A 3D lattice defect and efficient computations in topological MBQC

Gabrielle Tournaire[1,2], Marvin Schwiering[3], Robert Raussendorf[2,3], and Sven Bachmann[1,4]

[1]Department of Physics and Astronomy, University of British Columbia Vancouver, Canada
[2]Stewart Blusson Quantum Matter Institute, University of British Columbia Vancouver, Canada
[3]Institut für Theoretische Physik, Leibniz Universität Hannover, Germany
[4]Department of Mathematics, University of British Columbia Vancouver, Canada

We describe an efficient, fully fault-tolerant implementation of Measurement-Based Quantum Computation (MBQC) in the 3D cluster state. The two key novelties are (i) the introduction of a lattice defect in the underlying cluster state and (ii) the use of the Rudolph-Grover rebit encoding. Concretely, (i) allows for a topological implementation of the Hadamard gate, while (ii) does the same for the phase gate. Furthermore, we develop general ideas towards circuit compaction and algorithmic circuit verification, which we implement for the Reed-Muller code used for magic state distillation. Our performance analysis highlights the overall improvements provided by the new methods.

1 Introduction

Decoherence is an obstacle to scaling up quantum computers. It was once thought fundamental [1], but was later shown to be surmountable by the techniques of fault-tolerant quantum computation [2, 3, 4]. The central result in this regard is the Threshold Theorem [5, 6], which says that an arbitrarily long and accurate quantum computation is possible if the amount of decoherence per elementary quantum gate is below a certain value—the error threshold. Furthermore, the operational and spatial costs of fault-tolerance are benign in the sense of scaling, albeit large in absolute terms.

After the threshold theorem was established, work on fault-tolerant quantum computation has focused on (i) relaxing the preconditions under which the theorem can be applied [7, 8], (ii) improving the value of the error threshold [9], (iii) reducing the cost of fault-tolerance [10, 11, 12, 13, 14], and (iv) adapting fault-tolerance to specific architectures with certain operational constraints, such as limitations in the range [15] or speed [16] of interaction.

The present work contributes to reducing the cost of fault tolerance, for a scheme of fault-tolerant quantum computation with high threshold that only requires nearest-neighbor entangling gates in a 3D or quasi-2D geometry. In this paper, we present a rigorous and complete description of Measurement-Based Quantum Computation in the RHG 3D cluster state [15]. We first state and prove a general theorem on the topological implementation of gates by braiding holes in a Toric code. The corresponding tangle supports correlation surfaces that carry quantum information. The theorem is a three-dimensional analog of the foundational result of [17], which exhibits the intrinsic relation

between the realization of a logical gate and the measurement pattern inside the 3D cluster. The theorem is flexible enough to allow for physical defects in the a priori regular and self-dual lattice supporting the cluster state. Using this, we show that the introduction of a local dislocation defect, which breaks the duality, allows for an intrinsically fault-tolerant implementation of the Hadamard gate, which was so far a missing gate in this setup. The dislocation defect may be regarded as a cluster state counterpart to the twist-defect in the Toric code [18, 19]. However, in the present case, topological gates are obtained by braiding hole-type defects with the new dislocation defect, not by braiding twist defects among themselves. This and the Rudolph-Grover rebit encoding allow, in turn, for a topological implementation of the S gate, yielding an intrinsically fault-tolerant, distillation-free, realization of the full Clifford group.

Altogether, these novelties in the setup of MBQC yield a gain of one order of magnitude in the overhead of the S gate, compared to previous proposals involving magic state distillation. The final gate needed to achieve universality, the T gate, remains in need of distillation. We propose an approach towards reducing the overhead using both topological and geometric optimization of the circuit, with the goal of simplifying and compacting the tangle needed to implement the gates. Since implemented gates are given by surfaces supported on the tangle, the validity of the tangle is hard to verify by hand: we present a computer program that does this automatically throughout the circuit. For the concrete example of the Reed-Muller encoding used in the magic state distillation procedure required for a T gate, our proposal has been verified by our program and reduces the overhead by half an order of magnitude.

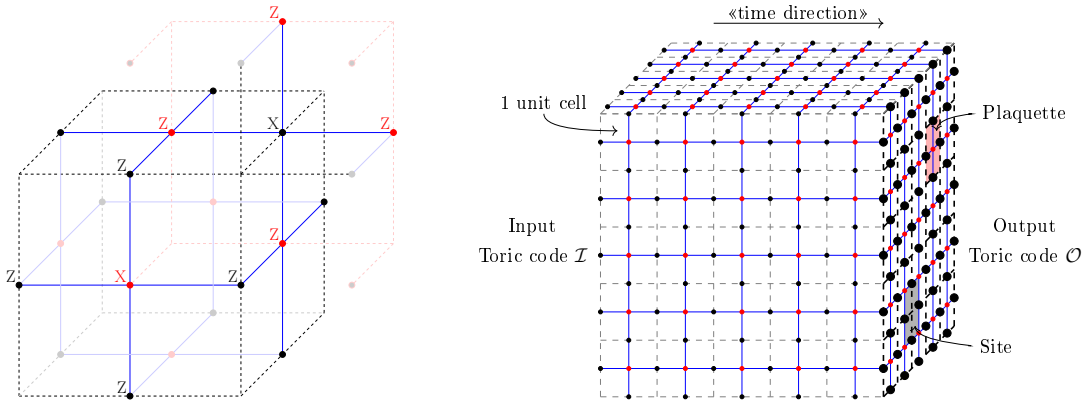
The paper is organized as follows. Sec. 2 first reviews the structure of the 3D cluster state as introduced in [20, 15] and the basics of MBQC: how individually measuring the qubits in the bulk projects a Toric code on its boundary. Sec. 3 introduces the notions of measurement patterns and surfaces compatible with them. This allows for the proof of Theorem 1 which connects measurement patterns with unitary gates. After briefly reviewing elementary gates, Sec. 4 discusses the Hadamard gate, the S gate, and analyzes their respective overhead. In Sec. 5, we concentrate on the T gate and compute the associated overhead. Finally, we turn to circuit volume optimization in Sec. 6. We propose equivalent circuits for magic state distillation, focusing on the improvement over a 'naive' implementation. We additionally developed and implemented an algorithm able to verify the optimization: It is described in the appendix and available at [21].

2 The 3D cluster state and the Toric code

We start with a brief review of the three-dimensional cluster state [22] and its relation to the two-dimensional Toric code [23, 24]. Following [15], physical qubits are placed on the edges of two simple cubic lattices (which we refer to as primal, respectively dual) that are shifted from each other so that the centers of the primal cubes are the corners of the dual ones and vice versa, see Fig. 1, where we distinguish the two lattices by a black, respectively red color. The fundamental graph defining the cluster state (in blue in the figure) is made up of nearest-neighbor qubits. To prepare the cluster state, one performs a CZ gate on every edge of the graph with all qubits initially in the $|+\rangle$ state. Note that these gates all commute with each other and that $CZ_{a,b}\,|{+}{+}\rangle = CZ_{b,a}\,|{+}{+}\rangle$. Therefore, the unambiguously defined resulting global state is given by

$$|\phi\rangle_{\mathcal{C}} = \prod_{\langle r,b\rangle} CZ_{r,b} \bigotimes_{q\in\mathcal{C}} |+\rangle_q \tag{1}$$

where $\langle r, b\rangle$ stands for nearest neighbors (which are necessarily of different color) and $\mathcal{C}$ denotes the set of sites carrying a qubit.



(a) Unit cell of the 3D cluster state.

(b) $5 \times 5 \times 5$ cluster state lattice.

Figure 1: The cluster state lattice: one unit cell (a) and a $5 \times 5 \times 5$ lattice (b). The physical qubits are located on the red and black dots. The entanglement induced by the CZ gates is represented as solid blue lines. The underlying cubic lattice is represented by dashed lines, black qubits sit on the edges, red qubits on the faces. (b) For computation purposes, the input Toric code $\mathcal{I}$ is inserted on the cluster left plane (not visible on the figure because of perspective). It is foliated in the horizontal direction, thus simulating a «time» evolution. The output Toric code $\mathcal{O}$ is made of the highlighted black qubits on the right plane.

We recall the homological language introduced in [15], which best describes the mathematical structure underlying the cluster state, see also [24, 25]. This language has recently received increasing interest, in particular in the context of Low Density Parity Check codes [26]. The primal cubes and dual cubes are in fact the three-dimensional cells of two cell complexes that are dual to each other. We denote the set of primal volumes by G_3, of primal square by G_2, of primal edges by G_1 and of primal vertices by G_0. The n-chain vector spaces will be denoted by $C_n = C_n(G_n, \mathbb{F}_2)$ and it is made of formal linear combinations of n-cells with $\mathbb{F}_2$ coefficients, namely $c = \sum_{g \in G_n} \alpha_g g$ with $\alpha_g \in \mathbb{F}_2 = \{0, 1\}$. We denote the usual boundary operator $\partial_n : C_n \to C_{n-1}$. The same can be done with the dual complex, which we denote $\overline{C}_3, \overline{C}_2, \overline{C}_1, \overline{C}_0$. The physical qubits lie on primal and dual edges. The two complexes are dual to each other with the duality given by

$$\overline{C}_n \cong C_{3-n} \tag{2}$$

Since

$$X_a Z_b C Z_{a,b} = C Z_{a,b} X_a, \tag{3}$$

a cluster state can equivalently be characterized by the fact that its stabilizer group is generated by the local operators

$$K(a) = X_a \bigotimes_{b \in \partial a} Z_b$$

where $a \in G_2$ or $a \in \overline{G}_2$, see again Fig. 1. We further note that $K(a)^2 = \mathbb{I}$ and that this is a set of commuting operators. Consequently, the stabilizer group $\mathcal{K}$ of the cluster state is isomorphic to the group $C_2 \oplus \overline{C}_2$.

This isomorphism can be made explicit by the following expression for a general stabilizer in terms of Pauli operators. For any $c_2 \in C_2$, we define the following stabilizer:

$$K(c_2) = \prod_{a \in G_2} K(a)^{\alpha_a}.$$

Since all edges that are not on the boundary ∂c_2 appear twice in the product and $Z^2 = \mathbb{I}$, the stabilizer has the simpler expression

$$K(c_2) = X(c_2)Z(\partial c_2), \tag{4}$$

where $X(c_2) = \bigotimes_{r\in c_2} X_r$ and similarly for $Z(\partial c_2)$. For later consistency, we also set $X(\emptyset) = \mathbb{I} = Z(\emptyset)$. In other words, this is the product of X on the red qubits inside the surface times a product of Z on the black qubits on the boundary. The same holds of course for dual surfaces and yield

$$K(\bar{c}_2) = X(\bar{c}_2)Z(\partial \bar{c}_2). \tag{5}$$

Some care must be taken at the boundary of the cluster, where the surfaces may end. For simplicity, we consider again Fig. 1 and imagine that the primal face lies on the boundary of the cluster. Then the dual cube sketched is only partially in the cluster, and one checks that the product X's of all five primal qubits is a stabilizer. This is consistent with (5) under the convention that the truncated faces are bonafide dual faces. The same holds, of course, with dual objects.

With this in hand, we are ready to connect the 3D cluster state to the 2D Toric code. We consider the cluster state on a large cube bounded by primal surfaces (see for instance Fig. 1b for a $5 \times 5 \times 5$ lattice) and perform X-measurements on all qubits but those on the very last plane (output Toric code $\mathcal{O}$ in Fig. 1), on which we only measure the dual qubits (red qubits) in X as well. The resulting state is a product of an entangled state on the primal qubits on $\mathcal{O}$ with a random product state of $|\pm\rangle$ on the rest of the cluster — the bulk of the cluster has been 'reset'. By the above discussion, it is stabilized exactly by the Toric code stabilizers up to a sign given by the measurement outcomes. In other words, the state resulting from the measurements in the bulk is an eigenstate of the Toric code Hamiltonian, the eigenvalue being determined by the result of all measurements. This correspondence is central to the realization of topological Measurement-Based Quantum Computation. Note that here, the qubits on the input Toric code $\mathcal{I}$ in Fig. 1b are also measured in the X basis. We don't perform any computation here but just prepare a Toric code from a cluster state. The corresponding plaquette and site have been highlighted in red and black respectively.

Before delving further into this, we note that although this measurement-based procedure for creating a topologically ordered state has been known since the original [15], the idea was recently revived [27] and used to prepare even non-Abelian topological orders experimentally [28].

3 Measurement-Based Quantum Computation in the cluster state

The cluster state being obtained by a finite depth quantum circuit, it is short-range entangled, but in a non-trivial Symmetry Protected Topological phase. Local measurements then realize the gauging to a proper topological order (see also [27] for a general point of view on this), which is the fundamental process enabling MBQC.

3.1 Long-range entanglement through measurements

Information processing in MBQC is performed by teleportation from plane to plane in the cluster. We recall how qubits can be encoded in a Toric code. The fault-tolerant encoding is obtained by punching holes in the code: indeed, to any pair of stabilizers of the same type that is removed from the set of generators of the stabilizer group (a

pair of 'holes') corresponds a pair of operators $\mathcal{X}, \mathcal{Z}$ acting non-trivially on the code and satisfying the commutation relations of the Pauli matrices. In the case of two removed plaquette stabilizers, $\mathcal{X}$ is given by the product of X operators along a string extending from one hole to the other while $\mathcal{Z}$ is a loop winding once around the plaquette, see Fig. 2. Since closed string operators act trivially on the code, all strings extending from one hole to the other and all strings winding around the hole are equal when restricted to the code space. Since $\mathcal{X}, \mathcal{Z}$ are in the center of the stabilizer group, they are bonafide operators acting on the code space.

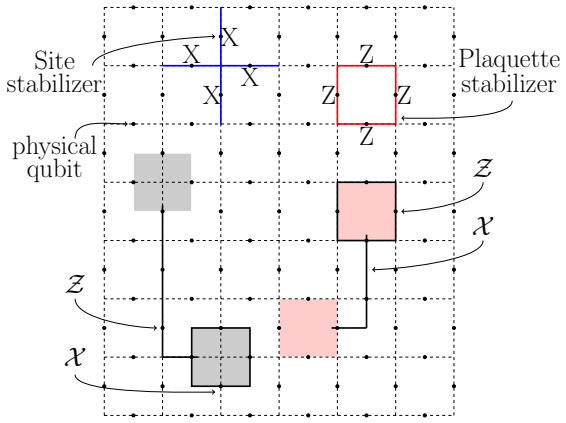


Figure 2: The Toric code. The physical qubits sit on the graph edges, represented by small black dots. At the top are defined two stabilizers of different types: plaquette Z and site X. Logical qubits are encoded in pairs of missing stabilizers (blue and red squares at the bottom), they can be of two types due to the duality of the surface code. A logical Pauli operator correspond to a product of Pauli operators on physical qubits along a string either extending from one hole to the other or winding around one hole.

This picture motivates the following. We consider a large cluster with two distinguished opposite planes, which we call the 'in' and 'out' planes. The qubits in the cluster are divided into three sets

$$\mathcal{C} = \mathcal{I} \cup \mathcal{B} \cup \mathcal{O}$$

corresponding to the 'in' plane $\mathcal{I}$, the 'out' plane $\mathcal{O}$ and the bulk $\mathcal{B}$. Let c^Z be a primal surface that is a rectangle having one side on each of $\mathcal{I}, \mathcal{O}$ while the other two extend in its bulk, see Fig. 3. We modify the measurement procedure described in Section 2, firstly by not measuring the qubits in $\mathcal{I}, \mathcal{O}$, and secondly by measuring the qubits on the inner boundary of the surface, $\partial c_2 \cap \mathcal{B}$, in the Z basis while all the other qubits in the bulk, including the ones inside the surface c^Z, are measured in X. Since $K(c^Z)$ is a stabilizer that commutes with the measurements, we conclude that the final state is an eigenstate of $\mathcal{Z}_{\text{in}} \otimes \mathcal{Z}_{\text{out}}$ where $\mathcal{Z}_{\text{in}} = Z(\partial c^Z \cap \mathcal{I})$ and similarly for $\mathcal{Z}_{\text{out}}$. Furthermore, let $\bar{c}^X$ be a boundaryless cylinder surrounding one of the edges of the rectangle extending from the initial to the final plane, see Fig. 3. The corresponding stabilizer $K(\bar{c}^X)$ is a product of X operators on the cylinder that commute with the X measurements in the bulk. Note that this surface cannot be contracted as it wraps around a line of qubits measured in Z. The state after the measurements in the bulk is also an eigenstate of $\mathcal{X}_{\text{in}} \otimes \mathcal{X}_{\text{out}}$. Together, we have therefore obtained an *encoded Bell pair* across the cluster. The fundamental theorem of topological MBQC, see Theorem 1 below, states that an appropriate choice of measurements, which correspond to primal and dual surfaces, implements a unitary between the encoded initial state and the encoded final state.

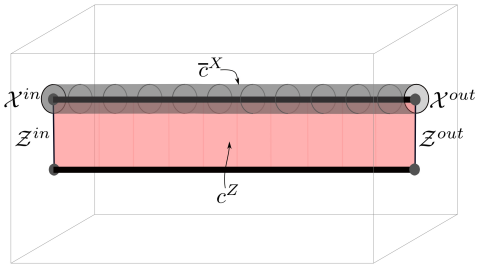


Figure 3: The surfaces that correspond to the identity gate. The surface c^Z is primal with black qubits on its boundary, while $\bar{c}^X$ is dual and has no boundaries at all. The black qubits in $\partial c^Z \cap \mathcal{B}$ are measured in Z, while all the other qubits in $\mathcal{B}$ are measured in X. After the measurements in the bulk, the state on $\mathcal{I} \cup \mathcal{O}$ is an encoded Bell pair.

3.2 A dynamical point of view

This 'global' picture can also be described in local, 'time dependent' way as follows. Initially, logical qubits are encoded in an eigenstate $|\varphi_{\text{in}}\rangle$ of a Toric code. Then the physical qubits in the Toric code are entangled to a cluster state via CZ gates so that the qubits in the Toric code become the black qubits on $\mathcal{I}$. The state obtained by this coupling is not exactly a cluster state because the physical qubits in the Toric code were not individually in the $|+\rangle$ state.

A measurement in $\mathcal{B}$ corresponds to the projector $P_{\mathcal{B}}(\underline{s}, \underline{\sigma}) = \bigotimes_{q\in\mathcal{B}} \left(\frac{1+s_q\sigma_q}{\sqrt{2}}\right)$, where $\underline{\sigma} = \{\sigma_q : q \in \mathcal{B}\}$ is a vector of (possibly rotated) Pauli operators and $\underline{s} = \{s_q : q \in \mathcal{B}\}$ is the $\{0,1\}$-valued vector containing the measurement outcomes. The resulting state on the entire cluster $\mathcal{C}$ is given by

$$|\psi\rangle = P_{\mathcal{B}}(\underline{s}, \underline{\sigma}) \prod_{\langle r,b\rangle} CZ_{r,b}\Big(|\varphi_{\text{in}}\rangle \bigotimes_{q\in\mathcal{B}\cup\mathcal{O}} |+\rangle_q \Big) \tag{6}$$

In practice, all qubits will be measured in either the X or the Z basis, with a few isolated exceptions measured in the $X-Y$ plane that we shall use for magic state distillation. Here again, a final measurement of the qubits on the initial plane yields a product state between $\mathcal{I}$, $\mathcal{B}$ and $\mathcal{O}$; We denote the latter part, which is a Toric code state, by $|\varphi_{\text{out}}\rangle$.

Given a $P_{\mathcal{B}}(\underline{s}, \underline{\sigma})$, an element $c_2 \in C_2$ (or $\bar{c}_2 \in C_2$) is said to be *compatible* with the measurement if

$$[K(c_2), P_{\mathcal{B}}(\underline{s}, \underline{\sigma})] = 0. \tag{7}$$

Concretely, if

$$\begin{cases} \sigma_q = X_q & \text{whenever} \quad q \in c_2 \\ \sigma_q = Z_q & \text{whenever} \quad q \in \partial c_2 \end{cases}$$

then c_2 is compatible with $P_{\mathcal{B}}(\underline{s}, \underline{\sigma})$ (the same holds for dual 2-chains).

The following theorem, which generalizes [20, 15], establishes the connection between the measurement pattern in the bulk of the cluster and the information processed between the input Toric code to an output layer. Its proof follows a reasoning similar to the two-dimensional argument of Theorem 1 in [17], but adapted to the 3D case. The idea is simple: given a measurement pattern, one searches for surfaces compatible with it. They in turn correspond to logical qubits on the input and output layers. By simply following the surfaces from the input qubits to the output qubits, one reads off transformation laws mapping elements of the real Pauli group to themselves, namely: to a set of compatible

surfaces corresponds a real Clifford gate. We recall that Clifford gates are all unitary elements of the normalizer of the Pauli group. The *real Clifford gates* are the unitary elements of the normalizer of the real Pauli group, namely of the group generated by X and Z only.

The setting is as follows. The initial and final Toric codes come with missing stabilizers and corresponding logical operators $\{\mathcal{X}_i^{\text{in}}, \mathcal{Z}_i^{\text{in}} : i = 1, \dots n\}$ and $\{\mathcal{X}_i^{\text{out}}, \mathcal{Z}_i^{\text{out}} : i = 1, \dots n\}$. The initial state $|\varphi_{\text{in}}\rangle$ is now entangled with the rest of the cluster. Measurements in the bulk are given and described by the projection $P_{\mathcal{B}}(\underline{s}, \underline{\sigma})$, yielding the state (6). A final measurement is carried out on the initial plane, in the X basis, which we describe by the projector $P_{\mathcal{I}}(\underline{t}, \underline{X})$. Slightly abusing notation, we denote $|\varphi_{\text{out}}\rangle$ both the state of the full cluster and the state on the final Toric code. The assumptions of the theorem ensure not only that surfaces are compatible with the measurements, but also with the encoded qubits of the initial and final codes.

Theorem 1 (Fundamental Theorem of Topological MBQC)**.** *Let U be a real Clifford gate. Assume that for each $i \in \{1, \dots, n\}$, there are $c_i^X, c_i^Z \in C_2$ and $\bar{c}_i^X, \bar{c}_i^Z \in \overline{C}_2$ that are all compatible with the measurements $P_{\mathcal{B}}(\underline{s}, \underline{\sigma})$ and such that $\partial c_i^X \cap \mathcal{I} = \emptyset$ and $\bar{c}_i^Z \cap \mathcal{I} = \emptyset$ and*

$$X(\bar{c}_i^X \cap \mathcal{I}) = \mathcal{X}_i^{\text{in}} \tag{8}$$

$$X(\bar{c}_i^X \cap \mathcal{O}) Z(\partial c_i^X \cap \mathcal{O}) = U \mathcal{X}_i^{\text{out}} U^\dagger \tag{9}$$

and

$$Z(\partial c_i^Z \cap \mathcal{I}) = \mathcal{Z}_i^{\text{in}} \tag{10}$$

$$X(\bar{c}_i^Z \cap \mathcal{O}) Z(\partial c_i^Z \cap \mathcal{O}) = U \mathcal{Z}_i^{\text{out}} U^\dagger. \tag{11}$$

Let $|\varphi_{\text{out}}\rangle = P_{\mathcal{I}}(\underline{t}, \underline{X}) |\psi\rangle$. Then

$$|\varphi_{\text{out}}\rangle = U U_P |\varphi_{\text{in}}\rangle ,$$

where $U_P = \mathrm{e}^{\mathrm{i}\eta} \bigotimes_{i=1}^n (\mathcal{X}_i^{\text{out}})^{\underline{s}_i^Z} (\mathcal{Z}_i^{\text{out}})^{\underline{t}_i^X + \underline{s}_i^X}$. Here, s_i^Z are the measurement outcomes in c_i^Z and similarly for the other exponents.

We point out that we assumed the initial and final planes $\mathcal{I}$ and $\mathcal{O}$ that define the input and output Toric codes are made of primal edges (black qubits on Fig. 1) only, which implies that

$$\partial \bar{c}_i^{X,Z} \cap (\mathcal{I} \cup \mathcal{O}) = \emptyset, \qquad c_i^{X,Z} \cap (\mathcal{I} \cup \mathcal{O}) = \emptyset. \tag{12}$$

From the cluster point of view, the red qubits belonging to these planes are part of the measured bulk $\mathcal{B}$ and are therefore measured according to the measurement pattern.

Proof. We first prove the theorem in the case where all logical qubits of the input are in the logical + state, namely

$$\mathcal{X}_i^{\text{in}} \left|\varphi_{\text{in}}^+\right\rangle = \left|\varphi_{\text{in}}^+\right\rangle \tag{13}$$

for all $i = 1, \dots, n$. We denote $|\psi(+)\rangle$ the corresponding state (6) after the bulk measurement. Since (8) implies that $X(\bar{c}_i^X \cap \mathcal{I}) \left|\varphi_{\text{in}}^+\right\rangle = \mathcal{X}_i^{\text{in}} \left|\varphi_{\text{in}}^+\right\rangle = \left|\varphi_{\text{in}}^+\right\rangle$ and the other physical qubits are eigenstate of the local X Pauli operators, we have that

$$X(c_i^X) X(\bar{c}_i^X) \Big(\left|\varphi_{\text{in}}^+\right\rangle \bigotimes_{q \in \mathcal{B} \cup \mathcal{O}} |+\rangle_q \Big) = \Big(\left|\varphi_{\text{in}}^+\right\rangle \bigotimes_{q \in \mathcal{B} \cup \mathcal{O}} |+\rangle_q \Big)$$

We now use (3) to commute the X operators through the CZ in $|\psi(+)\rangle$, resulting in

$$\prod_{\langle r,b\rangle} CZ_{r,b} X(c_i^X) X(\bar{c}_i^X) = K(c_i^X) K(\bar{c}_i^X) \prod_{\langle r,b\rangle} CZ_{r,b} \tag{14}$$

Since the surfaces are compatible with the measurements (7), the operators commute with $P_{\mathcal{B}}(\underline{s},\underline{\sigma})$ and so

$$\begin{aligned} |\psi(+)\rangle &= K(c_i^X) K(\bar{c}_i^X) |\psi(+)\rangle \\ &= (-1)^{\underline{s}_i^X} X(\bar{c}_i^X \cap \mathcal{I}) X(\bar{c}_i^X \cap \mathcal{O}) Z(\partial c_i^X \cap \mathcal{O}) |\psi(+)\rangle \\ &= (-1)^{\underline{s}_i^X} \mathcal{X}_i^{\text{in}} \otimes (U \mathcal{X}_i^{\text{out}} U^\dagger) |\psi(+)\rangle \end{aligned} \tag{15}$$

where we used $\partial c_i^X \cap \mathcal{I} = \emptyset$ and (8,9) in the last equality. Here, $(-1)^{\underline{s}_i^X}$ is the result of all measurements on the ith surface. After the measurements in the initial plane, the state is given by

$$P_{\mathcal{I}}(\underline{t},\underline{X}) |\psi(+)\rangle = (-1)^{\underline{t}^X + \underline{s}^X} |\underline{t}_i\rangle \otimes |\underline{s}_i\rangle \otimes U \mathcal{X}_i^{\text{out}} U^\dagger \left|\varphi_{\text{out}}^+\right\rangle \tag{16}$$

Since this holds for all $i = 1, \ldots, n$, we conclude that $\left|\varphi_{\text{out}}^+\right\rangle$ is an eigenstate of $U \mathcal{X}_i^{\text{out}} U^\dagger$ for all i, namely

$$U^\dagger \left|\varphi_{\text{out}}^+\right\rangle = (-1)^{\underline{t}_i^X + \underline{s}_i^X} \mathcal{X}_i^{\text{out}} U^\dagger \left|\varphi_{\text{out}}^+\right\rangle . \tag{17}$$

In other words, $U^\dagger \left|\varphi_{\text{out}}^+\right\rangle$ is, up to a phase $\mathrm{e}^{\mathrm{i}\eta(+)}$, in the logical $\pm$ state for each i. The theorem follows in this special case by comparing this with (13) and observing that the necessity or not of a spin flip $\mathcal{Z}_i^{\text{out}}$ is determined by the corresponding eigenvalue.

We now consider the case of an input state where the qubits are all in the logical Z basis, namely

$$\mathcal{Z}_i^{\text{in}} |\varphi_{\text{in}}^{\underline{z}}\rangle = (-1)^{z_i} |\varphi_{\text{in}}^{\underline{z}}\rangle \tag{18}$$

If $P(\underline{z}) = |\varphi_{\text{in}}^{\underline{z}}\rangle\langle\varphi_{\text{in}}^{\underline{z}}|$, then $|\varphi_{\text{in}}^{\underline{z}}\rangle = 2^{n/2} P(\underline{z}) \left|\varphi_{\text{in}}^+\right\rangle$. Moreover, $P(\underline{z})$ commutes with any CZ gate, and so the state (6) after the measurements in the bulk is given by

$$|\psi(\underline{z})\rangle = 2^{n/2} P(\underline{z}) |\psi(+)\rangle \tag{19}$$

We repeat the argument above, but with surfaces $c_i^Z, \bar{c}_i^Z$, which yields (14) for the corresponding stabilizer $K(c_i^Z) K(\bar{c}_i^Z)$. Since $\bar{c}_i^Z \cap \mathcal{I} = \emptyset$, the only operator acting on the initial surface is $Z(\partial c_i^Z \cap \mathcal{I})$, which commutes with $P(\underline{z})$. Hence,

$$\begin{aligned} |\psi(\underline{z})\rangle &= 2^{n/2} (-1)^{\underline{s}_i^Z} P(\underline{z}) \mathcal{Z}_i^{\text{in}} X(\bar{c}_i^Z \cap \mathcal{O}) Z(\partial c_i^Z \cap \mathcal{O}) |\psi(+)\rangle \\ &= (-1)^{\underline{s}_i^Z + z_i} X(\bar{c}_i^Z \cap \mathcal{O}) Z(\partial c_i^Z \cap \mathcal{O}) |\psi(\underline{z})\rangle . \end{aligned}$$

The state on the final plane is not entangled with the the qubits on the initial plane so that the measurement on $\mathcal{I}$ does not affect it. Using (11), we obtain the equation

$$U^\dagger |\varphi_{\text{out}}^{\underline{z}}\rangle = (-1)^{\underline{s}_i^Z + z_i} \mathcal{Z}_i^{\text{out}} U^\dagger |\varphi_{\text{out}}^{\underline{z}}\rangle . \tag{20}$$

Hence $U^\dagger |\varphi_{\text{out}}^{\underline{z}}\rangle$ is again a computational basis state with some logical qubits that are flipped as compared with the initial state (18). This is again the statement of the theorem in these cases, with a phase $\mathrm{e}^{\mathrm{i}\eta(\underline{z})}$. Note that here z_i is not due to measurements but is given by the initial state of the logical qubit i; Since it appears both in (20) and (18), the necessity of $\mathcal{X}_i^{\text{out}}$ is determined by $\underline{s}_i^Z$.

It remains to decompose $\left|\varphi_{\text{out}}^+\right\rangle$ in the basis $|\varphi_{\text{out}}^{\underline{z}}\rangle$ to conclude that all phases $\eta(\underline{z})$ are equal to $\eta(+)$, see [17]. This determines U_P (up to a global phase) on the basis $|\varphi_{\text{in}}^{\underline{z}}\rangle$, and thus by linearity on any input vector $|\varphi_{\text{in}}\rangle$, concluding the proof. □

Note that U is completely determined by the measurement in $\mathcal{B}$ and the existence of primal and dual surfaces that 'connect' the Pauli operators in $\mathcal{I}$ and $\mathcal{O}$: Indeed, if such surfaces exist, after the measurements in $\mathcal{I}$, the state $|\varphi_{\text{out}}\rangle$ is defined uniquely and so is U. Therefore, in order to implement a given Clifford gate U, one must find a suitable measurement pattern: Since this pattern is not unique, there is the possibility of optimization to reduce the size of the cluster $\mathcal{B}$. This will be the main focus of Section 6 for the case of multiple CNOT gates.

3.3 Illustrative examples: CSS gates

Here we briefly review the measurement patterns proposed in [15] to implement the identity and the CNOT (see Fig. 3 and 4). With these, we can implement what we call *CSS gates*, which map a logical $\mathcal{X}_i^{\text{in}}$ to a product of logical $\mathcal{X}_i^{\text{out}}$ and the same for $\mathcal{Z}$, in reference to the CSS codes [2].

For the identity, a side view of the cluster state is presented in Fig. 3 where the input Toric code is on the left and the output plane is on the right. The surfaces of Theorem 1 are: the primal c^Z is plotted in gray and the dual $\bar{c}^X$ is represented in light red, while $c^X, \bar{c}^Z = \emptyset$. We immediately see that (8)—(11) are satisfied for $U = \mathbb{I}$.

The CNOT gate is slightly more complex as it entangles two logical qubits. In Fig. 4, we represent only the lines of Z measurements: in black along primal boundaries and in red along dual boundaries. These lines form a non-trivial knot in three dimensions. The surfaces are here c_1^Z, c_2^Z and $\bar{c}_1^X, \bar{c}_2^X$ and they connect the qubits on $\mathcal{I}$ to those on $\mathcal{O}$ as follows:

$$\begin{aligned}
&\bar{c}_1^X : \mathcal{X}_1^{\text{in}} \to \mathcal{X}_1^{\text{out}}\mathcal{X}_2^{\text{out}} && \text{(its boundary in } \mathcal{B} \text{ is the red line, see Fig. 4)}\\
&\bar{c}_2^X : \mathcal{X}_2^{\text{in}} \to \mathcal{X}_2^{\text{out}} &&\\
&c_1^Z : \mathcal{Z}_1^{\text{in}} \to \mathcal{Z}_1^{\text{out}} && \text{(wraps around the 'top' part of the red line)}\\
&c_2^Z : \mathcal{Z}_1^{\text{in}} \to \mathcal{Z}_1^{\text{out}}\mathcal{Z}_2^{\text{out}} && \text{(wraps around 'lower right' part of the red line)}
\end{aligned} \tag{21}$$

Comparing this with the statement of the theorem yields $U = \text{CNOT}$.

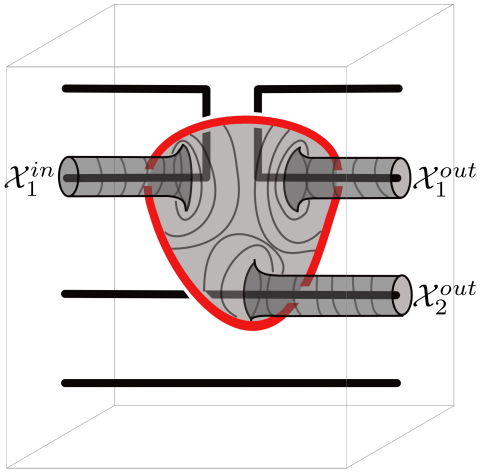


Figure 4: The input and output qubits and the measurement lines that correspond to the CNOT gate. The gray surface sketch the 'tubes' making up the $\bar{c}_1^X$ surface, while the red line is its boundary, which is made up of dual qubits.

3.4 Comparison with topological MBQC on surface codes

In this work, we focus on braiding hole defects to implement some computation on the Toric code. Another approach to topological MBQC (or topological circuit based QC in general) is to use one patch of surface code (same stabilizers as the Toric code but with open boundary conditions) to encode one logical qubit [23]. In circuit-based computation, a transversal logical CNOT gate can be implemented between two patches by applying a physical CNOT between a physical qubit in the first patch and its corresponding qubit in the second patch. A downside to this implementation is that every qubit in a patch needs to interact with another in the other patch, which could be practically hard to implement in an architecture where the patches were side by side. To resolve this issue, the notion of lattice surgery was introduced [29]. Two patches are stitched together on one of their boundary; their logical operators merge. The patches are then separated and regain their individual logical operators, but they are now logically entangled. This procedure can be translated into an MBQC scheme quite naturally [30].

In both frameworks, the Hadamard can be achieved topologically, and magic state distillation is used for universality. For instance, in circuit based, a transversal Hadamard is applied on every physical qubit of the patch followed by the patch rotation to recover the initial stabilizer structure[29]. Patch rotation can be achieved by expanding the surface code lattice and applying local measurements. To translate this Hadamard into MBQC, [30] proposes first to shift part of the cluster state corresponding to one patch, so that the output surface code after measurements corresponds to the input one up to a transversal Hadamard. The rotation is implemented by connecting the output surface code to another cluster and performing local measurements in X or Z basis.

The Hadamard defect we propose here uses a similar principle but in the 'hole braiding' framework, where there is only one patch of cluster state for the whole circuit. The defect can be embedded locally within the cluster lattice, creating a fusion surface across which the gate is realized, thereby providing a particularly simple topological implementation.

Promising physical implementations of surface code computation have been proposed and achieved for different architectures (for instance, photonic hardware [31, 32, 33] and superconducting processor [34, 35, 36] for a non-exhaustive list). Indeed, storing information on surface code patches has been shown to be less resource consuming than holes in the Toric code [37]. For now, only relatively small surface codes or cluster states can be tested experimentally. Important scaling up would be necessary to implement a hole-braiding computation on a hardware.

4 Fault-tolerant computation

To achieve universality in Quantum computing, one can use the group of logical gates generated by the Clifford group and the $\pi/8$ gate T. In this section, we show that the entire Clifford group can be implemented fault-tolerantly. Precisely, there are measurement patterns and corresponding compatible surfaces that implement any Clifford unitary in the sense of Theorem 1. Measurement patterns for the identity gate and the CNOT have been presented in Sec. 3.3. In this framework, the computational power is quite limited, not even achieving the whole real Clifford group. To improve this, we propose a new implementation of the Hadamard gate H, see Fig. 5, which in turn will allow building the phase gate S, thus leading to the full Clifford group. Finally, full universality is obtained by adding the T gate, which can be implemented through magic state injection, which will be discussed in Sec. 5.

4.1 Defect in the lattice: the Hadamard gate

We will first see that a Hadamard gate on all logical qubits is automatically implemented between two planes separated only by half-odd number of unit cells. Of course, such a global unitary is of no interest, and we would like only some chosen logical qubits to undergo a Hadamard, not all of them. So we would like the lines of some qubits to travel $n + \frac{1}{2}$ unit cells and others only n. As we shall see, this can be achieved by introducing a *dislocation in the physical lattice* underlying the cluster state.

4.1.1 Primal vs Dual planes: a global Hadamard

The natural setting of Theorem 1 is where there is an integer number of unit cells between the initial and final planes. We can extend the cluster by one half of a unit cell in the 'time' direction, i.e. adding a dual layer after the output plane in Fig. 1b. By measuring the qubit on the former output plane in the X basis, the arguments of Section 2 yield that the former output Toric code is mapped to its dual on the new final plane, see also the foliation principle [38]. Namely, all the site stabilizers on the black Toric code are transformed into plaquette stabilizers on the red one, and vise-versa. Similarly, site holes implementing primal logical qubits are mapped to plaquette holes, and so the logical operators on the final surface code are mapped to their dual. It follows that the claim of the theorem now holds with U replaced with HU, where $H = \bigotimes_{i=1}^{n} H_i$. In other words: extending the measurement pattern by half a unit cell corresponds to carrying out one *global* Hadamard gate. In fact, this is exactly what is used in [30] to implement a logical Hadamard on a foliated surface code.

However, as pointed out above, this is insufficient for our setting where we use only one global cluster. Therefore, our proposal goes beyond that by embedding a lattice defect in the bulk of the cluster. This creates what we shall refer to as a fusion surface that implements a *local* Hadamard gate. It is local in the sense that only those correlation surfaces that cross the fusion surface undergo the gate, but not those that pass around it.

4.1.2 A line defect

Consider the dislocation defect in Fig. 5. The lattice is twisted so that the unit cell on the top right (highlighted in yellow) is displaced compared to its normal position. Its left bottom corner now sits on the center of another unit cell. Some of its edges now also correspond to the faces of the other cell. Therefore, trying to bicolor all the qubits from the orange plane in the middle, all the way around the defect and back to the orange plane leads to a coloring conflict.

With this, the lattice of qubits is no longer bipartite, and the duality defined and used above is broken. A connected primal 1-chain starting on one side of the defect and winding around it once ends up displaced so that it appears as a dual 1-chain. Equivalently, there is a discrepancy emerging between two connected qubit lines traveling from the initial plane on different sides of the defect: if one of them extends for n unit cells and past the defect the other travels $n \pm \frac{1}{2}$. This is illustrated in Fig. 5. Two lines are initially of the same type, both made of Z-measured black edge qubits and accounting for site holes on the input Toric code $\mathcal{I}$ (left green plane), see sick black thick lines in Fig. 5. After passing the defect, they end up differently on the right green plane $\mathcal{O}$: The line having passed 'behind' the defect still corresponds to black edges and a site hole on $\mathcal{O}$, while the other one accounts now for a plaquette hole, as it sits on Z-measured red faces. In this picture, we assume that all the other surrounding qubits are measured in X except the ones in $\mathcal{O}$.

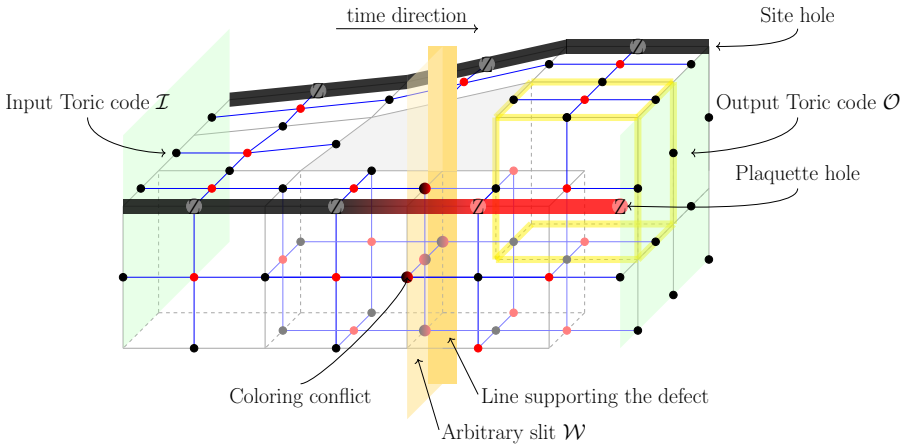


Figure 5: Hadamard defect in lattice at the unit cell level. For clarity, only the qubits on the external faces of the unit cells have been represented, except for 2 unit cells at the front where all qubits are shown. Every cell that is represented by a white cube (potentially distorted) is full: it has a qubit on every face and on every edge. On the output Toric code plane, the red qubits inside the plaquettes that are measured in X are not shown, only the one measure in X and that implements the plaquette hole is displayed. Nearest neighbor connections are represented by blue lines. Some qubits and connections are missing in the middle of the cluster to allow for the dislocation defect.

This defect can be embedded in the 3D lattice, by extending it into a line. It is topological in the sense that the lattice of qubits is locally completely normal. The cluster state can still be built using Eq. 1, and its stabilizers are still defined by $K(a) = X_a \bigotimes_{b\in\partial a} Z_b$, with ∂a the nearest neighbors of a. Except for the qubits sitting on the boundary of the defect for which some of their nearest neighbors have been removed, they all still have 4 nearest neighbors and the graph is well defined. The defect can only be detected by winding around it. Furthermore, the input and output Toric code are not geometrically impacted by the defect; in Fig. 5, they are made of a square lattice with black qubits on its edges.

Theorem 1 still applies in this new configuration. We consider a defect line forming a (large) closed loop inside the bulk and we let $\mathcal{W}$ be a surface supported by that loop, see Figure 6. In the homological description introduced in Section 2, $\mathcal{W}$ is an internal slit that must be removed from the cell complex. The slit lifts any discrepancy. The duality is completely recovered, with a well-defined final plane $\mathcal{O}$ which we again assume to be a primal surface, namely (12) holds true. It is important to note that $\mathcal{W}$ is not a domain wall as described in [18, 19] since the lattice shows no defect on $\mathcal{W}$ but only on its boundary. In particular, any surface as described above can be chosen.

The cluster state is defined as before by the graph of nearest-neighbors and so surfaces remain associated with stabilizers, *even across the slit*. From the point of view of homology, however, an 'incoming' surface $c \in C_2$ must stop on $\mathcal{W}$ and be connected to an outgoing surface $\bar{c} \in \overline{C}_2$ 'emerging' from $\mathcal{W}$. These are going to be such that $\partial c \cap \mathcal{W} = \partial \bar{c} \cap \mathcal{W}$, illustrating again the ambiguity of the nature of the slit (note the slight abuse here: we extend $c, \bar{c}$ to $\mathcal{W}$). If we introduce the *fusion* operator $\boxplus_{\mathcal{W}}$ in this situation, which is so that $c \boxplus_{\mathcal{W}} \bar{c}$ is one surface without boundary on $\mathcal{W}$, the corresponding stabilizer is simply $K(c \boxplus_{\mathcal{W}} \bar{c})$ and the surface is compatible with the measurements. Such surface $c_1^Z \boxplus_{\mathcal{W}} \bar{c}_1^Z$, is displayed in Fig. 6, the fusion happens on the purple line when the two surfaces meet on the slit $\mathcal{W}$. In order to have the same type of logical qubits for all (site holes in the Toric code), we can braid the red Z-measurement lines coming from the slit with black Z-measurement lines, as shown in Fig. 6. This only performs an identity gate between

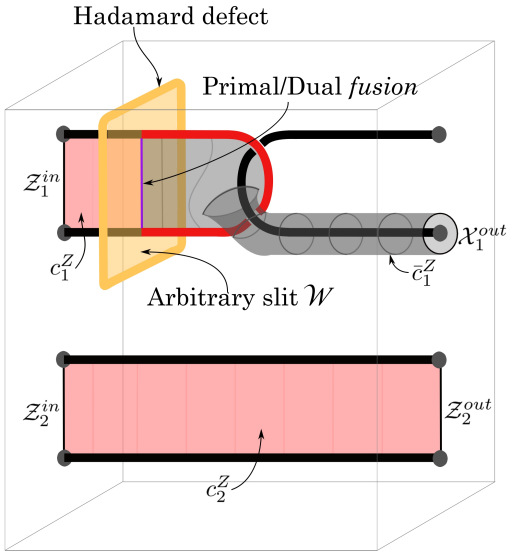


Figure 6: Hadamard defect and arbitrary virtual boundary connecting primal and dual surfaces. Thick black lines correspond to Z measurements on black qubits. Thick red lines to Z measurements on red qubits.

the plaquette-missing logical qubit after the slit and the site-missing logical qubit on the output plane.

It remains to observe that

$$\mathcal{Z}^{\text{in}} \to \mathcal{X}^{\text{out}}, \qquad \mathcal{X}^{\text{in}} \to \mathcal{Z}^{\text{out}},$$

to conclude with Theorem 1 that $U = H_1$ indeed.

This scheme is fault-tolerant provided the Z-measurement lines remain at a certain distance of the line defect and the circumference of the defect line itself is large enough. Locally the cluster unit cells are preserved and so error correction can be performed as usual, the Hadamard defect being interpreted merely as an internal cluster boundary.

4.1.3 Comparison with twists and domain walls

A parallel can be drawn between the defect in the 3D lattice just discussed and the twist introduced in [18] on the one hand, or the domain walls of [19], both in the 2D lattice of the Toric code. In all cases, the defect breaks the global duality of the lattice, introducing a map between the electric and magnetic sectors. In [18, 19], the analysis is from the point of view of the anyon theory. In general, lattice defects allow for anyon transmutation and impact anyon braiding. In the case of the Toric code, braiding an electric charge around a twist maps it to a magnetic one and vise-versa, which reduces the number of anyonic superselection sectors. The computational implementation of Bombin's lattice defects proposed in [39] is different from ours in that it ultimately uses braiding of twists around each other (see also [40] for a more complete discussion), while we braid a twist around holes. The hybrid scheme proposed in [41] braids Bombin's twist with holes in a planar setup. However, our defect is microscopically quite different: Rather than being purely spatial, it distorts the space-time lattice. In fact, the shift by one half time step induced by the defect is what realizes the gate, 'changing the color' of measurement lines. In the homological picture, this amounts to the transformation of a primal surface into a

dual one. Furthermore, locally, the lattice on which is supported the measurement line is untouched, the local stabilizers are preserved. The measurement lines travel only half the time step more than the one that does not braid with the defect (see Fig. 5), while in [41] the hole crosses a line where the surrounding stabilizers have to be modified.

4.2 Rebit encoding and complex S gate

In our arsenal, we now have the logical qubit creation and measurement in the X and Z bases [15], the CNOT and Hadamard gates. Thus, the accessible qubit states are, in fact, rebits, namely their density matrix ρ is real

$$\langle v| \rho |w\rangle \in \mathbb{R} \tag{22}$$

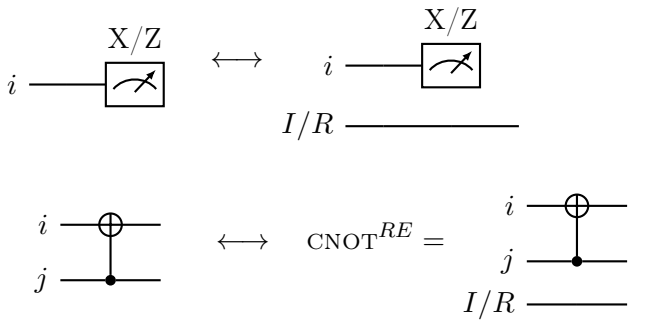
for any $|v\rangle$ and $|w\rangle$ in the computational basis. In order to go further, we use the Rudolph-Grover rebit encoding, see [42], which uses the simple observation that any complex state

$$|\psi\rangle = \sum_{\mathbf{v}\in\mathbb{Z}_2^n} r_{\mathbf{v}} e^{i\theta_{\mathbf{v}}} |\mathbf{v}\rangle$$

of n qubits can be encoded in $n+1$ rebits as follows

$$\left|\psi^{RE}\right\rangle = \sum_{\mathbf{v}\in\mathbb{Z}_2^n} r_{\mathbf{v}} \cos(\theta_{\mathbf{v}}) |\mathbf{v}\rangle \otimes |R\rangle + r_{\mathbf{v}} \sin(\theta_{\mathbf{v}}) |\mathbf{v}\rangle \otimes |I\rangle . \tag{23}$$

Here we denote $|R\rangle = |0\rangle$ and $|I\rangle = |1\rangle$ the two basis states of the additional rebit. Real gates and X,Z measurements are operations that do not affect the complex structure of the encoded state. Therefore, they can simply be applied without modifying the extra I/R rebit, and can be written schematically:



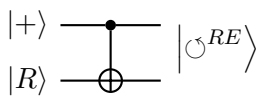
Universal Quantum computing can be achieved with a restricted set of gates, namely Paulis, CNOT, $H_{tot} = \bigotimes_{i=1}^{n+1} H_i$, provided the input is in any rebit state, see [43]. In the fault-tolerant framework presented above, being able to perform the Hadamard on only one rebit is a considerable advantage (no need for extra ancillas). However, universality is not yet achieved since the initial states at our disposal are restricted to eigenstates of real Pauli operators.

4.2.1 Encoding of the Y eigenstate

First of all, the Y eigenstate needs to be encoded: if we denote $|\circlearrowleft\rangle = \frac{|0\rangle+i|1\rangle}{\sqrt{2}}$, then its encoded version, following (23), is

$$\left|\circlearrowleft^{RE}\right\rangle = \frac{|0\rangle \otimes |R\rangle + |1\rangle \otimes |I\rangle}{\sqrt{2}}$$

Hence, the rebit state $\left|\circlearrowleft^{RE}\right\rangle$ is a Bell state and is accessible fault-tolerantly using the following circuit made of CSS gates only:



4.2.2 The complex Clifford gates

We turn to the implementation of the missing Pauli Y and the S gates on a single rebit encoded qubit. For this, we first note that the multiplication by i amounts to a $\pi/2$-rotation in the complex plane and so $\mathrm{Re}(ie^{i\theta}) = -\sin\theta$ while $\mathrm{Im}(ie^{i\theta}) = \cos\theta$ we conclude with (23) that

$$\left|(i\psi)^{RE}\right\rangle = 1 \otimes XZ \left|\psi^{RE}\right\rangle$$

which we shall denote as $i^{RE} = 1 \otimes XZ$ (the last tensor factor always refers to the additional I/R qubit). Since $Y = iXZ$, we conclude that

$$Y_j^{RE} = XZ \otimes XZ$$

In particular, it is achieved by CSS gates, hence implementable topologically.

Finally, we turn to the S gate

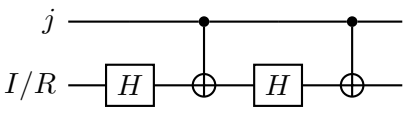
$$S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}, \qquad SXS^\dagger = Y,\ SZS^\dagger = Z.$$

In [15], the implementation of this gate in the framework of topological MBQC required magic state injection, and therefore was not fault-tolerant. In the rebit encoding scheme we propose here, the encoded S^{RE} can be obtained using purely real Clifford gates. With (23) again, we conclude that

$$\left|(S\psi)^{RE}\right\rangle \equiv \mathrm{CNOT}_{j,I/R} CZ_{j,I/R} \left|\psi^{RE}\right\rangle.$$

where the index j refers to the j^{th} rebit. This is a combination of real Clifford gates. Explicitly, we use $CZ_{j,I/R} = H_{I/R}\mathrm{CNOT}_{j,I/R}H_{I/R}$ to obtain the following topological implementation of the S gate

$S^{RE} =$



This representation emphasizes the importance of having a topological implementation of the Hadamard gate. In the above discussions, the rebits can be replaced by encoded rebits on the Toric code. Therefore, we have shown that the complete Clifford group can be implemented topologically, hence in a fault-tolerant fashion, in the framework of MBQC in the cluster state. This is a crucial improvement over previous proposals for topological MBQC on the Toric code, and we will quantify its performance gain in the following Section 4.3.

4.3 Overhead for the Clifford gates

The operational overhead quantifies the resource cost (number of physical qubits or basic operations in our case) of the computation. Reducing the circuit volume is key due to the practical difficulty of producing large scale quantum circuits. On the other hand, scaling up the size of the cluster exponentially reduces the logical error rate. It is therefore necessary to find an optimal trade-off between cluster state volume and correctability. In this section, we consider for any gate G, the ratio between the operational volume and the probability of success of a circuit made of many G gates.

Let V_G be the smallest number of unit cells required to implement the logical gate G. Fault-tolerance requires this volume to be scaled up by a factor λ. Since every elementary cell is built using 24 operations (6 qubit initializations, 12 CZ gates, 6 measurements), the total number of operations required for one G gate is $24V_G\lambda^3$.

Following the threshold and overhead study in [15], the single-gate error $\epsilon_{\text{topo}}(G, \lambda, d)$ is generally dependent on the scaling factor λ and the circumference d of the qubit lines. It is given by

$$\epsilon_{\text{topo}}(G, \lambda, d) = \lambda L_G \left[e^{-4\kappa(d+1)} + 2(d+1)e^{-\kappa(\lambda-d)} \right] \tag{24}$$

where the first term corresponds to the probability of a full cycle of errors surrounding one qubit line, while the second term arises from a line of errors emerging from one qubit and ending on another one. L_G is the total length of the Z-measurement lines implementing the gate. The parameter κ depends on the physical error. For the purpose of comparison, we shall use below the value $\kappa = 0.93$ calculated in [15] for a physical error $p = p_{th}/3$, $p_{th} = 6.7 \times 10^{-3}$.

If a circuit is made of Ω gates G, and we assume that the probabilities of single-gate errors are independent, the success probability of the entire circuit can be approximated by $e^{-\epsilon_{\text{topo}}(G,\lambda,d)\Omega}$. Therefore, the ratio between the total circuit volume and the success rate becomes:

$$R_G(\Omega, \lambda, d) = 24\lambda^3 V_G \Omega e^{\epsilon_{\text{topo}}(G,\lambda,d)\Omega} \tag{25}$$

Note that $R_G(\Omega, \lambda, d)$ can also be interpreted as the effective operational volume for the circuit. In fact, the quantity $e^{\epsilon_{\text{topo}}(G,\lambda,d)\Omega}$ also approximates the average number of trials needed before success. For circuits on which the final results can be classically verified, a failure would be discarded and the process would then be repeated until success.

For the topological gates, $R_G(\Omega, \lambda, d)$ can be easily computed and optimized over the parameters λ, d. In Figure 7, we show the overhead per gate

$$O_G(\Omega) = \Omega^{-1} \min \{R_G(\Omega, \lambda, d) : \lambda, d \geq 1\}, \tag{26}$$

as well as the the volume per gate and the success probability for the optimal $(\lambda_{opti},\ d_{opti})$, as a function of the number of gates Ω for the topological Hadamard, CNOT and S gates. In the S gate case, we can immediately compare with the previous implementation: Thanks to the topological Hadamard and the rebit encoding, the operational volume associated with the S gate is reduced by one full order of magnitude for large Ω, while the success rate reaches 90% around $\Omega = 10^8$ versus 92% for the magic state distillation method. In general, the optimization process is such that, for higher Ω, maximizing the success rate will be preferred over minimal volume, due to the exponential dependency in Ω. The volume and length of the gates used in the numerics are listed in Table 1.

We conclude this section by noting that a similar study has been carried in Ref. [44] but with the 2D color code instead of the Toric code as the base for the 3D cluster. The self-dual structure of the color code allows for a more direct implementation of the Hadamard and

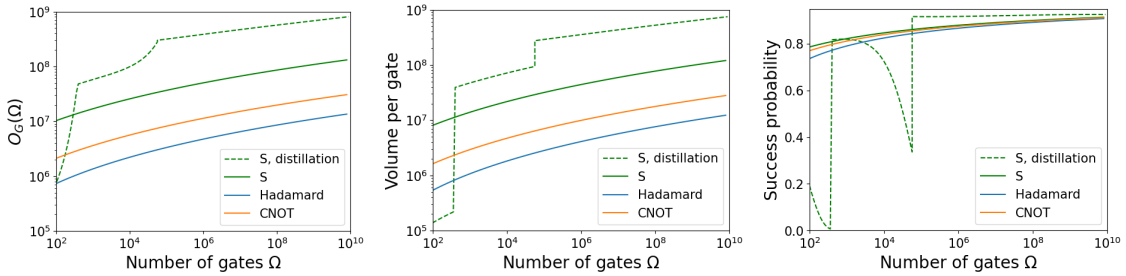


Figure 7: The overhead $O_G(\Omega)$ for the H, CNOT and S gates. From the optimized ratio $O_G(\Omega)$ (left), we compute the operational volume per gate $24V_G\lambda_{opti}^3$ (middle), and the success probability of the entire circuit $e^{\epsilon_{topo}(G,\lambda_{opti},d_{opti})\Omega}$ (right). The comparison is particularly striking for the S gate: The jumps in the dotted line correspond to additional rounds of distillation, and they are absent in the topological implementation (the solid green line) proposed in this work.

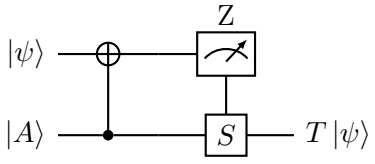
S gates. Indeed, in that framework, the correlation surfaces are more complex and can be compatible with measurements in the complex Y Pauli basis. In this paper, the overhead is defined as the number of physical qubits n or CZ gates N_{CZ} per logical qubit on the 2D base code, as a function of the code distance d. This amounts to finding the optimal arrangements for a maximum of logical qubit holes to fit in a lattice of a given size while preserving the logical distance d. The comparison between the Toric code based MBQC with the color code one is made for different color code lattices: the ratio n/k is about $6.6d^2$ for the Toric code versus $3.9d^2$ and $3.7d^2$ for 4-8-8 and 6-6-6 color codes respectively. No calculation of the logical gate overhead on the whole 3D cluster has been conducted in the mentioned paper, however it would be of interest to compare the performance of both cluster states.

5 Universal Quantum Computation: the T gate

In this final section, we complete the Clifford group with the implementation of the T gate and thus achieving universal MBQC. Compared to the gates presented above, the fault-tolerant version of the T gate remains the most expensive one due to the necessary distillation. Nonetheless, we show that a simple geometric optimization using equivalence transformations of the topological circuitry yields a significant advantage.

5.1 T gate by magic state injection

Universal Quantum computing requires the Clifford group to be extended by the rotation gate $T = \sqrt{S} = e^{\frac{i\pi Z}{4}}$. This gate cannot be implemented fault-tolerantly in MBQC on 3D cluster states. Nonetheless, fault-tolerance can be achieved by magic state distillation – details are left for the appendix. Given an ancilla $|A\rangle = \frac{|0\rangle + e^{i\pi/4}|1\rangle}{\sqrt{2}}$, the T gate can be implemented as follows:



(27)

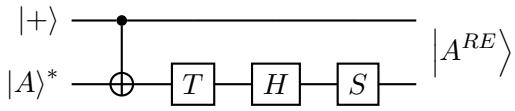
The S gate in the circuit is conditional on the result of the Z measurement and it must carried out on average every second time. A topological $|A\rangle$ state is obtained by measuring just one qubit of a stabilizer on the initial plane in the $X-Y$ plane[15, 20], which is not

a fault-tolerant. In order to recover a high quality topological $|A\rangle$ state, one must resort to magic state distillation, which requires itself 15 ancilla qubits encoded in the Reed-Muller code [45, 46, 47]. Repeating the procedure recursively suppresses the error on $|A\rangle$ exponentially, see [10] and the appendix, and the state can be used as in (27).

The above has all been described in the qubit setting and must now be adapted to our newly proposed rebit setting. The $|A\rangle$ state needs to be encoded in rebits:

$$\left|A^{RE}\right\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle \otimes |R\rangle + |1\rangle \otimes \frac{1}{\sqrt{2}}(|R\rangle + |I\rangle)\right).$$

This state can be obtained using the (complex) magic $|A\rangle^*$ state, S and T gates by:



(28)

thereby involving magic state distillation, as well as the new topological Hadamard. Injecting the resulting $\left|A^{RE}\right\rangle$ on the left of (27) yields the rebit-encoded T gate.

5.2 Overhead for the T gate

Temporarily putting aside the rebit encoding, the T gate requires $|A\rangle$ and $|Y\rangle$ states distillation which is resource intensive. Each round of distillation reduces the noise of the ancilla qubit but drastically increases the number of necessary operations. We summarize this in (29,30), which follows [15] and is detailed in the appendix. If ϵ_l^A is the probability of error on the ancilla after the l^{th} round of distillation and O_l^A is the overhead of the l^{th} round of distillation ($l = 0$ corresponds to no distillation at all), then

$$\begin{aligned}\epsilon_l^A &= 35(\epsilon_{l-1}^A)^3 + \epsilon_{\text{topo}}(A, \lambda_{l-1}, d_{l-1}) \\ O_l^A &= \frac{1}{1 - 15\epsilon_{l-1}^A - \epsilon_{\text{topo}(L_A,\lambda_{l-1},d_{l-1})}}(15O_{l-1}^A + \frac{1705}{512}O_{l-1}^Y + 24V_A\lambda_{l-1}^3)\end{aligned} \tag{29}$$

where $\epsilon_{\text{topo}}(A, \lambda_{l-1}, d_{l-1})$ is the failure rate of the topological Reed-Muller circuit. For the numerics, we shall use $\epsilon_0^A = 0.0134 = 6p = 2p_{th}$ following [15].

Let $l_{\max}$ denote the final round of distillation. The overhead for T is computed as in (25), taking into account the distillations of $|A\rangle$ and $|Y\rangle$ as well as the topological circuitry, yielding

$$O_T(\Omega) = \Omega\left(O_{l_{\max}}^A + (1/2)O_{l_{\max}}^Y + 36(\lambda_{l_{\max}})^3 V_T\right)\exp\left(\left(\epsilon_{l_{\max}}^A + \epsilon_{l_{\max}}^Y + \epsilon_{\text{topo}}(L_T, \lambda_{l_{\max}}, d_{l_{\max}})\right)\Omega\right) \tag{30}$$

We refer to Figure 8 for the resulting numerics. Not surprisingly, because distillation is still required, the improvement over past implementations is slightly less spectacular, but the volume per gate is reduced by one half of an order of magnitude nonetheless. The main contribution to the overhead (29) is the distillation, and in it the leading term is $\lambda_0 V_A$. The 'optimized' curve in the figure is obtained after the fundamental volume involved in the topological implementation of the Reed-Muller code was reduced by more than 40% by a mixture of topological and geometric optimization, see again Table 1. We shall discuss this process in more details in Section 6.

For our proposal which uses a rebit encoding, the T gate uses one extra magic $|A\rangle^*$ state and more topological circuitry, which increases the overhead. The jumps corresponding to

another round of distillation tend to happen for lower Ω as well. All in all, this T gate volume and success probability (orange curve in Figure 8) is roughly similar to the previous implementation (green-dotted curve).

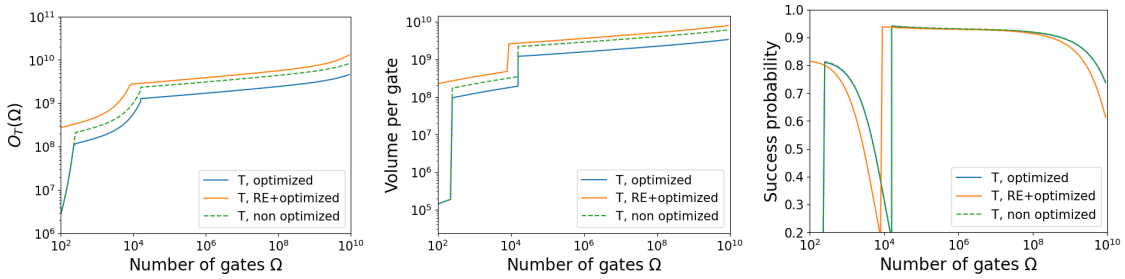


Figure 8: Overhead for the T gate using two different volumes for the $|A\rangle$ state distillation (see Table 1). The overhead is also calculated for the T gate in the rebit encoding. Left: optimized ratio $O_T(\Omega)$, middle: volume per gate, right: success probability.

Gate	Volume	Length
CNOT (compacted)	$V_{\text{CNOT}} = 12$	$L_{\text{CNOT}} = 22$
Hadamard	$V_H = 6$	$L_H = 6$
S (rebit encoding)	$V_S = 48$	$L_S = 48$
A distillation	$V_A = 336$	$L_A = 362$
A distillation (optimized)	$V_A = 192$	$L_A = 288$
Y distillation	$V_Y = 120$	$L_Y = 120$
Y distillation (optimized)	$V_Y = 70$	$L_Y = 105$
CSS circuit for T	$V_T = 2$	$L_T = 3$
CSS circuit for T (RE)	$V_{T,RE} = 89$	$L_{T,RE} = 89$

Table 1: Volume and length of different gates and circuits used in the overhead equations. Except from the Hadamard,the S gate and the optimized circuits, those value are the same as in Ref. [15].

6 Circuit volume optimization

We finally turn to the circuit optimization problem. By this, we mean the question of the reduction of the fundamental volume V_G for a gate G. The fundamental theorem does not claim uniqueness of the measurement pattern for a given G. Beyond the topological invariance — homological surfaces yield the same gate — which is intrinsic to the framework, there are other equivalent measurement patterns that are not homological. In this section, we first formalize the notion of local equivalence between measurement patterns. We then introduce a new set of non-topological transformations that have not yet been proposed in the literature to our knowledge. Finally, we apply these transformations to the Reed-Muller code and quantify the volume reduction thus achieved.

Before delving into further details, we would like to review previous works which achieve volume optimization for braiding computation in the Toric code [48, 49, 50]. Note that these works aim at optimizing the space-time volume of a circuit. The physical setup is a Toric code with hole defects that move in time, and this motion is represented as lines in a (2+1)D diagram. These lines have a direct translation into the 3D cluster state: they correspond to Z-measurement lines. Therefore, the methods presented in [48, 49, 50] are totally applicable to MBQC. First, [48] proposes two algorithms for compacting circuits

topologically (lines can be deformed arbitrarily as long as they are not cut or they do not merge). Beyond topological transformations, [49] uses 'bridges' transformation to optimize CNOT based circuits. The authors show that introducing a bridge between two CNOT, i.e. drawing a red line between two loops such as in Fig. 4, does not change the computation. They apply this transformation to distillation circuits (Steane code and Reed-Muller code) and perform further topological optimization as in [48]. For the Reed-Muller code, they obtain a final volume of 192, a volume 1 in their frame work would correspond to one unit cell for the 3D cluster. More recently, [50] translates the 3D topological representation into circuits in ZX calculus [51, 52]. The authors then rewrite the circuits using ZX-calculus rules and translate them back into the 3D representation. For the Reed-Muller code, they achieve a volume reduction of 65%, or a final volume of 125. Note that the works presented above optimize preexisting diagrams corresponding to some given circuits, while [53] proposes algorithmic tools to translate any quantum circuit into a topological MBQC pattern.

6.1 Equivalence between two measurement patterns

We start with a notion of *local equivalence* of measurement patterns. Importantly, it refers only to the *lines of measurements* and not to the surfaces (and hence qubits) associated with them: It is an equivalence that holds for any surface compatible with the lines under consideration.

For simplicity, we consider a cube B of the cluster: by topological invariance, any local patch of measurements can be isolated within a cube. Its surface contains a number of fixed marked (primal or dual) qubit holes that we denote $\{b_1, \dots, b_m\} \cup \{r_1, \dots, r_n\}$. Two sets of Z-measurement lines are given by two elements $(c_1, \bar{c}_1)$ and $(c_1', \bar{c}_1')$ of $C_1 \times \overline{C_1}$. We consider their equivalence under the condition that the marked points on the boundary are fixed, namely

$$\partial c_1 = \partial c_1' = \{b_1, \dots, b_m\}, \qquad \partial \bar{c}_1 = \partial \bar{c}_1' = \{r_1, \dots, r_n\}.$$

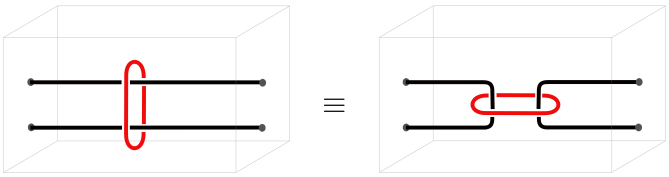
Then $(c_1, \bar{c}_1)$ and $(c_1', \bar{c}_1')$ are equivalent if, for any surface c_2, respectively $\bar{c}_2$, that is compatible with $(c_1, \bar{c}_1)$, there is surface c_2', respectively $\bar{c}_2'$, that is compatible with $(c_1', \bar{c}_1')$ and such that the intersections of c_2 and c_2', respectively $\bar{c}_2$ and $\bar{c}_2'$, with the boundary of the cube B are equal. An example of two equivalent patterns that are topologically inequivalent is given in Figure 9.

6.2 Loop equivalence

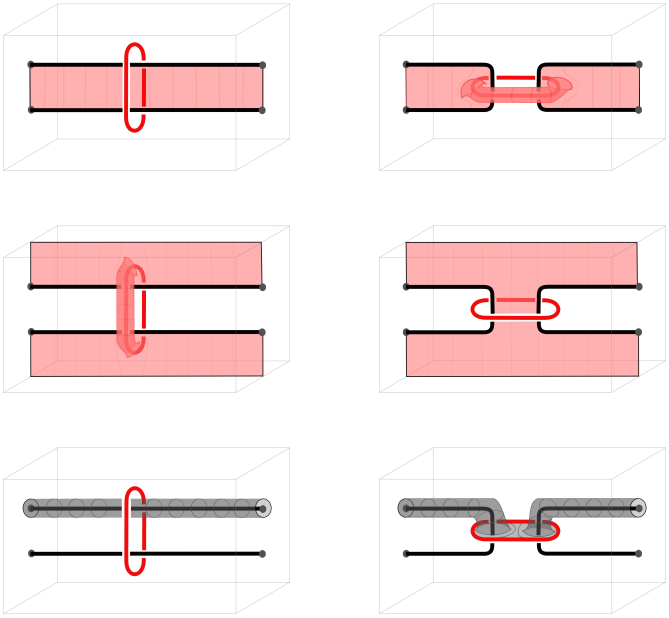
The example in Figure 9 is in fact most relevant as the pattern arises in the CNOT gate. In fact, it has an immediate generalization to a loop surrounding more than two lines: This will be relevant when we consider the Reed-Muller code, which contains five loops each linking eight defect lines, see Figure 13 in the appendix. The equivalence in this general case is displayed in Figure 10.

6.3 Compacting the magic state distillation circuit

We now use the remaining topological equivalence to further compact the Reed-Muller circuit. For this, we again consider Figure 10. The red pattern forms a 'panel' divided into eight compartments. Folding along the horizontal red line, the panel can be wrapped on itself to form a rectangular cuboid. The result is displayed in Figure 11. Not only is this very compact, we also immediately see that the volume is 16 unit cells that are shared



(a) Equivalence of a dual loop around two primal lines.



(b) Examples of corresponding surfaces in the two equivalent measurement patterns.

Figure 9: Two equivalent measurement patterns and examples of surfaces that connect the same operators on the boundary of the cube.

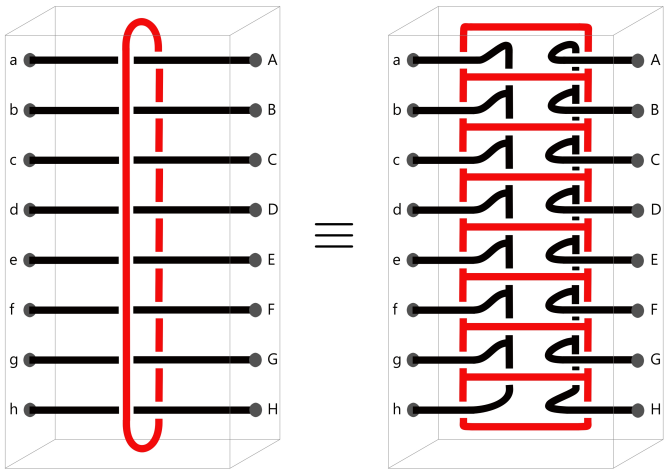


Figure 10: An equivalent pattern for a dual loop around 8 primal lines.

with the other boxes. The Reed-Muller code requires 5 of such boxes to be connected appropriately. The geometry proposed in Figure 13(b) does so in a compact way, resulting in a fundamental volume V_A for the Reed-Muller encoding of 192 unit cells. This is to be compared with the 'naive' circuit of Figure 13(a) in the appendix, which requires 336 unit cells. These are the values used in Table 1. Our optimization compares with the bridge transformations [49], but does not perform as well as [50] which achieves a final volume of 125 for the Reed-Muller encoding. The 'loop-to-box' transformations we present here is a new method that could be combined with previous ones to further improve MBQC in the 3D cluster state.

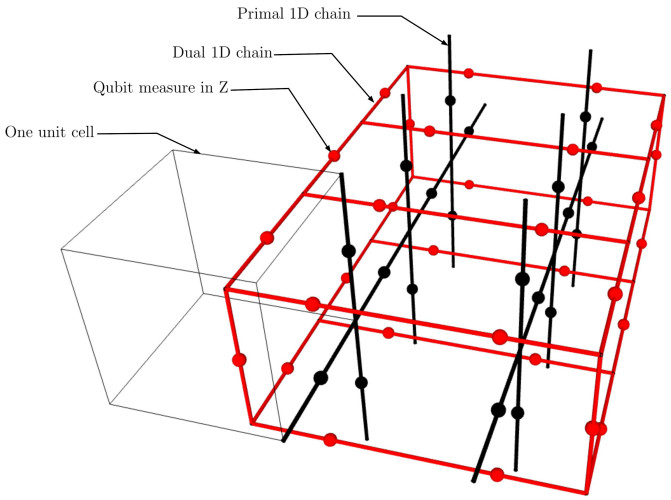


Figure 11: Folding the panel on the right hand side of Figure 10. The figure is at the unit cell scale, a primal unit cell is represented by a cube on the front left corner of the box, while the black and red dots represent the black and red qubits, respectively.

6.4 Algorithmic Circuit Verification

Verifying the equivalence of two measurement patterns by hand is a cumbersome process, as this requires checking every possible connection between non-trivial loops on the boundary. A numerical assistant for this task is glaringly necessary. Problematically, the number of surfaces to check will grow exponentially with the number of Z-measurement lines ending at the boundary, again quickly rendering the problem intractable. However, the goal of these local equivalence transformations is to reduce the volume of a global logical circuit. It is thereby both sufficient and more efficient to verify that the global circuit is generating the correct logical operator *posterior* to applying all local transformations. Predicated on this, we have built a program that takes as an input a measurement pattern of a CSS circuit, as well as a set of logical operators on the boundary, and returns whether respective compatible surfaces connecting these logical operators exist. The specifics of this algorithm are detailed in the Appendix, and its code is available under [21].

Every novel CSS circuit in this paper has been verified numerically using that algorithm. Furthermore, to ensure its correctness with a reasonable degree of certainty, it was previously tested with (1) already established circuits with well-known results,[1] as well as with (2) deliberately wrong constructions to reject the possibility of continuing false positives. As a result, we can argue with high confidence that all novel circuit decompositions

[1]Such as the familiar CNOT gate visualized in Fig. 4, which indeed possesses all compatible surfaces given by Eq. 21.

presented here are correct and significantly compactified variants of previous iterations. Possible research directions arising from this numerical verification are discussed in the following.

7 Conclusion and Discussion

In this paper, we studied the intrinsic relation between a measurement pattern inside the RHG 3D cluster state and the implementation of a gate in a Toric code. We proposed a general theorem using the correlation surfaces supported by the measurement pattern and relating them to the logical real Pauli operators in the Toric code. Thus, we showed the possibility of achieving all the gates in the real Clifford group fault-tolerantly. With that in mind, we introduced a local dislocation defect in the 3D cluster lattice, still compatible with the above theorem, that allows for an intrinsically fault-tolerant implementation of the Hadamard gate. An encoded version of the S gate was then obtained through the Rudolph-Grover encoding, completing the full Clifford group fault-tolerantly. We quantified the performance gains of such structural improvements, reducing the S gate overhead by one order of magnitude, as compared with previous proposals involving magic state distillation.

We showed also significant gains in the T gate overhead. To achieve universal MBQC in 3D cluster states, we still needed to resort to magic state distillation which is resource consuming. By optimizing the geometry of the distillation topological circuit, we managed to reduce the T gate overhead by about half of an order of magnitude.

Finally, in order to verify the relevance of the optimized circuit, we built a program that checks that a given measurement pattern in the cluster state is indeed achieving the desired logical operation.

The following questions remain open:

- The compact realization of the magic state distillation circuit displayed in Figs. 11 and 13 (b) was found by hand. In the continuity of the verifier program, can the search for more compact topological sub-circuits be numerically automated?
- We have introduced a new defect into the RHG lattice, of disclination type, that brings new topological computational power to 3D cluster states. Following the recent classification of defects in the Toric code and the color code [18, 19, 54, 55, 56, 57, 58], can the defects of the RHG lattice be classified? What are their computational powers?
- Moreover, the lattice structure is fundamentally changed by the introduction of this defect. It breaks down the lattice duality, and therefore the notion of chain complex too. In this paper, we deal with it physically by introducing a virtual cut where the lattice ends. Does a more general description of this phenomenon exist at a higher mathematical level?
- Finally, the topological framework studied here deals with correlation surfaces only compatible with X and Z measurements, restricting the possibilities of Quantum Computation. In Ref. [44], the foliation of the 2D color code results in correlation surfaces compatible with Y measurement, thus achieving the full Clifford group too. In the light of recent studies on 3D color codes [59, 60], could a higher dimensional cluster state be built where correlation surfaces would be compatible with non-Pauli measurements?

Acknowledgments

We thank Ugnė Liaubaitė, Ryohei Weil, and Ruben Campos Delgado for their more than helpful discussions.
SB acknowledges the support of NSERC of Canada.
RR is funded by the Humboldt Foundation.
This work was supported by NSERC and the European Commission under the Grant *Foundations of Quantum Computational Advantage*.

References

[1] J. M. Raimond and S. Haroche. "Quantum Computing: Dream Or Nightmare". In 31st Rencontres de Moriond: Dark Matter and Cosmology, Quantum Measurements and Experimental Gravitation. Pages 341–346. Ed. Frontieres (1996). url: https://doi.org/10.1063/1.881512.

[2] A. R. Calderbank and Peter W. Shor. "Good quantum error-correcting codes exist". Physical Review A **54**, 1098–1105 (1996).

[3] A. M. Steane. "Simple quantum error-correcting codes". Physical Review A **54**, 4741–4751 (1996).

[4] D. Gottesman. "Stabilizer codes and quantum error correction". PhD thesis. California Institute of Technology. (1997). url: https://doi.org/10.48550/arXiv.quant-ph/9705052.

[5] D. Aharonov and M. Ben-Or. "Fault-tolerant quantum computation with constant error". In Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing. Page 176–188. STOC '97. Association for Computing Machinery (1997).

[6] P. Aliferis, D. Gottesman, and J. Preskill. "Quantum accuracy threshold for concatenated distance-3 codes". Quantum Information and Computation **6**, 97–165 (2005).

[7] D. Aharonov, A. Kitaev, and J. Preskill. "Fault-tolerant quantum computation with long-range correlated noise". Physical Review Letters **96**, 050504 (2006).

[8] B. Terhal and G. Burkard. "Fault-tolerant quantum computation for local non-markovian noise". Physical Review A **71**, 012336 (2005).

[9] E. Knill. "Quantum computing with realistically noisy devices". Nature **434**, 39–44 (2005).

[10] S. Bravyi and A. Kitaev. "Universal quantum computation with ideal clifford gates and noisy ancillas". Physical Review A **71**, 022316 (2005).

[11] D. Gottesman. "Fault-tolerant quantum computation with constant overhead". Quantum Information and Computing **14**, 1338–1372 (2014).

[12] P. Panteleev and G. Kalachev. "Asymptotically good quantum and locally testable classical ldpc codes". In Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing. Page 375–388. STOC 2022. Association for Computing Machinery (2022).

[13] A. Wills, M.-H. Hsieh, and H. Yamasaki. "Constant-overhead magic state distillation". Nature Physics **21**, 1842–1846 (2025).

[14] A. Guernut and C. Vuillot. "Fault-tolerant constant-depth clifford gates on toric codes" (2024). arXiv:2411.18287.

[15] R. Raussendorf, J. Harrington, and K. Goyal. "Topological fault-tolerance in cluster state quantum computation". New Journal of Physics **9**, 199–199 (2007).

[16] C. Monroe and J. Kim. "Scaling the ion trap quantum processor". Science **339**, 1164–1169 (2013).

[17] R. Raussendorf, D. Browne, and H. Briegel. "Measurement-based quantum computation on cluster states". Physical Review A **68**, 022312 (2003).
[18] H. Bombin. "Topological order with a twist: Ising anyons from an abelian model". Physical Review Letters**105** (2010).
[19] A. Kitaev and L. Kong. "Models for gapped boundaries and domain walls". Communications in Mathematical Physics **313**, 351–373 (2012).
[20] R. Raussendorf, J. Harrington, and K. Goyal. "A fault-tolerant one-way quantum computer". Annals of Physics **321**, 2242–2270 (2006).
[21] M. Schwiering. "Topological-MBQC-Verification". `https://github.com/MarvinSchwi/Topological-MBQC-Verification` (2025).
[22] H. Briegel and R. Raussendorf. "Persistent entanglement in arrays of interacting particles". Physical Review Letters **86**, 910–913 (2001).
[23] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill. "Topological quantum memory". Journal of Mathematical Physics **43**, 4452–4505 (2002).
[24] A. Kitaev. "Fault-tolerant quantum computation by anyons". Annals of Physics **303**, 2–30 (2003).
[25] S. Bachmann. "Local disorder, topological ground state degeneracy and entanglement entropy, and discrete anyons". Reviews in Mathematical Physics **29**, 1750018 (2017).
[26] T. Rakovszky and V. Khemani. "The physics of (good) LDPC Codes II. Product constructions" (2024). arXiv:2402.16831.
[27] N. Tantivasadakarn, R. Thorngren, A. Vishwanath, and R. Verresen. "Long-range entanglement from measuring symmetry-protected topological phases". Physical Review X **14**, 021040 (2024).
[28] M. Iqbal, N. Tantivasadakarn, R. Verresen, S. Campbell, J. Dreiling, C. Figgatt, J. Gaebler, J. Johansen, M. Mills, S. Moses, et al. "Non-abelian topological order and anyons on a trapped-ion processor". Nature **626**, 505–511 (2024).
[29] D. Horsman, A. G. Fowler, S. Devitt, and R. V. Van Meter. "Surface code quantum computing by lattice surgery". New Journal of Physics **14**, 123011 (2012).
[30] D. Herr, A. Paler, S. J. Devitt, and F. Nori. "Lattice surgery on the Raussendorf lattice". Quantum Science and Technology **3**, 035011 (2018).
[31] D. A. Herrera-Martí, A. G. Fowler, D. Jennings, and T. Rudolph. "Photonic implementation for the topological cluster-state quantum computer". Physical Review A **82**, 032332 (2010).
[32] I. Tzitrin, T. Matsuura, R. Alexander, G. Dauphinais, J. E. Bourassa, K. Sabapathy, N. Menicucci, and I. Dhand. "Fault-tolerant quantum computation with static linear optics". PRX Quantum **2**, 040353 (2021).
[33] J. E. Bourassa, R. N. Alexander, M. Vasmer, A. Patil, I. Tzitrin, T. Matsuura, D. Su, B. Q. Baragiola, S. Guha, G. Dauphinais, K. K. Sabapathy, N. C. Menicucci, and I. Dhand. "Blueprint for a Scalable Photonic Fault-Tolerant Quantum Computer". Quantum **5**, 392 (2021).
[34] S. Krinner, N. Lacroix, A. Remm, and et al. "Realizing repeated quantum error correction in a distance-three surface code". Nature **605**, 669–674 (2022).
[35] Y. Zhao, Y. Ye, H.-L. Huang, and et al. "Realization of an error-correcting surface code with superconducting qubits". Physical Review Letters **129**, 030501 (2022).
[36] R. Acharya, D. A. Abanin, L. Aghababaie-Beni, I. Aleiner, T. I. Andersen, and et al. "Quantum error correction below the surface code threshold". Nature **638**, 920–926 (2024).
[37] A. G. Fowler and C. Gidney. "Low overhead quantum computation using lattice surgery" (2019). arXiv:1808.06709.

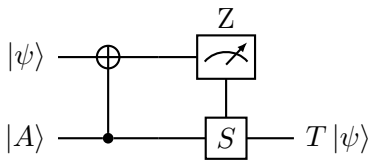
[38] A. Bolt, G. Duclos-Cianci, D. Poulin, and T. M. Stace. "Foliated quantum error-correcting codes". Phys. Rev. Lett. **117**, 070501 (2016).
[39] M. B. Hastings and A. Geller. "Reduced space-time and time costs using dislocation codes and arbitrary ancillas". Quantum Information and Computation **15**, 962–986 (2015).
[40] T. J. Yoder and I. H. Kim. "The surface code with a twist". Quantum **1**, 2 (2017).
[41] B. J. Brown, K. Laubscher, M. S. Kesselring, and J. R. Wootton. "Poking holes and cutting corners to achieve clifford gates with the surface code". Physical Review X **7**, 021029 (2017).
[42] T. Rudolph and L. Grover. "A 2 rebit gate universal for quantum computing" (2002). arXiv:quant-ph/0210187.
[43] N. Delfosse, P. Allard Guerin, J. Bian, and R. Raussendorf. "Wigner function negativity and contextuality in quantum computation on rebits". Physical Review X**5** (2015).
[44] S.-H. Lee and H. Jeong. "Universal hardware-efficient topological measurement-based quantum computation via color-code-based cluster states". Physical Review Research **4**, 013010 (2022).
[45] A. Steane. "Quantum Reed-Muller Codes" (1996). arXiv:quant-ph/9608026.
[46] E. Knill, R. Laflamme, and W. Zurek. "Threshold accuracy for quantum computation" (1996). arXiv:quant-ph/9610011.
[47] V. Albert and P. Faist. "The Error Correction Zoo: $[[15, 1, 3]]$ quantum Reed-Muller code". url: `https://errorcorrectionzoo.org/c/stab_15_1_3`.
[48] A. Paetznick and A. G. Fowler. "Quantum circuit optimization by topological compaction in the surface code" (2013). arXiv:1304.2807.
[49] A. G. Fowler and S. J. Devitt. "A bridge to lower overhead quantum computation" (2013). arXiv:1209.0510.
[50] M. Hanks, M. P. Estarellas, W. J. Munro, and K. Nemoto. "Effective Compression of Quantum Braided Circuits Aided by ZX-Calculus". Physical Review X **10**, 041030 (2020).
[51] B. Coecke and R. Duncan. "Interacting quantum observables". In Luca Aceto, Ivan Damgård, Leslie Ann Goldberg, Magnús M. Halldórsson, Anna Ingólfsdóttir, and Igor Walukiewicz, editors, Automata, Languages and Programming. Pages 298–310. Springer Berlin Heidelberg (2008).
[52] B. Coecke and R. Duncan. "Interacting quantum observables: categorical algebra and diagrammatics". New Journal of Physics **13**, 043016 (2011).
[53] A. Paler, S. J. Devitt, and A. G. Fowler. "Synthesis of Arbitrary Quantum Circuits to Topological Assembly". Scientific Reports**6** (2016).
[54] M. Barkeshli, C.-M. Jian, and X.-L. Qi. "Twist defects and projective non-Abelian braiding statistics". Physical Review B **87**, 045130 (2013).
[55] J. Bridgeman, S. Bartlett, and A. Doherty. "Tensor networks with a twist: Anyon-permuting domain walls and defects in projected entangled pair states". Physical Review B **96**, 245122 (2017).
[56] M. Kesselring, F. Pastawski, J. Eisert, and B. Brown. "The boundaries and twist defects of the color code and their applications to topological quantum computation". Quantum **2**, 101 (2018).
[57] M. Kesselring, J. Magdalena de la Fuente, F. Thomsen, J. Eisert, S. Bartlett, and B. Brown. "Anyon condensation and the color code". PRX Quantum **5**, 010342 (2024).
[58] Y.-Q. Wang, C. Liu, and Y.-M. Lu. "Theory of topological defects and textures in two-dimensional quantum orders with spontaneous symmetry breaking". Physical Review B **109**, 195165 (2024).

[59] M. Davydova, N. Tantivasadakarn, S. Balasubramanian, and D. Aasen. "Quantum computation from dynamic automorphism codes". Quantum **8**, 1448 (2024).
[60] Z. Song and G. Zhu. "Magic Boundaries of 3D Color Codes". Quantum **9**, 1831 (2025).
[61] M. Hostetter. "Galois: A performant NumPy extension for Galois fields". https://github.com/mhostetter/galois (2020).

A T gate and magic state distillation

A.1 T gate by magic state injection

The rotation gate $T = e^{\frac{i\pi Z}{4}}$ cannot be implemented topologically in our framework. To circumvent this issue, one can consider the circuit presented in (31) using the ancilla $|A\rangle = \frac{|0\rangle+e^{i\pi/4}|1\rangle}{\sqrt{2}}$. The resulting state depends on the Z measurement outcomes, and a S gate needs to be subsequently applied every second times, in average.



(31)

The logical state $|A\rangle$ cannot be created by Clifford gates and X, Z creations and measurements. Figure 12 illustrates the creation of a magic state: Here, the measurement lines must be shrunk on the initial plane to a single qubit, resulting in a Bell state between that qubit and the state on the 'out' plane. A measurement of the single 'in' qubit in the $\frac{X+Y}{\sqrt{2}}$ basis yields the desired encoded $|A\rangle$ state (or its orthogonal counterpart) on the 'out' plane. Since the measurement lines must shrink, this procedure is not topologically protected and hence prone to errors. This will be resolved by subsequent distillations.

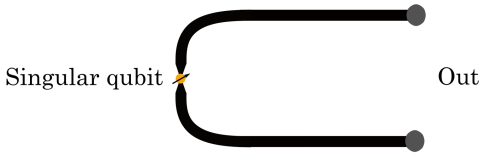
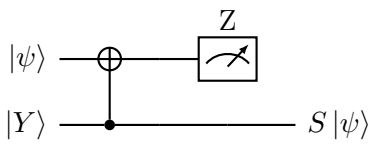


Figure 12: State injection in the Cluster state. The black lines represent the qubits that are measured in Z, they result in an encoded qubit on the output Toric code. The orange point is a punctual qubit that will be measured in $\frac{X+Y}{\sqrt{2}}$ or Y basis, to enforce the $|A\rangle$ or $|Y\rangle$ state on the encoded qubit, respectively.

In [15], the S similarly required $|Y\rangle = \frac{|0\rangle+i|1\rangle}{\sqrt{2}}$ ancillas, see (32), which could only be obtained by the procedure above, but with a Y measurement. The rebit encoding proposed here resolved that issue, except for the circuit (28), which must be done with qubits.



(32)

In the following, we analyze the distillation circuit in details, and propose a circuit compactification that drastically improves the overhead of the T gate.

A.2 Magic state distillation

To reduce the errors arising from the creation of the $|A\rangle$ state, one can resort to magic state distillation [10]. In this paper, we focus on the methods using the 15 qubits Reed-Muller code [45, 46, 47]. A Toric code state is first created fault-tolerantly by MBQC with 16 logical qubits in it. Those logical qubits are not created randomly but are eigenstates of very specific Pauli operators. The 15 first qubits are in the code space of the 15 qubit Reed-Muller code (they are eigenstates of 4 X stabilizers and 10 Z stabilizers, thus creating another level of encoding). This state behaves as an over encoded qubit which is, on top of that, entangled as a Bell pair with the 16th remaining qubit. This procedure involves only creation of individual qubit in the $\mathcal{X}$ and $\mathcal{Z}$ bases, and CNOT gates, and thus can be made fault-tolerant in MBQC. For the exact quantum circuit, we refer the reader to Ref. [15], Fig. A.1. Translating it to MBQC, we show the corresponding measurement pattern in Fig. 13 . The 16 pairs of lines ending on the right are the 16 logical qubits mentioned above. The overall state after this first step is

$$|\varphi\rangle = \frac{\left|0^{RM}\right\rangle \otimes |0\rangle_{16} + \left|1^{RM}\right\rangle \otimes |1\rangle_{16}}{\sqrt{2}} \tag{33}$$

where $\left|0^{RM}\right\rangle$ and $\left|1^{RM}\right\rangle$ are eigenstates of the 15 qubits Reed-Muller code stabilizers and of the associated logical $\mathcal{Z}^{RM}$ operator for the value $+1$ and -1 respectively.

Then, a transversal encoded T gate is applied to the 15 qubits of the Reed-Muller code. This is done by injecting a magic $|A\rangle$ state like in Fig. 12 on each qubit pair of lines and performing the circuit in Eq. 31. The 15 Reed-Muller code is the smallest code where the T^{RM} and $\mathcal{X}^{RM}$ gates are transversal. Therefore, the resulting state is still an eigenstate of the Reed-Muller stabilizers and entangled with the 16th qubit, but in a rotated basis:

$$T^{RM}|\varphi\rangle = \frac{\left|0^{RM}\right\rangle \otimes |0\rangle_{16} + e^{\frac{i\pi}{4}}\left|1^{RM}\right\rangle \otimes |1\rangle_{16}}{\sqrt{2}} \tag{34}$$

Finally, the 15 first qubits are all measured individually in the $\mathcal{X}$ basis and thus the encoded state is measured in the $\mathcal{X}^{RM}$ basis too. That projects the 16th qubit on the $|A\rangle$ state or its complex conjugate depending on the measurement outcome, which can be fixed applying a S gate. On top of it, the $\mathcal{X}$ measurement outcomes are used to infer the Reed-Muller X-type stabilizers. If an error is thereby detected, the state is discarded and the process is started again. Then, the kept $|A\rangle$ states have a reduced error probability compared to the previously injected ones. This procedure can be repeated many times and by concatenation, the error probability shrinks exponentially.

Note that T gates are used to distillate the ancillas, therefore S gates must be performed too (see Eq. 31). We will use $|Y\rangle$ state injection and Eq. 32. Thanks to the Reed-Muller stabilizers, the number of S gates needed in total can be reduced from $15/2$ to $1705/512 \sim 3.33$. Therefore, we need to inject, in average, that many $|Y\rangle$ state ancillas. Those ancillas are also noisy, so we need to distillate them too. The procedure is similar to what we just described for the $|A\rangle$ state, except that we use the 7 qubits Steane code instead, which is much less volume consuming.

A.3 Overhead

A.3.1 Magic state distillation, error probability and overhead

The distillation circuit fails if either one of the T gates fails. In other words if one ancilla $|A\rangle$ from the previous round of distillation is faulty, or if the correction in the topological

circuit fails. By topological circuit, we mean all the parts in the distillation procedure that can be implemented fault-tolerantly by MBQC in the cluster and on which we can do error correction. Then, the probability of failure of the round l of distillation is : $15\epsilon^A_{l-1}+\epsilon_{top}(L^A_{topo},\lambda_{l-1},d_{l-1})$, with ϵ^A_{l-1} the error probability of the ancillas at the previous round and $\epsilon_{top}(L^A_{topo},\lambda_{l-1},d_{l-1})$ the failure rate of the correction in the cluster. L^A_{topo} is the minimal cluster size and λ_{l-1},d_{l-1} the scaling parameters to optimize. As we measure the Reed-Muller code stabilizers, we can detect some errors induced by faulty ancillas and we discard those cases. The remaining cases, where we don't know if an error has happened or not, have an error probability ϵ^A_l given in Eq. 35 at first order. The power 3 in ϵ^A_{l-1} is due the Reed-Muller error detection. The overhead for the $|A\rangle$ distillation at round l can now be computed, involving the failure rate of the process and its volume (volume of the topological circuit + volume of the ancillas $|A\rangle$ and $|Y\rangle$ distilled in the previous round).

$$\begin{aligned}\epsilon^A_l &= 35(\epsilon^A_{l-1})^3+\epsilon_{topo}(L_Y,\lambda_{l-1},d_{l-1})\\ O^A_l &= \frac{1}{1-15\epsilon^A_{l-1}-\epsilon_{topo}(L_A,\lambda_{l-1},d_{l-1})}(15O^A_{l-1}+\frac{1705}{512}O^Y_{l-1}+24V_A\lambda^3_{l-1})\end{aligned} \tag{35}$$

In the same fashion, we can derive the recursion relations for the $|Y\rangle$ state distillation (see Eq. 36). The error probability of the ancilla after the l^{th} round of distillation depends on the performance of the Steane code correction, the error probability of the previous round and the failure rate of the topological circuit. The overhead involves the probability of failure of the distillation times the volume it takes. λ_{l-1} and d_{l-1} are taken to be the same as in Eq. 35 for the $|A\rangle$ state.

$$\begin{aligned}\epsilon^Y_l &= 7(\epsilon^Y_{l-1})^3+\epsilon_{top}(L_Y,\lambda_{l-1},d_{l-1})\\ O^Y_l &= \frac{1}{1-7\epsilon^Y_{l-1}-\epsilon_{top}(L_Y,\lambda_{l-1},d_{l-1})}(7O^Y_{l-1}+24V_Y\lambda^3_{l-1})\end{aligned} \tag{36}$$

A.3.2 T gate overhead

Finally, we can compute the T gate overhead (see Eq. 37). As the circuit involves magic states injection, we need to take into account the volume induced by the distillation and the ancillas error probabilities. After the round of distillation l_{max} for both $|Y\rangle$ and $|A\rangle$ states, the volume taken by the T gate is given by the volume of the topological circuit plus the overheads for the ancillas (the 0.5 factor in front of $O^Y_{l_{max}}$ accounts for the S gate to be performed only one time over two in average). Then the probability of failure of the circuit is due to faulty ancillas or topological circuit.

$$O_T(\Omega)=\Omega(O^A_{l_{max}}+0.5O^Y_{l_{max}}+36(\lambda_{l_{max}})^3V_T)\exp((\epsilon^A_{l_{max}}+\epsilon^Y_{l_{max}}+\epsilon_{top}(L_T,\lambda_{l_{max}},d_{l_{max}}))\Omega) \tag{37}$$

To find the optimal parameters, we need to minimize the overhead with respect to l_{max} and all the rescaling parameters used in each round of distillation : $\{\lambda_{l_{max}},d_{l_{max}},...,\lambda_0,d_0\}$.

In the rebit encoding, the T gate takes an additional $|A\rangle^*$ distilled state and more volume for the topological circuit. The overheads is then as follows, and needs to be optimized as well.

$$O_T(\Omega)=\Omega(2O^A_{l_{max}}+1.5O^Y_{l_{max}}+24(\lambda_{l_{max}})^3V_{T,RE})\exp((2\epsilon^A_{l_{max}}+1.5\epsilon^Y_{l_{max}}+\epsilon_{top}(L_{T,RE},\lambda_{l_{max}},d_{l_{max}}))\Omega) \tag{38}$$

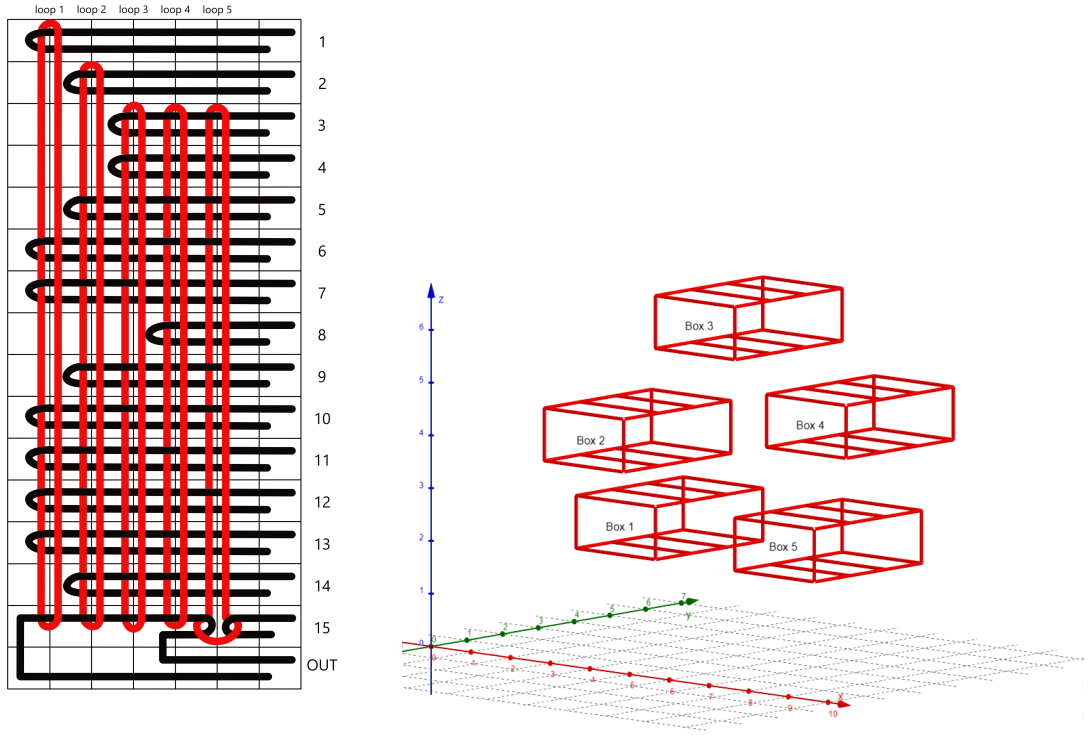
Note that the $|A\rangle$ state is converted in the rebit encoding only after the last round of distillation. One could do it directly at the first round but that would require 2^l15^l ancillas

instead of only 2×15^l, and the topological circuit would be wider as due to the additional qubits necessary to the rebit encoding.

A.3.3 Topological circuit for the A state distillation

The topological circuit for the magic $|A\rangle$ state distillation proposed in [15] is shown in Figure 13(a). It is made only of qubit creations, CNOT gates which can be implemented in a topological fashion, see Figure 4. On the output side, the Reed-Muller state of the first 15 qubits are in a Bell-pair with the 16th qubit ('OUT' label in the figure): $\frac{1}{\sqrt{2}}\left(\left|0^{RM}\right\rangle \times |0\rangle_{\text{out}} + \left|1^{RM}\right\rangle \times |1\rangle_{\text{out}}\right)$.

The almost planar circuit of Figure 13(a) requires a large volume. In order to improve this, we first use the equivalence introduced in Figure 10, then contract the loops into boxes as in 11, and finally optimize the relative positioning of five blocks. The result presented in Figure 13(b) (showing all the red boxes but not the connecting black lines for clarity) spans 192 unit cells. The architecture mirrors the geometrical description of the 15 qubit Reed-Muller code in [47]. Indeed, the boxes implement the X stabilizers and they should be close to each other whenever they share a logical qubit, yielding a natural 'tetrahedral' geometry. Both circuits presented in Fig. 13 have been positively checked by our algorithmic verifier. We leave it to subsequent work to develop a computer code to search of the optimal architecture for any given circuit.



(a) Topological circuit from Ref. [15].

(b) Optimized topological circuit.

Figure 13: Topological circuit of the $|A\rangle$ state distillation with the 15 qubits Reed-Muller encoding. (a) Reference circuit from Ref. [15]. The grid in the background represents the unit cells of the lattice. (b) Circuit of the $|A\rangle$ sate distillation with the rectangular boxes introduce in Figure 11. There are five boxes corresponding to the five loops in (a). The black lines connecting the boxes and encoding the logical qubits are not represented here.

B Algorithmic Circuit Verification

B.1 Mathematical foundation

B.1.1 Homology and relative homology

Recall that by G_0, G_1, G_2, and G_3 we respectively denote the set of the lattice's primal vertices, edges, squares, and volumes, as well as that C_n denotes the vector space associated with G_n. C_n is composed of formal linear combination of n-cells, i.e., of vectors $c = \sum_{g\in G_n} \alpha_g\, g$ with $\alpha_g \in \mathbb{F}_2$. We have then $C_n \cong \mathbb{F}_2^{m_n}$ with $m_n = |G_n|$, and the elements of G_n can be represented by $\{e_i\}$, the canonical basis of $\mathbb{F}_2^{m_n}$.

Furthermore, recall that the linear operators $\partial_n : C_n \to C_{n-1}$, called the *boundary operators*, in particular satisfy $\partial_n \circ \partial_{n-1} = 0$. With respect to the canonical basis described above, the boundary operators are $(m_{n-1}\times m_n)$-matrices with $\{0,1\}$ entries, where $(\partial_n)_{ij} = 1$ if and only if the ith edge belongs to the boundary of the jth square.

The sequence (C_n, ∂_n) forms a chain complex. Its dual chain complex $(\overline{C}_n, \overline{\partial}_n)$ consists of its algebraic dual spaces and its dual maps, i.e., $\overline{C}_n = C^*_{3-n}$ and $\overline{\partial}_n = \partial^*_{3-n}$. Since all considered vector spaces are finite, they are naturally isomorphic to their respective dual counterparts, resulting in $\overline{C}_n \cong C_{3-n}$ (see Eq. 2).

We now turn to the impact of qubit measurement lines, which correspond to a chain $l \in C_1$ and a cochain $\bar{l} \in \overline{C}_1 \cong C_2$, and effectively remove the edges and faces that have been measured from the primal, respectively dual chain complexes. This is best described in terms of relative homology, see [20], since relative cycles are chains of the original space whose boundaries are chains in the subspace.

For any chain $c = \sum_i \alpha_i e_i \in C_n$, we let

$$V_n(c) = \mathrm{span}\{e_i \in C_n : \alpha_i = 1\} \tag{39}$$

be the n-chains made up of the measured qubits, and let $W_n(c) = C_n/V_n(c)$ be the quotient space of C_n by $V_n(c)$. $W_n(c)$ is naturally isomorphic to $\mathrm{span}\{e_i \in C_n : \alpha_i = 0\}$ and we shall use this identification to write $C_n = V_n(c) \oplus W_n(c)$ and introduce the projection $P_n : C_n \to W_n(c)$ (equivalently: the quotient map). The original complex projects down to

$$C_3 \xrightarrow{\partial^r_3} W_2(\bar{l}) \xrightarrow{\partial^r_2} W_1(l) \xrightarrow{\partial^r_1} C_0, \tag{40}$$

where the relative boundary operator ∂^r_n is given by $\partial^r_n = P_{n-1}\partial_n P_n$.

In this new complex, two chains are equivalent if they differ by a relative boundary, namely $c - c' \in \mathrm{Ran}(\partial^r)$. Concretely in terms of the original complex and in the case $n = 1$, two lines are equivalent if their difference is a sum of the boundary of a surface and a measured line (since the latter is in the kernel of ∂^r).

In terms of the matrix representatives of the boundary maps, ∂^r_n is obtained from ∂_n by setting those rows and columns to zero whenever they correspond to measured qubits. Equivalently by simply deleting these rows and columns, which amounts to considering $\partial^r_n : \mathrm{Ran}(P_n) \to \mathrm{Ran}(P_{n-1})$.

B.1.2 Correlation surfaces

The algorithm's design is limited to CSS gates, namely those unitaries in Theorem 1 for which (9) and (11) hold without the Z, respectively X operator on their left hand side. Accordingly, the equations for primal and dual surfaces are separated and (8, 9) can be treated independently of (10, 11).

Verifying that a measurement pattern specified by l corresponds to a circuit represented by the primal surface boundaries $c_1, \ldots, c_n \in C_1$ on the 'in' plane $\mathcal{I}$ and the 'out' plane $\mathcal{O}$ means checking whether correlation surfaces with these boundaries exist. In terms of Theorem 1, this means finding solutions c_i^Z of (10,11) such that $\partial_2^r c_i^Z \cap (\mathcal{I} \cup \mathcal{O}) = c_i$. The use of the relative boundary operator in this equation ensures that the solution surfaces are supported by the measurement lines inside the cluster. This is equivalent to verifying whether all of the linear systems of equations

$$\partial_2^r x = c_i \tag{41}$$

for $i = 1, \ldots, n$ have a solution. The Rouché-Capelli theorem ensures that this holds if the matrix ∂_2^r has the same rank as the augmented matrices $(\partial_2^r \,|\, c_i)$. The same must hold for dual surfaces and their boundaries, yielding the additional equations

$$\overline{\partial}_2^r y = \overline{c}_j \tag{42}$$

and the corresponding rank-computing problem for $\overline{\partial}_2^r$ and $(\overline{\partial}_2^r \,|\, \overline{c}_i)$.

Summarizing, after the preliminary steps of building the lattice, constructing the boundary and relative boundary matrices described above, the verification algorithm reduces to the standard linear algebra task of computing the rank (over $\mathbb{F}_2$) of $2(n+1)$ matrices. The whole process, which we describe below in detail, can be done in polynomial time in the linear size of the cluster.

B.2 Algorithmic Verification

We assume the lattice to be a rectangular cuboid whose side lengths (or shape) is given by $(s_1, s_2, s_3) \in \mathbb{N}^3$. The most decisive design choice is selecting how to represent the sets G_n numerically. We place the $m_0 = \prod_i s_i$ primal vertices along the equidistant points $(x_1, x_2, x_3) \in [s_1] \times [s_2] \times [s_3]$. The $m_3 = \prod_i (s_i - 1)$ primal cubes can then be represented by their midpoints, i.e., by $(z_1, z_2, z_3) = (y_1 + 0.5, y_2 + 0.5, y_3 + 0.5)$ with $y_i \in [s_i - 1]$. The m_1 primal links and m_2 primal faces are subsequently characterized by the primal sites or primal cubes they respectively connect; specifically, by the midpoint of these primal sites or primal cubes. One advantage of these human readable representations is that they allow for manual data entry of circuits to verify. Foremost, they enable a straightforward computation of the (dual) boundary of an object. For example, the boundary of a primal link $(x_1, x_2 + 0.5, x_3) \in G_2$ is given by

$$\{(x_1, x_2, x_3), (x_1, x_2 + 1, x_3)\} \subset G_1, \tag{43}$$

and the boundary of a primal face $(z_1 + 0.5, z_2, z_3) \in G_3$ is given by

$$\begin{aligned} \{&(z_1 + 0.5,\ z_2 - 0.5,\ z_3), (z_1 + 0.5,\ z_2,\ z_3 - 0.5), \\ &(z_1 + 0.5,\ z_2 + 0.5,\ z_3), (z_1 + 0.5,\ z_2,\ z_3 + 0.5)\} \subset G_2. \end{aligned} \tag{44}$$

This generalizes trivially to links $(x_1 + 0.5, x_2, x_3), (x_1, x_2, x_3 + 0.5) \in G_2$ and faces $(z_1, z_2 + 0.5, z_3)$, $(z_1, z_2, z_3 + 0.5) \in G_3$.

These preliminaries enable us to provide a high-level outline of the algorithm. We assume to be given the shape (s_1, s_2, s_3) of the lattice, along with the sets of qubit measurement lines $L \subset G_1$ and $\overline{L} \subset \overline{G}_1 = G_2$. Lastly, two lists consisting of primal surface boundaries $B_1, \ldots, B_{n_1} \subset G_1$ and dual surface boundaries $\overline{B}_1, \ldots, \overline{B}_{n_2} \subset \overline{G}_1 = G_2$ to verify are required. The program then proceeds as follows.

1. **Building the lattice:** For convenience, the representations of all n-cells are stored in an iterable data type. Afterwards, a bijection between the sets G_n and their respective index sets $[m_n]$ can be created by simply enumerating all objects in G_n and storing the result in a dictionary. All of this is done once in time $\mathcal{O}(\mathrm{poly}(s))$ where $s = \|(s_1, s_2, s_3)\|_\infty$. Since s is typically of order $< 10^2$, optimizing polynomial run-times is not a dominant concern, both here and in the following. Further note that accessing elements in such an associative array can be done in time $\mathcal{O}(1)$, i.e., switching between an n-cell's descriptive representation in G_n and its index in $[m_n]$ or vector in $\mathbb{F}_2^{m_n}$ in subsequent calculations introduces a negligible computational overhead at all reasonable problem sizes.

2. **Computing the relevant boundary operators:** The matrix representation of an operator ∂_n can be computed by calculating its action on all basis vectors e_i of $C_n \cong \mathbb{F}_2^{m_n}$. Said action can be determined by mapping e_i to its representation in G_n, computing the boundary with that representation (as exemplified above), and mapping the result back from G_{n-1} to $C_{n-1} \cong \mathbb{F}_2^{m_{n-1}}$. In our concrete case, where we are ultimately interested in the operators ∂_2^r and $\overline{\partial}_2^r$, this is done for $n \in \{1, 2\}$, as the matrix representation of $\overline{\partial}_2$ is given by the transpose of the matrix representation of ∂_1. This step hence generates matrices ∂_1 and ∂_2 with $(\partial_n)_{ij} = 1$ if and only if the i-th $(n-1)$-cell is in the boundary of the j-th n-cell, again requiring time $\mathcal{O}(\mathrm{poly}(s))$.

3. **Computing the respective relative boundary operators:** Given a boundary operator ∂_n, the corresponding relative boundary operator ∂_n^r acting on $W_n \cong C_n/V_n$ is calculated with $\partial_n^r = P_{n-1}\,\partial_n\,P_n$. The projectors P_n are dependent on the qubit measurement lines L and $\overline{L}$, and can be calculated by removing the i-th diagonal element of an identity matrix if and only if the i-th n-cell is an element of $L \cup \overline{L}$. Alternatively, to avoid storing the projectors in memory, one can iterate through all elements in $L \cup \overline{L}$ and equate to zero the possibly corresponding row or column of ∂_n. This approach is taken by us, again for $n \in \{1, 2\}$ while requiring time $\mathcal{O}(\mathrm{poly}(s))$. Note that the dual lattice must also be 'smooth', there cannot be half dual unit cells on the cluster boundary, which can effectively be implemented and ignoring the outer-most primal links by removing relevant entries from the dual relative boundary operator.

4. **Computing the ranks of the (augmented) relative boundary operators:** After using the created dictionary to map the surface boundaries $B_1, \ldots, B_{n_1} \subset G_1$ and $\overline{B}_1, \ldots, \overline{B}_{n_2} \subset \overline{G}_1 = G_2$ to the target vectors $c_1, \ldots, c_{n_1} \in C_1$ and $\bar{c}_1, \ldots, \bar{c}_{n_2} \in \overline{C}_1$, the ranks of $\partial_2^r, (\partial_2^r \,|\, c_1), \ldots, (\partial_2^r \,|\, c_{n_1}), \bar{\partial}_2^r, (\bar{\partial}_2^r \,|\, \bar{c}_1), \ldots, (\bar{\partial}_2^r \,|\, \bar{c}_{n_2})$ can be calculated in time $\mathcal{O}(\mathrm{poly}(s))$ by, for example, using any linear algebra library operating over finite fields. We employed the *galois* package [61] for Python.