and one identity arrow.

The choice of generator can be rather important. By choosing a discrete category (i.e. a set) the resulting logic will have that set as its propositional variables. By choosing a category with non-trivial arrows, we introduce non-logical axioms: if cut-elimination for the resulting logic is to be retained we must lift the composition operation of $\mathcal{C}$ into the cut-elimination procedure of L itself. If other equations between arrows are required these too must be hoisted into the logic. In the case we consider here, the situation is even worse: we will take as a generator a *compact symmetric polycategory* [Dun06]. This esoteric creature will be described in a later section, but for now we note that the intricacy of the required composition forces the adoption of the proof-net formalism, an illustration of the power of graphical methods over conventional syntax.

The development in the following sections will be the reverse of the exposition above. We first describe the mathematical basis of quantum computation in its concrete setting—finite dimensional Hilbert spaces—then we identify certain structures of the category **fdHilb** which are the essential features for carrying out quantum computation. Next we present the syntax of tensor-sum logic, and finally prove that the category of generalised proof-nets is the free compact closed category with biproducts generated by a compact symmetric polycategory.

Prior Work The original formulation of quantum mechanics in terms of compact closed categories with biproducts was due to Abramsky and Coecke [AC04]. An early attempt to formalise quantum computations in terms of proof-nets for **MLL** was [Dun04]. The first description of a logic based on compact closed categories and biproducts was given by Abramsky and the author in [AD06], however this logic is essentially restricted to quantum systems with only bipartite entanglement. This restriction was lifted in [Dun06] via the use of polycategories, although only the multiplicative fragment of the logic was treated. The present article is essentially a fusion of the previous two: giving a complete presentation of two-sided proof-nets with generalised axioms, and both the tensor and sum connectives. A notable distinction between the treatment of the biproduct here as compared to that of [AC04] is that here the biproduct is freely generated, hence the our treatment is closer to that of [Sel04]; we will discuss the vexed position of this connective at the end of Section 1.1.2.

1.1.2 Quantum Mechanics Concretely, and Abstractly

The main work of this article is to characterise the structure of certain kinds of category in terms of proof-nets. The particular categories we are interested in are *compact closed categories with biproducts*, and in the next section we'll go into considerable detail defining, characterising, and giving some of the basic properties of such categories. In this section we describe, at a more intuitive level, how the structure relates to quantum mechanics and quantum computation in particular.

We start with a schematic description of quantum mechanics. Since we are interested in quantum computation, we will restrict ourselves to quantum systems with finite dimensional state spaces. Consider the following axioms.

(i) To each quantum system we associate a finite dimensional Hilbert space, its *state space*; the possible states of the system are unit vectors in the state space, modulo a global phase factor.

(ii) If two systems are combined, then their joint state space is the *tensor product* of the two state spaces.

(iii) Measuring a quantum system is *non-deterministic*; possible outcomes are the eigenvectors of some self-adjoint operator on the state space, and the probability of observing a particular outcome depends on the inner product of the current state and the eigenvector for that outcome. After the measurement the state is updated to match the observed vector†, assuming it is not destroyed by the act of measuring.

(iv) For any discrete time step, the next state of the system is determined by a *unitary map* on its state space.

Of course, we have omitted some important details, but the above axioms approximate the level of abstraction of our categorical formalisation.

† We have implicitly assumed that the measurement is non-degenerate.

The first thing we should note that the **fdHilb**, the category of finite dimensional Hilbert spaces and linear maps, is the "natural" category in which to formalise the above axioms‡. The second point is that **fdHilb** is compact closed and has biproducts. Thirdly, the above axioms can be rephrased in terms of the categorical structure alone.

Before showing how the ingredients of the axioms translate into categorical terms, let's dispose of some unnecessary baggage. Consider a linear map $\psi : \mathbb{C} \to \mathcal{H}$. Since it is linear, and its domain is one-dimensional, its value is fixed by its value on 1, hence maps of this type are in 1-1 correspondence with vectors of $\mathcal{H}$. Hence we can forget about vectors and talk only of linear maps.

Now consider measurements. There are three parts to a quantum measurement: the non-deterministic possibilities, the calculation of the probabilities, and the updated state. We will deal with these in reverse order. The new state of a measured system depends only on which outcome i of the measurement happened, hence it is just a new state $|\phi_i\rangle$, with no particular relation to the old one. Of course, the measurement process may destroy the system, in which case there is no new state. To calculate the probability of seeing outcome i when we are in state $|\psi\rangle$ we must calculate the inner product $\langle\phi_i \mid \psi\rangle$. This process too can be seen as a linear map, namely the projection map $\langle\phi_i| : \mathcal{H} \to \mathbb{C}$; when composed with $|\psi\rangle$ this yields the inner product. Hence the state transformation associated to the ith outcome of some measurement is described by the map:

$$\mathcal{H} \xrightarrow{\langle\phi_i|} \mathbb{C} \xrightarrow{|\phi_i\rangle} \mathcal{H}\ .$$

Note that this is desired transformation even in the case when only part of a composite system is measured. For the purposes of this work, calculating the probabilities is not so important, but the transformation of the state is essential.

Finally let's consider the non-deterministic aspect of measurement. The main point is that there several possible outcomes *and we know which one happened*. Suppose we perform a measurement with two outcomes; we can view this process as map of type

$$\mathcal{H} \to \mathcal{H} \oplus \mathcal{H}$$

‡ As the first and last axioms suggest **fdHilb** is actually too big: it contains vectors and maps which do not correspond to anything in quantum mechanics. The general program of describing quantum mechanics in categorical terms aims to find the *minimal* structure required.

where the two sides of the direct sum correspond to the two "possible worlds" induced by the two outcomes of the measurement. One can then represent conditional operations by acting on only one subspace or the other; for example $f \oplus g$ behaves as the linear map $f : \mathcal{H} \to \mathcal{H}$ if the first outcome was observed, and $g : \mathcal{H} \to \mathcal{H}$ if the second was observed.

How then can we write the axioms of quantum mechanics in the language of compact closed categories and biproducts? Firstly, compact closed categories are equipped with a tensor product, and this tensor product has a neutral element I. In **fdHilb** the tensor product is simply the usual Kronecker product, and its neutral element is the base field, namely $\mathbb{C}$. This gives us the first two axioms:

- To each quantum system we associate an object A, its state space; the possible states of the system are given by arrows $\psi : I \to A$.
- If two systems with state spaces A and B respectively are combined, then their joint state space is $A \otimes B$.

In **fdHilb** the biproduct is the direct sum of Hilbert spaces, and this will allow the formalisation of measurements.

- An n-outcome measurement of a quantum system whose state space is A is represented by an arrow

  $$m : A \longrightarrow \bigoplus_i B_i$$

  where each of the the projections $\pi_i \circ m : A \to B_i$ factors as

  $$A \longrightarrow I \longrightarrow B_i \; .$$

This is very general notion of measurement. The common cases are when $B_i = I$ and the original system is destroyed; and, when $B_i = A$ when the original system is preserved.

Since we are interested in formalising as much of quantum mechanics as possible within one category we will not restrict state transformations to unitary evolutions; note that a measurement is a valid transformation of a quantum state which is not unitary. Hence the last axiom is simply the following.

- A quantum system A may transform to another system B by means of any arrow $f : A \to B$.

We have not yet mentioned the role of the compact structure of the category. While not required to paraphrase the axioms, the compact structure plays an important role in capturing quantum phenomena.

Recall that the tensor product of two vector spaces contains points, $\Psi : I \to A \otimes B$ which cannot be factored into a pair of vectors $\psi_1 : I \to A$ and $\psi_2 : I \to B$. The such quantum states are called *entangled* and they a central role in quantum computation. The compact structure guarantees the existence of certain entangled states, namely *Bell states* for every finite dimensional Hilbert space:

$$\eta_A : I \to A^* \otimes A \ .$$

If A is two-dimensional, i.e. a qubit, then the corresponding vector is

$$\eta_A(1) = |00\rangle + |11\rangle \ .$$

Further more, the compact structure also provides a projection onto this state

$$\epsilon_A : A \otimes A^* \to I$$

hence we can define measurements on Bell states. These two operations will allow many more entangled states to be defined.

This completes our impressionistic description of how quantum mechanics may be formalised in the categorical setting. Before moving on, it worth pointing out what has been excluded from our formalisation. Perhaps the most striking omission in moving between the concrete axioms and the abstract is the concept of *unitarity*.

The abstract formulation of quantum mechanics described here is derived from that introduced in [AC04] which uses *strongly* compact closed categories. Also called †-compact, these categories are equipped with a contravariant involutive functor which sends each map f to its adjoint $f^\dagger$, and has no effect on objects. This functor can then be used to define unitarity and the inner product. In this article, we focus on freely constructing the compact closed and biproduct structure from some underlying category of generators. One could consider the case when these structures cohere with the $(\cdot)^\dagger$ operation, giving a †-compact category with †-biproducts; however the only difference here between the †-structure and the original is that the structural isomorphisms are required to be unitary. That is to say that the only *new* maps which are introduced are the adjoints of the generators. Hence we can simply enlarge the class of generators beforehand, and thereafter ignore the †-structure. Of course, when working on concrete examples it is important to be aware of the adjoints, and the equational theory of the generators more generally, but that is not the focus of the present article.

The other important deviation from usual quantum mechanics is that we have been extremely liberal about measurements. In particular we do not make any restriction on the number of outcomes a measurement may have. Of course, in quantum mechanics the outcomes are the spectrum of some operator, and hence are bounded by the dimension of the space. Considering these issues would take us too far afield but [AC04] has one approach; a more recent categorical treatment of quantum observables is found in [CD08]. In any case, it seems unlikely that the structure of quantum measurements—being fundamentally connected to the bases of the underlying space—will yield to a description in terms of natural transformations of some functors.

Some Remarks on the Biproduct

In their original paper [AC04] Abramsky and Coecke used the biproduct of **fdHilb** in two roles: firstly, to encode classical branching, as described above; and secondly, to construct *bases* for the underlying space. In particular, they define state preparations and destructive measurements as isomorphisms of the forms

$$\mathrm{base} : \bigoplus_i I \to A \quad \text{and} \quad \mathrm{meas} : A \to I \bigoplus_i \ .$$

This second use of the biproduct has been criticised by later works [Coe05a, Sel05] on two main accounts. In the original approach the composite

$$A \xrightarrow{\mathrm{meas}} \bigoplus_i I \xrightarrow{\mathrm{meas}} A$$

yields the identity map, contradicting physical reality—a real experiment would transform a pure state to a mixed state, something not handled within this simple framework. More importantly, when moving from a "vector space" setting like **fdHilb** to a "projective" setting, such as Selinger's CPM construction or Coecke's WProj, the direct sum of the underlying space no longer yields a biproduct. The only option is to construct the biproduct as formal vectors and matrices. The works cited above show that, in the projective setting, if there is a biproduct then the scalars are essentially restricted to *probabilities* rather than *amplitudes*. The immediate consequence is that we must give up any hope of using the additive structure to encode interference effects: we are essentially restricted to a classical probabilistic setting.

The approach to biproducts taken in this work is absolutely consonant with these restrictions. We construct both the multiplicative and additive structures freely, and hence the scalars are simply a (free) semiring.

The theory of processes thus produced is much like that introduced in [Sel04], based on classically controlled quantum operations.

1.1.3 An Example Proof-net

Sections 1.3 and 1.4 will introduce tensor-sum logic, and its proof-net notation. Since those sections will focus on the technical details of the formalism we present now an illustrative example of how proof-nets can be used to to model quantum processes.

We will describe an old favourite: the quantum teleportation protocol [BBC+93]. The sketch of the protocol goes like this: Two parties, Alice and Bob, initially share an entangled pair of particles in a Bell state,

$$|\mathrm{Bell}\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}}$$

The parties then separate, and at some later point Alice wants to send a qubit to Bob, but unfortunately she has only a classical channel. However, it is still possible to transmit the qubit by using the shared entanglement between the two parties.

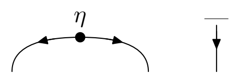
To proceed, Alice performs a joint measurement on the qubit she wishes to transmit together with her half of the entangled pair. She measures in the Bell basis, so her state will be projected onto one of the following vectors:

$$|\mathrm{Bell}_1\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}} \quad |\mathrm{Bell}_Z\rangle = \frac{|00\rangle - |11\rangle}{\sqrt{2}}$$
$$|\mathrm{Bell}_X\rangle = \frac{|01\rangle + |10\rangle}{\sqrt{2}} \quad |\mathrm{Bell}_Y\rangle = \frac{|01\rangle - |10\rangle}{\sqrt{2}}$$

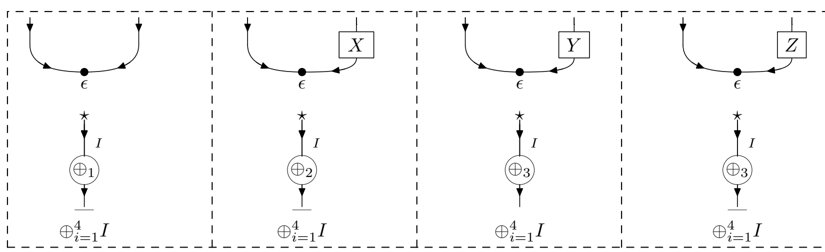
These states are all entangled, and further, each of them can be produced by starting with $|Bell_1\rangle$ (aka η) and applying one of the Pauli operators; hence we can associate a Pauli operator to each outcome of the measurement. In order to complete the protocol, Alice transmits a classical message to Bob, saying which of the four outcomes she observed. Bob then applies the corresponding Pauli operation to his qubit and—as if by magic—it is now in the state that Alice wished to transmit.

We now show how this protocol can be represented in term of proof-nets using the compact closed structure and biproducts. By normalising the proof-net we will effectively simulate the execution of the protocol.

We start with a *premise* representing Alice's input, and a unit link, representing the initial shared Bell state

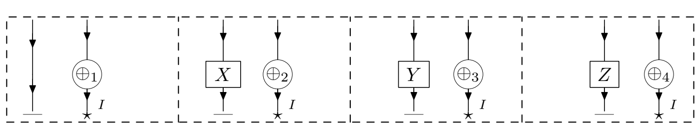


Note that in these diagrams time flows from the top to the bottom: input at the top of the page, outputs at the bottom. The right two qubits are taken to belong to Alice, the leftmost belonging to Bob. The next element is the Bell basis measurement. We will assume this is a destructive measurement, so the four possible transformations are simply projections. To indicate that these form an exclusive choice, we put them in a box, as shown below.

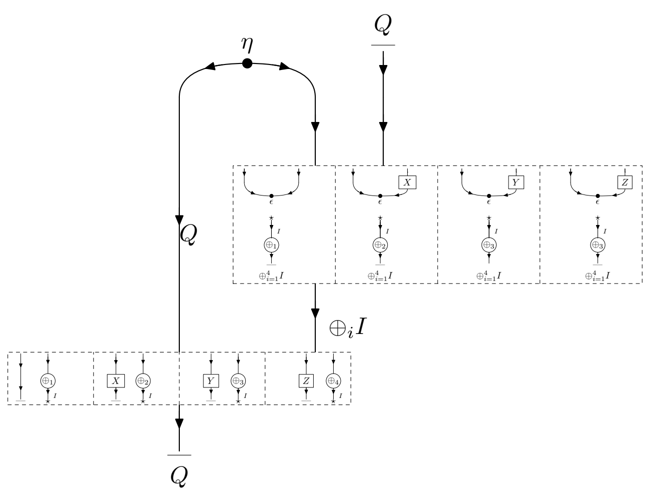


Notice the output of type $\oplus_i I$ serves simply to indicate which outcome occurred

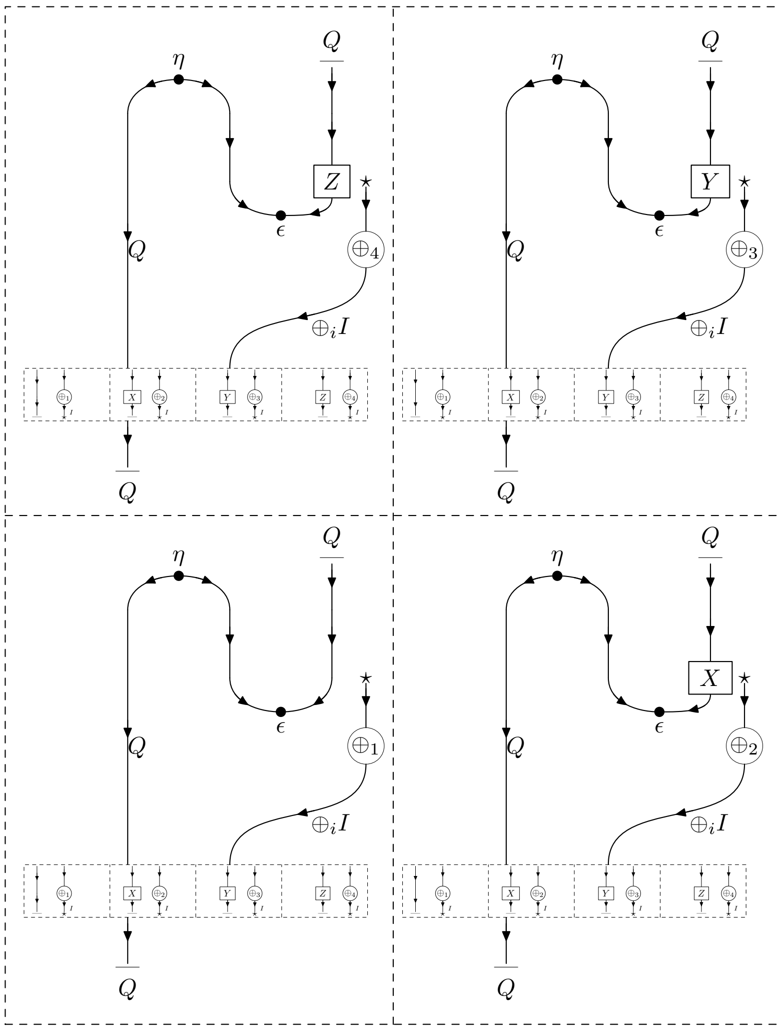
Finally we consider Bob's correction. Since his behaviour is conditional on a classical input, he has a box with an input of type $\oplus_i I$ as shown below.



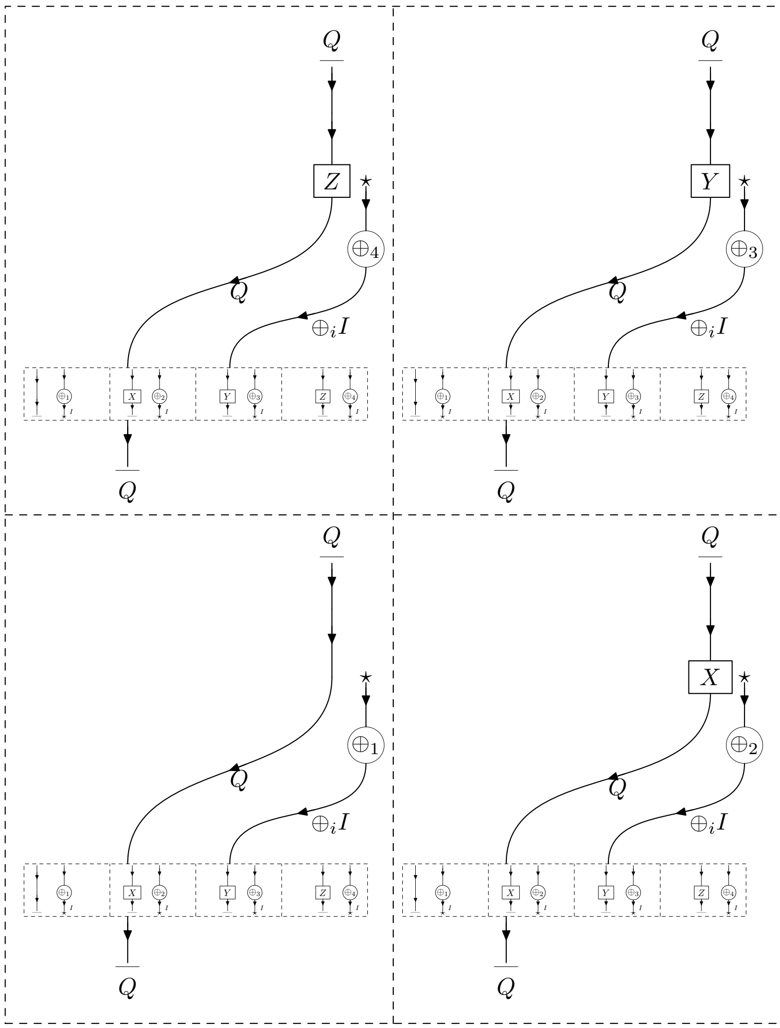
Putting it all together we have the following picture:



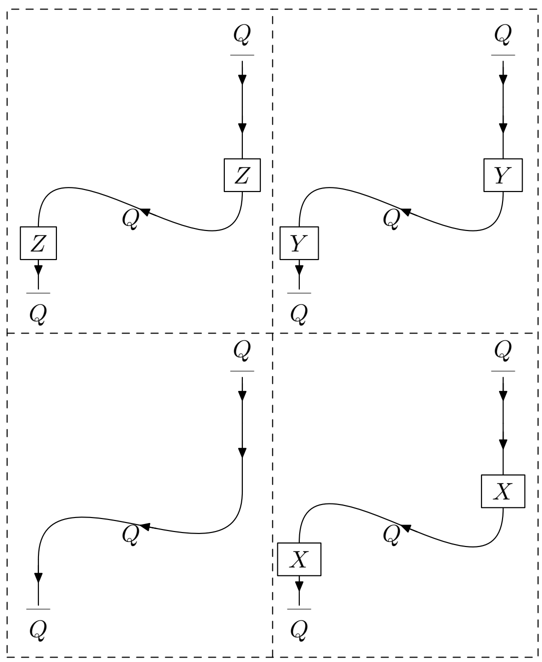
Now we can begin to simulate the protocol. The first step is to resolve the non-determinism of Alice's measurement. We do this by "opening the box", essentially making four copies of the whole system, one for each possible outcome of the measurement.



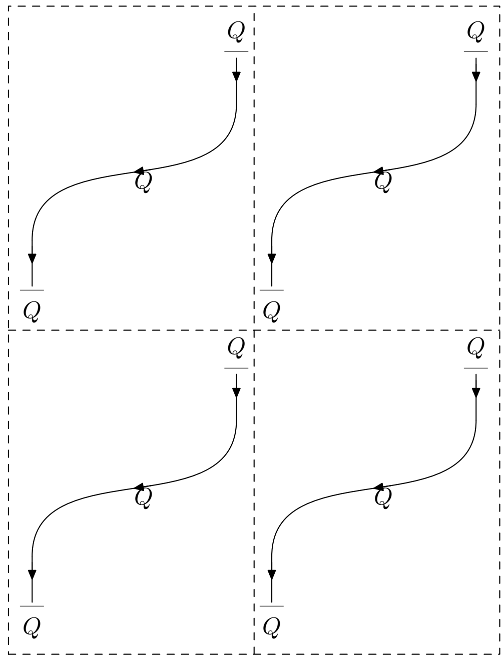
Next, in every copy the interaction of the entangled state and the measurement can be rewritten as shown.



Now we can open the box corresponding to Bob's non-determinism; this will leave us with sixteen copies of the system. We won't draw all of these copies, since twelve of them can be erased: theses are the cases the input that Bob is expecting does not match what Alice sends. We are left with:



Now we simply note (and this is not a *logical* axiom) that $X^2 = Y^2 = Z^2 = 1$ so we can simply remove these maps. Hence we have the normal form:



which show that in every possible world, Alice has successfully formed a channel to Bob along which her state can be transmitted.

Although we presented the rewrites in the order that the steps of the protocol would be carried out, in fact our proof-nets are strongly normalising, so any order would produce the same results.

Having sketched the system, we move to the details. First the category theory, then in Sections 1.3 and 1.4 the logic.

1.2 Categorical Preliminaries

In this section, we introduce the necessary categorical structures, compact closed categories and biproducts, and present their basic properties. Much of this material is well known so proofs are omitted. Standard references are Mac Lane [ML97] and Kelly-Laplaza [KL80]; other material is derived from the author's thesis [Dun06]. Other sources are cited as needed.

Compact closed categories [KL80] are abundant throughout mathematics and computer science. Examples include **Rel**, the category of sets and relations, finitely-generated projective modules over a commutative ring and Conway games (as categorified in [Joy77]). In **Hilb**, the category of all Hilbert spaces, the sub-category of determined by the Hilbert-Schmitt maps is compact closed, and more generally, the nuclear maps of any tensored *-category [ABP99] for a compact closed sub-category. Of course, **fdHilb**, the category of finite dimensional Hilbert spaces is compact closed.

In computer science compact closed categories have been studied in the context of typed concurrency as interaction categories [AGN96]; in logic, compact closed categories are degenerate models of multiplicative linear logic [AJ94, Loa94, HS03]; in physics the category of n-dimensional cobordisms, used in topological quantum field, theory is compact closed [BD95]. More examples are easy to find.

We define *biproducts* and investigate their basic properties in relation to compact closed categories. Categories with biproducts have been studied since the earliest days of category theory as part of the theory of Abelian categories [Mit65, ML97]. Compact closed categories with biproducts have been studied by Soloviev [Sol87]. A special case of compact closed categories with biproducts is *Tannakien category* [Del91]. A recent contribution is by Houston [Hou08], who proved that every compact closed category with products has biproducts.

1.2.1 Monoidal Categories

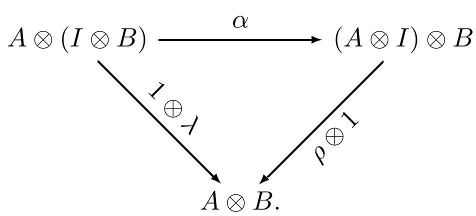
Definition 1 *A category $\mathcal{C}$ is* monoidal *if equipped with a functor $\otimes : \mathcal{C} \times \mathcal{C} \to \mathcal{C}$, a distinguished neutral object I, and natural isomorphisms*

$$\alpha_{A,B,C} : A \otimes (B \otimes C) \xrightarrow{\cong} (A \otimes B) \otimes C,$$
$$\lambda_A : I \otimes A \xrightarrow{\cong} A, \qquad \rho_A : A \otimes I \xrightarrow{\cong} A.$$

For the associativity morphism α we require that the pentagon

$$\begin{array}{ccccc} A \otimes (B \otimes (C \otimes D)) & \xrightarrow{\alpha} & (A \otimes B) \otimes (C \otimes D) & \xrightarrow{\alpha} & ((A \otimes B) \otimes C) \otimes D \\ \downarrow{\scriptstyle 1 \otimes \alpha} & & & & \uparrow{\scriptstyle \alpha \otimes 1} \\ A \otimes ((B \otimes C) \otimes D) & & \xrightarrow[\alpha]{\hspace{6em}} & & (A \otimes (B \otimes C)) \otimes D \end{array}$$

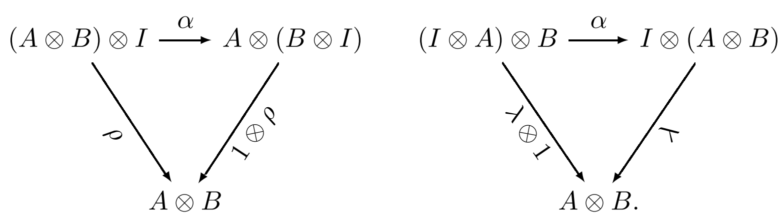
commutes. The isomorphisms λ and ρ express the neutrality of I; we *require that the following diagram commutes:*



Proposition 2 *In a monoidal category the equality*

$$\lambda_I = \rho_I$$

holds and the following diagrams commute:

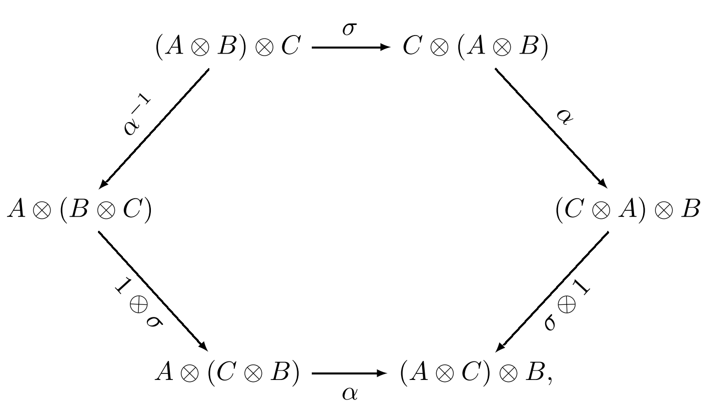


Proof See [JS93]. □

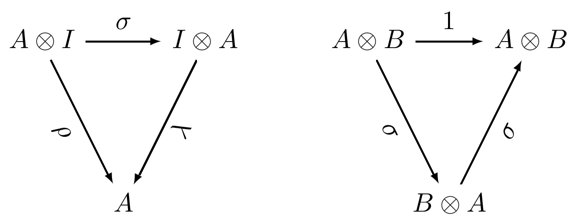
Definition 3 *A monoidal category is* symmetric *if it has a natural isomorphism*

$$\sigma_{A,B} : A \otimes B \to B \otimes A$$

such that



and



commute.

Mac Lane's celebrated coherence theorem states that any formal diagram constructed from the α, ρ, λ and σ will commute. A monoidal category is called *strict* if the isomorphisms α, λ,and ρ are all identities. To minimise syntactic overhead we will make use of the following theorem through this section:

Theorem 4 (Mac Lane) *Every monoidal category $\mathcal{C}$ is equivalent to some strict monoidal category $\mathcal{A}$.*

Note, however, that the category of proof-nets constructed in Section 1.4 is *not* strict: we will produce non-trivial associativity and unit morphisms.

1.2.2 Compact Closed Categories

Let $\mathcal{C}$ be a symmetric monoidal category. We say that $\mathcal{C}$ is *compact closed* if every object A has a chosen *dual*† A^* and maps

$$\eta_A : I \to A^* \otimes A$$
$$\epsilon_A : A \otimes A^* \to I$$

such that the composites

$$A \xrightarrow{1_A \otimes \eta} A \otimes A^* \otimes A \xrightarrow{\epsilon \otimes 1_A} A$$

and

$$A^* \xrightarrow{\eta \otimes 1_{A^*}} A^* \otimes A \otimes A^* \xrightarrow{1_{A^*} \otimes \epsilon} A^*$$

are equal to 1_A and 1_{A^*} respectively. We call η_A and ϵ_A the *unit* and *counit* maps.

† Some writers call this the "adjoint" in light of the relation between A and A^*; we use "dual" here to avoid confusion with the linear algebraic use of the word adjoint.

Proposition 5 *In a compact closed category we have natural isomorphisms:*

$$\begin{aligned} u : (A \otimes B)^* &\cong B^* \otimes A^* \\ v : I^* &\cong I \\ w : A^{**} &\cong A, \end{aligned}$$

A compact closed category which, in addition to being strictly monoidal, has all of the isomorphisms u, v, w equal to the identity is called a *strict compact closed category*. Kelly and Laplaza [KL80] show that any compact closed category is equivalent to a strict one, hence we will take the isomorphisms above to be equalities whenever convenient.

Proposition 6 *In a compact closed category the units and counits define dinatural transformations (see [GSS91])*

$$\eta : I \Rightarrow ((-)^* \otimes -)$$
$$\epsilon : (- \otimes (-)^*) \Rightarrow I.$$

We have a bijection between $\mathcal{C}(A, B)$ and $\mathcal{C}(B^*, A^*)$: given $f : A \to B$, define $f^* : B^* \to A^*$ by

$$\begin{array}{ccc} B^* & \xrightarrow{\eta_A \otimes 1_{B^*}} & A^* \otimes A \otimes B^* \\ {\scriptstyle f^*}\downarrow & & \downarrow{\scriptstyle 1_{A^*} \otimes f \otimes 1_{B^*}} \\ A^* & \xleftarrow[1_{A^*} \otimes \epsilon_B]{} & A^* \otimes B \otimes B^*. \end{array}$$

We call f^* the *dual* of f.

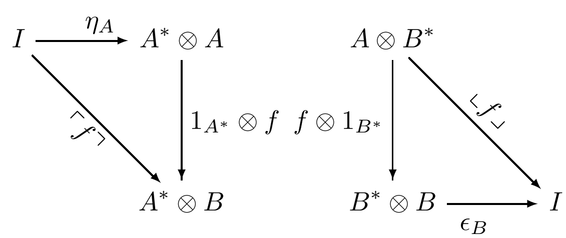
Proposition 7 *The operation $(-)^*$ defines a functor $\mathcal{C}^{op} \to \mathcal{C}$, which is an equivalence of categories.*

Proof *We have $1_A^* = 1_{A^*}$ immediately from the definition of dual, and $(f \circ g)^* = g^* \circ f^*$ follows from a routine calculation. Taking $\mathcal{C}$ to be strict, we we have $A^{**} = A$, it follow from the defining property of compact closure that $f^{**} = f$, which gives the equivalence.* □

Since we have the equivalence between $\mathcal{C}$ and $\mathcal{C}^{\mathrm{op}}$, any statement about some arrow applies equally well to its dual. In particular, results concerning units translate directly into results about counits and vice-versa.

The duality of a compact closed category gives a particularly strong form of monoidal closure. Every arrow in the category has a *point* which represents it, and dually a *copoint*. These representatives, the names and conames, will be crucial to our treatment of entangled quantum states.

Definition 8 *Let $f : A \to B$ in a compact closed category $\mathcal{C}$. Define the* name *and* coname *of f to be the maps $\ulcorner f \urcorner : I \to A^* \otimes B$ and $\llcorner f \lrcorner : A \otimes B^* \to I$ which are defined by the diagrams below.*



An immediate consequence of this definition is the isomorphism of homsets

$$\mathcal{C}(I, A^* \otimes B) \cong \mathcal{C}(A, B) \cong \mathcal{C}(A \otimes B^*, I).$$

Lemma 9 *Let $\mathcal{C}$ be compact closed and suppose we have arrows*

$$D \xrightarrow{h} A \xrightarrow{f} B \xrightarrow{g} C;$$

then the following equations hold:

$$(1_{A^*} \otimes g) \circ \ulcorner f \urcorner = \ulcorner g \circ f \urcorner \,,$$
$$(h^* \otimes 1_B) \circ \ulcorner f \urcorner = \ulcorner f \circ h \urcorner \,,$$

and

$$(\llcorner f \lrcorner \otimes 1_C) \circ (1_A \otimes \ulcorner g \urcorner) = g \circ f \ .$$

We can also define partial versions of the name and coname; essentially currying and uncurrying.

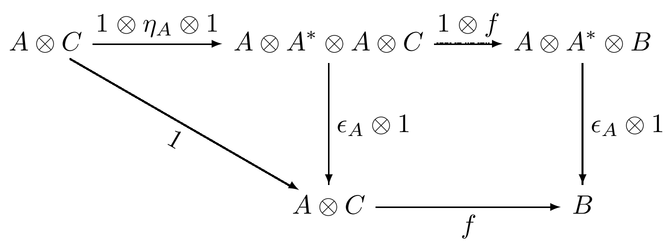
Lemma 10 (Partial Names and Conames) *In any compact closed category we have the following isomorphisms:*

$$\mathcal{C}(A \otimes C, B) \quad \cong \quad \mathcal{C}(A, C^* \otimes B) \tag{1.1}$$
$$\mathcal{C}(A, C \otimes B) \quad \cong \quad \mathcal{C}(A \otimes C^*, B) \tag{1.2}$$

Proof *Since the two isomorphisms are dual, we prove only the first. Define* $F : \mathcal{C}(A \otimes C, B) \to \mathcal{C}(A, C^* \otimes B)$ *and* $G : \mathcal{C}(A, C^* \otimes B) \to \mathcal{C}(A \otimes C, B)$ *by*

$$F : f \mapsto (1 \otimes f) \circ (\eta_A \otimes 1)$$
$$G : g \mapsto (\epsilon_A \otimes 1) \circ (1 \otimes g)$$

Their composition gives $GFf = (\epsilon_A \otimes 1) \circ (1 \otimes f) \circ (1 \otimes \eta_A 1)$ *from which*



and hence $GF = Id$*. Similarly* $Id = FG$*, which establishes the isomorphism.* □

Equation (1.1) essentially states that compact closed categories are indeed closed with $B^A = A^* \otimes B$. Since $A^* \otimes I \cong A^*$ this gives immediately the following.

Corollary 11 *Compact closed categories are* $*$*-autonomous [Bar79].*

Hence compact closed categories are models of **MLL**, and in particular the linear λ-calculus; albeit, these are rather strange models equipped with only one, self-dual, tensor.

Definition 12 *Let* $\mathcal{C}$ *be compact closed and define a map*

$$\mathrm{Tr} : \mathcal{C}(A \otimes C, B \otimes C) \to \mathcal{C}(A, B)$$

by setting

$$\mathrm{Tr}^C_{A,B}(f) = (1_B \otimes \epsilon_C) \circ (f \otimes 1_{C^*}) \circ (1_A \otimes \eta_{C^*})$$

The map $\mathrm{Tr}(f)$ *is called the* trace of f.

The trace so defined makes $\mathcal{C}$ into a *traced monoidal category* in the sense of Joyal, Street, and Verity [JSV96]. Few of the properties of the trace are required here so we will not recapitulate the definition—in any case the relevant facts can be deduced from the properties of η and ϵ. We will, however, need the following lemma:

Lemma 13 ([AHS02]) *Suppose we have arrows* $A \xrightarrow{f} B \xrightarrow{g} C$ *in a symmetric traced monoidal category; then:*

$$g \circ f = Tr^B_{A,C}(\sigma_{B,C} \circ (f \otimes g)) \ .$$

The *partial* trace defined above may be extended to a full trace over any endomorphism $f : A \to A$ by setting

$$\mathrm{Tr}(f) = \mathrm{Tr}^A_{I,I}(\rho \circ f \circ \rho^{-1}).$$

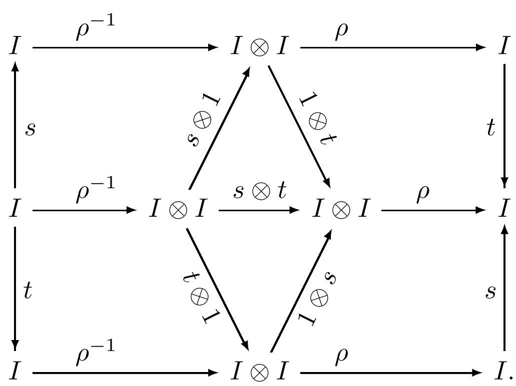
In **fdHilb**, the category of finite dimensional Hilbert spaces, this coincides with the usual trace, explicitly given by summing the diagonal elements of a matrix representation of f.

1.2.3 Scalars and Loops

Definition 14 *In any monoidal category* $\mathcal{C}$ *the endomorphisms of the neutral element* $\mathcal{C}(I, I)$ *are called the* scalars.

Lemma 15 *The scalars form a commutative monoid with respect to composition.*

Proof *Let* $s, t \in \mathcal{C}(I, I)$*; then*



□

Corollary 16 *For scalars* s, t *the composite*

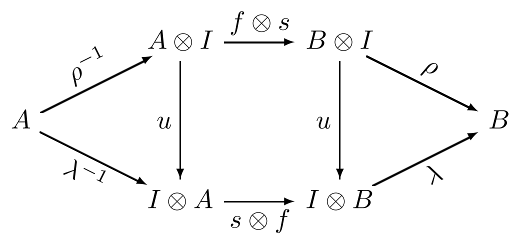
$$I \cong I \otimes I \xrightarrow{s \otimes t} I \otimes I \cong I$$

is equal to $s \circ t = t \circ s$.

Definition 17 *Let $\mathcal{C}$ be a monoidal category. Given a scalar s and some arrow $f : A \to B$ define a scalar multiplication $s \bullet f$ by the composition:*

$$A \xrightarrow{\rho^{-1}} A \otimes I \xrightarrow{f \otimes s} B \otimes I \xrightarrow{\rho} B.$$

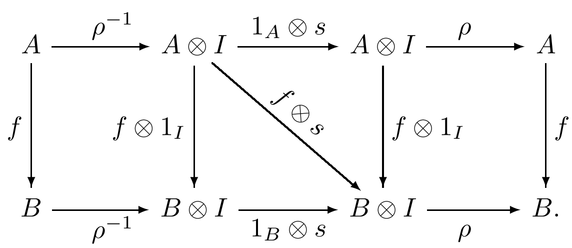
We could have defined $s \bullet f$ equivalently by multiplication on the left rather than on the right as above. Note that $u := \lambda^{-1} \circ \rho$ is a natural isomorphism $(- \otimes I) \Rightarrow (I \otimes -)$, so the following diagram commutes



and hence the two definitions coincide.

Lemma 18 *Each scalar s determines a natural transformation $Id \Rightarrow Id$ such that $s \bullet f = f \circ s_A = s_B \circ f$.*

Proof *The top and bottom edges define s_A and s_B respectively:*



The outer squares commute due to naturality of ρ, and the middle due to the functoriality of the tensor. Hence s defines a natural transformation. Note that the middle path from A to B is the definition of $s \bullet f$. $\square$

Corollary 19 *The following are immediate.*

(i) $s \bullet (t \bullet f) = (s \circ t) \bullet f$

(ii) $(s \bullet f) \circ (t \bullet g) = (s \circ t) \bullet (f \circ g)$

(iii) $(s \bullet f) \otimes (t \bullet g) = (s \circ t) \bullet (f \otimes g)$

Definition 20 *In a compact closed category* $\mathcal{C}$ *define the* dimension *of an object* A*, to be the following composite:*

$$dim_A = I \xrightarrow{\eta_A} A^* \otimes A \xrightarrow{\sigma} A \otimes A^* \xrightarrow{\epsilon_A} I.$$

Of course, this is nothing more than the trace of 1_A. The presence of these non-trivial scalars gives a qualitative aspect even to freely constructed compact closed categories.

1.2.4 Freely Constructed Compact Closed Categories

Define the set of *endomorphisms* $E(\mathcal{A})$ by the disjoint union

$$E(\mathcal{A}) = \sum_{A \in |\mathcal{A}|} \mathcal{A}(A, A),$$

and let the set of *loops* $[\mathcal{A}]$ be the quotient of $E(\mathcal{A})$ generated by the relation $f \circ g \sim g \circ f$ whenever $A \xrightarrow{f} B \xrightarrow{g} A$. Let $\tau : E(\mathcal{A}) \to [\mathcal{A}]$ be the canonical map onto the loops, and for each endomorphism f write $[f]$ for its image under τ.

The key theorem is the following of [KL80].

Theorem 21 *Let*

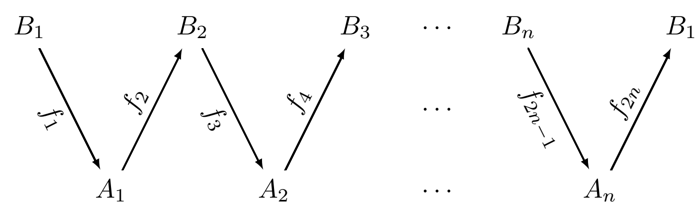
$$T : \mathcal{A}^{op} \times \mathcal{A} \times \mathcal{A}^{op} \times \mathcal{A} \times \cdots \times \mathcal{A}^{op} \times \mathcal{A} \longrightarrow \mathcal{B}$$

be a functor of $2n$ *variables, let* K *and* L *be objects of* $\mathcal{B}$ *and let* $\alpha : K \Rightarrow T$ *and* $\beta : T \Rightarrow L$ *be natural transformations with typical components*

$$\alpha : K \longrightarrow T(A_1, A_1, A_2, A_2, A_3, \ldots, A_{n-1}, A_n, A_n)\ , \tag{1.3}$$

$$\beta : T(B_1, B_2, B_2, B_3, \ldots, B_{n-1}, B_n, B_n, B_1) \longrightarrow L\ ; \tag{1.4}$$

given maps



the composite of (1.3), $T(f_1, f_2, f_3, \ldots, f_{2n-1}, f_n)$ *and (1.4), depends only on* $[f_{2n} f_{2n-1} \cdots f_2 f_1]$ *so that* α *and* β *give rise to a function* $[\mathcal{A}] \to \mathcal{B}(K, L)$*.*

Taking $\mathcal{A} = \mathcal{B}$, $K = L = I$, $\alpha = \eta$ and $\beta = \epsilon$ gives a ready source of scalars in any compact closed category $\mathcal{A}$; indeed this is the dimension map given in Definition 20. If the category is freely constructed these are the *only* non-trivial scalars. This is a consequence of the more general coherence theorem of Kelly and Laplaza. Before stating the theorem we must introduce some additional terminology, which will be also be required later in this section.

Definition 22 *A* signed set *S is a function from a carrier set $|S|$ to the set $\{+, -\}$. Given signed sets R and S, let R^* denote the signed set with the opposite signing to R; let $R \otimes S$ be the disjoint union of R and S, such that $|R \otimes S| = |R| + |S|$.*

Definition 23 *An* involution *is a category which is a coproduct of copies of the category* **2**. *Given an involution σ, its object set $|\sigma|$ can form a signed set by assigning $-$ to the source and $+$ to the target of each arrow of* **2**. *Call σ an* involution on the signed set S *when this signing agrees with that of S.*

Given some category $\mathcal{A}$, we can construct the free compact closed category generated by $\mathcal{A}$, which we call $F\mathcal{A}$. The objects of the $F\mathcal{A}$ are constructed from those of $\mathcal{A}$ by repeated application of the functors $- \otimes -$, $(-)^*$ and the constant I. This characterisation may be used to inductively construct a signed set $S(X)$ corresponding to each object X of $F\mathcal{A}$. Let

$$\begin{aligned} S(I) &= \emptyset \ , \\ S(X \otimes Y) &= S(X) \otimes S(Y) \ , \\ S(X^*) &= S(X)^* \ , \\ S(A) &= \{A \mapsto +\} \qquad \text{if } A \text{ is an object of } \mathcal{A} \ . \end{aligned}$$

The basic structure of arrows in $F\mathcal{A}$ depends upon involutions on the signed sets generated by its objects.

Theorem 24 (Kelly-Laplaza) *Let $\mathcal{A}$ be a category; each arrow $f : A \to B$ of the free compact closed category generated by $\mathcal{A}$ is completely described by the following data:*

(i) *An involution σ on $S(A^* \otimes B)$;*

(ii) *A functor $\theta : \sigma \to \mathcal{A}$ agreeing with σ on objects (i.e. a labelling of σ with arrows of $\mathcal{A}$);*

(iii) *A multiset L of loops from $\mathcal{A}$.*

The baroque statement of this theorem conceals its graphical content. One can view the objects of $F\mathcal{A}$ as lists of positively and negatively occurring objects of $\mathcal{A}$, and an arrow between two such lists is simply a collection of arcs, each connecting a negative occurrence to a positive one, and labelled by an arrow of $\mathcal{A}$. To compose arrows in $F\mathcal{A}$ we simply connect up the arcs, using the underlying composition in $\mathcal{A}$.

From this point of view we can see an immediate limitation in the use of such freely generated compact closed categories to model quantum states. Recall that when we interpret processes in categorical terms, we view the objects as state spaces; hence the objects of the generating category $\mathcal{A}$ are the state spaces of the elementary subsystems from which our composite systems will be built. A state of a compound system, that is, an arrow $\psi : I \to X$ in $F\mathcal{A}$, is thus composed of *pairs* of elementary systems related by some arrow from $\mathcal{A}$, and each pair is unconnected to the others. Quantum informatics attaches great importance to *entangled states*; that is, states which cannot be broken down into their constituent parts. However the above result states that free compact closed categories can only result in bipartite entanglement, which does not suffice to describe all entangled states. To extend our reach we now introduce *polycategories.*

1.2.5 Compact Symmetric Polycategories

The reason that the free construction described above yields only bipartite states is simple. The states are based on the arrows of the underlying category $\mathcal{A}$, and an arrow has exactly two ends. In order to represent multipartite states we will need generators with more than one and input and output, suggesting the need to construct the compact closed category from a category which already has a monoidal structure. However the direct route leaves open the problem of ensuring that the downstairs tensor (from the monoidal category $\mathcal{A}$) and the upstairs tensors (freely generated in $F\mathcal{A}$) cohere correctly. Worse, there is no reason to believe that an arrow in a monoidal category is in any sense indecomposable among its subsystems.

Fortunately there is a natural generalisation of category, a *polycategory*, whose arrows may have more than one object in their domain and codomain. The original notion of polycategory [Lam68, Sza75] was introduced to study classical logic, where a sequent may have multiple premises and conclusions; composition is defined by the cut-rule, so one output is connected to one input. Here we consider *compact symmetric* polycategories [Dun06], where composition is defined by the *multi-cut* rule, allowing arbitrary vectors of inputs and outputs to be composed.

Definition 25 *A* compact symmetric polycategory*, $\mathcal{P}$, consists of a class of objects* $\mathrm{Obj}_{\mathcal{P}}$ *and, to each pair* (Γ, Δ) *of finite sequences over* $\mathrm{Obj}_{\mathcal{P}}$*, a set of polyarrows* $\mathcal{P}(\Gamma, \Delta)$*. Given a non-empty sequence of objects* Θ *and poly-arrows*

$$\Gamma \xrightarrow{f} \Delta_1, \Theta, \Delta_2 \qquad \text{and} \qquad \Gamma_1, \Theta, \Gamma_2 \xrightarrow{g} \Delta$$

we may form the composition

$$\Gamma_1, \Gamma, \Gamma_2 \xrightarrow{g \, {}_i\overset{k}{\circ}_j \, f} \Delta_1, \Delta, \Delta_2$$

where $|\Delta_1| = i$*,* $|\Gamma_1| = j$ *and* $|\Theta| = k > 0$*. For each object* A *there is an identity arrow* $1_A : \langle A \rangle \to \langle A \rangle$ *for the singleton sequence* $\langle A \rangle$*.*

In general there are many ways to compose the polyarrows, and many equations which must be satisfied. We will spare the reader the full definition†, and instead we will offer a theorem in the spirit of the Kelly-Laplaza result cited above, characterising the free compact closed category generated by such a polycategory. Before proceding we note the most important point about these polycategories: there is no nullary composition and no tensor product. Each input of a polyarrow has a path (not necessarily directed) to each output, and hence despite having many inputs and outputs, polyarrows cannot be decomposed into non-interacting parts.

Before we can state the representation theorem we must make some definitions.

Definition 26 *A* graph *consists of a 5-tuple* (V, E, C, s, t) *where* V*,*E*, and* C *are sets, respectively of* vertices*,* edges*, and* circles*, and* s *and* t *are maps*

$$E \underset{t}{\overset{s}{\rightrightarrows}} V$$

which we call source *and* target*. Let* in(v) *and* out(v) *be* V*-indexed*

† For the full glory of its coherence equations, and also the proof of the theorem cited, see [Dun06].

subsets of E defined by

$$in(v) = t^{-1}(v)$$
$$out(v) = s^{-1}(v).$$

The in-degree *of a vertex v is the cardinality of $in(v)$ and the* out-degree *is the cardinality of $out(v)$. The* degree *of a vertex is the sum of its in- and out-degrees.*

Definition 27 *A* open graph *is a pair $(G, \partial G)$ of an underlying graph $G = (V, E, C, s, t)$ and a distinguished subset of the degree one vertices ∂G called the* boundary *of G; $V - \partial G$ is called the* interior *of G, written I_G. If a vertex $x \in \partial G$ it is an* outer *or* boundary node*; otherwise it is an* inner *or* interior node*.*

Definition 28 *A* circuit $\Gamma = (G, \operatorname{dom}\Gamma, \operatorname{cod}\Gamma, <_{in(\cdot)}, <_{out(\cdot)})$ *where:*

- $G = ((V, E, C, s, t), \partial G)$ *is an open graph;*
- $\operatorname{dom}\Gamma$ *and* $\operatorname{cod}\Gamma$ *are totally ordered sets such that* $\partial G = \operatorname{dom}\Gamma + \operatorname{cod}\Gamma$*;*
- $<_{in(\cdot)}$ *is a family of maps, indexed by V such that*

$$<_{in(v)} \colon in(v) \xrightarrow{\cong} \mathbb{N}_k$$

  where $k = |in(v)|$.
- $<_{out(\cdot)}$ *is a family of maps, indexed by V such that*

$$<_{out(v)} \colon out(v) \xrightarrow{\cong} \mathbb{N}_{k'}$$

  where $k' = |out(v)|$.

As suggested by their name, the purpose of the two maps $<_{\mathrm{in}(\cdot)}$ and $<_{\mathrm{out}(\cdot)}$ is to impose a linear order on $\mathrm{in}(v)$ and $\mathrm{out}(v)$. Since the maps give a bijective correspondence between $\mathrm{in}(v), \mathrm{out}(v)$ and an initial segment of the naturals, the order in $\mathbb{N}$ lifts, and hence we will often simply treat these sets are ordered, and write $<$ for this ordering whenever unambiguous to do so.

For simplicity, in the following we will consider a polycategory $\mathcal{P}$ which is freely generated from some set of basic arrows† called $\mathrm{Arr}_{\mathcal{P}}$.

† This is not essential, but will greatly simplify the subsequent discussion of generalised proof-nets; see the discussion of homotopy in [Dun06] for the details.

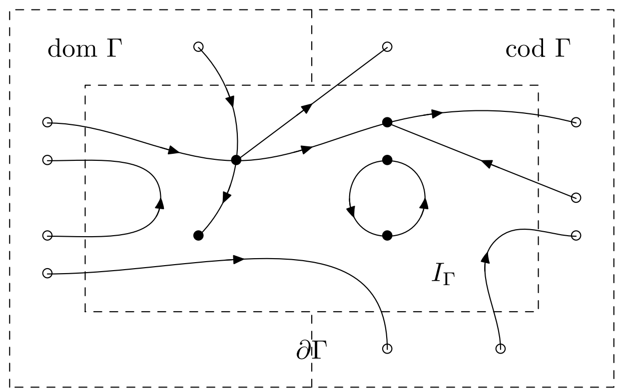


Fig. 1.1. Anatomy of a circuit

Definition 29 *Given a polycategory $\mathcal{P}$, an $\mathcal{P}$-*labelling *for a circuit* Γ *is a pair of maps $\theta = (\theta_O, \theta_A)$ where*

$$\theta_O : E{+}C \longrightarrow \mathrm{Obj}_{\mathcal{P}}$$
$$\theta_A : V \longrightarrow \mathrm{Arr}_{\mathcal{P}}$$

such that for each node f, in$(f) = \langle a_1, \ldots, a_n\rangle$ and out$(f) = \langle b_1, \ldots, b_m\rangle$ imply

$$\mathrm{dom}(\theta f) = \theta a_1, \ldots, \theta a_n$$
$$\mathrm{cod}(\theta f) = \theta b_1, \ldots, \theta b_m,$$

*and subject to the further restriction that $\theta_A(v) = 1_A$ if and only if $v \in \partial\Gamma$. Call a circuit Γ $\mathcal{P}$-*labellable *if there exists an $\mathcal{P}$-labelling for it; if θ is a labelling for Γ, then the pair (Γ, θ) is an $\mathcal{P}$-*labelled circuit*.*

The boundary nodes perform a different role to the interior nodes. The incidence of the unique edge at a boundary vertex b defines a signing on the boundary: we say that b is positive if it has in-degree 1; and negative if its out-degree is 1. We will take the boundary vertices as labelled by objects of $\mathcal{P}$ rather than the corresponding identity maps, and treat the boundary as an Obj_P-labelled signed set.

We denote the class of $\mathcal{P}$-labelled circuits $\mathbf{Circ}(\mathcal{P})$; it forms a monoidal category in a rather natural way. The objects of $\mathbf{Circ}(\mathcal{P})$ are signed sequences of objects from Obj_P. An arrow from $f : A \to B$ is defined by a $\mathcal{P}$-labelled circuit whose codomain is B and whose domain is A^* (i.e A with the opposite signing). Composition is defined by joining two circuits at their respective domain and codomain vertices, and erasing the vertices. The tensor product can be defined by taking the disjoint union of the circuits, and concatenating the domain and codomains. We leave the reader to fill in the details. We can now state the promised representation theorem:

Theorem 30 ***Circ***$(\mathcal{P})$ *is the free compact closed category generated by the compact symmetric polycategory* $\mathcal{P}$*.*

Since categories are a special case of polycategory (where all the arrows are between singleton sequences) we can ask: what is **Circ**$(\mathcal{P})$ when $\mathcal{P}$ is just a normal category? In this case, **Circ**$(\mathcal{P})$ is exactly the same as $F\mathcal{P}$ as per Kelly-Laplaza.

One way to understand the generalisation in going from a category of generators to a polycategory of generators is by considering the case with only one ground type. If we have a category with just one object, we can view its arrows as evolutions of this state space. On the other hand, if we have a polycategory with a single object, the arrows are in some sense *interactions* between systems of that type, possibly fusing or splitting, producing a different number of systems than began the interaction. By moving to the compact closed category generated by a polycategory of interactions we avoid the restriction to bipartite states mentioned earlier.

This concludes the multiplicative structures, now we move onto the additives.

1.2.6 Zero Objects

Definition 31 *In any category* $\mathcal{C}$ *a* zero object *is an object, denoted* $\mathbf{0}$*, which is both initial and terminal.*

By its initiality, there is a unique map from $\mathbf{0}$ to every object, and dually there is a unique map from each object to $\mathbf{0}$. Hence there is unique map

$$A \longrightarrow \mathbf{0} \longrightarrow B$$

between every pair of objects A and B. This map is called the *zero map* and denoted $0_{A,B}$. Since $\mathbf{0}$ is initial and terminal any map composed with a zero map is again a zero map; the zeros form a two-sided ideal with respect to composition among the arrows $\mathcal{C}$. Hence the following diagram commutes:

$$\begin{array}{ccccc} A & \longrightarrow & \mathbf{0} & \longrightarrow & B \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle 1_{\mathbf{0}}} & & \downarrow{\scriptstyle g} \\ C & \longrightarrow & \mathbf{0} & \longrightarrow & D, \end{array}$$

which makes the family $0_{A,B}$ natural in both A and B.

A useful family of arrows in a category with $\mathbf{0}$ is the Kronecker delta $\delta_{ij} : A_i \longrightarrow A_j$, defined for all pairs of objects A_i, A_j as

$$\delta_{ii} = 1_{A_i} \qquad \delta_{ij} = 0_{A_i,A_j}$$

Lemma 32 *If* $1_A = 0_{A,A}$ *then* $A \cong \mathbf{0}$.

Proof Note that the composite

$$\mathbf{0} \xrightarrow{\;!_A\;} A \xrightarrow{\;!^A\;} \mathbf{0}$$

is equal to $0_{\mathbf{0},\mathbf{0}}$, which by uniqueness is equal to $1_{\mathbf{0}}$. Thus $!_A \circ !^A = 1_A$ and $!^A \circ !_A = 1_{\mathbf{0}}$, which gives the isomorphism. □

Proposition 33 *Let* $\mathcal{C}$ *be a monoidal closed category with a zero object. Then* $A \otimes \mathbf{0} \cong \mathbf{0}$.

Proof Since $\mathcal{C}$ is closed,

$$\mathcal{C}(A \otimes \mathbf{0}, B) \;\cong\; \mathcal{C}(\mathbf{0}, A \multimap B) \;\cong\; \{*\}.$$

Taking $B = A \otimes \mathbf{0}$ implies $1_{A\otimes\mathbf{0}} = 0$, and hence the result follows. □

Monoidal closure is required; if we take the tensor to be a coproduct, e.g direct sum of vector spaces, it is clear that the isomorphism does not hold.

Corollary 34 *Given* $f : A \to B$ *an arrow of* $\mathcal{C}$, $f \otimes 0_{C,D} = 0_{A\otimes C, B\otimes D}$.

If the zero object is also the neutral object for the tensor, then the entire category collapses to a single object via $A \cong A \otimes \mathbf{0} \cong \mathbf{0}$. So any Cartesian closed category with zero is trivial. Note that in a compact closed category $\mathcal{C}$ with a terminal object $\mathbf{1}$, by duality $\mathbf{1}^*$ is initial. If the terminal object is the monoidal unit then the isomorphism $I \cong I^*$ makes I the zero object, and hence the category collapses.

Proposition 35 *If $\mathcal{C}$ is compact closed with respect to a product then it is trivial.*

1.2.7 Biproducts

In any category $\mathcal{C}$ with finite products and coproducts every map

$$\coprod A_i \xrightarrow{f} \prod A_j$$

has a "matrix" representation (f_{ij}) where each f_{ij} is given by the composite

$$A_i \xrightarrow{\mathrm{in}_i} \coprod A_i \xrightarrow{f} \prod A_j \xrightarrow{\pi_j} A_j$$

with in_i and π_j the appropriate injections and projections. Supposing that $\mathcal{C}$ also has a zero object there is canonical map $\nvDash : \coprod A_i \longrightarrow \prod A_i$ whose matrix is the identity $\nvDash = (\delta_{ij})$.

Definition 36 *A category $\mathcal{C}$ has finite biproducts if it has finite products and coproducts, such that*

- *the unique map $0 \longrightarrow 1$ is invertible; and*
- *the canonical map $\nvDash : A \coprod B \longrightarrow A \prod B$ is an isomorphism for all objects A, B.*

If $\mathcal{C}$ has biproducts, for all objects A and B, there is a unique (upto isomorphism) object $A \oplus B$ and maps

$$A \underset{\pi_1}{\overset{\mathrm{in}_1}{\rightleftarrows}} A \oplus B \underset{\pi_2}{\overset{\mathrm{in}_2}{\leftrightarrows}} B \tag{1.5}$$

such that $(A \oplus B, \pi_1, \pi_2)$ is a product and $(A \oplus B, \mathrm{in}_1, \mathrm{in}_2)$ is a coproduct. A choice of $A \oplus B$ for every pair of objects makes $- \oplus -$ into a functor $\mathcal{C} \times \mathcal{C} \longrightarrow \mathcal{C}$ whose action on arrows $f_1 \oplus f_2$ is given by

$$\pi_i \circ (f_1 \oplus f_2) = f_i \circ \pi_i \qquad \text{for } i = 1, 2$$

or alternatively

$$(f_1 \oplus f_2) \circ \mathrm{in}_i = \mathrm{in}_i \circ f_i \qquad \text{for } i = 1, 2.$$

Lemma 37 *In a category with biproducts we have the following natural isomorphisms:*

- $(A \oplus B) \oplus C \cong A \oplus (B \oplus C)$ *;*
- $A \oplus B \cong B \oplus A$ *;*

- $A \oplus \mathbf{0} \cong A \cong \mathbf{0} \oplus A$.

Proof All these isomorphisms hold for products (and also coproducts) hence they hold for the biproduct. □

We have natural diagonal and codiagonal maps,

$$\Delta_A : A \longrightarrow A \oplus A$$
$$\nabla_A : A \oplus A \longrightarrow A$$

defined as

$$\Delta_A = \langle 1_A, 1_A \rangle \qquad \nabla_A = [1_A, 1_A].$$

It is useful to note the equations

$$(f \oplus g) \circ \Delta_A = \langle f, g \rangle,$$
$$\nabla_A \circ (f \oplus g) = [f, g].$$

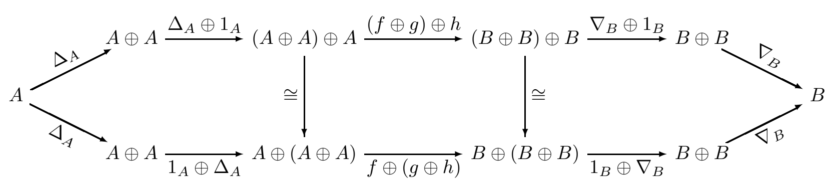
Definition 38 *Let $f, g : A \longrightarrow B$; then define $f + g$ as the composite*

$$A \xrightarrow{\Delta_A} A \oplus A \xrightarrow{f \oplus g} B \oplus B \xrightarrow{\nabla_B} B.$$

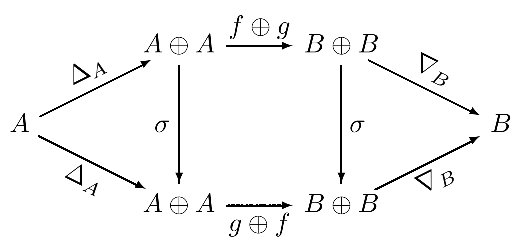
Proposition 39 *In a category with biproducts $\mathcal{C}$, the addition of Definition 38:*

- *makes each hom-set $\mathcal{C}(A, B)$ is a commutative monoid; and*
- *distributes over composition.*

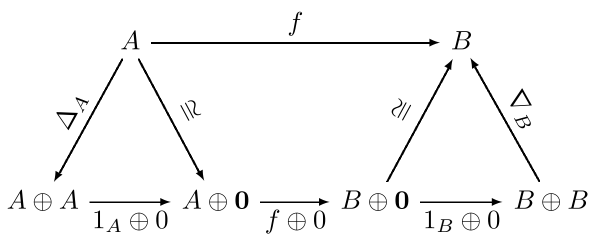
Proof The addition is associative due to the following diagram



and commutative since



commutes. The neutral element of $\mathcal{C}(A, B)$ is $0_{A,B}$ since the following commutes:



hence $\mathcal{C}$ is enriched over commutative monoids. To see that the addition distributes over composition recall the identities

$$\begin{aligned} \langle f, g\rangle \circ h &= \langle f \circ h, g \circ h\rangle \\ h \circ [f, g] &= [h \circ f, h \circ g] \end{aligned}$$

hence

$$\begin{aligned} g \circ (f + f') \circ h &= g \circ \nabla \circ (f \oplus f')\Delta \circ h \\ &= [g, g] \circ (f \oplus f') \circ \langle h, h\rangle \\ &= \nabla \circ ((g \circ f \circ h) \oplus (g \circ f' \circ h)) \circ \Delta \\ &= (g \circ f \circ h) + (g \circ f' \circ h). \end{aligned}$$

□

Proposition 40 *In a category with biproducts the injections and projections shown in Eq (1.5) satisfy*

$$\begin{aligned} \pi_i \circ \mathrm{in}_j &= \delta_{ij} \qquad \textit{for } i, j = 1, 2 \\ \mathrm{in}_1 \circ \pi_1 + \mathrm{in}_2 \circ \pi_2 &= 1_{A_1 \oplus A_2}. \end{aligned}$$

Proof For any product $A \times B$ we have $\pi_1 \times \pi_2 \circ \Delta = \langle \pi_1, \pi_2\rangle = 1_{A \times B}$ and dually for any coproduct $\nabla \circ \mathrm{in}_1 + \mathrm{in}_2 = 1_{A+B}$. Hence

$$\mathrm{in}_1 \circ \pi_1 + \mathrm{in}_2 \circ \pi_2 = \nabla \circ (\mathrm{in}_1 \oplus \mathrm{in}_2) \circ (\pi_1 \oplus \pi_2) \circ \Delta = 1_{A \oplus B}.$$

Due to the universal property of the biproduct, the canonical map from $A \oplus B$ to itself is equal to $1_{A \oplus B}$. Therefore

$$\pi_i \circ \mathrm{in}_j = \pi_i \circ \nvDash \circ \mathrm{in}_j = \delta_{ij}.$$

□

The binary biproduct may be generalised to arbitrary finite families of objects $A_1, \ldots, A_n$ by iteration. Upto an associativity isomorphism, the n-fold biproduct is characterised by the diagram

$$A_i \xrightarrow{\mathrm{in_i}} \bigoplus_k A_k \xrightarrow{\pi_j} A_j$$

subject to the equations

$$\pi_i \circ \mathrm{in}_j = \delta_{ij},$$
$$\sum_k \pi_k \circ in_k = 1_{\oplus_k A_k}.$$

Arrows between biproducts have matrix representations as described at the start of this section, and composition of of arrows gives the usual matrix multiplication.

Proposition 41 *Suppose we have arrows*

$$\bigoplus_i A_i \underset{f'}{\overset{f}{\rightrightarrows}} \bigoplus_j B_j \xrightarrow{g} \bigoplus_k C_k$$

then $(g \circ f)_{ik} = \sum_j (g_{jk} \circ f_{ij})$ *and* $(f + f')_{ij} = f_{ij} + f'_{ij}$.

Proof Let $h = g \circ f$; then

$$\begin{aligned} h_{ik} &= \pi_k \circ g \circ f \circ \mathrm{in}_i \\ &= \pi_k \circ g \circ 1_{\oplus B_j} \circ f \circ \mathrm{in}_i \\ &= \pi_k \circ g \circ (\sum_j \pi_j \circ \mathrm{in}_j) \circ f \circ \mathrm{in}_i \\ &= \sum_j \pi_k \circ g \circ \mathrm{in}_j \circ \pi_j \circ f \circ \mathrm{in}_i \\ &= \sum_j g_{jk} \circ f_{ij}. \end{aligned}$$

The second equation follows directly from the naturality of the diagonal and codiagonal maps. □

Proposition 39 implies that any category with biproducts is enriched over **CMon**, the category of commutative monoids; conversely, we have the following.

Proposition 42 *Let $\mathcal{C}$ be a **CMon**-category with a $\mathbf{0}$ object and, for every pair of objects A and B, a diagram (1.5) such that proposition 40 holds; then $\mathcal{C}$ has biproducts.*

Theorem 43 *A* **CMon**-*category has products (or coproducts) if and only if it has biproducts.*

Proof This is a fairly trivial modification of Mac Lane [ML97] VIII.2, Theorem 2. □

Definition 44 *Call a* **CMon**-*category* semi-additive *if it has* $\mathbf{0}$ *and a biproduct for each pair of its objects. Let $\mathcal{A}$ and $\mathcal{B}$ be* **CMon**-*categories with zero objects; a functor $F : \mathcal{A} \longrightarrow \mathcal{B}$ is* semi-additive *if $F\mathbf{0} = \mathbf{0}$ and $Ff + Fg = F(f + g)$ for all parallel arrows f, g in $\mathcal{A}$.*

Proposition 45 *Let $\mathcal{A}$ have biproducts and let $\mathcal{B}$ be* **CMon**-*enriched with* $\mathbf{0}$*; then a functor $F\mathcal{A} \to \mathcal{B}$ is semi-additive if and only if it carries every biproduct diagram in $\mathcal{A}$ to a biproduct diagram in $\mathcal{B}$.*

Proof See Mac Lane [ML97] VIII.2, Proposition 4. □

Given a category $\mathcal{C}$ we can construct the free biproduct structure on $\mathcal{C}$, first by freely enriching $\mathcal{C}$ over **CMon** and then taking matrices over the resulting category.

Proposition 46 *Let $\mathcal{C}_{\mathbb{N}}$ be the category whose objects are those of $\mathcal{C}$ and where $\mathcal{C}_{\mathbb{N}}(A, B) = \mathbb{N}(\mathcal{C}(A, B))$, the free commutative monoid on $\mathcal{C}(A, B)$. Then $\mathcal{C}_{\mathbb{N}}$ is* **CMon** *enriched and the inclusion of $\mathcal{C}$ into $\mathcal{C}_{\mathbb{N}}$ is a universal arrow from $\mathcal{C}$ to a* **CMon**-*category.*

Proposition 47 *Let $\mathcal{C}$ be a* **CMon**-*category and let* $\mathbf{Matr}(\mathcal{C})$ *be the category whose objects are n-tuples of objects of $\mathcal{C}$, for $n \geq 1$, and whose arrows are matrices of arrows $\mathcal{C}$. Then* $\mathbf{Matr}(\mathcal{C})$ *is semi-additive, and the evident semi-additive embedding of $\mathcal{C}$ into* $\mathbf{Matr}(\mathcal{C})$ *is universal among semi-additive functors from $\mathcal{C}$ to semi-additive categories.*

1.2.8 Compact Closed Categories with Biproducts

Proposition 48 *Let $\mathcal{C}$ be a monoidal closed category with biproducts; then there are natural distribution isomorphisms*

$$A \otimes (B \oplus C) \cong (A \otimes B) \oplus (A \otimes C)$$
$$(A \oplus B) \otimes C \cong (A \otimes C) \oplus (B \otimes C)$$

Proof Since $A \otimes -$ is a left adjoint it preserves colimits and hence the diagram

$$A \otimes B \xrightarrow{1_A \otimes \mathrm{in}_1} A \otimes (B \oplus C) \xleftarrow{1_A \otimes \mathrm{in}_2} A \otimes C$$

is a coproduct and hence $A \otimes (B \oplus C) \cong (A \otimes B) \oplus (A \otimes C)$. The right hand distribution is similar. □

Corollary 49 *In a monoidal closed category $\mathcal{C}$ with biproducts, the functor $A \otimes - : \mathcal{C} \to \mathcal{C}$ is additive.*

In fact we can easily construct the distribution isomorphisms explicitly. Let

$$d_{A,B,C} = \langle 1_A \otimes \pi_1, 1_A \otimes \pi_2 \rangle$$

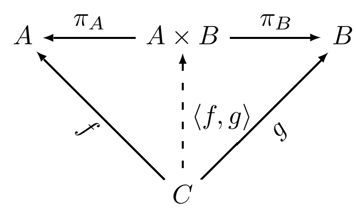
and

$$d^{-1}_{A,B,C} = [1_A \otimes \mathrm{in}_1, 1_A \otimes \mathrm{in}_2].$$

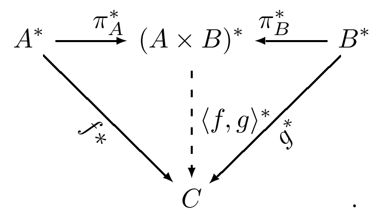
Then

$$\begin{aligned}
&[1_A \otimes \mathrm{in}_1, 1_A \otimes \mathrm{in}_2] \circ \langle 1_A \otimes \pi_1, 1_A \otimes \pi_2 \rangle \\
&\qquad = \nabla \circ ((1_A \otimes \mathrm{in}_1) \oplus (1_A \otimes \mathrm{in}_2)) \circ ((1_A \otimes \pi_1) \oplus (1_A \otimes \pi_C)) \circ \Delta \\
&\qquad = \nabla \circ (1_A \otimes (\mathrm{in}_1 \circ \pi_1)) \oplus (1_A \otimes (\mathrm{in}_2 \circ \pi_2)) \circ \Delta \\
&\qquad = (1_A \otimes (\mathrm{in}_1 \circ \pi_1)) + (1_A \otimes (\mathrm{in}_2 \circ \pi_2)) \\
&\qquad = 1_A \otimes (\mathrm{in}_1 \circ \pi_1 + \mathrm{in}_2 \circ \pi_2) \\
&\qquad = 1_A \otimes 1_{B \oplus C} = 1_{A \otimes (B \oplus C)}
\end{aligned}$$

If a compact closed category $\mathcal{C}$ has a binary product $- \times -$ then the duality $(\cdot)^*$ sends every product diagram



to a coproduct diagram



As mentioned earlier, if $\mathcal{C}$ has a terminal object $\mathbf{1}$ then $\mathbf{1}^*$ is initial. Hence the question of whether or not $\mathcal{C}$ has biproducts boils down to whether the canonical maps

$$\mathbf{0} \longrightarrow \mathbf{1}$$
$$(A \times B)^* \longrightarrow A^* \times B^*$$

are isomorphisms. It turns out that this is always the case.

Proposition 50 *(Houston) If a compact closed category $\mathcal{C}$ has all finite products (or coproducts) it has all finite biproducts.*

Proof See Houston [Hou08]. □

Corollary 51 *In any compact closed category with biproducts:*

- *we have natural isomorphisms*

$$\mathbf{0} \cong \mathbf{0}^* \qquad\qquad (A \oplus B)^* \cong A^* \oplus B^*;$$

- *the duality $(\cdot)^*$ is an additive functor.*

It then follows that we may choose the biproduct in any compact closed category so that the equation

$$\mathrm{in_A}^* = \pi_{A^*}$$

holds for all objects A.

We now turn our attention to the construction of the free compact closed category with biproducts upon some polycategory $\mathcal{P}$. The earlier results of Propositions 46 and 47 described how to freely construct the biproduct as matrices whose elements are drawn from some category $\mathcal{C}$, and this will provide the core of our proof.

Write $CB\mathcal{P}$ to denote the free compact closed category with biproducts generated by a compact polycategory $\mathcal{P}$. We refer to the objects of $\mathcal{P}$, their images under $(\cdot)^*$, and the constants $\mathbf{0}$ and I as the *literals* of $CB\mathcal{P}$. According to Corollary 51, in any compact closed category with biproducts, $(\cdot)^*$ commutes with $\oplus$, and since both the biproduct and tensor structures are freely generated, the objects of $CB\mathcal{P}$ are formed from the literals by repeated application of the functors $(- \otimes -)$ and $(- \oplus -)$†. Any object may therefore be described by such a functor and a vector of literals.

Let $\otimes_n : CB\mathcal{P} \times \cdots \times CB\mathcal{P} \to CB\mathcal{P}$ be the n-fold tensor; similarly let $\oplus_n$ be the n-fold biproduct. Call N a *normal* functor if it is has the form

$$N = \oplus_n(\otimes_{m_1}(-), \ldots, \otimes_{m_n}(-)).$$

Lemma 52 *Let G be a functor constructed from $(- \otimes -)$ and $(- \oplus -)$; then G is naturally isomorphic to a normal functor N_G*

Proof The required isomorphism is constructed from the distributivity isomorphisms. □

Hence we have that all arrows in $CB\mathcal{P}$ have the form

$$\begin{array}{ccc} F\overline{A} & \xrightarrow{f} & G\overline{B} \\ \downarrow \cong & & \downarrow \cong \\ N_F\overline{A} & \xrightarrow[f']{} & N_G\overline{B} \end{array}$$

and since f' is an arrow between normal functors, it has matrix elements

$$f_{ij} : \otimes_{m_i} A_i \to \otimes_{n_j} B_j$$

† See [Sol87] for a more general treatment of this.

each of which is a (possibly empty) sum of arrows from the freely constructed compact structure, $\mathbf{Circ}(\mathcal{P})$.

Hence the free compact closed category with biproducts is produced by forming $\mathbf{Matr}(\mathbf{Circ}(\mathcal{P})_{\mathbb{N}})$—that is, the free biproduct category on top of the free compact closed category—and simply adjoining the distributivity isomorphisms.

This free construction is the final piece of category theory needed in this article. In Section 1.4 we'll introduce a system of proof-nets that represent this category.

1.3 Tensor-Sum Logic

In this section we will introduce the syntax of tensor-sum logic in a sequent calculus **LTS**, and give it a semantics over a suitable category. Let $\mathcal{A}$ be a category and denote by $F\mathcal{A}$ the free compact closed category with biproducts generated by $\mathcal{A}$. The atomic formulae of **LTS** will be the objects of $\mathcal{A}$, and the arrows of $\mathcal{A}$ will give its non-logical axioms. In the next section we will generalise to the situation where the generators are a polycategory, but that requires a proof-net presentation. For now we stick to this simpler case, since the essence of the connectives can be seen equally well via a sequent presentation.

Definition 53 *The* formulae *of **LTS** are given by the following grammar:*

$$F ::= \mathbf{0} \mid I \mid A \mid A^* \mid F \otimes F \mid F \oplus F$$

where $A \in \mathrm{Obj}_{\mathcal{A}}$ are called atoms. *Given a formula F we define its* de Morgan dual F^* *by:*

$$\begin{aligned} \mathbf{0}^* &:= \mathbf{0} \\ I^* &:= I \\ A^{**} &:= A \\ (F_1 \otimes F_2)^* &:= F_2^* \otimes F_1^* \\ (F_1 \oplus F_2)^* &:= F_1^* \oplus F_2^* \ . \end{aligned}$$

*An **LTS** formula is called* multiplicative *if neither $\mathbf{0}$ nor $\oplus$ occur in it.*

We use the convention that letters A,B,C etc, range over the atoms, while X,Y,Z etc, range over arbitrary formulae. We take for granted that all formulae are in de Morgan normal form—that is, with the negation symbol $(\cdot)^*$ occurring only on atoms.

Definition 54 *A* sequent *of **LTS** has the form*

$$\Gamma \vdash \Delta; L$$

where Γ *and* Δ *are lists of formula, respectively called the* antecedent *and* succedent *of the sequent, and* L *is a tree whose leaves are labelled by* loops *from* $\mathcal{A}$*. Given two such trees* L_1*,* L_2*, we write* $L_1 \cdot L_2$ *for the tree formed by fusing their roots; we write* $L_1 + L_2$ *for the tree whose root has* L_1 *and* L_2 *as its only subtrees. We don't distinguish between a loop* l *in* $\mathcal{A}$ *and the tree whose only leaf node is* l*.*

Definition 55 *An **LTS*** proof *is a tree of inferences drawn from the rules shown in Figure 1.2; the leaves of the tree must be drawn from the axiom group. A proof is called* multiplicative *if (1) only multiplicative formulae occur in it; and, (2) no rule from the additive group occurs. The reduced sequent calculus consisting only of multiplicative proofs we call **LT**.*

One could summarise the rules of **LTS** as "multiplicative-additive linear logic with self-dual connectives". Certainly one can embed **MALL** into **LTS** by translating both multiplicative connectives as $\otimes$ and both additives as $\oplus$ and nothing will go terribly wrong. However, since both connectives of **LTS** are self-dual, many cuts which would be forbidden in **MALL** are allowed in **LTS**, and we must introduce some novel rules to deal with this. It is worthwhile to point out some of the more idiosyncratic rules.

Axiom Rule In the case that $\mathcal{A}$ is a discrete category then the only arrows are identities so we regain the usual $A \vdash A$ axioms. The restriction of axioms to ground types is for technical convenience; identity axioms for every type are constructable, and indeed admissible.

Unit Rule An distinctive feature of compact closed categories is the presence of loops, so incorporating this rule allows an exact connection between the syntax and the semantics to be established. Perhaps more importantly, the unit rule allows "circular" cuts to be eliminated.

Cut Rule The cut rule, as shown here, might be better described as a trace rule. The more traditional cut rule,

$$\frac{\Gamma \vdash \Delta, A \qquad A, \Gamma' \vdash \Delta'}{\Gamma, \Gamma' \vdash \Delta, \Delta'}\ (\mathsf{cut})$$

Axiom Group: where $f : A \to B$ and $h : A \to A$ are arrows of $\mathcal{A}$.

$$\frac{f}{A \vdash B\ ;\emptyset}\ (f\text{-axiom}) \qquad\qquad \frac{}{\vdash\ ;[h]}\ (h\text{-unit})$$

The Cut:

$$\frac{\Gamma, X \vdash \Delta, X\ ; L}{\Gamma \vdash \Delta\ ; L}\ (\mathsf{cut})$$

Multiplicative Group: σ, τ permutations.

$$\frac{\Gamma \vdash \Delta\ ; L \qquad \Gamma' \vdash \Delta'\ ; L'}{\Gamma, \Gamma' \vdash \Delta, \Delta'\ ; L \cdot L'}\ (\mathsf{mix}) \qquad \frac{\Gamma \vdash \Delta\ ; L}{\tau(\Gamma) \vdash \sigma(\Delta)\ ; L}\ (\mathsf{exchange})$$

$$\frac{\Gamma, X, Y \vdash \Delta\ ; L}{\Gamma, X \otimes Y \vdash \Delta\ ; L}\ (\otimes\text{-L}) \qquad \frac{\Gamma \vdash X, Y, \Delta\ ; L}{\Gamma \vdash X \otimes Y, \Delta\ ; L}\ (\otimes\text{-R})$$

$$\frac{\Gamma \vdash \Delta\ ; L}{\Gamma, I \vdash \Delta\ ; L}\ (I\text{-L}) \qquad \frac{\Gamma \vdash \Delta\ ; L}{\Gamma \vdash \Delta, I\ ; L}\ (I\text{-R})$$

$$\frac{\Gamma \vdash \Delta, X\ ; L}{\Gamma, X^* \vdash \Delta\ ; L}\ (*\text{-L}) \qquad \frac{\Gamma, X \vdash \Delta\ ; L}{\Gamma \vdash \Delta, X^*\ ; L}\ (*\text{-R})$$

Additive Group: where $i = 1$ or 2.

$$\frac{\Gamma, X_i \vdash \Delta\ ; L}{\Gamma, X_1 \oplus X_2 \vdash \Delta\ ; L}\ (\oplus_i\text{-L}) \qquad \frac{\Gamma \vdash \Delta, X_i\ ; L}{\Gamma \vdash \Delta, X_1 \oplus X_2\ ; L}\ (\oplus_i\text{-R})$$

$$\frac{0_Y^X}{X \vdash Y\ ;\emptyset}\ (\mathsf{zero}) \qquad \frac{\Gamma \vdash \Delta\ ; L \qquad \Gamma \vdash \Delta\ ; L'}{\Gamma \vdash \Delta\ ; L + L'}\ (\mathsf{sum})$$

Fig. 1.2. Inference Rules for **LTS**

can be defined in **LTS** using the mix and exchange rules, viz:

$$\cfrac{\cfrac{\cfrac{\Gamma \vdash \Delta, A \qquad \Gamma', A \vdash \Delta'}{\Gamma, \Gamma', A \vdash \Delta, A, \Delta'}\ (\mathsf{mix})}{\Gamma, \Gamma', A \vdash \Delta, \Delta', A}(\mathsf{exchange})}{\Gamma, \Gamma' \vdash \Delta, \Delta'}\ (\mathsf{cut})$$

Mix Rule The mix rule (combined with the two rules for tensor) asserts that the comma is the same on both sides of the sequent, unlike

Axiom Group: where $f : A \to B$ and $h : A \to A$ are arrows of $\mathcal{A}$.

$$\frac{}{f : A \to B}\ (f\text{-axiom}) \qquad \frac{}{\mathrm{Tr}^{A}_{I,I}(h) : I \to I}\ (h\text{-unit})$$

The Cut:

$$\frac{\pi : \Gamma \otimes X \to \Delta \otimes X}{\mathrm{Tr}^{X}_{\Gamma,\Delta}(\pi) : \Gamma \to \Delta}\ (\mathsf{cut})$$

Multiplicative Group: σ, τ permutations.

$$\frac{\pi : \Gamma \to \Delta}{\sigma \circ \pi \circ \tau^{-1} : \tau(\Gamma) \to \sigma(\Delta)}\ (\mathsf{exchange})$$

$$\frac{\pi : \Gamma \to \Delta \qquad \pi' : \Gamma' \to \Delta'}{(\pi \otimes \pi') : \Gamma \otimes \Gamma' \to \Delta \otimes \Delta'}\ (\mathsf{mix})$$

(No interpretation for tensor or I rules)

$$\frac{\pi : \Gamma \to \Delta \otimes X}{(1_\Delta \otimes \epsilon_X) \circ (\pi \otimes 1_{X^*}) : \Gamma \otimes X^* \to \Delta}\ (\mathsf{*\text{-}L})$$

$$\frac{\pi : X \otimes \Gamma \to \Delta}{(1_{X^*} \otimes \pi) \circ (\eta_X \otimes 1_\Gamma) : \Gamma \to X^* \otimes \Delta}\ (\mathsf{*\text{-}R})$$

Additive Group: where $i = 1$ or 2.

$$\frac{\pi : \Gamma \otimes X_i \to \Delta}{\pi \circ (1_\Gamma \otimes p_i) : \Gamma \otimes (X_1 \oplus X_2) \to \Delta}\ (\oplus_i\text{-}\mathsf{L})$$

$$\frac{\pi : \Gamma \to \Delta \otimes X_i}{(1_\Delta \otimes q_i) \circ \pi : \Gamma \to \Delta \otimes (X_1 \oplus X_2)}\ (\oplus_i\text{-}\mathsf{R})$$

$$\frac{\pi : \Gamma \to \Delta \qquad \pi' : \Gamma \to \Delta}{\pi + \pi' : \Gamma \to \Delta}\ (\mathsf{sum})$$

$$\frac{}{0^X_Y : X \to Y}\ (\mathsf{zero})$$

Fig. 1.3. Semantics for rules of **LTS**

most logics. It allows usual two-premise cut and tensor rules to be constructed.

Zero Rule Without the zero rule certain cuts are impossible to remove. It has been noted that the logic of biproducts is inconsistent: every sequent is provable. By including the zero axiom we embrace this inconsistency. A more computational point of view is that every type is inhabited, at least by the divergent program, or in the quantum setting, the evolution with zero probability.

Sum Rule This rule asserts that each **LTS** proof is a (finite) formal weighted sum of **LTS** proofs, with the weights given by the pair L and L'. Otherwise this rule performs a similar role to the mix, allowing the usual binary rules for additives to be constructed.

The formulae of **LTS** are just the objects of $\mathcal{A}$ hence we shall not even bother to distinguish them notationally. To give semantics for **LTS** its remains to translate proofs into arrows of $F\mathcal{A}$.

Definition 56 *Let π be an **LTS** proof of the sequent*

$$X_1, \ldots, X_n \vdash Y_1, \ldots Y_m; L\ .$$

We define its denotation, *an arrow*

$$[\![\pi]\!] : X_1 \otimes \cdots \otimes X_n \to Y_1 \otimes \cdots \otimes Y_m$$

by recursion over the structure of π according to the rules shown in figure 1.3.

Theorem 57 (Cut Elimination) *For every **LTS** proof π of the sequent $\Gamma \vdash \Delta; L$ there exists a proof π' of $\Gamma \vdash \Delta; L$ which contains no occurrence of the cut rule, and such that $[\![\pi]\!] = [\![\pi']\!]$.*

The proof proceeds in the standard way so we omit it here. (The general strategy can be translated from the proof-net version presented below.) We remark that preserving the denotation is the non-trivial part; otherwise the zero rule can be used to give a cut-free proof immediately.

We would like the formulae of **LTS** to be in exact correspondence with the objects of the free category $F\mathcal{A}$. Unfortunately we have several equations between syntactically distinct formulae. To work around this blemish we will introduce a special class of formulae.

Definition 58 *A formula is called* multiplicatively reduced *if it is different to I and contains neither $X \otimes I$ nor $I \otimes X$ as a subformula, for any formula X. A sequent is monoidally reduced if all its formula are monoidally reduced.*

A formula is called additively reduced *if it has no subformula of the forms* $\mathbf{0} \oplus X$, $X \oplus \mathbf{0}$, $\mathbf{0} \otimes X$ *or* $X \otimes \mathbf{0}$. *A sequent is additively reduced if all its formulae are.*

A formula or sequent which is both multiplicatively and additively reduced is simply called reduced.

The content of this definition is that the only place that I may occur in a reduced formula is under the $\oplus$ connective; the only reduced formula containing $\mathbf{0}$ is $\mathbf{0}$ itself.

Proposition 59 *Every sequent is provably equivalent to a reduced one.*

Proof We have the following provable equivalences:

$$X \otimes I \equiv X$$
$$I \otimes X \equiv X$$
$$X \otimes \mathbf{0} \equiv \mathbf{0}$$
$$\mathbf{0} \otimes X \equiv \mathbf{0}$$
$$X \oplus \mathbf{0} \equiv X$$
$$\mathbf{0} \oplus X \equiv X$$

and the denotation of each proof is an isomorphism in $F\mathcal{A}$. The only remaining case is that of a sequent containing the formula I; in this case it can be removed by means of a cut, possibly after adjoining a new I on the left or right as needed. □

This result means that the *reduced* formulae of **LTS** are in 1-1 correspondence with the objects of $F\mathcal{A}$. We call a proof reduced if its conclusion is reduced. Note that we cannot restrict to reduced formulae throughout, since they must be introduced to construct certain formulae, for example $I \oplus I$. Having dealt with the objects of $F\mathcal{A}$ we turn out attention to the arrows.

Theorem 60 (Completeness) *Let f be an arrow in $F\mathcal{A}$; there exists a cut-free **LTS** proof π such that $f = [\![\pi]\!]$.*

We again omit the proof since it follows from a more general result proved below. However this theorem marks the end of the line as far as the sequent system is concerned. Our attempt to find a proof-theoretic characterisation of $F\mathcal{A}$ founders on the usual curse of sequent calculi:

the existence of distinct cut-free forms for the same proof. **LTS** is especially bad in this respect since it enjoys a great many sound commuting conversions.

1.4 Generalised Proof-Nets

In this section we define a system of two-sided proof-nets constructed over the generators of a compact symmetric polycategory $\mathcal{P}$. The resulting system of proof-nets will be denoted $\mathrm{PN}(\mathcal{P})$. Using these proof-nets we obtain a logical system closer in behaviour to a term system: every proof-net has a *unique* normal form. The availability of such normal forms allows us to make an exact correspondence between the $\mathrm{PN}(\mathcal{P})$ and the free compact closed category with biproducts.

1.4.1 Tensor-Sum Proof-Nets

Graphical notations for monoidal categories have been studied as far back as the early seventies [Kel72, Pen71, JS91] and such 2-dimensional representations provide for beautifully simple reasoning in a setting normally awash with coherence equations. When proofs are represented graphically, as in proof-nets for multiplicative linear logic, a further advantage is gained: by relaxing the allowed shapes of proofs, from trees to graphs, the artificial sequentiality imposed by the use of sequent proofs is removed. Work on **MLL** [Gir87b, DR89, BCST96] extended the graphical tensor notation to the case of two tensors which enjoy a "weak" distribution law† [CS97]. In these settings the two tensors are essentially similar, and indeed can be made formally dual. In the following we study the case a single self-dual tensor so while much of the work of [BCST96] applies, we can make some significant simplifications, and are forced into some complications too, but the we retain a purely graphical language for the multiplicative fragment of **LTS**, closely related to the diagrams of [Coe05b]. We note that because **LTS** is so permissive, no correctness criterion, à la Danos and Regnier [DR89], is needed here.

However, into our multiplicative paradise we must admit the additive connectives and this complicates matters. We handle the additive structure using a system of *slices and boxes*. The notion of slices in linear proof-nets first appeared in Girard's original [Gir87a] but was not entirely satisfactory for the unrestricted multiplicative-additive fragment

† Also called *linear* distribution.

of linear logic; the correctness of the proof-structure as a whole could not be derived from the correctness of its slices [Gir96]. A similar notion was later employed in [HvG03] to give a satisfactory notion of MALL proof-net. In the more restricted setting of polarised linear logic, the naive use of slices works very well since the additional constraint of polarity forces the additive connectives to cohere nicely [LTdF04].

Just as compact closed categories are degenerate models of the multiplicative part of linear logic, the biproduct is a degenerate version of the linear logic's additive connectives. Happily, this degeneracy means that slicing will give good results, for essentially the opposite reasons to the polarised case: we have so many equations that the question of correctness becomes trivial. Slices and boxes are defined by mutual recursion.

Definition 61 *A* PN($\mathcal{P}$) proof-slice *is a finite directed graph with edges labelled by* ***LTS*** *formulae. The graph is constructed by composing the following links, while respecting the labelling on the incoming and outgoing edges.*

Premise: *No incoming edges; one outgoing edge. The edge is labelled with an arbitrary formula and the link is unlabelled.*

Conclusion: *One incoming edge; no outgoing edges. The edge is labelled with an arbitrary formula and the link is unlabelled.*

Unit: *No incoming edges; two outgoing edges. The first outgoing edge is labelled X^*, the other, X, for some formula X. The link itself is labelled by η.*

Counit: *Two incoming edges; no outgoing edges. Each counit is labelled by ϵ and its incoming edges are labelled by X and X^* for an arbitrary formula X.*

Tensor: *Two incoming edges labelled X and Y; one outgoing edge labelled $X \otimes Y$.*

Cotensor: *One incoming edge labelled $X \otimes Y$; two outgoing edges labelled X and Y.*

Circle: *No incoming or outgoing edges; a circle is a closed loop labelled by a formula.*

Axiom: *Each polyarrow $f : \langle A_i \rangle_i \to \langle B_j \rangle_j$ in $\mathrm{Arr}_{\mathcal{P}}$ defines a link labelled by f. Its n incoming edges are labelled by $A_1, \ldots, A_n$ and its m outgoing edges are labelled by $B_1, \ldots, B_m$.*

Plus 1: *One incoming edge labelled X; one outgoing edge labelled $X \oplus Y$, for an arbitrary formula Y.*

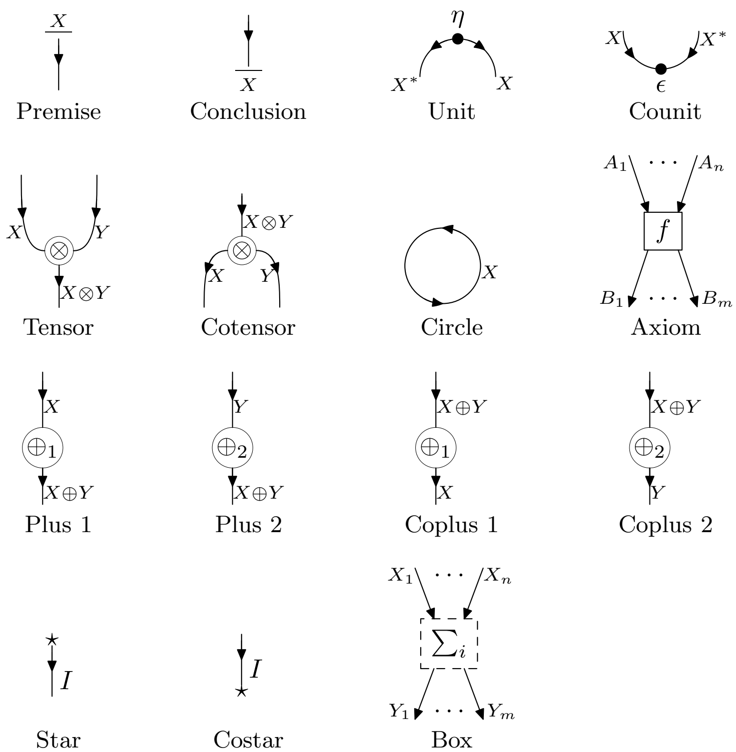


Fig. 1.4. Links for PN($\mathcal{P}$) Proof-nets

Plus 2: *One incoming edge labelled* Y*; one outgoing edge labelled* $X \oplus Y$ *, for an arbitrary formula* X*.*

CoPlus 1: *One incoming edge labelled* $X \oplus Y$*; one outgoing edge labelled* X*.*

CoPlus 2: *One incoming edge labelled* $X \oplus Y$*; one outgoing edge labelled* Y*.*

Star: *One outgoing edge labelled* I*; no incoming edges.*

Costar: *One incoming edge labelled* I*; no outgoing edges.*

Box: *Any numbers of incoming and outgoing formulae edge, labelled by arbitrary formulae—see definition 62 below.*

A proof-slice is oriented *such that edges enter the node from the top, and exit from the bottom. This implies that any premise, star or unit link is above the links they are connected to and, conversely, any conclusion, costar, or counit links are below the links they are connected to.*

The order of premises and conclusions is significant, and the type *of a proof-slice is the pair* (Γ, Δ) *of lists of formulae determined by the premises and conclusions respectively. Usually this will be written as a sequent* $\Gamma \vdash \Delta$*. The empty slice is valid slice, with type* $\ \vdash\ $.

A premise or conclusion link is called atomic *if the formula labelling it is a literal; a proof-slice is called atomic if all its premises and conclusions are atomic. A slice is called* flat *if it contains no boxes*

Proof-slices are permitted to be disconnected or cyclic, when considered as directed or undirected graphs. In particular, an edge may leave a link and return as an input to the same link, although the labelling on edges will prohibit this for all except axiom links. If a proof-slice is directed-acyclic then it is called *process-like*.

Definition 62 *A* PN($\mathcal{P}$) box *is a finite multiset of proof-slices, all of the same type; if its component slices are of type* $\Gamma \vdash \Delta$ *then the formula of* Γ *are the* inputs *of the box, and those of* Δ *are its* outputs*.*

A box may be empty; in which case it may have any inputs and outputs. Indeed, such an empty box is the only normal proof of the formula **0**.

Operationally a box may be viewed as local classical knowledge (or rather, non-determinism) embedded in one part of the system — the distribution of addition over composition codes the transmission of this information. The details of this distribution, presumably mediated by some classical control structure, will not investigated here, but it seems an interesting direction for further exploration.

Definition 63 *Let* s *be a proof-slice; define its* depth $d(s)$ *as*

$$d(s) = \sum_{i=1}^{k} d(b_i) + k$$

where the b_i *range over the boxes occurring in* s*. Let* b *be a box containing slices* $\{s_i\}_i$*; then define its depth* $d(b)$ *by*

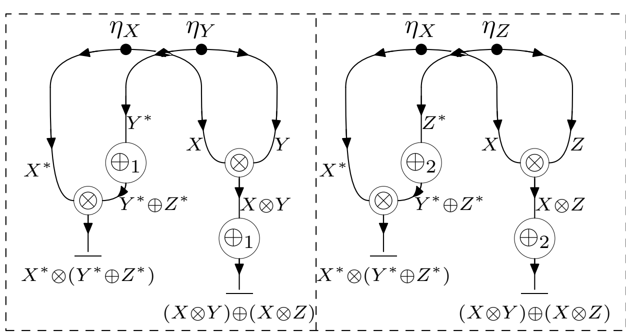
$$d(b) = \sum_{i} d(s_i)$$

Definition 64 *A* PN($\mathcal{P}$) proof-net *is a* PN($\mathcal{P}$) *box of finite depth.*

According to Definition 64 every slice is contained in a box, which is called the *ambient box* for that slice. Also, if a slice s contains a box, the slices contained within that box are not considered part of s; that is, from the point of view of their containing slice boxes are considered totally opaque. On the other hand, the phrase "a slice in proof-net π" should be understood unrestrictedly, as denoting a slice at any level of nesting with the proof-net structure. Since proof-nets have finite depth, any box occurring within a proof-net is itself a valid proof-net; hence there is no loss in generality by assuming that ambient box is always the top-level.

Since PN($\mathcal{P}$) proof-nets are rather unwieldy objects, it is helpful to introduce a symbolic shorthand for working with them algebraically. Writing s_i for a sequence of slices, a box containing those slices is written as a summation, $\sum_i s_i$. The crucial ingredient is a "slice with a hole". A hole can be thought of as a link with arbitrary incoming and outgoing edges; we write $s\{\ \}$ to represent a slice with a hole. Such an object is not a valid part of our notation, we introduce it only in order write such expressions as $s\{t\}$, where t is a proof-net fragment having the same incoming and outgoing edges as the hole in s such that the slice produced by replacing the hole in s with the fragment t is a valid slice. Simply writing the empty brackets $\{\ \}$ denotes a slice which all hole – it has no structure besides its type. We use letters π, π' to denote proof-nets; $\pi\{\ \}$ should be understood as a proof-net with a hole in one of its slices. When we write $\pi\{\ \}\{\ \}$ to denote a proof-net with two holes, it should always be understood that both holes are in the same slice. It will never be necessary to speak of holes in separate slices.

Example 65 *This 2 sliced net encodes the distribution of $\otimes$ over $\oplus$.*



Since categories are a special case of polycategories, we can define PN($\mathcal{A}$) equally well when $\mathcal{A}$ is just a category. In this case the axiom links have exactly one input and one output; there is one for each arrow of $\mathcal{A}$. In this situation we can translate from **LTS** sequent proofs to proof-nets.

Definition 66 (Translation from sequents) *Given an **LTS** proof π, we define a proof-net $N\pi$ by recursion over the structure of π.*

- *If proof π is just an f-axiom, let $N\pi$ be the single slice containing just the corresponding axiom link, connected to a premise and a conclusion link, leaving a net of type $A \vdash B$.*
- *If proof π is a just an application of the h-unit rule for some $h : A \to A$, we form $N\pi$ by introducing h as an axiom, as described above, and forming a cut between, as described below, between its input and output.*
- *If π is simply an application of the zero rule then $N\pi$ is an empty box with the desired type.*
- *If π arises from π' by an application of the cut rule for arrow on some formula X form $N\pi$ from $N\pi'$ by replacing, in every slice of $N\pi'$, the premise link corresponding to the negative occurrence of X with an η_X link, and replacing the conclusion link corresponding to the positive occurrence of X with an ϵ_X link. The X^* output of the new unit link is connected to the X^* input of the new counit link.*
- *Suppose π arises from subproofs π_1 and π_2 by the mix rule. Then let $N\pi = \{N\pi_1\}\{N\pi_2\}$ i.e a single slice containing two boxes, one for each subproof.*
- *If π arises from π' by an application of the ($\otimes$-R) rule, form $N\pi$ adding, in every slice, a tensor-link between the conclusions of $N\pi'$ corresponding to the active formulae of the rule.*
- *If π arises from π' by an application of the ($\otimes$-L) rule, form $N\pi$ adding, in every slice, a cotensor-link between the premises of $N\pi'$ corresponding to the active formulae of the rule.*
- *If π arises from π' by an application of the (*-R) rule on some formula X, form $N\pi$ adding, in every slice, an η_X -link between to the premise of $N\pi'$ corresponding to the active formulae of and connect its X^* output to a new conclusion link.*
- *If π arises from π' by an application of the (*-L) rule on some formula X, form $N\pi$ adding, in every slice, an ϵ_X -link between to the conclusion of $N\pi'$ corresponding to the active formulae of and connect its X^* output to a new premise link.*
- *If π arises from π' by an application of the (I-R) rule, form $N\pi$*

adding, in every slice, a star-link, connected a new conclusion link.

- *If π arises from π' by an application of the (I-L) rule, form $N\pi$ adding, in every slice, a costar-link, connected a new premise link.*
- *If π arises from π' by an application of the ($\oplus_i$-R) rule, form $N\pi$ by adding a plus-i-link to the conclusion corresponding to the active formula in every slice of $N\pi'$.*
- *If π arises from π' by an application of the ($\oplus_i$-L) rule, form $N\pi$ by adding a coplus-i-link to the premise corresponding to the active formula in every slice of $N\pi'$.*
- *Suppose π arises via an application of the sum rule to proofs π_1 and π_2; suppose also that $N\pi_1 = \sum_i s_i$ and $N\pi_2 = \sum_j t_j$. Then let $N\pi = \sum_i s_i + \sum_j t_j$.*

1.4.2 Normalisation

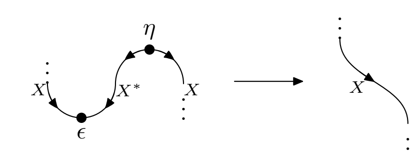
Definition 67 *Let e be an edge in a slice, going from some link L_1 to link L_2. We say that e is* expandable *when:*

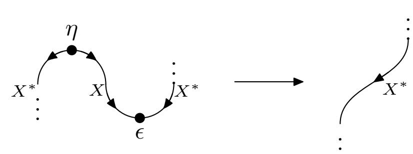
(i) *e is labelled by a a compound formula (i.e. either $X \otimes Y$ or $X \oplus Y$);*
(ii) *L_1 is a premise, cotensor, or coplus link; and,*
(iii) *L_2 is a conclusion, tensor, or plus link.*

Definition 68 (Rewrite Steps) *Let ν, μ be proof-nets; define a one step reduction relation on proof-nets R_β such that $\nu \, R_\beta \, \mu$ if ν can be rewritten to μ by one of the following local rewrite rules.*

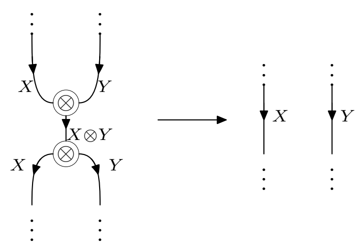
Elimination Rules

ηϵ-elim:

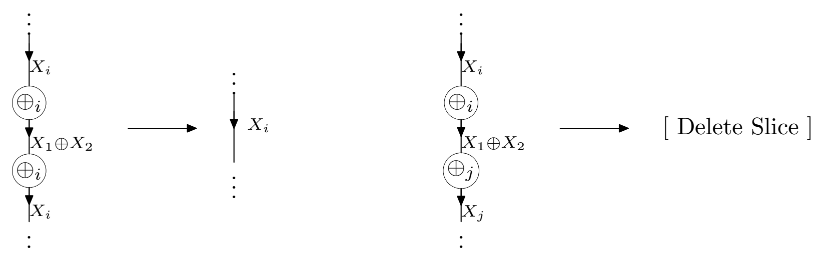




⊗***-elim****:*

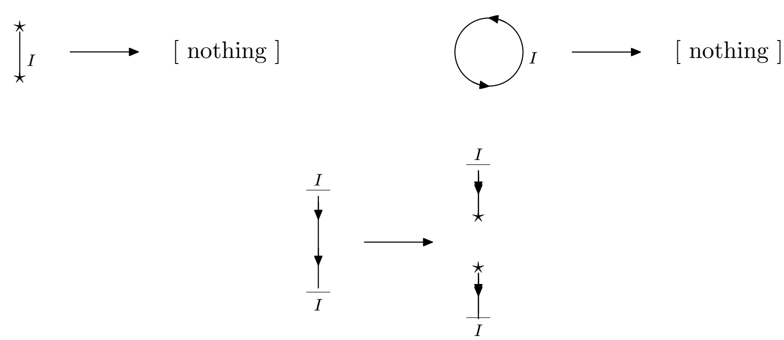


⊕***-elim***

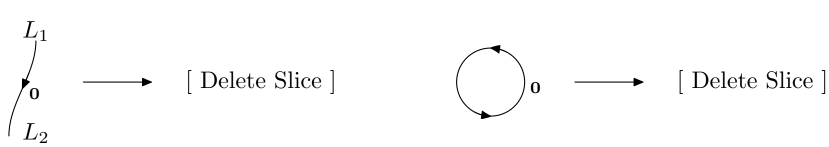


where $i \neq j$*.*

I***-elim****:*

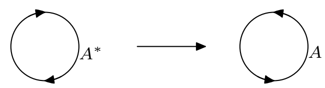


0*-elim*



where either of the links L_1, L_2 *is a premise, conclusion, tensor, cotensor, unit, or counit.*

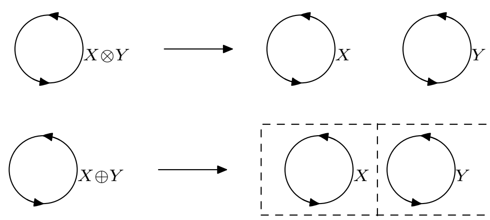
Circle reversal:



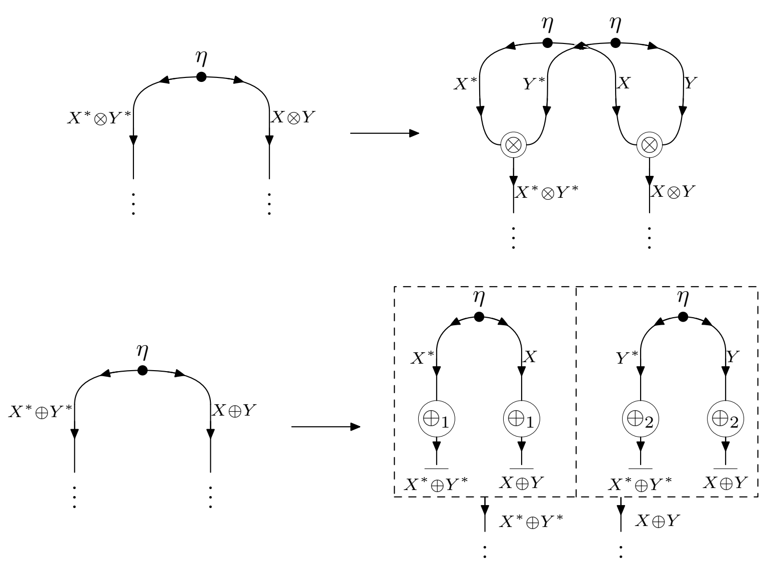
where A is an atom.

Expansion Rules

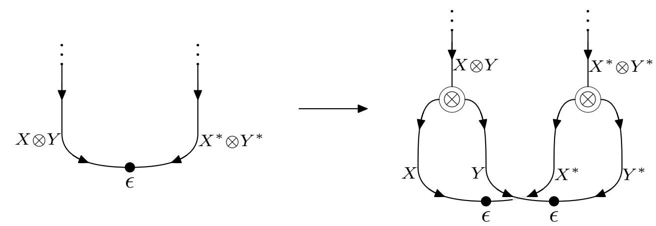
circle expansion:

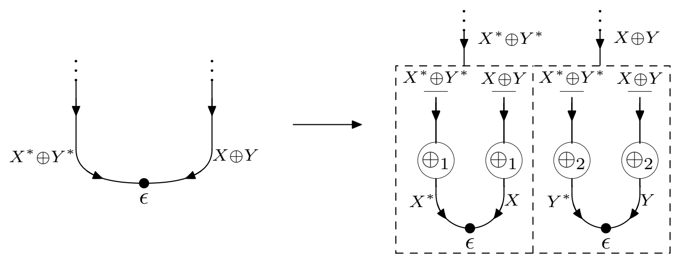


η-expansion:

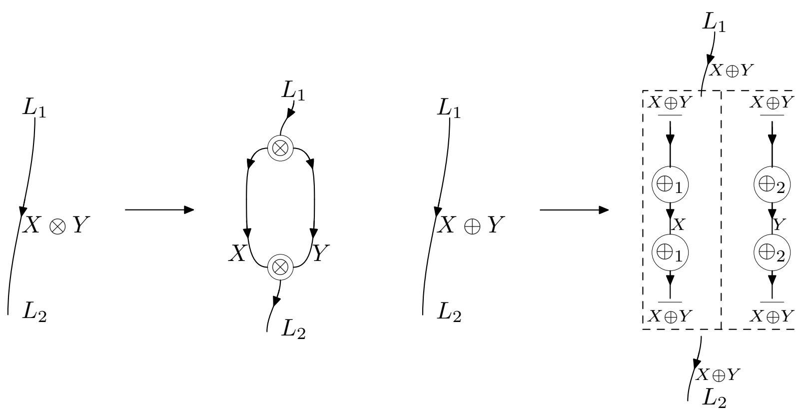


ϵ-expansion:





⊗- and ⊕-expansions:



An edge from link L_1 to link L_2 and labelled by a compound formula X is expanded when both of the following hold:

- *L_1 is a cotensor coplus, or premise link; and*
- *L_2 is a tensor plus, or conclusion link.*

Unboxing Rule

If a slice s contains a box $b = \sum_i t_i$, replace s in the ambient box via

$$s\{\sum_i t_i\} \xrightarrow{\beta} \sum_i s\{t_i\}.$$

i.e. make a new copy of s for each slice in b, and in each replace b with the slice.

Definition 69 *Let $\xrightarrow[\beta]{}$ be the transitive, reflexive closure of R_β and let $=_\beta$ be the symmetric closure of $\xrightarrow[\beta]{}$.*

Lemma 70 (Subject Reduction) *Suppose that ν is a proof-net with type (Γ, Δ) and $\nu \xrightarrow[\beta]{} \mu$; then μ also has type (Γ, Δ).*

Proof *No rewrites change the premises or conclusions, hence the type is unchanged by β-reduction.* □

We now begin the approach the proof that β-reduction is strongly normalising. First some intermediary definitions.

Definition 71 *Let X be a formula; define its* depth *$d(X)$ by*

$$d(A) = d(A^*) = d(I) = d(\mathbf{0}) = 1$$

$$d(X \otimes Y) = d(X)d(Y) \qquad\qquad d(X \oplus Y) = d(X) + d(Y)$$

Lemma 72 *Let e be an expandable edge labelled by X; then e can be expanded to give a box with at most $d(X)$ slices.*

Proof We use induction on X. Suppose that X contains a connective; the expansion rule for that connective will introduce expandable edges labelled by the subformulae Y and Z. By induction, these yield boxes with $d(Y)$ and $d(Z)$ slices respectively. If $X = Y \otimes Z$; then the expansion rule introduces the new edges in parallel; applying the unboxing rule to one then the other we, obtain $d(Y)d(Z)$ slices.

Alternatively suppose $X = Y \oplus Z$. The expansion rule introduces a box containing two slice, with an expandable edge in each. Again, we can apply the unboxing rule twice, and obtain a box with $d(Y) + d(Z)$ slices. □

Definition 73 *We define the* size *of a proof-net π, written $n(\pi)$ by mutual recursion over slices and boxes. Let s be a slice with boxes b_i; then*

$$n(s) = \prod_i d(b_i) \left(N_s + \sum_i \frac{n(b_i)}{d(b_i)} \right)$$

where N_s is the number of links found in s, except conclusions, premises and boxes. If we have a box $b = \sum_j t_j$ then let

$$n(b) = \sum_j n(t_j) \ .$$

Definition 74 *We define the* rank *of a proof-net π, written $r(\pi)$ by mutual recursion over slices and boxes. Let s be a slice with boxes b_i; then*

$$r(s) = \prod_i d(b_i) \left(K_s + \sum_i \frac{r(b_i)}{d(b_i)} \right)$$

where K_s is the total number of times the symbols $\otimes$ and $\oplus$ occur in the labels of expandable edges of s. If we have a box $b = \sum_j t_j$ then let

$$r(b) = \sum_j r(t_j) \ .$$

The following lemma is immediate from the definitions:

Lemma 75 *Let $s\{\sum_i t_i\}$ be a slice in a proof-net; then the following hold:*

$$n(s\{\sum_i t_i\}) = \sum_i n(s\{t_i\})$$
$$r(s\{\sum_i t_i\}) = \sum_i r(s\{t_i\})$$
$$d(s\{\sum_i t_i\}) > \sum_i d(s\{t_i\})$$

Theorem 76 (Termination) *Every β-reduction sequence is finite.*

Proof We define an order on proof-nets by setting $\pi \succ \pi'$ whenever $(r(\pi), n(\pi), d(\pi)) > (r(\pi'), n(\pi'), d(\pi'))$ in the lexicographic order. Note that these quantities are all non-negative integers so this order has no infinite decreasing chain.

Suppose now that $\pi R_\beta \pi'$. By inspection of the rules we notice:

- if the rewrite is an expansion, then we have $r(\pi) > r(\pi')$;
- if the rewrite is an elimination rule then $n(\pi) > n(\pi')$ and $r(\pi) \geq r(\pi')$; and,
- if the rewrite is the unboxing rule then by Lemma 75 we have that $r(\pi) = r(\pi')$, $n(\pi) = n(\pi')$ and $d(\pi) > d(\pi')$.

Hence $\pi \xrightarrow[\beta]{} \pi'$ then necessarily $\pi \succ \pi'$, and therefore every rewrite sequence terminates. □

Theorem 77 (Local Confluence) *If a proof-net π β-reduces to π_1 and π_2 by different rewrites r_1, r_2, then there exist sequences of rewrites s_1, s_2 such that*

$$\begin{array}{ccc} \pi & \xrightarrow{r_1} & \pi_1 \\ {\scriptstyle r_2}\downarrow & & \downarrow{\scriptstyle s_1} \\ \pi_2 & \xrightarrow[s_2]{} & \pi^* \end{array}$$

Proof Within a box, rewrites on one slice do not affect any of the other slices. Hence, for a conflict to exist, either r_1 and r_2 both affect the same slice or that r_2 operates on a child slice of that where r_1 acts. Otherwise there is no conflict between the rewrites and they can be trivially unified.

The rules also exhibit locality in the vertical direction. The only rule which allows slices on different levels to interact is the unboxing rule—and this only pulls slices up from the level below. Hence if r_1 and r_2 act on slices which are more than two levels apart they do not conflict.

Observe that there are three kinds of rules in the system: those that add slices to the ambient box (just the unboxing rule); those that delete a slice (the zero rule and the incoherent case of $\oplus$-elimination); and those which have purely local effect (all the rest). We'll deal with the cases in that order.

Suppose that the rewrite r_1 is the unboxing rule; without loss of generality we have

$$\pi = \sum_i s_i + s\{\sum_j t_j\} \xrightarrow{r_1} \sum_i s_i + \sum_j s\{t_j\} = \pi_1$$

Since we need only consider the case where r_2 acts on $s\{\ \}$ or one of the t_j, the other slices s_i will be neglected. Suppose r_2 acts on $s\{\ \}$:

- If r_2 is the unboxing rule acting on some other box then we have

$$\sum_i s\{t_i\}\{\sum_j t'_j\} \xleftarrow{r_1} s\{\sum_i t_i\}\{\sum_j t'_j\} \xrightarrow{r_2} \sum_j s\{\sum_i t_i\}\{t'_j\}$$

  which can be unified by repeating r_2 in each slice on the left, and

r_1 in each slice on the right:

$$\sum_i s\{t_i\}\{\sum_j t'_j\} \overset{\sum_j r_2}{\dashrightarrow} \sum_i \sum_j s\{t_i\}\{t'_j\} \overset{\sum_i r_1}{\dashleftarrow} \sum_j s\{\sum_i t_i\}\{t'_j\}$$

- If r_2 deletes s then this must be due some structure in $s\{\ \}$ hence the same rule can delete each of the $s\{t_i\}$, which suffices to unify the divergence,
- If r_2 is any other rewrite then we have

  $$\sum_i s\{t_i\} \xleftarrow{r_1} s\{\sum_i t_i\} \xrightarrow{r_2} s'\{\sum_i t_i\}$$

  where r_2 matches some structure in $s\{\ \}$, hence it is still available in each of the $s\{t_i\}$, permitting the unification:

  $$\sum_i s\{t_i\} \overset{\sum_i r_2}{\dashrightarrow} \sum_i s'\{t_i\} \xleftarrow{r_1} s'\{\sum_i t_i\}$$

Now suppose r_s acts on one of the t_i, which we simply call t.

- Suppose r_2 is the unboxing rule acting on some box in t:

  $$t \xrightarrow{r_2} \sum_j t'_j$$

  Then we have the divergence

  $$\sum_i s\{t_i\} + s\{t\} \xleftarrow{r_1} s\{\sum_i t_i + t\} \xrightarrow{r_2} s\{\sum_i t_i + \sum_j t'_j\}$$

  which we unify using repeated application of the unboxing rule.

  $$\sum_i s\{t_i\} + s\{t\} \dashrightarrow \sum_i s\{t_i\} + \sum_j s\{t'_j\} \xleftarrow{s} \{\sum_i t_i + \sum_j t'_j\}$$

- Suppose that r_2 deletes t; then same rule will delete $s\{t\}$, which will unify the divergence.
- Otherwise r_2 rewrites t to some t'; again this same rewrite will do $s\{t\} \to s\{t'\}$ which will unify the divergence.

This shows that the unboxing rule cannot conflict with the others.

Now suppose that r_1 deletes slice s. This implies that s contains either a pair of incoherent $\oplus$-links or an edge labelled by $\mathbf{0}$. Notice that none of the slice-local rules can remove either of these features from the graph. Hence regardless of which rule it is, no slice-local rule r_2 can block r_1, so the divergence can always be unified by deleting s. Of course, If r_2 also deletes s then there is no divergence.

Finally we consider the case where both r_1 an r_2 are slice-local. Since all the action is within a single slice, it suffices to show that any pair of overlapping rewrites which diverge can be unified.

Due to the large number of expansion rules, there are a very large number of potential critical pairs. Fortunately the rules are very regular and have been carefully designed to ensure their confluence. For reasons of space we do not include this analysis here, but checking all the pairs is a routine, albeit lengthy, exercise.

All divergent rewrites can be unified, hence $\mathrm{PN}(\mathcal{P})$ is locally confluent under β-reduction. □

Theorem 78 (Strong Normalisation) *β-reduction for proof-nets is strongly normalising.*

Proof Since β-reduction is confluent, each proof-net has a unique normal form; since it is terminating, every rewrite sequence must arrive at the normal form. □

Having established the existence of β-normal proof-nets, we now characterise them intrinsically. Recall that for multiplicative linear logic proof nets [Gir87b], the structure of a cut free proof can be separated into the axiom structure and the connective structure. The following lemmas give a similar result, flattening the additive structure and pushing the connectives to the outside of the proof-net.

Lemma 79 *Let π be a normal proof-net; then every slice of π is flat.*

Proof If any slice contains a box, we can apply the unboxing rule, contradicting the normality of π. □

Since the box structure of a normal net is trivial, we turn our attention to the structure of the slices. Notice that we will assume that a normal slice is in a normal net, and hence the rule for $\mathbf{0}$-elimination has been applied, implying that a normal slice contains no edge labelled by $\mathbf{0}$.

Lemma 80 *Let π be a normal proof-slice, and suppose x is a link in π.*

- *If x is a tensor or a plus link, all links below x are tensors, pluses, or conclusions.*
- *If x is a cotensor or coplus link, all links above x are cotensors, copluses, or premises.*

Proof Let x be either a tensor link, or a plus link. Its outgoing edge is labelled by some formula, either $X \otimes Y$ or $X \oplus Y$; suppose there is a link below it, called x'. Note that since π is normal, x' cannot be a box.

- If x' is a counit then it is labelled by a non-atomic formula, hence an ϵ-expansion rewrite applies and π is not normal.
- If x' is an axiom, it has an incoming edge labelled by a non-atomic formula, which contradicts the definition of axiom link.

Now there are two cases depending on what kind of link x is.

- Suppose that x is a tensor link; then x' cannot be a coplus link because its incoming formula is $X \otimes Y$. Suppose that x' is a cotensor link: then rewrite rule $\otimes$-elim applies, hence π is not normal.
- Suppose that x is a plus link; then x' cannot be a cotensor link because its incoming formula is $X \oplus Y$. Suppose that x' is a coplus link: then rewrite rule $\oplus$-elim applies, hence π is not normal.

Hence x' cannot be a coplus, cotensor, counit, or axiom link. If it is a conclusion then the hypothesis is satisfied. If x' is a tensor or plus link, then by induction all the links below x' are also tensors, pluses, or conclusions.

The case when x is a cotensor or coplus is exactly dual. □

Corollary 81 *Any normal proof-slice π can be formed from a normal atomic slice π' by adding tensor and plus links to its conclusions and cotensors and copluses to its premises.*

Corollary 82 *All the edges of a normal atomic proof-net are labelled by literals.*

Proposition 83 *An atomic proof-slice is normal if and only if: all its edges are labelled by atomic formulae; its circles are labelled by positive atoms; no edge is labelled by* **0***; every edge labelled by I connects a premise to a costar, or a star to conclusion; and no unit link is connected to a counit link;*

Proof If π is normal, Corollary 82 gives that all its edges' labels are atomic; by its normality no unit is connected to a counit since otherwise rewrite $\eta\epsilon$-elim 1 or 2 would apply. Since π is normal, it contains no boxes, hence the formula **0** can only introduced by a premise of conclusion link, but which in this case the **0** elimination rule would apply.

Since π is atomic, the formula I may only be introduced by star, costar, premise, or conclusion links; any such edge labelled by I is can be eliminated unless it connects a premise to a costar, or a star to conclusion as required.

Conversely, suppose that π is atomic, such that all the above conditions are satisfied. Since all its edges are labelled by literals, none of the expansion rules can apply. For the same reason it contains no tensor, cotensor, plus, coplus, or box links, hence rewrites for $\otimes$, $\oplus$, $\mathbf{0}$, and circle elimination do not apply, nor does unboxing. Star and costar links can only appear in forms such that the I elimination rules do not apply. By hypothesis, no unit is connected to a counit, hence rewrites $\eta\epsilon$-elim 1 and 2 cannot apply, and circles are labelled by positive literals the circle reversal rule does not apply. Since, no rewrites are possible, π is in its normal form. □

1.4.3 The categorical structure of PN($\mathcal{P}$)

In this section we prove main remaining theorems about PN($\mathcal{P}$). First we show that PN($\mathcal{P}$) forms a compact closed category with biproducts; and then we show that it is a representation of the free compact closed category with biproducts generated by $\mathcal{P}$.

Proposition 84 *The class of proof-nets,* PN($\mathcal{P}$)*, forms a category.*

The objects of PN($\mathcal{P}$) are **LTS** formulae. An arrow $\pi : X \to Y$ is a proof-net whose only premise is X and whose only conclusion is Y. Two arrows in PN($\mathcal{P}$) are considered equal if they have the same normal form.

Note that the restriction to single formulae is rather weak since the comma of tensor-sum logic is implicitly the tensor; given a proof-net not in this form, we may insert tensor links between the conclusions, and cotensors between the premises, to obtain a proof-net of the desired kind. The restriction to single formulae also avoids having to provide a bracketing, since the connectives of **LTS** are not strictly associative.

We define the identity proof-net 1_X to be a net with one slice, containing only a premise link and a conclusion link, both labelled by X. (Note that since the edge linking them may be expandable, this is not usually the normal form.)

Before defining composition of nets, we first define it for slices. Suppose s, t are proof-slices such that both the conclusion of s, and the premise of t, are some formula X; we define $t \circ s$ by removing the conclusion link of s, removing the premise link of t, and forming a new slice by joining the two graphs along the resulting open edges. Notice that this operation is manifestly associative. Further, we have equations

$$1_X \circ s = s \qquad \text{and} \qquad t \circ 1_X = t\ ,$$

since, considering the first case only, we have simply removed a conclusion link from s and adjoined an identical conclusion link. The other case is the same.

Now let $f : X \to Y$ and $g : Y \to Z$ be proof-nets, with slices f_i and g_j respectively; their composition $g \circ f = \sum_{ij} g_j \circ f_i$ where the composition on slices is as above. Given a third net $h : Z \to W$, we have

$$h \circ (g \circ f) = \sum_{ijk} h_k \circ (g_j \circ f_i) = \sum_{ijk} (h_k \circ g_j) \circ f_i = (h \circ g) \circ f$$

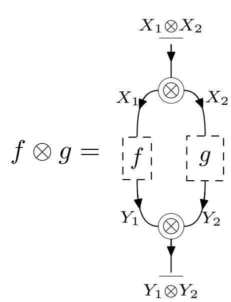
so composition of proof-nets is associative as required. The identity equations

$$1_Y \circ f = 1_Y \circ (\sum_i f_i) = \sum_i (1_Y \circ f_i) = \sum_i f_i = f$$
$$f \circ 1_Y = (\sum_i f_i) \circ 1_Y = \sum_i (f_i \circ 1_Y) = \sum_i f_i = f$$

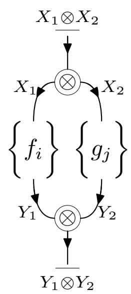
follow directly from the slice case. Hence all the axioms required to be a category are satisfied.

Proposition 85 PN($\mathcal{P}$) *is compact closed.*

First we define the monoidal structure of PN($\mathcal{P}$). Let $f : X_1 \to Y_1$ and $g : X_2 \to Y_2$ be proof-nets; then define their tensor product as



If $f = \{f_i\}_i$ and $g = \{g_j\}$ then by unboxing we have

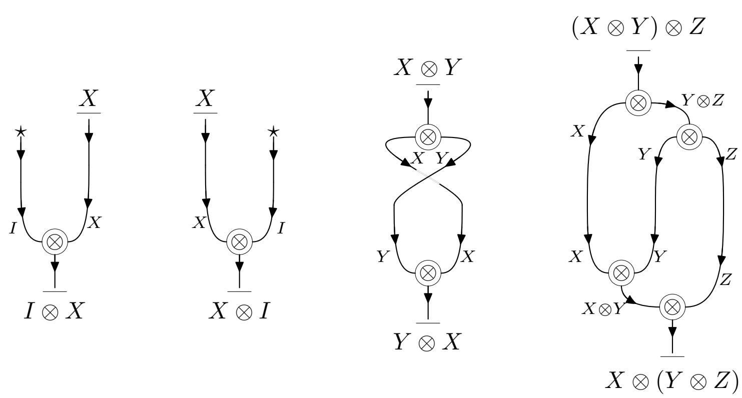
$(f \otimes g)_{ij} =$ 

Let $f' : Y_1 \to Z_1$ and $g' : Y_2 \to Z_2$ be proof-nets then we have the equation

$$(f' \otimes g') \circ (f \otimes g) = (f' \circ f) \otimes (g' \circ g)$$

via the reduction sequence shown in Figure 1.5. To see that $1_{X \otimes Y} = 1_X \otimes 1_Y$ we simply observe that $1_{X \otimes Y} \xrightarrow[\beta]{} 1_X \otimes 1_Y$ by $\otimes$-expansion. Hence $\otimes$ does indeed define a functor.

The left unit, right unit, symmetry, and associativity isomorphisms are defined by



We leave as an easy exercise to check that the required coherence equations are satisfied.

Turning attention now to the compact structure, recall that every formula X has its de Morgan dual X^* as defined in Definition 53. The

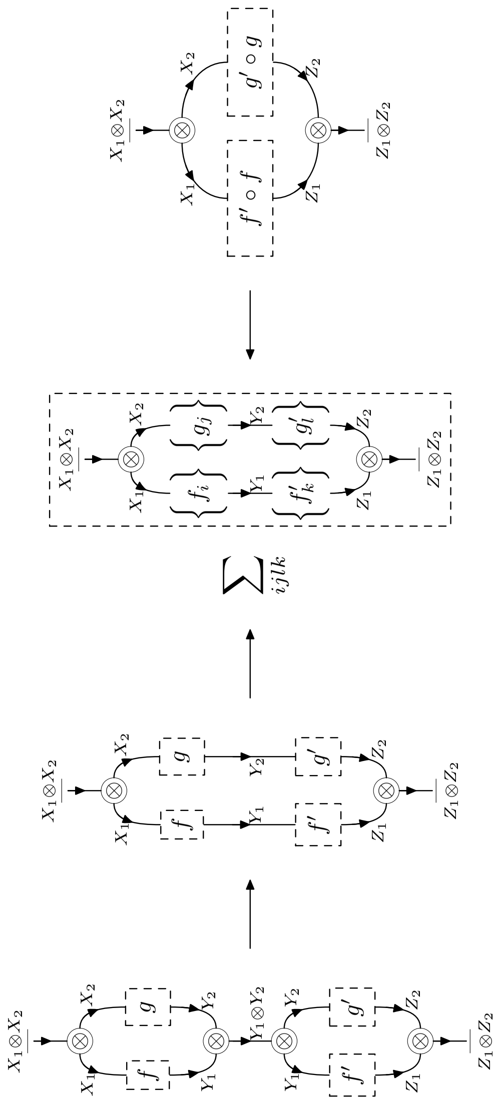
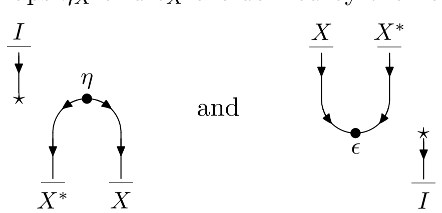


Fig. 1.5. Reduction sequence showing that $(f' \otimes g') \circ (f \otimes g) = (f' \circ f) \otimes (g' \circ g)$

unit and counit maps η_X and ϵ_X are defined by the nets



The required equations follows more or less immediately from the $\eta\epsilon$-elimination rules. Hence PN($\mathcal{P}$) is compact closed.

Proposition 86 PN($\mathcal{P}$) *is enriched over commutative monoids.*

Proof This property follows directly from the slice structure of proof-nets. If $f, g : X \to Y$ are proof nets then $f + g$ is just the proof net containing all slices of both f and g; since the order of the slices is not significant this operation is commutative. The net with no slices, denoted $\emptyset$, gives the zero element. □

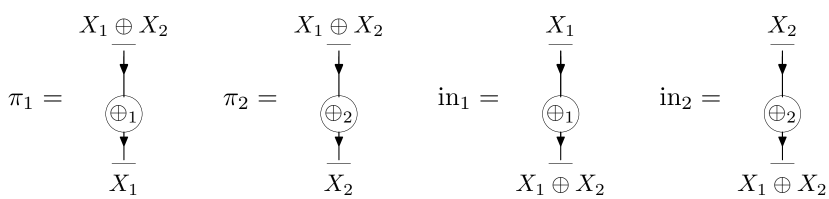
Proposition 87 PN($\mathcal{P}$) *has a* $\mathbf{0}$ *object.*

Proof Obviously, the formula $\mathbf{0}$ is the zero object. Note that for any formula X, the empty proof-net (i.e the net with no slices) provides a proof $\emptyset : X \to \mathbf{0}$ and also $\emptyset : \mathbf{0} \to X$.

Suppose that we have a proof-net $f : X \to \mathbf{0}$. Each slice in f must contain a conclusion link labelled by $\mathbf{0}$; hence by the rule for $\mathbf{0}$-elimination, every slice of f must be deleted, so the normal form of f is the empty proof-net. Hence, for every X, there is exactly one arrow of type $X \to \mathbf{0}$, and similarly there is exactly one arrow $\mathbf{0} \to X$, so $\mathbf{0}$ is both initial and terminal in PN($\mathcal{P}$). □

Proposition 88 PN($\mathcal{P}$) *has biproducts.*

Proof Consider the following one-sliced proof-nets:



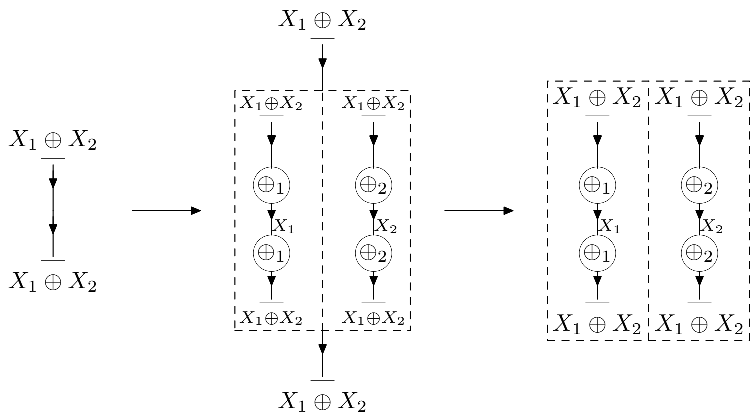
Observe that the rules for $\oplus$-eliminations imply that

$$\pi_j \circ \mathrm{in}_i = \begin{cases} 1_{X_i} & \text{if } i = j \\ \emptyset & \text{if } i \neq j \end{cases}$$

Next, consider the identity map $1_{X_1 \oplus X_2}$. We have the equation

$$1_{X_1 \oplus X_2} = \sum_{i=1,2} \mathrm{in}_i \circ \pi_i$$

via the rewrite sequence below.

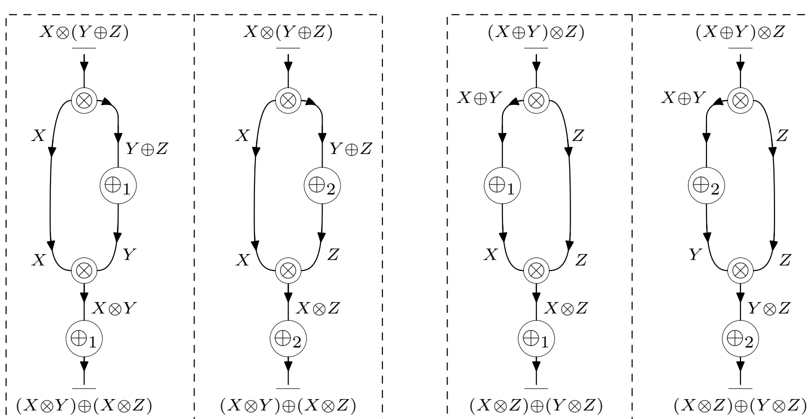


Since we can form these maps for any pair of objects and, by Propositions 86 and 87, PN($\mathcal{P}$) is a **CMon**-category with a **0** object, the result now follows by Proposition 42. □

Proposition 89 *In* PN($\mathcal{P}$) *we have natural distribution isomorphisms:*

$$X \otimes (Y \oplus Z) \cong (X \otimes Y) \oplus (X \otimes Z)$$
$$(X \oplus Y) \otimes Z \cong (X \otimes Z) \oplus (Y \otimes Z)\ .$$

Proof The required maps are given by the proof-nets show below. We leave the reader to check that these nets do indeed define natural isomorphisms.



□

The preceding six propositions established that PN($\mathcal{P}$) is indeed a compact closed category with biproducts as described in Section 1.2.8. Note further that the objects of PN($\mathcal{P}$)—the **LTS** formulae— are freely generated from the atoms, which are themselves the objects the underlying polycategory $\mathcal{P}$. Every object of PN($\mathcal{P}$) is therefore isomorphic to a formula in disjunctive normal form,

$$X \cong \bigoplus_i \otimes_{j_i} X_{j_i} \ ,$$

where the X_{j_i} are literals, and the constants $\mathbf{0}$ and I occur only when a sum or product is empty. (We assume some given bracketing of the connectives.) Hence every proof-net $f : X \to Y$ is equivalent to some f' of the form:

$$f' : \bigoplus_i \otimes_{j_i} X_{j_i} \to \bigoplus_{i'} \otimes_{j'_i} Y_{j'_i} \ .$$

Since f' is a arrow between sums, we can consider its matrix elements $\pi_i \circ f' \circ \mathrm{in}_j$. Without loss of generality take f' to be in normal form; by Lemmas 79 and 80 f' consists of flat slices, whose connective links are all at the outside, and since its type is in disjunctive normal form all its plus and coplus links are outside all its tensor and cotensor links. Hence

$$f' = \sum_k \mathrm{in}_{i_k} \circ f_k \circ \pi_{j_k}$$

where each f_k is a proof-slice between multiplicative formulae. Hence,

$$\begin{aligned} \pi_i \circ f' \circ \mathrm{in}_j &= \pi_i \circ (\sum_k \mathrm{in}_{i_k} \circ f_k \circ \pi_{j_k}) \circ \mathrm{in}_j \\ &= \sum_k \pi_i \circ \mathrm{in}_{i_k} \circ f_k \circ \pi_{j_k} \circ \mathrm{in}_j \\ &= \sum_{k'} f_{k'} \end{aligned}$$

where $k' \in \{k \mid j_k = j \text{ and } i_k = i\}$. By Corollary 81 each of the $f_{k'}$ corresponds to a unique normal atomic slice, which is monoidally reduced. Hence, the only part the structure of PN($\mathcal{P}$) which is not freely generated by its connectives are the normal atomic slices; we now characterise these, and by so doing prove that PN($\mathcal{P}$) is a representation of the free compact closed category with biproducts generated by $\mathcal{P}$.

The reduced normal atomic proof-slices are very closely related to the $\mathcal{P}$-labellable circuits. Let PN($\mathcal{P}$)$_N$ denote the subcategory of PN($\mathcal{P}$)

determined by the multiplicative formulae, and flat, single-sliced proof-nets. We take $\mathrm{PN}(\mathcal{P})_N$ to be monoidally strict, hence its arrows are in 1-1 correspondence to the reduced atomic proof-slices. A simple formal transformation produces a circuit from each such proof-slice, and vice-versa. This correspondence can be boosted upto a pair of functors

$$\mathbf{Circ}(\mathcal{P}) \underset{U}{\overset{F}{\rightleftarrows}} \mathrm{PN}(\mathcal{P})_N$$

which form an equivalence of categories.

Lemma 90 *Suppose ν is an atomic normal proof-net; suppose e is an edge in ν labelled by a negative literal. One of the following holds:*

- *e connects a premise to a conclusion;*
- *e connects a premise to a counit link;*
- *e connects a unit link to a conclusion.*

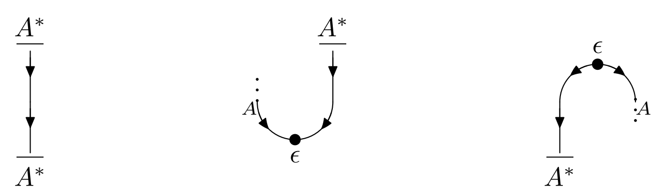


Fig. 1.6. Negative Edges

Proof By Lemma 80, ν contains no tensor, cotensor, plus, or coplus links, nor any boxes; neither axioms nor stars nor costars can introduce negative negative edges, therefore e must connect either a premise, unit, counit or conclusion. Since the proof-net is normal, e cannot join a unit to a counit by the preceding lemma. Since an edge cannot be incoming or outgoing at both endpoints the pairings unit/unit, counit/counit, premise/premise, conclusion/conclusion, conclusion/counit and premise /unit are excluded. This leaves the three possibilities claimed. These can occur validly in a normal proof-net as shown by Fig. 1.6. □

Suppose that $\pi : \Gamma \to \Delta$ is normal and atomic; then π can be rewritten to produce an $\mathcal{P}$-labelled circuit $c(\pi) : \bigotimes \Gamma \to \bigotimes \Delta$ by the following procedure.

(i) The premises and conclusions of π become the boundary nodes of $c(\pi)$; the premises form $\mathrm{dom}\, c(\pi)$ and the conclusions $\mathrm{cod}\, c(\pi)$. They are labelled by the edges formulae and signed according to whether the atom is positive or negative.

(ii) For all edges e labelled by a negative literal A^*, reverse e's direction, and change its labelling to A. This guarantees that negatively signed nodes in the codomain have incoming edges, and vice versa.

(iii) Erase every unit and counit node, merging their incident edges, which are now pointing in the same direction.

(iv) The remaining links of π must all be axioms links. These become the internal nodes of $c(\pi)$. At each node x, the ordering on $\mathrm{in}(x)$ and $\mathrm{out}(x)$ is simply that of the components of the domain and codomain of the arrow (in $\mathcal{A}$) which labels that node.

Lemma 90 guarantees that $c(n)$ really is a circuit. There is a dual procedure, taking a circuit $f : \bigotimes_i A_i \to \bigotimes_j B_j$ to a normal atomic proof-net.

(i) The nodes in $\mathrm{dom}\, f$ become premises; those of $\mathrm{cod}\, f$, conclusions.

(ii) If e is an edge, labelled by A, going from some node n to a premise p, replace e with a counit-link whose incoming edges are from e and p, labelled by A and A^* respectively.

(iii) If e is an edge, labelled by B, going to some node n from a conclusion c, replace e with a unit-link whose outgoing edges go to e and c and are labelled by B and B^* respectively.

(iv) The interior nodes of f become axiom links, each determined by the label on the corresponding node.

This defines a proof-net $p(f) : A_1, \ldots, A_n \vdash B_1, \ldots B_m$, which by Proposition 83 is normal. The two procedures are mutually inverse, which leads to the following characterisation result.

Definition 91 *Let X be formula; an* additive path *for X is a map which assigns a boolean value to each occurrence of $\oplus$ in X.*

Given an additive path b we can define a purely multiplicative formula $X_{(b)}$ by replacing each subformula $Y \oplus Z$ with Y if b assigns 0 to this $\oplus$ and Z if b assigns 1.

Theorem 92 *Let π be a normal proof-slice; then π is completely determined by its type, an additive path for its domain and codomain, and a $\mathcal{P}$-labellable circuit.*

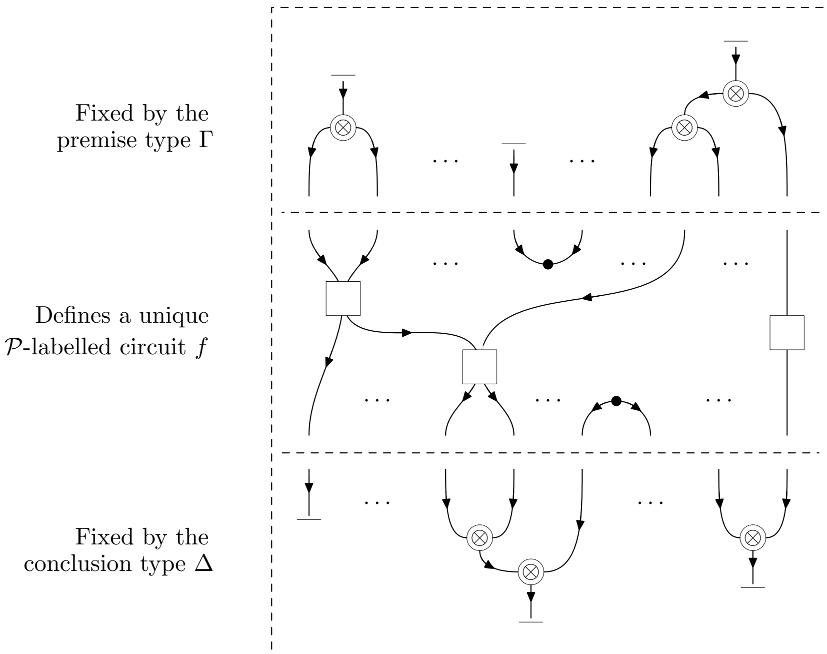


Fig. 1.7. Normal Proof-net decomposition

Proof Suppose that π has type $X \vdash Y$. Given a formula X, and an additive path b, let $\langle X, b \rangle$ be the list of literals produced by replacing every occurrence of $\otimes$ in $X_{(b)}$ by a comma. By Lemma 80, π can be decomposed into three layers: on top $\pi_{X,b}$ of type $X \vdash \langle X, b \rangle$ consisting only of cotensor, coplus, and costar links; the middle $\pi^- : X \vdash \langle X, b \rangle \vdash \langle Y, b' \rangle$ which is both normal, reduced, and atomic; and at the bottom $\pi_{Y,b'} : \langle Y, b' \vdash Y$ consisting only of tensors, pluses and stars. The layers π_Γ and π_Δ are uniquely determined by X, b and Y, b', while π^- is uniquely determined by the circuit $c(\pi^-)$. □

Corollary 93 *A normal reduced atomic slice is completely determined by a $\mathcal{P}$-labelled circuit f.*

Justified by the corollary we write $\pi \sim f$ for any normal reduced atomic slice π. The required functors F and U are now easily defined.

For each A of $\mathrm{PN}(\mathcal{P})$, let UA be the positively signed singleton, labelled by A; then define $U(A^*) = (UA)^*$ and $U(X \otimes Y) = UX \otimes UY$. Let $\pi \xrightarrow[\beta]{} \nu \sim f$ where ν is normal; then define $U\pi = f$.

To map **Circc**$(\mathcal{P})$ into $\mathrm{PN}(\mathcal{P})$, let f be a circuit; then let Ff be the proof-net obtained from $p(f)$ by adding tensor links to all the conclusions (bracketed to the left) and, similarly, cotensors to all the premises.

Theorem 94 *The 4-tuple* $(\textbf{\textit{Circ}}(\mathcal{P}), \mathrm{PN}(\mathcal{P})_N, F, U)$ *is an equivalence of categories.*

Proof Obviously, from the construction of U and F, we have $UF = \mathrm{Id}$. On the other hand, a proof-net $\pi : X \to Y$ only differs from $FU\pi : FUX \to FUY$ by the associativity of the tensor, hence $\mathrm{Id} \cong FU$. □

This theorem establishes that the matrix elements of a proof-net π in $\mathrm{PN}(\mathcal{P})$ are nothing more than formal sums of circuits over $\mathcal{P}$; i.e. element of the free compact closed category generated by $\mathcal{P}$. Hence we have the main result:

Theorem 95 *The category of proof-nets* $\mathrm{PN}(\mathcal{P})$ *is the free compact closed category with biproducts generated by the compact symmetric polycategory* $\mathcal{P}$*.*

1.5 Conclusions

To recap: we sketched how key parts of quantum mechanics can be formalised in the language of compact closed categories and biproducts; we demonstrated how to represent quantum processes as proof-nets, and showed that normalisation of such proof-nets allows some of the behaviour of the corresponding processes to be simulated.

We introduced the formal syntax of tensor-sum logic, and its proof-net notation. We showed that proof-nets are strongly normalising, and characterised the normal forms. Finally we proved the main theorem: that the category of proof-nets is exactly the free compact closed category with biproducts generated by the polycategory from which its axioms are drawn. This result can be viewed as a coherence theorem for compact closed categories with biproducts, in the style of Kelly and Laplaza's classic result for compact closed categories [KL80].

To return to our starting point, tensor-sum logic is almost an orthogonal theory to Birkhoff-von Neumann quantum logic. Tensor-sum logic is entirely preoccupied with the areas that quantum logic neglects: compoundness, interaction, and control. However, as the main theorem shows, we abdicate all responsibility for the internal structure of our quantum systems. Since our arrows are characterised by normal proof-nets, they are nothing more than type constructors wrapped around the generators: the fine structure must be described by an equational theory of the generators. We can view this as a strength: the logic is extremely general and could be easily applied to situations other than quantum computing. On the other hand, we suffer strong limitations on how much of quantum mechanics can be formulated in this setting without adjoining ad hoc rules to account for the particular situations we are modelling.

In a sense, this work is the end of the road for those "logical" approaches to quantum mechanics deriving from linear logic†. Already the dividing line substructural logic and algebra is thin, and what we have shown here is that, while proof-theoretic tools may suffice for the coarse business of putting together systems and pulling them apart again, the true quantum structure is living in the (poly)category of generators, and more subtle algebraic tools are needed to tease out the details. In particular the importance of spectra in quantum mechanics weighs against any approach based on natural transformations. Recent work [CPV08, CPP08, CD08] provides a categorical account of observables which is essentially algebraic. Fittingly, such theories have graphical representations which allow them to slot into the proof-net framework as generators. In that case combining the two systems would yield a well behaved two-level system of types and terms suitable for representing quantum processes under classical control.

Bibliography

[ABP99] S. Abramsky, R. Blute, and P. Panangaden. Nuclear and trace ideals in tensored *-categories. *Journal of Pure and Applied Algebra*, 143:3–47, 1999.

[AC04] S. Abramsky and B. Coecke. A categorical semantics of quantum protocols. In *Proceedings of the 19th Annual IEEE Symposium on Logic in Computer Science: LICS 2004*, pages 415–425. IEEE Computer Society, 2004.

[AD06] S. Abramsky and R. Duncan. A categorical quantum logic. *Mathematical Structures in Computer Science*, 16(3):469–489, 2006. Special Issue for the Proceedings of QPL 2004.

[AGN95] S. Abramsky, S. Gay, and R. Nagarajan. Specification structures and propositions-as-types for concurrency. In F. Moller and G. Birtwistle, editors, *Logics for Concurrency: Structure vs Automata — Proceedings*

† Those approaches deriving from topos theory, for example [DI08, HLS09], are a different matter entirely.

of the Banff Higher Order Workshop. Springer-Verlag Lecture Notes in Computer Science, 1995.

[AGN96] S. Abramsky, S. Gay, and R. Nagarajan. Interaction categories and the foundations of typed concurrent programming. In M Broy, editor, *Proceedings of the 1994 Marktoberdorf Summer School on Deductive Program Design*, pages 35–113. Springer-Verlag, 1996.

[AHS02] S. Abramsky, E. Haghverdi, and P. Scott. Geometry of interaction and linear combinatory algebras. *Mathematical Structures in Computer Science*, 12:625–665, 2002.

[AJ94] S. Abramsky and R. Jagadeesan. New foundations for the geometry of interaction. *Information and Computation*, 111(1):53–119, 1994. Conference version appeared in LiCS '92.

[Bar79] M. Barr. $*$-*Autonomous Categories*, volume 752 of *Lecture Notes in Mathematics*. SpringerVerlag, 1979.

[Bar91] M. Barr. *-autonomous categories and linear logic. *Mathematical Structures in Computer Science*, 1:159–178, 1991.

[BBC$^+$93] C. Bennett, G. Brassard, C. Crepeau, R. Jozsa, A. Peres, and W. Wootters. Teleporting an unknown quantum state via dual classical and EPR channels. *Phys. Rev. Lett.*, pages 1895–1899, 1993.

[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*, 113:229–296, 1996.

[BD95] J. Baez and J. Dolan. Higher-dimensional algebra and topological quantum field theory. *J.Math.Phys*, 36:6073–6105, 1995.

[BvN36] G. Birkhoff and J. von Neumann. The logic of quantum mechanics. *Annals of Mathematics*, 37(4):823–843, October 1936.

[CD08] B. Coecke and R. Duncan. Interacting quantum observables. In L. Aceto, I. Damgård, L. A. Goldberg, M. M. Halldórsson, A Ingólfsdóttir, and I. Walukiewic, editors, *Automata, Languages and Programming, 35th International Colloquium, ICALP 2008, Reykjavik, Iceland, July 7-11, 2008, Proceedings, Part II*, volume 5126 of *Lecture Notes in Computer Science*, pages 298–310. Springer, 2008.

[CF58] H. B. Curry and R. Feys. *Combinatory Logic I.* North-Holland Publishing, 1958.

[Coe05a] B. Coecke. De-linearizing linearity: Projective quantum axiomatics from strong compact closure. In *Proceedings of the 3rd International Workshop on Quantum Programming Languages*, 2005.

[Coe05b] B. Coecke. Kindergarten quantum mechanics. Lecture Notes, 2005.

[CPP08] B. Coecke, E. O. Paquette, and S. Perdrix. Bases in diagrammatic quantum protocols. In *Proceedings of Mathematical Foundations of Programming Semantics XXIV*. 2008.

[CPV08] B. Coecke, D. Pavlovic, and J. Vicary. A new description of orthogonal bases. *Math. Structures in Comp. Sci.*, 2008. 13pp, to appear, arxiv.org/abs/0810.0812.

[CS97] R. Cockett and R.A.G. Seely. Weakly distributive categories. *Journal of Pure and Applied Algebra*, 113(2):229–296, 1997.

[Del91] P. Deligne. Catégories tannakiennes. In *Grothendieck Festschrift*, volume 2, pages 111–194. Birkhauser, 1991.

[DI08] A. Doering and C. J. Isham. A topos foundation for theories of physics: I. formal languages for physics. *J. Math. Phys*, 49, 2008.

[DR89] V. Danos and L. Regnier. The structure of multiplicatives. *Arch. Math.*

Logic, 28(3):181–203, 1989.

[Dun04] R. Duncan. Believe it or not, bell states are a model of multiplicative linear logic. Technical Report PRG-RR-04-18, Oxford University Computing Laboratory, 2004.

[Dun06] R Duncan. *Types for Quantum Computation*. PhD thesis, Oxford University, 2006.

[Gen35] G. Gentzen. Untersuchungen über das schliessen. In M.E Szabo, editor, *The Collected Papers of Gerhard Gentzen*. North-Holland Publishing, 1935.

[Gir87a] J.-Y. Girard. Linear logic. *Theoretical Computer Science*, 50(1), 1987.

[Gir87b] J.-Y. Girard. Multiplicatives. In G. Lolli, editor, *Logic and Computer Science: New Trends and Applcations*, pages 11–37. Rendiconti del Seminario Mathematico dell'Universita e Politecnico di Torino, 1987.

[Gir96] J.-Y. Girard. Proof-nets: the parallel syntax for proof theory. In M. Dekker, editor, *Logic and Algebra*. 1996.

[GSS91] J.-Y. Girard, A. Scedrov, and P. J. Scott. Normal forms and cut-free proofs as natural transformations. In Y. N. Moschovakis, editor, *Proceedings Workshop Logic from Computer Science, Berkeley, CA, USA, 13–17 Nov 1989*, volume 21, pages 217–241. Springer-Verlag, New York, 1991.

[HLS09] C. Heunen, K. Landsman, and B. Spitters. A topos for algebraic quantum theory. *Communications in mathematical physics*, 2009.

[Hou08] R. Houston. Finite products are biproducts in a compact closed category. *Journal of Pure and Applied Algebra*, 212(2):394–400, June 2008.

[How80] W. A. Howard. The formulæ-as-types notion of construction. In J. R. Hindley and J. P. Seldin, editors, *To H. B. Curry, Essays on Combinatory Logic, Lambda Calculus and Formalism*, pages 479–490. Academic Press, 1980.

[HS03] M. Hyland and A. Schalk. Gluing and orthogonality for models of linear logic. *Theoretical Computer Science*, 294:183–231, 2003.

[HvG03] D. Hughes and R. van Glabbeek. Proof nets for unit-free multiplicative-additive linear logic. In *Proc. Logic in Computer Science 2003*. IEEE, 2003.

[Joy77] A Joyal. Remarques sur la théories des jeux à deux personnes. *Gazette des sciences mathématiques du Québec*, 1(4), 1977.

[JS91] A. Joyal and R. Street. The geometry of tensor categories i. *Advances in Mathematics*, 88:55–113, 1991.

[JS93] A. Joyal and R. Street. Braided tensor categories. *Advances in Mathematics*, 102:20–78, 1993.

[JSV96] A. Joyal, R. Street, and D. Verity. Traced monoidal categories. *Math. Proc. Camb. Phil. Soc.*, 119:447–468, 1996.

[Kel72] G.M. Kelly. Many-variable functorial calculus I. volume 281 of *Lecture Notes in Mathematics*, pages 66–105. Springer, 1972.

[KL80] G.M. Kelly and M.L. Laplaza. Coherence for compact closed categories. *Journal of Pure and Applied Algebra*, 19:193–213, 1980.

[Lam68] J. Lambek. Deductive systems and categories I. syntactic calculus and residuated categories. *Mathematical Systems Theory*, 2(4):287–318, 1968.

[Lam69] J. Lambek. Deductive systems and categories (ii). In R. J. Hilton, editor, *Category Theory, Homology Theory and their Applications I*, volume 87 of *Lecture Notes in Mathematics*, pages 76–122. Springer, 1969.

[Loa94] R. Loader. *Models of Lambda Calculi and Linear Logic: Structural,* *Equational and Proof-theoretic Characterisations*. PhD thesis, St Hugh's College, Oxford, 1994.

[LS86] J. Lambek and P. J. Scott. *Introduction to higher order categorical logic*, volume 7 of *Cambridge studies in advanced mathematics*. Cambridge University Press, 1986.

[LTdF04] O. Laurent and L. Tortora de Falco. Slicing polarized additive normalisation. In T. Ehrhard, J.-Y. Girard, P. Ruet, and P. Scott, editors, *Linear Logic in Computer Science*, volume 316 of *London Mathematical Society Lecture Notes*, pages 247–282. Cambridge University Press, 2004.

[Mit65] B. Mitchell. *Theory of Categories*. Academic Press, 1965.

[ML97] S. Mac Lane. *Categories for the Working Mathematician (2nd Ed.)*. Springer-Verlag, 1997.

[Pen71] R. Penrose. Applications of negative dimensional tensors. In *Combinatorial Mathematics and its Applications*, pages 221–244. Academic Press, 1971.

[Pra65] D. Prawitz. *Natural Deduction. A Proof-Theoretic Study*. Almqvist & Wiksell, 1965.

[Sel04] P. Selinger. Towards a quantum programming language. *Mathematical Structures in Computer Science*, 14(4):527–586, 2004.

[Sel05] P. Selinger. Dagger compact closed categories and completely positive maps. In *Proceedings of the 3rd International Workshop on Quantum Programming Languages*, 2005.

[Sme01] S. Smets. *The Logic of Physical Properties in Static and Dynamic Perspective*. PhD thesis, Vrije Universiteit Brussel, May 2001.

[Sol87] S. V. Soloviev. On natural transformations of distinguished functors and their superpositions in certain closed categories. *Journal of Pure and Applied Algebra*, 47, 1987.

[Sza75] M.E Szabo. Polycategories. *Comm. Algebra*, 3:663–689, 1975.