Quantum linear network coding as one-way quantum computation

Niel de Beaudrap*
beaudrap@cwi.nl
CWI, Amsterdam

Martin Roetteler
martinro@microsoft.com
Microsoft Research

1 July 2014

Abstract

Network coding [1] is a technique to maximize communication rates within a network, in communication protocols for simultaneous multi-party transmission of information. Linear network codes are examples of such protocols in which the local computations performed at the nodes in the network are limited to linear transformations of their input data (represented as elements of a ring, such as the integers modulo 2). The quantum linear network coding protocols of Kobayashi *et al*. [17, 18] coherently simulate classical linear network codes, using supplemental classical communication. We demonstrate that these protocols correspond in a natural way to measurement-based quantum computations with graph states over qudits [21, 4, 8] having a structure directly related to the network.

1 Introduction

Network coding [1] is a technique to maximize the rate at which a set of *source nodes* can simultaneously transmit a set of independent messages to certain *target nodes* through a fixed network. For this purpose, it is sufficient to give each communication link enough bandwidth to accommodate multiple messages to be transmitted at once: however, less bandwidth may be required at each link if one allows nodes to distribute information about the messages across the network. A classic example is the *two-pair problem* on the "butterfly network" (illustrated in Figure 1): rather than halve the bandwidth between two messages at an apparent bottleneck in the network, the internal nodes may perform simple local computations on the messages, to allow the input data to be reconstructed at the targets. *Linear network coding* is the special case in which the protocol only requires each node to compute a linear transformation of its inputs to achieve this goal.

We consider *quantum network coding*, in which we perform similar tasks with quantum states transmitted through noiseless quantum channels. It is immediately apparent that some problems which can be sensibly posed for "classical" network coding are impossible in general for quantum network coding. For instance, while a classical network code allows for the each of the source nodes to each send a copy of their inputs to *both* targets in the butterfly network (see page 4), this is clearly not possible for quantum states due to the no-cloning theorem [24]. Other problems which do not require multiple copies of the input states to be re-created at the output (such as the two-pairs problem above) are still potentially unsolvable with fixed-capacity quantum channels alone, even when the corresponding classical problem is solvable [15, 19]. However, some of these problems become feasible for quantum states when the network nodes

*Supported by a Vidi grant from the Netherlands Organisation for Scientific Research (NWO) and by the European Commission project QALGO.

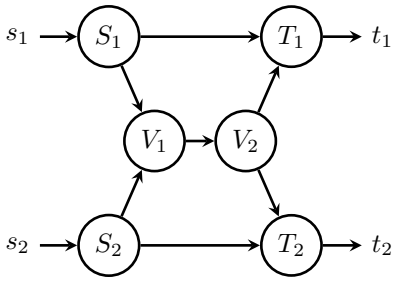


Figure 1: The *butterfly network*, with source nodes S_1 and S_2 and target nodes T_1 and T_2. The two-pair problem on this network is for S_1 to communicate their input to the target T_2, and simultaneously for S_2 to communicate their input to the target T_1, assuming that each edge can carry at most one message (represented *e.g.* by a single bit, 0 or 1). The classic solution is for S_1, S_2, and V_2 to duplicate their inputs, and for V_1, T_1, and T_2 to compute the parity of their inputs, in which case $(t_1, t_2) = (s_2, s_1)$.

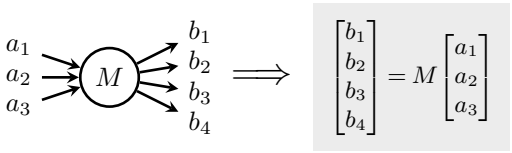
share prior entanglement [14], or if the capacities of the communication links scale as the logarithm of the number of target nodes [22].

Because classical information is easier to faithfully transmit and transform than quantum information, it is common to consider quantum protocols which also allow classical communication, and where fewer restrictions are imposed on the classical than the quantum communication (see Ref. [20]). In a setting where *no* restrictions are imposed on classical communication, Kobayashi *et al*. [17] describe a quantum protocol for the *k-pairs problem*: the problem in which each of k source nodes wish to communicate their input message to one of k distinct target nodes. Their protocol is in effect a coherent simulation of a classical linear network code. More generally, for any classical linear network code which performs some injective linear transformation $\mathbf{t} = M\mathbf{s}$ of the input data, Ref. [17] yields a corresponding quantum procedure to coherently simulate that network over for arbitrary superpositions of input data. We call such a protocol a (classically assisted) *quantum linear network code*. For the k-pairs problem, the protocols of Ref. [17] were subsequently extended in two different ways by Ref. [18]: to restrict the classical communication to the same network as the quantum communication (albeit with multiple rounds of communication, and sending a single message backwards as well as forwards along each communication link) and to accommodate non-linear protocols as well.

In this article we show that classically assisted quantum linear network codes in the style of Ref. [18] are in effect an instance of *one-way measurement based quantum computation* (MBQC) [21, 4, 8, 9]: a model of quantum computation in which one may entangle an arbitrary input state $|\psi\rangle$ with a graph state, which is then subjected to a sequence of measurements, leaving a final residual state which contains a transformed state $U|\psi\rangle$ for some unitary transformation[1] U. Furthermore, the graph state used as a resource is closely related structurally to the network used in the coding protocol. This demonstrates a link between MBQC and linear network coding, construed as distributed models of computation, and suggests novel ways of interpreting measurement-based procedures. At the same time, this suggests MBQC as a unifying framework in which to consider multi-party quantum networking protocols, including cryptographic applications formulated in the one-way model [3, 16] as well as standard security proofs of BB84 [23].

[1] In general, the transformation which is performed on an input state $|\psi\rangle$ is not necessarily a unitary transformation, but rather some completely positive trace preserving map Φ acting on $\rho_0 = |\psi\rangle\langle\psi|$. However, standard treatments of the one-way model describe how measurements on graph states may be used to simulate the transformations performed by unitary circuits, which by construction would transform the input state $|\psi\rangle$ unitarily.

Figure 2: An illustration of the transformation of messages performed by a single network node in a linear coding protocol.



2 Preliminaries

In this section, we present introductory remarks on classical linear network coding, and summarize the development of Refs. [17, 18]. We assume familiarity with standard models of quantum computation on qubits, as well as measurement-based quantum computation (see *e.g.* Refs. [21, 4, 8, 9] for introductory references). We introduce the notation and the definitions for the operators used over qudits of dimension d below.

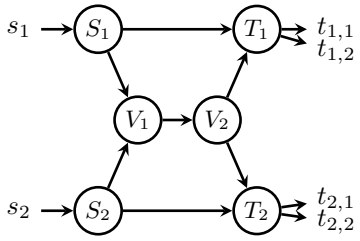
2.1 Classical network coding

We model a communications network by a directed graph of communications links, each of which can be used to transmit a single message from some message set M. In this article we suppose that M consists of a cyclic ring[2] $\mathbb{Z}_d = \mathbb{Z}/d\mathbb{Z}$. The messages are sent between co-operative agents (represented by nodes of the digraph) who may perform some non-trivial transformation of the data they receive from ingoing links. In the context of linear network codes, the transformations performed by each node are linear transformations, as represented in Figure 2. The result of this computation is then sent as output messages to other nodes. We restrict ourselves to directed acyclic networks, and assume that each node waits for all inputs to arrive before computing its outputs.

The canonical network coding problems involve distributing information from a collection of *source* nodes $\mathsf{S} = \{S_1, S_2, \ldots\}$ to a collection of *target* nodes $\mathsf{T} = \{T_1, T_2, \ldots\}$, such as the *multicast problem* (in which each source S_h must transmit their data to every one of the targets T_j), and the *k-pairs problem* (in which each source S_h tries to send their message to a single target $T_{\pi(h)}$, for some permutation $\pi \in \mathfrak{S}_k$ of the indices). The source nodes S_j each have some piece of information, usually represented as a single element $s_j \in \mathbb{Z}_d$ or vector $\mathbf{s}_j \in \mathbb{Z}_d^{n_j}$. To put the source and target nodes on an equal footing to the other network nodes, we suppose that the inputs s_j of the sources S_j are messages received from elsewhere (*e.g.* storage devices owned by the source nodes), and the outputs t_j to be computed by the targets T_j are also transmitted to somewhere, as depicted in Figure 1. A solution via linear network codes simply assigns linear transformations to each node, in such a way that the composite transformation performs the correct redistribution of input messages.

We regard linear network coding as a distributed model of computation, in which linear transformations are decomposed into block matrices, where each non-trivial block is represented by a single node. For *any* linear function f — of which the k-pairs and multicast problems are special cases — we consider which transformations the nodes may perform (if any) to compute f. Figure 3 presents the multicast problem on the butterfly network in this form, to which one solution is the following assignment

[2] In the setting where messages represent elements of a finite field $\mathrm{GF}(p^r)$ (see *e.g.* Ref. [13]), we may replace each communication link with r parallel communications links, representing elements of $\mathrm{GF}(p^r)$ as r-dimensional vectors over $\mathrm{GF}(p) \cong \mathbb{Z}_p$. In the case of linear network codes, this leads to no loss of generality, as every $\mathrm{GF}(p^r)$-linear transformation of messages is also a $\mathrm{GF}(p)$-linear transformation.



Multicast problem on the butterfly network:

$$\begin{bmatrix} t_{1,1} \\ t_{1,2} \\ t_{2,1} \\ t_{2,2} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix}$$

Decompose:

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} T_1 & 0 \\ 0 & T_2 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & V_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & V_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} S_1 & 0 \\ 0 & S_2 \end{bmatrix}$$

where $S_1, S_2 : \mathbb{Z}_d \to \mathbb{Z}_d^2$
$V_1 : \mathbb{Z}_d^2 \to \mathbb{Z}_d$
$V_2 : \mathbb{Z}_d \to \mathbb{Z}_d^2$
$T_1, T_2 : \mathbb{Z}_d^2 \to \mathbb{Z}_d^2$

Figure 3: The multicast problem on the butterfly network, formulated as a linear transformation over the ring $\mathbb{Z}_d$. A solution by linear network coding decomposes this transformation as a product of block matrices according to the network structure. A typical solution to this problem is presented in Eqn. (1).

of matrices to each node in the network:

$$S_1 = S_2 = V_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad V_1 = \begin{bmatrix} 1 & 1 \end{bmatrix}, \quad T_1 = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}, \quad T_2 = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}. \quad (1)$$

2.2 Classically assisted quantum network coding

We now outline the constructions of Ref. [17], and also of Ref. [18] in the special case of linear coding protocols over the ring $\mathbb{Z}_d$ of integers modulo d, for protocols using message qudits of dimension d.

Consider a node V performing some coding operation $\mathbf{y} = V\mathbf{x}$ for $\mathbf{x} \in \mathbb{Z}_d^\ell$ and $\mathbf{y} \in \mathbb{Z}_d^m$ in a classical coding network. We may simulate this node by initializing an output register $\mathbf{y} = \mathbf{0} \in \mathbb{Z}_d^m$, performing a bijective mapping $(\mathbf{x}, \mathbf{y}) \mapsto (\mathbf{x}, \mathbf{y} + V\mathbf{x})$ in the larger space $\mathbb{Z}_d^{\ell+m}$, and then discarding the input $\mathbf{x}$. The bijective mapping can be performed by elementary row transformations on $\mathbf{x}$, which in the quantum setting may be performed by controlled-X operations,

$$\Lambda X_{j,k} = \sum_{c=0}^{d-1} |c\rangle\langle c|_j \otimes X_k^c, \quad (2)$$

where $X\,|q\rangle = |q + 1 \bmod d\rangle$ is an analogue of the unitary Pauli operator σ_x on qubits. Consider a generic node V which accepts a collection of input qudits $a_1, \ldots, a_\ell$ as input and produces output qudits $b_1, \ldots, b_m$, coherently simulating the transformation $|\mathbf{x}\rangle_{a_1 \cdots a_\ell} \longmapsto |T\mathbf{x}\rangle_{b_1 \cdots b_m}$. In the construction of Ref. [17] for quantum linear codes, V simulates this transformation by preparing the qudits $b_1, \ldots, b_k$ in the $|0\rangle$ state, and performing the transformations

$$\Lambda X^{V_{j,k}} \Big(|x_k\rangle \otimes |0\rangle \Big) = |x_k\rangle \otimes |V_{j,k} x_k\rangle \quad (3)$$

on the qudits a_k and b_j, for every index $1 \leqslant j \leqslant \ell$ and $1 \leqslant k \leqslant m$ in any order. For standard basis states, the result is to transform $|\mathbf{x}\rangle\,|\mathbf{0}\rangle \mapsto |\mathbf{x}\rangle\,|V\mathbf{x}\rangle$. This characterizes a

linear transformation

$$\tilde{U}_V \;=\; \left(\prod_{j=1}^{m}\prod_{k=1}^{\ell} \Lambda X^{V_{j,k}}_{a_k,b_j}\right)\left(\mathbb{1}_{\mathbf{a}} \otimes |\mathbf{0}\rangle_{\mathbf{b}}\right), \tag{4}$$

which is a unitary embedding for any transformation V. (An example of such a circuit is illustrated in Figure 4.) If the qudits $a_1, \ldots, a_\ell$ where originally in standard basis states, we could simply discard them; but if they are initially not in standard basis states, they will become entangled with $b_1, \ldots, b_m$. To decouple them, we attempt to project each of the qudits a_j to the $|+\rangle$ state by measurement,

$$|+\rangle \;=\; \tfrac{1}{\sqrt{d}}\Big(|0\rangle + |1\rangle + \cdots + |d-1\rangle\Big). \tag{5}$$

Successfully doing so on a generic input state $|\psi\rangle = \sum_{\mathbf{x}} u_{\mathbf{x}} |\mathbf{x}\rangle$ would lead to the sequence of transformations

$$\begin{aligned} |\psi\rangle \;\longmapsto\; \sum_{\mathbf{x}} u_{\mathbf{x}} |\mathbf{x}\rangle_{\mathbf{a}} |\mathbf{0}\rangle_{\mathbf{b}} \;&\longmapsto\; \sum_{\mathbf{x}} u_{\mathbf{x}} |\mathbf{x}\rangle_{\mathbf{a}} |V\mathbf{x}\rangle_{\mathbf{b}} \\ &\longmapsto\; \frac{1}{\sqrt{d^\ell}} \left(\bigotimes_{k=1}^{\ell} |+\rangle_{a_k}\right) \otimes \sum_{\mathbf{x}} u_{\mathbf{x}} |V\mathbf{x}\rangle_{\mathbf{b}}\,. \end{aligned} \tag{6}$$

This mapping is of course non-unitary: projection onto $|+\rangle$ must be performed as part of a measurement onto some basis. Ref. [17] considers a measurement of the qudits a_j in the Fourier basis,

$$|\omega_r\rangle = \frac{1}{\sqrt{d}} \sum_{x=0}^{d-1} \mathrm{e}^{2\pi i x r/d} |x\rangle = F|r\rangle, \quad \text{where } F = \frac{1}{\sqrt{d}} \sum_{x,r=0}^{d-1} \mathrm{e}^{2\pi i k x/d} |x\rangle\langle r|\,. \tag{7}$$

The operator F is the *quantum Fourier transform over* $\mathbb{Z}_d$. We may attempt to simulate projection of each qudit a_j onto $|+\rangle$ by Fourier basis measurements, where a result of $|\omega_0\rangle$ is a success, as $|\omega_0\rangle = |+\rangle$. If we obtain results $|\omega_{r_j}\rangle$ for $r_j \neq 0$ instead of $|+\rangle$, the post-measurement state is

$$\left(\bigotimes_{k=1}^{\ell} |\omega_r\rangle_{a_k}\right) \otimes \sum_{\mathbf{x}} u_{\mathbf{x}} \mathrm{e}^{-2\pi i(\mathbf{r}\cdot\mathbf{x})/d} |V\mathbf{x}\rangle_{\mathbf{b}} \tag{8}$$

up to normalization. If V is injective, the relative phase $\mathrm{e}^{-2\pi i(\mathbf{r}\cdot\mathbf{x})/d}$ can be undone by a suitable application of Z operations on the qudits $b_1, \ldots, b_m$, where Z is the unitary generalization of σ_z:

$$Z \;=\; \sum_{q=0}^{d-1} \mathrm{e}^{2\pi i q/d} |q\rangle\langle q|\,. \tag{9}$$

If V is not injective, then only certain vectors $\mathbf{r}$ of measurement outcomes can be immediately corrected, resulting in a non-unitary CP map. However, regardless of whether some nodes in coding network perform non-invertible operations, the relative phases which accumulate on the entire state are linear functions. Then if the transformation performed by the whole network is injective, the phases which have accumulated due to the measurements can be undone if the target nodes have sufficient information about the measurement outcomes.

The protocol of Ref. [17] solves the k-pairs problem: thus the transformation it performs is indeed injective. Each node simply transmits their measurement outcomes to each target node, which performs a suitable combination of Z operations to correct the relative phases. Ref. [18] presents an alternative protocol in which the measurements are deferred until after all quantum messages have been sent, and in which the internal nodes of the network do the majority of the phase corrections, as follows. Consider a node which attempts to coherently simulate a transformation $L : \mathbb{Z}_d^\ell \to \mathbb{Z}_d^m$ in the middle of a coding network which attempts to coherently simulate a transformation $M : \mathbb{Z}_d^{\mathscr{S}} \to \mathbb{Z}_d^{\mathscr{T}}$ on an input state $|\psi\rangle = \sum_{\mathbf{x}} u_{\mathbf{x}} |\mathbf{x}\rangle$. Suppose that we perform the simulation procedure above, but omitting the Fourier basis measurements. For some linear maps H and K, the state after the final quantum messages is in general an entangled state of the form[3]

$$|\Psi\rangle = \sum_{\mathbf{x}} u_{\mathbf{x}} |\mathbf{x}\rangle_{\mathsf{S}} \otimes |M\mathbf{x}\rangle_{\mathsf{T}} \otimes \Big(|K\mathbf{x}\rangle_{a_1,\ldots,a_\ell} \otimes |LK\mathbf{x}\rangle_{b_1,\ldots,b_m} \Big) \otimes |H\mathbf{x}\rangle_{\text{rest}} \,, \quad (10)$$

where the factors in parentheses are the input and output qudits to the node L. If the qudits $b_1, \ldots, b_m$ are measured in the Fourier basis by the nodes to which they are sent, they yield some outcomes $r_1, \ldots, r_m$, and the remaining qudits are transformed to

$$|\Psi'\rangle = \sum_{\mathbf{x}} u_{\mathbf{x}} |\mathbf{x}\rangle_{\mathsf{S}} \otimes |M\mathbf{x}\rangle_{\mathsf{T}} \otimes \Big(\mathrm{e}^{-2\pi i(\mathbf{r}\cdot LK\mathbf{x})/d} |K\mathbf{x}\rangle_{a_1,\ldots,a_\ell} \Big) \otimes |H\mathbf{x}\rangle_{\text{rest}} \,, \quad (11)$$

where $\mathbf{r}$ is the vector of the outcomes. Let $\boldsymbol{\tau} = L^\top \mathbf{r}$: we have $\boldsymbol{\tau} \cdot K\mathbf{x} = \mathbf{r} \cdot LK\mathbf{x}$ by construction. If the nodes which perform these measurements send the outcomes to the node L, then L can undo the phases induced by measurement of the qudits b_k by performing the operation $Z^{\boldsymbol{\tau}} := Z_{a_1}^{\tau_1} Z_{a_2}^{\tau_2} \cdots Z_{a_\ell}^{\tau_\ell}$, which performs the mapping

$$\begin{aligned} Z_{a_1}^{\tau_1} Z_{a_2}^{\tau_2} \cdots Z_{a_\ell}^{\tau_\ell} \Big| \big(K\mathbf{x}\big)_1 \big(K\mathbf{x}\big)_2 \cdots \big(K\mathbf{x}\big)_\ell \Big\rangle & \\ = \exp\Big(\tfrac{2\pi i}{d} \big[\tau_1 (K\mathbf{x})_1 + \cdots + \tau_\ell (K\mathbf{x})_\ell \big] \Big) |K\mathbf{x}\rangle & \\ = \mathrm{e}^{2\pi i(\boldsymbol{\tau}\cdot K\mathbf{x})/d} |K\mathbf{x}\rangle \,. & \quad (12) \end{aligned}$$

Performing these corrections on $|\Psi'\rangle$ then yields the state

$$|\Psi''\rangle = \sum_{\mathbf{x}} u_{\mathbf{x}} |\mathbf{x}\rangle_{\mathsf{S}} \otimes |M\mathbf{x}\rangle_{\mathsf{T}} \otimes |K\mathbf{x}\rangle_{a_1,\ldots,a_\ell} \otimes |H\mathbf{x}\rangle_{\text{rest}} \,, \quad (13)$$

which has fewer unmeasured qudits than $|\Psi\rangle$, and no relative phases. This simulates projecting the qudits $b_1, \ldots, b_m$ to the $|+\rangle$ state. By induction, if each node aside from the source nodes (but including the target nodes) measures their input qudits in the Fourier basis, and communicates the outcomes backwards along their incoming links to the nodes which provided those qudits, those nodes can correct for the effect of the measurements. Eventually one obtains the state

$$\big|\Psi^{(n)}\big\rangle = \sum_{\mathbf{x}} u_{\mathbf{x}} |\mathbf{x}\rangle_{\mathsf{S}} \otimes |M\mathbf{x}\rangle_{\mathsf{T}}, \quad (14)$$

[3]The final tensor factor is on the remaining nodes entangled with the sources, whose components in the standard basis are again some linear transformations of the standard basis on the source nodes' inputs; by induction on the depth of the coding network, one may show that H and K are indeed linear transformations.

which is an entangled state of the (collective) inputs to the source nodes and the outputs of the target nodes. If the source nodes measure their qudits in the Fourier basis, it suffices for them to communicate the outcomes to target nodes in such a way that the outcomes can be corrected.

For arbitrary linear transformations M, direct communication among target nodes or between the source and the target nodes may be required to undo the relative phases induced by measurement. If the source nodes measure their qudits and collectively obtain a vector $\mathbf{s}$ of outcomes, the resulting state on the remaining qudits is

$$\left|\Psi^{(n+1)}\right\rangle = \sum_{\mathbf{x}} u_{\mathbf{x}} \mathrm{e}^{-2\pi i(\mathbf{s}\cdot\mathbf{x})/d} \left|M\mathbf{x}\right\rangle_{\mathsf{T}}. \tag{15}$$

If M has a left-inverse A, and we let $B = A^{\top}$, it suffices for the sources to somehow communicate $\sigma_j := \sum_k B_{jk} s_k$ to the target node T which is responsible for producing the message t_j. This would allow T to perform a Z^{σ_j} correction and undo the relative phase on the j^{th} output qudit. Specifically, if the sources collectively communicate $\boldsymbol{\sigma} = B\mathbf{s}$ to the targets, who collectively perform the phase operations $Z^{\boldsymbol{\sigma}} = Z_{t_1}^{\sigma_1} Z_{t_2}^{\sigma_2} \cdots$ on the target qudits, the resulting state is

$$\begin{aligned}\left|\Psi^{(n+2)}\right\rangle \;=\; \sum_{\mathbf{x}} u_{\mathbf{x}} \mathrm{e}^{2\pi i \left[\boldsymbol{\sigma}\cdot(M\mathbf{x}) - \mathbf{s}\cdot\mathbf{x}\right]/d} \left|M\mathbf{x}\right\rangle_{\mathsf{T}} \;&=\; \sum_{\mathbf{x}} u_{\mathbf{x}} \mathrm{e}^{2\pi i [\mathbf{s}^{\top}(AM-\mathbb{1})\mathbf{x}]/d} \left|M\mathbf{x}\right\rangle_{\mathsf{T}} \\ &= \sum_{\mathbf{x}} u_{\mathbf{x}} \left|M\mathbf{x}\right\rangle_{\mathsf{T}}; \end{aligned} \tag{16}$$

There are special cases where the amount of communication required outside of the network can be bounded. In particular, for the k-pairs problem where M is a permutation matrix (so that $(M^{-1})^{\top} = M$), it suffices to perform the classical linear coding protocol on the vector $\mathbf{s}$ to transmit $\boldsymbol{\sigma} = M\mathbf{s}$ to the target nodes. In this case, all classical communications may be restricted to the same network as the quantum communications — albeit using each communication link once in reverse, for the measurements of the qudits involved in the intermediate messages. More generally, if M is injective and there is a block-diagonal matrix B (where the blocks act on collections of messages held by individual target nodes) such that $M^{\top}BM = \mathbb{1}$, the sources may communicate $M\mathbf{s}$ to the targets, allowing the target nodes to compute $\boldsymbol{\sigma} = B^{\top}M\mathbf{s}$ and use this to govern phase corrections.

3 Classically assisted quantum linear coding is one-way MBQC

We now show how any coherent linear coding protocol, as described in Section 2.2, is in essence a measurement computation in the one-way model. The graph states of the MBQC procedures constructed in this way are easily derived from the coding network itself: allocate two entangled qudits at either end of each communications link in the network (one for the node on either side of the link), with further entangling operations between the qudits corresponding to the incoming links and the outgoing links. The corrections are the same as for the coherent coding network, albeit with some supplemental corrections arising from the way that the ΛX operations are simulated. If we follow the protocol of Ref. [17], the corrections are all deferred to the end of the procedure, as in standard treatments of measurement-based computation.

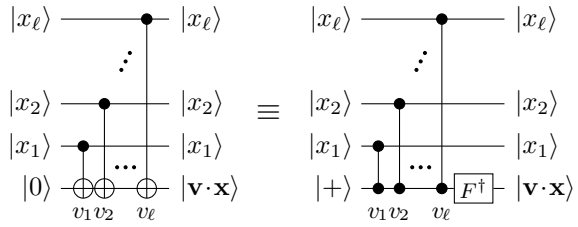


Figure 4: Equivalent ways to decompose a unitary transformation $\tilde{U}_V$ which prepares a single message qudit, for a single-row matrix $V = \mathbf{v}^\top$. The left-hand circuit represents the decomposition of Eqn. (4). Variables v_j below operations denote the power to which the circuit operation is raised. Multi-row coding transformations V may be simulated by several such circuits, acting on different target qudits.

Again, we assume familiarity with the measurement based model: see Refs. [21, 7, 4, 9] for references applicable to qubits (similar results and constructions apply over arbitrary qudits).

3.1 MBQC simulation of a single coding node

The main element of the correspondence between quantum linear network coding and MBQC is the observation that ΛX operations differ by only a Fourier transform from a controlled-phase operation,

$$\Lambda Z \;=\; (\mathbb{1}\otimes F)\Lambda X(\mathbb{1}\otimes F^\dagger) \;=\; \sum_{c=0}^{d-1} |c\rangle\langle c|\otimes Z^c, \tag{17}$$

which are the diagonal operations used to construct the entanglement structures in measurement-based computation. This means that the injective maps $\tilde{U}_V$ used to perform the coding at each node may be straightforwardly represented in terms of preparing the state $|+\rangle = F|0\rangle$ for each output qudit b_j to be sent, performing the entangling operation $\Lambda Z^{V_{j,k}}$ between each input qudit a_k and each output qudit b_j, and then acting on b_j with a Fourier transform, as represented in Figure 4.

Note that the inverse Fourier transform acting on the output-message qudit may be simulated by a Fourier basis measurement by introducing another auxiliary qudit, using a standard MBQC construction. Consider a qudit v in an arbitrary pure state $|\psi\rangle = \sum_{x=0}^{d-1} u_x |x\rangle$. We may introduce a qudit w prepared in the state $|+\rangle$, and entangle them using a $\Lambda Z^\dagger$ operation, obtaining the state

$$|\Psi\rangle_{vw} = \Lambda Z^\dagger_{vw} |\psi\rangle_v |+\rangle_w . \tag{18}$$

We then measure v in the Fourier basis, obtaining a state $|\omega_r\rangle$, and perform the operation X^{-r} on w. We may use the stabilizer formalism (see *e.g.* Ref. [10]) to succinctly verify how this sequence of transformations, considered as CP maps, transform X and Z: as these generate an operator basis for single-qudit states, this will suffice to show how it transforms $|\psi\rangle_v$ to $F^\dagger |\psi\rangle_w$. Specifically, we wish to see how the group of Pauli operators which *stabilize* the state (*i.e.*, at each point in time, those Pauli operators for which the state is a $+1$-eigenvector) transforms, for states on v and/or w. We use the following facts:

- We write $\omega = \exp(\frac{2\pi i}{d}) \in \mathbb{C}$ as a minor abuse of notation: it is easy to verify that $X|\omega_r\rangle = \omega^r|\omega_r\rangle$. In particular, $|+\rangle$ is the unique $+1$-eigenvector of X up to scalar factors.
- Measuring v in the Fourier basis is equivalent to measuring the eigenstates of X_v, obtaining some state $|\omega_r\rangle$: the post-measurement state is then stabilized by $\omega^{-r}X_v$, as well as by operators (but only those operators) which commute with X_v and stabilized the pre-measurement state.
- Conjugating X_v by $\Lambda Z^\dagger_{vw}$ yields $X_v Z^\dagger_w$, and similarly conjugating X_w by ΛZ_{vw} yields $Z^\dagger_v X_w$. As they are diagonal, conjugating Z_v or Z_w by ΛZ_{vw} has no effect. Conjugating by X_w^{-r} transforms $Z^\dagger_w$ to $\omega^{-r}Z^\dagger$, and leaves X_w unchanged.

We may then describe the sequence of transformations on $|\psi\rangle_v$ as follows: for any scalar $\phi \in \mathbb{C}$, the operator ϕX_v transforms as follows:

$$\begin{aligned}\langle \phi X_v\rangle &\xmapsto{\text{prep. } |+\rangle_w} \left\langle \phi X_v\,,\, X_w\right\rangle \\ &\xmapsto{\Lambda Z^\dagger_{vw}} \left\langle \phi X_v Z^\dagger_w\,,\, Z^\dagger_v X_w\right\rangle \\ &\xmapsto{X_v \text{ meas.}} \left\langle \phi X_v Z^\dagger_w\,,\, \omega^{-r}X_v\right\rangle \;=\; \left\langle \omega^{-r}X_v\right\rangle \otimes \left\langle \phi\omega^r Z^\dagger_w\right\rangle \\ &\xmapsto{X_w^{-r}\text{ corr.}} \left\langle \omega^{-r}X_v\right\rangle \otimes \left\langle \phi Z^\dagger_w\right\rangle\;, \end{aligned} \tag{19a}$$

so that these operations transform $\phi X_v \mapsto \phi Z^\dagger_w$; and similarly,

$$\begin{aligned}\langle \phi Z_v\rangle \xmapsto{\text{prep. } |+\rangle_w} \left\langle \phi Z_v\,,\, X_w\right\rangle \xmapsto{\Lambda Z^\dagger_{vw}} \left\langle \phi Z_v\,,\, Z^\dagger_v X_w\right\rangle \;=\; \left\langle \phi Z_v\,,\, \phi X_w\right\rangle \\ \xmapsto{X_v \text{ meas.}} \left\langle \omega^{-r}X_v\,,\, \phi X_w\right\rangle \\ \xmapsto{X_w^{-r}\text{ corr.}} \left\langle \omega^{-r}X_v\right\rangle \otimes \left\langle \phi X_w\right\rangle\;, \end{aligned} \tag{19b}$$

so that we obtain $\phi Z_v \mapsto \phi X_w$. Similarly, for any Weyl operator $W_{a,b}$ [10, Definition II], the operator $\phi W_{a,b}$ acting on v will be transformed to a Weyl operator $\phi W_{-a,b}$ on w; the calculation is straightforward. This implies (*c.f.* [10, Eqn. 17]) that aside from the teleportation from v to w, the effect is an inverse Fourier transform of the state.

Thus, we may simulate the coding procedure of a node V as described in Section 2.2 as follows. Provided a collection of incoming qudits $a_1, \ldots, a_\ell$, we may prepare output qudits $b_1, \ldots, b_m$ by:

1. preparing output message qudits $b_1, \ldots, b_m$ and auxiliary qudits $b'_1, \ldots, b'_m$ in the state $|+\rangle$;
2. entangling the qudits b_j and b'_j by a $\Lambda Z^\dagger$ operation, and performing $\Lambda Z^{V_{jk}}$ operations between each pair of qudits a_k and b'_j;
3. measuring each qudit b'_j in the Fourier basis, obtaining some outcome r_j, and performing an X^{-r_j} operation on the corresponding output qudit b_j.

This describes a MBQC procedure with inputs and outputs which we may illustrate by a *geometry* (in the terminology of Ref. [9, 7]) specifying the input and output qubits. Figure 5 presents geometries for the partial coding operation performed by $\tilde{U}_V$ as in Figure 4, and for the entire operation of a single coding node (including the eventual measurement of the input qubits): input qudits have arrows pointing inwards, and output qudits have arrows pointing outwards.

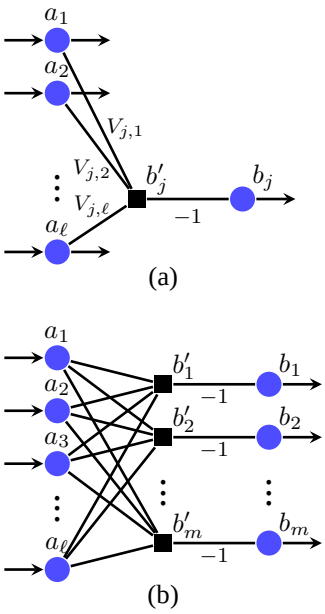


Figure 5: Geometries of MBQC procedures for a single node performing a transformation $V : \mathbb{Z}_d^\ell \to \mathbb{Z}_d^m$ of the standard basis. Incoming/outgoing message qudits are represented by blue circles; auxiliary qudits by black squares. **(a)** The geometry associated to coding a single message qudit, simulating the right-hand circuit of Figure 4. Edges are labeled by their "weights", *i.e.* the necessary power of ΛZ in the procedure. As the qudits a_k remain unmeasured, these are depicted as being outputs as well as inputs of this procedure. **(b)** The geometry associated to the entire operation of a coding node, including measurement of the incoming message qudits. Edge weights between the qudits a_k and α_j depend on the coding operation being simulated: if the coding operation being performed is sparse, many of these edge weights will be zero (corresponding to edges which should be omitted entirely). Only the qudits b_j form the output of this procedure.

3.2 MBQC geometries to simulate entire network coding protocols

In the diagrammatic convention of this article, composition of MBQC procedures may be represented by contracting the arrows between the outputs of earlier procedures and the inputs of later ones. For MBQC procedures to simulate the linear network codes, composing the geometries associated to each node yields a bipartite graph with a structure closely related to that of the coding network itself. Specifically, one associates a qudit for the output qudits of the coding network, as well as for each incoming and outgoing message qudit at each node (with qudits at the outgoing links being the "auxiliary" qudits described above), and connecting them by a bipartite graph corresponding to the non-zero coefficients V_{jk} of the coding node. The edges of the coding network are replaced by *undirected* edges with weights -1, corresponding to the entangling operations between the outgoing message qudits (which are either the inputs for some other node, or the outputs of the entire network). The directionality of the communication links are represented by the order of the measurement and correction operations, as well as the classical communication involved in the correction subroutine.

As an example, we illustrate this construction in Figure 6 for procedure for the two-pair problem performing a SWAP operation on two qudits (*e.g.* in which we use the coding operations $S_1 = S_2 = V_2 = [\,1 \;\; 1\,]^\top$ and $V_1 = T_1 = T_2 = [\,-1 \; -1\,]$).

As every measurement involved is performed in the Fourier basis (equivalently: the eigenbasis of the X operator), the only information which this graphical representation omits are the order in which the measurements occur, and the correction procedures, which we consider next.

3.3 Measurement and communication of outcomes

The corrections required to use X measurements to simulate projection onto $|+\rangle$ may be performed in two natural ways, corresponding to the protocols of Refs. [17] and [18] respectively.

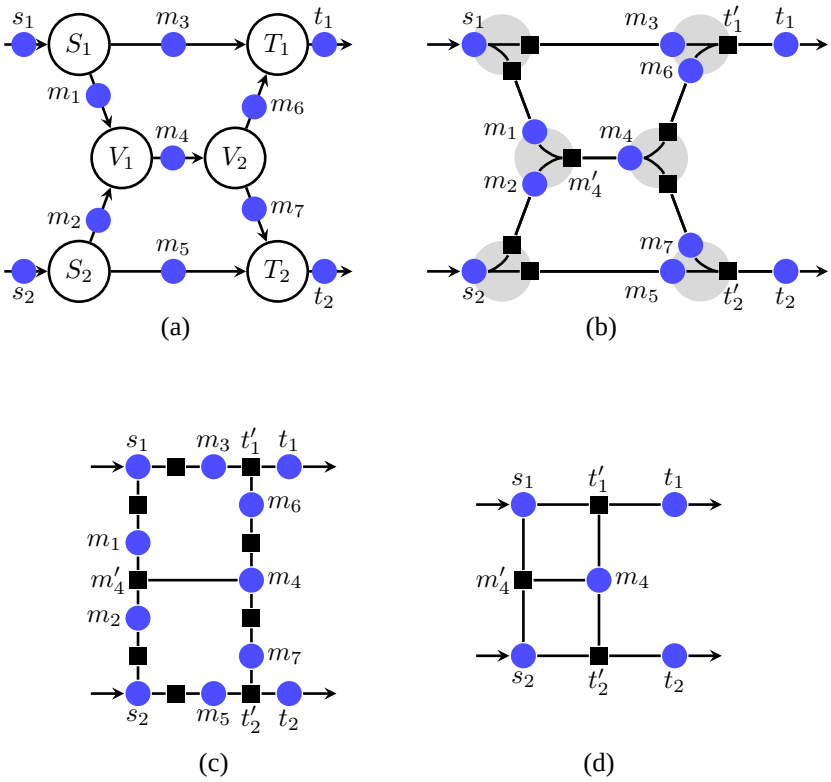


Figure 6: Construction of a MBQC geometry for a procedure simulating a coding protocol for the 2-pair problem on **(a)** the butterfly network, shown with message qudits for each communication link. **(b)** The graph obtained by substituting each coding node, with the geometry for the corresponding MBQC procedure. This is derived by adding vertices for "auxiliary" qudits (black squares) for each output message qudit, and associating each "auxiliary–output" pair to an outbound network link. Edges represent powers of ΛZ operations, which are used for single-qudit teleportation along the network links. The input and output message qudits of the linear code become the source and target subsystems of the MBQC procedure. **(c)** The same geometry, presented in grid formation. **(d)** The geometry of a MBQC procedure (*c.f.* Ref. [5, Figure 7]) for the SWAP operation.

3.3.1 Free classical communication

In a setting as in Ref. [17] where classical communication is free, all corrections may be deferred to the target nodes of the coding network, which prepare the output qudits. This is a natural approach for simulating the network code as a MBQC procedure: in measurement-based computation, it is conventional to simulate CP maps in such a way that the output qudits are the only qudits on which unitary correction operations are performed. As in Ref. [17], successful projection onto the $|+\rangle$ state (or a "0" outcome of a X measurement) is the ideal case; it then suffices to determine how the errors (or *byproduct operations* in the terminology of Ref. [21]) propagate to the output qudits, in order to correct them. We describe this in terms of communication directly to the targets, as well as some amount of communication within the coding network.

When simulating the coding procedure at each node using auxiliary qudits, measuring those auxiliary qudits introduces an additional source of error: if the correction is

not immediately performed on the outgoing message qudits, this induces additional phase errors. Commuting an $X_{b_j}^{-r}$ operation past an entangling operation $\Lambda Z_{b_j c_i'}^{U_{ij}}$, where c_i' is an auxiliary qudit for a subsequent node performing a coding operation U, yields an error operation $X_{b_j}^{-r} Z_{c_i'}^{-rU_{ij}}$. The operation $X_{b_j}^{-r}$ does not affect the outcome of the measurement on b_j, as the states $|\omega_r\rangle$ are eigenvectors of X. The Z error on c_i' induced by postponing the correction on b_j is significant, but we may account for this error by classical post-processing of the measurement result r' on c_i' itself. Let $\tilde{r} = rU_{ij}$ for the sake of brevity: because $XZ^{-\tilde{r}} \propto= \omega^r Z^{-\tilde{r}} X$, we may account for an uncorrected $Z^{-\tilde{r}}$ operation on c_i' by performing an X measurement, obtaining some outcome r_0', and then subtracting $\tilde{r}$ from that outcome to obtain an adjusted outcome $r' = r_0' - \tilde{r}$ for future corrections.

More generally, c_i' will accumulate uncorrected Z errors arising from the uncorrected X errors on each of the input messages on which it depends. If those input qubits b_j have errors X^{-r_j} associated with them, these collectively induce an error

$$Z^{-(r_1 U_{i1} + r_2 U_{i2} + \cdots)} = Z^{-\hat{\mathbf{e}}_i \cdot U \mathbf{r}} \tag{20}$$

on c_i'. We may simulate this correction after the Z measurement by subtracting $\tilde{r} = \hat{\mathbf{e}}_i \cdot U\mathbf{r}$ from the measurement outcome r_0', yielding $r' = r_0' - \hat{\mathbf{e}}_i \cdot U\mathbf{r}$. By propagating the results of the auxiliary qudit measurements forward through the coding network, subsequent coding nodes may locally adapt the measurement outcomes in order to simulate the correction of errors on their own auxiliary qudits, allowing the target nodes to perform the necessary X corrections on the output qudits of the network. Alternatively, all of the results may be transmitted directly to the target nodes, which can simulate this sequential adaptation of measurement outcomes themselves.

For a coding network performing an injective transformation $M : \mathbb{Z}_d^{\mathscr{S}} \to \mathbb{Z}_d^{\mathscr{T}}$, the phase errors induced by measurement of the message qudits may be corrected in the manner described in Ref. [17]. Without loss of generality, we may suppose that the agents at each network coding node prepare their auxiliary and message qudits, and all nodes except the target nodes communicate their outgoing messages to their recipients. Afterwards, they measure their auxiliary nodes in some order consistent with the topological ordering of the network, and similarly communicate the outcomes forward, allowing subsequent nodes to adjust their auxiliary measurement outcomes, and allowing target nodes to perform what X corrections are necessary on the output qudits. The remaining measurement operations and classical messages are identical to those of Ref. [17], in which it does not matter if nodes transmit outgoing message qudits before they measure incoming message qudits.

For the sake of completeness, we sketch an inductive approach to the Z correction protocol of the target nodes in this setting. Let A be a left-inverse of M, and consider an input state $|\psi\rangle$ to the coding network, expressed as

$$|\psi\rangle = \sum_{\mathbf{x} \in \mathbb{Z}_d^{\mathscr{S}}} u_{\mathbf{x}} |\mathbf{x}\rangle = \sum_{\mathbf{y} \in \mathrm{img}(M)} u_{A\mathbf{y}} |A\mathbf{y}\rangle . \tag{21}$$

The state obtained after performing the preparation and entanglement phases of the MBQC procedure, and after performing the auxiliary qudit measurements and X corrections on the output qudits, is exactly a state of the form in Eqn. (10), of the form

$$\begin{aligned} |\Psi\rangle &= \sum_{\mathbf{y} \in \mathrm{img}(M)} u_{A\mathbf{y}} |A\mathbf{y}\rangle_{\mathsf{S}} \otimes |MA\mathbf{y}\rangle_{\mathsf{T}} \otimes |HA\mathbf{y}\rangle_{\text{rest}} \\ &= \sum_{\mathbf{y} \in \mathrm{img}(M)} u_{A\mathbf{y}} |A\mathbf{y}\rangle_{\mathsf{S}} \otimes |\mathbf{y}\rangle_{\mathsf{T}} \otimes |HA\mathbf{y}\rangle_{\text{rest}} \end{aligned} \tag{22}$$

for some linear map H. (The latter equality holds because for any $\mathbf{y} = M\mathbf{x}$, we have $MA\mathbf{y} = MAM\mathbf{x} = \mathbf{y}$.) Indeed, the distinction between the input qudits S and the other non-target qudits is unimportant: we may subsume the linear map A on the standard basis of $\mathscr{S}$ and the map HA on the standard basis of the other qudits into a map

$$K \;=\; \left[\frac{A}{HA}\right] \tag{23}$$

where the upper rows correspond to indices in S, and the lower rows to the other non-output qudits. We may then write

$$|\Psi\rangle \;=\; \sum_{\mathbf{y}\in\mathrm{img}(M)} u_{A\mathbf{y}}\,|\mathbf{y}\rangle_{\mathsf{T}} \otimes |K\mathbf{y}\rangle_{\Omega\smallsetminus\mathsf{T}}\;. \tag{24}$$

We may isolate any non-output qudit $u \in \Omega \smallsetminus T$. Let $\Omega' = \Omega \smallsetminus \{u\}$, and consider another decomposition

$$K \;=\; \left[\frac{\boldsymbol{\kappa}_u^{\top}}{K'}\right] \tag{25}$$

where the upper row corresponds to the index for the qudit u and contains a row-vector $\boldsymbol{\kappa}_u^{\top}$, and K' corresponds to all of the other non-output qudits; we may then once more re-write

$$|\Psi\rangle \;=\; \sum_{\mathbf{y}\in\mathrm{img}(M)} u_{A\mathbf{y}}\,|\mathbf{y}\rangle_{\mathsf{T}}\;\;|\boldsymbol{\kappa}_u\cdot\mathbf{y}\rangle_u \otimes |K'\mathbf{y}\rangle_{\Omega'\smallsetminus\mathsf{T}}. \tag{26}$$

Measuring u in the Fourier basis and obtaining the outcome r, the resulting state on the remaining qudits is

$$|\Psi'\rangle \;=\; \sum_{\mathbf{y}\in\mathrm{img}(M)} u_{A\mathbf{y}}\,\omega^{-r(\boldsymbol{\kappa}_u\cdot\mathbf{y})}\,|\mathbf{y}\rangle_{\mathsf{T}}\;\;|K'\mathbf{y}\rangle_{\Omega'\smallsetminus\mathsf{T}}, \tag{27}$$

following Eqn. (11). If the outcome r is transmitted to the target nodes, and who know the value of $\boldsymbol{\kappa}_u$, they may simply compute $\boldsymbol{\sigma} := r\boldsymbol{\kappa}_u$ and collectively perform $Z^{\boldsymbol{\sigma}} = Z_{t_1}^{\sigma_1} Z_{t_2}^{\sigma_2}\cdots$ on the qudits of T, thereby obtaining

$$|\Psi''\rangle \;=\; \sum_{\mathbf{y}\in\mathrm{img}(M)} u_{A\mathbf{y}}\,|\mathbf{y}\rangle_{\mathsf{T}}\;\;|K'\mathbf{y}\rangle_{\Omega'\smallsetminus\mathsf{T}}, \tag{28}$$

which is again a state of the same form as in Eqn. (10), on one fewer qudits. By induction, we may measure each of the qudits of $\Omega\smallsetminus\mathsf{T}$ in any order (or simultaneously), and transmit them to the target nodes, which then make the appropriate Z corrections to obtain the state

$$\left|\Psi^{(n)}\right\rangle \;=\; \sum_{\mathbf{y}\in\mathrm{img}(M)} u_{A\mathbf{y}}\,|\mathbf{y}\rangle_{\mathsf{T}} \;=\; \sum_{\mathbf{x}\in\mathbb{Z}_d^{\mathscr{S}}} u_{\mathbf{x}}\,|M\mathbf{x}\rangle_{\mathsf{T}}\;. \tag{29}$$

In summary, provided free classical communication to the targets and within the coding network, all measurements may be performed simultaneously, with the results of the measurement of incoming messages being transmitted directly to the targets to perform Z corrections on the output qudits. Measurement results of the auxiliary qudits may be communicated along the coding network, and used to adapt the outcomes of subsequent measurements, culminating in measurement information useful to the target nodes to perform X corrections on the output qudits.

3.3.2 Constrained classical communication

In the setting of Ref. [18], we attempt to reduce the amount of classical communication which takes place outside of the network (but allowing messages to pass in either direction). To this end, we allow the source nodes and the intermediate nodes of the network to perform Z corrections. The way in which these corrections are performed follows from **(a)** the description of how X corrections may be simulated in the setting of "free" classical communication, as this already can be performed only with communication within the coding network; and **(b)** the phase correction procedure of Ref. [18] which was outlined in Section 2.2. These corrections may be performed as follows:

- All auxiliary qudits may be measured simultaneously, and their outcomes propagated forward through the network, as in the previous section. Alternatively, one may instead perform X correction operations for the auxiliary qudits at each node: this imposes an order on the measurement of the auxiliary qudits which is consistent with the topological order of the network, so that each node may use the measurement outcomes for preceding auxiliary qudits when correcting its own auxiliary qudits.

- The measurement of each node's incoming message qudits must be performed in an order opposite to the topological order of the coding network, in order to allow the node which sent each message qudit to perform the necessary corrections involving its own incoming message qudits.

From this, one may derive schedules for measuring each qudit in the network, and for communicating classical messages forward or backward through the network to allow the necessary X or Z corrections.

For the correction of phases induced by measurement of the input qubits of the source, following As in Section 2.2, whether the corrections arising from the measurement of the input qudits managed by the source nodes can be corrected without communicating outside of the network, may depend on the transformation which the network performs. For any linear transformation M for which $M^\top B M = \mathbb{1}$ for some block-diagonal B acting on blocks of qudits held by target nodes — *e.g.* for permutation matrices M — classical network coding of of the outcomes of measuring the inputs of the source nodes will suffice.

3.4 Overview of the MBQC construction

The above construction rests on the fact that the protocol of Ref. [17] is unaffected if the measurements are deferred until each node sends its messages. (The protocol of Ref. [18] in fact requires this modification.) The result of doing so causes these protocols to give rise to large distributed entangled states, on which local measurements are performed to simulate projection onto the $|+\rangle$ state. In this sense, these protocols are literally quantum computation by measurements; the modifications described in this Section — namely, replacement of ΛX operations by ΛZ operations, introduction and measurement of auxiliary qudits in order to make the previous modification possible, and communication of the results of measuring auxiliary qudits — are straightforward modifications which demonstrate that they are effectively computations in the one-way MBQC model of Refs. [21, 7].

The MBQC procedures which result from these transformations have comparable complexity to the original protocols of Refs. [17, 18], differing essentially only in the

various operations performed on the auxiliary qudits, as well as the communication and transformation of their measurement outcomes. For a coding network with k input messages, ℓ output messages, and m internal communication links, the total number of qudits involved in the MBQC procedure is easily verified to be $k + 2\ell + 2m$, following Section 3.2. The number of entangling operations involved for each node (disregarding exponents) is simply the same as the number of ΛX operations involved in simulating $\tilde{U}_V$, plus twice the out-degree (involved in entangling the auxiliary and outgoing message qudits for the node). Thus there are exactly $2(m + \ell)$ more entangling operations, in the form of ΛZ operations, in the MBQC protocol than there are ΛX operations in the original presentation of the protocols in Refs. [17, 18]. There are also exactly $2(m + \ell)$ additional classical messages sent in the MBQC protocol, either directly to the targets or entirely within the network, again as a result of measuring the auxiliary qudits.

4 Open questions

In this article, we have illustrated the way in which classically-assisted quantum linear network coding over $\mathbb{Z}_d$ as described by Kobayashi *et al.* [17, 18] is in effect an instance of measurement-based computation in the one-way model [21, 7], in particular using measurements only in the Fourier basis (the eigenbasis of the X cyclic shift operator on d-dimensional qudits). While not explicitly presented as an instance of MBQC, the differences between the protocols of Refs. [17, 18] and one-way measurement-based procedures are straightforward, and involve no substantial differences in *e.g.* the amount of classical communication required. We may ask to what extent these results (particularly the bounds on classical communication outside of the network) hold for classically assisted *non-linear* quantum codes as well.

While the MBQC model is sometimes described as a distributed model of computation, little emphasis has been placed on the communication cost of MBQC computation. A common presentation (*e.g.* as in Refs. [3, 2]) is that measurement results are recorded by an effectively delocalized classical control, which receives messages containing measurement outcomes from one or more agents which manage individual qudits, and which responds with instructions of how to perform subsequent measurements. Bounding the communication requirements of a MBQC procedure, to eliminate the need of a delocalised control center, may be necessary to realize the reduction in the computational depth of a MBQC procedure (one of the theoretical selling points of the MBQC model [21]).

As network coding subsumes constant-depth distributed computation, we may interpret these results as recommending measurement-based computation as a framework for analyzing multiparty communication protocols, as we have suggested in the introduction. We may also consider this as an alternative means of approaching the problem of assigning semantics to measurement-based computations, a problem of some interest in models of quantum computation [7, 9, 12, 6]. Specifically: rather than interpreting a measurement-based procedure as a quantum circuit with some potentially exotic features (such as closed time-like curves [6]), we may interpret pieces of measurement-based computations as coherently simulating transformations of the standard basis on several qudits at once. Such simple semantics is likely to prove useful to any programme to find novel ways of using measurement-based computation as a medium in which to develop algorithms (see Ref. [11]).

As a final open question, we ask whether a converse to our results hold, the form of a classical simulation algorithm for certain measurement-based computations by

linear network codes. This article shows that (a coherent quantum simulation of) a classical linear network code is in effect a measurement-based procedure which performs only X-eigenbasis measurements, on a graph state with similar structure to the coding network. This is a special case of an efficiently simulatable class of computations: the unitary transformations realized by MBQC procedures performing only Pauli-eigenbasis measurements are *Clifford group operations*,[4] which can be simulated *e.g.* on standard basis states by linear transformations on a cyclic ring [10]. This raises the question: is there a sense in which a MBQC procedure on a graph G, which implements unitary a transformation using only measurements in a Pauli eigenbasis (or only the X-eigenbasis) and Pauli corrections, can be "locally" simulated by a classical linear code — in such a way that the expectation value of any observable on a single given qudit can be evaluated from information available at a corresponding target node — on a network similar to G?

Acknowledgements.

This work was done in part while MR was with NEC Laboratories America, and NdB was at the University of Cambridge with support from the EC project QCS. NdB would like to thank Peter Høyer for helpful comments at the beginning of this research.

References

[1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung. Network information flow. *IEEE Transactions on Information Theory*, 46:1204–1216, 2000.

[2] J. Anders and D. E. Browne. Computational power of correlations. *Phys. Rev. Lett.*, 2009. arXiv:0805.1002.

[3] A. Broadbent, J. Fitzsimons, and E. Kashefi. Universal blind quantum computation. In *Proc. 50th IEEE FOCS*, pages 517–526, 2009. arXiv:0807.4154.

[4] D. E. Browne and H. J. Briegel. One-way quantum computation — a tutorial introduction. arXiv:quant-ph/0603226, 2006.

[5] D. E. Browne, E. Kashefi, M. Mhalla, and S. Perdrix. Generalized flow and determinism in measurement-based quantum computation. *New J. Phys.*, 9, 2007. arXiv:quant-ph/0702212.

[6] R. D. da Silva, E. F. Galvão, and E. Kashefi. Closed timelike curves in measurement-based quantum computation. *Phys. Rev A*, 83, 2011. arXiv:1003.4971.

[7] V. Danos, E. Kashefi, and P. Panangaden. Robust and parsimonious realisations of unitaries in the one-way model. *Phys. Rev. A.*, 72, 2006. arXiv:quant-ph/0411071.

[8] V. Danos, E. Kashefi, and P. Panangaden. The measurement calculus. *J. ACM*, 54, 2007. arXiv:0704.1263.

[9] N. de Beaudrap. Unitary-circuit semantics for measurement-based computations. pages 1–91, 2010. arXiv:0906.4261.

[4] This is well-known for qubits [4]; on qudits it follows from how stabilizer states are transformed under measurements, see Ref. [10].

[10] N. de Beaudrap. A linearized stabilizer formalism for systems of finite dimension. *Quant. Info. & Comp.*, pages 73–115, 2013. arXiv:1102.3354.

[11] N. de Beaudrap, V. Danos, E. Kashefi, and M. Roetteler. Quadratic form expansions for unitaries. In *Proc. TQC 2008*, pages 29–46, 2008. arXiv:0801.2461.

[12] R. Duncan. A graphical approach to measurement-based quantum computing. arXiv:1203.6242, 2012.

[13] M. Grassl, M. Roetteler, and T. Beth. Efficient quantum circuits for non-qubit quantum error-correcting codes. *Intl. J. Found. Comp. Sci.*, 14:757–775, 2003. arXiv:quant-ph/0211014.

[14] M. Hayashi. Prior entanglement between senders enables perfect quantum network coding with modification. *Phys. Rev. A*, 76, 2007. arXiv:0706.0197.

[15] M. Hayashi, K. Iwama, H. Nishimura, R. Raymond, and S. Yamashita. Quantum network coding. In *Proc. 24th annual STACS*, pages 610–621, 2007. arXiv:quant-ph/0601088.

[16] E. Kashefi, D. Markham, M. Mhalla, and S. Perdrix. Information flow in secret sharing protocols. *EPTCS*, 9:87–97, 2009. arXiv:0909.4479.

[17] H. Kobayashi, F. Le Gall, H. Nishimura, and M. Roetteler. General scheme for perfect quantum network coding with free classical communication. In *Proc. 36th ICALP*, pages 622–633, 2009. arXiv:0908.1457.

[18] H. Kobayashi, F. Le Gall, H. Nishimura, and M. Roetteler. Constructing quantum network coding schemes from classical nonlinear protocols. In *Proc. 2011 IEEE Intl. Symp. Info. Theory*, pages 109–113, 2011. arXiv:1012.4583.

[19] D. Leung, J. Oppenheim, and A. Winter. Quantum network communication — the butterfly and beyond. *IEEE Trans. Inf. Theory*, 56:3478–3490, 2010. arXiv:quant-ph/0608223.

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

[21] R. Raussendorf, D. E. Browne, and H. J. Briegel. Measurement-based quantum computation on cluster states. *Phys. Rev. A*, 68, 2003. arXiv:quant-ph/0301052.

[22] Y. Shi and E. Soljanin. On multicast in quantum network. In *Proc. 40th Annual Conf. Info. Sci. and Systems*, pages 871–876, 2006.

[23] P. W. Shor and J. Preskill. Simple proof of security of the BB84 quantum key distribution protocol. *Phys. Rev. Lett.*, 85:441–444, 2000. arXiv:quant-ph/0003004.

[24] W. K. Wootters and W. H. Zurek. A single quantum cannot be cloned. *Nature*, 299:802–803, 1982.