New Journal of Physics

CrossMark

Generalized cluster states based on finite groups

Courtney G Brell

Centre for Engineered Quantum Systems, School of Physics, The University of Sydney, Sydney, Australia
Institut für Theoretische Physik, Leibniz Universität Hannover, Appelstraße 2, D-30167 Hannover, Germany

E-mail: courtney.brell@gmail.com

Abstract

We define generalized cluster states based on finite group algebras in analogy to the generalization of the toric code to the Kitaev quantum double models. We do this by showing a general correspondence between systems with CSS structure and finite group algebras, and applying this to the cluster states to derive their generalization. We then investigate properties of these states including their projected entangled pair state representations, global symmetries, and relationship to the Kitaev quantum double models. We also discuss possible applications of these states.

1. Introduction

Cluster states (or graph states) [1, 2] are the prototypical resource for measurement-based quantum computation (MBQC) [3]. This is broad category of strategies for implementing fault-tolerant quantum information processing equivalent but in contrast to the quantum circuit model, adiabatic quantum computation, and topological quantum computation. MBQC proceeds by taking a suitable entangled many-particle resource state and performing computation by sequential single-particle measurements that may be chosen adaptively based on previous measurement results.

The cluster states have a particularly simple structure that allows for straightforward analysis of their properties. They also have many desirable features for a resource state: for example, they are the output of a finite-depth quantum circuit, and also the frustration-free ground states of a (gapped) commuting local Hamiltonian.

Apart from their usefulness for standard MBQC, the cluster states are also related to topologically ordered systems such as the toric code [4–8] and the colour codes [9, 10]. In fact, this relationship can be leveraged to define a protocol for MBQC that exploits the natural fault-tolerance properties of topological quantum computation schemes [11]. The cluster states can also be used to study general stabilizer states, as all stabilizer states are equivalent to cluster states under local Clifford operations [12]. Additionally, the cluster states have been used in studies of such diverse topics as the origin of quantum computational power [13] and contextuality [14].

Since the cluster states are such an important theoretical tool for studying these topics, it is of interest to ask if their properties can be generalized in any interesting ways. In this paper we take steps towards answering this question, in particular motivated by the relation between cluster states and topologically ordered systems. In so doing, we provide a general framework for similar generalization programmes.

The toric code is the simplest member of the family of topologically ordered Kitaev quantum double models [4]. These models are defined by a finite group G (of which the toric code corresponds to $\mathbb{Z}_2$). These models are of significant interest for condensed matter physics, where they are an important testbed for the phenomenology of topological order, as well as quantum information, where they can be used to implement topological quantum computation, for example through braiding of quasiparticles [15] or code deformation [16], or can be used as quantum memories [5]. Though extensions of the Kitaev quantum double models have also been proposed based on a Hopf algebra H [17], or twisted quantum doubles labelled by finite group G and three-cocycle ω [18], we will restrict our consideration to the finite group case here. Most of the important phenomenology of these generalizations can be captured by finite groups, and in particular the qualitative

distinction between abelian and non-abelian topological phases is manifested in Kitaev quantum double models for abelian and non-abelian groups respectively. Abelian phases such as the toric code cannot be used for quantum computation by braiding of quasi-particles, and are not known to be able to implement universal topological quantum computation via code-deformation (though universal quantum computation can still be achieved by using non-topological operations such as magic state distillation [19, 20]).

For these reasons, we define a family of generalized cluster states based on arbitrary finite groups G, where the standard qubit cluster state corresponds to the simplest group $\mathbb{Z}_2$. The previously known higher-qudit cluster states [21, 22] correspond to the cyclic groups $\mathbb{Z}_d$. In order to make this generalization process as clear as possible, we first make a general connection between general qubit CSS states and the group $\mathbb{Z}_2$. This allows an intuitive generalization of any such system, which may be more generally applicable.

We explore the properties of the generalized cluster states defined in this way, such as their global symmetries and projected entangled pair state (PEPS) representations, in analogy to the qubit case [23, 24]. We also show how the generalized cluster states retain a relation to the corresponding quantum double models, and discuss possible applications of the generalized cluster states making use of this relation.

The paper is organized as follows. In section 2 we outline the group structure of CSS states and the general method of generalization we will follow. This is then used in section 3 to define the generalized cluster states for arbitrary finite groups G. Section 4 is devoted to exploring some properties of these states. Finally, in section 5 we briefly consider applications for generalized cluster states including preparation of the Kitaev quantum double states and a generalization of the topological cluster state computation scheme, before discussing possible extensions of the kind of generalization scheme proposed here and providing some concluding remarks.

2. From CSS structure to finite groups

In this paper, we will generalize the familiar qubit cluster state to states based on finite group algebras. In order to do this, we will make use of a general prescription to translate from a system with CSS structure [25, 26] to one based on the group $\mathbb{Z}_2$.

Many interesting models defined on spin-½ systems have a CSS structure. While the term CSS has a specific technical meaning as a class of stabilizer codes, we will use it in its more colloquial sense to mean any system involving interactions that consist either of products of Pauli X operators or products of Pauli Z operators.

Systems with a CSS structure have a natural interpretation in terms of the group algebra of $\mathbb{Z}_2$. Recall that $\mathbb{Z}_2$ has two elements (labelled 1 and the identity 0), group multiplication is addition modulo 2 ($\oplus$), and there are two irreducible representations. The trivial and alternating irreps we label $+$ and $-$ respectively, with $\Gamma_+(0) = \Gamma_+(1) = \Gamma_-(0) = 1$ and $\Gamma_-(1) = -1$.

We can associate qubit states and operators with objects related to the group $\mathbb{Z}_2$ by considering the computational basis states of our qubit to be labelled by group elements. That is, we take $|0\rangle$ and $|1\rangle$ to be associated with the respective group elements 0 and 1. Following this, the Pauli X operator can be seen to act as group multiplication by the 1 element. For this reason we denote it by $X_1 \equiv X$. We can also see that $X_0 \equiv X^0 = I$ acts as group multiplication by the 0 element. Thus we consider X_g to act as group multiplication by an element $g \in \mathbb{Z}_2$. We can also interpret the CNOT gate in this context as a controlled group multiplication operation. This can easily be seen by CNOT $|g\rangle|h\rangle = |g\rangle|g \oplus h\rangle$.

The conjugate basis states $|+\rangle$ and $|-\rangle$ can be associated with the irreducible representations of $\mathbb{Z}_2$ by noticing that $|\pm\rangle \propto \Gamma_\pm(0)|0\rangle + \Gamma_\pm(1)|1\rangle$ (throughout this paper we will consider only unitary irreducible representations over $\mathbb{C}$ unless otherwise specified). We can similarly consider the powers of the Pauli Z operator to be associated with the representations of $\mathbb{Z}_2$ as $Z_+ \equiv Z^1$ and $Z_- \equiv Z^0$. Notice then that $Z_\pm |g\rangle = \Gamma_\pm(g)|g\rangle$.

The group $\mathbb{Z}_2$ has much structure that is absent for general groups. In particular, there is a natural isomorphism between group elements and irreps, taking 0 to $+$ and 1 to $-$. In this way we can interpret the Hadamard gate as implementing this isomorphism. Although the CPHASE gate has no fundamental interpretation in terms of objects from the group, it can be brought into this framework by noticing that CPHASE can be constructed from a circuit of Hadamard and CNOT gates.

This entire structure is summarized in the first two columns of table 1. It also gives us an avenue to generalize many of these concepts to arbitrary groups. In interpreting the structure of the qubit in terms of $\mathbb{Z}_2$, the most significant property of this group that does not still hold in the general case is the existence of a natural isomorphism from group elements to irreps. The fact that this is not available for a general group G is the reason that the CSS structure is important when generalizing to arbitrary finite groups, as we will show.

Now that the properties of the qubit have been related to properties of $\mathbb{Z}_2$, the generalization to an arbitrary finite group G is relatively straightforward. Two-dimensional qubits are replaced by $|G|$-dimensional qudits. The analogue of computational basis states for these qudits are labelled by group elements $g \in G$ (we call this the

Table 1. Summary of the correspondence between group algebras and states or operators on quantum systems.

$\mathbb{Z}_2$	Qubit	Qudit	G
Elements $\{0, 1\}$	$\lvert 0 \rangle, \lvert 1 \rangle$	$\lvert g \rangle$ $(d = \lvert G \rvert)$	Elements $g \in G$
Multiplication $\oplus g$	$X_g \lvert h \rangle = \lvert g \oplus h \rangle$	$X_g^{\leftarrow} \lvert h \rangle = \lvert gh \rangle$ $X_g^{\rightarrow} \lvert h \rangle = \lvert hg^{-1} \rangle$	Multiplication
Irreps $\{\Gamma_+, \Gamma_-\}$	$\lvert \pm \rangle = \frac{1}{\sqrt{2}}(\Gamma_\pm(0)\lvert 0 \rangle + \Gamma_\pm(1)\lvert 1 \rangle)$	$\lvert \Gamma^{ij} \rangle = \sqrt{\frac{d_\Gamma}{\lvert G \rvert}} \sum_g [\Gamma(g)]_{ij} \lvert g \rangle$	Irreps $\Gamma \in \mathrm{Rep}(G)$
Irrep action $\Gamma(g)$	$Z_\Gamma \lvert h \rangle = \Gamma(h)\lvert h \rangle$	$Z_{\Gamma^{ij}} \lvert h \rangle = [\Gamma(h)]_{ij} \lvert h \rangle$	Irrep action $[\Gamma(g)]_{ij}$
$\mathbb{Z}_2 \cong \mathrm{Rep}(\mathbb{Z}_2)$	Hadamard gate	—	—
Controlled Not	CNOT $\lvert g \rangle \lvert h \rangle = \lvert g \rangle \lvert g \oplus h \rangle$	CMULT$^{\leftarrow}\lvert g \rangle \lvert h \rangle = \lvert g \rangle \lvert gh \rangle$ CMULT$^{\rightarrow}\lvert g \rangle \lvert h \rangle = \lvert g \rangle \lvert hg^{-1} \rangle$	Controlled multiplication

group element basis). It is worthwhile noting that there is a preferred state corresponding to the identity group element e. Thus the $\lvert 0 \rangle$ state of the qubit corresponds in general to the $\lvert e \rangle$ state of the group element basis.

The Pauli X operator of the qubit generalizes to left and right group multiplication operators $X_g^{\leftarrow}$ and $X_g^{\rightarrow}$ acting as

$$X_g^{\leftarrow} \lvert h \rangle = \lvert gh \rangle \quad \text{and} \quad X_g^{\rightarrow} \lvert h \rangle = \lvert hg^{-1} \rangle. \tag{1}$$

Similarly, the CNOT gate generalizes straightforwardly to a controlled left or right group multiplication gate.

The conjugate basis of the qubit was interpreted as representation states, and in the general case we label these states by matrix elements of an unitary irreducible representation (or irrep) of G, and define

$$\lvert \Gamma^{ij} \rangle = \sqrt{\frac{d_\Gamma}{\lvert G \rvert}} \sum_{g \in G} [\Gamma(g)]_{ij} \lvert g \rangle, \tag{2}$$

where $[\Gamma(g)]_{ij}$ is given by the group element g at a matrix element (i, j) of a representation Γ and d_Γ is the dimension of Γ. We call this basis the representation basis as compared to the group element basis. Noting that $\sum_\Gamma d_\Gamma^2 = \lvert G \rvert$, orthonormality of the representation basis follows directly from the grand orthogonality theorem of group representations:

$$\sum_{g \in G} \left[\Gamma_\lambda(g)\right]_{ij}^* \left[\Gamma_\sigma(g)\right]_{i'j'} = \delta_{\lambda\sigma} \delta_{ii'} \delta_{jj'} \frac{\lvert G \rvert}{d_\lambda}. \tag{3}$$

The transformation from the representation basis to the group element basis can also be found as

$$\lvert g \rangle = \sum_{\Gamma^{ij}} \sqrt{\frac{d_\Gamma}{\lvert G \rvert}} [\Gamma(g)]_{ij} \lvert \Gamma^{ij} \rangle. \tag{4}$$

As in the group element basis, there is a preferred state in the representation basis. This corresponds to the trivial irrep I. The trivial irrep state is given by an equal superposition over all group element states, and reduces to the $\lvert + \rangle$ state in the case that $G = \mathbb{Z}_2$.

The generalizations of the Pauli Z operators are labelled by matrix elements of irreps of G. As in the $\mathbb{Z}_2$ case, they act on group element basis states by accumulating amplitudes corresponding to the relevant matrix element of the group element in the given representation. That is

$$Z_{\Gamma^{ij}} \lvert h \rangle = [\Gamma(h)]_{ij} \lvert h \rangle. \tag{5}$$

These generalizations are summarized in table 1.

The Hadamard gate and CPHASE gate of the qubit have no natural generalization to an arbitrary group in this way. Similarly, interactions that do not have CSS structure cannot be generalized for the following reason. If we consider an operator constructed from mixed X and Z operators on a set of qubits, to generalize this operator to the group G we must associate a group element to the X operator and a representation to the Z operator. There is no natural way to choose a representation corresponding to each group element, and so we cannot consistently generalize a mixed operator of this type. This is a direct consequence of the fact that the Hadamard operator has no analogue for arbitrary groups. Given that we can only consider systems with CSS structure in this framework, it is clear that the Hadamard and CPHASE gates cannot be generalized, as they take CSS systems to non-CSS systems.

One way to interpret table 1 is as a prescription to design quantum systems with algebraic properties inherited from an arbitrary finite group. In general, the structure of these algebraic properties will be closely related to the quantum double of the group under consideration, as is the case for example in the Kitaev quantum double models, which can be interpreted in this framework. One could also consider generalizing this correspondence from groups to more general objects such as Hopf algebras (as has been done for the Kitaev quantum double models). This will briefly be discussed in section 5.2, as well as speculation on further possible generalizations.

Of course, the machinery introduced here simplifies significantly if G is chosen to be a cyclic group $\mathbb{Z}_d$. In this case, the representations are all one-dimensional (1D) dth roots of unity. The basis change between the group element and representation bases for these groups is simply the discrete Fourier transform. Significantly, there also exists a natural isomorphism from the space of irreps of $\mathbb{Z}_d$ to the space of elements. This allows us to generalize the Hadamard gate and removes the necessity of considering only systems with CSS structure. The cluster states corresponding to the cyclic groups have been previously defined and studied [21, 22].

3. Generalized cluster states

We can use the general machinery established in the previous section to generalize the cluster state. This will give a family of states labelled by decorated graphs and a group G. As a prelude to the introduction of these general states, let us first review the definition of the qubit cluster state. Since the standard definition does not have CSS structure, we will need to consider a slightly modified definition of the cluster state that is more amenable to generalization.

3.1. Qubit cluster states

A qubit cluster state [1, 2] is uniquely specified by an underlying graph Λ. One convenient way to define these states is as the output of a certain constant depth quantum circuit. Equivalently, they can be described as the common +1 eigenstate of a set of commuting stabilizer operators. We will describe the qubit cluster state in both of these languages, as these two descriptions showcase different desirable features of the cluster state. In the first case, the cluster state can be prepared in constant time by an appropriate parallel quantum circuit, while in the latter case the cluster state can be seen as the unique gapped ground state of a local Hamiltonian, namely that formed by the (negative) sum of the relevant stabilizer operators.

The cluster state is given by

$$|\mathcal{C}_\Lambda\rangle = \prod_{\langle m,n\rangle} \mathrm{CPHASE}(m, n) \bigotimes_{v\in\Lambda} |+\rangle_v \tag{6}$$

with $\langle m, n\rangle$ running over all edges and v running over all vertices of Λ. The CPHASE gate is given by $\mathrm{CPHASE}(m, n)|a\rangle_m|b\rangle_n = (-1)^{ab}|a\rangle_m|b\rangle_n$ for $a, b \in \mathbb{Z}_2$. As a circuit, we place qubits in the $|+\rangle$ state at each site of Λ and then perform CPHASE gates between the qubits connected by an edge. Since the CPHASE gates commute, this is always a constant-depth circuit (assuming bounded degree of Λ).
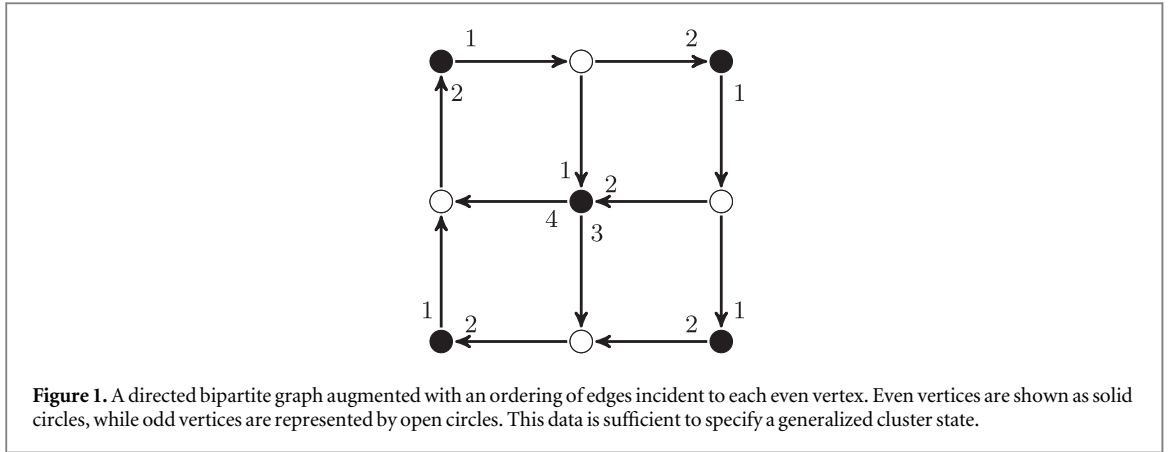
It is straightforward to derive the stabilizers of this state by considering the circuit of equation (6) in the Heisenberg representation. The qubit cluster state is thus the common +1 eigenstate of the stabilizer operators

$$S(v) = X(v) \prod_{w\sim v} Z(w) \tag{7}$$

for every site v, where $w \sim v$ runs over neighbours of v. Clearly, this state does not have the CSS structure discussed in section 2. This can be seen in two ways: firstly the stabilizers (7) involve both X and Z operators in the same stabilizer, and secondly the circuit (6) involved CPHASE operators. The toolkit introduced in section 2 cannot be used to generalize states of this form.

We can, however modify the definition of the cluster state to endow it with CSS structure. In order to do this, we must restrict to bipartite graphs Λ. On these graphs, we can partition the sites of Λ into an odd set Λ_o and an even set Λ_e such that all edges of Λ involve one vertex from Λ_o and one vertex from Λ_e. Given this structure, we can define the CSS cluster state as

$$|\mathcal{C}_\Lambda^{\mathrm{CSS}}\rangle \equiv \prod_{v\in\Lambda_e} H(v)|\mathcal{C}_\Lambda\rangle, \tag{8}$$

Figure 1. A directed bipartite graph augmented with an ordering of edges incident to each even vertex. Even vertices are shown as solid circles, while odd vertices are represented by open circles. This data is sufficient to specify a generalized cluster state.

where H is the Hadamard operator. This allows us to rewrite the circuit constructing the CSS cluster state as

$$|C_\Lambda^{\mathrm{CSS}}\rangle = \prod_{\substack{\langle m,n\rangle \\ m\in\Lambda_o, n\in\Lambda_e}} \mathrm{CNOT}(m,n) \bigotimes_{w\in\Lambda_o} |+\rangle_w \bigotimes_{v\in\Lambda_e} |0\rangle_v, \qquad (9)$$

where here the CNOT gates act with the odd qubit m as control and even qubit n as target. We will generally use the notation that a controlled gate takes two site labels as arguments and acts on the first as the control. The bipartite property of Λ guarantees that each CNOT gate acts on one odd and one even qubit. Even though CNOT gates do not commute in general, two CNOT gates with common targets or common controls will commute. This gives us the fact that the circuit specified by (9) is again always constant-depth as expected.

The stabilizers of the CSS cluster state are given by

$$S^e(v) = Z(v) \prod_{w\sim v} Z(w), \text{ and} \qquad (10)$$

$$S^o(w) = X(w) \prod_{v\sim w} X(v) \qquad (11)$$

for v and w even and odd sites respectively. This demonstrates the claimed CSS structure of these states. This can also be seen by noting that the circuit (9) consists only of operators that preserve CSS structure (i.e. CNOT gates). Although it can only be defined on bipartite graphs, when it exists the CSS cluster state is locally equivalent to the corresponding standard cluster state and so has all the same fundamental properties, notably including the ability to perform universal MBQC.

3.2. Finite group cluster states

Given the toolkit of section 2 and the definition of the CSS cluster state given above, we will now describe the generalization of the cluster state to arbitrary finite groups G. These cluster states will inherit many of the algebraic properties of the group that defines them.

As compared to the qubit cluster states where an undirected graph completely specifies the state, for an arbitrary group G this is insufficient to uniquely determine a generalized cluster state. This is similar to the generalization of the toric code to the quantum double models, where the former is completely specified by an undirected graph, while the latter requires directed edges to specify the model. In particular, to describe a generalized cluster state we require a directed bipartite graph Λ, together with an ordering $\#_v(w)$ of the neighbours w of each even vertex v. An example of such a structure is shown in figure 1. The roles that are played by the additional direction and ordering parameters will become clear during the construction of the generalized cluster states.

Given such a graph Λ and ordering $\#$, together with a finite group G, the generalized cluster state is a state on a system of $|G|$-dimensional qudits at each site of Λ. It is most convenient to generalize the qubit cluster state by considering the circuit that defines it according to equation (9). The elements in this circuit are CNOT gates, $|+\rangle$ states and $|0\rangle$ states, each of which has a natural analogue in table 1.

3.2.1. Circuit representation

The qubit $|+\rangle$ state is interpreted as a special case of the trivial irrep state $|I\rangle = \frac{1}{|G|}\sum_g |g\rangle$. Similarly the $|0\rangle$ state corresponds to the identity group element state $|e\rangle$ in general. Finally, the last ingredient in the CSS cluster state

preparation circuit is the CNOT gate, which generalizes to either the left or right controlled multiplication gates CMULT$^{\leftrightarrows}$.

Loosely speaking then, we define the generalized cluster state as

$$\left| C_{\Lambda,\#,G}^{\text{CSS}} \right\rangle \sim \prod_{\substack{<m,n> \\ m\in\Lambda_o, n\in\Lambda_e}} \text{CMULT}(m, n) \bigotimes_{w\in\Lambda_o} |I\rangle_w \bigotimes_{v\in\Lambda_e} |e\rangle_v. \tag{12}$$

There are two ways in which (12) does not yet constitute a full specification of the state $|C_{\Lambda,\#,G}^{\text{CSS}}\rangle$. The first is that there is an ambiguity regarding whether CMULT gates should act as left or right multiplication. The second is that we have not specified an order in which to apply the CMULT gates, which in general do not commute on their targets. Resolving these ambiguities requires the additional structure we have specified in the # orderings and in the directions of the edges of Λ. Note that since the CMULT gates commute on their controls, we need not be concerned about the relative order of gate application on odd qudits, and so need not specify a global order of CMULT application. It suffices to specify the ordering separately at each even vertex.

We can complete the specification of equation (12) by firstly choosing the ordering of the CMULT gates at a given even site v according to $\#_v$. Secondly, we choose the CMULT(m, n) gate to act as left multiplication if the edge (m, n) runs from n to m, and as right multiplication otherwise. This convention can be remembered by placing the control (odd) qudit on the left of the target (even) qudit. The multiplication sense is then given directly by the direction of the edge connecting the qudits. We denote these conventions in the following way:

$$\left| C_{\Lambda,\#,G}^{\text{CSS}} \right\rangle \equiv \prod_{\substack{<m,n> \\ m\in\Lambda_o, n\in\Lambda_e}}^{\#} \text{CMULT}^{\leftrightarrows}(m, n) \bigotimes_{w\in\Lambda_o} |I\rangle_w \bigotimes_{v\in\Lambda_e} |e\rangle_v, \tag{13}$$

where CMULT$^{\leftrightarrows}(m, n)$ acts as CMULT$^{\rightarrow}$ (CMULT$^{\leftarrow}$) for an edge directed from m to n (n to m). This is now a complete specification of the generalized cluster state $|C_{\Lambda,\#,G}^{\text{CSS}}\rangle$. Note that since the CMULT operators commute on their controls, the circuit (13) is always constant-depth as in the qubit case (again assuming Λ has bounded degree).

Strictly speaking, the orderings # contain more information than is necessary to specify the state $|C_{\Lambda,\#,G}^{\text{CSS}}\rangle$. Since left multiplication commutes with right multiplication, we could rearrange the order of CMULT application without affecting the state $|C_{\Lambda,\#,G}^{\text{CSS}}\rangle$ as long as we do not change the relative order of CMULT gates acting on edges directed outwards and the relative order of gates acting on edges directed inwards from a given even vertex. For notational simplicity we will continue to use the redundant description # of the ordering at each even vertex.

The circuit (13) is a perfectly adequate definition of the generalized cluster states. However, the stabilizer formalism has proved to be an extremely powerful tool in the analysis of cluster states and similar qubit systems. For this reason we will compute the stabilizer representation of the generalized cluster states.

3.2.2. Stabilizer representation

The term 'stabilizer' has slightly different meanings in different contexts. Often it is used to mean a subgroup of the n-qubit Pauli group that does not contain -1. The relevant stabilizer states are then the common $+1$ eigenstates of the stabilizer group. This is the sense in which the qubit cluster state and the CSS cluster state we discussed in section 3.1 are stabilizer states of their relative stabilizer groups.

More generally, stabilizer states can be thought of as the $+1$ eigenstates of an arbitrary set of operators. Varying degrees of structure can be introduced to make the set more amenable to study, for example requiring that the stabilizer operators are monomial matrices as in the monomial stabilizer formalism [27]. As with the stabilizers of the Kitaev quantum double models, the generalized cluster states can be cast in the monomial stabilizer formalism. We may abuse the language slightly by referring to a set of stabilizers in the generalized sense as a group, even if we do not present them in a form that has the algebraic structure of a group.

One of the most attractive features of the Pauli stabilizer formalism is the ability to efficiently simulate operations that map stabilizer states to stabilizer states (through the Gottesman–Knill theorem). This ability carries directly over to stabilizer formalisms based on abelian group algebras [28–30]. Such strong simulability properties seem to require more structure than is available for general stabilizer states. However, some notions of classical simulability remain in certain cases. For example, for sufficiently structured states such as the Kitaev quantum double models, the monomial stabilizer formalism allows the efficient evaluation of expectation values for local observables [27]. We expect analogous simulability results to apply to our models as they are based on the same algebraic structure.

Apart from the motivation of classical simulation, the existence of a local stabilizer description guarantees the existence of a local, commuting, gapped parent Hamiltonian for these states. It can also be a useful way of illustrating phenomena that would be more difficult to describe in the Schrödinger representation, for example

as is the case in the Kitaev quantum double models where stabilizer operators correspond directly to quantities of interest such as charges or Wilson loops.

In order to explicitly determine a set of stabilizer operators for the generalized cluster states, we examine the action of the circuit implied by equation (13) in the Heisenberg representation. This circuit begins with a product state of $|I\rangle$ on odd vertices and $|e\rangle$ on even vertices.

Now, notice that the state $|I\rangle = \frac{1}{\sqrt{|G|}}\sum_g|g\rangle$ is stabilized by all group multiplication operators. That is

$$X_g^{\leftrightarrows}|I\rangle = |I\rangle \tag{14}$$

for all g, and either multiplication sense $\leftrightarrows$. Similarly, we can define a set of stabilizers for $|e\rangle$, noting that

$$Z_{\Gamma^{ii}}|e\rangle = |e\rangle \tag{15}$$

for all Γ, i.

Although this is a perfectly legitimate stabilizer description of the state $|e\rangle$, for the purposes of calculating the stabilizers of the state $|C_{A,\#,G}^{\mathrm{CSS}}\rangle$ it will be more convenient to use an alternative representation. Defining the projectors to group element basis states $T_g \equiv |g\rangle\langle g|$, it is clear that $|e\rangle$ is stabilized by T_e. A more roundabout way to see this is to write the T_g operators in terms of the $Z_{\Gamma^{ij}}$ operators as

$$T_g = \frac{1}{|G|}\sum_{\Gamma} d_{\Gamma} \sum_{i,j}[\Gamma(g)]_{ij}^* Z_{\Gamma^{ij}} \tag{16}$$

and find

$$T_e|e\rangle = \frac{1}{|G|}\sum_{\Gamma} d_{\Gamma} \sum_{i,j}[\Gamma(e)]_{ij}^* Z_{\Gamma^{ij}}|e\rangle \tag{17}$$

$$= \frac{1}{|G|}\sum_{\Gamma} d_{\Gamma}^2 |e\rangle \tag{18}$$

$$= |e\rangle \tag{19}$$

as expected.

We thus have that prior to the application of any CMULT gates in equation (13) there are stabilizers for the system given by

$$\tilde{S}^e(v) = T_e(v), \text{ and} \tag{20}$$

$$\tilde{S}_g^o(w) = X_g^{\leftrightarrows}(w) \tag{21}$$

for every even site v and odd site w, and all $g \in G$. In order to determine the stabilizers of the generalized cluster state then, we simply need to study the evolution of these operators under the action of controlled multiplication gates. This can be computed straightforwardly, and found to give

$$
\begin{array}{cc}
\mathrm{CMULT}^{\leftarrow} & \mathrm{CMULT}^{\rightarrow} \\
X_g^{\leftarrow}\otimes I \;\rightarrow\; X_g^{\leftarrow}\otimes X_g^{\leftarrow} & X_g^{\leftarrow}\otimes I \;\rightarrow\; X_g^{\leftarrow}\otimes X_g^{\rightarrow} \\
X_g^{\rightarrow}\otimes I \;\rightarrow\; \sum_h X_g^{\rightarrow}T_h \otimes X_{hg^{-1}h^{-1}}^{\leftarrow} & X_g^{\rightarrow}\otimes I \;\rightarrow\; \sum_h X_g^{\rightarrow}T_h \otimes X_{hg^{-1}h^{-1}}^{\rightarrow} \\
I \otimes X_g^{\leftarrow} \;\rightarrow\; \sum_h T_h \otimes X_{hgh^{-1}}^{\leftarrow} & I \otimes X_g^{\leftarrow} \;\rightarrow\; I \otimes X_g^{\leftarrow} \\
I \otimes X_g^{\rightarrow} \;\rightarrow\; I \otimes X_g^{\rightarrow} & I \otimes X_g^{\rightarrow} \;\rightarrow\; \sum_h T_h \otimes X_{hgh^{-1}}^{\leftarrow} \\
T_g \otimes I \;\rightarrow\; T_g \otimes I & T_g \otimes I \;\rightarrow\; T_g \otimes I \\
I \otimes T_g \;\rightarrow\; \sum_h T_h \otimes T_{hg} & I \otimes T_g \;\rightarrow\; \sum_h T_h \otimes T_{gh^{-1}}
\end{array}
$$

where the first tensor factor is the control qudit and the second the target.

To construct the stabilizers of the generalized cluster states, we now build the state one CMULT gate at a time according to equation (13). We will first develop the stabilizers for the even sites of the lattice, and then return and compute the stabilizers corresponding to odd sites.

At each even site v of the lattice, prior to the application of any CMULT gates the state is stabilized by $\tilde{S}^e(v) = T_e(i)$. This site will be the target of any CMULT gates applied to it. For each edge directed outwards from v, a CMULT$^{\leftarrow}$ gate will be applied, and similarly a CMULT$^{\rightarrow}$ gate will be applied for each inwards directed edge. In order to describe the evolution of $\tilde{S}^e(v)$, it will be convenient to define $n_w^{\leftarrow v}$ and $n_u^{\rightarrow v}$ as the neighbours of v corresponding to the wth outward directed edge or uth inward directed edge respectively (where the ordering is given by $\#_v$).

Recalling that $\text{CMULT}^{\leftarrow}$ gates commute on common targets with $\text{CMULT}^{\rightarrow}$ gates, we will consider first performing the $\text{CMULT}^{\leftarrow}$ gates on site v. As we apply the CMULT gates on outward directed edges sequentially in the order specified by $\#_v$, the stabilizer for this site becomes

$$\tilde{S}^e(v) \rightarrow \sum_{g_1} T_{g_1}(v) T_{g_1}\left(n_1^{\leftarrow v}\right) \tag{22}$$

$$\rightarrow \sum_{g_1, g_2} T_{g_2 g_1}(v) T_{g_1}\left(n_1^{\leftarrow v}\right) T_{g_2}\left(n_2^{\leftarrow v}\right) \tag{23}$$

$$\vdots$$

$$\rightarrow \sum_{\substack{g_w \\ \forall w \leftarrow v}} T_{\left(\prod_{w'} g_{w'}\right)}(v) \prod_{w''} T_{g_{w''}}\left(n_{w''}^{\leftarrow v}\right), \tag{24}$$

where in the final expression w runs over all outward directed neighbours of v and $\prod_w g_w = \cdots g_3 g_2 g_1$ (we also suppress the range of product and summation indices when they are clear from context). We can then similarly apply the gates on the inward directed edges of v (again in the appropriate order) and obtain

$$\tilde{S}^e(v) \rightarrow \sum_{\substack{h_u \\ \forall u \rightarrow v}} \sum_{\substack{g_w \\ \forall w \leftarrow v}} T_{\left(\prod_{w'} g_{w'}\right)\left(\prod_{u'} h_{u'}\right)^{-1}}(v) \prod_{u''} T_{h_{u''}}\left(n_{u''}^{\rightarrow v}\right) \prod_{w''} T_{g_{w''}}\left(n_{w''}^{\leftarrow v}\right) \equiv S^e(v). \tag{25}$$

Since it now has support on sites other than v, one might now worry that we will also need to calculate how $\tilde{S}^e(v)$ transforms under the other CMULT gates acting on the neighbours of v. However, since these CMULT operations all commute with each other (they share a common control, not target), this is not a concern. Thus (37) is the stabilizer corresponding to each odd site of our cluster state, acting only on the given site and its nearest neighbours.

We can also derive the stabilizers corresponding to odd sites of Λ in a similar fashion. In this case, the stabilizers of an odd site w prior to the application of any CMULT gates are given by

$$\tilde{S}_g^o(w) = X_g^{\leftrightarrows}(w) \tag{26}$$

for all g. Although both the left and right multiplication operators form valid stabilizers for an odd site, only one set is needed to specify the state completely. For simplicity we will choose to study the left multiplication stabilizers though the analysis for the right multiplication operators would proceed in an analogous way. With this in mind, we restrict to the case

$$\tilde{S}_g^o(w) = X_g^{\leftarrow}(w). \tag{27}$$

The site w will act as the control for the CMULT gates applied to it. As such, it does not matter in which order the CMULT gates are applied at site w. However, in contrast to the even sites, the odd site stabilizers will be affected by the order of CMULT gates applied to the neighbouring (even) vertices.

The CMULT gate corresponding to an outward directed edge from w will act as $\text{CMULT}^{\rightarrow}$ and inward directed edges correspond to $\text{CMULT}^{\leftarrow}$. Applying a CMULT gate on an inward directed edge to neighbour v, we find $\tilde{S}_g^o(w)$ becomes

$$\tilde{S}_g^o(w) \rightarrow X_g^{\leftarrow}(w) X_g^{\leftarrow}(v). \tag{28}$$

Now this operator has support on v as well as w. This means that we must consider the effects of the CMULT gates acting not only between w and its neighbours, but also between v and its neighbours. In doing so, we will only need to consider the effects of those gates applied to w after that corresponding to the edge $(w \leftarrow v)$. In addition, we need only consider edges with the same direction from v (i.e. outwards from v in this case).

We can make this effect explicit by finding the effect of these CMULT gates on an $X_g^{\leftarrow}$ operator acting on site v. As each successive $\text{CMULT}^{\leftarrow}$ is applied to edges connecting v to sites $\tilde{w}_1$, $\tilde{w}_2$, $\ldots$ that follow w in the $\#_v$ order, this operator transforms as

$$X_g^{\leftarrow}(v) \rightarrow \sum_{h_1} X_{h_1 g h_1^{-1}}^{\leftarrow}(v) T_{h_1}\left(\tilde{w}_1\right) \tag{29}$$

$$\rightarrow \sum_{h_1, h_2} X_{(h_2 h_1) g (h_2 h_1)^{-1}}^{\leftarrow}(v) T_{h_1}\left(\tilde{w}_1\right) T_{h_2}\left(\tilde{w}_2\right) \tag{30}$$

$$\vdots$$

$$\rightarrow \sum_{\substack{h_{\tilde{w}} \\ \forall \tilde{w} \leftarrow v, \tilde{w} > w}} X_{(\prod_{\tilde{w}'} h_{\tilde{w}'})g(\prod_{\tilde{w}''} h_{\tilde{w}''})^{-1}}^{\leftarrow}(v) \prod_{\tilde{w}'''} T_{h_{\tilde{w}''}}(\tilde{w}'''), \tag{31}$$

where $\tilde{w} > w$ restricts to those $\tilde{w}$ that come after w in $\#_v$.

This operator still acts as multiplication on site v, but by a group element conjugated by terms taking into account states of the neighbours of v. It will be convenient to define this operator for any site v neighbouring w:

$$\tilde{X}_g^{\leftarrow}\left(v|_w\right) \equiv \sum_{\substack{h_{\tilde{w}} \\ \forall \tilde{w} \leftarrow v, \tilde{w} > w}} X_{(\prod_{\tilde{w}'} h_{\tilde{w}'})g(\prod_{\tilde{w}''} h_{\tilde{w}''})^{-1}}^{\leftarrow}(v) \prod_{\tilde{w}'''} T_{h_{\tilde{w}''}}(\tilde{w}'''). \tag{32}$$

Now having considered all the effects of site v and its neighbours on the stabilizer for site w, we have the stabilizer

$$\tilde{S}_g^o(w) \rightarrow X_g^{\leftarrow}(w)\tilde{X}_g^{\leftarrow}\left(v|_w\right). \tag{33}$$

As we apply this procedure to each of the edges directed inwards to w, the stabilizer transforms to

$$\tilde{S}_g^o(w) \rightarrow X_g^{\leftarrow}(w) \prod_{v \rightarrow w} \tilde{X}_g^{\leftarrow}\left(v|_w\right). \tag{34}$$

In a similar way, we can also consider the effects of CMULT gates on outwards directed edges from w. If we apply a CMULT gate corresponding to an outward directed edge connecting site w and u, the $X^{\leftarrow}(w)$ operator will be transformed by each gate as

$$X^{\leftarrow}(w) \rightarrow X^{\leftarrow}(w)X^{\rightarrow}(u). \tag{35}$$

Following an analogous prescription to that used for the analysis of the inward directed edges, we are led to define the operator

$$\tilde{X}_g^{\rightarrow}\left(u|_w\right) \equiv \sum_{\substack{h_{\tilde{w}} \\ \forall \tilde{w} \rightarrow u, \tilde{w} > w}} X_{(\prod_{\tilde{w}'} h_{\tilde{w}'})g(\prod_{\tilde{w}''} h_{\tilde{w}''})^{-1}}^{\rightarrow}(u) \prod_{\tilde{w}'''} T_{h_{w''}}(\tilde{w}'''). \tag{36}$$

This allows us to compute the stabilizers corresponding to each odd site w of a generalized cluster state as

$$\tilde{S}_g^o(w) \rightarrow X_g^{\leftarrow}(w) \prod_{v \rightarrow w} \tilde{X}_g^{\leftarrow}\left(v|_w\right) \prod_{u \leftarrow w} \tilde{X}_g^{\rightarrow}\left(u|_w\right) \equiv S_g^o(w). \tag{37}$$

In contrast to the qubit cluster state, whose stabilizers only act on a site and its nearest neighbours, in general the operators (37) act on a site, its nearest neighbours, and also its next-nearest neighbours.

Together, the even site stabilizers S^e (25) and the odd site stabilizers S^o (37) completely specify a generalized cluster state. The odd site stabilizers are monomial matrices in their current form, while the even site stabilizers are projectors and so can be made monomials by considering the alternative operators $2S^e - 1$. This means that the generalized cluster states can be studied in the framework of monomial stabilizer groups [27] as claimed.

4. Properties of generalized cluster states

We will now explore some salient features of the generalized cluster states defined in the previous section. It is not our intention to provide a comprehensive specification of the properties of the states, but merely to comment on some features of the qubit cluster states and how they survive in the general case.

One of the most important features of the qubit cluster state is that (on a suitable graph) it is a resource for universal MBQC [2]. Since the qubit cluster state already has this universality property, considering generalized cluster states as resources for standard MBQC will not give any qualitative advantage in this sense. However, it is possible that generalized cluster states may yield advantages in terms of practical application, for example by finding more efficient implementations of non-Clifford gates or by having desirable spectral properties (when considered as ground states of Hamiltonians). We will not directly address these issues of practical advantage here, though some of the general properties we discuss may be relevant to any such study.

4.1. Measurements
The results of measurements in the standard qubit cluster states can easily be determined using the stabilizer formalism. The properties of the resulting states are largely insensitive to the measurement outcomes obtained. By comparison, measurements on generalized cluster states give rise to some qualitatively new phenomena that do not appear in the qubit cluster state, such as significant dependence of the output state on measurement outcomes.

Figure 2. Two different three qudit cluster states. Solid (open) circles represent even (odd) vertices. The simple structure of these graphs means that a vertex ordering is not required to specify the relevant cluster state. (a) An even–odd–even three qudit cluster state. (b) An odd–even–odd three qudit cluster state.

Ideally, we would like to understand the effect of measurements in the group element basis and the representation basis (the analogues of Z and X Pauli measurements) on an arbitrary generalized cluster state. However, since we can no longer use the simple Pauli stabilizer formalism to compute these effects, this general specification is outside the scope of this work. For this reason we will not exhaustively describe the results of measurement procedures on general cluster states. Instead, we will illustrate the kinds of new phenomena that can appear in a simple example. For this purpose, it is sufficient to consider three qudits on a line as in figure 2(a) or (b).

For a group G, these states are stabilized by the operators

$$S(\text{oeo}, 1) = \sum_{g \in G} T_g \otimes T_g \otimes I, \qquad S_g(\text{oeo}, 1) = X_g^{\leftarrow} \otimes X_g^{\leftarrow} \otimes I, \qquad (38)$$

$$S_g(\text{oeo}, 2) = X_g^{\leftarrow} \otimes X_g^{\leftarrow} \otimes X_g^{\leftarrow}, \qquad S(\text{oeo}, 2) = \sum_{g,h \in G} T_g \otimes T_{gh^{-1}} \otimes T_h \qquad (39)$$

$$S(\text{oeo}, 3) = \sum_{g \in G} I \otimes T_g \otimes T_g, \text{ and} \qquad S_g(\text{oeo}, 3) = I \otimes X_g^{\rightarrow} \otimes X_g^{\rightarrow}, \qquad (40)$$

where eoe or oeo represent the even–odd–even and odd–even–odd cluster states of figures 2(a) and (b) respectively.

Explicitly then, these states can be written as analogues of the GHZ state in different bases:

$$|C_{\text{eoe}}\rangle = \frac{1}{\sqrt{|G|}} \sum_{g \in G} |g\rangle \otimes |g\rangle \otimes |g\rangle \qquad (41)$$

$$|C_{\text{oeo}}\rangle = \frac{1}{|G|} \sum_{g,h \in G} |g\rangle \otimes |gh^{-1}\rangle \otimes |h\rangle. \qquad (42)$$

The states $|C_{\text{eoe}}\rangle$ are clearly identical for any different choices of group G with the same order. For a general graph of course this is not true (as with $|C_{\text{oeo}}\rangle$), but nonetheless we will see group structure arise in the analysis of measurement outcomes on $|C_{\text{eoe}}\rangle$. This is because the natural measurements to consider are analogues of Pauli X and Z operators in the qubit case, and these operators also inherit structure of the group G.

The two measurements we consider here are those in the group element basis $\{|g\rangle\}$ (corresponding to Pauli Z) and in the representation basis $\{|\Gamma^{ij}\rangle\}$ (corresponding to Pauli X). If we measure the central qubit of our cluster in these bases, we will find that some phenomena arise which have no counterpart in the qubit cluster state.

Let us first recall what happens in the CSS qubit case (i.e. $G = \mathbb{Z}_2$). Since the eoe and oeo qubit cluster states can be transformed into one another by Hadamard gates on each qubit, we need only consider one of these states (the behaviour of the other can be determined by exchanging X and Z). If we measure the central qubit of the eoe cluster in the Z basis and find outcome $m_z \in \mathbb{Z}_2$, the state on the remaining qubits becomes a product state $|m_z\rangle \otimes |m_z\rangle$. Alternatively, if we measure in the X basis with measurement outcome $m_X \in \{\pm\}$, we find a maximally entangled state $|0\rangle \otimes |0\rangle + m_x |1\rangle \otimes |1\rangle$.

For a general group G, the group element basis measurement proceeds in much the same fashion. For a outcome $m_g \in G$ of this measurement on an eoe cluster state, the resulting state is given by the product state $|m_g\rangle \otimes |m_g\rangle$. Similarly, the oeo cluster state is transformed to the maximally entangled state $\sum_{h \in G} |h\rangle \otimes |h^{-1}m_g\rangle$. In contrast, when performing a measurement in the representation basis, we find a qualitative departure from the qubit case. Such a measurement yields a triple of measurement outcomes (m_Γ, m_i, m_j) representing a matrix element of an irrep of G. Beginning with the eoe cluster state, we note that it can be rewritten as

$$|C_{\text{eoe}}\rangle = \sum_g \sum_{\Gamma^{ij}} \frac{\sqrt{d_\Gamma}}{|G|} [\Gamma(g)]_{ij}^* |g\rangle |\Gamma^{ij}\rangle |g\rangle. \qquad (43)$$

Thus if we measure the central qudit in the representation basis, the resulting state is given as

$$\sum_g \sqrt{\frac{d_{m_\Gamma}}{|G|}} \left[m_\Gamma(g) \right]^*_{m_i m_j} |g\rangle |g\rangle. \qquad (44)$$

For any abelian group, this will give a maximally entangled state as in the qubit case. However for a general group, the state (44) will be some less-than-maximally entangled state for any representation m_Γ with dimension greater than 1. We can see this by calculating the reduced density matrix of the first qudit as $\rho^{(1)}_{C_{eoe}} = \sum_g \frac{d_{m_\Gamma}}{|G|} |[m_\Gamma(g)]_{m_i m_j}|^2 |g\rangle\langle g|$ which is not, in general, equal to $\sum_g \frac{1}{|G|} |g\rangle\langle g|$. Of course, for any abelian group all $d_{m_\Gamma} = 1 = |[m_\Gamma(g)]_{m_i m_j}|^2$, and we recover the maximally entangled state as claimed.

The analogous measurement on an oeo cluster state produces a similar phenomenon. The resulting state after measurement is given by

$$|C_{oeo}\rangle \rightarrow \sum_{g,h} \frac{\sqrt{d_{m_\Gamma}}}{|G|} \left[m_\Gamma\left(gh^{-1}\right) \right]^*_{m_i m_j} |g\rangle |h\rangle \qquad (45)$$

$$= \sum_{g,h \in G} \sum_{k=1}^{d_{m_\Gamma}} \frac{\sqrt{d_{m_\Gamma}}}{|G|} \left[m_\Gamma(g) \right]^*_{m_i k} \left[m_\Gamma\left(h^{-1}\right) \right]^*_{k m_j} |g\rangle |h\rangle \qquad (46)$$

$$= \sum_{g,h \in G} \sum_{k=1}^{d_{m_\Gamma}} \frac{\sqrt{d_{m_\Gamma}}}{|G|} \left[\overline{m_\Gamma}(g) \right]_{k m_i} \left[m_\Gamma(h) \right]_{m_j k} |g\rangle |h\rangle \qquad (47)$$

$$= \sum_{k=1}^{d_{m_\Gamma}} \frac{1}{\sqrt{d_{m_\Gamma}}} \left| \overline{m_\Gamma}^{k m_i} \right\rangle \left| m_\Gamma^{m_j k} \right\rangle \qquad (48)$$

for $\overline{m_\Gamma}$ the conjugate representation to m_Γ (recall that we consider only unitary irreducible representations over $\mathbb{C}$). This leaves us with a maximally entangled state of dimension d_{m_Γ}. For abelian groups with only 1D irreps, this gives the product state as in the qubit case. However, for general groups the behaviour is non-trivial.

In particular, these phenomena mean that the property of maximal connectedness (i.e. any two qudits in the state can be brought into a maximally entangled state with certainty by measurement) is not always present for a generalized cluster state as it is for the qubit cluster state.

In order to develop a standard MBQC protocol making use of the generalized cluster states, compensating for the subtle interplay between measurement outcomes and remaining entanglement would require tools outside the scope of standard cluster state computation methods. It may be possible to develop such a scheme explicitly by making use of techniques in [31], combined with the PEPS representation of the generalized cluster states as found in section 4.3. However, we will not consider this question further in this work, instead focussing on the relationship between the generalized cluster states and the Kitaev quantum double models as discussed in sections 4.4 and 5.1. For this relationship, that the remaining entanglement in the state depends on prior measurement outcomes is crucial in reproducing the phenomenology of non-abelian topological orders.

4.2. Global symmetries

The 1D cluster state is well known to have a global $\mathbb{Z}_2 \times \mathbb{Z}_2$ symmetry that has a significant effect on its properties [23, 32]. This allows it to be placed in the framework of symmetry-protected topological order (SPTO) [33–35]. The symmetry group can also be shown to be related to the power and robustness of the cluster state as a measurement-based resource in both 1 and higher dimensions [36, 37].

The relevant global symmetry group of the infinite 1D CSS cluster state is represented by the operators

$$U^o = \prod_{s \text{ odd}} X(s), \text{ and} \qquad (49)$$

$$U^e = \prod_{s \text{ even}} Z(s). \qquad (50)$$

Since U^o and U^e clearly commute and are self-inverse, these operators generate a representation of $\mathbb{Z}_2 \times \mathbb{Z}_2$.

Considering the analogous property of a generalized cluster state for arbitrary finite groups G, for simplicity we will restrict to the specific infinite 1D graph as shown in figure 3.
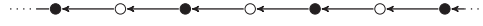
Figure 3. An infinite 1D generalized cluster state graph.

Explicitly, the cluster state corresponding to this graph can be written as

$$|C_{\text{line}}\rangle = \sum_{g_i} \cdots \left|g_i g_{i+1}^{-1}\right\rangle |g_{i+1}\rangle \left|g_{i+1} g_{i+2}^{-1}\right\rangle |g_{i+2}\rangle \left|g_{i+2} g_{i+3}^{-1}\right\rangle |g_{i+3}\rangle \left|g_{i+3} g_{i+4}^{-1}\right\rangle \cdots . \tag{51}$$

It is clear by inspection that the global symmetries of this state can be written as

$$U_g^o = \prod_{s \text{ odd}} X_g^{\rightarrow}(s), \tag{52}$$

$$U_\Gamma^e = \frac{1}{d_\Gamma} \prod_{s \text{ even}} \sum_{i_s} Z_{\Gamma^{i_s - 2i_s}}(s) \tag{53}$$

as can be directly verified. As in the qubit case the U_g^o and U_Γ^e commute trivially. The U_g^o multiply as elements of G, and the U_Γ^e transform as representations of G (i.e. $U_{\Gamma_1}^e U_{\Gamma_2}^e = U_{\Gamma_1 \otimes \Gamma_2}^e$ and $U_{\Gamma_1}^e + U_{\Gamma_2}^e = U_{\Gamma_1 \oplus \Gamma_2}^e$). Thus we deduce that the symmetry algebra of the generalized cluster state is given by the product of the group algebra and the dual (representation) algebra. For abelian groups such as $\mathbb{Z}_2$, the representation algebra is isomorphic to that of the group itself, which recovers the familiar result for the qubit cluster state.

Although the framework of SPTO typically deals with states that have a group symmetry, we anticipate that many of the tools and results of SPTO may be extended to this more general setting.

4.3. PEPS representations

The qubit cluster state has an exact tensor network representation as a PEPS [24]. The PEPS ansatz [38–43] is extremely useful for both analytical and numerical analysis of states. In particular, the description of the cluster state as a PEPS allows a reinterpretation of MBQC as a teleportation-based computation scheme [24] or in the correlation space picture [31]. Furthermore, it has also enabled an approximate two-body parent Hamiltonian to be derived for an arbitrary cluster state [44–46]. Here we show how the PEPS representation of a cluster state generalizes for an arbitrary finite group G.

A PEPS is defined by an interaction graph Λ and a set of 'projection' maps $\mathcal{P}_s$ associated with each site of Λ. The state represented by the PEPS can be constructed by beginning with a set of D-dimensional maximally entangled 'virtual' pairs $\sum_{i=1}^{D} |i\rangle_{mn} |i\rangle_{nm} \equiv |\Phi_D\rangle_{(m,n)}$ along each edge (m, n) of Λ, with one qudit of each of these pairs associated with each of the vertices forming the edge (i.e. the qudit labelled mn is associated with vertex m and vice versa). At each site $s \in \Lambda$, the projection map $\mathcal{P}_s$ is then applied to the qudits at each site, taking the combined D^k-dimensional virtual Hilbert space (for a site of valency k) to a d-dimensional 'physical' Hilbert space. These physical qudits at each site of Λ then typically correspond to the individual qudits of the state being represented. That is, the PEPS state is given by

$$|\psi_{\text{PEPS}}\rangle \equiv \bigotimes_{v \in \Lambda} \mathcal{P}(v) \bigotimes_{(m,n) \in \Lambda} |\Phi_D\rangle_{(m,n)}. \tag{54}$$

For the CSS qubit cluster state, the interaction graph Λ is simply the (bipartite) graph specifying the cluster state. $D = d = 2$ and the projection maps $\mathcal{P}$ are defined on odd and even sites of Λ as [24]

$$\mathcal{P}^o = |0\rangle\langle 0, \ldots, 0| + |1\rangle\langle 1, \ldots, 1|, \tag{55}$$

$$\mathcal{P}^e = |+\rangle\langle +, \ldots, +| + |-\rangle\langle -, \ldots, -|. \tag{56}$$

For generalized cluster states, the PEPS interaction graph is again given by the cluster state graph. $D = d = |G|$ as might be expected, and the projection maps are given (up to normalization) by

$$\mathcal{P}^o = \sum_g |g\rangle\langle g, \ldots, g|, \tag{57}$$

$$\mathcal{P}^e = \sum_{g_1, \ldots, g_k, h_{k+1}, \ldots, h_l} \left| \left(\prod_i g_i \right) \left(\prod_j h_j \right)^{-1} \right\rangle \langle g_1, \ldots, g_k, h_{k+1}, \ldots, h_l |, \tag{58}$$

where the g_i represent the states of qudits corresponding to edges leaving the relevant even vertex, and the h_j correspond to the edges entering the vertex. The products $\prod_i g_i$ and $\prod_j h_j$ are taken in the appropriate order specified by #.

In order to demonstrate that equations (57)–(58) define a PEPS for the generalized cluster states as claimed, it is useful to note that the effect of applying controlled multiplication gates on the physical level with target $|e\rangle$ can be reproduced by application of corresponding gates on the virtual level. We will distinguish typographically between CMULT gate acting on a physical target qudit and cmult gates acting on the virtual qudits.

Consider CMULT gates acting on virtual control qudits and a common physical target qudit:

$$\prod_{i=1}^{l}{}^{\#} \text{CMULT}^{\leftrightarrows}(i, s)\left|g_1, ..., g_k, h_{k+1}, ..., h_l\right\rangle |e\rangle_s$$

$$= \left|g_1, ..., g_k, h_{k+1}, ..., h_l\right\rangle \left|\left(\prod_i g_i\right)\left(\prod_j h_j\right)^{-1}\right\rangle_s, \tag{59}$$

where the i index runs over the qudits grouped in the first ket. Compare this to CMULT gates acting on virtual control and target qudits, before projecting the target qudits into a single physical qudit space:

$$I \otimes \mathcal{P}_s^e \prod \text{cmult}^{\leftarrow}\left|g_1, ..., g_k, h_{k+1}, ..., h_l\right\rangle \otimes |e, e, ..., e\rangle_s$$

$$= I \otimes \mathcal{P}_s^e \left|g_1, ..., g_k, h_{k+1}, ..., h_l\right\rangle \otimes \left|g_1, ..., g_k, h_{k+1}, ..., h_l\right\rangle_s \tag{60}$$

$$= \left|g_1, ..., g_k, h_{k+1}, ..., h_l\right\rangle \otimes \left|\left(\prod_i g_i\right)\left(\prod_j h_j\right)^{-1}\right\rangle_s \tag{61}$$

$$= \prod_{i=1}^{l}{}^{\#} \text{CMULT}^{\leftrightarrows}(i, s)\left|g_1, ..., g_k, h_{k+1}, ..., h_l\right\rangle |e\rangle_s, \tag{62}$$

where here only one CMULT gate acts on each (virtual) control and each (virtual) target qudit. We can use this equivalence to give us the result that

$$\left|\psi_{\text{PEPS}}\right\rangle = \left(\bigotimes_{w' \in \Lambda_o} \mathcal{P}^o(w') \bigotimes_{v' \in \Lambda_e} \mathcal{P}^e(v')\right) \bigotimes_{(w,v) \in \Lambda} \left|\Phi_{|G|}\right\rangle_{wv} \tag{63}$$

$$= \left(\bigotimes_{w' \in \Lambda_o} \mathcal{P}^o(w') \bigotimes_{v' \in \Lambda_e} \mathcal{P}^e(v')\right) \prod_{\substack{<m,n> \\ m \in \Lambda_o, n \in \Lambda_e}} \text{cmult}^{\leftarrow}(m, n)\left(\bigotimes_{w \in \Lambda_o} |I, ..., I\rangle_w \bigotimes_{v \in \Lambda_e} |e, ..., e\rangle_v\right) \tag{64}$$

$$\propto \prod_{\substack{<m,n> \\ m \in \Lambda_o, n \in \Lambda_e}}^{\#} \text{CMULT}^{\leftrightarrows}(m, n)\left(\bigotimes_{w \in \Lambda_o} |I\rangle_w \bigotimes_{v \in \Lambda_e} |e\rangle_v\right) \tag{65}$$

$$= \left|C_{\Lambda, \#, G}\right\rangle \tag{66}$$

as claimed, where on the penultimate line we used the facts that $\mathcal{P}_s^o$ commutes with CMULT on the control qudit, and $|I\rangle_s \propto \mathcal{P}_s^o |I, ..., I\rangle_s$.

4.4. Generalized cluster states and Kitaev quantum double states

The qubit cluster state is related to the topologically ordered toric code in several ways. One important way is that the toric code can be defined by preparing a suitable cluster state and projecting (or measuring) a subset of the qubits into suitable states [6]. Here we show that this relationship also extends between the generalized cluster states and the Kitaev quantum double models based on the same group. This relationship may also have practical application, for example in preparing certain quantum double states or in generalizing the topological cluster state computation protocol, as we will discuss in section 5.1.

Recall that a Kitaev quantum double model is defined on a directed lattice. For simplicity, we make a convenient canonical choice of edge direction (such that the edges around each plaquette either run clockwise or anticlockwise, as shown in figure 4(a)) and note that all alternative choices may be reached by local basis change. Quantum double states on these lattices have stabilizers [4]
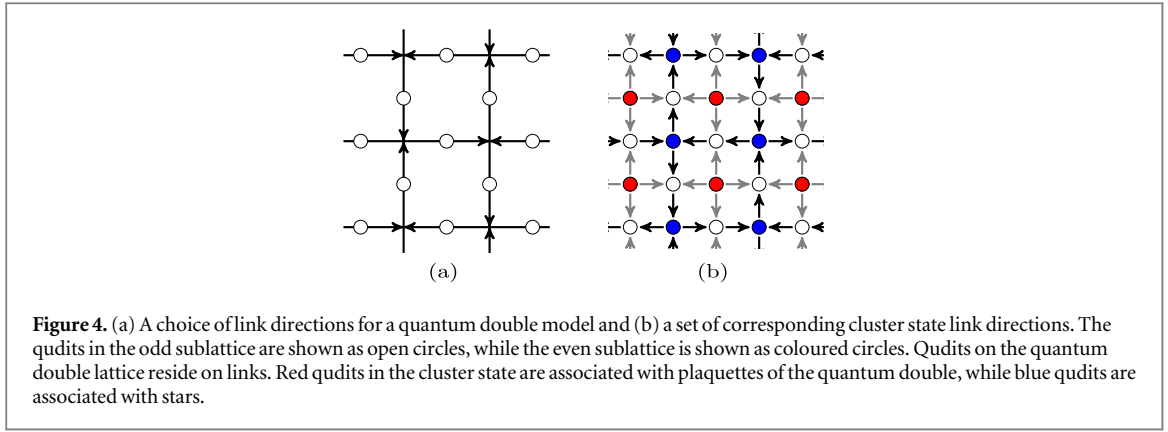
Figure 4. (a) A choice of link directions for a quantum double model and (b) a set of corresponding cluster state link directions. The qudits in the odd sublattice are shown as open circles, while the even sublattice is shown as coloured circles. Qudits on the quantum double lattice reside on links. Red qudits in the cluster state are associated with plaquettes of the quantum double, while blue qudits are associated with stars.

$$A(s) = \sum_{g \in G} A_g(s), \quad \text{and} \tag{67}$$

$$B(p) = B_e(p) \tag{68}$$

for

$$A_g^h(s) = X_g^{\rightarrow}(s_U) X_g^{\leftarrow}(s_L) X_g^{\rightarrow}(s_D) X_g^{\leftarrow}(s_R), \tag{69}$$

$$A_g^v(s) = X_g^{\leftarrow}(s_U) X_g^{\rightarrow}(s_L) X_g^{\leftarrow}(s_D) X_g^{\rightarrow}(s_R), \tag{70}$$

$$B_g^{\text{CW}}(p) = \sum_{g_1 g_2 g_3 g_4 = g} T_{g_1}(p_U) T_{g_2}(p_L) T_{g_3}(p_D) T_{g_4}(p_R), \tag{71}$$

$$B_g^{\text{ACW}}(p) = \sum_{g_1^{-1} g_2^{-1} g_3^{-1} g_4^{-1} = g} T_{g_1}(p_U) T_{g_2}(p_L) T_{g_3}(p_D) T_{g_4}(p_R), \tag{72}$$

with s_U, s_L, s_D, s_R and p_U, p_L, p_D, p_R the neighbouring links located up, left, right, and down from the star s or plaquette p under consideration, where CW or ACW denotes whether the edges around the plaquette p run clockwise or anticlockwise, and where h or v denote whether the horizontal or vertical edges point into the star s. The $A_g(s)$ and $B_e(p)$ commute pairwise, and the Kitaev quantum double ground state is defined as the common $+1$ eigenspace of all $A_g(s)$ and $B_e(p)$ operators.

In order to produce such a quantum double ground state, we begin with a generalized cluster state on a square lattice with half the lattice spacing of the quantum double lattice, as shown in figure 4(b). The even qudits of the generalized cluster state (shown as coloured circles) will be projected into suitable states, while the odd sublattice qudits will form the space on which the quantum double state is defined. The red odd qudits correspond to plaquettes of the quantum double state, and as such can be labelled as either clockwise or anti-clockwise, depending on the direction of the links around the plaquette. Blue (odd) cluster state qudits correspond to vertices of the quantum double model, and can be labelled as horizontal or vertical depending on whether the horizontal or vertical incident links run into the vertex. An equivalent procedure can be found to project the odd qudits and retain the even qudits, but for simplicity we will treat only one case.

The edge directions and vertex orderings of the cluster state lattice will have a significant effect on the final state. In particular, if we wish to recover a quantum double state on the lattice with edge directions as in figure 4(a), one suitable choice of edge directions is shown in figure 4(b). Explicitly, the edges of the cluster state drawn in black in figure 4(b) (i.e. those that will form the quantum double lattice) should run in the opposite direction to the corresponding quantum double links. All grey edges run away from (odd) red sites. For each red site corresponding to an anticlockwise quantum double plaquette, the vertex ordering should be taken anticlockwise beginning from the topmost edge, while for each red site corresponding to a clockwise quantum double plaquette, the vertex ordering should be taken clockwise beginning from the topmost edge. The ordering of the edges around the blue sites is not particularly important, for simplicity we choose them anticlockwise beginning from the topmost edge. The origin of these convention choices may not appear particularly clear at this stage, but it will become apparent how they figure as we proceed in the analysis.

The stabilizers for this generalized cluster state are given for red, blue, and odd sites respectively as

$$S^{r,\text{CW}}(p) = \sum_h T_h(p) \sum_{g_1 g_2 g_3 g_4 = h} T_{g_4}(p_U) T_{g_1}(p_L) T_{g_2}(p_D) T_{g_3}(p_R), \tag{73}$$

$$S^{r,\mathrm{ACW}}(p) = \sum_h T_h(p) \sum_{g_3 g_2 g_1 g_4 = h} T_{g_4}\big(p_U\big) T_{g_1}\big(p_L\big) T_{g_2}\big(p_D\big) T_{g_3}\big(p_R\big), \tag{74}$$

$$S^{b,v}(s) = \sum_h T_h(s) \sum_{g_3 g_1 g_4^{-1} g_2^{-1} = h} T_{g_4}(s_U) T_{g_1}(s_L) T_{g_2}(s_D) T_{g_3}(s_R), \tag{75}$$

$$S^{b,h}(s) = \sum_h T_h(s) \sum_{g_2 g_4 g_1^{-1} g_3^{-1} = h} T_{g_4}(s_U) T_{g_1}(s_L) T_{g_2}(s_D) T_{g_3}(s_R), \tag{76}$$

$$S^{o,u}(l) = X_g^{\leftarrow}(l) \tilde{X}_g^{\leftarrow}\big(l_D|_l\big) \tilde{X}_g^{\leftarrow}\big(l_L|_l\big) \tilde{X}_g^{\rightarrow}\big(l_U|_l\big) \tilde{X}_g^{\rightarrow}\big(l_R|_l\big), \tag{77}$$

$$S^{o,d}(l) = X_g^{\leftarrow}(l) \tilde{X}_g^{\rightarrow}\big(l_D|_l\big) \tilde{X}_g^{\rightarrow}\big(l_L|_l\big) \tilde{X}_g^{\leftarrow}\big(l_U|_l\big) \tilde{X}_g^{\rightarrow}\big(l_R|_l\big), \tag{78}$$

$$S^{o,l}(l) = X_g^{\leftarrow}(l) \tilde{X}_g^{\leftarrow}\big(l_D|_l\big) \tilde{X}_g^{\leftarrow}\big(l_L|_l\big) \tilde{X}_g^{\rightarrow}\big(l_U|_l\big) \tilde{X}_g^{\rightarrow}\big(l_R|_l\big), \tag{79}$$

$$S^{o,r}(l) = X_g^{\leftarrow}(l) \tilde{X}_g^{\rightarrow}\big(l_D|_l\big) \tilde{X}_g^{\rightarrow}\big(l_L|_l\big) \tilde{X}_g^{\leftarrow}\big(l_U|_l\big) \tilde{X}_g^{\leftarrow}\big(l_R|_l\big), \tag{80}$$

where, as in the definition of the quantum double models, CW and ACW refer to the type of plaquette corresponding to the red site, v and h denote the type of star corresponding to the blue site, stabilizers associated to even sites have a u, d, l, r designation according to the direction of the corresponding quantum double link, and vertices of the cluster state have neighbours denoted by U, D, L, and R subscripts.

To produce the Kitaev quantum double ground state from the generalized cluster state we have just described, we project the red even qudits (corresponding to plaquettes of the quantum double model) onto the $|e\rangle$ state, and the blue even qudits (corresponding to vertices) into the $|I\rangle$ state. In the case of the qubit cluster state, this corresponds to projections to the $|0\rangle$ or $|+\rangle$ states, respectively.

Projecting even sites into $|I\rangle$ effectively removes them from the cluster state (indeed, we could alternatively have begun with a cluster state on a lattice lacking the blue sites). After performing just these blue site projections, the red site stabilizers are unchanged, the blue site stabilizers vanish and the odd site stabilizers are transformed to

$$S_g^{o,u}(l) \rightarrow X_g^{\leftarrow}(l) \tilde{X}_g^{\leftarrow}\big(l_D|_l\big) \tilde{X}_g^{\rightarrow}\big(l_U|_l\big), \tag{81}$$

$$S_g^{o,d}(l) \rightarrow X_g^{\leftarrow}(l) \tilde{X}_g^{\rightarrow}\big(l_D|_l\big) \tilde{X}_g^{\leftarrow}\big(l_U|_l\big), \tag{82}$$

$$S_g^{o,l}(l) \rightarrow X_g^{\leftarrow}(l) \tilde{X}_g^{\leftarrow}\big(l_L|_l\big) \tilde{X}_g^{\rightarrow}\big(l_R|_l\big), \tag{83}$$

$$S_g^{o,r}(l) \rightarrow X_g^{\leftarrow}(l) \tilde{X}_g^{\rightarrow}\big(l_L|_l\big) \tilde{X}_g^{\leftarrow}\big(l_R|_l\big), \tag{84}$$

where the $\tilde{X}$ operators are now defined with respect to the lattice where the blue sites have been removed.

If we then proceed with the projection of the red sites, the red plaquette stabilizers can easily be seen to evolve to

$$S^{\mathrm{CW}}(p) \rightarrow \sum_{g_1 g_2 g_3 g_4 = e} T_{g_4}\big(p_U\big) T_{g_1}\big(p_L\big) T_{g_2}\big(p_D\big) T_{g_3}\big(p_R\big), \tag{85}$$

$$S^{\mathrm{ACW}}(p) \rightarrow \sum_{g_1^{-1} g_2^{-1} g_3^{-1} g_4^{-1} = e} T_{g_4}\big(p_U\big) T_{g_1}\big(p_L\big) T_{g_2}\big(p_D\big) T_{g_3}\big(p_R\big). \tag{86}$$

It is less obvious, but products of the odd site stabilizers around each former blue site s can be rearranged to form the following stabilizers after red site projection:

$$S_g^h(s) = X_g^{\rightarrow}(s_U) X_g^{\leftarrow}(s_L) X_g^{\rightarrow}(s_D) X_g^{\leftarrow}(s_R), \tag{87}$$

$$S_g^v(s) = X_g^{\leftarrow}(s_U) X_g^{\rightarrow}(s_L) X_g^{\leftarrow}(s_D) X_g^{\rightarrow}(s_R). \tag{88}$$

This can most easily be seen by considering an explicit example of a cluster state in the neighbourhood of a single (removed) h-type blue site:

$$\sim \sum_{g_1, g_2, g_3, g_4} \begin{array}{ccc} |g_4 g_1\rangle & |g_4\rangle & |g_3 g_4\rangle \\ |g_1\rangle & & |g_3\rangle \\ |g_2 g_1\rangle & |g_2\rangle & |g_3 g_2\rangle \end{array}. \tag{89}$$

This state is not stabilized by the appropriate operator (87). However, if we then project the red sites into $|e\rangle$, we find:

$$\sum_{g_1, g_2, g_3, g_4} \delta_{g_2 g_1 = e} \delta_{g_3 g_2 = e} \delta_{g_3 g_4 = e} \delta_{g_4 g_1 = e} \quad |g_1\rangle \quad \begin{matrix} |g_4\rangle \\ \\ |g_2\rangle \end{matrix} \quad |g_3\rangle \tag{90}$$

which is indeed stabilized by S_g^h as claimed. Note that if we had chosen a projection such that, for example, $g_3 g_2 = h$ for some $h \neq e$, then this would no longer be true. If instead, we projected into $\sum_{h \in C} |h\rangle$ for some conjugacy class C, S_g^h would again stabilize the state. In the language of the quantum double anyons, this is due to the fact that only uniform superpositions over conjugacy classes correspond to pure magnetic type excitations.

    The stabilizers we just derived (85)–(88) are precisely the $A_g(s)$ and $B_e(p)$ stabilizers of the quantum double (67)–(68). Thus we have a procedure to define the topologically ordered Kitaev quantum double states by a sequence of projections on the generalized cluster states.

    One might hope that that for any scheme which builds some interesting state by measurement of Z and X on a (bipartite) qubit cluster state, we could define a generalized model for a finite group G built by performing suitable projections on the generalized cluster state as we have done here. These models may or may not be able to be actually constructed from a cluster state in general by measurement (their interesting properties may or may not survive the complications of non-ideal measurement outcomes, as will be mentioned in section 5.1.1), but simply being able to define such models may be of independent interest. An example of a model that can be defined in such a way is a non-abelian generalization of the colour codes, presented in [47]. One should also note the generality (and hence complexity) of using the projection procedure in this way. We are able to define many generalizations within this same framework by choosing different even and odd sublattices, different link directions, and different link orderings at vertices. Furthermore, alternative projections can be chosen to generalize the measurement of the Z or X operators, as is done in [47]. In general, the properties of the resulting state will be significantly affected by the choices of these parameters.

5. Discussion

5.1. Applications

5.1.1. Producing the quantum double states

In practice, one may be interested in using the relationship between the generalized cluster states and quantum double models discussed in section 4.4 as a resource to prepare quantum double states in the laboratory, by replacing the projections by suitable measurements. In the case of the toric code, replacing $|0\rangle$ and $|+\rangle$ projections with measurements in the Z and X bases respectively does not significantly affect the properties of the resulting state. However, in general the effects outlined in section 4.1 mean that the situation is not so simple.

    To illustrate some of the resulting phenomena, consider replacing the $|e\rangle$ projections at each red site in figure 4(b) by measurements in the group element basis. After projection of the blue sites of the cluster state shown in figure 4(b) into the $|I\rangle$ state (equivalently, removal of the blue sites), the stabilizers of the state corresponding to the red sites are given as in equations (73)–(74), while the remaining odd site stabilizers are as in (81)–(84). The effect on the red site stabilizers of measuring these sites in the group element basis is clear: with measurement outcomes $\{m_p\}$, the stabilizers transform to

$$S^{\mathrm{CW}}(p) = \sum_{g_1 g_2 g_3 g_4 = m_p} T_{g_4}\left(p_U\right) T_{g_1}\left(p_L\right) T_{g_2}\left(p_D\right) T_{g_3}\left(p_R\right), \tag{91}$$

$$S^{\mathrm{ACW}}(p) = \sum_{g_3 g_2 g_1 g_4 = m_p} T_{g_4}\left(p_U\right) T_{g_1}\left(p_L\right) T_{g_2}\left(p_D\right) T_{g_3}\left(p_R\right). \tag{92}$$

    The effect on the odd site stabilizers is less obvious. As in the previous section where red sites were projected into $|e\rangle$, there will be a stabilizer corresponding to each star of the lattice. These can also be straightforwardly calculated by considering a single isolated vertex and its four neighbouring plaquettes.

    We can interpret the state after measurement of the blue sites as a superposition of excited states of quantum double anyons. Explicitly, anyons in the quantum double models can be created and transported by so-called ribbon operators $F_\rho^{(h,g)}$ corresponding to pairs of group elements (h, g) and fattened paths (ribbons) ρ on the lattice [4, 48]. The ends of ribbons, 'sites', are composed of a neighbouring vertex-star pair. Ribbon operators create anyonic charges at each end of the ribbon ρ. In particular, the operator $F_\rho^{(m_p,e)}$ ending on an appropriate site will transform a stabilizer of the form (85) or (86) into (91) or (92) respectively. Similarly, the star-type stabilizers can be obtained by transforming the naive stabilizers (87) and (88) under $F_\rho^{(m_p,e)}$ on suitable ribbons.

Further, if we replaced all blue site projections with measurements in the representation basis, we could interpret the remaining state in a similar way.

In some senses, this is a trivial statement, as all states on the lattice (up to boundary conditions) are superpositions of some excited states of the quantum double model. However, using this interpretation may be fruitful for example in a detailed analysis of the topological cluster state computation protocol sketched in the following section.

5.1.2. Adiabatic topological cluster state quantum computation (TCSQC)

Apart from standard MBQC, the qubit cluster state can also be used to implement TCSQC [11]. This is a computation scheme that combines practical advantages of MBQC with some natural robustness to noise from topological computation schemes. It is based on the relationship between the cluster states and the toric code discussed and generalized in section 4.4.

In the TCSQC protocol, measurements are performed on a cubic lattice cluster state. By interpreting one direction of the lattice as a 'simulated time' direction, the resulting procedure can be interpreted as the evolution of punctures in the surface of the toric code in simulated time. The world-lines of the punctures are determined by the chosen measurement settings, while the measurement outcomes may be interpreted as specifying world-lines of toric code anyons, which can then be compensated for by classical post-processing. Though this scheme enjoys the native robustness that accompanies topological computation, the gates that can be performed in this way are insufficient for universal quantum computation, and so must be supplemented with non-topological operations whose fault-tolerance is guaranteed separately.

It is an interesting question as to whether a similar measurement-based computation scheme exists that enjoys universal topological quantum computation. An obvious candidate strategy is to attempt to replace the simulated toric code punctures with simulated defects in another more complicated topological model, such as the quantum doubles. If we were to construct an analogous procedure using the generalized cluster states for a non-abelian group G, appropriately generalizing the qubit TCSQC protocol would not generally succeed for the reason that the effect of random measurement outcomes could not be efficiently classically processed. Random measurement outcomes would be interpreted as (possibly superpositions of) anyons, and their dynamics—and hence effect on the logical state—is unlikely to be efficiently classically calculable in general (particularly if the anyon dynamics is BQP complete).

One possible way to salvage such a generalized topological cluster state computation scheme is by removing the measurements from the protocol altogether. Adiabatic cluster state computation [49–51] makes use of the cluster state as a modular prototype for computation by adiabatic deformation of a Hamiltonian. A standard measurement-based cluster state computation may be translated to the adiabatic setting by simulating the measurement with the adiabatic application of a strong field in the direction of the desired measurement result. Thus in this scheme, the analogue of measurement results are not random, and so need not be compensated for by classical post-processing.

Though the adiabatic cluster state computation approach is typically applied to a standard cluster state computation, we could equally well apply it to the topological cluster state scheme. In this setting, the choice of fields would effectively set the world-lines of punctures and anyons in simulated time, without randomness. Making use of these techniques, it should be possible to perform universal topological cluster state computation with the generalized cluster states by simulating the evolution of punctures (as with the standard TCSQC scheme) in a sufficiently complicated Kitaev quantum double model[3]. Alternatively, since the world-lines of anyons may now be controlled directly by choosing the adiabatic fields appropriately, and since braiding of suitable anyons is sufficient to implement universal quantum computation [4, 52, 53], it should also be possible to compute in a generalized topological cluster state computation scheme by adiabatically simulating the evolution of a suitable class of anyons. These methods for adiabatic topological cluster state computation may be contrasted with the direct approach of adiabatic topological quantum computation [20].

5.1.3. Universal topological MBQC

In the scheme laid out in section 5.1.2 for adiabatic topological quantum computation with these generalized cluster states, it was in part because of the infeasibility of classical processing that standard measurement-based computing techniques could not be used. The universality of quasiparticle braiding in the relevant anyon model meant that it was infeasible to calculate the effect of any stray braids caused by undesirable measurement outcomes (the most naive implementation of the scheme would in fact produce superpositions of anyon states

---

[3] Though computation by braiding punctures in non-abelian anyon systems has not been extensively studied, the fact that punctures contain anyonic charges and the braiding of anyonic charges can be sufficient for universal quantum computing [4, 52, 53] suggests that the braiding of punctures should also be sufficient for universal quantum computation in general.

which would generally be non-simulable in any case, but we believe that it may be possible to circumvent this with a more sophisticated protocol in some cases).

However, recall that in the qubit topological cluster state computation scheme the computation proceeds not by simulating anyon braiding, but by simulating the braiding of punctures in the surface of the system. It is not clear that the computational power of such puncture braiding and the computational power of anyon braiding in such a model must always coincide. In fact, for another type of topological defect known as a twist, braiding of these objects can be universal for quantum computation while the braiding of the relevant anyons remains classically simulatable [54]. Thus it may be possible to find some anyon model where braiding of punctures is universal, while braiding of anyons is classically simulatable. Although the interaction between topological defects and anyons can be quite complex, this could conceivably lead to a universal topological MBQC scheme based on an appropriate generalized cluster state (whether as described in this paper or from the possible extensions noted below).

5.2. Extensions
Although as presented here, the generalized cluster states are based on finite group algebras, there is little in the construction that is particularly restricted to finite groups. Several related or further generalizations of this construction suggest themselves.

The clearest example is the extension from finite groups to Lie groups. In particular, by considering the group of real numbers under addition, our construction reproduces the continuous-variable cluster states [55, 56]. This gives a unified framework for all previously known variants of the cluster state. This will be presented elsewhere [57].

As this construction was inspired by the generalization of the toric code to the quantum double models based on finite groups, it is natural to ask if the later generalizations of the toric code to arbitrary finite-dimensional Hopf $C^\star$ algebras [17] and the conjectured generalization to weak Hopf algebras [17, 58] could also be applied to generalized cluster states. There is an obstacle in directly extending the construction to Hopf algebras, as the natural generalization of the controlled multiplication operation would be according to the coproduct Δ of the Hopf algebra. This is the operation that maps from one system to a tensor product of two systems.

The coproduct of a group algebra is simply $\Delta(g) = g \otimes g$ for $g \in G$. In a general Hopf algebra, the coproduct of an element is given by $\Delta(a) = \sum_i a_{(1)}^{(i)} \otimes a_{(2)}^{(i)}$. The natural generalization of the controlled multiplication gate is defined using this structure, e.g. CMULT $|a\rangle|b\rangle = \sum_i |a_{(1)}^{(i)}\rangle|a_{(2)}^{(i)}b\rangle$. However, this gate no longer need commute on control qudits. For this reason, if we were to use this CMULT gate to construct a cluster state, the circuit would no longer be finite-depth and the stabilizer operators derived in analogy to those in section 3 would no longer be finite weight in general.

One way to define cluster states based on Hopf algebras that can be constructed by constant depth circuits and have local stabilizers would be to restrict the graphs on which these states are defined. In particular, consider a graph Λ that is bipartite and every site v in the even sublattice Λ_e is two-valent. Then one edge incident to v could act as controlled left multiplication and the other edge acts as controlled right mutliplication. Since these two operations always commute on common targets, this construction would yield a state with the desired properties. Of course choosing graphs of this kind is a very restrictive constraint, and so it would be interesting to see if an alternative mechanism to retain locality is possible.

Apart from Lie groups and Hopf algebras as motivated by the Kitaev quantum double models, it would also be interesting to study generalizations of the cluster states motivated by the Levin–Wen string net models [59]. Instead of a group, these models are specified by a fusion category. The simplest choice gives the toric code as in the quantum double models. As in table 1, we could build qudits and associated operator algebras inheriting features of a given fusion category. It would be interesting to develop cluster states for a given fusion category in this way.

Finally, we note that in some of these extensions it may be possible to construct some cluster states that are related to the standard definition (6) instead of the CSS definition (9). This should be possible when considering algebraic objects which are self-dual in the relevant sense (as with abelian groups).

5.3. Broader implications
As noted in section 4.2, the generalized cluster states on an infinite chain extend the standard notion of symmetry protected states with a symmetry group to states that simply have a symmetry algebra. The consequences of this are not clear, and it may be of interest to consider the notion of symmetry protected phases that are labelled by more general objects than groups.

The Pauli stabilizer formalism has proved spectacularly successful at describing a wide variety of states, and making them amenable to both analytical and numerical study. Several similar constructions have been proposed, including the monomial stabilizer formalism [27] mentioned earlier, among others [60, 61]. It is

known that all Pauli stabilizer states are equivalent (under local Clifford circuits) to a qubit cluster state [12]. It may be of interest to determine whether a similar family of states can be defined by local equivalence to a generalized cluster state for any given group G. Although these all fall under the umbrella of monomial stabilizer states, it may be advantageous to define smaller classes that have more structure than the general case.

Acknowledgments

References

[1] Raussendorf R and Briegel H J 2001 A one-way quantum computer *Phys. Rev. Lett.* **86** 5188
[2] Raussendorf R, Browne D E and Briegel H J 2003 Measurement-based quantum computation on cluster states *Phys. Rev. A* **68** 022312
[3] Briegel H, Browne D, Dür W, Raussendorf R and Van den Nest M 2009 Measurement-based quantum computation *Nat. Phys.* **5** 19
[4] Kitaev A 2003 Fault-tolerant quantum computation by anyons *Ann. Phys.* **303** 2
[5] Dennis E, Kitaev A, Landahl A and Preskill J 2002 Topological quantum memory *J. Math. Phys.* **43** 4452
[6] Raussendorf R, Bravyi S and Harrington J 2005 Long-range quantum entanglement in noisy cluster states *Phys. Rev. A* **71** 062313
[7] Han Y-J, Raussendorf R and Duan L-M 2007 Scheme for demonstration of fractional statistics of anyons in an exactly solvable model *Phys. Rev. Lett.* **98** 150404
[8] Brown B J, Son W, Kraus C V, Fazio R and Vedral V 2011 Generating topological order from a two-dimensional cluster state using a duality mapping *New J. Phys.* **13** 065010
[9] Bombin H and Martin-Delgado M A 2006 Topological quantum distillation *Phys. Rev. Lett.* **97** 180501
[10] Bombin H and Martin-Delgado M A 2008 Statistical mechanical models and topological color codes *Phys. Rev. A* **77** 042322
[11] Raussendorf R, Harrington J and Goyal K 2007 Topological fault-tolerance in cluster state quantum computation *New J. Phys.* **9** 199
[12] Van den Nest M, Dehaene J and De Moor B 2004 Graphical description of the action of local Clifford transformations on graph states *Phys. Rev. A* **69** 022316
[13] Anders J and Browne D E 2009 Computational power of correlations *Phys. Rev. Lett.* **102** 050502
[14] Raussendorf R 2013 Contextuality in measurement-based quantum computation *Phys. Rev. A* **88** 022322
[15] Nayak C, Simon S H, Stern A, Freedman M and Das Sarma S 2008 Non-abelian anyons and topological quantum computation *Rev. Mod. Phys.* **80** 1083
[16] Bombin H and Martin-Delgado M A 2009 Quantum measurements and gates by code deformation *J. Phys. A: Math. Theor.* **42** 095302
[17] Buerschaper O, Mombelli J M, Christandl M and Aguado M 2013 A hierarchy of topological tensor network states *J. Math. Phys.* **54** 012201
[18] Hu Y, Wan Y and Wu Y-S 2013 Twisted quantum double model of topological phases in two dimensions *Phys. Rev. B* **87** 125114
[19] Bravyi S and Kitaev A 2005 Universal quantum computation with ideal Clifford gates and noisy ancillas *Phys. Rev. A* **71** 022316
[20] Cesare C, Landahl A J, Bacon D, Flammia S T and Neels A 2014 Adiabatic topological quantum computing (arXiv:1406.2690)
[21] Zhou D L, Zeng B, Xu Z and Sun C P 2003 Quantum computation based on d-level cluster state *Phys. Rev. A* **68** 062303
[22] Hall W 2007 Cluster state quantum computation for many-level systems *Quantum Inf. Comput.* **7** 184
[23] Son W, Amico L, Fazio R, Hamma A, Pascazio S and Vedral V 2011 Quantum phase transition between cluster and antiferromagnetic states *Europhys. Lett.* **95** 50001
[24] Verstraete F and Cirac J I 2004 Valence-bond states for quantum computation *Phys. Rev. A* **70** 060302
[25] Calderbank A R and Shor P W 1996 Good quantum error-correcting codes exist *Phys. Rev. A* **54** 1098
[26] Steane A M 1996 Error correcting codes in quantum theory *Phys. Rev. Lett.* **77** 793
[27] Van den Nest M 2011 A monomial matrix formalism to describe quantum many-body states *New J. Phys.* **13** 123004
[28] Van den Nest M 2013 Efficient classical simulations of quantum Fourier transforms and normalizer circuits over abelian groups *Quantum Inf. Comput.* **13** 1007
[29] Bermejo-Vega J and Van den Nest M 2014 Classical simulations of abelian-group normalizer circuits with intermediate measurements *Quantum Inf. Comput.* **14** 181
[30] Bermejo-Vega J, Lin C Y-Y and Nest M V d 2014 Normalizer circuits and a Gottesman–Knill theorem for infinite-dimensional systems (arXiv:1409.3208)
[31] Gross D, Eisert J, Schuch N and Perez-Garcia D 2007 Measurement-based quantum computation beyond the one-way model *Phys. Rev. A* **76** 052315
[32] Son W, Amico L and Vedral V 2012 Topological order in 1D Cluster state protected by symmetry *Quantum Inf. Process.* **11** 1961
[33] Gu Z-C and Wen X-G 2009 Tensor-entanglement-filtering renormalization approach and symmetry-protected topological order *Phys. Rev. B* **80** 155131
[34] Chen X, Gu Z-C and Wen X-G 2011 Classification of gapped symmetric phases in one-dimensional spin systems *Phys. Rev. B* **83** 035107
[35] Schuch N, Perez-Garcia D and Cirac I 2011 Classifying quantum phases using matrix product states and projected entangled pair states *Phys. Rev. B* **84** 165139
[36] Else D V, Schwarz I, Bartlett S D and Doherty A C 2012 Symmetry-protected phases for measurement-based quantum computation *Phys. Rev. Lett.* **108** 240505
[37] Else D V, Bartlett S D and Doherty A C 2012 Symmetry protection of measurement-based quantum computation in ground states *New J. Phys.* **14** 113016
[38] Affleck I, Kennedy T, Lieb E H and Tasaki H 1988 Valence bond ground states in isotropic quantum antiferromagnets *Commun. Math. Phys.* **115** 477

[39] Fannes M, Nachtergaele B and Werner R F 1992 Finitely correlated states on quantum spin chains *Commun. Math. Phys.* **144** 443

[40] Hastings M B 2006 Solving gapped Hamiltonians locally *Phys. Rev.* B **73** 085115

[41] Pérez-García D, Verstraete F, Wolf M M and Cirac J I 2007 Matrix product state representations *Quantum Inf. Comput.* **7** 401

[42] Verstraete F, Wolf M, Pérez-García D and Cirac J I 2006 Projected entangled states: properties and applications *Int. J. Mod. Phys.* B **20** 5142

[43] Vidal G 2003 Efficient classical simulation of slightly entangled quantum computations *Phys. Rev. Lett.* **91** 147902

[44] Bartlett S D and Rudolph T 2006 Simple nearest-neighbor two-body Hamiltonian system for which the ground state is a universal resource for quantum computation *Phys. Rev.* A **74** 040302

[45] Griffin T and Bartlett S D 2008 Spin lattices with two-body Hamiltonians for which the ground state encodes a cluster state *Phys. Rev.* A **78** 062306

[46] Brell C G, Bartlett S D and Doherty A C 2014 Perturbative 2-body parent Hamiltonians for projected entangled pair states *New J. Phys.* **16** 123056

[47] Brell C G 2014 Generalized color codes supporting non-abelian anyons (arXiv:1408.6238)

[48] Bombin H and Martin-Delgado M A 2008 Family of non-abelian kitaev models on a lattice: topological condensation and confinement *Phys. Rev.* B **78** 115421

[49] Bacon D and Flammia S T 2010 Adiabatic cluster-state quantum computing *Phys. Rev.* A **82** 030303

[50] Bacon D, Flammia S T and Crosswhite G M 2013 Adiabatic quantum transistors *Phys. Rev.* X **3** 021015

[51] Antonio B, Markham D and Anders J 2013 Adiabatic graph-state quantum computation *New J. Phys.* **16** 113070

[52] Mochon C 2003 Anyons from nonsolvable finite groups are sufficient for universal quantum computation *Phys. Rev.* A **67** 022315

[53] Mochon C 2004 Anyon computers with smaller groups *Phys. Rev.* A **69** 032306

[54] Barkeshli M, Jian C-M and Qi X-L 2013 Twist defects and projective non-abelian braiding statistics *Phys. Rev.* B **87** 045130

[55] Zhang J and Braunstein S L 2006 Continuous-variable Gaussian analog of cluster states *Phys. Rev.* A **73** 032318

[56] Menicucci N C, van Loock P, Gu M, Weedbrook C, Ralph T C and Nielsen M A 2006 Universal quantum computation with continuous-variable cluster states *Phys. Rev. Lett.* **97** 110501

[57] Brell C G and Menicucci N C in preparation

[58] Buerschaper O, Christandl M, Kong L and Aguado M 2013 Electric-magnetic duality of lattice systems with topological order *Nucl. Phys.* B **876** 619

[59] Levin M A and Wen X-G 2005 String-net condensation: a physical mechanism for topological phases *Phys. Rev.* B **71** 045110

[60] Gottesman D 1999 Fault-tolerant quantum computation with higher-dimensional systems *Quantum Computing and Quantum Communications* (*Lecture Notes in Computer Science* vol 1509) ed C Williams (Berlin: Springer) pp 302–13

[61] Ni X, Buerschaper O and Nest M V d 2014 A non-commuting stabilizer formalism (arXiv:1404.5327)