Semantic Techniques for Quantum Computation

Semantic Techniques for Quantum Computation

Edited by
Simon Gay and Ian Mackie

Contents

1

Extended Measurement Calculus

Vincent Danos, Elham Kashefi, Prakash Panangaden, Simon Perdrix

1.1 Introduction

The emergence of quantum computation has changed our perspective on many fundamental aspects of computing: the nature of information and how it flows, new algorithmic design strategies and complexity classes and the very structure of computational models. New challenges have been raised in the physical implementation of quantum computers. This chapter is an investigation into the structure, scope and limits of quantum computation. The main issues are questions about how quantum processes are defined, how quantum algorithms compose, how quantum resources are used and how classical and quantum information interact.

Traditionally, the main framework to explore quantum computation has been the circuit model Deutsch (1989), based on unitary evolution. This is very useful for algorithmic development and complexity analysis Bernstein and Vazirani (1997). There are other models such as quantum Turing machines Deutsch (1985) and quantum cellular automata Watrous (1995); van Dam (1996); Dürr and Santha (1996); Schumacher and Werner (2004). Although they are all proved to be equivalent from the point of view of expressive power, there is no agreement on what is the canonical model for exposing the key aspects of quantum computation.

On the other hand, physicists have introduced novel ideas based on the use of measurement and entanglement to perform computation Gottesman and Chuang (1999); Raussendorf and Briegel (2001); Raussendorf et al. (2003); Nielsen (2003). This is very different from the circuit model where measurement is done only at the end to extract classical output. In measurement-based quantum computation the main operation to manipulate information and control computation is measurement. This is surprising because measurement creates indeterminacy, yet it is used to express deterministic computation defined by a unitary evolution.

The idea of computing based on measurements emerged from the teleportation protocol Bennett et al. (1993). The goal of this protocol is for an agent to transmit an unknown qubit to a remote agent without actually sending the qubit. This protocol works by having the two parties share a maximally entangled state called a Bell pair. The parties perform *local* operations – measurements and unitaries – and communicate only classical bits. Remarkably, from this classical information the second party can reconstruct the unknown quantum state. In fact one can actually use this to compute via teleportation by choosing an appropriate measurement Gottesman and Chuang (1999). This is the key idea of measurement-based computation.

It turns out that the above method of computing is actually universal. This was first shown by Gottesman and Chuang Gottesman and Chuang (1999) who used two-qubit measurements and given Bell pairs. The one-way computer was then invented by Raussendorf and Briegel Raussendorf and Briegel (2001, 2002) which used only single-qubit measurements with a particular multi-party entangled state, the cluster state. In another approach, Nielsen Nielsen (2003) showed that one could do universal quantum computing with only 4-qubit measurements with no prior Bell pairs, however this works only probabilistically. Later Leung Leung (2004) improved Nielsen's method using only two qubits measurements and finally Perdrix and Jorrand Perdrix (2003); Perdrix and Jorrand (2004) gave the minimal set of

measurements to perform universal quantum computing – but still in the probabilistic setting – and introduced the state-transfer and measurement-based quantum Turing machine.

More precisely, a computation consists of a phase in which a collection of qubits are set up in a standard entangled state. Then measurements are applied to individual qubits and the outcomes of the measurements may be used to determine further adaptive measurements. Finally – again depending on measurement outcomes – local adaptive unitary operators, called corrections, are applied to some qubits; this allows the elimination of the indeterminacy introduced by measurements. The phrase "one-way" is used to emphasize that the computation is driven by irreversible measurements.

There are at least two reasons to take measurement-based models seriously: one conceptual and one pragmatic. The main pragmatic reason is that the *one-way* model is believed by physicists to lend itself to easier implementations Nielsen (2004); Clark et al. (2005); Browne and Rudolph (2005); Tame et al. (2004, 2006); Walther et al. (2005); Kay et al. (2006); Benjamin et al. (2005); Chen et al. (2006); Benjamin et al. (2006). Physicists have investigated various properties of the cluster state and have accrued evidence that the physical implementation is scalable and robust against decoherence Schlingemann (2003); Hein et al. (2004); Dür et al. (2003); den Nest et al. (2004b,a); Mhalla and Perdrix (2004); Gilbert et al. (2005); Hartmann et al. (2005); Dawson et al. (2006). Conceptually the measurement-based model highlights the role of entanglement and separates the quantum and classical aspects of computation; thus it clarifies, in particular, the interplay between classical control and the quantum evolution process.

Our approach to understanding the structural features of measurement-based computation is to develop a formal calculus Danos et al. (2007). One can think of this as an "assembly language" for measurement-based computation. Ours is the first programming framework specifically based on the one-way model. We first develop a notation for such classically correlated sequences of entanglements, measurements, and local corrections. Computations are organized in patterns, we use the word "pattern" rather than "program", because this corresponds to the commonly used terminology in the physics literature. We give a careful treatment of the composition and tensor product (parallel composition) of patterns. We show next that such pattern combinations reflect the corresponding combinations of unitary operators. An easy proof of universality follows.

So far, this is primarily a clarification of what was already known from the series of papers introducing and investigating the properties of the one-way model Raussendorf and Briegel (2001, 2002); Raussendorf et al. (2003). However, we work here with an extended notion of pattern, where inputs and outputs may overlap in any way one wants them to, and this results in more efficient – in the sense of using fewer qubits – implementations of unitaries. Specifically, our universal set consists of patterns using only 2 qubits. From it we obtain a 3 qubit realisation of the R_z rotations and a 14 qubit realisation for the controlled-U family: a significant reduction over the hitherto known implementations.

We then introduce a calculus of local equations over patterns that exploits some special algebraic properties of the entanglement, measurement and correction operators. More precisely, we use the fact that 1-qubit measurements are closed under conjugation by Pauli operators and the entanglement command belongs to the normalizer of the Pauli group. We show that this calculus is sound in that it preserves the interpretation of patterns. Most importantly, we derive from it a simple algorithm by which any general pattern can be put into a standard form where entanglement is done first, then measurements, then corrections. We call this *standardization*.

The consequences of the existence of such a procedure are far-reaching. Since entangling comes first, one can prepare the entire entangled state needed during the computation right at the start: one never has to do "on the fly" entanglements. Furthermore, the rewriting of a pattern to standard form reveals parallelism in the pattern computation. In a general pattern, one is forced to compute sequentially and to strictly obey the command sequence, whereas, after standardization, the dependency structure is relaxed, resulting in lower computational depth complexity Broadbent and Kashefi (2009).

Last, the existence of a standard form for any pattern also has interesting corollaries beyond implementation and complexity matters, as it follows from it that patterns using no dependencies, or using only the restricted class of Pauli measurements, can only realise a unitary belonging to the Clifford group, and hence can be efficiently simulated by a classical computer Gottesoman (1997).

As we have noted before, there are other methods for measurement-based quantum computing: the

teleportation technique based on two-qubit measurements Bennett et al. (1993); Gottesman and Chuang (1999) and the state-transfer approach based on single qubit measurements and incomplete two-qubit measurements Perdrix (2003, 2007). We will analyze both models and their relations to the one-way model. We will show how our calculus can be smoothly extended to cover these cases as well as other generalisation of the one-way model. We get several benefits from our treatment through a workable syntax for handling the dependencies of operators on previous measurement outcomes just by mimicking the one obtained in the one-way model. This has never been done before for the teleportation or state transform models. Furthermore, we can use these embeddings to obtain a standardisation procedure for these models. Finally the extended calculi can be compositionally embedded back in the original one-way model. This clarifies the relation between different measurement-based models and shows that the one-way model of Raussendorf and Briegel is the canonical one.

Having obtained the rigourous mathematical model underlying the measurement-based quantum computing, we explore whether this model may suggest new techniques for designing quantum algorithms and protocols. We start with the observation that one-way patterns implicitly define a particular decomposition of unitary maps into a preparation map enlarging the input space, a diagonal map with unit coefficients, and a restriction map contracting back the space to the output space, which we call a *phase map decomposition* de Beaudrap et al. (2006, 2008). However this decomposition does not directly correspond to any physical procedure leading to the definition of projection-based quantum computing. In other words, a projection-based pattern encapsulates most of the non-adaptive aspects of a measurement-based computation. We then demonstrate how phase map decomposition can be used to implement unitary maps directly into the projection-based model for quantum computing de Beaudrap et al. (2006, 2008).

A natural step to take from there, is to investigate how to transform a projection-based pattern specification to a measurement-based implementation. This is the basis of our next structural result which goes some way to explain another key property of MBQC, namely that although quantum measurements are inherently not deterministic, one can sometimes ensure the global determinism of the computation using suitable dependencies between measurements Danos and Kashefi (2006); Browne et al. (2007). The result asserts that under a graph-theoretic condition on the entanglement underlying a given computation, namely the existence of a *flow*, it is possible to construct such dependencies. This is significant progress in the direct understanding of the specifics of measurement-based information processing. Building on this criterion and the well known stabilizer formalism, we then present a characterisation of both the quantum and classical MBQC information flow Browne et al. (2007). An efficient algorithm for finding optimal flow will be also represented Mhalla and Perdrix (2008).

Finally we conclude this chapter with demonstrating how the obtained MBQC tools can be used in the traditional quantum circuit model. Indeed, we present a procedure for translating a given circuit into an MBQC computation which is no longer based on a gate by gate translation Broadbent and Kashefi (2009). With this in place, the measurement calculus leads to a rewriting system for quantum circuits which decreases computation depth, this being an important issue in relation with decoherence time.

1.2 MBQC - Syntax

We first develop a notation for 1-qubit measurement-based computations. The basic commands one can use in a pattern are:

- 1-qubit auxiliary preparation N_i
- 2-qubit entanglement operators E_{ij}
- 1-qubit measurements M_i^α
- and 1-qubit Pauli operators corrections X_i and Z_i

The indices i, j represent the qubits on which each of these operations apply, and α is a parameter in $[0, 2\pi]$. Expressions involving angles are always evaluated modulo 2π. These types of command will be referred to as N, E, M and C. Sequences of such commands, together with two distinguished – possibly

overlapping – sets of qubits corresponding to inputs and outputs, will be called *measurement patterns*, or simply patterns. These patterns can be combined by composition and tensor product.

Importantly, corrections and measurements are allowed to depend on previous measurement outcomes. We shall prove later that patterns without these classical dependencies can only realise unitaries that are in the Clifford group. Thus, dependencies are crucial if one wants to define a universal computing model; that is to say, a model where all unitaries over $\otimes^n\mathbb{C}^2$ can be realised. It is also crucial to develop a notation that will handle these dependencies. This is what we do now.

Preparation N_i prepares qubit i in state $|+\rangle_i$. The entanglement commands are defined as $E_{ij} := \wedge Z_{ij}$ (controlled-Z), while the correction commands are the Pauli operators X_i and Z_i.

Measurement M_i^α is defined by orthogonal projections on

$$\begin{aligned} |+_\alpha\rangle &:= \tfrac{1}{\sqrt{2}}(|0\rangle + e^{i\alpha}|1\rangle) \\ |-_\alpha\rangle &:= \tfrac{1}{\sqrt{2}}(|0\rangle - e^{i\alpha}|1\rangle) \end{aligned}$$

followed by a trace-out operator. The parameter $\alpha \in [0, 2\pi]$ is called the *angle* of the measurement. For $\alpha = 0$, $\alpha = \frac{\pi}{2}$, one obtains the X and Y Pauli measurements. Operationally, measurements will be understood as destructive measurements, consuming their qubit. The *outcome* of a measurement done at qubit i will be denoted by $s_i \in \mathbb{Z}_2$. Since one only deals here with patterns where qubits are measured at most once (see condition (D1) below), this is unambiguous. We take the specific convention that $s_i = 0$ if under the corresponding measurement the state collapses to $|+_\alpha\rangle$, and $s_i = 1$ if to $|-_\alpha\rangle$.

Outcomes can be summed together resulting in expressions of the form $s = \sum_{i\in I} s_i$ which we call *signals*, and where the summation is understood as being done in $\mathbb{Z}_2$. We define the *domain* of a signal as the set of qubits on which it depends.

As we have said before, both corrections and measurements may depend on signals. Dependent corrections will be written X_i^s and Z_i^s and dependent measurements will be written ${}^t[M_i^\alpha]^s$, where $s, t \in \mathbb{Z}_2$ and $\alpha \in [0, 2\pi]$. The meaning of dependencies for corrections is straightforward: $X_i^0 = Z_i^0 = I$, no correction is applied, while $X_i^1 = X_i$ and $Z_i^1 = Z_i$. In the case of dependent measurements, the measurement angle will depend on s, t and α as follows:

$$ {}^t[M_i^\alpha]^s := M_i^{(-1)^s\alpha + t\pi} \tag{1.1}$$

so that, depending on the parities of s and t, one may have to modify the α to one of $-\alpha$, $\alpha + \pi$ and $-\alpha + \pi$. These modifications correspond to conjugations of measurements under X and Z:

$$X_i M_i^\alpha X_i = M_i^{-\alpha} \tag{1.2}$$

$$Z_i M_i^\alpha Z_i = M_i^{\alpha+\pi} \tag{1.3}$$

accordingly, we will refer to them as the X and Z-actions. Note that these two actions commute, since $-\alpha + \pi = -\alpha - \pi$ up to 2π, and hence the order in which one applies them does not matter.

As we will see later, relations (1.2) and (1.3) are key to the propagation of dependent corrections, and to obtaining patterns in the standard entanglement, measurement and correction form. Since the measurements considered here are destructive, the above equations actually simplify to

$$M_i^\alpha X_i = M_i^{-\alpha} \tag{1.4}$$

$$M_i^\alpha Z_i = M_i^{\alpha-\pi} \tag{1.5}$$

Another point worth noticing is that the domain of the signals of a dependent command, be it a measurement or a correction, represents the set of measurements which one has to do before one can determine the actual value of the command.

We have completed our catalog of basic commands, including dependent ones, and we turn now to the definition of measurement patterns. For convenient reference, the language syntax is summarized in 1.1. We proceed now with the formal definition of a measurement pattern.

Definition 1 *Patterns consists of three finite sets V, I, O, together with two injective maps $\iota : I \to V$ and $o : O \to V$ and a finite sequence of commands $A_n \dots A_1$, read from right to left, applying to qubits in V in that order, i.e. A_1 first and A_n last, such that:*

$$
\begin{array}{lcll}
S & := & 0,\ 1,\ s_i,\ S+S & \text{Signals} \\
A & := & N_i & \text{Preparations} \\
 & & E_{ij} & \text{Entanglements} \\
 & & {}^t[M_i^\alpha]^s & \text{Measurements} \\
 & & X_i^s,\ Z_i^s & \text{Corrections}
\end{array}
$$

Fig. 1.1. 1-qubit based measurement language syntax

(D0) *no command depends on an outcome not yet measured;*

(D1) *no command acts on a qubit already measured;*

(D2) *no command acts on a qubit not yet prepared, unless it is an input qubit;*

(D3) *a qubit i is measured if and only if i is not an output.*

The set V is called the pattern *computation space*, and we write $\mathfrak{H}_V$ for the associated quantum state space $\otimes_{i\in V}\mathbb{C}^2$. To ease notation, we will omit the maps ι and o, and write simply I, O instead of $\iota(I)$ and $o(O)$. Note, however, that these maps are useful to define classical manipulations of the quantum states, such as permutations of the qubits. The sets I, O are called respectively the pattern *inputs* and *outputs*, and we write $\mathfrak{H}_I$, and $\mathfrak{H}_O$ for the associated quantum state spaces. The sequence $A_n \dots A_1$ is called the pattern *command sequence*, while the triple (V, I, O) is called the pattern *type*.

To run a pattern, one prepares the input qubits in some input state $\psi \in \mathfrak{H}_I$, while the non-input qubits are all set to the $|+\rangle$ state, then the commands are executed in sequence, and finally the result of the pattern computation is read back from outputs as some $\phi \in \mathfrak{H}_O$. Clearly, for this procedure to succeed, we had to impose the (D0), (D1), (D2) and (D3) conditions. Indeed if (D0) fails, then at some point of the computation, one will want to execute a command which depends on outcomes that are not known yet. Likewise, if (D1) fails, one will try to apply a command on a qubit that has been consumed by a measurement (recall that we use destructive measurements). Similarly, if (D2) fails, one will try to apply a command on a non-existent qubit. Condition (D3) is there to make sure that the final state belongs to the output space $\mathfrak{H}_O$, *i.e.*, that all non-output qubits, and only non-output qubits, will have been consumed by a measurement when the computation ends.

We write (D) for the conjunction of our definiteness conditions (D0), (D1), (D2) and (D3). Whether a given pattern satisfies (D) or not is statically verifiable on the pattern command sequence. We could have imposed a simple type system to enforce these constraints but, in the interests of notational simplicity, we chose not to do so.

Here is a concrete example:

$$\mathcal{H} \quad := \quad (\{1,2\}, \{1\}, \{2\}, X_2^{s_1} M_1^0 E_{12} N_2)$$

with computation space $\{1,2\}$, inputs $\{1\}$, and outputs $\{2\}$. To run $\mathcal{H}$, one first prepares the first qubit in some input state ψ, and the second qubit in state $|+\rangle$, then these are entangled to obtain $\wedge Z_{12}(\psi_1 \otimes |+\rangle_2)$. Once this is done, the first qubit is measured in the $|+\rangle$, $|-\rangle$ basis. Finally an X correction is applied on the output qubit, if the measurement outcome was $s_1 = 1$. We will do this calculation in detail later, and prove that this pattern implements the Hadamard operator H.

In general, a given pattern may use auxiliary qubits that are neither input nor output qubits. Usually one tries to use as few such qubits as possible, since these contribute to the *space complexity* of the computation.

A last thing to note is that one does not require inputs and outputs to be disjoint subsets of V. This, seemingly innocuous, additional flexibility is actually quite useful to give parsimonious implementations of unitaries Danos et al. (2005). While the restriction to disjoint inputs and outputs is unnecessary, it has been discussed whether imposing it results in patterns that are easier to realise physically. Recent work Hein et al. (2004); Browne and Rudolph (2005); Clark et al. (2005) however, seems to indicate it is not the case.

We are interested in how one can combine patterns in order to obtain bigger ones.

The first way to combine patterns is by composing them. Two patterns $\mathcal{P}_1$ and $\mathcal{P}_2$ may be composed if $V_1 \cap V_2 = O_1 = I_2$. Provided that $\mathcal{P}_1$ has as many outputs as $\mathcal{P}_2$ has inputs, by renaming the pattern qubits, one can always make them composable.

Definition 2 *The composite pattern $\mathcal{P}_2\mathcal{P}_1$ is defined as:*
— $V := V_1 \cup V_2$, $I = I_1$, $O = O_2$,
— *commands are concatenated.*

The other way of combining patterns is to tensor them. Two patterns $\mathcal{P}_1$ and $\mathcal{P}_2$ may be tensored if $V_1 \cap V_2 = \varnothing$. Again one can always meet this condition by renaming qubits in such a way that these sets are made disjoint.

Definition 3 *The tensor pattern $\mathcal{P}_1 \otimes \mathcal{P}_2$ is defined as:*
— $V = V_1 \cup V_2$, $I = I_1 \cup I_2$, *and* $O = O_1 \cup O_2$,
— *commands are concatenated.*

In contrast to the composition case, all the unions involved here are disjoint. Therefore commands from distinct patterns freely commute, since they apply to disjoint qubits, and when we say that commands have to be concatenated, this is only for definiteness. It is routine to verify that the definiteness conditions (D) are preserved under composition and tensor product.

Before turning to this matter, we need a clean definition of what it means for a pattern to implement or to realise a unitary operator, together with a proof that the way one can combine patterns is reflected in their interpretations. This is key to our proof of universality.

1.3 MBQC - Semantics

In this section we give a formal operational semantics for the pattern language as a probabilistic labeled transition system. We define deterministic patterns and thereafter concentrate on them. We show that deterministic patterns compose. We give a denotational semantics of deterministic patterns; from the construction it will be clear that these two semantics are equivalent.

Besides quantum states, which are non-zero vectors in some Hilbert space $\mathfrak{H}_V$, one needs a classical state recording the outcomes of the successive measurements one does in a pattern. If we let V stand for the finite set of qubits that are still active (i.e. not yet measured) and W stands for the set of qubits that have been measured (i.e. they are now just classical bits recording the measurement outcomes), it is natural to define the computation state space as:

$$\mathcal{S} \quad := \quad \Sigma_{V,W} \mathfrak{H}_V \times \mathbb{Z}_2^W.$$

In other words the computation states form a V, W-indexed family of pairs q, Γ, where q is a quantum state from $\mathfrak{H}_V$ and Γ is a map from some W to the outcome space $\mathbb{Z}_2$. We call this classical component Γ an *outcome map*, and denote by $\varnothing$ the empty outcome map in $\mathbb{Z}_2^{\varnothing}$. We will treat these states as pairs unless it becomes important to show how V and W are altered during a computation, as happens during a measurement.

Operational semantics

We need some preliminary notation. For any signal s and classical state $\Gamma \in \mathbb{Z}_2^W$, such that the domain of s is included in W, we take s_Γ to be the value of s given by the outcome map Γ. That is to say, if $s = \sum_I s_i$, then $s_\Gamma := \sum_I \Gamma(i)$ where the sum is taken in $\mathbb{Z}_2$. Also if $\Gamma \in \mathbb{Z}_2^W$, and $x \in \mathbb{Z}_2$, we define:

$$\Gamma[x/i](i) = x,\ \Gamma[x/i](j) = \Gamma(j) \text{ for } j \neq i$$

which is a map in $\mathbb{Z}_2^{W \cup \{i\}}$.

We may now view each of our commands as acting on the state space $\mathcal{S}$; we have suppressed V and W in the first 4 commands:

$$\begin{array}{lll} q, \Gamma & \xrightarrow{N_i} & q \otimes |+\rangle_i, \Gamma \\ q, \Gamma & \xrightarrow{E_{ij}} & \wedge Z_{ij} q, \Gamma \\ q, \Gamma & \xrightarrow{X_i^s} & X_i^{s_\Gamma} q, \Gamma \\ q, \Gamma & \xrightarrow{Z_i^s} & Z_i^{s_\Gamma} q, \Gamma \\ V \cup \{i\}, W, q, \Gamma & \xrightarrow{{}^t[M_i^\alpha]^s} & V, W \cup \{i\}, \langle +_{\alpha_\Gamma}|_i q, \Gamma[0/i] \\ V \cup \{i\}, W, q, \Gamma & \xrightarrow{{}^t[M_i^\alpha]^s} & V, W \cup \{i\}, \langle -_{\alpha_\Gamma}|_i q, \Gamma[1/i] \end{array}$$

where $\alpha_\Gamma = (-1)^{s_\Gamma}\alpha + t_\Gamma \pi$ following equation (1.1). Note how the measurement moves an index from V to W; a qubit once measured cannot be neasured again. Suppose $q \in \mathfrak{H}_V$, for the above relations to be defined, one needs the indices i, j on which the various command apply to be in V. One also needs Γ to contain the domains of s and t, so that s_Γ and t_Γ are well-defined. This will always be the case during the run of a pattern because of condition (D).

All commands except measurements are deterministic and only modify the quantum part of the state. The measurement actions on $\mathcal{S}$ are not deterministic, so that these are actually binary relations on $\mathcal{S}$, and modify both the quantum and classical parts of the state. The usual convention has it that when one does a measurement the resulting state is *renormalized* and the probabilities are associated with the transition. We do not adhere to this convention here, instead we leave the states unnormalized. The reason for this choice of convention is that this way, the probability of reaching a given state can be read off its norm, and the overall treatment is simpler. As we will show later, all the patterns implementing unitary operators will have the same probability for all the branches and hence we will not need to carry these probabilities explicitly.

We introduce an additional command called *signal shifting*:

$$q, \Gamma \xrightarrow{S_i^s} q, \Gamma[\Gamma(i) + s_\Gamma / i]$$

It consists in shifting the measurement outcome at i by the amount s_Γ. Note that the Z-action leaves measurements globally invariant, in the sense that $|+_{\alpha+\pi}\rangle, |-_{\alpha+\pi}\rangle = |-_\alpha\rangle, |+_\alpha\rangle$. Thus changing α to $\alpha + \pi$ amounts to swapping the outcomes of the measurements, and one has:

$$ {}^t[M_i^\alpha]^s \quad = \quad S_i^t \, {}^0[M_i^\alpha]^s \tag{1.6}$$

and signal shifting allows to dispose of the Z action of a measurement, resulting sometimes in convenient optimizations of standard forms.

Denotational semantics

Let $\mathcal{P}$ be a pattern with computation space V, inputs I, outputs O and command sequence $A_n \dots A_1$. To execute a pattern, one starts with some input state q in $\mathfrak{H}_I$, together with the empty outcome map $\varnothing$. The input state q is then tensored with as many $|+\rangle$s as there are non-inputs in V (the N commands), so as to obtain a state in the full space $\mathfrak{H}_V$. Then E, M and C commands in $\mathcal{P}$ are applied in sequence from right to left. We can summarize the situation as follows:

$$\begin{array}{ccccc} \mathfrak{H}_I & \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots & & & \mathfrak{H}_O \\ \downarrow & & & & \uparrow \\ \mathfrak{H}_I \times \mathbb{Z}_2^{\varnothing} & \xrightarrow{prep} & \mathfrak{H}_V \times \mathbb{Z}_2^{\varnothing} & \xrightarrow{A_1 \dots A_n} & \mathfrak{H}_O \times \mathbb{Z}_2^{V \setminus O} \end{array}$$

If m is the number of measurements, which is also the number of non outputs, then the run may follow 2^m different branches. Each branch is associated with a unique binary string $\mathbf{s}$ of length m, representing the classical outcomes of the measurements along that branch, and a unique *branch map* $A_{\mathbf{s}}$ representing

the linear transformation from $\mathfrak{H}_I$ to $\mathfrak{H}_O$ along that branch. This map is obtained from the operational semantics via the sequence (q_i, Γ_i) with $1 \leq i \leq n+1$, such that:

$$q_1, \Gamma_1 = q \otimes |+\ldots+\rangle, \varnothing$$
$$q_{n+1} = q' \neq 0$$
$$\text{and for all } i \leq n : q_i, \Gamma_i \xrightarrow{A_i} q_{i+1}, \Gamma_{i+1}.$$

Definition 4 *A pattern $\mathcal{P}$ realises a map on density matrices ρ given by $\rho \mapsto \sum_{\mathbf{s}} A_{\mathbf{s}}^\dagger(\rho) A_{\mathbf{s}}$. We write $[\![\mathcal{P}]\!]$ for the map realized by $\mathcal{P}$.*

Proposition 5 *Each pattern realizes a completely positive trace preserving map.*

Proof. Later on we will show that every pattern can be put in a semantically equivalent form where all the preparations and entanglements appear first, followed by a sequence of measurements and finally local Pauli corrections. Hence branch maps decompose as $A_{\mathbf{s}} = C_{\mathbf{s}} \Pi_{\mathbf{s}} U$, where $C_{\mathbf{s}}$ is a unitary map over $\mathfrak{H}_O$ collecting all corrections on outputs, $\Pi_{\mathbf{s}}$ is a projection from $\mathfrak{H}_V$ to $\mathfrak{H}_O$ representing the particular measurements performed along the branch, and U is a unitary embedding from $\mathfrak{H}_I$ to $\mathfrak{H}_V$ collecting the branch preparations, and entanglements. Note that U is the same on all branches. Therefore,

$$\begin{aligned} \textstyle\sum_{\mathbf{s}} A_{\mathbf{s}}^\dagger A_{\mathbf{s}} &= \textstyle\sum_{\mathbf{s}} U^\dagger \Pi_{\mathbf{s}}^\dagger C_{\mathbf{s}}^\dagger C_{\mathbf{s}} \Pi_{\mathbf{s}} U \\ &= \textstyle\sum_{\mathbf{s}} U^\dagger \Pi_{\mathbf{s}}^\dagger \Pi_{\mathbf{s}} U \\ &= U^\dagger (\textstyle\sum_{\mathbf{s}} \Pi_{\mathbf{s}}) U \\ &= U^\dagger U = I \end{aligned}$$

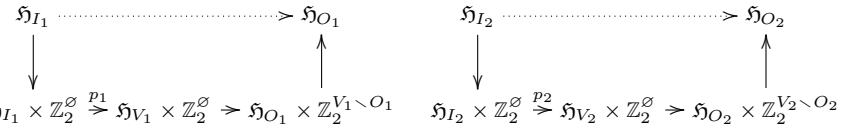
where we have used the fact that $C_{\mathbf{s}}$ is unitary, $\Pi_{\mathbf{s}}$ is a projection and U is independent of the branches and is also unitary. Therefore the map $T(\rho) := \sum_{\mathbf{s}} A_{\mathbf{s}}(\rho) A_{\mathbf{s}}^\dagger$ is a trace-preserving completely-positive map (cptp-map), explicitly given as a Kraus decomposition. □

Hence the denotational semantics of a pattern is a cptp-map. In our denotational semantics we view the pattern as defining a map from the input qubits to the output qubits. We do not explicitly represent the result of measuring the final qubits; these may be of interest in some cases. Techniques for dealing with classical output explicitly are given by Selinger Selinger (2004) and Unruh Unruh (2005). With our definitions in place, we will show that the denotational semantics is compositional.

Theorem 1 *For two patterns $\mathcal{P}_1$ and $\mathcal{P}_2$ we have $[\![\mathcal{P}_1 \mathcal{P}_2]\!] = [\![\mathcal{P}_2]\!][\![\mathcal{P}_1]\!]$ and $[\![\mathcal{P}_1 \otimes \mathcal{P}_2]\!] = [\![\mathcal{P}_2]\!] \otimes [\![\mathcal{P}_1]\!]$.*

Proof. Recall that two patterns $\mathcal{P}_1$, $\mathcal{P}_2$ may be combined by composition provided $\mathcal{P}_1$ has as many outputs as $\mathcal{P}_2$ has inputs. Suppose this is the case, and suppose further that $\mathcal{P}_1$ and $\mathcal{P}_2$ respectively realize some cptp-maps T_1 and T_2. We need to show that the composite pattern $\mathcal{P}_2 \mathcal{P}_1$ realizes $T_2 T_1$.

Indeed, the two diagrams representing branches in $\mathcal{P}_1$ and $\mathcal{P}_2$:



can be pasted together, since $O_1 = I_2$, and $\mathfrak{H}_{O_1} = \mathfrak{H}_{I_2}$. But then, it is enough to notice 1) that preparation steps p_2 in $\mathcal{P}_2$ commute with all actions in $\mathcal{P}_1$ since they apply on disjoint sets of qubits, and 2) that no action taken in $\mathcal{P}_2$ depends on the measurements outcomes in $\mathcal{P}_1$. It follows that the pasted diagram describes the same branches as does the one associated to the composite $\mathcal{P}_2 \mathcal{P}_1$.

A similar argument applies to the case of a tensor combination, and one has that $\mathcal{P}_2 \otimes \mathcal{P}_1$ realizes $T_2 \otimes T_1$. □

If one wanted to give a categorical treatment one can define a category where the objects are finite sets representing the input and output qubits and the morphisms are the patterns. This is clearly a monoidal category with our tensor operation as the monoidal structure. One can show that the denotational semantics gives a monoidal functor into the category of superoperators or into any suitably enriched

strongly compact closed category Abramsky and Coecke (2004) or dagger category Selinger (2005a). It would be very interesting to explore exactly what additional categorical structures are required to interpret the measurement calculus presented below. Duncan Ross Duncan (2005) has skectched a polycategorical presentation of our measurement calculus.

Determinism

We conclude this section by presenting various notions of determinism which will be used later when we return to the question of transforming a projection-based pattern to a measurement-based pattern. A pattern is said to be *deterministic* if it realizes a cptp-map that sends pure states to pure states. This is equivalent to saying that for a deterministic pattern branch maps are proportional, that is to say, for all $q \in \mathfrak{H}_I$ and all $\mathbf{s_1}$, $\mathbf{s_2} \in \mathbb{Z}_2^n$, $A_{\mathbf{s_1}}(q)$ and $A_{\mathbf{s_2}}(q)$ differ only up to a scalar. The class of deterministic patterns include projections, see example below.

A more restricted class contains all the unitary and unitary embedding operators: a pattern is said to be *strongly deterministic* when branch maps are equal (up to a global phase), *i.e.* for all $\mathbf{s}_1$, $\mathbf{s}_2 \in \mathbb{Z}_2^n$, $A_{\mathbf{s}_1} = e^{i\phi_{s_1,s_2}} A_{\mathbf{s}_2}$. These are the patterns implementing quantum algorithms and hence understanding their structural properties is of particular interest.

Proposition 6 *If a pattern is strongly deterministic, then it realizes a unitary embedding.*

Proof. Define T to be the map realized by the pattern. We have $T = \sum_{\mathbf{s}} A_{\mathbf{s}}^\dagger A_{\mathbf{s}}$. Since the pattern in strongly deterministic all the branch maps are the same. Define A to be $2^{n/2} A_{\mathbf{s}}$, then A must be a unitary embedding, because $A^\dagger A = I$. □

An important sub-class of deterministic patterns are robust under the changes of the angles: a pattern is said to be *uniformly deterministic* if it is deterministic for all values of its measurement angles. In another words a uniformly deterministic pattern defines a class of quantum operators that can be performed given the same initial entanglement resources. On the other hand it is known that if we fix the angle of measurements to be Pauli the obtained operators is in Clifford group Danos et al. (2007). That means uniform determinism allow us to associate to a family of quantum operators a canonical pattern implementing a Clifford operator, a potential valuable abstract reduction for the study of quantum operators.

Finally a pattern is said to be *stepwise deterministic* if it is deterministic after performing each single measurement together with all the corrections depending on the result of that measurement. In another words a pattern is stepwise deterministic if after each single measurements there exists a set of local corrections depending only on the result of this measurements to be performed on some or all of the non-measured qubits that will make the two branches equal (up to a global phase).

We assume that all the non-input qubits are prepared in the state $|+\rangle$ and hence for simplicity we omit the preparation commands N_{I^c}. First we give a quick example of a deterministic pattern that has branches with different probabilities. Its type is $V = \{1, 2\}$, $I = O = \{1\}$, and its command sequence is M_2^α. Therefore, starting with input q, one gets two branches:

$$q \otimes |+\rangle, \varnothing \quad \xrightarrow{M_2^\alpha} \quad \begin{cases} \frac{1}{2}(1 + e^{-i\alpha})q, \varnothing[0/2] \\ \\ \frac{1}{2}(1 - e^{-i\alpha})q, \varnothing[1/2] \end{cases}$$

Thus this pattern is indeed deterministic, and implements the identity up to a global phase, and yet the two branches have respective probabilities $(1+\cos\alpha)/2$ and $(1-\cos\alpha)/2$, which are not equal in general and hence this pattern is not strongly deterministic.

There is an interesting variation on this first example. The pattern of interest, call it $\mathcal{T}$, has the same type as above with command sequence $X_1^{s_2} M_2^0 E_{12}$. Again, $\mathcal{T}$ is deterministic, but not strongly deterministic: the branches have different probabilities, as in the preceding example. Now, however, these probabilities may depend on the input. The associated transformation is a cptp-map, $T(\rho) :=$

$A\rho A^\dagger + B\rho B^\dagger$ with:

$$A := \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, B := \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$$

One has $A^\dagger A + B^\dagger B = I$, so T is indeed a completely positive and trace-preserving linear map and $T(|\psi\rangle\langle\psi|) = \langle\psi,\psi\rangle|0\rangle\langle 0|$ and clearly for no unitary U does one have $T(\rho) := U\rho U^\dagger$.

For our final example, we return to the pattern $\mathcal{H}$, already defined above. Consider the pattern with the same qubit space $\{1,2\}$, and the same inputs and outputs $I = \{1\}$, $O = \{2\}$, as $\mathcal{H}$, but with a shorter command sequence namely $M_1^0 E_{12}$. Starting with input $q = (a|0\rangle + b|1\rangle)|+\rangle$, one has two computation branches, branching at M_1^0:

$$\begin{aligned} (a|0\rangle + b|1\rangle)|+\rangle, \varnothing \quad &\xrightarrow{E_{12}} \quad \tfrac{1}{\sqrt{2}}(a|00\rangle + a|01\rangle + b|10\rangle - b|11\rangle), \varnothing \\ &\xrightarrow{M_1^0} \quad \begin{cases} \frac{1}{2}((a+b)|0\rangle + (a-b)|1\rangle), \varnothing[0/1] \\ \\ \frac{1}{2}((a-b)|0\rangle + (a+b)|1\rangle), \varnothing[1/1] \end{cases} \end{aligned}$$

and since $\|a+b\|^2 + \|a-b\|^2 = 2(\|a\|^2 + \|b\|^2)$, both transitions happen with equal probabilities $\frac{1}{2}$. Both branches end up with non proportional outputs, so the pattern is *not* deterministic. However, if one applies the local correction X_2 on either of the branches' ends, both outputs will be made to coincide. If we choose to let the correction apply to the second branch, we obtain the pattern $\mathcal{H}$, already defined. We have just proved $H = U_{\mathcal{H}}$, that is to say $\mathcal{H}$ realizes the Hadamard operator.

1.4 MBQC - Universality

In this section we first introduce a simple parameterized family $J(\alpha)$ that generates all unitaries over $\mathbb{C}^2$. By adding the unitary operator controlled-Z ($\wedge Z$) defined over $\mathbb{C}^2 \otimes \mathbb{C}^2$, one then obtains a set of generators for all unitary maps over $\otimes^n \mathbb{C}^2$. Both $J(\alpha)$ and $\wedge Z$, have simple realizations in the one-way model, using only two qubits. As a consequence, one obtains an implementation of the controlled-U ($\wedge U$) family of unitaries, using only 14 qubits. Combining these as building blocks, any general unitary can be obtained by using relatively few auxiliary qubits Danos et al. (2005). Furthermore, our building blocks have an interesting property, namely that their underlying entanglement graphs have no odd-length cycles, and such states have been shown to be robust against decoherence Dür et al. (2003).

Consider the following one-parameter family $J(\alpha)$:

$$J(\alpha) \quad := \quad \tfrac{1}{\sqrt{2}} \begin{pmatrix} 1 & e^{i\alpha} \\ 1 & -e^{i\alpha} \end{pmatrix},$$

we can see already that the Pauli spin matrices, phase and Hadamard operators can be described using only $J(\alpha)$:

$$\begin{array}{ll} X = J(\pi)J(0) & P(\alpha) = J(0)J(\alpha) \\ Z = J(0)J(\pi) & H = J(0) \end{array}$$

We will also use the following equations:

$$\begin{array}{lcl} J(0)^2 & = & I \\ J(\alpha)J(0)J(\beta) & = & J(\alpha+\beta) \\ J(\alpha)J(\pi)J(\beta) & = & e^{i\alpha} Z\, J(\beta-\alpha) \end{array}$$

The second and third equations are referred to as the *additivity* and *subtractivity* relations. Additivity gives another useful pair of equations:

$$XJ(\alpha) = J(\alpha+\pi) = J(\alpha)Z \tag{1.7}$$

Any unitary operator U on $\mathbb{C}^2$ can be written:

$$U = e^{i\alpha} J(0)J(\beta)J(\gamma)J(\delta)$$

for some α, β, γ and δ in $\mathbb{R}$. We will refer to this as a *J-decomposition* of U. To prove this note that all three Pauli rotations are expressible in terms of $J(\alpha)$:

$$R_x(\alpha) = e^{-i\frac{\alpha}{2}} J(\alpha)J(0) \tag{1.8}$$
$$R_y(\alpha) = e^{-i\frac{\alpha}{2}} J(0)J(\frac{\pi}{2})J(\alpha)J(-\frac{\pi}{2}) \tag{1.9}$$
$$R_z(\alpha) = e^{-i\frac{\alpha}{2}} J(0)J(\alpha) \tag{1.10}$$

From the Z–X decomposition, we know that every 1-qubit unitary operator U can be written as:

$$U = e^{i\alpha} R_z(\beta)R_x(\gamma)R_z(\delta)$$

and using equations (1.10) and (1.8) we get:

$$U = e^{i\alpha} e^{-i\frac{\beta+\gamma+\delta}{2}} J(0)J(\beta)J(\gamma)J(\delta)$$

We conclude in particular, that $J(\alpha)$ generates all 1-qubit unitary operators.

Next, we turn to the decomposition of $\wedge U$ in terms of $J(\alpha)$ and $\wedge Z$. Subscripts to operators indicate the qubit to which they apply, and we sometimes abbreviate $J_i(\alpha)$ as J_i^α.

Suppose U has J-decomposition $e^{i\alpha}J(0)J(\beta)J(\gamma)J(\delta)$, then $\wedge U$ can also be decomposed as follows:

$$\wedge U_{12} = J_1^0 J_1^{\alpha'} J_2^0 J_2^{\beta+\pi} J_2^{-\frac{\gamma}{2}} J_2^{-\frac{\pi}{2}} J_2^0 \wedge Z_{12} J_2^{\frac{\pi}{2}} J_2^{\frac{\gamma}{2}} J_2^{\frac{-\pi-\delta-\beta}{2}} J_2^0 \wedge Z_{12} J_2^{\frac{-\beta+\delta-\pi}{2}}$$

with $\alpha' = \alpha + \frac{\beta+\gamma+\delta}{2}$.

To prove the above decomposition, we first define auxiliary unitary operators:

$$\begin{aligned} A &= J(0)J(\beta+\pi)J(-\tfrac{\gamma}{2})J(-\tfrac{\pi}{2}) \\ B &= J(0)J(\tfrac{\pi}{2})J(\tfrac{\gamma}{2})J(\tfrac{-\pi-\delta-\beta}{2}) \\ C &= J(0)J(\tfrac{-\beta+\delta-\pi}{2}) \end{aligned}$$

Then, using the additivity relation we obtain $ABC = I$. On the other hand, using both the subtractivity relation and equations (1.7), we get:

$$\begin{aligned} AXBXC &= J(0)J(\beta+\pi)J(-\tfrac{\gamma}{2})J(-\tfrac{\pi}{2})J(\pi)J(\tfrac{\pi}{2})J(\tfrac{\gamma}{2})J(\tfrac{-\pi-\delta-\beta}{2})J(\pi)J(\tfrac{-\beta+\delta-\pi}{2}) \\ &= e^{-i\frac{\delta+\beta+\gamma}{2}} J(0)J(\beta)J(\gamma)J(\delta) \end{aligned}$$

Therefore one also has $e^{i\frac{2\alpha+\beta+\gamma+\delta}{2}} AXBXC = U$.

Combining our two equations in A, B, C, we obtain $\wedge U_{12} = P_1(\alpha')A_2 \wedge X_{12} B_2 \wedge X_{12} C_2$ with $\alpha' = \alpha + \frac{\beta+\gamma+\delta}{2}$; a decomposition which we can rewrite using our generating set:

$$\begin{aligned} P(\alpha)_1 &= J_1^0 J_1^\alpha \\ \wedge X_{12} &= H_2 \wedge Z_{12} H_2 = J_2^0 \wedge Z_{12} J_2^0 \end{aligned}$$

to obtain the above decomposition of $\wedge U$.

As we will see, this decomposition leads to an implementation for the $\wedge U$ operator using only 14 qubits. Using Y or Z in place of X in the argument above, one finds costlier decompositions using 15 and 16 qubits. No comparable decomposition was given previously.

Having all unitaries U over $\mathbb{C}^2$ and all unitaries of the form $\wedge U$ over $\mathbb{C}^2 \otimes \mathbb{C}^2$ we can conclude that:

Theorem 2 (Universality) *The set $\{J(\alpha), \wedge Z\}$ generates all unitaries.*

The following unitaries $H = J(0)$, $P(\frac{\pi}{4}) = J(0)J(\frac{\pi}{4})$, and $\wedge X = J(0) \wedge Z J(0)$, are known to be *approximately universal*, in the sense that any unitary can be approximated within any precision by combining these Nielsen and Chuang (2000). Therefore the set $J(0)$, $J(\frac{\pi}{4})$ and $\wedge Z$ is also approximately universal.

It is easy to verify that the following patterns implement our generators

$$\begin{aligned} \mathcal{J}(\alpha) &:= X_2^{s_1} M_1^{-\alpha} E_{12} \\ \wedge\mathcal{Z} &:= E_{12} \end{aligned}$$

where in the first pattern 1 is the only input and 2 is the only output, while in the second both 1 and 2 are inputs and outputs (note that we are allowing patterns to have overlapping inputs and outputs). Combining these two patterns, by composition and tensoring, will therefore generate patterns realizing all unitaries over $\otimes^n \mathbb{C}^2$. These patterns are indeed among the simplest possible. Remarkably, there is only one single dependency overall, which occurs in the correction phase of $\mathcal{J}(\alpha)$. No set of patterns without any measurement could be a generating set, since those can only implement unitaries in the Clifford group as we prove later.

Let us now examine the implementation of $\wedge U$, based on the decomposition which we recall:

$$\wedge U_{12} \quad = \quad J_1^0 J_1^{\alpha'} J_2^0 J_2^{\beta+\pi} J_2^{-\frac{\gamma}{2}} J_2^{-\frac{\pi}{2}} J_2^0 \wedge Z_{12} J_2^{\frac{\pi}{2}} J_2^{\frac{\gamma}{2}} J_2^{\frac{-\pi-\delta-\beta}{2}} J_2^0 \wedge Z_{12} J_2^{\frac{-\beta+\delta-\pi}{2}}$$

with $\alpha' = \alpha + \frac{\beta+\gamma+\delta}{2}$. Replacing each of the generators in the above expression by the corresponding pattern, we get the following long equation that should be red from bottom to top and from right to left:

$$X_C^{s_B} M_B^0 E_{BC} X_B^{s_A} M_A^{-\alpha'} E_{AB} X_k^{s_j} M_j^0 E_{jk} X_j^{s_i} M_i^{-\beta-\pi} E_{ij} X_i^{s_h} M_h^{\frac{\gamma}{2}} E_{hi} X_h^{s_g} M_g^{\frac{\pi}{2}} E_{gh} X_g^{s_f} M_f^0 E_{fg} E_{Af} X_f^{s_e} M_e^{-\frac{\pi}{2}}$$
$$E_{ef} X_e^{s_d} M_d^{-\frac{\gamma}{2}} E_{de} X_d^{s_c} M_c^{\frac{\pi+\delta+\beta}{2}} E_{cd} X_c^{s_b} M_b^0 E_{bc} E_{Ab} X_b^{s_a} M_a^{\frac{\beta-\delta+\pi}{2}} E_{ab}$$

with input qubits $\{A, a\}$ and output qubits $\{C, k\}$. We have reserved uppercase (lowercase) letters for qubits used in implementing the J operators on control qubit A (target qubit a), and have indeed used a total number of 14 qubits. Figure (1.2) shows the corresponding entanglement graph, where vertices represent qubits and edges connect the qubits of an entangled pair. This graph has only one cycle of length 6.

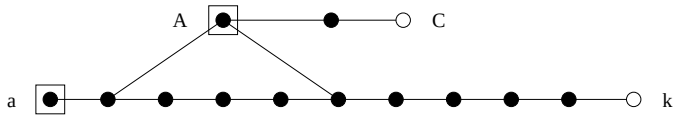


Fig. 1.2. Graph state for the $\wedge U$ pattern: input qubits A and a are boxed, output qubits are C and k, measured qubits are solid circles.

This graph also has a further interesting property, namely that all possible paths linking boundary vertices (inputs and outputs) are of even length (2, 6, 10 as it happens) as we can see in Figure (1.3).

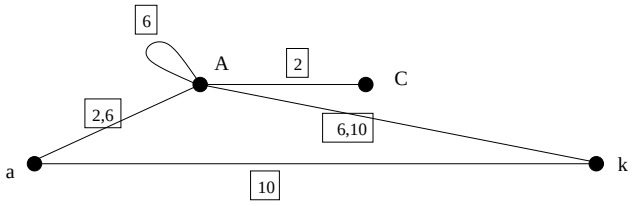


Fig. 1.3. Extreme paths in $\wedge U$ pattern: numbers represent the length of paths; solid circles represent the pattern input and output qubits.

Say that a path is *extreme* in a graph with inputs and outputs, if it goes from the boundary to itself; say that a graph with inputs and outputs is *even* if all its extreme paths are of even length; say that a pattern is even if its entanglement graph is even. We may then rephrase the last observation by saying that the pattern for $\wedge U$ is even. Patterns with an empty command sequence - among which one finds those implementing permutations over $\otimes^n \mathbb{C}^2$ - are even. Indeed, all their paths, therefore all their extreme paths, are of length zero, and zero is even. Furthermore, even patterns are closed under tensor product and composition. Indeed, the graph associated to the tensor product of two patterns is the juxtaposition of the components graphs, and therefore has a path space which is the disjoint sum of the path spaces of its components. On the other hand any extreme path in a composite pattern is a product of extreme paths of the components, and has therefore even length. This has a nice consequence:

Theorem 3 *Any unitary can be realized by a pattern with a 2-colorable underlying graph.*

Any unitary can be realized by a pattern obtained from the J-decomposition pattern, the $\wedge U$ pattern, and the permutation patterns, combined by tensor and composition. Any cycle in such a pattern is either a cycle internal to some basic pattern (which rules out J-decomposition and permutation patterns which have a linear entanglement graph), so living inside a $\wedge U$ pattern, therefore of length 6 and hence even, or the cycle is a product of extreme paths, therefore even, because all basic patterns are even, and by the above discussion, so is any combination of them. This completes the proof of the theorem. □

As said, 2-colorable entanglement graphs are interesting since purification protocols exist for their associated graph states, making them physically implementable in such a way that is robust against decoherence Dür et al. (2003). So it is good news that such robust implementations can be obtained in the one-way model for all unitary operators. It is important to note a similar result was known for a particular graph states so called cluster states Raussendorf et al. (2003). The underlying graph of the cluster states are rectangular grids and hence 2-colorable. However to realize an arbitrary unitary operator one needs to consume many qubits and perform many Pauli measurements to respect the underlying structure, whereas in our proposed implementation all the unnecessary measurements have been removed and yet a 2-colorable graph is obtained.

1.5 Measurement Calculus

We turn now to our structural result on the one-way model asserting that the key factorisation property, namely that entanglement can be done first, and then local measurements, can be reduced to confluence properties of a simple algebraic rewriting system Danos et al. (2007).

The expressions appearing as commands are all linear operators on Hilbert space. At first glance, the appropriate equality between commands is equality as operators. For the deterministic commands, the equality that we consider is indeed equality as operators. This equality implies equality in the denotational semantics. However, for measurement commands one needs a stricter definition for equality in order to be able to apply them as rewriting rules. Essentially we have to take into the account the effect of different branches that might result from the measurement process. The precise definition is below.

Definition 7 *Given two patterns $\mathcal{P}$ and $\mathcal{P}'$ we define $\mathcal{P} = \mathcal{P}'$ if and only if for any branch s, we have $A_s^{\mathcal{P}} = A_s^{\mathcal{P}'}$, where $A_s^{\mathcal{P}}$ and $A_s^{\mathcal{P}'}$ are the branch map A_s defined in Section 1.3.*

The first set of equations gives the means to propagate local Pauli corrections through the entangling operator E_{ij}.

$$E_{ij}X_i^s = X_i^s Z_j^s E_{ij} \quad (1.11)$$
$$E_{ij}X_j^s = X_j^s Z_i^s E_{ij} \quad (1.12)$$
$$E_{ij}Z_i^s = Z_i^s E_{ij} \quad (1.13)$$
$$E_{ij}Z_j^s = Z_j^s E_{ij} \quad (1.14)$$

These equations are easy to verify and are natural since E_{ij} belongs to the Clifford group, and therefore maps under conjugation the Pauli group to itself. Note that, despite the symmetry of the E_{ij} operator *qua* operator, we have to consider all the cases, since the rewrite system defined below does not allow one to rewrite E_{ij} to E_{ji}. If we did allow this the rewrite process could loop forever.

A second set of equations allows one to push corrections through measurements acting on the same qubit. Again there are two cases:

$${}^t[M_i^\alpha]^s X_i^r = {}^t[M_i^\alpha]^{s+r} \quad (1.15)$$
$${}^t[M_i^\alpha]^s Z_i^r = {}^{t+r}[M_i^\alpha]^s \quad (1.16)$$

These equations follow easily from equations (1.4) and (1.5). They express the fact that the measurements M_i^α are closed under conjugation by the Pauli group, very much like equations (1.11),(1.12),(1.13) and (1.14) express the fact that the Pauli group is closed under conjugation by the entanglements E_{ij}.

Define the following convenient abbreviations:

$$[M_i^\alpha]^s := {}^0[M_i^\alpha]^s \quad {}^t[M_i^\alpha] := {}^t[M_i^\alpha]^0 \quad M_i^\alpha := {}^0[M_i^\alpha]^0 \quad M_i^x := M_i^0 \quad M_i^y := M_i^{\frac{\pi}{2}}$$

Particular cases of the equations above are:

$$\begin{array}{lcl} M_i^x X_i^s & = & M_i^x \\ M_i^y X_i^s & = & [M_i^y]^s \quad = \quad {}^s[M_i^y] \quad = \quad M_i^y Z_i^s \end{array}$$

The first equation, follows from the fact that $-0 = 0$, so the X action on M_i^x is trivial; the second equation, is because $-\frac{\pi}{2}$ is equal $\frac{\pi}{2} + \pi$ modulo 2π, and therefore the X and Z actions coincide on M_i^y. So we obtain the following:

$$\begin{array}{lcll} {}^t[M_i^x]^s & = & {}^t[M_i^x] & (1.17) \\ {}^t[M_i^y]^s & = & {}^{s+t}[M_i^y] & (1.18) \end{array}$$

which we will use later to prove that patterns with measurements of the form M^x and M^y may only realize unitaries in the Clifford group.

We now define a set of rewrite rules, obtained by orienting the equations above. Recall that patterns are executed from right to left:

$$\begin{array}{lcll} E_{ij} X_i^s & \Rightarrow & X_i^s Z_j^s E_{ij} & EX \\ E_{ij} X_j^s & \Rightarrow & X_j^s Z_i^s E_{ij} & EX \\ E_{ij} Z_i^s & \Rightarrow & Z_i^s E_{ij} & EZ \\ E_{ij} Z_j^s & \Rightarrow & Z_j^s E_{ij} & EZ \\ {}^t[M_i^\alpha]^s X_i^r & \Rightarrow & {}^t[M_i^\alpha]^{s+r} & MX \\ {}^t[M_i^\alpha]^s Z_i^r & \Rightarrow & {}^{r+t}[M_i^\alpha]^s & MZ \end{array}$$

to which we need to add the *free commutation rules*, obtained when commands operate on disjoint sets of qubits:

$$\begin{array}{lcll} E_{ij} A_{\vec{k}} & \Rightarrow & A_{\vec{k}} E_{ij} & \text{where } A \text{ is not an entanglement} \\ A_{\vec{k}} X_i^s & \Rightarrow & X_i^s A_{\vec{k}} & \text{where } A \text{ is not a correction} \\ A_{\vec{k}} Z_i^s & \Rightarrow & Z_i^s A_{\vec{k}} & \text{where } A \text{ is not a correction} \end{array}$$

where $\vec{k}$ represent the qubits acted upon by command A, and are supposed to be distinct from i and j. Clearly these rules could be reversed since they hold as equations but we are orienting them this way in order to obtain termination. Condition (D) is easily seen to be preserved under rewriting.

Under rewriting, the computation space, inputs and outputs remain the same, and so do the entanglement commands. Measurements might be modified, but there is still the same number of them, and they still act on the same qubits. The only induced modifications concern local corrections and dependencies. If there was no dependency at the start, none will be created in the rewriting process.

In order to obtain rewrite rules, it was essential that the entangling command ($\wedge Z$) belongs to the normalizer of the Pauli group. The point is that the Pauli operators are the correction operators and they can be dependent, thus we can commute the entangling commands to the beginning without inheriting any dependency. Therefore the entanglement resource can indeed be prepared at the outset of the computation.

Write $\mathcal{P} \Rightarrow \mathcal{P}'$, respectively $\mathcal{P} \Rightarrow^\star \mathcal{P}'$, if both patterns have the same type, and one obtains the command sequence of $\mathcal{P}'$ from the command sequence of $\mathcal{P}$ by applying one, respectively any number, of the rewrite rules of the previous section. We say that $\mathcal{P}$ is *standard* if for no $\mathcal{P}'$, $\mathcal{P} \Rightarrow \mathcal{P}'$ and the procedure of writing a pattern to standard form is called standardization. We use the word "standardization" instead of the more usual "normalization" in order not to cause terminological confusion with the physicists' notion of normalization.

One of the most important results about the rewrite system is that it has the desirable properties of determinacy (confluence) and termination (standardization). In other words, we will show that for all $\mathcal{P}$, there exists a unique standard $\mathcal{P}'$, such that $\mathcal{P} \Rightarrow^\star \mathcal{P}'$. It is, of course, crucial that the standardization

process leaves the semantics of patterns invariant. This is the subject of the next simple, but important, proposition,

Proposition 8 *Whenever* $\mathcal{P} \Rightarrow^\star \mathcal{P}'$, $[\![\mathcal{P}]\!] = [\![\mathcal{P}']\!]$.

Proof. It is enough to prove it when $\mathcal{P} \Rightarrow \mathcal{P}'$. The first group of rewrites has been proved to be sound in the preceding subsections, while the free commutation rules are obviously sound. □

We now begin the main proof of this section. First, we prove termination.

Theorem 4 (Termination) *All rewriting sequences beginning with a pattern* $\mathcal{P}$ *terminate after finitely many steps. For our rewrite system, this implies that for all* $\mathcal{P}$ *there exist finitely many* $\mathcal{P}'$ *such that* $\mathcal{P} \Rightarrow^\star \mathcal{P}'$ *where the* $\mathcal{P}'$ *are standard.*

Proof. Suppose $\mathcal{P}$ has command sequence $A_n \dots A_1$; so the number of commands is n. Let $e \leq n$ be the number of E commands in $\mathcal{P}$. As we have noted earlier, this number is invariant under $\Rightarrow$. Moreover E commands in $\mathcal{P}$ can be ordered by increasing depth, read from right to left, and this order, written $<_E$, is also invariant, since EE commutations are forbidden explicitly in the free commutation rules.

Define the following depth function d on E and C commands in $\mathcal{P}$:

$$d(A_i) = \begin{cases} i & \text{if } A_i = E_{jk} \\ n - i & \text{if } A_i = C_j \end{cases}$$

Define further the following sequence of length e, $d_E(\mathcal{P})(i)$ is the depth of the E-command of rank i according to $<_E$. By construction this sequence is strictly increasing. Finally, we define the measure $m(\mathcal{P}) := (d_E(\mathcal{P}), d_C(\mathcal{P}))$ with:

$$d_C(\mathcal{P}) \quad = \quad \textstyle\sum_{C \in \mathcal{P}} d(C)$$

We claim the measure we just defined decreases lexicographically under rewriting, in other words $\mathcal{P} \Rightarrow \mathcal{P}'$ implies $m(\mathcal{P}) > m(\mathcal{P}')$, where $<$ is the lexicographic ordering on $\mathbb{N}^{e+1}$.

To clarify these definitions, consider the following example. Suppose $\mathcal{P}$'s command sequence is of the form $EXZE$, then $e = 2$, $d_E(\mathcal{P}) = (1, 4)$, and $m(\mathcal{P}) = (1, 4, 3)$. For the command sequence EEX we get that $e = 2$, $d_E(\mathcal{P}) = (2, 3)$ and $m(\mathcal{P}) = (2, 3, 2)$. Now, if one considers the rewrite $EEX \Rightarrow EXZE$, the measure of the left hand side is $(2, 3, 2)$, while the measure of the right hand side, as said, is $(1, 4, 3)$, and indeed $(2, 3, 2) > (1, 4, 3)$. Intuitively the reason is clear: the Cs are being pushed to the left, thus decreasing the depths of Es, and concomitantly, the value of d_E.

Let us now consider all cases starting with an EC rewrite. Suppose the E command under rewrite has depth d and rank i in the order $<_E$. Then all Es of smaller rank have same depth in the right hand side, while E has now depth $d-1$ and still rank i. So the right hand side has a strictly smaller measure. Note that when $C = X$, because of the creation of a Z (see the example above), the last element of $m(\mathcal{P})$ may increase, and for the same reason all elements of index $j > i$ in $d_E(\mathcal{P})$ may increase. This is why we are working with a lexicographical ordering.

Suppose now one does an MC rewrite, then $d_C(\mathcal{P})$ strictly decreases, since one correction is absorbed, while all E commands have equal or smaller depths. Again the measure strictly decreases.

Next, suppose one does an EA rewrite, and the E command under rewrite has depth d and rank i. Then it has depth $d-1$ in the right hand side, and all other E commands have invariant depths, since we forbade the case when A is itself an E. It follows that the measure strictly decreases.

Finally, upon an AC rewrite, all E commands have invariant depth, except possibly one which has smaller depth in the case $A = E$, and $d_C(\mathcal{P})$ decreases strictly because we forbade the case where $A = C$. Again the claim follows.

So all rewrites decrease our ordinal measure, and therefore all sequences of rewrites are finite, and since the system is finitely branching (there are no more than n possible single step rewrites on a given sequence of length n), we get the statement of the theorem. □

The next theorem establishes the important determinacy property and furthermore shows that the standard patterns have a certain canonical form which we call the NEMC form. The precise definition is:

Definition 9 *A pattern has a* NEMC *form if its commands occur in the order of Ns first, then Es , then Ms, and finally Cs.*

We will usually just say "EMC" form since we can assume that all the auxiliary qubits are prepared in the $|+\rangle$ state and we usually just elide these N commands.

Theorem 5 (Confluence) *For all $\mathcal{P}$, there exists a unique standard $\mathcal{P}'$, such that $\mathcal{P} \Rightarrow^\star \mathcal{P}'$, and $\mathcal{P}'$ is in EMC form.*

Proof. Since the rewriting system is terminating, confluence follows from local confluence by Newman's lemma, see, for example, Barendregt (1984). This means that whenever two rewrite rules can be applied to a term t yielding t_1 and t_2, one can rewrite both t_1 and t_2 to a common third term t_3, possibly in many steps. Then the uniqueness of the standard form is an immediate consequence.

In order to prove the local confluence we look for critical pairs, that is occurrences of three successive commands where two rules can be applied simultaneously. One finds that there are only five types of critical pairs, of these the three involve the N command, these are of the form: NMC, NEC and NEM; and the remaining two are: $E_{ij}M_kC_k$ with i, j and k all distinct, $E_{ij}M_kC_l$ with k and l distinct. In all cases local confluence is easily verified.

Suppose now $\mathcal{P}'$ does not satisfy the EMC form conditions. Then, either there is a pattern EA with A not of type E, or there is a pattern AC with A not of type C. In the former case, E and A must operate on overlapping qubits, else one may apply a free commutation rule, and A may not be a C since in this case one may apply an EC rewrite. The only remaining case is when A is of type M, overlapping E's qubits, but this is what condition (D1) forbids, and since (D1) is preserved under rewriting, this contradicts the assumption. The latter case is even simpler. □

We have shown that under rewriting any pattern can be put in EMC form, which is what we wanted. We actually proved more, namely that the standard form obtained is unique. However, one has to be a bit careful about the significance of this additional piece of information. Note first that uniqueness is obtained because we dropped the CC and EE free commutations, thus having a rigid notion of command sequence. One cannot put them back as rewrite rules, since they obviously ruin termination and uniqueness of standard forms.

A reasonable thing to do, would be to take this set of equations as generating an equivalence relation on command sequences, call it $\equiv$, and hope to strengthen the results obtained so far, by proving that all reachable standard forms are equivalent.

But this is too naive a strategy, since $E_{12}X_1X_2 \equiv E_{12}X_2X_1$, and:

$$\begin{array}{rcl} E_{12}X_1^sX_2^t & \Rightarrow^\star & X_1^sZ_2^sX_2^tZ_1^tE_{12} \\ & \equiv & X_1^sZ_1^tZ_2^sX_2^tE_{12} \end{array}$$

obtaining an expression which is not symmetric in 1 and 2. To conclude, one has to extend $\equiv$ to include the additional equivalence $X_1^sZ_1^t \equiv Z_1^tX_1^s$, which fortunately is sound since these two operators are equal up to a global phase. Thus, these are all equivalent in our semantics of patterns. We summarize this discussion as follows.

Definition 10 *We define an equivalence relation $\equiv$ on patterns by taking all the rewrite rules as equations and adding the equation $X_1^sZ_1^t \equiv Z_1^tX_1^s$ and generating the smallest equivalence relation.*

With this definition we can state the following proposition.

Proposition 11 *All patterns that are equivalent by $\equiv$ are equal in the denotational semantics.*

This $\equiv$ relation preserves both the type (the (V, I, O) triple) and the underlying entanglement graph. So clearly semantic equality does not entail equality up to $\equiv$. In fact, by composing teleportation patterns one obtains infinitely many patterns for the identity which are all different up to $\equiv$. One may wonder whether two patterns with same semantics, type and underlying entanglement graph are necessarily

equal up to $\equiv$. This is not true either. One has $J(\alpha)J(0)J(\beta) = J(\alpha+\beta) = J(\beta)J(0)J(\alpha)$ (where $J(\alpha)$ is defined in Section 1.4), and this readily gives a counter-example.

We can now formally describe a simple standardization algorithm.

Algorithm 1 Input*: A pattern $\mathcal{P}$ on $|V| = N$ qubits with command sequence $A_M \cdots A_1$.*
Output*: An equivalent pattern $\mathcal{P}'$ in NEMC form.*

(i) *Commute all the preparation commands (new qubits) to the right side.*
(ii) *Commute all the correction commands to the left side using the EC and MC rewriting rules.*
(iii) *Commute all the entanglement commands to the right side after the preparation commands.*

Note that since each qubit can be entangled with at most $N-1$ other qubits, and can be measured or corrected only once, we have $O(N^2)$ entanglement commands and $O(N)$ measurement commands. According to the definiteness condition, no command acts on a qubit not yet prepared, hence the first step of the above algorithm is based on trivial commuting rules; the same is true for the last step as no entanglement command can act on a qubit that has been measured. Both steps can be done in $O(N^2)$ time. The real complexity of the algorithm comes from the second step and the EX commuting rule. In the worst case scenario, commuting an X correction to the left might create $O(N^2)$ other Z corrections, each of which has to be commuted to the left themselves. Thus one can have at most $O(N^3)$ new corrections, each of which has to be commuted past $O(N^2)$ measurement or entanglement commands. Therefore the second step, and hence the algorithm, has a worst case complexity of $O(N^5)$ time.

We conclude this subsection by emphasizing the importance of the EMC form. Since the entanglement can always be done first, we can always derive the entanglement resource needed for the whole computation right at the beginning. After that only local operations will be performed. This will separate the analysis of entanglement resource requirements from the classical control. Furthermore, this makes it possible to extract the maximal parallelism for the execution of the pattern since the necessary dependecies are explicitly expressed, see the example in section 1.6 for further discussion. Finally, the EMC form provides us with tools to prove general theorems about patterns, such as the fact that they always compute cptp-maps and the expressiveness theorems of section 1.5.2.

1.5.1 Signal shifting

One can extend the calculus to include the signal shifting command S_i^t to dispose of dependencies induced by the Z-action Danos et al. (2007), and obtain sometimes standard patterns with smaller computational depth complexity Broadbent and Kashefi (2009).

$$\begin{array}{lcl}
{}^t[M_i^\alpha]^s & \Rightarrow & S_i^t[M_i^\alpha]^s \\
X_j^s S_i^t & \Rightarrow & S_i^t X_j^{s[t+s_i/s_i]} \\
Z_j^s S_i^t & \Rightarrow & S_i^t Z_j^{s[t+s_i/s_i]} \\
{}^t[M_j^\alpha]^s S_i^r & \Rightarrow & S_i^{r\ t[r+s_i/s_i]}[M_j^\alpha]^{s[r+s_i/s_i]} \\
S_i^s S_j^t & \Rightarrow & S_j^t S_i^{s[t+s_j/s_j]}
\end{array}$$

where $s[t/s_i]$ denotes the substitution of s_i with t in s, s, t being signals. Note that when we write a t explicitly on the upper left of an M, we mean that $t \neq 0$. The first additional rewrite rule was already introduced as equation (1.6), while the other ones merely propagate the signal shift. Clearly one can dispose of S_i^t when it hits the end of the pattern command sequence. We will refer to this new set of rules as $\Rightarrow_S$. Note that we always apply first the standardization rules and then signal shifting, hence we do not need any commutation rule for E and S commands.

It is important to note that both Theorem 4 and 5 still hold for this extended rewriting system. In order to prove termination one can start with the EMC form and then adapt the proof of Theorem 4 by defining a depth function for a signal shift similar to the depth of a correction command. As with the correction, signal shifts can also be commuted to the left hand side of a command sequence. Now our measure can be modified to account for the new signal shifting terms and shown to be decreasing under each step of signal shifting. Confluence can be also proved from local confluence using again Newman's

Lemma Barendregt (1984). One typical critical pair is ${}^t[M_j^\alpha]S_i^s$ where i appears in the domain of signal t and hence the signal shifting command S_i^s will have an effect on the measurement. Now there are two possible ways to rewrite this pair, first, commute the signal shifting command and then replace the left signal of the measurement with its own signal shifting command:

$$\begin{aligned} {}^t[M_j^\alpha]\, S_i^s &\Rightarrow S_i^s\, {}^{t+s}[M_j^\alpha] \\ &\Rightarrow S_i^s\, S_j^{s+t}\, M_j^\alpha \end{aligned}$$

The other way is to first replace the left signal of the measurement and then commute the signal shifting command:

$$\begin{aligned} {}^t[M_j^\alpha]\, S_i^s &\Rightarrow S_j^t\, M_j^\alpha\, S_i^s \\ &\Rightarrow S_j^t\, S_i^s\, M_j^\alpha \end{aligned}$$

Now one more step of rewriting on the last equation will give us the same result for both choices.

$$S_j^t\, S_i^s\, M_j^\alpha \Rightarrow S_i^s\, S_j^{s+t}\, M_j^\alpha$$

All other critical terms can be dealt with similarly.

1.5.2 The no dependency theorems

From standardization we can also infer results related to dependencies Danos et al. (2007). We start with a simple observation which is a direct consequence of standardization. In what follows the *computational depth complexity* is defined to be the number of measurement rounds plus one final correction round. More details on depth complexity can be found in Broadbent and Kashefi (2009).

Lemma 12 *Let $\mathcal{P}$ be a pattern implementing some cptp-maps T, and suppose $\mathcal{P}$'s command sequence has measurements only of the M^x and M^y kind, then U has a standard implementation, having only independent measurements, all being of the M^x and M^y kind (therefore of computational depth complexity at most 2).*

Proof. Write $\mathcal{P}'$ for the standard pattern associated to $\mathcal{P}$. By equations (1.17) and (1.18), the X-actions can be eliminated from $\mathcal{P}'$, and then Z-actions can be eliminated by using the extended calculus. The final pattern still implements T, has no longer any dependent measurements, and has therefore computational depth complexity at most 2. □

Theorem 6 *Let U be a unitary operator, then U is in the Clifford group iff there exists a pattern $\mathcal{P}$ implementing U, having measurements only of the M^x and M^y kind.*

Proof. The "only if" direction is easy, since we have seen in the example section, standard patterns for $\wedge X$, H and $P(\frac{\pi}{2})$ which had only independent M^x and M^y measurements. Hence any Clifford operator can be implemented by a combination of these patterns. By the lemma above, we know we can actually choose these patterns to be standard.

For the "if" direction, we prove that U belongs to the normalizer of the Pauli group, and hence by definition to the Clifford group. In order to do so we use the standard form of $\mathcal{P}$ written as $\mathcal{P}' = C_{\mathcal{P}'} M_{\mathcal{P}'} E_{\mathcal{P}'}$ which still implements U, and has only M^x and M^y measurements. Recall that, because of equations (1.17) and (1.18), these measurements are independent.

Let i be an input qubit, and consider the pattern $\mathcal{P}'' = \mathcal{P}' C_i$, where C_i is either X_i or Z_i. Clearly $\mathcal{P}''$ implements UC_i. First, one has:

$$C_{\mathcal{P}'} M_{\mathcal{P}'} E_{\mathcal{P}'} C_i \quad \Rightarrow_{EC}^{\star} \quad C_{\mathcal{P}'} M_{\mathcal{P}'} C' E_{\mathcal{P}'}$$

for some *non-dependent* sequence of corrections C', which, up to free commutations can be written uniquely as $C'_O C''$, where C'_O applies on output qubits, and therefore commutes to $M_{\mathcal{P}'}$, and C'' applies on non-output qubits (which are therefore all measured in $M_{\mathcal{P}'}$). So, by commuting C'_O both through $M_{\mathcal{P}'}$ and $C_{\mathcal{P}'}$ (up to a global phase), one gets:

$$C_{\mathcal{P}'} M_{\mathcal{P}'} C' E_{\mathcal{P}'} \quad \Rightarrow^{\star} \quad C'_O C_{\mathcal{P}'} M_{\mathcal{P}'} C'' E_{\mathcal{P}'}$$

Using equations (1.17), (1.18), and the extended calculus to eliminate the remaining Z-actions, one gets:

$$M_{\mathcal{P}'}C'' \quad \Rightarrow^{\star}_{MC,S} \quad SM_{\mathcal{P}'}$$

for some product $S = \prod_{\{j\in J\}} S_j^1$ of constant shifts, applying to some subset J of the non-output qubits. Note that we have used the trivial equations $Z_i^{a+1} = Z_i Z_i^a$ and $X_i^{a+1} = X_i X_i^a$. Therefore we have

$$\begin{array}{lll} C'_O C_{\mathcal{P}'} M_{\mathcal{P}'} C'' E_{\mathcal{P}'} & \Rightarrow^{\star}_{MC,S} & C'_O C_{\mathcal{P}'} S M_{\mathcal{P}'} E_{\mathcal{P}'} \\ & \Rightarrow^{\star} & C'_O C''_O C_{\mathcal{P}'} M_{\mathcal{P}'} E_{\mathcal{P}'} \end{array}$$

where C''_O is a further constant correction obtained by signal shifting $C_{\mathcal{P}'}$ with S. This proves that $\mathcal{P}''$ also implements $C'_O C''_O U$, and therefore $UC_i = C'_O C''_O U$ which completes the proof, since $C'_O C''_O$ is a non dependent correction. □

The "only if" part of this theorem already appears in previous work (Raussendorf et al., 2003, p.18). The "if" part can be construed as an internalization of the argument implicit in the proof of the Gottesman-Knill theorem (Nielsen and Chuang, 2000, p.464).

We can further prove that dependencies are crucial for the universality of the model. Observe first that if a pattern has no measurements, and hence no dependencies, then it follows from (D2) that $V = O$, *i.e.*, all qubits are outputs. Therefore computation steps involve only X, Z and $\wedge Z$, and it is not surprising that they compute a unitary which is in the Clifford group. The general argument essentially consists in showing that when there are measurements, but still no dependencies, then the measurements play no part in the result.

Theorem 7 *Let $\mathcal{P}$ be a pattern implementing some unitary U, and suppose $\mathcal{P}$'s command sequence doesn't have any dependencies, then U is in the Clifford group.*

Proof. Write $\mathcal{P}'$ for the standard pattern associated to $\mathcal{P}$. Since rewriting is sound, $\mathcal{P}'$ still implements U, and since rewriting never creates any dependency, it still has no dependencies. In particular, the corrections one finds at the end of $\mathcal{P}'$, call them C, bear no dependencies. Erasing them from $\mathcal{P}'$, results in a pattern $\mathcal{P}''$ which is still standard, still deterministic, and implementing $U' := C^{\dagger}U$.

Now how does the pattern $\mathcal{P}''$ run on some input ϕ ? First $\phi \otimes |+\ldots+\rangle$ goes by the entanglement phase to some $\psi \in \mathfrak{H}_V$, and is then subjected to a sequence of independent 1-qubit measurements. Pick a basis $\mathbf{B}$ spanning the Hilbert space generated by the non-output qubits $\mathfrak{H}_{V\setminus O}$ and associated to this sequence of measurements.

Since $\mathfrak{H}_V = \mathfrak{H}_O \otimes \mathfrak{H}_{V\setminus O}$ and $\mathfrak{H}_{V\setminus O} = \oplus_{\phi_b\in\mathcal{B}}[\phi_b]$, where $[\phi_b]$ is the linear subspace generated by ϕ_b, by distributivity, ψ uniquely decomposes as:

$$\psi = \textstyle\sum_{\phi_b\in\mathcal{B}} x_b \otimes \phi_b$$

where ϕ_b ranges over $\mathcal{B}$, and $x_b \in \mathfrak{H}_O$. Now since $\mathcal{P}''$ is deterministic, there exists an x, and scalars λ_b such that $x_b = \lambda_b x$. Therefore ψ can be written $x \otimes \psi'$, for some ψ'. It follows in particular that the output of the computation will still be x (up to a scalar), no matter what the actual measurements are. One can therefore choose them to be all of the M^x kind, and by the preceding theorem U' is in the Clifford group, and so is $U = CU'$, since C is a Pauli operator. □

From this section, we conclude in particular that any universal set of patterns has to include dependencies (by the preceding theorem), and also needs to use measurements M^{α} where $\alpha \neq 0$ modulo $\frac{\pi}{2}$ (by the theorem before). This is indeed the case for the universal set $\mathcal{J}(\alpha)$ and $\wedge\mathcal{Z}$.

1.6 MBQC - Examples

In this section we develop some examples illustrating pattern composition, pattern standardization, and signal shifting. More examples can be found in the reference paper Raussendorf et al. (2003). To combine patterns one needs to rename their qubits as we already noted. We use the following concrete notation: if $\mathcal{P}$ is a pattern over $\{1,\ldots,n\}$, and f is an injection, we write $\mathcal{P}(f(1),\ldots,f(n))$ for the same pattern with qubits renamed according to f. We also write $\mathcal{P}_2 \circ \mathcal{P}_1$ for pattern composition, in order to make it more readable.

Teleportation.

Consider the composite pattern $\mathcal{J}(\beta)(2,3) \circ \mathcal{J}(\alpha)(1,2)$ with computation space $\{1,2,3\}$, inputs $\{1\}$, and outputs $\{3\}$. We run our standardization procedure so as to obtain an equivalent standard pattern. In what follows boxes are used to indicate where rewriting occurs:

$$\begin{array}{rcl} \mathcal{J}(\beta)(2,3) \circ \mathcal{J}(\alpha)(1,2) & = & X_3^{s_2} M_2^{-\beta} \boxed{E_{23} X_2^{s_1}} M_1^{-\alpha} E_{12} \\ & \Rightarrow_{EX} & X_3^{s_2} \boxed{M_2^{-\beta} X_2^{s_1}} Z_3^{s_1} M_1^{-\alpha} E_{23} E_{12} \\ & \Rightarrow_{MX} & X_3^{s_2} Z_3^{s_1} [M_2^{-\beta}]^{s_1} M_1^{-\alpha} E_{23} E_{12} \end{array}$$

Let us call the pattern just obtained $\mathcal{J}(\alpha, \beta)$. If we take as a special case $\alpha = \beta = 0$, we get:

$$X_3^{s_2} Z_3^{s_1} M_2^x M_1^x E_{23} E_{12}$$

and since we know that $\mathcal{J}(0)$ implements H and $H^2 = I$, we conclude that this pattern implements the identity, or in other words it teleports qubit 1 to qubit 3. As it happens, this pattern obtained by self-composition, is the same as the one given in the reference paper (Raussendorf et al., 2003, p.14).

x-rotation.

Here is the reference implementation of an x-rotation (Raussendorf et al., 2003, p.17), $R_x(\alpha)$:

$$X_3^{s_2} Z_3^{s_1} [M_2^{-\alpha}]^{s_1} M_1^x E_{23} E_{12}$$

with type $\{1,2,3\}$, $\{1\}$, and $\{3\}$. There is a natural question which one might call the recognition problem, namely how does one know this is implementing $R_x(\alpha)$? Of course there is the brute force answer to that, which we applied to compute our simpler patterns, and which consists in computing down all the four possible branches generated by the measurements at qubits 1 and 2. Another possibility is to use the stabilizer formalism as explained in the reference paper Raussendorf et al. (2003). Yet another possibility is to use *pattern composition*, as we did before, and this is what we are going to do.

We know that $R_x(\alpha) = J(\alpha)H$ up to a global phase, hence the composite pattern $\mathcal{J}(\alpha)(2,3) \circ \mathcal{H}(1,2)$ implements $R_x(\alpha)$. Now we may standardize it:

$$\begin{array}{rcl} \mathcal{J}(\alpha)(2,3) \circ \mathcal{H}(1,2) & = & X_3^{s_2} M_2^{-\alpha} \boxed{E_{23} X_2^{s_1}} M_1^x E_{12} \\ & \Rightarrow_{EX} & X_3^{s_2} Z_3^{s_1} \boxed{M_2^{-\alpha} X_2^{s_1}} M_1^x E_{23} E_{12} \\ & \Rightarrow_{MX} & X_3^{s_2} Z_3^{s_1} [M_2^{-\alpha}]^{s_1} M_1^x E_{23} E_{12} \end{array}$$

obtaining exactly the implementation above. Since our calculus preserves the semantics, we deduce that the implementation is correct.

z-rotation.

Now, we have a method here for synthesizing further implementations. Let us replay it with another rotation $R_z(\alpha)$. Again we know that $R_z(\alpha) = HR_x(\alpha)H$, and we already know how to implement both components H and $R_x(\alpha)$.

So we start with the pattern $\mathcal{H}(4,5) \circ \mathcal{R}_x(\alpha)(2,3,4) \circ \mathcal{H}(1,2)$ and standardize it:

$$\begin{array}{rcl} \mathcal{H}(4,5) \circ \mathcal{R}_x(\alpha)(2,3,4) \circ \mathcal{H}(1,2) & = & \mathcal{H}(4,5) X_4^{s_3} Z_4^{s_2} [M_3^\alpha]^{1+s_2} M_2^x E_{34} \boxed{E_{23} X_2^{s_1}} M_1^x E_{12} \\ & \Rightarrow_{EX} & \mathcal{H}(4,5) X_4^{s_3} Z_4^{s_2} [M_3^\alpha]^{1+s_2} M_2^x X_2^{s_1} \boxed{E_{34} Z_3^{s_1}} M_1^x E_{123} \\ & \Rightarrow_{EZ} & \mathcal{H}(4,5) X_4^{s_3} Z_4^{s_2} [M_3^\alpha]^{1+s_2} Z_3^{s_1} \boxed{M_2^x X_2^{s_1}} M_1^x E_{1234} \\ & \Rightarrow_{MX} & \mathcal{H}(4,5) X_4^{s_3} Z_4^{s_2} \boxed{[M_3^\alpha]^{1+s_2} Z_3^{s_1}} M_2^x M_1^x E_{1234} \\ & \Rightarrow_{MZ} & X_5^{s_4} M_4^x \boxed{E_{45} X_4^{s_3}} Z_4^{s_2 s_1} [M_3^\alpha]^{1+s_2} M_2^x M_1^x E_{1234} \\ & \Rightarrow_{EX} & X_5^{s_4} Z_5^{s_3} \boxed{M_4^x X_4^{s_3}} Z_4^{s_2 s_1} [M_3^\alpha]^{1+s_2} M_2^x M_1^x E_{12345} \\ & \Rightarrow_{MX} & X_5^{s_4} Z_5^{s_3} \boxed{[M_4^x]^{s_3} Z_4^{s_2}}^{s_1} [M_3^\alpha]^{1+s_2} M_2^x M_1^x E_{12345} \\ & \Rightarrow_{MZ} & X_5^{s_4} Z_5^{s_3 s_2} [M_4^x]^{s_3 s_1} [M_3^\alpha]^{1+s_2} M_2^x M_1^x E_{12345} \end{array}$$

To aid reading $E_{23}E_{12}$ is shortened to E_{123}, $E_{12}E_{23}E_{34}$ to E_{1234}, and ${}^t[M_i^\alpha]^{1+s}$ is used as shorthand for ${}^t[M_i^{-\alpha}]^s$.

Here for the first time, we see MZ rewritings, inducing the Z-action on measurements. The resulting standardized pattern can therefore be rewritten further using the extended calculus:

$$X_5^{s_4} Z_5^{s_3 s_2} [M_4^x]^{s_3 s_1} [M_3^\alpha]^{1+s_2} M_2^x M_1^x E_{12345} \quad \Rightarrow_S \quad X_5^{s_2+s_4} Z_5^{s_1+s_3} M_4^x [M_3^\alpha]^{1+s_2} M_2^x M_1^x E_{12345}$$

obtaining the pattern given in the reference paper (Raussendorf et al., 2003, p.5).

However, just as in the case of the R_x rotation, we also have $R_z(\alpha) = HJ(\alpha)$ up to a global phase, hence the pattern $\mathcal{H}(2,3)\mathcal{J}(\alpha)(1,2)$ also implements $R_z(\alpha)$, and we may standardize it:

$$\begin{array}{rcl} \mathcal{H}(2,3) \circ \mathcal{J}(\alpha)(1,2) & = & X_3^{s_2} M_2^x \boxed{E_{23} X_2^{s_1}} M_1^{-\alpha} E_{12} \\ & \Rightarrow_{EX} & X_3^{s_2} Z_3^{s_1} \boxed{M_2^x X_2^{s_1}} M_1^{-\alpha} E_{123} \\ & \Rightarrow_{MX} & X_3^{s_2} Z_3^{s_1} M_2^x M_1^{-\alpha} E_{123} \end{array}$$

obtaining a 3 qubit standard pattern for the z-rotation, which is simpler than the preceding one, because it is based on the $\mathcal{J}(\alpha)$ generators. Since the z-rotation $R_z(\alpha)$ is the same as the phase operator:

$$P(\alpha) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\alpha} \end{pmatrix}$$

up to a global phase, we also obtain with the same pattern an implementation of the phase operator. In particular, if $\alpha = \frac{\pi}{2}$, using the extended calculus, we get the following pattern for $P(\frac{\pi}{2})$: $X_3^{s_2} Z_3^{s_1+1} M_2^x M_1^y E_{123}$.

General rotation.

The realization of a general rotation based on the Euler decomposition of rotations as $R_x(\gamma)R_z(\beta)R_x(\alpha)$, would results in a 7 qubit pattern. We get a 5 qubit implementation based on the $J(\alpha)$ decomposition Danos et al. (2005):

$$R(\alpha,\beta,\gamma) \quad = \quad J(0)J(-\alpha)J(-\beta)J(-\gamma)$$

(The parameter angles are inverted to make the computation below more readable.) The extended standardization procedure yields:

$$\begin{array}{rcl} \mathcal{J}(0)(4,5)\mathcal{J}(-\alpha)(3,4)\mathcal{J}(-\beta)(2,3)\mathcal{J}(-\gamma)(1,2) & = & X_5^{s_4} M_4^0 E_{45} X_4^{s_3} M_3^\alpha E_{34} X_3^{s_2} M_2^\beta \boxed{E_{23} X_2^{s_1}} M_1^\gamma E_{12} \\ & \Rightarrow_{EX} & X_5^{s_4} M_4^0 E_{45} X_4^{s_3} M_3^\alpha E_{34} X_3^{s_2} \boxed{M_2^\beta X_2^{s_1}} Z_3^{s_1} M_1^\gamma E_{123} \\ & \Rightarrow_{MX} & X_5^{s_4} M_4^0 E_{45} X_4^{s_3} M_3^\alpha \boxed{E_{34} X_3^{s_2} Z_3^{s_1}} [M_2^\beta]^{s_1} M_1^\gamma E_{123} \\ & \Rightarrow_{EXZ} & X_5^{s_4} M_4^0 E_{45} X_4^{s_3} \boxed{M_3^\alpha X_3^{s_2} Z_3^{s_1}} Z_4^{s_2} [M_2^\beta]^{s_1} M_1^\gamma E_{1234} \\ & \Rightarrow_{MXZ} & X_5^{s_4} M_4^0 \boxed{E_{45} X_4^{s_3} Z_4^{s_2}}^{s_1} [M_3^\alpha]^{s_2} [M_2^\beta]^{s_1} M_1^\gamma E_{1234} \\ & \Rightarrow_{EXZ} & X_5^{s_4} \boxed{M_4^0 X_4^{s_3} Z_4^{s_2}} Z_5^{s_3 s_1} [M_3^\alpha]^{s_2} [M_2^\beta]^{s_1} M_1^\gamma E_{12345} \\ & \Rightarrow_{MXZ} & X_5^{s_4} Z_5^{s_3 s_2} [M_4^0]^{s_1} [M_3^\alpha]^{s_2} [M_2^\beta]^{s_1} M_1^\gamma E_{12345} \\ & \Rightarrow_S & X_5^{s_2+s_4} Z_5^{s_1+s_3} M_4^0 [M_3^\alpha]^{s_2} [M_2^\beta]^{s_1} M_1^\gamma E_{12345} \end{array}$$

CNOT ($\wedge X$).

This is our first example with two inputs and two outputs. We use here the trivial pattern $\mathcal{I}$ with computation space $\{1\}$, inputs $\{1\}$, outputs $\{1\}$, and empty command sequence, which implements the identity over $\mathfrak{H}_1$.

One has $\wedge X = (I \otimes H) \wedge Z (I \otimes H)$, so we get a pattern using 4 qubits over $\{1,2,3,4\}$, with inputs

$\{1,2\}$, and outputs $\{1,4\}$, where one notices that inputs and outputs intersect on the control qubit $\{1\}$:

$$\begin{array}{lll}(\mathcal{I}(1)\otimes\langle(3,4)\rangle)\wedge\mathcal{Z}(1,3)(\mathcal{I}(1)\otimes\langle(2,3)\rangle) & = & X_4^{s_3}M_3^xE_{34}\boxed{E_{13}X_3^{s_2}}M_2^xE_{23}\\ & \Rightarrow_{EX} & X_4^{s_3}Z_1^{s_2}M_3^x\boxed{E_{34}X_3^{s_2}}M_2^xE_{13}E_{23}\\ & \Rightarrow_{EX} & X_4^{s_3}Z_4^{s_2}Z_1^{s_2}\boxed{M_3^xX_3^{s_2}}M_2^xE_{13}E_{23}E_{34}\\ & \Rightarrow_{MX} & X_4^{s_3}Z_4^{s_2}Z_1^{s_2}M_3^xM_2^xE_{13}E_{23}E_{34}\end{array}$$

Note that, in this case, we are not using the E_{1234} abbreviation, because the underlying structure of entanglement is not a chain. This pattern was already described in Aliferis and Leung's paper Aliferis and Leung (2004). In their original presentation the authors actually use an explicit identity pattern (using the teleportation pattern $\mathcal{J}(0,0)$ presented above), but we know from the careful presentation of composition that this is not necessary.

GHZ.

We present now a family of patterns preparing the GHZ entangled states $|0\ldots0\rangle+|1\ldots1\rangle$. One has:

$$\mathrm{GHZ}(n) \quad = \quad (H_n\wedge Z_{n-1n}\ldots H_2\wedge Z_{12})|+\ldots+\rangle$$

and by combining the patterns for $\wedge Z$ and H, we obtain a pattern with computation space $\{1,2,2',\ldots,n,n'\}$, no inputs, outputs $\{1,2',\ldots,n'\}$, and the following command sequence:

$$X_{n'}^{s_n}M_n^xE_{nn'}E_{(n-1)'n}\ldots X_{2'}^{s_2}M_2^xE_{22'}E_{12}$$

With this form, the only way to run the pattern is to execute all commands in sequence. The situation changes completely, when we bring the pattern to extended standard form:

$$\begin{array}{lll} & X_{n'}^{s_n}M_n^xE_{nn'}E_{(n-1)'n}\ldots X_{3'}^{s_3}M_3^xE_{33'}\boxed{E_{2'3}X_{2'}^{s_2}}M_2^xE_{22'}E_{12}\\ \Rightarrow & X_{n'}^{s_n}X_{2'}^{s_2}M_n^xE_{nn'}E_{(n-1)'n}\ldots X_{3'}^{s_3}\boxed{M_3^xZ_3^{s_2}}M_2^xE_{33'}E_{2'3}E_{22'}E_{12}\\ \Rightarrow & X_{n'}^{s_n}X_{2'}^{s_2}M_n^xE_{nn'}E_{(n-1)'n}\ldots X_{3'}^{s_3\,s_2}[M_3^x]M_2^xE_{33'}E_{2'3}E_{22'}E_{12}\\ \Rightarrow^\star & X_{n'}^{s_n}\ldots X_{3'}^{s_3}X_{2'}^{s_2\,s_{n-1}}[M_n^x]\ldots{}^{s_2}[M_3^x]M_2^xE_{nn'}E_{(n-1)'n}\ldots E_{33'}E_{2'3}E_{22'}E_{12}\\ \Rightarrow_S & X_{n'}^{s_2+s_3+\cdots+s_n}\ldots X_{3'}^{s_2+s_3}X_{2'}^{s_2}M_n^x\ldots M_3^xM_2^xE_{nn'}E_{(n-1)'n}\ldots E_{33'}E_{2'3}E_{22'}E_{12}\end{array}$$

All measurements are now independent of each other, it is therefore possible after the entanglement phase, to do all of them in one round, and in a subsequent round to do all local corrections. In other words, the obtained pattern has constant computational depth complexity 2.

1.7 Other MBQC Models

There are several other approaches to measurement-based computation as we have mentioned in the introduction. However, it is only for the one-way model that the importance of having all the entanglement in front has been emphasized. For example, Gottesman and Chuang describe computing with teleportation in the setting of the circuit model and hence the computation is very sequential Gottesman and Chuang (1999). What we will do is to give a general treatment of a variety of measurement-based models in the setting of our calculus. More precisely we would like to know other potential definitions for commands N, E, M and C that lead to a model that still satisfies the properties of: (i) being closed under composition; (ii) universality and (iii) standardization.

Moreover we are interested in obtaining a compositional embedding of these models into a single *one-qubit* measurement-based model. The teleportation and state transfer models can indeed be embedded into the one-way model. There is, however, a new model, the Pauli model which is motivated by considerations of fault tolerance Raussendorf et al. (2004); Danos and Kashefi (2005); Danos et al. (2006). The Pauli model can be embedded into a slight generalization of the one-way model called the phase model Danos et al. (2007). The one-way model will trivially embed in the phase model so by composition all the measurement-based models will embed in the phase model. We could have done everything *ab initio* in terms of the phase model but this would have made much of the presentation unnecessarily complicated at the outset.

We recall the remark from the introduction that these embeddings have three advantages: first, we get a workable syntax for handling the dependencies of operators on previous measurement outcomes, second, one can use these embeddings to transfer the measurement calculus previously developed for the one-way model to obtain a calculus for the new model including, of course, a standardization procedure that we get automatically; lastly, one can embed the patterns from the phase model into the new models and vice versa. In essence, these compositional embeddings will allow us to exhibit the phase model as being a core calculus for measurement-based computation. However different models are interesting from the point of view of implementation issues like fault-tolerance and ease of preparation of entanglement resources. Our embeddings allow one to move easily between these models and to concentrate on the one-way model for designing algorithms and proving general theorems. This section has been structured into several subsections, one for each model and its embedding.

1.7.1 Phase Model

In the one-way model the auxiliary qubits are initialized to be in the $|+\rangle$ state. We extend the one-way model to allow the auxiliary qubits to be in a more general state. We define the extended preparation command N_i^α to be the preparation of the auxiliary qubit i in the state $|+_\alpha\rangle$. We also add a new correction command Z_i^α, called a *phase correction* to guarantee that we can obtain determinate patterns. The dependent phase correction is written as $Z_i^{\alpha,s}$ with $Z_i^{\alpha,0} = I$ and $Z_i^{\alpha,1} = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\alpha} \end{pmatrix}$. Under conjugation, the phase correction, defines a new action over measurement:

$$(Z_i^{\beta,s})^\dagger M_i^\alpha Z_i^{\beta,s} = M_i^{\alpha - s\beta}$$

and since the measurement is destructive, it simplifies to $M_i^\alpha Z_i^{\beta,s} = M_i^{\alpha-s\beta}$. This action does not commute with Pauli actions and hence one cannot write a compact notation for dependent measurement, as we did before, and the computation of angle dependencies is a bit more complicated. Thereafter, a measurement preceded by a sequence of corrections on the same qubit will be called a dependent measurement. Note that, by the absorption equations, this indeed can be seen as a measurement, where the angle depends on the outcomes of some other measurements made beforehand.

To complete the extended calculus it remains to define the new rewrite rules:

$$\begin{array}{lcll} E_{ij} Z_i^{\alpha,s} & \Rightarrow & Z_i^{\alpha,s} E_{ij} & EP \\ \\ M_i^\alpha X_i^s & \Rightarrow & M_i^{(-1)^s \alpha} & MX \\ M_i^\alpha Z_i^s & \Rightarrow & M_i^{\alpha - s\pi} & MZ \\ M_i^\alpha Z_i^{\beta,s} & \Rightarrow & M_i^{\alpha - s\beta} & MP \end{array}$$

The above rules together with the rewriting rules of the one-way model described in Section 1.5, lead to a standardization procedure for the model. It is trivial that the one-way model is a fragment of this generalized model and hence universality immediately follows. It is also easy to check that the model is closed under composition and all the semantical properties of the one-way model can be extended to this general model as well.

The choice of extended preparations and its concomitant phase correction is actually quite delicate. One wishes to keep the standardizability of the calculus which constrains what can be added but one also wishes to have determinate patterns which forces us to put in appropriate corrections. The phase model is only a slight extension of the original one-way model, but it allows a discussion of the next model which is of great physical interest.

1.7.2 Pauli model

An interesting fragment of the phase model is defined by restricting the angles of measurements to $\{0, \frac{\pi}{2}, \pi, -\frac{\pi}{2}\}$ *i.e.* Pauli measurements and the angles of preparation to 0 and $\frac{\pi}{4}$. Also the correction commands are restricted to Pauli corrections X, Z and Phase correction $Z^{\frac{\pi}{8}}$. One readily sees that the

subset of angles is closed under the actions of the corrections and hence the Pauli model is closed under composition.

Proposition 13 *The Pauli model is approximately universal.*

Proof. We know that the set consisting of $J(0)$ (which is H), $J(\frac{\pi}{4})$, and $\wedge Z$ is approximately universal. Hence, to prove the approximate universality of Pauli model, it is enough to exhibit a pattern in the Pauli model for each of these three unitaries. We saw before that $J(0)$ and $\wedge Z$ are computed by the following 2-qubit patterns:

$$\begin{array}{lcl} \mathcal{J}(0) & := & X_2^{s_1} M_1^0 E_{12} \\ \wedge\mathcal{Z} & := & E_{12} \end{array}$$

where both belong also to the Pauli model. The pattern for $J_{\frac{\pi}{4}}$ in the one-way model is expressed as follows:

$$\begin{array}{lcl} \mathcal{J}(\frac{\pi}{4}) & := & X_2^{s_1} M_1^{-\frac{\pi}{4}} E_{12} \\ & = & X_2^{s_1} M_1^0 E_{12} Z_1^{\frac{\pi}{4}} \end{array}$$

The above forms do not fit in the Pauli model, since the first one uses a measurement with an angle $\frac{\pi}{4}$ and the second uses $Z^{\frac{\pi}{4}}$. However by teleporting the input qubit and then applying the $Z^{\frac{\pi}{4}}$ and finally running the standardization procedure we obtain the following pattern in the Pauli model for $J(\frac{\pi}{4})$:

$$\begin{array}{lcl} & & X_2^{s_1} M_1^0 E_{12} Z_1^{\frac{\pi}{4}} \\ = & & X_4^{s_3} M_3^0 E_{34} \boxed{Z_3^{\frac{\pi}{4}}} Z_3^{s_2} X_3^{s_1} M_2^0 M_1^0 E_{12} E_{23} \\ = & & X_4^{s_3} M_3^0 E_{34} Z_3^{s_2} Z_3^{\frac{\pi}{2}, s_2} X_3^{s_1} M_2^0 M_1^0 E_{12} E_{23} \boxed{Z_3^{\frac{\pi}{4}}} \\ = & & X_4^{s_3} M_3^0 E_{34} Z_3^{s_2} Z_3^{\frac{\pi}{2}, s_2} X_3^{s_1} M_2^0 M_1^0 E_{12} E_{23} \boxed{Z_3^{\frac{\pi}{4}}} \\ = & & X_4^{s_3+s_2} Z_4^{s_1} M_3^{-(-1)^{s_1} s_2 \frac{\pi}{2}} M_2^0 M_1^0 E_{12} E_{23} E_{34} N_3^{\frac{\pi}{4}} \end{array}$$

Approximate universality for the Pauli model is now immediate. □

Note that we cannot really expect universality (as we had for the phase model) because the angles are restricted to a discrete set. On the other hand it is precisely this restriction that makes the Pauli model interesting from the point of view of implementation. The other particular interest behind this model, apart from its simple structure, is based on the existence of a novel fault tolerant technique for computing within this framework Bravyi and Kitaev (2005); Raussendorf et al. (2004); Danos et al. (2006).

1.7.3 Teleportation

Another class of measurement-based models – older, in fact, than the one-way model – uses 2-qubit measurements. These are collectively referred to as *teleportation models* Leung (2004). Several papers that are concerned with the relation and possible unification of these models Childs et al. (2005); Aliferis and Leung (2004); Jorrand and Perdrix (2005) have already appeared. One aspect of these models that stands in the way of a complete understanding of this relation, is that, whereas in the one-way model one has a clearly identified class of measurements, there is less agreement concerning which measurements are allowed in teleportation models.

We propose here to take as our class of 2-qubit measurements a family obtained as the conjugate under the operator $\wedge Z$ of tensors of 1-qubit measurements. We show that the resulting teleportation model is universal. Moreover, almost by construction, it embeds into the one-way model, and thus exposes completely the relation between the two models.

Before embarking on the specifics of our family of 2-qubit measurements, we remark that the situation commented above is more general:

Lemma 14 *Let $\mathcal{A}$ be an orthonormal basis in $\otimes^n \mathbb{C}^2$, with associated n-qubit measurement $M^{\mathcal{A}}$, and $\mathcal{A}_i$*

with $i = 1, \ldots, n$ be orthonormal bases in $\mathbb{C}^2$, with associated 1-qubit measurements $M_i^{\mathcal{A}_i}$. Then there exists a unique (up to a permutation) n-qubit unitary operator U such that:

$$M_{1\cdots n}^{\mathcal{A}} \quad = \quad U_{1\cdots n}(\otimes_i M_i^{\mathcal{A}_i})U_{1\cdots n}^{\star}$$

Proof. Take U to map $\otimes_i \mathcal{A}_i$ to $\mathcal{A}$. □

This simple lemma says that general n-qubit measurements can always be seen as conjugated 1-qubit measurements, provided one uses the appropriate unitary to do so. As an example consider the orthogonal *graph basis* $\mathcal{G} = \wedge Z_{12}\{|\pm\rangle \otimes |\pm\rangle\}$ then the two-qubit graph basis measurements are defined as $M_{12}^{\mathcal{G}} = \wedge Z_{12}(M_1^0 \otimes M_2^0)\wedge Z_{12}$. It is now natural to extend our definition of $M_{12}^{\mathcal{G}}$ to obtain the family of 2-qubit measurements of interest:

$$M_{12}^{\alpha,\beta} \quad := \quad \wedge Z_{12}(M_1^{\alpha} \otimes M_2^{\beta})\wedge Z_{12} \tag{1.19}$$

corresponding to projections on the basis $\mathcal{G}_{\alpha,\beta} := \wedge Z_{12}(P_1(\alpha) \otimes P_2(\beta))(\{|\pm\rangle \otimes |\pm\rangle\}$. This family of two-qubit measurements together with the preparation, entanglement and corrections commands of the one-way model define the teleportation model.

Before we carry on, a clarification about our choice of measurements in the teleportation model is necessary. The usual teleportation protocol uses Bell basis measurement defined with

$$\begin{array}{lcl} \mathcal{B} & = & \wedge X_{12}\{|\pm\rangle \otimes |0/1\rangle\} \\ M_{12}^{\mathcal{B}} & = & \wedge X_{12}(M_1^z \otimes M_2^z)\wedge X_{12} \end{array}$$

where M^z is the computational-basis measurement. Note how similar these equations are to the equations defining the graph basis measurements. This is a clear indication that everything that follows can be transferred to the case where X replaces Z, and $\mathcal{B}$ replaces $\mathcal{G}$. However, since the methodology we adopt is to embed the 2-qubit measurement based model in the one-way model, and the latter is based on $\wedge Z$ and $\mathcal{G}$, we will work with the graph-basis measurements. Furthermore, since $\wedge Z$ is symmetric, whereas $\wedge X$ (*a.k.a.* as C-NOT) is not, the algebra is usually nicer to work with.

Now we prove that the family of measurements in Equation 1.19 leads to a universal model, which embeds nicely into the one-way model, but first we need to describe the important notion of dependent measurements. These will arise as a consequence of standardization; they were not considered in the existing teleportation models.

In what follows we drop the subscripts on the $\wedge Z$ unless they are really necessary. We write $(s(i), s(j)) \in \mathbb{Z}_2 \times \mathbb{Z}_2$ to represent outcome of a 2-qubit measurement, with the specific convention that $(0,0)$, $(0,1)$, $(1,0)$, and $(1,1)$, correspond respectively to the cases where the state collapses to $\wedge Z|+_\alpha\rangle|+_\alpha\rangle$, $\wedge Z|+_\alpha\rangle|-_\alpha\rangle$, $\wedge Z|-_\alpha\rangle|+_\alpha\rangle$, and $\wedge Z|-_\alpha\rangle|-_\alpha\rangle$.

We will use two types of dependencies for measurements associated with X-action and Z-action:

$$\begin{array}{lcl} [M_{ij}^{\alpha,\beta}]^{(s,t)} & = & M_{ij}^{(-1)^s\alpha,(-1)^t\beta} \\ {}^{(u,v)}[M_{ij}^{\alpha,\beta}] & = & M_{ij}^{\alpha+u\pi,\beta+v\pi} \end{array}$$

where s, t, u and v are in $\mathbb{Z}_2$. The two actions commute, so the equations above define unambiguously the full dependent measurement ${}^{(u,v)}[M_{ij}^{\alpha,\beta}]^{(s,t)}$. Here are some useful abbreviations:

$$\begin{array}{l} {}^{(0,0)}[M^{\alpha,\beta}]^{(s,t)} := [M^{\alpha,\beta}]^{(s,t)} \\ {}^{(u,v)}[M^{\alpha,\beta}]^{(0,0)} := {}^{(u,v)}[M^{\alpha,\beta}] \\ {}^{(0,0)}[M^{\alpha,\beta}]^{(0,0)} := M^{\alpha,\beta} \\ M^{\alpha,x} := M^{\alpha,0} \\ M^{\alpha,y} := M^{\alpha,\frac{\pi}{2}} \end{array}$$

As in the 1-qubit measurement case we obtain the following rewriting rules for the teleportation model:

$$\begin{array}{lcll}
E_{ij}X_i^s & \Rightarrow & X_i^s Z_j^s E_{ij} & EX \\
E_{ij}Z_i^s & \Rightarrow & Z_i^s E_{ij} & EZ \\
{}^{(u,v)}[M_{ij}^{\alpha,\beta}]^{(s,t)} X_i^r & \Rightarrow & {}^{(u,v+r)}[M_{ij}^{\alpha,\beta}]^{(s+r,t)} & MX \\
{}^{(u,v)}[M_{ij}^{\alpha,\beta}]^{(s,t)} X_j^r & \Rightarrow & {}^{(u+r,v)}[M_{ij}^{\alpha,\beta}]^{(s,t+r)} & MX \\
{}^{(u,v)}[M_{ij}^{\alpha,\beta}]^{(s,t)} Z_i^r & \Rightarrow & {}^{(u+r,v)}[M_{ij}^{\alpha,\beta}]^{(s,t)} & MZ \\
{}^{(u,v)}[M_{ij}^{\alpha,\beta}]^{(s,t)} Z_j^r & \Rightarrow & {}^{(u,v+r)}[M_{ij}^{\alpha,\beta}]^{(s,t)} & MZ
\end{array}$$

to which we add also the trivial commutation rewriting which are possible between commands that don't overlap (meaning, acting on disjoint sets of qubits).

We now describe how to translate 2-qubit EMC patterns to 1-qubit patterns and vice versa. The following equation plays the central role in the translation:

$$M_{ij}^{\alpha,\beta} = E_{ij}(M_i^\alpha \otimes M_j^\beta)E_{ij} \tag{1.20}$$

Note that this immediately gives the denotational semantics of two-qubit measurements as cptp-maps. Furthermore, all other commands in the teleportation model are the same as in the one-way model, so we have right away a denotational semantics for the entire teleportation model in terms of cptp-maps.

We write $\mathfrak{P}$ for the collection of patterns in the one-way model and $\mathfrak{T}$ for the collection of patterns in the teleportation model.

Theorem 8 *There exist functions $[\cdot]_f : \mathfrak{P} \to \mathfrak{T}$ and $[\cdot]_b : \mathcal{T} \to \mathcal{P}$ such that*

(i) $\forall \mathcal{P} \in \mathfrak{P} : [\![\mathcal{P}]\!] = [\![[\mathcal{P}]_f]\!]$;
(ii) $\forall \mathcal{T} \in \mathfrak{T} : [\![\mathcal{T}]\!] = [\![[\mathcal{T}]_b]\!]$;
(iii) $[\cdot]_f \circ [\cdot]_b$ *and* $[\cdot]_b \circ [\cdot]_f$ *are both identity maps.*

Proof. We first define the forward map $[\cdot]_f$ in stages as follows for any patterns $\mathcal{P} = (V, I, O, A_n \dots A_1)$:

(i) For any $i \in V \setminus O$ (i.e. measured qubits) we add an auxiliary qubit i_d called a *dummy* qubit to the space V.
(ii) For any $i \in V \setminus O$ we replace any occurrence of M_i^α with $M_i^\alpha M_{i_d}^x$.
(iii) We then replace each of the newly created occurrences of $M_i^\alpha M_{i_d}^x$ by $M_{ii_d}^{\alpha,x} E_{ii_d}$.

Now we show that the first condition stated in the theorem holds; we do this stage wise. The first two stages are clear because we are just adding qubits that have no effect on the pattern because they are not entangled with any pre-existing qubit, and no other command depends on a measurement applied to one of the dummy qubits. Furthermore, we add qubits in the state $|+\rangle$ and measure them in the $|\pm\rangle$ basis. The invariance of the semantics under stage 3 is an immediate consequence of Equation 1.20 and the fact that all the measurements are destructive, and hence an entanglement command on qubits appearing after a measurement of any of those qubits can just be removed.

The map $[\cdot]_b$ is defined similarly except that there is no need to add dummy qubits. One only needs to replace any two-qubit measurement $M_{ij}^{\alpha,\beta}$ with $M_i^\alpha M_j^\beta E_{ij}$. Again, this clearly preserves the semantics of patterns because of Equation 1.20 and the above remark about destructive measurements. Thus condition 2 of the theorem holds.

The fact that the two maps are mutual inverses follows easily. As all the steps in the translations are local we can reason locally. Looking at the forward mapping followed by the backward mapping we get the following sequence of transformations

$$\begin{array}{lll}
M_i^\alpha & \Rightarrow_{\text{stage 1,2}} & M_i^\alpha M_{i_d}^x \\
& \Rightarrow_{\text{Equation 1.20}} & M_{ii_d}^{\alpha,x} E_{ii_d} \\
& \Rightarrow_{\text{Equation 1.20}} & M_i^\alpha M_{i_d}^x E_{ii_d} E_{ii_d} \\
& \Rightarrow & M_i^\alpha M_{i_d}^x \\
& \Rightarrow & M_i^\alpha
\end{array}$$

This shows that we have the third condition of the theorem. □

Note that the translations are compositional since the denotational semantics is and also it follows immediately that the teleportation model is universal and admits a standardization procedure.

Example. Consider the teleportation pattern in the teleportation model given by the command sequence: $X_3^{s_1} Z_3^{s_2} M_{12}^{x,x} E_{23}$, we perform the above steps:

$$\begin{array}{ll} X_3^{s_1} Z_3^{s_2} \boxed{M_{12}^{x,x}} E_{23} & \Rightarrow_{\text{Equation 1.20}} \\ X_3^{s_1} Z_3^{s_2} M_1^x M_2^x E_{12} E_{23} & \end{array}$$

and hence obtain the teleportation pattern with 1-qubit measurements.

Example. We saw before, the following EMC 1-qubit pattern for $R_z(\alpha)$ which can be embedded to an EMC 2-qubit pattern using the above steps:

$$\begin{array}{ll} X_3^{s_2} Z_3^{s_1} \boxed{[M_2^x]^{s_1} M_1^{-\alpha}} E_{12} E_{23} & \Rightarrow_{\text{stage 1,2}} \\ X_3^{s_2} Z_3^{s_1} \boxed{[M_2^x]^{s_1} M_{2_d}^x M_1^{-\alpha} M_{1_d}^x} E_{12} E_{23} & \Rightarrow_{\text{Equation 1.20 and standardization}} \\ X_3^{s_2} Z_3^{s_1} [M_{22_d}^{x,x}]^{(s_1,0)} M_{11_d}^{-\alpha,x} E_{11_d} E_{22_d} E_{12} E_{23} & \end{array}$$

Note that we have explicit algorithmic translations between the models and not just illustrative examples. This is the main advantage of our approach in unifying these two models compared to the extant work Childs et al. (2005); Aliferis and Leung (2004); Jorrand and Perdrix (2005).

1.7.4 State Transfer

In this section we consider the state transfer model Perdrix (2003, 2007), a measurement-based model of quantum computation where the initial entanglement is created by means of measurements instead of the 2-qubit unitary transformation ΛZ. This model has been originally introduced for reducing the resources of the teleportation model. The creation of entanglement is done by means of 2-qubit measurements which are supposed to be non destructive i.e., qubits can be re-used after their measurement. Moreover, contrary to the teleportation model, the 2-qubit measurements are partial, meaning that they are not projecting the state of the measured qubits on a vector, but on a plane. In this section, we consider partial measurement of the following form:

$$M_{12}^{\alpha} := \Lambda Z_{12} M_1^{\alpha} \Lambda Z_{12}$$

M_{12}^{α} is a measurement on two qubits with only two possible classical outcomes 0 or 1. More generally, a partial measurement is defined by a collection of projectors with eigenspaces of dimension possibly larger than one. An example of such a measurement is the 2-qubit parity measurement. The parity measurement consists of two projectors: one on the even subspace $span(|00\rangle, |11\rangle)$ and another on the odd subspace $span(|01\rangle + |10\rangle)$.

A state transfer pattern consists of the 1-qubit measurement and the correction commands of the measurement calculus, together with the non destructive measurement M_{12}^0 and the $|0\rangle$-state preparation. We show that the resulting model is universal, and we present how this model embeds in the one-way model and vice versa.

We use the command M_{ij}^{α} and N_i^Z for representing respectively the partial two qubit measurement M_{12}^{α} and the $|0\rangle$-initialisation. We write $s_{ij} \in \mathbb{Z}_2$ to represent the outcome of $M_{i,j}^{\alpha}$. $M_{i,j}^{\alpha}$ is supposed to be non destructive, some commands may act on i or j after the application of $M_{i,j}^{\alpha}$. However, we assume that each pair of qubits is measured at most once in order to ensure that the use of the signal s_{ij} is not ambiguous. Notice that the one-qubit measurements M_i^{α} are still supposed to be destructive.

The operational semantics of the commands M_{ij}^{α} and N_i^Z is:

$$\begin{array}{lll} V, W, q, \Gamma & \xrightarrow{N_i^Z} & V \cup \{i\}, W, |0\rangle_i \otimes q, \Gamma \\ V, W, q, \Gamma & \xrightarrow{{}^t[M_{ij}^{\alpha}]^s} & V, W \cup \{s_{ij}\}, \Lambda Z_{ij} |+_{\alpha_\Gamma}\rangle\langle +_{\alpha_\Gamma}|_i \Lambda Z_{ij} q, \Gamma[0/s_{ij}] \\ V, W, q, \Gamma & \xrightarrow{{}^t[M_{ij}^{\alpha}]^s} & V, W \cup \{s_{ij}\}, \Lambda Z_{ij} |-_{\alpha_\Gamma}\rangle\langle -_{\alpha_\Gamma}|_i \Lambda Z_{ij} q, \Gamma[1/s_{ij}] \end{array}$$

We are now defining rewriting rules for the state transfer model. First we notice that $M_{12}^{\alpha} X_1 = X_1 M_{12}^{-\alpha}$:

$$\begin{aligned} M_{12}^{\alpha} X_1 &= \Lambda Z_{12} M_1^{\alpha} \Lambda Z_{12} X_1 \\ &= \Lambda Z_{12} M_1^{\alpha} X_1 Z_2 \Lambda Z_{12} \\ &= \Lambda Z_{12} X_1 Z_2 M_1^{-\alpha} \Lambda Z_{12} \\ &= X_1 \Lambda Z_{12} M_1^{-\alpha} \Lambda Z_{12} \\ &= X_1 M_{12}^{-\alpha} \end{aligned}$$

Moreover $M_{12}^{\alpha} Z_1 = Z_1 M_{12}^{\alpha+\pi}$, $M_{12}^{\alpha} X_2 = X_2 M_{12}^{\alpha+\pi}$ and $M_{12}^{\alpha} Z_2 = Z_2 M_{12}^{\alpha+\pi}$. As a consequence, we obtain the following rewriting rules for the state transfer model, where ${}^t[M_{ij}^{\alpha}]^s := M_{ij}^{(-1)^s \alpha + t\pi}$:

$$\begin{array}{lcll} {}^t[M_i^{\alpha}]^s X_i^r & \Rightarrow & {}^t[M_i^{\alpha}]^{s+r} & MX \\ {}^t[M_i^{\alpha}]^s Z_i^r & \Rightarrow & {}^{r+t}[M_i^{\alpha}]^s & MZ \\ {}^t[M_{ij}^{\alpha}]^s X_i^r & \Rightarrow & X_i^r \ {}^t[M_{ij}^{\alpha}]^{s+r} & MX1 \\ {}^t[M_{ij}^{\alpha}]^s X_j^r & \Rightarrow & X_j^r \ {}^{t+r}[M_{ij}^{\alpha}]^s & MX2 \\ {}^t[M_{ij}^{\alpha}]^s Z_i^r & \Rightarrow & Z_i^r \ {}^{t+r}[M_{ij}^{\alpha}]^s & MZ1 \\ {}^t[M_{ij}^{\alpha}]^s Z_j^r & \Rightarrow & Z_j^r \ {}^t[M_{ij}^{\alpha}]^s & MZ2 \end{array}$$

Lemma 15 *The rewriting system $\Rightarrow$ is terminating.*

Proof Notice that the size of the pattern is not increasing when the rewriting rules are applied. Moreover, for every rewriting rule, the total distance of the measurements from the right hand side of the term is strictly decreasing. It guaranties that the rewriting system is terminating. □

Now we are considering examples of state transfer patterns. Here are the implementations in this model of a universal family of unitary transormations:

- The following pattern is implementing the unitary transformation $J(\alpha)$:

$$\mathcal{J}(\alpha) = (\{i,j\}, \{i\}, \{j\}, X_j^{s_i} Z_j^{s_{ji}} M_i^{-\alpha} M_{ji}^0 N_j^Z)$$

- The following pattern is implementing ΛX:

$$\Lambda\mathcal{X} = (\{i,j,k\}, \{i,j\}, \{i,k\}, X_k^{s_j} Z_k^{s_{kj}\, s_{ji}} [M_j^0] M_{kj}^0 M_{ji}^0 N_j^Z)$$

We write $\mathfrak{S}$ the collection of patterns of this state transfer model.

Lemma 16 *There exist functions $[.]_f : \mathfrak{P} \to \mathfrak{S}$ and $[.]_b : \mathfrak{S} \to \mathfrak{P}$ such that*

(i) $\forall \mathcal{P} \in \mathfrak{P} : [\![\mathcal{P}]\!] = [\![[\mathcal{P}]_f]\!]$;
(ii) $\forall \mathcal{S} \in \mathfrak{S} : [\![\mathcal{S}]\!] = [\![[\mathcal{S}]_b]\!]$.

Proof The backward map $[.]_b$ consists in replacing each command N_i^Z by $Z_i^{s_j} M_j^0 E_{ij} N_i N_j = \mathcal{J}(0)|+\rangle$ and M_{ij}^{α} by $E_{jk} Z_k^{s_i} N_k M_i^0 E_{ij}$. This last term comes from the decomposition of the partial measurement $M_{12}^{\alpha} = \Lambda Z_{12} M_1^{\alpha} \Lambda Z_{12}$. Since the one-qubit measurements are destructive, an ancillary qubit k is added to replace the qubit i after the measurement, leading to the sequence of commands $E_{jk} Z_k^{s_i} N_k M_i^0 E_{ij}$. The forward map $[.]_f$ consists in replacing each command N_i by $Z_j^{s_{ji}} M_i^0 M_{ji}^0 N_j^Z N_i^Z = \mathcal{J}(0)|0\rangle$, and E_{ij} by $\mathcal{J}(0)_j \circ \Lambda\mathcal{X}_{ij} \circ \mathcal{J}(0)_j$, where the implementation of $\mathcal{J}(0)$ and $\Lambda\mathcal{X}$ in the state transfer model is given above. □

Notice that neither $[\cdot]_f \circ [\cdot]_b$ nor $[\cdot]_b \circ [\cdot]_f$ is the identity. We can observe that contrary to Theorem 8 relating the teleportation and one-way model, the above backward translation $[.]_b$ requires more ingenuity, since one has to "unshare" the qbits of the state-transfer pattern. Regarding the forward translation $[.]_f$ we can observe further that the translation of E_{ij} is purely semantic: we know how to implement $H = \mathcal{J}(0)$ and ΛX so we know how to implement ΛZ. This leads to a 5-qubit pattern composed of 16 commands which works but which seems not particularly meaningful.

1.8 Projection-based Quantum Computing

Based on measurements, a pattern has a probabilistic evolution, as it is illustrated by its semantics (see section 1.3). However the correction mechanism based on Pauli corrections enables globally deterministic behaviour. In section 1.3, definitions of deterministic and strongly deterministic patterns are given: a deterministic pattern has an evolution which maps pure states with pure states, and a strong deterministic pattern realizes a unitary embedding (or isometry.)

The semantics of a strongly deterministic pattern is entirely characterised by its 'zero' branch, i.e. the branch where all the classical outcomes, or signals, are 0. According to the definition of the measurement calculus, a 'zero' branch is Pauli correction free. Thus, the corrections of a strongly deterministic can be abstracted away, leading to a projective quantum computation where every measurement is replaced by its projector associated with the classical outcome 0. An arbitrary projective quantum computation based on unitary transformations and projections is not *a priori* a valid quantum evolution, however some of them are specifications of strongly deterministic pattern of the measurement calculus.

In this section, a formalism for representing projection-based quantum computations is introduced. The syntax and the semantics of the language are presented as well as the composition of projective terms. Moreover a general technique based on phase map decompositions is presented for translating unitary transformation into a projective quantum computation.

A projective quantum computation consists in:

- Initialising some ancillary qubits in a $|+\rangle$ state;
- Creating entanglement by means of entanglement operators;
- Applying projections $\langle +_\alpha| = \frac{1}{\sqrt{2}}(\langle 0| + e^{-i\alpha}\langle 1|)$ on non output qubits.

The creation of entanglement, i.e. the first two steps, can be represented by an open graph (G, I, O), where I is the set of input qubits, O the set of output qubits, and $G = (V, E)$, such that $I, O \subseteq V$, is an undirected simple graph. All the qubits in $V \setminus I$ are prepared in the $|+\rangle$ state, then for any edge $(u, v) \in E$, the entanglement operator $E_{u,v}$ is applied on the corresponding qubits. Notice that since for any $u, u', v, v' \in V$, $E_{u,v}$ commutes with $E_{u',v'}$, the resulting state does not depend on the order chosen for enumerating the edges of G.

The third step which consists in applying on every non output qubit $u \in V \setminus O$ a projector $\langle +_{\alpha_u}|$ can be represented as a labelling $\alpha : V \setminus O \to [0, 2\pi)$. Notice that, contrary to the measurement calculus, there is no classical signal and thus no dependancy between the projections, and moreover the projectors act on individual qubits. As a consequence, for any $u \neq v \in V \setminus O$, $\langle +_{\alpha_u}|$ and $\langle +_{\alpha_v}|$ commute, thus the resulting state does not depend on the order chosen for enumerating the labels of the graph.

Definition 17 *A projective term is a labelled open graph (G, I, O, α) where $G = (V, E)$ is a simple undirected graph, $I, O \subseteq V$ are sets of input and output vertices, and $\alpha : V \setminus O \to [0, 2\pi)$ associates with any non output vertex u an angle $\alpha(u)$.*

The sequential composition of two projective terms is defined as follows:

Definition 18 *Let $(G_1, I_1, O_1, \alpha_1)$ and $(G_2, I_2, O_2, \alpha_2)$ be two projective terms such that $V_1 \cap V_2 = O_1 = I_2$,*

$$(G_2, I_2, O_2, \alpha_2) \circ (G_1, I_1, O_1, \alpha_1) = (G_2 \Delta G_1, I_1, O_2, \alpha)$$

where $G_1 \Delta G_2 := (G_1 \cup G_2) \setminus (G_1 \cap G_2)$ is the symmetric difference of G_1 and G_2, and

$$\alpha : (V_1 \cup V_2) \setminus O_2 \to [0, 2\pi) = u \mapsto \begin{cases} \alpha_2(u) & \text{if } u \in V_2 \\ \alpha_1(u) & \text{otherwise} \end{cases}$$

The tensor of two projective terms is defined as follows:

Definition 19 *Let $(G_1, I_1, O_1, \alpha_1)$ and $(G_2, I_2, O_2, \alpha_2)$ be two projective terms*

$$(G_1, I_1, O_1, \alpha_1) \otimes (G_2, I_2, O_2, \alpha_2) = (G_1 \cup G_2, I_1 \cup I_2, O_1 \cup O_2, \alpha)$$

where

$$\alpha : (V_1 \cup V_2) \setminus (O_1 \cup O_2) \to [0, 2\pi) = u \mapsto \begin{cases} \alpha_1(u) & \text{if } u \in V_1 \\ \alpha_2(u) & \text{otherwise} \end{cases}$$

The denotational semantics of a projective term is a linear map which consists in creating ancillary qubits in the $|+\rangle$ state, then applying entanglement operators, and finally applying projectors on the non output qubits:

Definition 20 *Let (G, I, O, α) be a projective term, its denotational semantics $[\![(G, I, O, \alpha)]\!] : \mathfrak{H}_I \to \mathfrak{H}_O$ is:*

$$[\![(G, I, O, \alpha)]\!] : \mathfrak{H}_I \to \mathfrak{H}_O = \left(\prod_{u \in O^c} \langle \alpha(u)|_u \right) \Lambda Z_G |+\rangle_{I^c}$$

where $\Lambda Z_G = \prod_{(u,v)\in G} \Lambda Z_{u,v}$. Note that ΛZ_G is well-defined since for any u, u', v, v', $\Lambda Z_{u,v} = \Lambda Z_{v,u}$ and $\Lambda Z_{v,u}$ commutes with $\Lambda Z_{u',v'}$.

It is easy to verify that the denotational semantics preserves the sequential and tensorial composition of projective terms:

Lemma 21

$$[\![(G_2, I_2, O_2, \alpha_2) \circ (G_1, I_1, O_1, \alpha_1)]\!] = [\![(G_2, I_2, O_2, \alpha_2)]\!] \circ [\![(G_1, I_1, O_1, \alpha_1)]\!]$$

$$[\![(G_1, I_1, O_1, \alpha_1) \otimes (G_2, I_2, O_2, \alpha_2)]\!] = [\![(G_1, I_1, O_1, \alpha_1)]\!] \otimes [\![(G_2, I_2, O_2, \alpha_2)]\!]$$

1.8.1 From Unitary to PBQC

We wish to explore whether measurement-based quantum computing suggests new techniques for designing quantum algorithms. Previously, one would typically start with an algorithm already implemented in the circuit model and replace each gate by a corresponding pattern. To transform the pattern to a standard form where all entangling operations are performed first, one could then use the by-product method Raussendorf et al. (2003) or the more general standardization algorithm presented in Section 1.5. Here, we propose a direct method that is free from any reference to the circuit model and leads into a direct decomposition of a given unitary map into a projection-based pattern de Beaudrap et al. (2006, 2008). Then in the next section we address the question of how to transform a projection-based pattern to a one-way pattern. We start with the observation that the positive branch of a one-way pattern implicitly defines a particular decomposition of the corresponding unitary map into a preparation map enlarging the input space, a diagonal map with unit coefficients, and a restriction map contracting back the space to the output space, which we call a *phase map decomposition*. Note that this decomposition does not directly correspond to any physical procedure as it defines a projection-based pattern

However since the one-way model is universal, this alternative decomposition is also universal; and there is a straightforward procedure which allows us to determine a phase map decomposition for a unitary from a pattern implementing the same unitary. Remarkably, one can define a reverse procedure as well, which breaks into two steps. First, given a unitary map, one enumerates such phase map decompositions by constructing the right set of coefficients in the middle diagonal map (Lemma 22 and Algorithm 2). Then for each such decomposition, one verifies whether there exists a matching projection-based pattern. This reduces to finding for any phase map decomposition a matching entangled graph state (with inputs) and choice of measurements angles (Algorithm 3).

We now turn to the formulation of our decomposition. Various operators over $\mathcal{H}$ preserve the computational basis, up to phase: for example, Pauli maps, Z^α (defined as $Z^\alpha|0\rangle = |0\rangle$, and $Z^\alpha|1\rangle = e^{i\alpha}|1\rangle$), and controlled-Paulis. One-qubit measurements also map the standard basis of one space to those of

another, up to a scalar factor. In particular, one has the following simple equations where $|x\rangle$ is an n-qubit computational basis state and j is an index for a qubit in $|x\rangle$:

$$\begin{array}{lcl} \langle \pm_\alpha| \otimes I^{\otimes n}(|0\rangle|x\rangle) & = & 2^{-1/2}|x\rangle \\ \langle \pm_\alpha| \otimes I^{\otimes n}(|1\rangle|x\rangle) & = & \pm 2^{-1/2} e^{-i\alpha}|x\rangle \\ \wedge Z_{1j}|0\rangle|x\rangle & = & |0\rangle|x\rangle \\ \wedge Z_{1j}|1\rangle|x\rangle & = & Z_j|1\rangle|x\rangle \end{array}$$

We will call a map $\Phi : \mathcal{H}_V \to \mathcal{H}_V$ a *phase map* if it is diagonal in the computational basis and has only unit coefficients. The typical example of such a map is $\wedge Z$. It is important to note that the above definition depends on the choice of a basis.

Following one-way model terminology we also define a *preparation map* $P_{I\to V} : \mathcal{H}_I \to \mathcal{H}_V$ that expands the input space by tensoring auxiliary qubits:

$$|x\rangle \mapsto |x\rangle \otimes |+\cdots+\rangle_{I^c}$$

And a *restriction map* $R_{V\to O} : \mathcal{H}_V \to \mathcal{H}_O$ that projects the space to the output space:

$$|x\rangle \mapsto \langle +\cdots+|_{O^c}|x\rangle$$

It is easy to see that the restriction map is the adjoin of the preparation map.

As we have seen above, measurement and entangling commands in the one-way model define phase maps, and hence from the universality of the model we obtain the following decomposition:

Theorem 9 *For all unitary $U : \mathcal{H}_I \to \mathcal{H}_O$, there exists a phase map $\Phi : \mathcal{H}_V \to \mathcal{H}_V$ such that:*

$$U = R_{V\to O} \circ \Phi \circ P_{I\to V}$$

Proof. There exist a deterministic one-way pattern $\mathcal{P}$ implementing U where its positive branch can be written as:

$$\begin{array}{rcl} U & = & 2^{|O^c|/2} \prod_{i\notin O} \langle +_{\alpha_i}|_i \; E_G \; P_{I\to V} \\ & = & 2^{|O^c|/2} \prod_{i\notin O} \langle +|_i Z_i^{-\alpha_i} \; E_G \; P_{I\to V} \\ & = & R_{V\to O} \prod_{i\notin O} Z_i^{-\alpha_i} E_G \; P_{I\to V} \\ & = & R_{V\to O} \; \Phi_{V\to V} \; P_{I\to V} \end{array}$$

where $\Phi = \prod_{i\notin O} Z_i^{-\alpha_i} \prod_{ij\in E} \wedge Zij$ is the phase map corresponding to the entanglement operations and measurement angles. □

One can think of the above theorem as a special kind of diagonalization for unitaries where one is allowed to inflate the dimension of the underlying space. This will prove to be useful for direct programming in the one-way model. We present first a couple of examples, clarifying our first set of definitions and preparing the ground for a direct proof of the above theorem which does not invoke universality of the one-way model. The examples already hint at the construction behind the direct phase map decomposition algorithm.

Example 1

Consider the unitary map $J_\alpha : \mathcal{H}_{\{1\}} \to \mathcal{H}_{\{1\}}$ which decomposes in the computational basis as:

$$J_\alpha = 2^{-1/2} \begin{pmatrix} 1 & e^{-i\alpha} \\ 1 & -e^{-i\alpha} \end{pmatrix} = \begin{pmatrix} 1 & 0 & e^{-i\alpha} & 0 \\ 0 & 1 & 0 & -e^{-i\alpha} \end{pmatrix} \cdot 2^{-1/2} \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{pmatrix}$$

This decomposition is obtained from the one-way pattern $X_2^{s_1} M_1^\alpha E_{12}$ which implements J_α, and has as positive branch the phase map:

$$2^{1/2}\langle +_\alpha|_1 \wedge Z_{12} : \mathcal{H}_{\{1,2\}} \to \mathcal{H}_{\{2\}}$$

Factoring out the restriction operator gives the decomposition:

$$J_\alpha \quad = \quad \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & e^{-i\alpha} & 0 \\ 0 & 0 & 0 & -e^{-i\alpha} \end{pmatrix} \cdot 2^{-1/2} \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{pmatrix}$$

where the left matrix is the restriction $R_1 : \mathcal{H}_{\{1,2\}} \to \mathcal{H}_{\{2\}}$. The phase map here is $Z_1^{-\alpha} \wedge Z_{12}$, and the decomposition above can be rewritten:

$$J_\alpha \quad = \quad R_1(Z_1^{-\alpha} \wedge Z_{12})P_2 \,.$$

Example 2

Example 1 uses only one auxiliary qubit, and as such is a special case where the required number of auxiliary qubits is equal to the number of inputs. This is of course not always the case and the general algorithm for phase map decomposition will take care of this. We will present exact bounds on how much one needs to expand the computational space to be able to obtain the decomposition; however to realize a decomposition as a pattern we will need further restrictions. The following example demonstrate this case. The shortest known pattern for the Z^α-rotation is $X_3^{s_2} Z_3^{s_1} M_2^0 M_1^{-\alpha} E_{12} E_{23}$ with positive branch:

$$\langle +|_2 \langle +_{-\alpha}|_1 E_{12} E_{23}$$

which induces the 3-qubit phase map $\Phi|xyz\rangle = (-1)^{xy+yz} e^{i\alpha y}|xyz\rangle$ (indeed a diagonal of units) and corresponds to the following decomposition of Z^α:

$$R_{12}\, D(1, 1, 1, -1, e^{i\alpha}, e^{i\alpha}, -e^{i\alpha}, e^{i\alpha})\, P_{23}$$

where D is a diagonal matrix. Note that some permutations of the diagonal lead to other solutions, and most decompositions won't correspond to a pattern. Also if one uses only 1 additional qubit one obtains another decomposition with middle map:

$$D(\sqrt{2}, 0, 0, \sqrt{2}e^{i\alpha})$$

which is not a phase map since coefficients are not units. The natural question is whether it is possible to generate all such decompositions without using a pattern, and next find a pattern matching one of these decompositions.

Direct Decomposition

Supposing one adds n auxiliary qubits to the input space I, a simple calculation shows that for each coefficient u in the the computational basis matrix representation of U, there will be $2^{|V|-|O|}/2^{|I|} = 2^{n-|I|}$ 'slots' to spread over the diagonal of the phase map Φ (since U is a unitary $|I| = |O|$). Thus, finding a decomposition amounts in this case to finding complex numbers $x^{(i)}$ such that the following two conditions hold:

$$u = \sum_{i \leq 2^{n-|I|}} x^{(i)} \tag{1.21}$$

$$2^{n/2}|x^{(i)}| = 1 \tag{1.22}$$

The first equation says that the restriction map R will sum up all the $x^{(i)}$s to give u, while the second one asks for unit diagonal elements (note that the preparation map of n auxiliary qubits introduces an overall factor of $2^{-n/2}$).

Lemma 22 *If $n > |I|$, Equations 1.21 and 1.22 have joint solutions iff $|u| \leq 2^{n/2-|I|}$.*

Proof. Each complex $x^{(i)}$ can be seen as a real plane vector of constant length $2^{-n/2}$, and all one has to do is to choose their angles such that they will globally add up to u. If one aligns all $x^{(i)}$s with u, the

resulting sum is at least as long as u iff $|u| \leq 2^{n/2-|I|}$; thus this inequality is necessary for Equations 1.21 and 1.22 to have joint solutions.

If $n > |I|$, then $2^{n-|I|} \geq 2$, so there are at least two terms $x^{(i)}$. We may pick any two of them and rotate them at opposite angles $\pm\theta$. If θ reaches $\frac{\pi}{2}$ before the global sum matches u, then the corresponding two $x^{(i)}$s contribute nothing, and we pick two additional terms to rotate. Clearly, at some stage, for some value of θ the sum will coincide with that of u. □

Due to unitarity of U, $|u| \leq 1$: so a safe choice of n is one such that $2^{n/2-|I|} \geq 1$. Thus, $n \geq 2|I|$ is always sufficient for a phase map to exist. Another consequence of the above lemma is that for any given unitary map U on $|I|$ qubits, unless U is itself a phase map and also requires no auxiliary qubits, we have $n > |I|$: then a lower bound on the number of required qubits to implement it as a one-way pattern is $2|I|$, if at least one coefficient of U is larger than $\frac{1}{2}$.

For a unitary U , once we have fixed n, an output space O, and a choice of $x_{pq}^{(i)}$ satisfying Equations 1.21 and 1.22 for the coefficients $u_{pq} = \langle p|U|q\rangle$, the following algorithm will enumerate all possible decompositions:

Algorithm 2 Input*: for sets V, I, and O and $n = |I^c|$:*
— *a unitary U on $\mathcal{H}_I$;*
— *complex numbers $\{ x_{pq}^{(i)} \}_{i=1}^{2^{n-|I|}}$ satisfying Equations 1.21 and 1.22 for each u_{pq} ;*
— *a permutation σ over $\{1, \cdots, 2^{n-|I|}\}$.*

Output*: diagonal elements $\{d_{kk}\}_{k=1}^{2^{|V|}}$, such that $d_{kk} = \sqrt{2^n}\, x_{pq}^{(i)}$, where:*
— *the binary representation of p agrees with that of k after restriction to O;*
— *$q \equiv k \bmod 2^{|I|}$;*
— *$i = \sigma(\lfloor k/2^{|I|} \rfloor)$.*

The elements $\{d_{kk}\}_{k=1}^{2^n}$ are the solutions of the linear equations $RDP = U$ and due to the simple structure of matrices R and P we derive the above algorithm.

Pattern Synthesis

Note that obtaining a decomposition is not sufficient for the existence of a projection-based pattern. To determine whether a phase map decomposition $R\,\Phi\,P$ of a unitary U has a corresponding projection-based pattern, one wants a graph G_E over V, and angles α_j for $j \in O^c$ such that

$$\Phi = \prod_{j \in O^c} Z_j^{-\alpha_j} \prod_{jk \in E} \wedge Z_{jk}$$

That means for all x in the V-computational basis:

$$d_{xx} = e^{-i \sum_{O^c} \alpha_j x_j} \, (-1)^{\sum_{jk \in E} x_j x_k} \tag{1.23}$$

where d_{xx} is the diagonal coefficient of the phase map corresponding to x basis. Based on this observation we propose the following algorithm for the above graph matching problem.

Algorithm 3 Input*: A phase map decomposition for U — i.e. the diagonal elements $\{d_{xx}\}_{x=1}^{2^{|V|}}$ from Algorithm 2.*
Output*: either (i) A labelled open graph (G, I, O, α) defining a projection-based pattern or (ii) no matching graph exists.*

1. *For $j \in \{1, \cdots, |O^c|\}$, consider the $|V|$-bit string $\mathbf{z}_j$ that only has a 1 at position j, and set α such that $e^{-i\alpha_j} = d_{\mathbf{z}_j \mathbf{z}_j}$.*
2. *For all j, k, consider the $|V|$-bit string $\mathbf{z}_{jk}$ having a 1 only at positions j and k. Check whether $d_{\mathbf{z}_{jk}\mathbf{z}_{jk}} = \pm e^{-i(\alpha_j + \alpha_k)}$ (the angles for the corresponding qubit in O is taken to be 0).*
   — *(i) if YES and the sign is -1, return E_{jk} as an edge in G.*
   — *(ii) if NO, no matching graph exists.*

Although there are exponentially many elements on the diagonal of the phase map, testing for the existence of G will query the middle diagonal map only quadratically in $|I| + n$, to read off the measurement angles and the entanglement graph. This in practice could accelerate the detection of bad decompositions before obtaining all diagonal elements in the phase map.

When the procedure fails, one backtracks by: **(1)** trying a different decomposition given by Algorithm 2, **(2)** trying another solution from Lemma 22, **(3)** revising the choice of outputs, and ultimately **(4)** expanding further the computational space. Without any additional constraints, it seems that there are many solutions to be checked. One might be able to infer additional constraints to Equations 1.21 and 1.22 from the requirement that there be a corresponding entanglement graph, reducing the set of phase maps which we consider. How this may be done is, however, an open question.

1.9 From Projection Specification to Measurement Implementation

A projective term t has no physical meaning, however it can be seen as a specification of a strongly deterministic measurement pattern $\mathcal{P}$ if $[\![t]\!] = [\![\mathcal{P}]\!]$. Indeed, for a given strongly deterministic pattern $\mathcal{P}$, the Pauli corrections can be abstracted away by considering only the projections of the 'zero' branch of the computation, where no Pauli operator are applied. $\mathcal{P}$ is abstracted in $t_{\mathcal{P}} = (G, I, O, \alpha)$ such that $[\![\mathcal{P}]\!] = [\![t_{\mathcal{P}}]\!]$, if $\mathcal{P}$ can be rewritten in the following NEMC form:

$$\mathcal{P} = C^s \left(\prod_{i \in V \setminus O}^{\prec} M_i^{\alpha_i} \right) \left(\prod_{i,j \in G} E_{i,j} \right) \left(\prod_{i \in V \setminus I} N_i \right)$$

In this section, we consider the problem of producing a strongly deterministic pattern from its specification given in terms of projective term. It turns out that there exist projective terms which have no physical implementation by means of measurement patterns. Thus, sufficient conditions for the existence of such implementations are presented in this section. They are based on the existence of a flow in the underlying open graph of the projective term. The implementation of projective terms plays a crucial role in the phase map decomposition. Indeed, a phase map decomposition produces a projective term from a unitary transformation (see subsection 1.8.1), which can be implemented with a strongly deterministic pattern if the flow condition is satisfied.

Moreover, the use of projective terms, permits to address the key issue of the depth minimisation of strongly deterministic measurement patterns. The depth of a measurement pattern depends on the corrections and the dependency between the measurements. Thus, finding an implementation of a projective term (in which all the projectors can be applied on parallel since they act on distinct qubits) in a deterministic measurement pattern of minimal depth is a crucial issue.

Causal Flow

A variety of methods for constructing measurement patterns have been already proposed Raussendorf et al. (2003); Hein et al. (2004); Childs et al. (2005) that guarantee determinism by construction. We introduce a direct condition on open graph states which guarantees a strong form of deterministic behavior for a class of one-way measurement patterns defined over them Danos and Kashefi (2006). Remarkably, our condition bears only on the geometric structure of the entangled graph states. This condition singles out a class of patterns with flow, which is stable under sequential and parallel compositions and is large enough to realize all unitary and unitary embedding maps.

Patterns with flow have interesting additional properties Danos and Kashefi (2006). First, they are uniformly deterministic, in the sense that no matter what the measurement angles are, the obtained set of corrections, which depends only on the underlying geometry, will make the global behaviour deterministic. Second, all computation branches have equal probabilities, which means in particular, that these probabilities are independent of the inputs, and as a consequence, one can show that all such patterns implement unitary embeddings. Third, a more restricted class of patterns having both flow and

reverse flow supports an operation of adjunction, corresponding to time-reversal of unitary operations. This smaller class implements all and only unitary transformations.

Definition 23 *$(f, \prec)$ is a causal flow of (G, I, O), where $f : O^c \to I^c$ and $\prec$ is a strict partial order over V, if and only if*

1. *$i \prec f(i)$*
2. *if $j \in N_G(f(i))$ then $j = i$ or $i \prec j$*
3. *$i \in N_G(f(i))$.*

As one can see, a flow consists of two structures: a function f over vertices and a matching partial order over vertices. In order to obtain a deterministic pattern for an open graph state with flow, dependent corrections will be defined based on function f. The order of the execution of the commands is given by the partial order induced by the flow. The matching properties between the function f and the partial order $\prec$ will make the obtained pattern runnable. An example of causal flow is given in Figure 1.4.

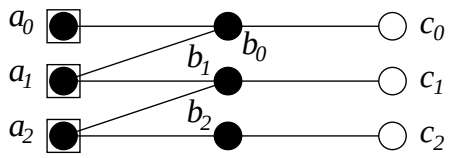


Fig. 1.4. Example of open graph – squared vertices represent inputs, white vertices represent outputs – which has a causal flow $(g, \prec)$, where $g(a_i) = b_i$, $g(b_i) = c_i$ and $a_0 \prec a_1 \prec a_2 \prec \{b_0, b_1, b_2\} \prec \{c_0, c_1, c_2\}$.

The existence of a causal flow is a sufficient condition for determinism. First we need the following simple lemma which describes an essential property of graph state.

Lemma 24 *For any open graph (G, I, O) and any $i \in I^c$,*

$$E_G N_{I^c} = X_i Z_{N_G(i)} E_G N_{I^c}$$

Proof. The proof is based on equations 1.11, 1.13 of the Measurement Calculus, and the additional equation $X_i N_i = N_i$, which follows from the fact that N_i produces a qubit in the $|+\rangle$ state which is a fix point of X.

$$\begin{aligned} E_G N_{I^c} &= E_G X_i N_{I^c} \\ &= \left(\prod_{(k,l)\in G, k\neq i, l\neq i} E_{k,l}\right)\left(\prod_{j\in N_G(i)} E_{i,j}\right) X_i N_{I^c} \\ &= \left(\prod_{(k,l)\in G, k\neq i, l\neq i} E_{k,l}\right)\left(X_i \prod_{j\in N_G(i)} Z_j\right)\left(\prod_{j\in N_G(i)} E_{i,j}\right) N_{I^c} \\ &= \left(X_i \prod_{j\in N_G(i)} Z_j\right) E_G N_{I^c} \\ &= X_i Z_{N_G(i)} E_G N_{I^c} \end{aligned}$$

□

The operator $K_i := X_i(\prod_{j\in N_G(i)} Z_j)$ is called *graph stabiliser* Hein et al. (2004) at qubit i and the above lemma proves $K_i E_G N_{I^c} = E_G N_{I^c}$. Note that this equation is slightly more general than the common graph stabiliser Hein et al. (2004) as it can be applied to open graph states where input qubits are prepared in arbitrary states.

Theorem 10 *For a given projective term (G, I, O, α), if open graph (G, I, O) has causal flow $(f, \prec)$, then the pattern:*

$$\mathcal{P}_{f,G} \;:=\; \prod_{i\in O^c}^{\prec} \left(X^{s_i}_{f(i)} Z^{s_i}_{N_G(f(i))\setminus\{i\}} M_i^{\alpha_i}\right)\; E_G\; N_{I^c}\,,$$

where the product follows the dependency order $\prec$, is strongly deterministic, and realizes the unitary embedding:

$$[\![\mathcal{P}]\!] \;=\; [\![t]\!] \;=\; \left(\prod_{i\in O^c} \langle +_{\alpha_i}|_i\right)\; E_G N_{I^c}\,.$$

Proof. The proof is based on *anachronical* patterns, i.e. patterns which do not satisfy the D0 condition (see section 1.2) saying that no command depends on an outcome not yet measured. Indeed, in the anachronical pattern $M_i^\alpha Z_i^{s_i}$, the command $Z_i^{s_i}$ depends on the outcome s_i whereas the qubit i is not yet measured. However, by relaxing the D0 condition, we have the following equation:

$$\langle +_\alpha|_i = M_i^\alpha Z_i^{s_i}$$

Indeed, if $s_i = 0$ the measurement realises the projection $\langle +_\alpha|_i$, and if $s_i = 1$ the measurement realises the projection $\langle -_\alpha|_i = \langle +_\alpha|_i Z_i$. Thus, any correction-free pattern $\prod_{i\in O^c} M_i^{\alpha_i} E_G N_{I^c}$ can be turned into an anachronical strongly deterministic pattern $\prod_{i\in O^c} M_i^{\alpha_i} Z_i^{s_i} E_G N_{I^c}$ which realises U_G. The rest of the proof consists in transforming this anachronical pattern into a pattern which satisfies the D0 condition:

$$\begin{aligned}\prod_{i\in O^c} M_i^{\alpha_i} Z_{s_i}^i E_G N_{I^c} &= \prod_{i\in O^c} M_i^{\alpha_i} Z_i^{s_i} \left(X_{f(i)}^{s_i} \prod_{j\in N_G(f(i))} Z_j^{s_i}\right) E_G N_{I^c} \\ &= \prod_{i\in O^c}^{\prec} \left(X_{f(i)}^{s_i} \prod_{j\in N_G(f(i))\setminus\{i\}} Z_j^{s_i}\right) M_i^{\alpha_i} E_G N_{I^c}\end{aligned}$$

Lemma 24 and condition 3 of the causal flow are used in the previous equation for eliminating the command $Z_{s_i}^i$, whereas conditions 1 and 2 ensure that the pattern satisfies the D0 condition.□

The intuition of the proof is that entanglement between two qubits i and j converts an anachronical Z correction at i, given in the term $M_i^\alpha Z_i^{s_i}$, into a pair of a 'future' X correction on qubit j. It is easy to verify that Theorem 10 is also valid in the more generalised setting of the Phase model, by replace the Pauli X operator in Lemma 24 with $X_i^\alpha = Z_i^\alpha X_i Z_i^{-\alpha}$ which stabilises the state $|+_\alpha\rangle$. This will allow us to define the adjunction operator over the class of patterns with flow Danos and Kashefi (2006). Say an open graph state (G, I, O) has *bi-flow*, if both (G, I, O) and its dual state (G, O, I) have flow. Say a pattern has flow (bi-flow) if its underlying open graph state does.

The class of patterns with flows (bi-flows) is closed under composition and tensorization. It is also universal, in the sense that all unitaries can be realised within this class. This follows from the existence of a set of generating patterns having bi-flow as we saw in Section 1.4. Patterns with bi-flows realize unitary operators. Indeed, a flow $(f, \prec)$ is one-to-one and therefore the orbits $f^n(i)$ for $i \in I$ define an injection from I into O. In the case of a bi-flow, I and O are therefore in bijection, and since one knows already that patterns with flows realize unitary embeddings, it follows that patterns with bi-flow implement unitaries.

Interestingly, one can define directly the adjoint of a pattern in the subcategory of patterns with bi-flows. Specifically, given $(f, \prec)$ a flow for (G, I, O), and angles $\{\alpha_i; i \in I^c\}$ for preparations, and $\{\beta_j; j \in O^c\}$ for measurements, we write $\mathcal{P}_{f,G,\vec{\alpha},\vec{\beta}}$ for the pattern obtained as in the extension to general preparations of Theorem 10. Suppose a reverse flow $(g, \prec)$ is given on (G, O, I), one can define:

$$\mathcal{P}_{f,G,\vec{\alpha},\vec{\beta}}{}^\dagger \quad := \quad \mathcal{P}_{g,G,\vec{\beta},\vec{\alpha}}$$

There are two things to note here: first, for this definition to make sense, one needs to have general preparations; second, this adjunction operation depends on the choice of a reverse flow $(g, \prec)$. It is easy to see that $\mathcal{P}_{f,G,\alpha,\beta}\dagger$ and $\mathcal{P}_{f,G,\beta,\alpha}$ realize adjoint unitaries.

An example is the pattern $\mathcal{H} := X_2^{s_1} M_1^0 E_{12} N_2$ with $I = \{1\}$ and $O = \{2\}$. It has a unique bi-flow, and is self-adjoint in the sense that $\mathcal{H}\dagger = \mathcal{H}$, therefore it must realize a self-adjoint operator, and indeed it realizes the Hadamard transformation.

Generalised Flow

The existence of the causal flow is only a sufficient condition for determinism. We now extend the construction and present a necessary and sufficient condition for the stepwise uniformly deterministic computation in this model Browne et al. (2007). As we saw a flow function f assign to every single measured qubit a unique correcting vertices $f(i)$. A natural generalisation is to consider a set of vertices as a *correcting set*. Hence instead of working with a function $f : O^c \to I^c$ defining the correcting vertices, we will have a function $g : O^c \to \mathcal{P}^{I^c}$ defining the correcting sets of vertices, where $\mathcal{P}^{I^c}$ denotes the power set of all the subsets of vertices in I^c. We define the odd neighborhood of a set of vertices K to

be the set $\mathrm{Odd}(K) = \{u\,,\, |N_G(u) \cap K| = 1 \mod 2\}$, *i.e.* the set of vertices which have an odd number of neighbours in K.

Definition 25 (gflow) *$(g, \prec)$ is a gflow of (G, I, O), where $g : O^c \to \wp(I^c)$ and $\prec$ is a strict partial order over V, if and only if*

1. *if $j \in g(i)$ then $i \prec j$*
2. *if $j \in Odd(g(i))$ then $j = i$ or $i \prec j$*
3. *$i \in Odd(g(i))$*

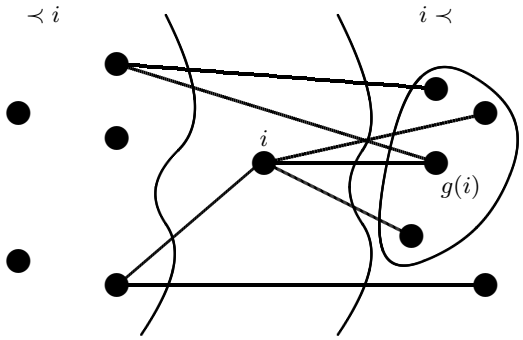


Fig. 1.5. Graphical interpretation of a gflow $(g, \prec)$: for a given vertex i all the vertices larger than i represent qubits that will be measured after the qubit i. The set $g(i)$ has to be composed of qubits measured after i (i.e. in the right layer), and such that the following parity conditions are satisfied: there is an odd number of edges between $g(i)$ and i and there is an even number of edges between $g(i)$ and any vertex which is not larger than i (in the left layer.)

A graphical interpretation of the generalised flow is given in Figure 1.5. Notice that if an open graph has a causal flow $(f, \prec)$, then it has a gflow $(g, \prec)$, where $g : i \mapsto \{f(i)\}$. Like the causal flow, the gflow is a sufficient condition for determinism:

Theorem 11 *For a given projective term $t = (G, I, O, \alpha)$, if (G, I, O) has gflow $(g, \prec)$, then the pattern:*

$$\mathcal{P}_{g,G} \;:=\; \prod_{i\in O^c}^{\prec} \prod_{j\in g(i)} \left(X_j^{s_i} Z_{N_G(j)\setminus\{i\}}^{s_i}\right) M_i^{\alpha_i}\; E_G\; N_{I^c}\,,$$

where the product follows the dependency order $\prec$, is strongly deterministic, and realises the unitary embedding:

$$[\![\mathcal{P}]\!] \;=\; [\![t]\!] \;=\; \left(\prod_{i\in O^c} \langle +_{\alpha_i}|_i\right)\; E_G N_{I^c}\,.$$

Proof. The proof is similar to the proof of theorem 10, except that for each measured qubit i, the lemma 24 is applied for every element of $g(i)$ so that it realises an X on every qubit in $g(i)$ and a Z on every qubit which has an odd number of neighbours in $g(i)$. Those qubits who have an even number of neighbours in $g(i)$ receive an even number of Zs, i.e. the identity. □

Contrary to the causal flow, the existence of a gflow is necessary for a certain kind of determinism. Recall that a pattern $\mathcal{P}$ is said to be uniformly deterministic if it is deterministic for any measurement angles. Moreover, a pattern is said to be *stepwise deterministic* if it is deterministic after performing each single measurement together with all the corrections depending on the outcomes of that measurement. More formally,

Definition 26 *A pattern $\mathcal{P}$ is* stepwise*, uniformly, and strongly deterministic if $\mathcal{P}$ is uniformly and strongly deterministic and is either measurement-free or it can be rewritten as $(V, I, O, C^{s_n} M_i^{\alpha_i} A)$ such that C is composed of Pauli commands only and $P' = (V, I, O \setminus \{n\}, A)$ is stepwise uniformly, and strongly deterministic.*

Theorem 10 and 11 can be extended to stepwise, uniform, and strong determinism.

Theorem 12 *If a pattern $\mathcal{P}$ is stepwise, uniformly, and strongly deterministic, then the underlying open graph (G, I, O) of $\mathcal{P}$ has a generalised flow and the pattern is an implementation of the projective term $t = (G, I, O, \alpha)$:*

$$[\![\mathcal{P}]\!] \quad = \quad [\![t]\!] \quad = \quad \left(\prod_{i\in O^c}\langle+_{\alpha_i}|_i\right) E_G N_{I^c}\,.$$

The next lemma will be used in the proof of Theorem 12 and illustrates the role played by the uniformity condition.

Lemma 27 *Let U and V be two n-qubit unitary transformations. If for any angle α, $\langle+_\alpha|U = \langle+_\alpha|V$ then $U = V$*

Proof. For a given angle α, and any n-qubit state $|\Psi\rangle$, $U|\Psi\rangle$ and $V|\Psi\rangle$ can be decomposed as follows:

$$U|\Psi\rangle = |+_\alpha\rangle|\phi_+\rangle + |-_\alpha\rangle|\phi_-\rangle \quad , \quad V|\Psi\rangle = |+_\alpha\rangle|\psi_+\rangle + |-_\alpha\rangle|\psi_-\rangle\,.$$

Since $\langle+_\alpha|U|\Psi\rangle = \langle+_\alpha|V|\Psi\rangle$, $|\phi_+\rangle = |\psi_+\rangle$. Moreover, since $|+_{\alpha+\pi}\rangle = |-_\alpha\rangle$, $\langle+_{\alpha+\pi}||\phi\rangle = \langle+_{\alpha+\pi}||\psi\rangle$ implies $|\phi_-\rangle = |\psi_-\rangle$. Thus $U|\Psi\rangle = V|\Psi\rangle$ for any $|\Psi\rangle$, so $U = V$. □

The following lemma extends a well-known result of the stabiliser formalism to the Measurement calculus:

Lemma 28 *For a given open graph (G, I, O) and a Pauli operator C, if $CE_G N_{I^c} = E_G N_{I^c}$ then there exists $S \subseteq I^c$ such that $C = \prod_{u\in S} X_u Z_{N_G(u)}$.*

Proof. Since $CE_G N_{I^c} = E_G N_{I^c}$, $CE_G N_V = E_G N_V$. The state $E_G N_V$ is stabilised by $\{X_u Z_{N_G(u)}, u \in V\}$, thus there exists $S \subseteq V$ such that $C = \prod_{u\in S} X_u Z_{N_G(u)}$. The rest of the proof consists in proving that $S \subseteq I^c$. By contradiction, let $i \in S \cap I$,

$$\begin{aligned}
E_G N_{I^c} &= E_G N_{I^c} Z_i Z_i \\
&= \textstyle\prod_{u\in S} X_u Z_{N_G(u)} E_G N_{I^c} Z_i Z_i \\
&= X_i Z_{N_G(i)} Z_i \textstyle\prod_{u\in S\setminus\{i\}} X_u Z_{N_G(u)} E_G N_{I^c} Z_i \\
&= -Z_i X_i Z_{N_G(i)} \textstyle\prod_{u\in S\setminus\{i\}} X_u Z_{N_G(u)} E_G N_{I^c} Z_i \\
&= -Z_i E_G N_{I^c} Z_i \\
&= -E_G N_{I^c}
\end{aligned}$$

□

Proof of Theorem 12. The proof is by induction on the number of measurements of $\mathcal{P}$. If $\mathcal{P}$ is measurement-free then the proof is obvious. Otherwise, $\mathcal{P}$ can be rewritten as $(V, I, O, C_D^{s_n} M_n^{\alpha_n} A)$, where C_D acts on qubits $D \subseteq O$. By induction, $\mathcal{P}' = (V, I, O\cup\{n\}, A)$ has a gflow $(g', \prec')$ and it realises $U' = \left(\prod_{i\in O^c\setminus\{n\}}\langle+_{\alpha_i}|_i\right) E_G N_{I^c}$.

Thus $\mathcal{P}$ realises $\langle+_{\alpha_n}|U'$ if $s_n = 0$ and $C_D\langle-_{\alpha_n}|U'$. Since P' is strongly deterministic and $\langle-_{\alpha_n}| = \langle+_{\alpha_n}|Z_n$,

$$\left(\prod_{i\in O^c\setminus\{n\}}\langle+_{\alpha_i}|_i\right) C_D Z_n E_G N_{I^c} = \left(\prod_{i\in O^c\setminus\{n\}}\langle+_{\alpha_i}|_i\right) E_G N_{I^c}$$

Thus, according to Lemma 27, $C_D Z_n E_G N_{I^c} = E_G N_{I^c}$, and thanks to lemma 28, there exists $S \subseteq I^c$ s.t. $C_D = Z_n \prod_{u\in S} X_u Z_{N_G(u)}$.

Notice that $S \subseteq D$ since X commands cannot be cancelled out in C_D. Moreover, for any $v \notin D$, an even number of Zs are applied on v such that they cancelled out. As a consequence, $v \in \mathrm{Odd}(S)$ implies $v \in D \subseteq O \cup \{n\}$. Let $g : O^c \to I^c$ s.t. $g(i) := g'(i)$ if $i \neq n$ and $g(n) := S$. Let R be a relation s.t. $(u, v) \in R$ if $u \prec' v \vee (u = n \wedge v \in S)$, and let $\prec$ be the transitive closure of R. $(g, \prec)$ is then a gflow of (G, I, O). □

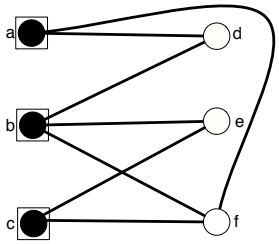


Fig. 1.6. A graph with generalised flow but no flow: $g(a) = d, g(b) = e, g(c) = \{d, f\}$. The blue arrows represent the flow edges, where as black arrow indicates a virtual flow edge, (an edge that is not an edge of the graph state).

The open graph state in Figure 1.6 has no flow (due the cyclic connections), but it admits a generalized flow. This example demonstrates the fact having flow is not a necessary condition for uniform determinism, contrary to the existence of the generalized flow.

1.10 Finding Optimal Flow Efficiently

In this section, efficient algorithms for finding causal flow and generalised flow of open graph state are presented Mhalla and Perdrix (2008). These algorithms can be used for deciding whether a given open graph admits a deterministic pattern. Moreover, these algorithms produce flows of minimal depth such that they can be used for an automatic complexity depth optimisation. First note that a *flow* (flow denotes a gflow or a causal flow), $(g, \prec)$ of (G, I, O) induces a partition of the vertices of the open graph:

Definition 29 *For a given open graph (G, I, O) and a given flow $(g, \prec)$ of (G, I, O), let*

$$V_k^{\prec} = \begin{cases} \mathsf{max}_{\prec}(V) & \text{if } k = 0 \\ \mathsf{max}_{\prec}(V \setminus L_{k-1}^{\prec}) & \text{if } k > 0 \end{cases}$$

where $\mathsf{max}_{\prec}(X) = \{u \in X \text{ s.t. } \forall v \in X, \neg(u \prec v)\}$ is the set of the maximal elements of X and $L_k^{\prec} = (\cup_{i \leq k} V_i^{\prec})$. The depth *$d^{\prec}$ of the flow is the smallest d such that $V_{d+1}^{\prec} = \varnothing$. $(V_k^{\prec})_{k=0 \ldots d^{\prec}}$ is a partition of V into $d^{\prec} + 1$ layers.*

A causal flow or a gflow $(g, \prec)$ of (G, I, O) leads to a correction strategy for the corresponding pattern $\mathcal{P}$, which consists in measuring the non output qubits of each layer in parallel, from the layer $V_{d^{\prec}}^{\prec}$ to the layer $V_0^{\prec}$:

$$\mathcal{P} = \left(V, I, O, R^{(0)} R^{(1)} \ldots R^{(d^{\prec})} E_G N_{I^c}\right)$$

where $R^{(k)} := \left(\prod_{i \in V_k^{\prec} \setminus O} \left(\prod_{j \in g(i)} X_i Z_{N_G(i)}\right) M_i^{\alpha_i}\right)$.

The complexity depth of such a pattern is upperbounded by $d^{\prec} + 1$. Causal and generalised flows are not unique in general. In the following, a subfamily of flows are considered, the *maximally delayed* flows, which have an inductive structure (lemmas 31, 32) and are of minimal depth (theorem 13.)

Definition 30 *For a given open graph (G, I, O) and two given causal flows (resp. gflows) $(g, \prec)$ and $(g', \prec')$ of (G, I, O), $(g, \prec)$ is more delayed than $(g', \prec')$ if $\forall k$, $|L_k^{\prec}| \geq |L_k^{\prec'}|$ and there exists a k such that the inequality is strict. A causal flow (resp. gflow) $(g, \prec)$ is* maximally delayed *if there exists no causal flow (resp. gflow) of the same open graph that is more delayed.*

For instance, the flow $(g, \prec)$ described in Figure 1.4 is a maximally delayed causal flow. However, $(g, \prec)$ is not a maximally delayed gflow since $(g', \prec')$ is a more delayed gflow, where $g'(a_0) = \{b_0, b_1, b_2\}, g'(a_1) = \{b_1, b_2\}, g'(a_2) = \{b_2\}, g'(b_0) = \{c_0\}, g'(b_1) = \{c_1\}, g'(b_2) = \{b_2\}$, and $\{a_0, a_1, a_2\} \prec' \{b_0, b_1, b_2\} \prec' \{c_0, c_1, c_2\}$. One can prove that $(g', \prec')$ is a maximally delayed gflow.

Lemma 31 *If $(g, \prec)$ is a maximally delayed gflow of (G, I, O), then*

$$\begin{array}{lcl} V_0^{\prec} & = & O \\ V_{k+1}^{\prec} & = & \{u \in V \setminus L_k^{\prec}, \exists K \subseteq L_k^{\prec} \setminus I, \mathrm{Odd}(K) \setminus L_k^{\prec} = \{u\}\} \end{array}$$

Proof. First, notice that for any gflow $V_0^{\prec} \subseteq O$ and $V_{k+1}^{\prec} \subseteq \{u \in V \setminus L_k^{\prec}, \exists K \subseteq L_k^{\prec} \setminus I, , \mathrm{Odd}(K) \setminus L_k^{\prec} = \{u\}\}$. The rest of the proof consists in proving that these inclusions are saturated for maximally delayed gflows. Let $\prec' := \prec \setminus ((O \setminus V_0^{\prec}) \times V)$. Notice that $V_0^{\prec'} = O$. Moreover, $(g, \prec')$ is a gflow of (G, I, O). Since $(g, \prec')$ is not more delayed than $(g, \prec)$, $|V_0^{\prec}| \geq |V_0^{\prec'}| = |O|$, so $V_0^{\prec} = O$.

For the second inclusion, by contradiction, assume that there exist k, u_0, and $K \subseteq L_k^{\prec}$ such that $u_0 \notin V_k$, $u_0 \in V \setminus L_k^{\prec}$, and $\mathrm{Odd}(K) \setminus L_k^{\prec} = \{u\}$. Let $(g'', \prec'')$ such that $g''(u) := g(u)$ if $u \neq u_0$; $g''(u_0) := K$, $u \prec'' v$ if $u \neq u_0 \wedge u \prec v$; and $u_0 \prec v$ if $v \in K$. It leads to a contradiction since $(g'', \prec'')$ is a gflow of (G, I, O) which is more delayed than $(g, \prec)$. □

In a similar way, one can prove that:

Lemma 32 *If $(g, \prec)$ is a maximally delayed causal flow of (G, I, O), then*

$$\begin{array}{lcl} V_0^{\prec} & = & O \\ V_{k+1}^{\prec} & = & \{u \in V \setminus L_k^{\prec}, \exists v \in L_k^{\prec} \setminus I, N_G(v) \setminus L_k^{\prec} = \{u\}\} \end{array}$$

Theorem 13 *A maximally delayed causal flow (resp. gflow) is of minimal depth.*

Proof. Let $(g, \prec)$ be a minimal depth causal flow (resp. gflow) of a given open graph. If $(g, \prec)$ is a maximally delayed causal flow (resp. gflow), then let $(g', \prec') := (g, \prec)$. Otherwise, let $(g', \prec')$ be a maximally delayed causal flow (resp. gflow) which is more delayed than $(g, \prec)$. $(g', \prec')$ and $(g, \prec)$ have the same depth. Indeed $|L_k^{\prec'}| \geq |L_k^{\prec}| = |V|$, thus $\forall k > d^{\prec}$, $V_k^{\prec} = \varnothing$, so $d^{\prec} \geq d^{\prec'}$. Since $(g, \prec)$ is of minimal depth $d^{\prec} \leq d^{\prec'}$, so $d^{\prec} = d^{\prec'}$. As a consequence $(g', \prec')$ is a minimal depth maximally delayed causal flow (resp. gflow). Moreover, even if a maximally delayed causal flow (resp. gflow) of a given open graph is not unique, one can prove, using Lemmas 31 and 32, that all the maximally delayed causal flows (resp. gflows) of a given open graph induce the same partition of the vertices, and as a consequence, have the same depth. Thus, the maximally delayed causal flow (resp. gflow) produced by the algorithm has the same depth than $(g', \prec')$. □

Generalised flow algorithm

Theorem 14 *There exists a polynomial time algorithm for deciding whether a given open graph has a gflow. Moreover, if the open graph has a gflow then the algorithm outputs a gflow of minimal depth.*

Sketch of Proof. The algorithm and a complete proof are given in Mhalla and Perdrix (2008). Here we give the main ideas of the algorithm. This algorithm is searching for a maximally delayed gflow. As a consequence, if the open graph has a gflow, then the gflow produced by the algorithm is of minimal depth. This is a backward recursive algorithm based on the inductive structure of maximally delayed gflows (Lemma 31.) At the first recursive call, the algorithm produces $V_0 := O$. At the k^{th} recursive call, the algorithm produces the set V_{k+1} of vertices $u \in V \setminus L_k$ such that $\exists K \subseteq L_k \setminus I, \mathrm{Odd}(K) \setminus L_k = \{u\}$. Notice that for each u, there is a exponential number of candidates $K \subseteq L$, however an exponential slowdown is avoided by encoding this problem into a linear system over $\mathbb{F}_2$:

$$\Gamma_{V \setminus L_k, L_k \setminus I} \begin{pmatrix} x_1 \\ \vdots \\ x_{|L_k \setminus I|} \end{pmatrix} = \mathbb{I}_{\{u\}}^{V \setminus L_k}$$

where $\Gamma_{A,B} := \{\gamma_{i,j}\}_{i\in A, j\in B}$ is a $|A| \times |B|$-sub-matrix of the adjacency matrix $\Gamma = \{\gamma_{i,j}\}_{i,j\in V}$ of the graph G, and $\mathbb{I}_A^B$ stands for a $|B|$-dimensional vector defined by $\mathbb{I}_A^B(i) = 1$ if $i \in A$ and $\mathbb{I}_A^B(i) = 0$ otherwise.

If $x_1, \ldots, x_{|L_k \setminus I|} \in \mathbb{F}_2$ is a solution to the system, then on can prove that the set $K \subseteq L_k \setminus I$ such that $\mathbb{I}_K^{L_k \setminus I} = \begin{pmatrix} x_1 \\ \vdots \\ x_{|L_k \setminus I|} \end{pmatrix}$ satisfies $\mathrm{Odd}(K) \setminus L_k = \{u\}$. □

Causal flow algorithm

A first algorithm for finding a causal flow has been proposed in de Beaudrap (2008), and works only if the numbers of inputs and outputs are the same. The complexity of the algorithm is in $O(nm)$ where n is the number of vertices and m the number of edges (more precisely $O(km)$ where k is the number of inputs (outputs) de Beaudrap (2008).) A more general and faster algorithm has been introduced in Mhalla and Perdrix (2008):

Theorem 15 *Mhalla and Perdrix (2008) For a given open graph (G, I, O), finding a causal flow can be done in $O(m)$ operations where $m = |E(G)|$ is the number of edges of the graph G. Moreover, if the open graph has a gflow then the algorithm outputs a gflow of minimal depth.*

The proof is based on the recursive structure of maximally delayed causal flow and is presented in Mhalla and Perdrix (2008). Notice that some specific data structures are used for optimising the complexity of the algorithm and obtaining a linear algorithm.

1.11 Extended Measurement Calculus

The measurements considered until now are (X, Y)-measurements, i.e. measurements defined by orthogonal projections on

$$\begin{aligned} |+_\alpha^{(X,Y)}\rangle &:= \tfrac{1}{\sqrt{2}}(|0\rangle + e^{i\alpha}|1\rangle) \\ |-_\alpha^{(X,Y)}\rangle &:= \tfrac{1}{\sqrt{2}}(|0\rangle - e^{i\alpha}|1\rangle) \end{aligned}$$

In this section the Measurement Calculus is extended to measurements on (X, Z) and (Y, Z) planes Perdrix (2006); Browne et al. (2007). Such an extension increases the expressive power of the formalism, allowing for instance the representation of one-way quantum computations based on (X, Z) measurements, while the computational power is unchanged Perdrix (2006). The extended Measurement Calculus consists in allowing (X, Y)-, (Y, Z)-, and (X, Z)-measurements on non input qubits:

- 1-qubit auxiliary preparation N_i
- 2-qubit entanglement operators E_{ij}
- 1-qubit measurements $M_i^{\lambda,\alpha}$, with $\lambda \in \{(X, Y), (Y, Z), (X, Z)\}$
- and 1-qubit Pauli operators corrections X_i and Z_i

(Y, Z)- and (X, Z)-measurements are defined by orthogonal projections on respectively,

$$\begin{aligned} |+_\alpha^{(Y,Z)}\rangle &:= \cos(\alpha/2)|0\rangle + i\sin(\alpha/2)|1\rangle \\ |-_\alpha^{(Y,Z)}\rangle &:= \sin(\alpha/2)|0\rangle - i\cos(\alpha/2)|1\rangle \end{aligned}$$

and

$$\begin{aligned} |+_\alpha^{(X,Z)}\rangle &:= \cos(\alpha/2)|0\rangle + \sin(\alpha/2)|1\rangle \\ |-_\alpha^{(X,Z)}\rangle &:= \sin(\alpha/2)|0\rangle - \cos(\alpha/2)|1\rangle \end{aligned}$$

Pauli operators X and Z can be pushed through the measurements:

$$M_i^{(X,Y),\alpha} X_i = M_i^{(X,Y),-\alpha} \quad (1.24)$$
$$M_i^{(X,Y),\alpha} Z_i = M_i^{(X,Y),\alpha-\pi} \quad (1.25)$$
$$M_i^{(X,Z),\alpha} X_i = M_i^{(X,Z),-\alpha} \quad (1.26)$$
$$M_i^{(X,Z),\alpha} Z_i = M_i^{(X,Z),-\alpha+\pi} \quad (1.27)$$
$$M_i^{(Y,Z),\alpha} X_i = M_i^{(Y,Z),\alpha-\pi} \quad (1.28)$$
$$M_i^{(Y,Z),\alpha} Z_i = M_i^{(Y,Z),-\alpha+\pi} \quad (1.29)$$

The semantics of the patterns is extended to the measurements in (X,Z) and (Y,Z) planes, as follows:

$$V \cup \{i\}, W, q, \Gamma \xrightarrow{t[M_i^{\lambda,\alpha}]^s} V, W \cup \{i\}, \langle +^{\lambda}_{\alpha_\Gamma}|_i q, \Gamma[0/i]$$
$$V \cup \{i\}, W, q, \Gamma \xrightarrow{t[M_i^{\lambda,\alpha}]^s} V, W \cup \{i\}, \langle -^{\lambda}_{\alpha_\Gamma}|_i q, \Gamma[1/i]$$

Theorem 16 *For any* extended *pattern $\mathcal{P}$ of size n, there exists a pattern $\mathcal{P}$ of size $O(n)$ composed of (X,Y)-measurements only, such that $[\![\mathcal{P}]\!] = [\![\mathcal{P}']\!]$.*

Proof. The proof consists in replacing in $\mathcal{P}$ all occurrences of (X,Z) and (Y,Z) measurements by (X,Y) measurements, using the following equations of projectors:

$$\langle +_\alpha^{(X,Z)}| = \langle +_\alpha| J(-\pi/2) J(0)$$
$$\langle -_\alpha^{(X,Z)}| = \langle -_\alpha| J(-\pi/2) J(0)$$

$$\langle +_\alpha^{(Y,Z)}| = \langle +_\alpha| J(-\pi/2) J(-\pi/2)$$
$$\langle -_\alpha^{(Y,Z)}| = \langle -_\alpha| J(-\pi/2) J(-\pi/2)$$

Thus, we define the following set of rewrite rules:

$$M_i^{(X,Z),\alpha} \Rightarrow_{XY} M_k^{(X,Y),\alpha} \circ \mathcal{J}(-\pi/2)(j,k) \circ \mathcal{J}(0)(i,j)$$
$$M_i^{(Y,Z),\alpha} \Rightarrow_{XY} M_k^{(X,Y),\alpha} \circ \mathcal{J}(-\pi/2)(j,k) \circ \mathcal{J}(-\pi/2)(i,j)$$

This rewriting system terminates, and the size of the resulting pattern is linear in the size of the original pattern. Moreover, the rewrite rules preserve the semantics of the patterns. □

An *extended open graph* (or open graph for short) is a quadruplet (G, I, O, λ), where $\lambda : O^c \to \{(X,Y),(Y,Z),(X,Z)\}$ is a labelling function which associates with any measurement, the plane in which this measurement is performed. A correction-free extended pattern is of the following form:

$$\mathcal{P} = (V, I, O, \prod_{i \in V \setminus O} M_i^{\lambda(i),\alpha_i} \prod_{(i,j)\in G} E_{ij} N_{V \setminus I})$$

In the context of the extended Measurement calculus, the *extended gflow* (or gflow for short) is defined as follows Browne et al. (2007):

Definition 33 (Extended gflow) *$(g, \prec)$ is an extended gflow of (G, I, O, λ), where $g : O^c \to \wp(I^c)$ and $\prec$ is a strict partial order over V, if and only if*

1. $j \in g(i) \vee j \in \mathrm{Odd}(g(i)) \implies j = i \vee i \prec j$
2. $i \in \mathrm{Odd}(g(i)) \implies \lambda(i) \in \{(X,Y),(X,Z)\}$
3. $i \in g(i) \implies \lambda(i) \in \{(Y,Z),(X,Z)\}$

Theorem 17 *A measurement-free extended pattern $\mathcal{P} = (G, I, O, \prod_{i\in O^c} M_i^{\lambda(i),\alpha_i} E_G N_{I^c})$ can be completed with Pauli corrections into a stepwise uniformly and strongly deterministic pattern if and only if (G, I, O, λ) has a gflow.*

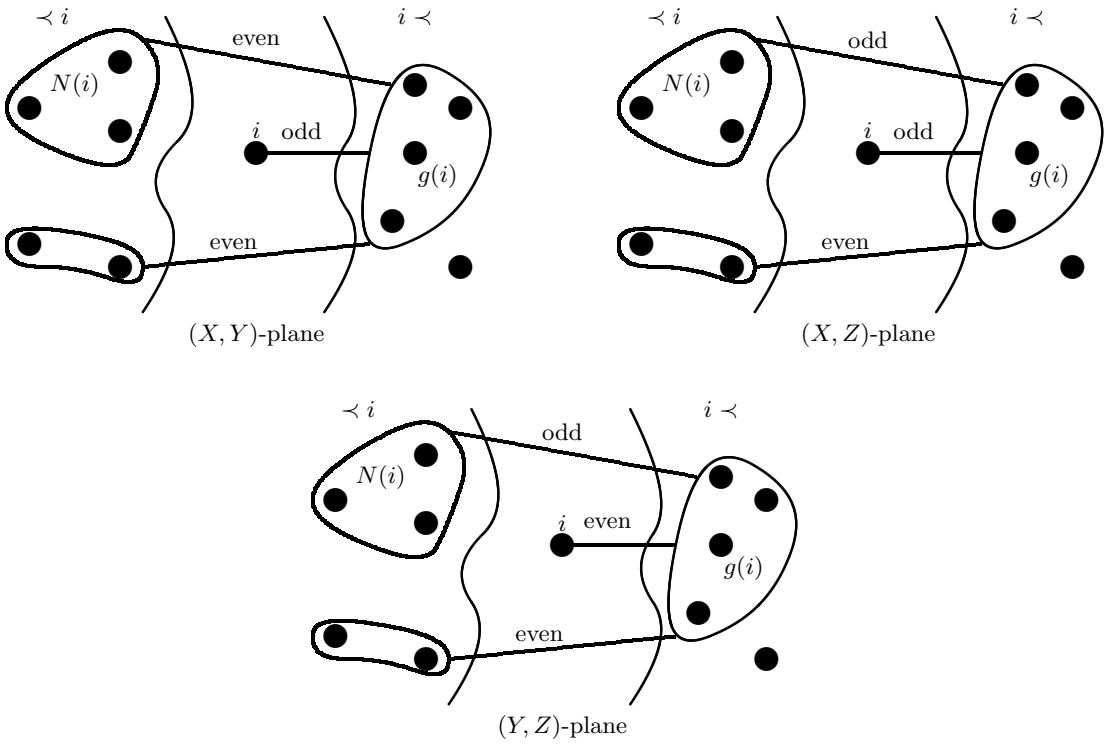


Fig. 1.7. The pictorial presentation of the extended flow conditions for different measurement planes. The straight lines stand for multiple edges in the entanglement graph where the labels give the parity of the number of these connections. The qubit i denotes the qubit to be measured, its correcting set $g(i)$ lays in '$i \prec$' layer. The '$\prec i$' layer of vertices $j \prec i$ is splitted into two sets: the neighbours of i labeled with $N(i)$ and the other vertices.

Proof. The proof is similar to Theorems 11 and 12. In particular, it is based on the following anachronical corrections:

$$\begin{aligned} M_i^{(X,Y),\alpha} Z_i^{s_i} &= \langle +_{\alpha_i}^{(X,Y)}| \\ M_i^{(Y,Z),\alpha} X_i^{s_i} &= \langle +_{\alpha_i}^{(Y,Z)}| \\ M_i^{(X,Z),\alpha} X_i^{s_i} Z_i^{s_i} &= \langle +_{\alpha_i}^{(X,Z)}| \end{aligned}$$

According to lemma 24, for any $i \in I^c$, $E_G N_{I^c} = C E_G N_{I^c}$, with $C = \pm Z_{\mathrm{Odd}(g(i))} X_{g(i)}$. Thanks to condition 1 of the extended gflow, C does not act on already measured qubit. Moreover, according to condition 2 and 3, the action of C on qubit i is: Z if $\lambda(i) = (X, Y)$; X if $\lambda(i) = (Y, Z)$; and ZX if $\lambda(i) = (X, Z)$. So the action of C on i is exactly the action required for transforming an anachronical pattern into a valid one. □

Finally, the algorithm for finding optimal gflow efficiently can also be adapted to the case of the extended gflow.

Pauli Measurements

Pauli measurements play a central role in one-way quantum computing. In particular, it is known that the action of such a measurement on a graph state is to leave the remaining qubits in a graph state (up to a local Clifford-group correction) Hein et al. (2004). Definition 33 provides conditions for determinism when single qubits at any angle in specified Bloch-sphere planes are allowed. The special properties of Pauli measurements (for example, that they simultaneously lie in two measurement-planes) mean that if one restricts the measurement of certain qubits to certain specific Pauli measurements, one must extend the generalised flow conditions in order to account for these extra properties.

In this section, we introduce such an extension Browne et al. (2007). We will use the convention that the labeling function $\lambda(i)$ for any non output qubit i, is either a plane – (X, Y), (X, Z), or (Y, Z) – or a

vector – X, Y, or Z (*i.e.* Pauli measurements). First, notice that a Pauli measurement, say X, can be interpreted as a (X,Y) or (X,Z) measurement and thus it may satisfies the conditions of either a (X,Y) or a (X,Z) measurement. Second, when a qubit is measured according to a Pauli operator, say X, then, after the measurement, the state of this qubit takes $\pm X$ as its stabiliser. We use this property to allow already-measured qubits to be included in a correcting set. Finally the following relation between Pauli correction and Pauli measurements will be used for the Pauli flow construction

$$M^X X = M^X \tag{1.30}$$
$$M^Y Y = M^Y \tag{1.31}$$
$$M^Z Z = M^Z \tag{1.32}$$

Definition 34 *An open graph state (G,I,O,λ) has* Pauli flow *if there exists a map $p : O^c \to \mathcal{P}^{I^c}$ (from measured qubits to a subset of prepared qubits) and a partial order $<$ over V such that for all $i \in O^c$,*
—(P1) if $j \in p(i)$, $i \neq j$, and $\lambda(j) \notin \{X,Y\}$ then $i < j$,
—(P2) if $j \leq i$, $i \neq j$, and $\lambda(j) \notin \{Y,Z\}$ then $j \notin \mathrm{Odd}(p(i))$,
—(P3) if $j \leq i$, $j \in p(i)$ and $\lambda(j) = Y$ then $j \in \mathrm{Odd}(p(i))$,
—(P4) if $\lambda(i) = (X,Y)$ then $i \notin p(i)$ and $i \in \mathrm{Odd}(p(i))$,
—(P5) if $\lambda(i) = (X,Z)$ then $i \in p(i)$ and $i \in \mathrm{Odd}(p(i))$,
—(P6) if $\lambda(i) = (Y,Z)$ then $i \in p(i)$ and $i \notin \mathrm{Odd}(p(i))$,
—(P7) if $\lambda(i) = X$ then $i \in \mathrm{Odd}(p(i))$,
—(P8) if $\lambda(i) = Z$ then $i \in p(i)$,
—(P9) if $\lambda(i) = Y$ then either: $i \notin p(i)$ & $i \in \mathrm{Odd}(p(i))$ or $i \in p(i)$ & $i \notin \mathrm{Odd}(p(i))$.

Theorem 18

Suppose the open graph state (G,I,O,λ) has Pauli flow $(g,>)$, then the pattern:

$$\mathcal{P}_{g,G} := \prod\nolimits^{>}_{i \in O^c} \left(X^{s_i}_{g(i) \cap \{j\,,\, j>i\}} Z^{s_i}_{\mathrm{Odd}(g(i)) \cap \{j\,,\, j>i\}} \, M_i^{\lambda(i),\alpha_i} \right) E_G N_{I^c}\,,$$

where the product follows the dependency order $>$, is deterministic and realizes the unitary embedding:

$$U_G := \left(\prod\nolimits_{i \in O^c} \langle +_{\lambda(i),\alpha_i}|_i \right) E_G N_{I^c}\,.$$

Proof: The proof is similar to the proof of theorem 17. In $(P1)$, if $\lambda(j) \in \{X,Y\}$, j may be in the $p(i)$ even if $j \leq i$ since $M_i^X X_i = M_i^X$ and $M_i^Y X_i Z_i = M_i^Y$. Notice that if $\lambda(j) = Y$, $j \leq i$ and $j \in p(i)$ then j must be in $\mathrm{Odd}(p(i))$ – $(P3)$ – because of the Z_i command in $M_i^Y X_i Z_i = M_i^Y$. In $(P2)$, if $\lambda(j) = Z$, then j may be in $\mathrm{Odd}(p(i))$ even if $j \leq i$, since $M_i^Z Z_i = M_{iz}$. The condition $\lambda(j) \neq Y$ in $(P2)$ is necessary because of $(P3)$. Finally, $(P7), (P8)$, and $(P9)$ are obtained from $(P4), (P5)$, and $(P6)$ since a X measurement is both a (X,Y) and a (X,Z) measurement, and so on.□

1.12 MBQC vs Circuit

In this final section we present a novel automated technique for parallelizing quantum circuits via the forward and backward translation to measurement-based quantum computing patterns, and analyze the trade off in terms of depth and space complexity Broadbent and Kashefi (2009). The development of parallel quantum circuits seems almost essential if we wish to implement quantum algorithms in the near future with the available technology. Due to decoherence, qubits have a tendency to spontaneously change their state, hence we can only operate on them for a very short period of time. Parallel circuits could maximize the use of these fragile qubits. As for theoretical motivation, the study of parallel quantum algorithms could lead to new results in complexity theory. For instance, one interesting open question is whether the class of decision problems solvable in polynomial time, **P**, is included in the class of decision problems solvable in polylogarithmic depth, **NC**. Let **QNC** be the class of decision problems solvable in polylogarithmic depth with a quantum computer, one can ask similarly whether **P** is included in **QNC**. Finally, Richard Jozsa conjectured that:

Jozsa Conjecture. Jozsa (2005) *Any polynomial-time quantum algorithm can be implemented with only $O(\log(n))$ quantum layers interspersed with polynomial-time classical computations.*

We first introduce few definitions, recall that the size of a circuit is the number of gates and its depth is the largest number of gates on any input-output path. Equivalently, the depth is the number of layers that are required for the parallel execution of the circuit, where a qubit can be involved in at most one interaction per layer. Here we adopt the model according to which at any given timestep, a single qubit can be involved in at most one interaction. This differs from the concurrency viewpoint, according to which all interactions for commuting operations can be done simultaneously.

Naturally the depth of a standard one-way pattern is the sum of the depths of the preparation (minimum numbers of layers required for the parallel preparation of the entanglement graph) and computation parts (the longest chain of adaptive measurements). Here, we only consider standard patterns, this is justified due to the existence of the universal standardisation procedure and the result in Broadbent and Kashefi (2009) which proved that the procedure of standardisation will decrease the depth of a pattern.

Lemma 35 *The preparation depth for a given entanglement graph G, is either $\Delta(G)$ or $\Delta(G)+1$.*

Proof. At each timestep, a given qubit can interact with at most one other qubit. In terms of the entanglement graph, this means that at each timestep, a given node can interact with at most one of its neighbours. Assign a colour to each timestep and colour the edge in the entanglement graph G accordingly. With this view, the entire preparation corresponds to an edge colouring of the entanglement graph. By Vizing's theorem Diestel (2005), the edge-chromatic number of G, $\chi'(G)$ satisfies $\Delta(G) \leq \chi'(G) \leq \Delta(G)+1$. □

For the purpose of parallelising quantum circuits it is enough to consider only standard patterns with flow as any quantum circuit will corresponds to such a pattern. This will allow is to obtain an upper bound on the depth of a pattern based on the depth of its flow, based on the important notion of *influencing walks* for open graph states with flow Broadbent and Kashefi (2009).

Definition 36 *Let $(f, \preceq)$ be the flow of a geometry (G, I, O). Any input-output walk in G that starts with a flow edge, has no two consecutive non-flow edges and traverses flow edges in the forward direction, is called an* influencing walk.

Proposition 37 *Let a and b be two qubits in a standard pattern with flow. If b depends on a, then a appears before b on a common influencing walk, and this holds both before and after signal shifting.*

Proof. This is a consequence of the flow theorem. Recall that before signal shifting, a measurement at a qubit j is X-dependent on the result of a measurement at another qubit i if and only if $j = f(i)$ that is, a flow edge between qubits i and j. Also a measurement at a qubit k is Z-dependent on the result of a measurement at another qubit i if and only if $j = f(i)$ and k is connected to j, that is a non-flow edge between qubits j and k connected to a flow edge between qubits i and j. Therefore signal shifting creates new dependencies only through influencing walks. Hence if qubit b depends on qubit a, it is either via a direct X or Z dependency or due to a sequence of dependencies after signal shifting, in all the cases a and b must be on a common influencing walk. □

Proposition 37 tells us that in order to compute the quantum depth of a standard pattern with flow (to which we either have or haven't applied signal shifting), it suffices to consider the depth along influencing walks. Furthermore, it's not hard to see that if a geometry has a flow, all of its influencing walks are of finite length. Note that after signal shifting, Z-dependencies coming from the non-flow edges on an influencing walk no longer contribute to the pattern depth, as the dependencies that they represent are pushed to the final correction on an output qubit. On the other hand, signal shifting can create new X-dependencies. The following proposition presents an upper bound on the effect of signal shifting on the pattern depth.

Proposition 38 *Let $\mathcal{P}$ be a pattern with flow where standardization and signal shifting have been performed. Then the maximum number of flow edges, minus the number of the non-flow edges on such*

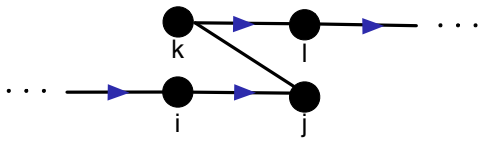


Fig. 1.8. Part of an influencing walk where two sequence of consecutive flow edges are connected with a non-flow edge.

walk (maximum taken over all possible influencing walks), plus 1 is an upper bound for the depth of the pattern.

Proof. We show that for any influencing walk, its number of flow edges minus the non-flow edges gives an upper bound on its depth. Then, by Proposition 37, it suffices to find the largest number of flow edges along any influencing walk in order to have an upper bound on the depth. We add 1 to this depth since the depth is the number of vertices of such walk, and not the number of edges.

Consider an influencing walk I. The flow edges represent X-dependencies hence each flow edge in a sequence of consecutive flow edges contributes to the depth along I. Now, consider a configuration with a non-flow edge as shown in Figure 1.8. Before signal shifting, the dependent measurements on qubits i, j, k and ℓ are given as follows where A, B and C stand for general signals not including s_i, s_j, s_k and s_ℓ

$$\dots {}^{D}[M_\ell^{\alpha_\ell}]^{s_k}\, {}^{C+s_i}[M_k^{\alpha_k}]^{B}\, {}^{A}[M_j^{\alpha_j}]^{s_i} \dots$$

and after signal shifting we have

$$\begin{aligned} & \dots {}^{D}[M_\ell^{\alpha_\ell}]^{s_k}\, {}^{C+\boxed{s_i}}[M_k^{\alpha_k}]^{B}\, {}^{A}[M_j^{\alpha_j}]^{s_i} \dots \\ \Rightarrow \quad & \dots \boxed{{}^{D}[M_\ell^{\alpha_\ell}]^{s_k} S_k^{s_i}}\, {}^{C}[M_k^{\alpha_k}]^{B}\, {}^{A}[M_j^{\alpha_j}]^{s_i} \dots \\ \Rightarrow \quad & \dots S_k^{s_i}\, {}^{D}[M_\ell^{\alpha_\ell}]^{s_k+s_i}\, {}^{C}[M_k^{\alpha_k}]^{B}\, {}^{A}[M_j^{\alpha_j}]^{s_i} \dots \end{aligned}$$

Therefore qubits j and ℓ are in the same layer. In other words, after signal shifting, the first flow edge after every non-flow edge does not contribute to the depth of the pattern. Also, any new X-dependency created with signal shifting will not increase the depth. Hence from the total number of flow edges on an influencing walk, we need to subtract the number of non-flow edges. □

So far we have not taken into account the information about the angles, which is why our bounds are not tight. We first describe the effect of the Pauli measurements on depth. The following identities are useful

$$M_i^{\frac{\pi}{2}} X_i^s = M_i^{\frac{\pi}{2}} Z_i^s \tag{1.33}$$

$$M_i^0 X_i^s = M_i^0 \tag{1.34}$$

According to Equation (1.33), when a qubit i is measured with angle $\frac{\pi}{2}$ (Pauli Y measurement), then any X-dependency on this qubit is the same as a Z-dependency. But after signal shifting, this Z-dependency does not directly contribute to the depth and hence we might obtain a smaller depth. Furthermore, there exists a special case where if qubit i is not an input qubit and also not the flow image of any other vertex ($\forall j : i \neq f(j)$) and qubit i is measured with $\frac{\pi}{2}$, then one can permit in the flow theorem, to have $f(i) = i$ and hence we will have one less flow edge Danos and Kashefi (2006). This allows an influencing walk to have a loop edge on this particular vertex measured with Pauli Y and hence the influencing walk will not start with an input qubit. We will consider only this extended notion of influencing walk that takes into account the angles of measurement. When we want to emphasize this extended definition, we will refer to *Pauli influencing walks*.

According to Equation (1.34), another special case is when qubit i is measured with angle 0 (Pauli X measurement), then any X-correction on qubit i can be ignored and in fact qubit i can be put at the first level of measurement. Consequently, again the flow depth can become smaller. By adding equations (1.33) and (1.34) to the flow theorem, the proof still works Danos and Kashefi (2006) and we get a potential improvement on the depth complexity. We refer to this procedure as *Pauli simplification*.

Another way of realizing these special cases is that after signal shifting, the Pauli measurements become independent measurements and hence can all be performed at the first level of the partial order. Hence in computing the depth of a pattern with flow after signal shifting is performed, one should disregard the Pauli measurements:

Proposition 39 *Let $\mathcal{P}$ be a pattern with flow where standardization, Pauli simplification and signal shifting have been performed. Let I_i be a Pauli influencing walk of $\mathcal{P}$, denote by e_i the number of the flow edges, by n_i the number of non-flow edges, by p_i number of flow edges pointing to a qubit to be measured with a Pauli measurement and by ℓ_i the number of loop edges ($\ell_i \in \{0, 1\}$). Then the depth of the pattern, call it $D_{\mathcal{P}}$ satisfies the following formula:*

$$D_{\mathcal{P}} \leq \max_{I_i} e_i - (n_i + p_i + \ell_i) + 1\,.$$

Proof. Along any Pauli influencing walk, any flow edge pointing to a qubit to be measured by a Pauli X will not require a separate layer (Equation (1.34)) and for the Pauli Y case, such a flow edge is converted to a Z-dependency (Equation (1.33)), to be signal shifted as in Corollary 38. Also if the influencing walk starts with a Y measurement followed by a non-Pauli measurement, we have a loop edge and hence the immediate following non-Pauli measurement can also be put in the first layer and hence we subtract the loop edge from the total depth for this influencing walk. □

1.12.1 From circuits to patterns

The original universality proof for MBQC already contained a method to translate a quantum circuit containing arbitrary 1-qubit rotations and control-not gates to a pattern Raussendorf and Briegel (2001). Here, we give an alternate method for the translation of a given circuit to a standard pattern in the MBQC to attempt to reduce the quantum depth. We give the exact tradeoff in terms of the number of auxiliary qubits and depth.

Recall that $\wedge Z$ is self-inverse and symmetric, hence any circuit that contains consecutive $\wedge Z$ gates acting on the same qubits can be simplified. In what follows, we suppose that this simplification has been performed.

Definition 40 *Let C be a circuit of $\wedge Z$ and J gates on n logical qubits. The corresponding standard pattern $\mathcal{P}$ is obtained by replacing each gate in C with its corresponding pattern, and then performing standardization and signal shifting.*

To present the exact tradeoff for the above translation, in particular to prove that the quantum depth cannot increase, we construct directly the underlying geometry of a given circuit. Following the literature, we refer to the circuit qubits as *logical* qubits. Other qubits that are added during construction of the entanglement graph will be referred to as *auxiliary* qubits.

Definition 41 *Let C be a circuit of $\wedge Z$ and J gates on n logical qubits. The* entanglement graph G_C *is constructed as a layer that is initially built on top of the circuit C by the following steps (see also the example of Figure 1.9).*

1. *Replace each $\wedge Z$ gate on logical qubits i and j with a vertical edge between two vertices: one on the i^{th} wire and one on the j^{th} wire. Label both vertices* Input/Output*. Replace each J gate on a logical qubit i with an horizontal edge between two vertices on the i^{th} wire, label the left vertex* Input *and the right vertex* Output*.*
2. *To connect the above components, on each wire, start from the left and contract consecutive non-adjacent vertices as follows (the* contraction *of vertices v_1 and v_2 of a graph G is obtained by replacing v_1 and v_2 by a single vertex v, which is adjacent to all the former neighbours of v_1 and v_2):*
   - *Two vertices labelled* Input/Output *are contracted as one vertex with* Input/Output *label;*

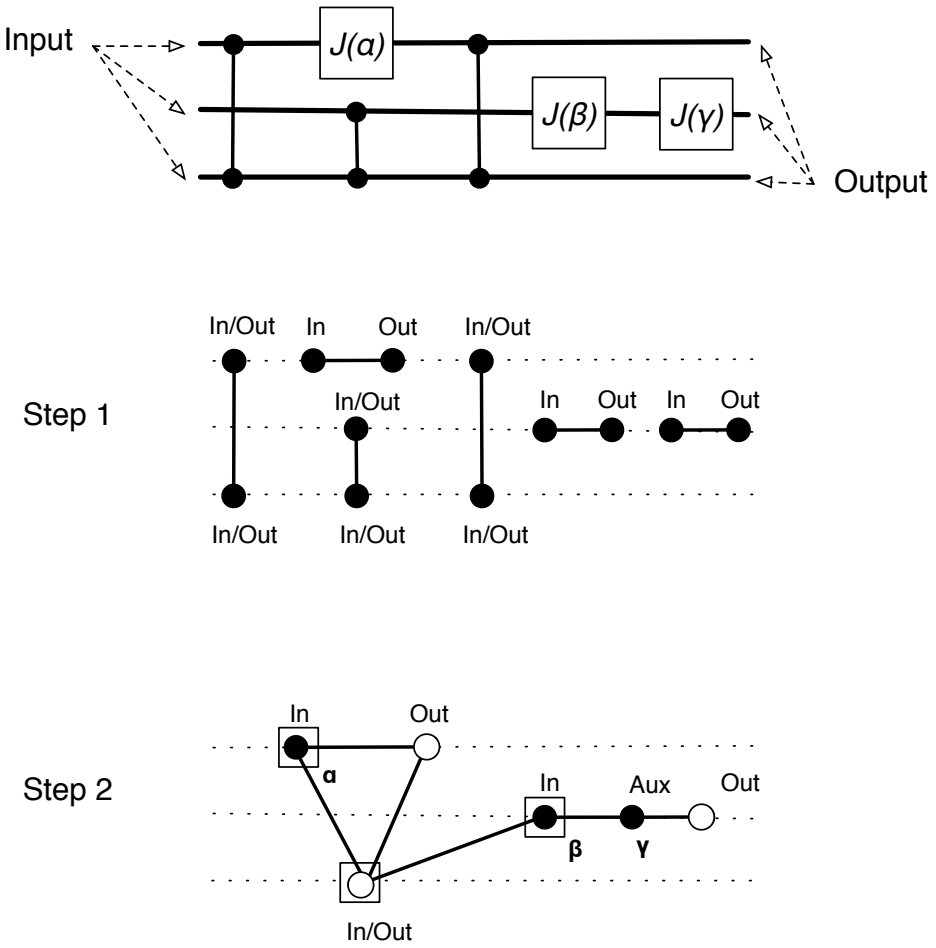


Fig. 1.9. A quantum circuit with $\wedge Z$ and $J(\alpha)$ gates, together with the two-step construction of the corresponding entanglement graph. In the final step, an input qubit is represented by a boxed vertex and an output qubit with a white vertex. The black vertices will be measured with angles α, β and γ, as shown in the figure.

- *A vertex labelled* Input/Output *and a vertex labelled* Input *are contracted as one vertex with* Input *label;*
- *A vertex labelled* Output *and a vertex labelled* Input/Output *are contracted as one vertex with* Output *label;*
- *Two vertices labelled* Output *and* Input *are contracted as one vertex with* auxiliary *label.*

It is easy to verify the following proposition that justifies the above construction.

Proposition 42 *The graph G_C obtained from Definition 41 is the entanglement graph for the measurement pattern that is obtained from Definition 40. Furthermore, input-output paths of vertices sitting on the same wire define the flow of G_C.*

Proof. Standardization does not change the underlying entanglement graph, hence it follows that G_C is indeed the entanglement graph for the measurement pattern. From de Beaudrap (2008), for the case that $|I| = |O|$, a collection of vertex-disjoint $I - O$ paths in G_C define the successor function f in its flow. Therefore, input-output paths of vertices sitting on the same wire define the flow of G_C. □

In order to obtain a full pattern corresponding to the circuit C, one needs to add measurement commands with angles being the same angles of the $J(\alpha)$ gates. These angles are assigned to the qubits labelled *Input* in Step (1) of the construction of Definition 41. The dependency structure is the one obtained from the flow theorem.

Fig. 1.10. The geometry of the teleportation pattern given in Equation (1.35) with one input, one auxiliary and one output qubit.

Proposition 43 *Let C be a quantum circuit on n logical qubits with only $\wedge Z$ and J gates. Let G_2 be the number of J gates and $D(n)$ the circuit depth. The corresponding pattern $\mathcal{P}$ given by Definition 40 has $n+G_2$ qubits, G_2 measurement commands, n corrections commands, and depth smaller than or equal to $D(n)$.*

Proof. The proof is based on construction of Definition 41, which is obtained from replacing the following patterns

$$\begin{array}{lcl} \mathcal{J}(\alpha) & := & X_2^{s_1} M_1^{-\alpha} E_{12} \\ \wedge\mathcal{Z} & := & E_{12} \end{array}$$

for J and $\wedge Z$ gates and then performing the standardization procedure. It is clear from the construction that we start with n qubits corresponding to each wire, then any $\wedge Z$ connects the existing qubits (wires) and hence will not add to the total number of qubits. On the other hand any J gate extends the wire by adding a new qubit. This leads to the total number of $n + G_2$ qubits for the pattern. There are G_2 measurement commands since all but n qubits are measured. Since C has depth $D(n)$, any influencing walk in $\mathcal{P}$ has at most $D(n)$ flow edges. Hence the theorem is obtained from Proposition 38 after performing signal shifting on the corresponding pattern. □

Alternatively, for a given circuit, one can use another construction to obtain a corresponding pattern with cluster geometry, hence to achieve constant depth for the entanglement graph preparation stage. Naturally, the price is to have more qubits. First note that the following pattern implements teleportation from input qubit i to output qubit k that is simply the identity map (see Figure 1.10):

$$X_k^{s_j} Z_k^{s_i} M_j^0 M_i^0 E_{jk} E_{ij} \tag{1.35}$$

Now, if before Step (2) of the construction of Definition 41, we insert the teleportation pattern between any two consecutive $\wedge Z$ acting on a common wire, then the degree of each vertex remains less than 4 as desired. We will refer to this graph as the *cluster graph*, GC_C. In order to compute the number of qubits for the pattern obtained from this new construction, consider the positions in the circuit where two $\wedge Z$ appear after each other. These are the places where we need to apply the above teleportation pattern to keep the degree less than 4. With this construction, the depth of the pattern does not increase by more than a multiplicative constant. Therefore we have:

Lemma 44 *Let C be a quantum circuit on n qubits with only $\wedge Z$ and J gates. Let G_2 be the number of J gates, s the size of C and m the number of positions in C where two $\wedge Z$ appear after each other. Then the pattern $\mathcal{P}$ with the cluster graph construction (obtained as in Proposition 42 with the addition of the teleportation pattern above) has $n + G_2 + m \in O(n + s)$ qubits and depth in $O(D(n))$.*

In what follows, we always assume the cluster geometry for patterns corresponding to a circuit and hence the preparation depth is 4

1.12.2 From patterns to circuits

The construction of Definition 41 can be also used in reverse order to transfer a pattern with flow to a corresponding circuit, where all the auxiliary qubits will be removed and hence by doing so the quantum depth might increase. However, we now show how to obtain another transformation from patterns to circuits where one keeps all the auxiliary qubits. This construction is simply based on the well-known method of coherently implementing a measurement. Recall that a controlled-unitary operator where the

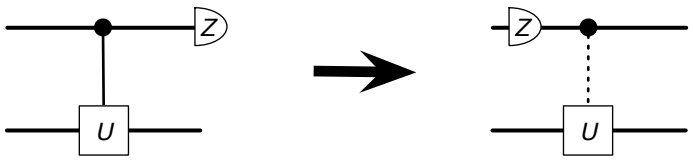


Fig. 1.11. A classically controlled implementation of a controlled-unitary gate. The computational basis measurement operator is represented by the half-circle box with Z label. After pushing the measurement to the beginning of the wire, the unitary U is only classically dependent (doted line) on the first wire.

control qubit is measured in the computational basis $\{|0\rangle, |1\rangle\}$ can be written as a classical controlled unitary by pushing the measurement before the controlled-unitary operator Griffiths and Niu (1996), see Figure 1.11.

Given a pattern in the standard form, we use the above scheme in the reverse order to convert the classically dependent measurements and corrections, and then push all the independent measurements to the end of the pattern. However since the scheme works only for the computational basis measurement, we have to first simplify all the arbitrary measurements M^α. Let $Z(\alpha)$ be the phase gate and H the Hadamard gate, and let M^Z be the computational basis measurement (*i.e.* Pauli Z measurement). Then we have

$$M^\alpha = M^{\{|+_\alpha\rangle, |-_\alpha\rangle\}} = M^{HZ(-\alpha)^\dagger\{|0\rangle, |1\rangle\}} = M^Z HZ(-\alpha)\,. \tag{1.36}$$

Additionally, we replace any classical X- and Z-dependencies of measurements and any dependent corrections with a sequence of $\wedge X$ and $\wedge Z$, which might create a quantum depth linear in the number of the dependencies, as shown in Figure 1.12. However to reduce this linear depth, we can use the following result on parallelizing a circuit with only controlled-Pauli gates to logarithmic depth:

Proposition 45 *(Moore and Nilsson (2002)) Circuits on n qubits consisting of controlled-Pauli gates and the Hadamard gate can be parallelized to a circuit with $O(\log n)$ depth and $O(n^2)$ auxiliary qubits.*

We can now formalize the above translation of patterns to circuits.

Definition 46 *Let $\mathcal{P}$ be a standard pattern with computational space (V, I, O), underlying geometry (G, I, O) (where G has a constant maximum degree) and command sequence (after signal shifting):*

$$\cdots C_j^{C_j} \cdots [M_i^{\alpha_i}]^{A_i} \cdots E_G$$

where A_i is the set of qubits that the measurement of qubit i depends on, and C_j is the set of qubits that the correction of qubit j depends on. Note that due to the signal shifting, we only have X dependencies. The corresponding coherent circuit *C with $|I|$ logical qubits and $|V \setminus I|$ auxiliary qubits, is constructed in the following steps (see also Figure 1.12):*

(i) *Apply individual Hadamard gates on all the auxiliary qubits.*
(ii) *Apply a sequence of $\wedge Z$ gates according to the edges of G.*
(iii) *Replace any dependent measurement $[M_i^{\alpha_i}]^{A_i}$ with $M_i^Z H_i Z_i(-\alpha) \wedge_{A_i,i} X$ where $\wedge_{A,i} X$ is a sequence of controlled-not with control qubits in A and target qubit i. Note that since the M^Z is independent and can be pushed to the end of the corresponding wire it can be discarded.*
(iv) *Replace any dependent correction $X_j^{C_j}$ with $\wedge_{C_i,i} X$ and $Z_j^{C_j}$ with $\wedge_{C_i,i} Z$.*
(v) *Replace the joint sequence of added $\wedge X$ and $\wedge Z$ in steps* (iii) *and* (iv) *with the parallel form obtained from Proposition 45.*

Lemma 47 *Let $\mathcal{P}$ be a standard pattern with computational space (V, I, O) and underlying geometry (G, I, O) (where G has a constant maximum degree). Let $t = |V \setminus O|$ be the number of measured qubits and let d be the quantum computation depth of $\mathcal{P}$. Then the corresponding coherent circuit C obtained from Definition 46 has $|I|$ logical qubits, $O(t^3)$ auxiliary qubits and depth in $O(d \log t)$.*

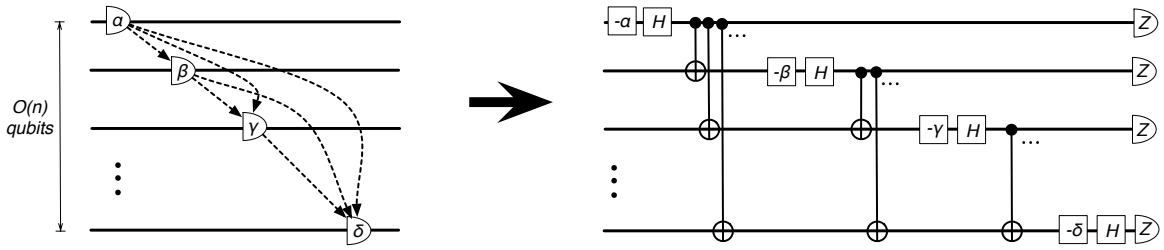


Fig. 1.12. Implementing coherently the sequence of dependent measurements in a pattern. An arbitrary measurement M^α is represented by a half circle labelled with its angle. The Hadamard and phase gates are shown with square boxes with the labels being H or the angle of the phase gate. The dotted arcs represent X-dependencies. Equation (1.36) is used to simplify the measurements. After replacing the X-dependencies by $\wedge X$ gates, we obtain a quantum depth linear in the number of dependencies.

Proof. We examine the cost at each step of the construction of Definition 46. Steps (i) and (ii) add a constant to the depth of C. At step (iii), each measurement has as most t dependencies, which, in step (v) translates to $O(\log t)$ depth with $O(t^2)$ auxiliary qubits. At step (iv), each output qubit has at most t dependencies, which again in step (v) translates to $O(\log t)$ depth with $O(t^2)$ auxiliary qubits. Since the depth of $\mathcal{P}$ is d, the total depth of C is in $O(d \log t)$, with $O(t^3)$ auxiliary qubits. □

Note that the logarithmic increase in the depth of C is due to the fact that the circuit model does not exploit any classical dependencies. Thus the classical computation of the measurement angles and corrections in $\mathcal{P}$ contributes to the quantum depth in C.

One can combine the forward and backward construction from circuit to patterns to obtain an automated rewriting system for the circuit which can decrease the depth by adding auxiliary qubits. The following theorem gives the tradeoff.

Theorem 19 *Let C be a quantum circuit on n qubits with only $\wedge Z$ and J gates. Suppose C has size s and depth D. Assume further that $\mathcal{P}$ is the corresponding pattern obtained from the forward translation as in Lemma 44 and that $\mathcal{P}$ has quantum depth D' (we know that $D' \leq D$). Then circuit C' constructed from $\mathcal{P}$ by Definition 46 has $O(s^3 + n)$ qubits, and depth in $O(D' \log s)$.*

Proof. The first step is to translate C to a pattern $\mathcal{P}$ using Lemma 44. The resulting pattern $\mathcal{P}$ has $O(s + n)$ qubits, and quantum depth in $O(D)$. Then we translate the pattern back to a circuit C' using Definition 46. By Lemma 47, the new circuit has $O(s^3)$ auxiliary qubits and depth in $O(D' \log s)$. □

At first glance it seems like applying Theorem 19 to a quantum circuit would not necessarily be beneficial, since the number of auxiliary qubits and the depth seem to increase. But note that we have given only upper bounds. However taking into account Pauli simplification and signal shifting can give a significant improvement Broadbent and Kashefi (2009).

1.13 Conclusion

In this chapter we have presented the mathematical model underlying measurement-based quantum computing and the algebra of pattern composition. More importantly, we have developed a rewrite system for patterns which preserves the semantics. We have shown further that our calculus defines a polynomial-time standardization algorithm transforming any pattern to a standard form where entanglement is done first, then measurements, then local corrections. We have inferred from this procedure that the denotational semantics of any pattern is a cptp-map and also proved that patterns with no dependencies, or using only Pauli measurements, may only implement unitaries in the Clifford group.

In addition we introduced some variations of the one-way and teleportation models and presented compositional back-and-forth embeddings of these models into the one-way model. This allows one to carry forward all the theory we have developed: semantics, rewrite rules, standardization, no-dependency

theorems and universality. This shows the generality of our formalism: we expect that any yet-to-be-discovered measurement-based computation frameworks can be treated in the same way.

Perhaps the most important aspect of standardization is the fact that now we can make patterns maximally parallel and distributed because all the entanglement operators, *i.e.* non-local operators, can be performed at the beginning of the computation. Then from the dependency structure that can be obtained from the standard form of a pattern the measurements can be organized to be as parallel as possible. This is the essence of the difference between measurement-based computation and the quantum circuit model or the quantum Turing machine.

We further presented a method for projection-based pattern synthesis by exploring the phase map decomposition of unitary maps into three successive operations P, Φ, and R. The first one represents the familiar preparation map and expands the computation space by introducing auxiliary qubits; the second one is diagonal in the computational basis and has only unit coefficients; and the last is a restriction map that contracts back the computational space into the chosen output space. It is important to emphasize that both R and P have a very simple structure, and hence the decomposition suggests that the whole quantum computing part of an algorithm is encoded in the phase map operator. Next, we addressed the fundamental question of how to transform the specification given by a projection-based pattern to an actual physical implementation as a measurement-based pattern.

Indeed, what makes the measurement-based quantum computing special is the fact that one can employ probabilistic measurement operators and yet perform a deterministic computation by imposing a causal dependent structure over the measurements sequence. On the other hand the MBQC highlights the role of entanglement as a resource for quantum computing. Hence a full understanding of the MBQC depends on gaining insight into the interplay of these two ingredients. To this end we demonstrated the notion of flow on the geometry of the entanglement graph and a full characterisation of stepwise uniformly deterministic computation in the one-way model independent of any reference to the circuit model together with efficient algorithm for finding the optimal depth flow.

One interesting consequence of patterns with generalised flow (but no flow) is that they can admit very compact implementations of a given unitaries. In particular, the generalised flow admits a great deal of flexibility in the causal structure of the corrections which can have have little in common with the structure of the associated quantum circuit. Further investigation of such features will be a line of future research.

Finally we demonstrated how forward and backward transformation between circuits and measurement patterns leads to an automated procedure of parallelization. A simple way of observing the advantages of the MBQC over the quantum circuit can be seen via the tradeoff between space and depth complexity as the transformation from a circuit to MBQC adds some auxiliary qubits and hence decreases the depth. On the other hand, one can also argue that the advantage is due to a clear separation of the types of depths that are involved in a computation: the preparation, quantum computation and classical depths. In other words, in the circuit model, all operations are done "*quantumly*" whereas in a pattern, some part of the computation can be performed via *classical processing*.

Acknowledgments

We would like to thank our collaborators and co-authors in the series of the papers that this chapter is based on: Niel de Beaudrap, Anne Broadbent, Daniel Browne, Mehdi Mhalla and Martin Roetteler.

Bibliography

Abramsky, S. and Coecke, B. (2004) A categorical semantics of quantum protocols. In Press, I. C. S., editor, *Proceedings of the 19th Annual IEEE Symposium on Logic in Computer Science (LiCS)*. Quant-ph/0402130.

Aliferis, P. and Leung, D. W. (2004) Computation by measurements: a unifying picture. *Physical Review A* **70**. Quant-ph/0404082.

Barendregt, H. P. (1984) *The Lambda Calculus, Its Syntax and Semantics*. Studies in Logic. North-Holland.

Benjamin, S., Eisert, J. and Stace, T. M. (2005) Optical generation of matter qubit graph states. *New Journal of Physics* **7**. Quant-ph/0506110.

Benjamin, S. C., Browne, D. E., Fitzsimons, J. and Morton, J. J. L. (2006) Brokered graph state quantum computing. *New Journal of Physics* **8**. Quant-ph/0509209.

Bennett, C., Brassard, G., Crepeau, C., Jozsa, R., Peres, A. and Wootters, W. (1993) Teleporting an unknown quantum state via dual classical and EPR channels. *Physical Review Letters* .

Bernstein, E. and Vazirani, U. (1997) Quantum complexity theory. *SIAM Journal of Computing* **5**(26).

Bravyi, S. and Kitaev, A. (2005) Universal quantum computation with ideal clifford gates and noisy ancillas. *Physical Review A* **71**. Quant-ph/0403025.

Broadbent, A. and Kashefi, E. (2009) On parallelizing quantum circuits. *Theoretical Computer Science* .

Browne, D., Kashefi, E., Mhalla, M. and Perdrix, S. (2007) Generalized flow and determinism in measurement-based quantum computation. *New Journal of Physics* **9**.

Browne, D. E. and Rudolph, T. (2005) Resource-efficient linear optical quantum computation. *Physical Review Letters* **95**. Quant-ph/0405157.

Chen, Q., Cheng, J., Wang, K. L. and Du, J. (2006) Efficient construction of two-dimensional cluster states with probabilistic quantum gates. *Physical Review A* **73**. Quant-ph/0507066.

Childs, A. M., Leung, D. W. and Nielsen, M. A. (2005) Unified derivations of measurement-based schemes for quantum computation. *Physical Review A* **71**. Quant-ph/0404132.

Clark, S. R., Alves, C. M. and Jaksch, D. (2005) Efficient generation of graph states for quantum computation. *New Journal of Physics* **7**. Quant-ph/0406150.

Danos, V. and Kashefi, E. (2005) Pauli measurements are universal. In Selinger (2005b).

Danos, V. and Kashefi, E. (2006) Determinism in the one-way model. *Physical Review A* .

Danos, V., Kashefi, E., Olivier, H. and Silva, M. (2006) A direct approach to fault-tolerance in measurement-based quantum computation via teleportation. *New Journal of Physics* Quant-ph/0611273.

Danos, V., Kashefi, E. and Panangaden, P. (2005) Parsimonious and robust realizations of unitary maps in the one-way model. *Physical Review A* **72**.

Danos, V., Kashefi, E. and Panangaden, P. (2007) The measurement calculus. *Journal of ACM* .

Dawson, C. M., Haselgrove, H. L. and Nielsen, M. A. (2006) Noise thresholds for optical cluster-state quantum computation. *Physical Review A* **73**. Quant-ph/0601066.

de Beaudrap, N. (2008) Finding flows in the one-way measurement model. *Physical Review A* **77**.

de Beaudrap, N., Danos, V. and Kashefi, E. (2006) Phase map decomposition for unitaries. Quant-ph/0603266.

de Beaudrap, N., Danos, V., Kashefi, E. and Roetteler, M. (2008) Quadratic form expansions for unitaries. In *Theory of Quantum Computation, Communication, and Cryptography Third Workshop, TQC 2008 Tokyo, Japan*, number 5106 in Lecture Notes in Computer Science.

den Nest, M. V., Dehaene, J. and Moor, B. D. (2004a) An efficient algorithm to recognize local clifford equivalence of graph states. *Physical Review A* **70**. Quant-ph/0405023.

den Nest, M. V., Dehaene, J. and Moor, B. D. (2004b) Graphical description of the action of local clifford transformations on graph states. *Physical Review A* **69**. Quant-ph/0308151.

Deutsch, D. (1985) Quantum theory, the Church-Turing principle and the universal quantum computer. In *Proceedings of the Royal Society of London*, volume A400.

Deutsch, D. (1989) Quantum computational networks. *Proc. Roy. Soc. Lond A* **425**.

Diestel, R. (2005) *Graph Theory*. Springer-Verlag.

Duncan, R. (2005) An abstract approach to entanglement. Mathematical Structures In Quantum Informatics, QDay II.

Dür, W., Aschauer, H. and Briegel, H. J. (2003) Multiparticle entanglement purification for graph state. *Physical Review Letters* **91**. Quant-ph/0303087.

Dürr, C. and Santha, M. (1996) A decision procedure for unitary linear quantum cellular automata. In *Proceedings of FOCS'96 – Symposium on Foundations of Computer Science*. LNCS. Quant-ph/9604007.

Gilbert, G., Hamrick, M. and Weinstein, Y. S. (2005) Efficient construction of photonic quantum computational clusters. Quant-ph/0512110.

Gottesman, D. and Chuang, I. L. (1999) Quantum teleportation is a universal computational primitive. *Nature* **402**.

Gottesoman, D. (1997) *Stabilizer codes and quantum error correction*. Ph.D. thesis, California Institute of Technology.

Griffiths, R. and Niu, C. (1996) Semiclassical Fourier transform for quantum computation. *Physical Review Letters* **76**:3228–3231.

Hartmann, L., Dür, W. and Briegel, H. J. (2005) Steady state entanglement in open and noisy quantum systems at high temperature. Quant-ph/0512219.

Hein, M., Eisert, J. and Briegel, H. J. (2004) Multi-party entanglement in graph states. *Physical Review A* **69**. Quant-ph/0307130.

Jorrand, P. and Perdrix, S. (2005) Unifying quantum computation with projective measurements only and one-way quantum computation. In Ozhigov, Y. I., editor, *Quantum Informatics 2004*, volume 5833 of *SPIE Proceedings*. Quant-ph/0404125.

Jozsa, R. (2005) An introduction to measurement based quantum computation. Quant-ph/0508124.

Kay, A., Pachos, J. K. and Adams, C. S. (2006) Graph-state preparation and quantum computation with global addressing of optical lattices. *Physical Review A* **73**.

Leung, D. W. (2004) Quantum computation by measurements. *International Journal of Quantum Information* **2**(1). Quant-ph/0310189.

Mhalla, M. and Perdrix, S. (2004) Complexity of graph state preparation. Quant-ph/0412071.

Mhalla, M. and Perdrix, S. (2008) Finding optimal flows efficiently. In *Proceeding of 35th International Colloquium on Automata, Languages and Programming*.
Moore, C. and Nilsson, M. (2002) Parallel quantum computation and quantum codes. *SIAM Journal on Computing* **31**.
Nielsen, M. A. (2003) Universal quantum computation using only projective measurement, quantum memory, and preparation of the 0 state. *Physical Review A* **308**.
Nielsen, M. A. (2004) Optical quantum computation using cluster states. *Physical Review Letters* **93**. Quant-ph/0402005.
Nielsen, M. A. and Chuang, I. L. (2000) *Quantum Computation and Quantum Information*. Cambridge University Press.
Perdrix, S. (2003) State transfer instead of teleportation in measurement-based quantum computation. *International Journal of Quantum Information* **3**(1). Quant-ph/0402204.
Perdrix, S. (2006) *Formal models of quantum computation: resources, abstract machines and measurement-based quantum computation*. Ph.D. thesis, Institut National Polytechnique de Grenoble, Laboratoire Leibniz.
Perdrix, S. (2007) Towards minimal resources of measurement-based quantum computation. *New Journal of Physics* **9**.
Perdrix, S. and Jorrand, P. (2004) Measurement-based quantum turing machines and their universality. Quant-ph/0404146.
Raussendorf, R., Anders, S. and Briegel, H. J. (2004) Fault-tolerant quantum computation using graph states. Communication to the Quantum Information and Quantum Control Conference, Fields Institute, Toronto. http://atlas-conferences.com/c/a/n/n/80.htm.
Raussendorf, R. and Briegel, H. J. (2001) A one-way quantum computer. *Physical Review Letters* **86**.
Raussendorf, R. and Briegel, H. J. (2002) Computational model underlying the one-way quantum computer. *Quantum Information & Computation* **2**. Quant-ph/0108067.
Raussendorf, R., Browne, D. E. and Briegel, H. J. (2003) Measurement-based quantum computation on cluster states. *Physical Review A* **68**.
Schlingemann, D. (2003) Cluster states, algorithms and graphs. Quant-ph/0305170.
Schumacher, B. and Werner, R. F. (2004) Reversible quantum cellular automata. Quant-ph/0405174.
Selinger, P. (2004) Towards a quantum programming language. *Mathematical Structures in Computer Science* **14**(4).
Selinger, P. (2005a) Dagger compact closed categories and completely positive maps. In Selinger (2005b).
Selinger, P., editor (2005b) *Proceedings of the 3nd International Workshop on Quantum Programming Languages*, Electronic Notes in Theoretical Computer Science.
Tame, M. S., Paternostro, M., Kim, M. S. and Vedral, V. (2004) Toward a more economical cluster state quantum computation. Quant-ph/0412156.
Tame, M. S., Paternostro, M., Kim, M. S. and Vedral, V. (2006) Natural three-qubit interactions in one-way quantum computing. *Physical Review A* **73**. Quant-ph/0507173.
Unruh, D. (2005) Quantum programs with classical output streams. In Selinger (2005b).
van Dam, W. (1996) *Quantum cellular automata*. Master's thesis, Computer Science Nijmegen.
Walther, P., k. J. Resch, Rudolph, T., Schenck, E., Weinfurter, H., Vedral, V., Aspelmeyer, M. and Zeilinger, A. (2005) Experimental one-way quantum computing. *Nature* **434**. Quant-ph/0503126.
Watrous, J. (1995) On one-dimensional quantum cellular automata. In *Proceedings of FOCS'95 – Symposium on Foundations of Computer Science*. LNCS.