Programming Research Group

BELIEVE IT OR NOT, BELL STATES ARE A MODEL OF MULTIPLICATIVE LINEAR LOGIC

Ross Duncan[1]

PRG-RR-04-18

Oxford University Computing Laboratory
Wolfson Building, Parks Road, Oxford OX1 3QD

[1]ross.duncan@comlab.ox.ac.uk

Abstract

By considering an explicit model of **MLL** over a compact closed category we show that cut-free **MLL** proof-nets (without units) may be realised as entangled quantum states and cut-elimination may be implemented as entanglement exchange.

1 Introduction

The phenomenon of entanglement is at the heart of quantum computation. It is implicated in the speed up of quantum algorithms [12], and exploited directly in quantum protocols [8]. The quantum teleportation protocol is a spectacular demonstration that entangled systems can transmit information. The transmission and manipulation of information via entanglement has been extensively explored by Coecke [10].

However, despite its seemingly central importance, most current formalisms for quantum programming [18, 27] do not represent entanglement explicitly and treat the quantum register as a mysterious black box. We aim to explicate information flow within the quantum realm by imposing a type system upon quantum states which captures entanglement between subsystems.

Abramsky and Coecke [1] have established that finite dimensional quantum mechanics can be carried out in the framework of compact closed categories with biproducts. A great many features of quantum theory arise very naturally in this setting, however at the level of types information flow is not exposed. In a compact closed category there is no distinction between the tensor and the internal hom, therefore one cannot separate the entangled states – that is, those which permit information flow – from the separable states, which do not. Achieving such a separation is the principle goal of this paper.

Starting from the compact closed subsystem of this model we explicitly construct a model of classical multiplicative linear logic (MLL) [13, 11] encoding the axiom links of a proof-net as maximally entangled pairs of quanta – Bell states in the case of qubits. Cut elimination is realised by entanglement swapping. The linear connectives $\invamp$ and $\otimes$ serve to classify systems as entangled or separable respectively.

This linear typing can be considered a specification on top of the compact closed structure, somewhat in the style of [3]. Looking from the top down, the categorical model immediately yields a faithful quantum representation of MLL proof-nets, or equivalently linear λ-terms. We specialise this to give a compilation process from proof-nets to quantum circuits.

The general paradigm is that of measurement based quantum computation [16, 5]. The approach lies midway between the one-way quantum computer of Raussendorf and Briegel [25] and the projective-measurements-only scheme introduced by Nielsen [24], in that computation is performed via projective measurements, starting from an initial entangled state. However, the initial state is simple: a collection of entangled pairs rather than a cluster state, and all measurements are on bipartite states. Further, once a pair has been measured, it is discarded.

In this presentation we have made life easy by ignoring the non-determinism of quantum measurements; the flip side is that we have also ignored the classical information so produced. In the framework of compact closed categories with biproducts [1] classical branching is represented by the biproduct structure; having added extra type information to the multiplicative aspect of this framework, how to best use the additive part is left for future work.

2 Categorical Preliminaries

In the following we shall assume that all monoidal categories are *strict*; that is the following equalities hold:

$$(A \otimes B) \otimes C = A \otimes (B \otimes C) \qquad\qquad I \otimes A = A = A \otimes I.$$

Every non-strict monoidal category is equivalent to a strict one [20] so there is no danger in this identification.

Definition 1. A symmetric monoidal category $(\mathcal{C}, \otimes, I)$ is *compact closed* [17] if for each object A we have the following data:

- a chosen adjoint object A^*
- a *unit morphism* $\eta_A : I \longrightarrow A^* \otimes A$
- a *co-unit morphism* $\epsilon_A : A \otimes A^* \longrightarrow I$

satisfying

$$\begin{tikzcd} A \arrow[rr, "\mathrm{id}_A"] \arrow[dr, "\mathrm{id}_A \otimes \eta_A"'] & & A \\ & A \otimes A^* \otimes A \arrow[ur, "\epsilon_A \otimes \mathrm{id}_A"'] & \end{tikzcd} \tag{1}$$

and the dual diagram for A^*.

Some useful facts about compact closed categories:

- The map $A \mapsto A^*$ extends to a contravariant functor.
- $\mathcal{C}(A, B) \cong A^* \otimes B$
- To every $f : A \longrightarrow B$ there is a *name* $\ulcorner f \urcorner : I \longrightarrow A^* \otimes B$ and a *coname* $\llcorner f \lrcorner : A \otimes B^* \longrightarrow I$.

The basic equations pertinent to names and conames can be found in [1]. We will make heavy use of the following lemma [17].

Lemma 2 (Decomposition). *For any objects A and B, the unit and co-unit on $A \otimes B$ may be decomposed into those for A and B. Precisely:*

$$\begin{tikzcd} (A \otimes B) \otimes (A \otimes B)^* \arrow[rr, "\cong"] \arrow[dr, "\epsilon_{A \otimes B}"'] & & (A \otimes A^*) \otimes (B \otimes B^*) \arrow[dl, "\epsilon_A \otimes \epsilon_B"] \\ & I & \end{tikzcd}$$

and the dual diagram for $\eta_{A \otimes B}$.

It is well known that compact closed categories are degenerate models of multiplicative linear logic, but the *double gluing* construction [19, 15, 28] can be used to construct a $*$-autonomous category [7] from a compact closed one, and hence arrive at model of MLL which separates the tensor and par.

Definition 3. Suppose $\mathcal{C}$ is compact closed, then the *Double Gluing on* $\mathcal{C}$, $G\mathcal{C}$, is the category whose objects are triples $\mathbf{A} = (A, U, X)$ where

- A is an object of $\mathcal{C}$
- $U \subseteq \mathcal{C}(I, A)$ is the set of *points* of $\mathbf{A}$.
- $X \subseteq \mathcal{C}(I, A^*)$ is the set of *co-points* of $\mathbf{A}$.

and whose arrows $\mathbf{f} : (A, U, X) \longrightarrow (B, V, Y)$ are arrows of $\mathcal{C}$ $f : A \longrightarrow B$ such that:

- $\forall u \in U \; f \circ u \in V$,
- $\forall y \in Y \; f^* \circ y \in X$.

Theorem 4. *Given* $\boldsymbol{A} = (A, U, X)$ *and* $\boldsymbol{B} = (B, V, Y)$*,* $G\mathcal{C}$ *as above is* $*$*-autonomous with the following structure:*

- $\boldsymbol{A}^\perp = (A^*, X, U)$
- $\boldsymbol{A} \otimes \boldsymbol{B} = (A \otimes B, W, Z)$ *where*

$$
\begin{array}{rcl}
W & = & \{I \cong I \otimes I \xrightarrow{u \otimes v} A \otimes B \mid u \in U, v \in V\}, \\
Z & = & \{I \xrightarrow{\ulcorner f \urcorner} A^* \otimes B^* \mid \forall u \in U \;\; I \xrightarrow{u} A \xrightarrow{f} B^* \in Y, \\
 & & \qquad \forall v \in V \;\; A \xrightarrow{f} B^* \xrightarrow{v} I^* \in X\}
\end{array}
$$

- $\boldsymbol{A} \parr \boldsymbol{B} = (\boldsymbol{A}^\perp \otimes \boldsymbol{B}^\perp)^\perp = (A \otimes B, Z', W')$ *where*

$$
\begin{array}{rcl}
Z' & = & \{I \xrightarrow{\ulcorner g \urcorner} A \otimes B \mid \forall x \in X \;\; I \xrightarrow{x} A^* \xrightarrow{g} B \in V, \\
 & & \qquad \forall y \in Y \;\; A^* \xrightarrow{f} B \xrightarrow{y} I^* \in U\} \\
W' & = & \{I \cong I \otimes I \xrightarrow{x \otimes y} A^* \otimes B^* \mid x \in X, y \in Y\}.
\end{array}
$$

3 Multiplicative Linear Logic

In this section we work in an arbitrary strict compact closed category $\mathcal{C}$ and construct a faithful model of MLL. Every compact closed category is also $*$-autonomous and hence yields a categorical model for MLL automatically [26]. This expository material is included to expose the exact categorical machinery needed so that it may be readily realised in concrete setting. Proofs of the results of this section are somewhat hard to find in the literature, so are included in an appendix.

$$\frac{}{\vdash A^{\perp}, A}\text{(identity)} \qquad \frac{\vdash \Gamma, A \quad \vdash A^{\perp}, \Delta}{\vdash \Gamma, \Delta}\text{(cut)} \qquad \frac{\vdash \Gamma, A, B, \Delta}{\vdash \Gamma, B, A, \Delta}\text{(exchange)}$$

$$\frac{\vdash \Gamma, A \qquad \vdash B, \Delta}{\vdash \Gamma, A \otimes B, \Delta}\text{(times)} \qquad \frac{\vdash \Gamma, A, B, \Delta}{\vdash \Gamma, A \invamp B, \Delta}\text{(par)}$$

Figure 1: Sequent rules for multiplicative linear logic

3.1 Interpreting Formulae

To each of our propositional letters $A, B, C, \ldots$ we assign corresponding objects $A, B, C, \ldots$ of $\mathcal{C}$. Linear negation $A^{\perp}$ is interpreted as the dual object A^*. Tensor is interpreted by the tensor of the category; since $\mathcal{C}$ is compact closed we have the isomorphism $\mathcal{C}[A, B] \cong A^* \otimes B$, so par will also be interpreted by the tensor.

3.2 Interpreting Cut-Free Proofs

The representation of a proof π of the sequent $\vdash \Gamma$ is a morphism in $\mathcal{C}$, $[\![\pi]\!] : I \longrightarrow \invamp\Gamma$. The morphism $[\![\pi]\!]$ will be built up inductively according to the construction of the proof π.

Since we have an explicit exchange rule, we consider $\vdash \Gamma, \Delta$ and $\vdash \Delta, \Gamma$ to be distinct sequents so a proof of one is not, in general, a proof of the other.

3.2.1 Axioms

Given the proof

$$\pi = \overline{\vdash A^{\perp}, A}$$

we define

$$[\![\pi]\!] = \eta_A : I \longrightarrow A^* \otimes A.$$

Justified by lemma 2, we only consider axioms of atomic formulae.

3.2.2 Times

We interpret times simply by taking the tensor product of two arrows. Suppose π is the proof

$$\pi = \frac{\begin{matrix}\pi_1 \\ \vdots \\ \vdash \Gamma, A\end{matrix} \qquad \begin{matrix}\pi_2 \\ \vdots \\ \vdash B, \Delta\end{matrix}}{\vdash \Gamma, A \otimes B, \Delta}$$

we have $[\![\pi_1]\!] : I \longrightarrow \Gamma \otimes A$ and $[\![\pi_2]\!] : I \longrightarrow B \otimes \Delta$. Define $[\![\pi]\!] : I \longrightarrow \Gamma \otimes (A \otimes B) \otimes \Delta$ by

$$[\![\pi]\!] = [\![\pi_1]\!] \otimes [\![\pi_2]\!]$$

which has the required type due strictness.

3.2.3 Par

We interpret par as a no-op. Given π arising via a par rule from a proof π':

$$\pi = \dfrac{\begin{matrix}\pi' \\ \vdots \\ \vdash \Gamma, A, B, \Delta\end{matrix}}{\vdash \Gamma, A \mathbin{⅋} B, \Delta}$$

we have $[\![\pi']\!] : I \longrightarrow \Gamma \otimes A \otimes B \otimes \Delta$; just define

$$[\![\pi]\!] = [\![\pi']\!].$$

Par internalises the comma of the sequent presentation, and since this is already interpreted as the tensor product, there is no requirement to do more.

3.2.4 Exchange

If $[\![\pi]\!]$ arises from $[\![\pi']\!]$ by an exchange of A and B then we have

$$\pi' : I \longrightarrow \Gamma \otimes A \otimes B \otimes \Delta.$$

Define $[\![\pi]\!]$ by

$$[\![\pi]\!] = [\![\pi']\!]; (\mathrm{id}_\Gamma \otimes s \otimes \mathrm{id}_\Delta)$$

where s is the canonical isomorphism $A \otimes B \longrightarrow B \otimes A$.

3.3 Faithfulness of the Interpretation

A cut-free MLL proof net without units can be specified entirely by its type and a permutation describing how the axiom links are arranged [14]. We will show a similar result in this framework, characterising the interpretation of a proof π as a permutation on its atoms. It will turn out that $[\![\pi]\!]$ is equivalent to the proof net version of π.

Lemma 5 (Separation). *If π is a cut-free proof of $\vdash \Gamma$ which contains the atoms*[2] *$A_1^\perp, A_1, \ldots, A_n^\perp, A_n$ then $[\![\pi]\!]$ has the form*

$$I \xrightarrow{\eta_{A_1} \otimes \cdots \otimes \eta_{A_n}} (A_1^* \otimes A_1) \otimes \cdots \otimes (A_n^* \otimes A_n) \xrightarrow{\quad s \quad} \Gamma$$

where s is a permutation on the components of the tensor.

[2]To avoid falling over language, we assume that all atoms occur only once.

Note that s is unique up to the initial choice of the ordering of the atoms $A_1, \ldots, A_n$. In the following some canonical ordering will be taken for granted; call s the *induced permutation* of $[\![\pi]\!]$.

Although we have not described it this model interprets the mix rule:

$$\frac{\vdash \Gamma \qquad \vdash \Delta}{\vdash \Gamma, \Delta}(\mathsf{mix})$$

The rule essentially states that tensor is a special case of par – everywhere that $A \otimes B$ is derivable, so is $A \parr B$. Mix is not a derivable rule of linear logic; it corresponds to permitting disconnected proof nets. We can view the arrow $\eta_{A_1} \otimes \cdots \otimes \eta_{A_n}$ as the mix of axioms

$$\overline{\vdash A_i^{\perp}, A_i}$$

giving a proof of $\vdash A_1^{\perp}, A_1, \ldots, A_n^{\perp}, A_n$ corresponding to the permutation

$$(1\ \ 2) \cdots (2n-1\ \ 2n)\ .$$

Denote this permutation ϱ_n.

As noted in [14, 4], any cut-free MLL proof-net π, with only atomic formulae as axioms, can be specified entirely by $\pi \cong (\Gamma; \sigma)$ where Γ is the sequent proven by π and σ is a type-inverting permutation which is fixpoint-free and a product of disjoint transpositions. The type alone is sufficient to construct the bottom part of the proof net, while the transpositions of σ correspond to the axiom links. Since each sequent proof gives rise to a unique proof net there is no danger in writing $\pi \cong (\Gamma; \sigma)$ when π is a proof rather than a proof net.

Lemma 6 (Characterisation). *Let π be a cut free MLL proof containing n axiom links, such that $\pi \cong (\Gamma; \sigma)$ as described above. If s is the permutation induced by $[\![\pi]\!]$ then $\sigma = s^{-1}\varrho_n s$.*

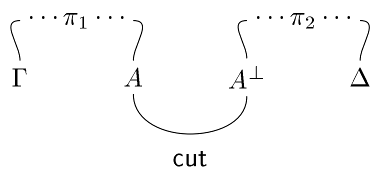
Theorem 7 (Faithfulness). *If π_1, π_2 are distinct cut-free proofs of the sequent $\vdash \Gamma$ then $[\![\pi_1]\!] \neq [\![\pi_2]\!]$.*

Proof. Let $\pi_1 \cong (\Gamma; \sigma_1)$ and $\pi_2 \cong (\Gamma; \sigma_2)$. The equation $[\![\pi_1]\!] = [\![\pi_2]\!]$ holds if and only if the induced permutations s_1, s_2 are equal. In this case $s_1^{-1}\varrho_n s_1 = s_2^{-1}\varrho_n s_2$ which by the previous lemma is equivalent to $\sigma_1 = \sigma_2$. $\square$

Remark. Note that if we extend the syntax with non-logical axioms $f_i : A \multimap B$ we can still faithfully represent these using names $\ulcorner f_i \urcorner : I \longrightarrow A^* \otimes B$ so long as $\mathcal{C}$ has distinct arrows $A \longrightarrow B$ for each f_i. In particular one can always freely construct a suitable category; see [2].

Since the model is faithful with respect to proof-nets rather than sequents we will prefer that syntax from here on.

3.4 Cut



To interpret a proof containing a cut, such as the one above, we must interpret the cut-link in such a way as to erase the cut-formulae $A^\perp$ and A. To do so we use the co-unit ϵ_A. Let π be the proof above, then construct $[\![\pi]\!] : I \longrightarrow \Gamma \otimes \Delta$ by

$$[\![\pi]\!] = ([\![\pi_1]\!] \otimes [\![\pi_2]\!]); (\mathrm{id}_\Gamma \otimes \epsilon_A \otimes \mathrm{id}_\Delta).$$

This completes the definition of the model; proof of the key theorem is in the appendix.

Theorem 8 (Soundness). *Given MLL proof nets π and π' such that π reduces to π' by eliminating cuts then $[\![\pi]\!] = [\![\pi']\!]$.*

4 Multiplicative Types and Entanglement: Interpreting the Model

In this section we discuss the interpretation of multiplicative linear types in the framework of quantum computing, and sketch how the model described above may be realised in this setting. A concrete implementation is postponed until the following section.

In order to stay completely within the framework of compact closed categories we will neglect the probabilistic aspect of quantum measurement and instead think of measurements simply as deterministic projections onto a chosen state. If one imagines a quantum system subject to measurement as some kind of repeated experiment, we are effectively selecting only the "good" runs of the experiment. To represent the entire branching structure one must consider categories with biproducts. This approach is pursued in [1]; for present purposes the extra structure of biproducts is unnecessary.

The most important example of a compact closed category for quantum computation is $\mathbf{Vect}_{\mathbb{C}}^{\mathrm{fd}}$, the category whose objects are finite-dimensional complex vector spaces and whose arrows are linear maps. $\mathbf{Vect}_{\mathbb{C}}^{\mathrm{fd}}$ is enriched over itself so the usual tensor product of vector spaces gives the symmetric monoidal structure with $I = \mathbb{C}$. Let $A^* = L(A, \mathbb{C})$, the space of linear functionals on A. For any basis $\{a_i\}_i$ of A, let $\{\overline{a_i}\}_i$ be the corresponding basis of A^*, then the unit and co-unit are given by:

$$\eta_A : z \mapsto z \cdot \sum_i \overline{a_i} \otimes a_i$$
$$\epsilon_A : \psi \mapsto \langle \sum_i a_i \otimes \overline{a_i} \mid \psi \rangle.$$

It is easily verified that these definitions satisfy the required equations.

We can also give a second, more degenerate, compact closed structure on this category by taking $(-)^*$ to be the identity on objects. This will prove useful for our second example. Consider the free compact closed category on one object Q. We can embed this category into $\mathbf{Vect}_{\mathbb{C}}^{\mathrm{fd}}$ (with respect to the degenerate * structure) by mapping $Q \mapsto \mathbb{C}^2$ and $Q^* \mapsto \mathbb{C}^2$. Call the sub-category of $\mathbf{Vect}_{\mathbb{C}}^{\mathrm{fd}}$ so obtained **Qubit**, the category of qubits.

Remark. Exactly *which* subcategory of $\mathbf{Vect}_{\mathbb{C}}^{\mathrm{fd}}$ to take — that is, which arrows to include — is an interesting question. Since we are not concerned here with the issue of junk in the model we will be deliberately coy on the subject. In almost every case the results require only the arrows of free category; in the discussion of gluing we require the hom-set (I, Q) to be non-empty which is not the case if **Qubit** is simply the image of the free category.

If we write the unit and co-unit maps of Q in the usual notation of quantum computing then we have

$$\eta_Q : 1 \mapsto |00\rangle + |11\rangle$$
$$\epsilon_Q : \psi \mapsto \langle 00 \mid \psi\rangle + \langle 11 \mid \psi\rangle.$$

Hence we can see that the unit and co-unit in this category represent respectively the production of, and projection onto, the Bell state Φ^+; the coherence diagram (1) may be therefore be read as expressing the correctness of a particular path of the quantum teleportation protocol[3][8].

It is easy to see that for every object A all points in the image of η_A share a precisely analogous entanglement and, by virtue of diagram (1), that they have a similar capacity for information transmission. Hence we shall call these states called *generalised Bell states*, whether in **Qubit** or $\mathbf{Vect}_{\mathbb{C}}^{\mathrm{fd}}$, and denote the generalised Bell state at type A by Φ_A

For the remainder of this section we operate in $\mathbf{Vect}_{\mathbb{C}}^{\mathrm{fd}}$; hence each type of MLL denotes a vector space in the model. The physical interpretation demands that we restrict our ground types to the state spaces of quantum systems. Translating the categorical model directly, both tensor and par will be realised by forming the appropriate compound state space.

Each MLL term denotes an arrow $[\![\pi]\!] : I \longrightarrow A$, and since in this category $I = \mathbb{C}$, every such arrow picks out a ray in the vector space A. Only vectors of unit length are considered physically meaningful, and these are only distinguishable up to a global complex phase. Therefore each one dimensional subspace of A contains only one quantum state. Hence the faithfulness of the categorical interpretation gives an injective mapping of cut-free MLL proof-nets to quantum states.

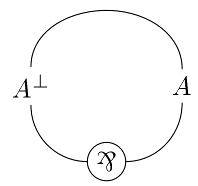
Considering proofs as programs, the intensional construction of $[\![\pi]\!]$ specifies a process to prepare the quantum state corresponding to the normal form of π. The required components are simply the physical realisation of the compact closed structure. Given physical systems implementing maps f and g their tensor product $f \otimes g$ is implemented simply by running the two systems in parallel. As was noted earlier, the unit morphism $\eta_A : I \longrightarrow A^* \otimes A$ corresponds to the creation of an entangled pair: the generalised Bell

[3] Note that since we have idealised away the non-determinism, there is no need for correction, and hence no need for classical communication.

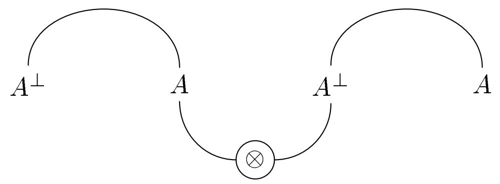
state Φ_A. To prepare Φ_A, we start with any state in $A^* \otimes A$ and apply a projection. Alternatively, if the initial state is known — say it is ϕ — then there exists a unitary transformation U such that $U\phi = \Phi_A$. Similarly the co-unit $\epsilon_A : A \otimes A^* \longrightarrow I$ at some ground type A is interpreted by a projection onto the entangled state Φ_{A^*}. The symmetry maps $s_{A,B} : A \otimes B \longrightarrow B \otimes A$ have natural interpretations in terms of rearranging the parts of the system. By our faithfulness result this machinery suffices to prepare the quantum states corresponding to any MLL proof net.

The states corresponding to proof nets form a restricted subset of possible states; we can use the MLL typing to characterise them. Recall that η_A was the categorical interpretation of an axiom link in a proof net. Hence the quantum implementation of an axiom link is an entangled pair. The separation lemma (Lemma 5) tells us that a cut-free net is nothing more than a collection of such entangled pairs, jumbled up according to the induced permutation. The pattern of entanglement between quanta exactly mirrors the arrangement of the axiom links.

Consider the simple proof net:



Since par receives no interpretation, this proof net is implemented by an entangled pair just as if it was a bare axiom link. Now consider the following proof-net:



The denotation of this term is the tensor product $\eta_A \otimes \eta_A$: that is two entangled pairs. Clearly the left and right sub-terms are not entangled with each other; indeed it is immediate from the model that every term of type $A \otimes B$ must consist of two totally distinct parts, hence their their quantum implementations must be *separable*. Conversely terms of type $A ⅋ B$ are realised by an *entanglement* between their subsystems. However neither the separable nor entangled states form a subspace, so there are no corresponding objects in the category. If we move to the glued version we can recover the separation.

Theorem 9. *Call a sequent* closed *if it contains no commas. Let π be a proof of some closed sequent $\vdash A$, with denotation $[\![\pi]\!]$ in some compact closed category $\mathcal{C}$. Then $[\![\pi]\!]$ is a point of $\boldsymbol{A}$ in the double glued category $G\mathcal{C}$.*

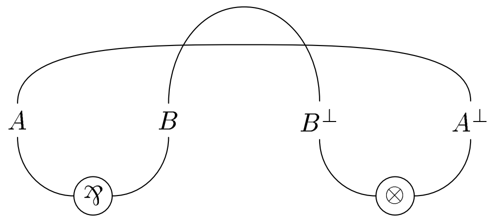
Proof sketch. For each A, η_A is a point of $\mathbf{A}^\perp ⅋ \mathbf{A}$; hence $\eta_{A_1} \otimes \cdots \otimes \eta_{A_n}$ is a point of $(\mathbf{A}_1^\perp ⅋ \mathbf{A}_1) \otimes \cdots \otimes (\mathbf{A}_n^\perp ⅋ \mathbf{A}_n)$. Let w_1, w_2 be proof-nets coding the weak distributivity maps

$$w_1 : A \otimes (B \parr C) \longrightarrow (A \otimes B) \parr C$$
$$w_2 : A \otimes (B \parr C) \longrightarrow B \parr (A \otimes C);$$

then $[\![w_1]\!], [\![w_2]\!]$ are points of the appropriate objects of $G\mathcal{C}$. Since any proof-net can be derived from these maps combined with a finite tensor of axioms [9] this suffices. $\square$

The $*$-autonomous structure of $G\mathcal{C}$ requires that each point of $\mathbf{A} \parr \mathbf{B}$ must name a map from A^* to B; in particular the only maps between atomic types types which are coded by proof-nets are identities. Hence any quantum realisation of a term of type $A \parr B$ consists of two subsystems sharing at least one maximally entangled pair. Similarly the definition of the points of $\mathbf{A} \otimes \mathbf{B}$ restrict to completely separable states.

Remark. The restriction to closed proof-nets is necessary to exclude examples such as



where there is clearly no entanglement between the A and B parts of the $A \parr B$ conclusion.

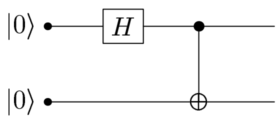
Finally we consider the quantum interpretation of the cut rule. It suffices to consider atomic cuts: decomposition will provide the rest. The co-unit ϵ_A is implemented by a projection onto the entangled pair Φ_A. The cut rule shares the same geometry as the times rule, so we can guarantee that the quanta which realise the cut-formulae are unentangled before the measurement. Further, by the separation lemma, each of these quanta is guaranteed to form one half of a maximally entangled pair. After the measurement, the measured quanta are left entangled and so are their erstwhile partners. This effect is called *entanglement swapping*[4].

5 Compiling Proof-nets to Quantum Circuits

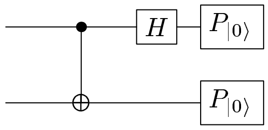
We now make the discussion above precise by considering the case of the category **Qubit**, and translating proof-nets into quantum circuits. We choose circuits principally for concreteness and notational convenience. Since the cut-elimination process is essentially equivalent to the entanglement swapping protocol there is no reason why the construction below could not be described by a multiparty protocol.

To implement the requisite morphisms as circuits we require only three components: Hadamard gates, CNOT gates and single qubit projections in the computational basis. For definition of these gates and the circuit notations see [23]. The only non-standard notation is $P_{|0\rangle}$ which represents the projection onto $|0\rangle$. The circuits implementing the

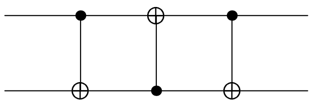
[4]In the usual account of this protocol, both entangled pairs are preserved.



$$\eta_Q : I \longrightarrow Q \otimes Q$$



$$\epsilon_Q : Q \otimes Q \longrightarrow I$$



$$s_{Q,Q} : Q \otimes Q \xrightarrow{\cong} Q \otimes Q$$

Figure 2: Quantum circuits for MLL normalisation

basic morphisms η_Q, ϵ_Q and $s_{Q,Q}$ are shown in figure 2. Aside from these basic arrows we also need to implement a general symmetry $s_{A,B}$ and a general unit $\eta_{A\otimes B}$ and co-unit $\epsilon_{A\otimes B}$. The following lemma will be useful.

Lemma 10 (Moore and Nilsson [22]). *Any permutation of of n qubits can be performed in 4 layers of CNOT gates with n ancillae, or in 6 layers with no ancillae.*

All compound types $A \otimes B$ are implemented as flat arrays of qubits; let the width of A be n qubits and the width of B be m qubits. The symmetry map $s_{A,B}$ can then be viewed as a permutation over the array indices. By lemma 10 two layers of single qubit swaps suffice.

The problem of computing the unit and co-unit can be broken down using the decomposition lemma. Let $Q_n = \bigotimes_{i=1}^{n} Q^{(i)}$, where the superscripts just distinguish occurrences

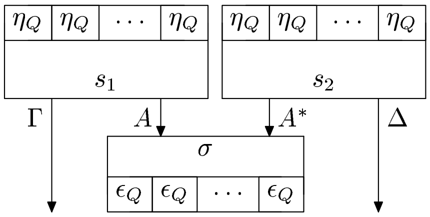


Figure 3: Circuit for a cut

of Q. Repeatedly applying the lemma until we reach the base type, we have

$$\eta_{Q_n} = \bigotimes_{i=1}^{n} \eta_{Q^{(i)}} ; \sigma_n$$

where σ_n is the permutation

$$\bigotimes_{i=0}^{n} (Q^{(i)} \otimes Q^{(i)}) \longrightarrow Q_n \otimes Q_n$$

Again, σ_n can be computed in two layers of swaps. The dual construction gives a similar constant depth circuit for ϵ_{Q_n}.

By the separation lemma (lemma 5) the denotation of a cut free proof-net has the form of many copies of η_Q in parallel, followed by the induced permutation s. Using lemma 10 again, we can prepare the quantum state corresponding to any cut free proof in a constant depth circuit. Similarly a cut of any type can be implemented by a constant depth circuit. Hence to normalise any proof-net one can compose cut-free circuits in parallel in one layer, followed by all the cuts in parallel as a second layer. The circuit corresponding to a cut is shown in figure 3. In fact one could make the circuit even flatter by computing the composition of the various permutations and implementing them in two layers of swaps.

Hence if there are n axiom links and m atomic cuts in a proof-net π, the circuit requires $O(n)$ gates total and is of constant depth. To identify the resulting normal form requires expected $O(\log n)$ repetitions of the experiment.

We note that, aside from the projection operators, the circuits constructed above fall into the complexity class $\mathbf{QNC}^0$, the lowest level of the of hierarchy of efficiently parallelisable quantum computations [22].

Definition 11. A *one-layer angelic quantum circuit* consists of a tensor product one-qubit unitary gates, CNOT gates and one-qubit projection operators.

A depth k angelic quantum circuit is a sequential composition of k one-layer angelic quantum circuits.

Definition 12. Let $\mathbf{QNC}^k_{angel}$ be the class of quantum operators on n inputs, computable by an angelic quantum circuit of depth $O(\log^k n)$, using polynomially many gates and m ancillae[5]; then let $\mathbf{QNC}_{angel} = \cup_k \mathbf{QNC}^k_{angel}$.

Although **Qubit** is the image of MLL with one ground type, the general case presents no obstacle. To handle any finite set of ground types simply work in a category of "decorated qubits" annotated by the syntax. Since the model is sound it is guaranteed that only qubits with matching annotations will be cut against each other. Further, given a circuit for a proof-net containing a type variable $[\![\pi(\alpha)]\!]$ we can lift the circuit to any higher type $[\![\pi[A/\alpha]]\!]$ using the decomposition lemma. This lifting only increases the width of the circuit, not the depth. Hence the circuit compilation described immediately gives the following.

Proposition 13. *The cut-elimination problem for MLL (without units) is in* $\boldsymbol{QNC}^0_{angel}$.

Recent work by Mairson and Terui [21] has shown, by reducing Boolean circuits to 2nd order linear λ-terms, that the cut-elimination problem for MLL is **P**-complete. Terui [29] has presented a direct encoding of Boolean circuits into unit-free MLL proof-nets. Based on this was have:

Proposition 14. $\boldsymbol{P} \subseteq \boldsymbol{QNC}^0_{angel}$.

Remark. $\mathbf{QNC}^0_{angel}$ is hardly a realistic complexity class, because angelic circuits are not a realistic model of quantum computation. If one replaces the projection operators with suitable 1-qubit observables then the resulting circuit computes the desired result upto a Pauli operator, which in principle can be corrected for. However, computing which Pauli operator to apply, given the measurement outcomes, seems to be problematic if one wants to preserve the complexity. The above proposition simply underlines the importance of classical information in quantum computation.

6 Conclusions and Further Work

We have shown that constructing a model of multiplicative linear logic atop the compact closed structure of quantum mechanics allows the distinction between entangled and separable states to be captured at the level of types with a very natural interpretation of multiplicative proof-nets as systems of entangled pairs. We have derived a compilation process from proof-nets, and hence also linear λ-terms, to quantum circuits, with normalisation realised by projective measurements. Looking the other way, we have shown that the behaviour of protocols based on entanglement swapping can be captured by the dynamics of cut-elimination in linear logic.

Four principal, orthogonal, desiderata remain. Firstly, to restore classical information to the picture in order to obtain a realistic "output" from the compilation process. The framework of compact closed categories with biproducts [1] provides a suitable categorical model. Secondly, to expand and refine the model. Clearly the class of quantum states which arise as models of MLL terms is much smaller than the circuit computable ones. However, by adding non-logical axioms to the system [2] it can be expanded – but how

[5]For present purposes there is no need for ancillae but see [22] for a discussion

far? Thirdly, to investigate more thoroughly how double gluing or specification structures can be used to give a type theory for entangled states. And finally, the relation of this work to other models of measurement based quantum computation has not yet been investigated.

7 Acknowledgements

Thanks to Samson Abramsky, Bob Coecke, Peter Hines, Elham Kashefi, Prakash Panangaden and Phil Scott for many helpful discussions and suggestions. Thanks also to the referees of an earlier version of this work, who pointed out the connection with **QNC** and drew attention to the role of weak distributivity. This work is supported by funding from the Oxford University Computing Laboratory. With apologies to Baillot, Danos, Erhard and Regnier [6].

References

[1] 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.

[2] S. Abramsky and R. Duncan. A categorical quantum logic. In Peter Selinger, editor, *Proceedings of the 2nd International Workshop on Quantum Programming Languages, Turku Centre for Computer Science General Publication No. 33*, pages 3–20, 2004.

[3] S. Abramsky, S. Gay, and R. Nagarajan. Specification structures and propostitions-as-types for concurrency. In F. Moller and G. Birtwistle, editors, *Logics for Concurrency: Structure vs Automata — Proceedings of the Banff Higher Order Workshop.* Springer-Verlag Lecture Notes in Computer Science, 1995.

[4] 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.

[5] P. Aliferis and D. W. Leung. Computation by measurement: a unifying picture. arxiv:quant-ph/0404082, 2004.

[6] P. Baillot, V. Danos, T. Ehrhard, and L. Regnier. Believe it or not, ajm's games model is a model of classical linear logic. In *Proceedings of the 12th Symposium on Logic in Computer Science*. IEEE Computer Society Press, 1997.

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

[8] 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.

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

[10] B. Coecke. Logic of entanglement. Technical Report PRG-RR-03-12, Oxford University Computing Laboratory, 2003. Short version: arxiv:quant-ph/0402014.

[11] V. Danos and L. Regnier. The structure of multiplicatives. *Arch. Math. Logic*, 28(3):181–203, 1989.

[12] A. Ekert and R. Jozsa. Quantum algorithms: Entanglement enhanced information processing. *Phil. Trans. Roy. Soc. London Ser A*, 356, 1998.

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

[14] 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.

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

[16] P. Jorrand and S. Perdrix. Unifying quantum computation with projective measurements only and one-way quantum computation. arxiv:quant-ph/0404125, 2004.

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

[18] E. Knill. Conventions for quantum pseudocode. Technical Report LAUR-96-2724, Los Alamos National Laboratory, 1996.

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

[20] S. MacLane. *Categories for the Working Mathematician.* Springer-Verlag, 1997.

[21] H. G. Mairson and K. Terui. On the computational complexity of cut-elimination in linear logic. In *Proceedings of ICTCS 2003, LNCS 2841*, pages 23–36. Springer-Verlag, October 2003.

[22] C. Moore and M. Nilsson. Parallel quantum computation and quantum codes. *SIAM J. Comput.*, 31(3):799–815, 2001.

[23] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information.* Cambridge University Press, 2000.

[24] M.A. Nielsen. Universal quantum computation using only projective measurement, quantum memory, and preparation of the 0 state. *Phys. Lett. A*, 308(96), 2003. arxiv:quant-ph/0108020.

[25] R. Raussendorf and H. J. Briegel. Computational model for the one-way quantum computer: Concepts and summary. In G. Leuchs and T. Beth, editors, *Quantum Information Processing*. Wiley, 2003.

[26] R. A. G. Seely. Linear logic, *-autonomous categories and cofree algebras. In *Conference on Categories in Computer Science and Logic*, volume 92 of *AMS Contemporary Mathematics*, pages 371–382. June 1989.

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

[28] A. M. Tan. *Full Completeness For Models Of Linear Logic.* PhD thesis, Cambridge University, 1997.

[29] K. Terui. Proof nets and boolean circuits. In *Proceedings of the 19th Annual IEEE Symposium on Logic in Computer Science: LICS 2004*. IEEE Computer Society, 2004.

8 Appendix

Proof of Lemma 5. Induction on the structure π.

- If π is an axiom then $[\![\pi]\!] = \eta_{A_1}$, giving the result.
- If π is derived from π_1, π_2 by the times rule, then $[\![\pi]\!] = [\![\pi_1]\!] \otimes [\![\pi_2]\!]$ by construction. By induction hypothesis $[\![\pi_1]\!] = (\bigotimes_{i=1}^{k} \eta_{A_i}); s_1$ and $[\![\pi_2]\!] = (\bigotimes_{i=k+1}^{n} \eta_{A_i}); s_2$ for some k, which gives the result, since $s_1 \otimes s_2$ is a permutation on the combined tensor.
- If π is derived from π' by the par rule then $[\![\pi]\!] = [\![\pi']\!]$, and result follows by induction.
- If π arises by an exchange, then $[\![\pi]\!] = [\![\pi']\!]; t$ where t is a transposition. Result follows trivially.

$\square$

Proof of Lemma 6. We use induction on the structure of π.

- π is an axiom : then $\pi \cong (\vdash A^{\perp}, A\ ; \varrho_1)$. In this case $[\![\pi]\!] = \eta_A$ and the induced permutation s is just the identity, so trivially $\sigma = s^{-1}\varrho_1 s$.

- π is formed by π' by application of the par rule: We have

$$\pi' \cong (\vdash \Gamma, A, B, \Delta\,; \sigma')$$
$$\pi \cong (\vdash \Gamma, A \parr B, \Delta\,; \sigma'),$$

  and $[\![\pi]\!] = [\![\pi']\!]$ by construction. By induction hypothesis, the result holds for π' and hence also for π.

- π is formed by applying the times rule to π_1 and π_2: in this case

$$\pi_1 \cong (\vdash \Gamma, A\,; \sigma)$$
$$\pi_2 \cong (\vdash B, \Delta\,; \sigma)$$
$$\pi \cong (\vdash \Gamma, A \otimes B, \Delta\,; \sigma_1\sigma_2),$$

  modulo a suitable renaming of the atoms of π_2. By construction $[\![\pi]\!] = [\![\pi_1]\!] \otimes [\![\pi_2]\!]$. Since the induced permutations s_1 and s_2 are disjoint, the permutation induced by $[\![\pi_1]\!] \otimes [\![\pi_2]\!]$ is just their product $s = s_1 s_2$. Now by induction hypothesis $\sigma_1 = s_1^{-1} \varrho_k s_1$ and $\sigma_2 = s_2^{-1} \varrho_{k'} s_2$, hence

$$\sigma_1\sigma_2 = (s_1^{-1}\varrho_k s_1)(s_2^{-1}\varrho_{k'} s_2) = s_2^{-1} s_1^{-1} \varrho_k \varrho_{k'} s_1 s_2$$

  which gives the result.

- π arises from π' by an exchange of A and B: It suffices to consider the case when A, B are atoms, since more general permutations can be constructed from products of transpositions. Let

$$\pi' \cong (\vdash \Gamma, A, B, \Delta\,; \sigma')$$

  so that π is a proof of $\vdash \Gamma, B, A, \Delta$. If i is the index of A in π, then

$$\sigma' = (a\, i)(b\, i+1)\sigma''$$

  and

$$\sigma \ = (a\, i+1)(b\, i)\sigma''$$

  for some a, b, σ''. By its construction $[\![\pi]\!] = [\![\pi']\!]; t$ where $t = (i\, i+1)$, and hence the permutation induced by $[\![\pi]\!]$ is $s = s't$, where s' is the induced permutation of π'. But note that

$$\sigma = (i\, i+1)\sigma'(i\, i+1) = t s'^{-1} \varrho_n s' t = s^{-1} \varrho s$$

  which is the desired result.

$\square$

The rest of this appendix is dedicated to a detailed proof of Theorem 8: i.e. the soundness of the model.

Lemma 15.

$$\begin{tikzcd} I \arrow[d, "f\otimes\eta_A"'] \arrow[drr, "f"] & & \\ \Delta\otimes A\otimes A^*\otimes A \arrow[rr, "id_\Delta\otimes\epsilon_A\otimes id_A"] & & \Delta\otimes A \end{tikzcd}$$

Proof. Immediate from the coherence equations (1) and the functoriality of tensor. □

This lemma actually establishes Theorem 8 for the special case where the left-hand proof net is just an axiom. We can generalise this result as the next lemma.

Lemma 16 (choice of fixpoint). *Suppose $f : I \longrightarrow \Gamma\otimes A_k$ is given by*

$$f = (\bigotimes_{i=1}^{n} \eta_{A_i}); s$$

where s is a permutation on $\bigotimes_{i=1}^{n}(A_i^*\otimes A_i)$, and let σ be a permutation on n such that $\sigma(n) = k$; there exists a unique permutation t such that for

$$g = (\bigotimes_{i=1}^{n} \eta_{A_{\sigma(i)}}); (t\otimes id_{A_k})$$

then $g = f$.

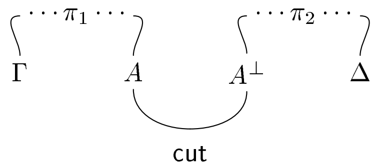
Proof. Let σ' be the permutation that makes

$$\begin{tikzcd} & I \arrow[dl, "\bigotimes_{i=1}^{n}\eta_{A_{\sigma(i)}}"'] \arrow[dr, "\bigotimes_{i=1}^{n}\eta_{A_i}"] & \\ \bigotimes_{i=1}^{n}(A_{\sigma(i)}^*\otimes A_{\sigma(i)}) \arrow[rr, "\sigma'"'] & & \bigotimes_{i=1}^{n}(A_i^*\otimes A_i) \end{tikzcd}$$

commute. Then $\sigma'; s = t\otimes \mathrm{id}_{A_k^*}$. □

Remark. The condition $\sigma(n) = k$ is stated for definiteness only; the important point is that one of the components of $\bigotimes_{i=1}^{n}(A_{\sigma(i)}^*\otimes A_{\sigma(i)})$ is fixed by $s;\sigma'$. The above applies equally well to any such situation.

Lemma 17 (atomic cuts). *Let A be an atom, and π_1 and π_2 be cut-free proof nets of $\vdash \Gamma, A$ and $\vdash A^\perp, \Delta$ respectively. Form the proof net π by*



(2)

and let π^* be the proof net formed by eliminating the cut from π. In this case,

$$[\![\pi]\!] = [\![\pi^*]\!].$$

Proof. By choice of fixpoint (and a slight abuse of notation) we can write

$$[\![\pi_1]\!] \otimes [\![\pi_2]\!] = (\eta_\Gamma \otimes \eta_A \otimes \eta_A \otimes \eta_\Delta); (t_1 \otimes \mathrm{id}_A \otimes \mathrm{id}_{A^*} \otimes t_2).$$

Hence

$$\begin{aligned}
[\![\pi]\!] &= (\eta_\Gamma \otimes \eta_A \otimes \eta_A \otimes \eta_\Delta); (t_1 \otimes \mathrm{id}_A \otimes \mathrm{id}_{A^*} \otimes t_2); (\mathrm{id}_\Gamma \otimes \epsilon_A \otimes \mathrm{id}_\Delta) \\
&= (\eta_\Gamma \otimes \eta_A \otimes \eta_A \otimes \eta_\Delta); (\mathrm{id}_\Gamma \otimes \mathrm{id}_A \otimes \mathrm{id}_{A^*} \otimes t_2); (\mathrm{id}_\Gamma \otimes \epsilon_A \otimes \mathrm{id}_\Gamma); (t_1 \otimes \mathrm{id}_\Delta) \\
&= (\eta_\Gamma \otimes \eta_A \otimes \eta_\Delta); (\mathrm{id}_\Gamma \otimes t_2); (t_1 \otimes \mathrm{id}_\Delta).
\end{aligned}$$

The last line follows by Lemma 15.

Now suppose that the atoms of π_1 are labelled $1, \ldots, k$ and those of π_2 are labelled $k-1, \ldots, n$ so t_1, t_2 act on elements $1, \ldots, k-1$ and $k, \ldots, n$ respectively. Note that k is necessarily even, as is n. Suppose $t_1(k-1) = a_1$ and $t_2(k) = a_2'$. Note that, assuming $\pi_1 \cong (\vdash \Gamma, A^\perp; \sigma_1)$ and $\pi_1 \cong (\vdash A, \Delta; \sigma_2)$,

$$\begin{aligned}
\sigma_1 &= (a_1\ k)\sigma_1' \\
\sigma_2 &= (a_2'\ k-1)\sigma_2'.
\end{aligned}$$

The permutation induced by $[\![\pi]\!]$ is

$$\sigma = t_1^{-1} t_2^{-1} \varrho_{n/2} t_2 t_1.$$

Except for the values a_1 and a_2', σ behaves exactly as σ_1 or σ_2. Now

$$\begin{aligned}
\sigma(a_1) &= (\varrho_{n/2} t_2 t_1)(k-1) \\
&= (t_2 t_1)(k) \\
&= a_2'
\end{aligned}$$

and similarly $\sigma(a_2') = a_1$. Hence

$$\sigma = (a_1\ a_2')\sigma_1'\sigma_2'$$

which is the permutation corresponding to π^*. $\square$

Corollary 18. *Theorem 8 follows immediately from decomposition (lemma 2 and the above.*