Supplementary material for: Correcting coherent errors with surface codes

Sergey Bravyi,[1] Matthias Englbrecht,[2,3] Robert König,[2,*] and Nolan Peard[2,4]

[1] *IBM T.J. Watson Research Center*
[2] *Institute for Advanced Study & Zentrum Mathematik, Technical University of Munich*
[3] *Institute for Theoretical Physics, University of Innsbruck*
[4] *Department of Physics, Massachusetts Institute of Technology*

(Dated: September 26, 2018)

SUPPLEMENTARY NOTE 1: THE MAJORANA ENCODING

Here we review the fermionic representation of the surface code (see Fig. 6 in the main text) used in our simulation algorithm. We also establish fundamental relations between logical states (respectively operators) of the surface code and fermionic states (respectively operators). We will subsequently use these in the Supplementary Notes 2 and 3.

A single Majorana mode p is described by a hermitian operator c_p satisfying $c_p^2 = I$. Operators c_p associated with different modes anti-commute, see Supplementary Note 4 for formal definitions. A system of four Majorana modes c_1, c_2, c_3, c_4 can be used to encode a qubit using a stabilizer code with a single stabilizer

$$S = -c_1 c_2 c_3 c_4 \tag{1}$$

and logical Pauli operators

$$\overline{X} = ic_1c_2 = ic_3c_4S \quad \text{and} \quad \overline{Z} = ic_2c_3 = ic_1c_4S. \tag{2}$$

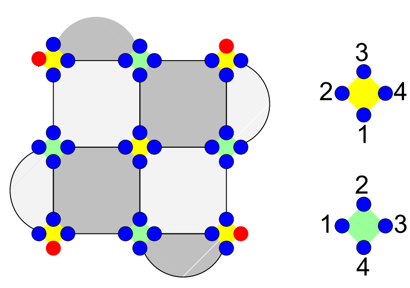
We shall refer to this encoding as a *C*4*-code*.

Consider a surface code with n qubits on a square $d \times d$ lattice, where $n = d^2$. It can be described by a planar graph $G = (V, E, F)$ with a set of n vertices V, a set of $2n-2$ edges E, and a set of $n-1$ faces F. Qubits are located at vertices $u \in V$ and stabilizers B_f are located at faces $f \in F$ of G. Consider a system of $4n$ Majorana modes $c_1, \ldots, c_{4n}$ distributed over edges and vertices of G as shown on Figure 6in the main text. There are exactly two *paired* modes located near the endpoints of every edge $e \in E$ and four *unpaired* modes c_1, c_2, c_3, c_4 located near the corners of the lattice as shown on Fig. 6.The paired modes are labeled as $c_5, c_6, \ldots, c_{4n}$ in an arbitrary order.

Let us orient the edges of G as shown on Fig. 6. Suppose $e \in E$ is an edge connecting some pair of modes c_p, c_q such that c_p is the tail of e and c_q is the head of e, see Fig. 6. Define the *link operator*

$$L_e = ic_pc_q. \tag{3}$$

Note that L_e is hermitian and all link operators pairwise commute. Furthermore, each L_e commutes with the unpaired modes c_1, c_2, c_3, c_4.



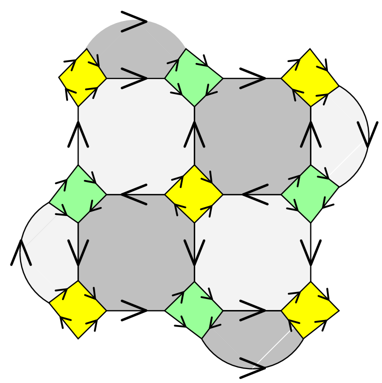
Supplementary Figure 1. Four-mode clusters Γ_u located at vertices encode qubits of the surface code. Modes in each cluster are ordered as shown on the right. Each vertex u has a stabilizer $S_u = -c_{u,1}c_{u,2}c_{u,3}c_{u,4}$, where $c_{u,j}$ is the j-th mode of Γ_u. Logical Pauli operators are $\overline{X}_u = ic_{u,1}c_{u,2} = ic_{u,3}c_{u,4}S_u$ and $\overline{Z}_u = ic_{u,2}c_{u,3} = ic_{u,1}c_{u,4}S_u$, see also Supplementary Figure 2.

* robert.koenig@tum.de

By construction, a small neighborhood of each vertex $u \in V$ contains a cluster of four modes, see Supplementary Figure 1. We shall denote this cluster Γ_u. Define a *vertex stabilizer*

$$S_u = -\prod_{p\in\Gamma_u} c_p, \tag{4}$$

where a particular product order is chosen for each vertex as shown on Supplementary Figure 1. Since $|\Gamma_u| = 4$ for all u and the subsets Γ_u are pairwise disjoint, vertex stabilizers are hermitian and pairwise commuting. We shall consider $S_1, \ldots, S_n$ as stabilizers for n independent copies of the $C4$-code defined in Eqs. (1,2) such that each qubit of the surface code is encoded into its own $C4$-code. Let $\overline{X}_u$, $\overline{Z}_u$, and $\overline{Y}_u = i\overline{X}_u\overline{Z}_u$ be the logical Pauli operators for the qubit located at a vertex u, see Eq. (2). By definition, each of these logical operators has the form ic_pc_q for some pair of Majorana modes $p, q \in \Gamma_u$, see Supplementary Figure 1. The logical operators $\overline{X}_u$, $\overline{Z}_u$ are indicated by small arrows on Supplementary Figure 2.



Supplementary Figure 2. Small arrows indicate the order of modes in the logical operators $\overline{X}_u$ and $\overline{Z}_u$. Each of these operators has the form ic_pc_q, where p is the tail and q is the head of the respective arrow. The orientation is chosen such that the boundary of each face has an odd number of arrows oriented clockwise. To avoid clutter, we do not show the Majorana modes.

Let P be a Pauli operator acting on the surface code qubits. We shall say that a Majorana operator $\overline{P}$ is a $C4$-encoding of P if $\overline{P}$ can be obtained from P by replacing each single-qubit Pauli operator X_u, Y_u, Z_u by its logical counterpart $\overline{X}_u$, $\overline{Y}_u$, $\overline{Z}_u$ and, possibly, multiplying by the stabilizer S_u. Given a single-qubit state ψ, one can define several encoded versions of ψ using the surface code, the $C4$-code, and the surface code concatenated with n-copies of the $C4$-code. We shall denote these encoded states ψ_L, $\overline{\psi}$, and $\overline{\psi}_L$ respectively. These notations are summarized in Table I.

	Hilbert space	Encoding
ψ_L	n qubits	surface code
$\overline{\psi}$	four Majorana modes	$C4$-code
$\overline{\psi}_L$	$4n$ Majorana modes	encode each qubit of ψ_L into the $C4$-code

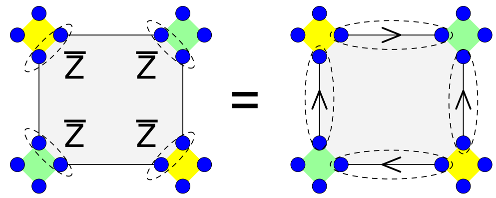
TABLE I. Encoded versions of a single-qubit state ψ.

The desired fermionic representation of the surface code is established in Lemmas 1,2,3 below. Consider a face $f \in F$ and let $\partial f \subseteq E$ be the boundary of f.

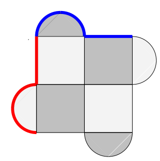
Lemma 1. *Let B_f be the surface code stabilizer located on a face f. Then a $C4$-encoding of B_f can be chosen as*

$$\overline{B}_f = \prod_{e\in\partial f} L_e. \tag{5}$$

We illustrate Eq. (5) on Supplementary Figure 3. The lemma shows that measuring the surface code syndrome can be reduced (after the $C4$-encoding) to measuring eigenvalues of pairwise commuting link operators L_e. We shall see that under certain circumstances such measurements can be efficiently simulated classically.



Supplementary Figure 3. After the $C4$-encoding, surface code syndromes can be obtained by measuring eigenvalues of pairwise commuting link operators (see Lemma 1). *Left:* $C4$-encoding of a Z-type surface code stabilizer B_f. *Right:* The same operator represented as a product of link operators L_e over the boundary of f.



Supplementary Figure 4. The sets of edges LEFT (red) and TOP (blue) used in Lemma 2. The logical operators (in the $C4$-encoding) can be expressed as products of two unpaired modes and the link operators associated with these subsets, see Eq. (6).

Proof. Consider a face f such that B_f is a Z-stabilizer. Then $\overline{B}_f = \prod_{u\in f} \overline{Z}_u$. Consider a vertex $u \in f$ and the $C4$-code located at u. The corresponding logical-Z operator $\overline{Z}_u = ic_pc_q$ can be chosen such that both modes c_p, c_q are located on the boundary of f, see Supplementary Figure 1. Thus $\overline{B}_f$ is proportional to the product of all modes located on the boundary of f. The same is true about the operator $\prod_{e\in\partial f} L_e$. Thus $\prod_{e\in\partial f} L_e = \pm\overline{B}_f$.

By construction, the boundary ∂f alternates between link operators L_e and logical-Z operators $\overline{Z}_u$, see Supplementary Figure 3 for an example. For each link operator L_e with $e \in \partial f$ define a quantity $\omega_f(L_e) = \pm 1$ such that $\omega_f(L_e) = -1$ iff e is oriented clockwise with respect to f. Likewise, for each logical-Z operator $\overline{Z}_u = ic_pc_q$ lying on the boundary of f define a quantity $\omega_f(\overline{Z}_u) = \pm 1$ such that $\omega_f(\overline{Z}_u) = -1$ iff an arrow $c_p \to c_q$ is oriented clockwise with respect to f. See Supplementary Figure 2 for examples of such arrows. A simple computation shows that $\prod_{e\in\partial f} L_e = -\omega_f \overline{B}_f$, where

$$\omega_f = \prod_{e\in\partial f} \omega_f(L_e) \prod_{u\in f} \omega_f(\overline{Z}_u).$$

Thus we need to check that $\omega_f = -1$ for each face $f \in F$. In other words, the boundary of each face must have an odd number of arrows oriented clockwise. Direct inspection shows that this is indeed the case for the distance-3 code, see Supplementary Figure 2. By translation invariance, this also holds for all code distances. The same arguments apply to X-type stabilizers. □

We shall need an analogue of Lemma 1 for logical operators of the surface code.

Lemma 2. *Let X_L and Z_L be the logical operators of the surface code located on the left and the top boundary. Then $C4$-encodings of X_L and Z_L can be chosen as*

$$\overline{X}_L = ic_1c_2 \prod_{e\in \mathrm{LEFT}} L_e \quad \textit{and} \quad \overline{Z}_L = ic_2c_3 \prod_{e\in \mathrm{TOP}} L_e, \tag{6}$$

where LEFT and TOP are the subsets of edges lying on the left and the top boundaries of the lattice, see Supplementary Figure 4.

Proof. Let us add a "logical edge" connecting the modes c_2 and c_3 to the graph G, see Fig. 6 in the main text. This creates an extra "logical face" f attached to the top boundary of the lattice. The new edge carries a link operator $L_e = ic_2c_3$. The same arguments as in the proof of Lemma 1 show that $\overline{Z}_L = -\omega_f \prod_{e\in\partial f} L_e$, where $\omega_f = \pm 1$ is the parity of the number of arrows lying on the boundary of f and oriented clockwise with respect to f. From Supplementary Figure 2 one gets $\omega_f = -1$. The same argument applies to $\overline{X}_L$. □

Suppose ψ is a single-qubit state. The following lemma allows one to switch between different encodings of ψ defined in Table I by measuring syndromes of the vertex stabilizers. Informally, it asserts that a qubit initially encoded into the four unpaired Majorana modes c_1, c_2, c_3, c_4 at the corners of the lattice can be "injected" into the logical subspace of the surface code by tensoring in unentangled pairs of modes located on edges and measuring syndromes of the vertex stabilizers.

Lemma 3. *Let ϕ_{link} be the state of $4n-4$ Majorana modes $c_5, c_6, \ldots, c_{4n}$ stabilized by all link operators,*

$$|\phi_{\mathsf{link}}\rangle\langle\phi_{\mathsf{link}}| = \prod_{e\in E} \frac{1}{2}(I + L_e). \tag{7}$$

Let ψ be any single-qubit state. Then

$$|\overline{\psi}_L\rangle \sim \prod_{u\in V} \frac{1}{2}(I + S_u)|\overline{\psi}\rangle \otimes |\phi_{\mathsf{link}}\rangle. \tag{8}$$

Here we used the notations from Table I.

The lemma will allow us to replace the initial logical state ψ_L in the error correction protocol by a simpler state ϕ_{link} at the cost of measuring certain additional stabilizers. We shall see that the state ϕ_{link} is a fermionic Gaussian state, see Supplementary Note 4 for details. Furthermore, a state obtained from ϕ_{link} by applying a coherent error (encoded by the $C4$-code) is also Gaussian. These features will be instrumental for our simulation algorithm.

Proof. Suppose first that $|\psi\rangle = |0\rangle$. Let us add a pair of "logical edges" connecting modes c_2, c_3 and c_1, c_4 to the graph G. This creates an extra pair of "logical faces" attached to the top and the bottom boundaries. The new edges carry link operators ic_2c_3 and ic_1c_4. Lemmas 1,2 imply that the state $|\overline{\psi}\rangle \otimes |\phi_{\mathsf{link}}\rangle$ is stabilized by operators $\overline{B}_f$ and $\overline{Z}_L$. Furthermore, since these operators commute with all vertex stabilizers, the state on the right-hand side of Eq. (8) is stabilized by $\overline{B}_f$ and $\overline{Z}_L$. Since it is also stabilized by all S_u, this state has the same set of stabilizers as $|\overline{0}_L\rangle$, which proves Eq. (8) for $|\psi\rangle = |0\rangle$. Note that

$$\overline{X}_L|\overline{0}\rangle \otimes |\phi_{\mathsf{link}}\rangle = |\overline{1}\rangle \otimes |\phi_{\mathsf{link}}\rangle\ , \tag{9}$$

see Lemma 2. Furthermore, $\overline{X}_L$ commutes with all vertex stabilizers. Thus applying $\overline{X}_L$ to both sides of Eq. (8) with $|\psi\rangle = |0\rangle$ proves Eq. (8) for $|\psi\rangle = |1\rangle$. By linearity, it holds for all ψ.

We argue that the expression on the rhs. of Eq. (8) does not vanish. Suppose we pick $|\psi\rangle$ such that

$$\prod_{u\in V} S_u|\overline{\psi}\rangle \otimes |\phi_{\mathsf{link}}\rangle = |\overline{\psi}\rangle \otimes |\phi_{\mathsf{link}}\rangle\ . \tag{10}$$

Such a state $|\psi\rangle$ exists because the product $\prod_{u\in V} S_u$ is – up to a sign – equal to a product of link operators and the unpaired Majorana operators c_1, c_2, c_3, c_4, see Fig. 6. Furthermore, the state $\overline{\psi}$ is a $+1$ eigenvector of the operators ic_2c_3 and ic_1c_4, see Eq. (2). Observe that states

$$|\psi_s\rangle = \prod_{u\in V} \frac{1}{2}(I + (-1)^{s_u} S_u)(|\overline{\psi}\rangle \otimes |\phi_{\mathsf{link}}\rangle) \tag{11}$$

associated with different even-weight syndromes $s \in \{0,1\}^{|V|}$ are related to the state with trivial syndrome by products of link-operators, that is

$$|\psi_s\rangle = \prod_{e\in E(s)} L_e|\psi_0\rangle \tag{12}$$

where $E(s)$ is a 1-chain of edges with $\mathbb{Z}_2$ coefficients whose boundary is equal to s. In particular, the states $\{\psi_s\}_s$ are unitarily related and thus $\|\psi_s\| = \|\psi_0\|$ for every $s \in \{0,1\}^{|V|}$ with even weight. Using the fact the projection onto the even-weight syndrome subspace is $\frac{1}{2}(I + \prod_{u\in V} S_u)$, we conclude that

$$2^{|V|-1}\|\psi_0\|^2 = \sum_{\substack{s\in\{0,1\}^{|V|} \\ |s| \text{even}}} \|\psi_s\|^2 = \langle\overline{\psi} \otimes \phi_{\mathsf{link}}|\frac{1}{2}(I + \prod_{u\in V} S_u)|\overline{\psi} \otimes \phi_{\mathsf{link}}\rangle = 1 \tag{13}$$

because of (10). This implies the claim for $|\psi\rangle$ satisfying (10). The general claim then follows from this by applying logical operators $\overline{X}_L$ and $\overline{Z}_L$, respectively, following the argument above (see Eq. (9)).

$\square$

SUPPLEMENTARY NOTE 2: SIMULATING STATE PREPARATION

Here we consider the problem of fault-tolerantly preparing an initial (encoded) state of the surface code. We focus on the robustness of a protocol which executes syndrome measurement and subsequent error correction on an initial product state. We find that this process can be efficiently simulated (just as the storage problem presented in the main text): We construct an algorithm which simulates the effect of general $SU(2)$ coherent errors, and returns both the sampled syndrome as well as a description of the resulting post-measurement state. The runtime of the algorithm is $O(n^2)$. We present numerical results showing that certain logical states can be prepared fault-tolerantly even in the presence of coherent noise. The algorithm for simulating the logical state preparation is simpler than that for storage simulation. The latter is presented in the Supplementary Note 3.

Logical state preparation in the toric code

To describe the problem of logical state preparation, assume first that we have access to noise-free qubits and operations. In this case, the following standard protocol [1] prepares the encoded stabilizer state $|+_L\rangle$ (the $+1$ eigenstate of X_L):

(i) prepare the initial product state $|+\rangle^{\otimes n}$.

(ii) measure the eigenvalues of the stabilizers $\{B_f\}_f$, resulting in a syndrome $s = \{s_f\}$.

(iii) apply a Pauli correction C_s returning the system to the logical subspace in some final state $|\phi_s\rangle$.

Using the fact the the initial state is a $+1$ eigenvector of X_L one can easily check that $|\phi_s\rangle = |+_L\rangle$ for all s, see [11]. [1]. How does this protocol fare in the presence of coherent noise? Let us consider a model where the initial product state cannot be prepared with perfect accuracy, but rather is obtained from $|+\rangle^{\otimes n}$ by applying some unwanted unitary operators to every qubit. Thus (i) is replaced by

(i') prepare an initial state $|\psi_1 \otimes \psi_2 \otimes \cdots \otimes \psi_n\rangle$, where ψ_j are arbitrary single-qubit pure states.

In this case the final state $|\phi_s\rangle$ may deviate from the target state $|+_L\rangle$ resulting in a logical error. To assess the performance of this protocol, we seek a polynomial time algorithm B which, on input $\psi_1, \ldots, \psi_n \in \mathbb{C}^2$, outputs a random syndrome s sampled from the distribution $p(s)$ specified by the measurement (ii) together with the final logical state $|\phi_s\rangle$.

We shall first describe the algorithm for simulating the logical state preparation because it is much simpler than the storage simulation. For each syndrome $s = \{s_f\}_{f\in F}$ define a syndrome projector

$$\Pi_s = \prod_{f\in F} \frac{1}{2}(I + s_f B_f). \tag{14}$$

It projects onto the subspace spanned by n-qubit states with the syndrome s. Note that $\sum_s \Pi_s = I$. Let Π_0 be the projector onto the logical subspace of the surface code. Since the Pauli correction C_s maps any state with a syndrome s to the logical subspace, it must satisfy

$$\Pi_s = C_s \Pi_0 C_s. \tag{15}$$

Here and below we assume that $C_s^\dagger = C_s$.

Suppose that our initial state has the product form

$$|\psi\rangle = |\psi_1 \otimes \psi_2 \otimes \cdots \otimes \psi_n\rangle.$$

Here ψ_j are arbitrary single-qubit states. Our goal is to sample a syndrome s from the probability distribution

$$p(s) = \langle\psi|\Pi_s|\psi\rangle \tag{16}$$

and compute the final logical state conditioned on the syndrome. The latter has the form

$$|\phi_s\rangle = \frac{1}{\sqrt{p(s)}} C_s \Pi_s |\psi\rangle \tag{17}$$

Simulating the syndrome measurement

Let us first discuss how to simulate the syndrome measurement. We shall encode each qubit into the $C4$-code as discussed in the Supplementary Note 1. Let $|\overline{\psi}_a\rangle$ be the encoded version of $|\psi_a\rangle$. Using the Euler angle decomposition one can write $|\psi_a\rangle = e^{-i\alpha X}e^{-i\beta Z}e^{-i\gamma X}|0\rangle$. Replacing Pauli operators by their $C4$-encodings defined in Eq. (2) gives

$$|\overline{\psi}_a\rangle = \exp\left(\alpha c_1 c_2\right)\exp\left(\beta c_2 c_3\right)\exp\left(\gamma c_1 c_2\right)|\overline{0}\rangle.$$

The state $|\overline{0}\rangle$ is stabilized by $\overline{Z} = ic_2c_3$ and $\overline{Z}S = ic_1c_4$, see Eq. (2). Thus the states $|\overline{\psi}_a\rangle$ can be prepared using only FLO gates. Applying this independently to each qubit gives a sequence of $O(n)$ FLO gates that prepares the state

$$|\overline{\psi}\rangle = |\overline{\psi}_1 \otimes \cdots \otimes \overline{\psi}_n\rangle.$$

Clearly, measuring syndromes of stabilizers B_f on the state $|\psi\rangle$ is equivalent to measuring syndromes of the encoded stabilizers $\overline{B}_f$ on the state $|\overline{\psi}\rangle$. By Lemma 1, the latter can be reduced to measuring syndromes $m_e = \pm 1$ of the link operators L_e and then classically computing face syndromes $s_f = \prod_{e\in\partial f} m_e$. Since each link operator has a form $L_e = ic_pc_q$, measuring the eigenvalue of L_e is a FLO gate. Thus we have realized the full syndrome measurement using $O(n)$ FLO gates.

Computing the final logical state

It remains to compute the final logical state ϕ_s. Let $\vec{b}_s = (b_s^x, b_s^y, b_s^z)$ be the Bloch vector of ϕ_s such that

$$b_s^x = \langle\phi_s|X_L|\phi_s\rangle, \quad b_s^y = \langle\phi_s|Y_L|\phi_s\rangle, \quad b_s^z = \langle\phi_s|Z_L|\phi_s\rangle.$$

Below we focus on computing b_s^x. Define $\lambda_s = \pm 1$ such that $C_sX_L = \lambda_s X_L C_s$. Then

$$b_s^x = \lambda_s \frac{\langle\psi|\Pi_s X_L|\psi\rangle}{\langle\psi|\Pi_s|\psi\rangle} = \lambda_s \frac{\langle\overline{\psi}|\overline{\Pi}_s \overline{X}_L|\overline{\psi}\rangle}{\langle\overline{\psi}|\overline{\Pi}_s|\overline{\psi}\rangle} \tag{18}$$

Here in the first equality we noted that X_L commutes with Π_s. In the second equality we encoded every qubit using the $C4$-code. Let $m \in \{+1,-1\}^E$ be the combined syndrome of link operators measured in the first part of the algorithm such that m_e is the measured eigenvalue of L_e. Define the corresponding syndrome projector

$$\Pi_m^{\mathsf{link}} = \prod_{e\in E} \frac{1}{2}(I + m_e L_e).$$

We claim that

$$b_s^x = \lambda_s \prod_{e\in \mathrm{LEFT}} m_e \cdot \frac{\langle\overline{\psi}|\Pi_m^{\mathsf{link}} ic_1c_2|\overline{\psi}\rangle}{\langle\overline{\psi}|\Pi_m^{\mathsf{link}}|\overline{\psi}\rangle}. \tag{19}$$

Here LEFT is the subset of edges lying on the left boundary of the lattice, see Supplementary Figure 4. The ratio in Eq. (19) can be computed by taking the normalized state $\Pi_m^{\mathsf{link}}|\overline{\psi}\rangle$ obtained after measuring the link syndromes (the first part of the algorithm) and measuring the eigenvalue of ic_1c_2. The latter requires a single FLO gate. This gives the desired value of b_s^x. Likewise, measuring the eigenvalues of ic_2c_3 and $-ic_1c_3$ on the final state $\Pi_m^{\mathsf{link}}|\overline{\psi}\rangle$ gives the remaining components of the Bloch vector b_s^z and b_s^y respectively.

To conclude, FLO gates enable simulation of the syndrome measurement and computation of the final logical Bloch vector conditioned on the measured syndrome. The resulting FLO circuit can be simulated classically in time $O(n^3)$ since it includes $O(n)$ projection gates $(1/2)(I + m_eL_e)$. The simulation runtime can be significantly improved using the following simple observations. First, once a link operator $L_e = ic_pc_q$ has been measured, the modes c_p, c_q are completely disentangled from the rest of the system. Such disentangled modes can be removed from the simulator reducing the total number of modes and the computational cost of subsequent steps. Second, we can exploit the fact that the initial state $\overline{\psi}$ has a product form. In particular, a four-mode cluster Γ_u that supports the state $\overline{\psi}_u$ only needs to be loaded into the simulator at a time step when some mode $p \in \Gamma_u$ participates in the measurement of a link operator. Thus at any given time step the simulator only needs to keep track of "active" clusters Γ_u such that at least one mode $p \in \Gamma_u$ has been measured and at least one mode $q \in \Gamma_u$ has not been measured. One can easily choose the order of measurements such that the number of active clusters is $O(n^{1/2})$ at any time step (for example, one can measure link operators column by column). Accordingly, the cost of simulating a single FLO gate is at most $O(n)$. Since the number of gates is $O(n)$, the total simulation cost is $O(n^2)$.

It remains to prove Eq. (19). Let δ be a map from link syndromes to the corresponding face syndromes, that is, $s = \delta(m)$ iff $s_f = \prod_{e\in\partial f} m_e$ for all $f \in F$. By Lemma 1,

$$\overline{\Pi}_s = \sum_{m\,:\,\delta(m)=s} \Pi_s^{\mathsf{link}}. \tag{20}$$

Below we prove the following

Proposition 1. *Suppose m and m' are link syndromes such that $\delta(m) = \delta(m')$. Then there exists a subset of vertices $W \subseteq V$ such that*

$$\Pi^{\mathsf{link}}_{m'} = T\Pi^{\mathsf{link}}_{m}T, \quad \textit{where} \quad T = \prod_{u\in W} S_u.$$

We postpone the proof until the end of the section. Let m and m' be any link syndromes with $\delta(m) = \delta(m')$. Then the proposition implies that

$$\langle\overline{\psi}|\Pi^{\mathsf{link}}_{m'}|\overline{\psi}\rangle = \langle\overline{\psi}|T\Pi^{\mathsf{link}}_{m}T|\overline{\psi}\rangle = \langle\overline{\psi}|\Pi^{\mathsf{link}}_{m}|\overline{\psi}\rangle \tag{21}$$

since $S_u\overline{\psi} = \overline{\psi}$ for all u. Likewise,

$$\langle\overline{\psi}|\Pi^{\mathsf{link}}_{m'}\overline{X}_L|\overline{\psi}\rangle = \langle\overline{\psi}|\Pi^{\mathsf{link}}_{m}\overline{X}_L|\overline{\psi}\rangle \tag{22}$$

since $\overline{X}_L$ commutes with T. From Eqs. (18,20,21,22) one gets

$$b^x_s = \lambda_s \frac{\langle\overline{\psi}|\Pi^{\mathsf{link}}_{m}\overline{X}_L|\overline{\psi}\rangle}{\langle\overline{\psi}|\Pi^{\mathsf{link}}_{m}|\overline{\psi}\rangle} \tag{23}$$

for any link syndrome m such that $\delta(m) = s$. In particular, one can choose m as the link syndrome measured in the first part of the algorithm. Substituting $\overline{X}_L$ from Lemma 2 gives the desired result Eq. (19). It remains to prove Proposition 1.

Proof. By linearity, we can assume that m is the trivial syndrome, that is, $m_e = 1$ for all $e \in E$. Then $\delta(m') = \delta(m)$ iff m' is a flat connection on the surface code lattice, that is, m' is a 1-chain with $\mathbb{Z}_2$ coefficients such that the $\mathbb{Z}_2$-valued magnetic flux through every face is $+1$. Since the lattice is topologically trivial, any flat connection is a co-boundary of some 0-chain $W \subseteq V$. Since a vertex stabilizer S_u flips the link syndromes on all edges incident to u, the condition that m' is a co-boundary of W is equivalent to $\Pi^{\mathsf{link}}_{m'} = T\Pi^{\mathsf{link}}_{m}T$ for $T = \prod_{u\in W} S_u$. □

Numerical results for state preparation

We implemented the algorithm for the simulation of state preparation with translation-invariant coherent noise, and numerically studied surface codes with distance $5 \le d \le 49$. Specifically, we considered preparing the logical basis state $|+_L\rangle$ by performing syndrome measurements on the initial product state

$$(\exp{(i\varphi X)}\exp{(i\theta Z)}|+\rangle)^{\otimes n}.$$

Here $\varphi, \theta \in [0, \pi)$ are noise parameters. The ideal protocol corresponds to $\theta = 0$. Define the logical error rate P^L as the average trace-norm distance between the final logical state $|\phi_s\rangle\langle\phi_s|$ and the target state $|+_L\rangle\langle +_L|$. Equivalently,

$$P^L = 2^{1/2}\sum_s p(s)\sqrt{1 - \langle\phi_s|X_L|\phi_s\rangle}. \tag{24}$$

Since the considered noise model generates correlations between X- and Z-syndromes, we opted not to use the minimum-weight matching decoder (which treats X- and Z-syndromes independently). Instead, we used a simplified decoder that chooses a Pauli correction C_s such that the final logical state ϕ_s always obeys $\langle\phi_s|X_L|\phi_s\rangle \ge 0$. The simplified decoder is optimal in the sense that it minimizes the logical error rate P^L under the constraint that C_s is a Pauli operator. Note that $0 \le P^L \le \sqrt{2}$ for all noise parameters. The symmetries of the surface code imply that P^L, considered as a function of θ and φ, is invariant under transformations

$$\theta \leftarrow \theta + \frac{\pi}{2}, \quad \varphi \leftarrow \varphi + \frac{\pi}{2}, \quad \theta \leftarrow -\theta, \quad \varphi \leftarrow -\varphi.$$

Thus it suffices to simulate the region $0 \le \theta, \varphi \le \pi/4$.

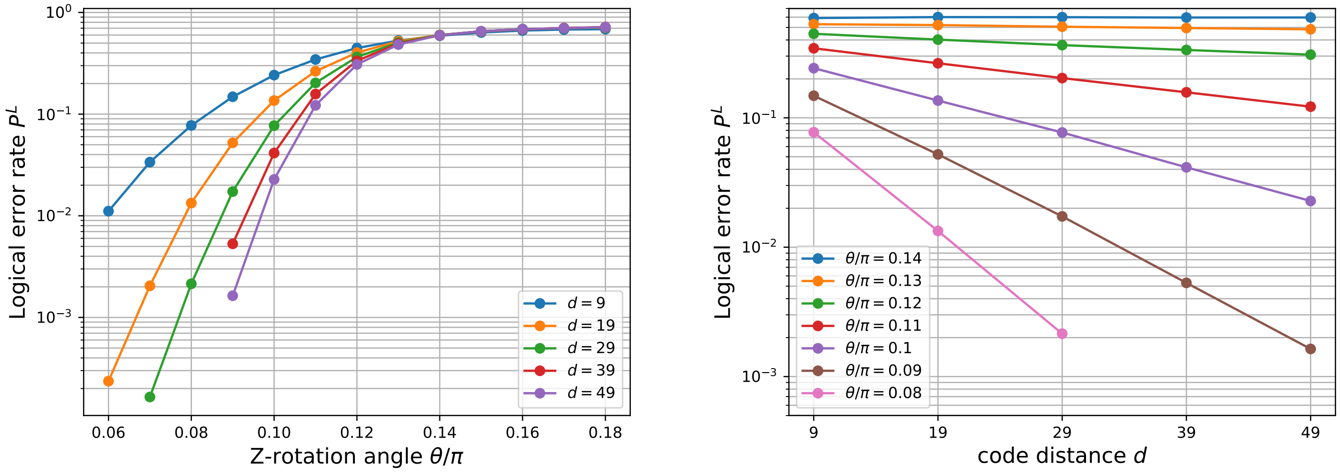
Our numerical results for a fixed code distance $d = 39$ are presented on Supplementary Figure 8. The data supports a natural conjecture that the asymptotic behavior of P^L in the limit $n \to \infty$ can be characterized by a single threshold function $\theta_0(\varphi)$ such that $\lim_{n\to\infty} P^L = 0$ for $\theta < \theta_0(\varphi)$ and P^L is lower bounded by a positive constant independent of n for $\theta > \theta_0(\varphi)$, see also Supplementary Figure 5. From Supplementary Figure 8 one gets an estimate

$$0.1\pi \le \theta_0(\varphi) \le 0.15\pi \tag{25}$$

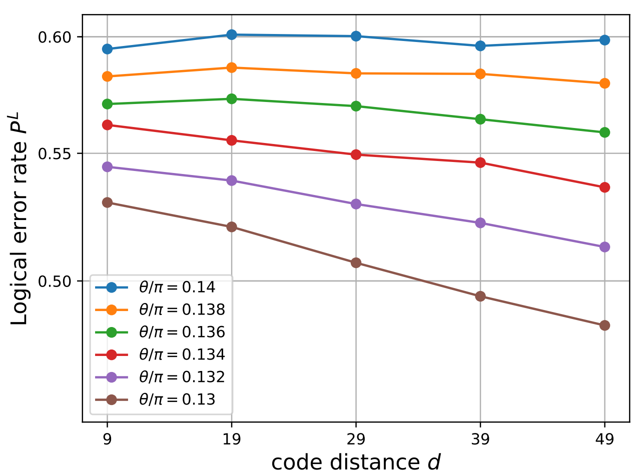
for all φ. The data indicates that $\theta_0(\varphi)$ has a very mild (if any) dependence on φ. To test the above conjecture, we performed more detailed simulations for $\varphi = 0$, see Supplementary Figures 5,6, obtaining a more refined estimate

$$0.13\pi \le \theta_0(0) \le 0.14\pi\ . \tag{26}$$

Eqs. (25) and (26) show that state preparation by measurement of the product state $|+\rangle^{\otimes n}$ and subsequent error correction is indeed robust against coherent errors.



Supplementary Figure 5. Logical error rate P^L for state preparation subject to coherent errors $\exp(i\theta Z)$ on each qubit. For each data point, the quantity P^L was estimated by the Monte Carlo method with $50,000$ syndrome samples.



Supplementary Figure 6. Logical error rate P^L for state preparation, estimated with $50,000$ syndrome samples per data point. Here we only consider angles θ near the threshold and $\varphi = 0$.

SUPPLEMENTARY NOTE 3: SIMULATING STORAGE

Here we present the details of the main algorithm used to simulate storage in the presence of coherent errors. We first prove that the resulting logical state for a fixed syndrome s indeed can be described by a single logical rotation angle θ_s as in Eq. (3) in the main text. We then present algorithms for sampling according to the distribution of syndromes, and for computing this logical rotation angle.

Storage in the toric code

Recall that we consider initial states of the form

$$|\psi\rangle = U|\psi_L\rangle, \qquad U = e^{i\eta_1 Z} \otimes e^{i\eta_2 Z} \otimes \cdots \otimes e^{i\eta_n Z}$$

where ψ_L is some (unknown) logical state of the surface code and $\eta_1, \ldots, \eta_n$ are arbitrary rotation angles. The unitary U describes a coherent error applied to each qubit before the syndrome measurement. Our goal is to sample a syndrome s from the probability distribution

$$p(s) = \langle\psi|\Pi_s|\psi\rangle \tag{27}$$

and compute the final logical state conditioned on the syndrome,

$$|\phi_s\rangle = \frac{1}{\sqrt{p(s)}} C_s \Pi_s |\psi\rangle\ . \tag{28}$$

Clearly, since ψ contains only Z-type errors, the observed syndrome of Z-stabilizers is always trivial. Accordingly, we shall assume that the correction C_s is a Z-type Pauli. We shall need the following fact.

Lemma 4. *The probability $p(s)$ does not depend on the initial logical state ψ_L. The map $\psi_L \to \phi_s$ is a logical Z-rotation by some angle $\theta_s \in [0,\pi)$, that is,*

$$|\phi_s\rangle = \exp\left[i\theta_s Z_L\right]|\psi_L\rangle. \tag{29}$$

We shall refer to θ_s as a *logical rotation angle.* Here and below all states are defined modulo an overall phase factor.

Proof. We have $C_s = Z(h_s)$ for some $h_s \in \{0,1\}^n$. Using the identities $\Pi_s = C_s \Pi_0 C_s$ and $\Pi_0 \psi_L = \psi_L$ one gets

$$p(s) = \|\Pi_0 C_s U \Pi_0 \psi_L\|^2. \tag{30}$$

Expanding the error U in the Pauli basis gives

$$U = \sum_{g\in\{0,1\}^n} \alpha_g i^{|g|} Z(g)$$

for some real coefficients α_g. Here $|g|$ denotes the Hamming weight of a string g. Thus

$$\Pi_0 C_s U \Pi_0 = \sum_{g\in\{0,1\}^n} \alpha_g i^{|g|} \Pi_0 Z(g\oplus h_s)\Pi_0.$$

Note that $\Pi_0 Z(f)\Pi_0 = 0$ unless $Z(f)$ or $Z_L Z(f)$ is a stabilizer of the surface code. Let $\mathcal{A} \subseteq \mathbb{F}_2^n$ be a linear subspace spanned by Z-stabilizers (considered as binary vectors) and let $Z_L = Z(l)$ for some $l \in \{0,1\}^n$. Then

$$\Pi_0 C_s U \Pi_0 = a_s \Pi_0 + b_s \Pi_0 Z_L,$$

where

$$a_s = \sum_{g\in\mathcal{A}\oplus h_s} \alpha_g i^{|g|} \quad \text{and} \quad b_s = \sum_{g\in\mathcal{A}\oplus h_s\oplus l} \alpha_g i^{|g|}.$$

Here $\oplus$ denotes addition of binary strings modulo two. Suppose first that h_s has even weight. Since any element of $\mathcal{A}$ has even weight, the sum that defines a_s runs over even-weight vectors g, that is, a_s is real. Likewise, since l has odd weight, the sum that defines b_s runs over odd-weight vectors g, that is, b_s is imaginary. Define $q_s = |a_s|^2 + |b_s|^2$. Note that q_s does not depend on ψ_L. One arrives at

$$\Pi_0 C_s U \Pi_0 = \sqrt{q_s}\cdot \Pi_0 \exp\left[i\theta_s Z_L\right], \tag{31}$$

where $\theta_s \in [0,2\pi)$ is chosen such that $a_s + b_s = \sqrt{q_s} e^{i\theta_s}$. Since $\exp(i\pi Z_L) = -I$ and overall phase factors do not matter, one can assume $\theta_s \in [0,\pi)$. Substituting Eq. (31) into Eq. (30) gives $p(s) = q_s$, that is, $p(s)$ does not depend on ψ_L, as claimed. Substituting $\Pi_s = C_s\Pi_0 C_s$ into Eq. (28), noting that $C_s^2 = I$, and using Eq. (31) with $q_s = p(s)$ proves Eq. (29).

The case when h_s has odd weight is completely analogous, except that now b_s is real and a_s is imaginary. Choose $\theta_s \in [0,2\pi)$ such that $a_s + b_s = i\sqrt{q_s}e^{i\theta_s}$. Then θ_s obeys Eq. (31) with an extra factor of i on the right-hand side. The rest of the proof is exactly as above. □

Simulating the syndrome measurement

Here we discuss how to sample s from the distribution $p(s)$ specified by (27). By Lemma 4, $p(s)$ does not depend on ψ_L, so below we set $|\psi_L\rangle = |+_L\rangle$. Since only syndromes of X-stabilizers may be non-trivial, it suffices to measure eigenvalues $m_u = \pm 1$ of single-qubit Pauli operators X_u and then classically compute the face syndrome $s_f = \prod_{u\in f} m_u$ for each face f that supports an X-stabilizer. Let $m \in \{+1,-1\}^V$ be the combined X-measurement outcome and $p^x(m)$ be the probability of an outcome m. We have

$$p^x(m) = \langle \psi | \Pi^x_m | \psi\rangle, \qquad \Pi^x_m \equiv \prod_{u\in V} \frac{1}{2}(I + m_u X_u). \tag{32}$$

Let us order qubits column by column as shown on Supplementary Figure 7. Let $p^x_t(m_1,\ldots,m_t)$ be the marginal distribution of $p^x(m)$ describing the first t qubits and let

$$p^x_t(m_t|m_1,\ldots,m_{t-1}) = \frac{p^x_t(m_1,\ldots,m_t)}{p^x_{t-1}(m_1,\ldots,m_{t-1})} \tag{33}$$

be the conditional distribution of m_t given $m_1,\ldots,m_{t-1}$.

Let us show how to sample m_t from the distribution Eq. (33) using FLO gates. Partition the set of all qubits as $[n] = AB$ where $A = \{1,\ldots,t\}$ and $B = [n]\setminus A$. We shall write

$$m_A = (m_1,\ldots,m_t), \quad U_A = \prod_{u\in A} e^{i\eta_u Z_u},$$

and

$$\Pi^x_A = \prod_{u\in A} \frac{1}{2}(I + m_u X_u).$$

Then $p^x_t(m_A) = \langle \psi_L | U_A^\dagger \Pi^x_A U_A |\psi_L\rangle$ Encoding each qubit into the $C4$-code as discussed in Section gives

$$p^x_t(m_A) = \langle \overline{\psi}_L | \overline{U}_A^\dagger \overline{\Pi}^x_A \overline{U}_A |\overline{\psi}_L\rangle. \tag{34}$$

The Majorana representation of the logical state $\overline{\psi}_L$ defined in Lemma 3 gives

$$|\overline{\psi}_L\rangle = \gamma^{1/2} \prod_{u\in V} \frac{1}{2}(I + S_u)|\overline{\psi}_L\rangle \otimes |\phi_{\mathsf{link}}\rangle, \tag{35}$$

where γ is a normalizing coefficient depending only on n, ϕ_{link} is a product state defined in Eq. (7), and $\overline{\psi}_L$ is the basis state of modes c_1,c_2,c_3,c_4 stabilized by ic_1c_2 and ic_3c_4 (recall that we have chosen $|\psi_L\rangle = |+_L\rangle$). The state $|\overline{\psi}_L\rangle \otimes |\phi_{\mathsf{link}}\rangle$ can be prepared using FLO gates since it is stabilized by two-mode operators L_e, ic_1c_2 and ic_3c_4. To simplify the notation, we shall absorb the pairs ic_1c_2 and ic_3c_4 into ϕ_{link}. Accordingly, below we assume that ϕ_{link} is a state of $4n$ modes defined as

$$|\phi_{\mathsf{link}}\rangle\langle\phi_{\mathsf{link}}| = \frac{1}{2}(I + ic_1c_2)\frac{1}{2}(I + ic_3c_4)\prod_{e\in E}\frac{1}{2}(I + L_e). \tag{36}$$

Plugging Eq. (35) into Eq. (34) gives

$$p^x_t(m_A) = \gamma \langle \phi_{\mathsf{link}} | \Omega \overline{U}_A^\dagger \overline{\Pi}^x_A \overline{U}_A \Omega |\phi_{\mathsf{link}}\rangle, \tag{37}$$

where $\Omega = \prod_{u\in V}\frac{1}{2}(I + S_u)$ is the projector onto the codespace of the $C4$-code. Write $\Omega = \Omega_A\Omega_B$, where

$$\Omega_A = \prod_{u\in A}\frac{1}{2}(I + S_u) \quad \text{and} \qquad \Omega_B = \prod_{u\in B}\frac{1}{2}(I + S_u).$$

Since $\overline{U}_A$ and $\overline{\Pi}^x_A$ commute with Ω_A, one gets

$$p^x_t(m_A) = \gamma \langle \phi_{\mathsf{link}} | \overline{U}_A^\dagger (\overline{\Pi}^x_A \Omega_A) \overline{U}_A \Omega_B |\phi_{\mathsf{link}}\rangle. \tag{38}$$

Assume first that $B \neq \emptyset$. Expand the projector Ω_B as

$$\Omega_B = 2^{-|B|} \sum_{C \subseteq B} S_C, \qquad S_C \equiv \prod_{u \in C} S_u.$$

Consider a link operator L_e associated with some edge e such that both endpoints of e belong to B. Such a link operator commutes with all operators acting on A. Note that L_e commutes with all vertex stabilizers S_u except for the two stabilizers located at the endpoints of e. It follows that $L_e S_C L_e = -S_C$ if exactly one end-point of e belongs to C. Furthermore, since ϕ_{link} is stabilized by all link operators one infers that $\langle \phi_{\mathsf{link}} | O_A S_C | \phi_{\mathsf{link}} \rangle = 0$ for any operator O_A unless $C = \emptyset$ or $C = B$. Here we used the fact that B is a connected set for any t, see Fig. 7. Substituting the expansion of Ω_B into Eq. (38) and using the above observation gives

$$p_t^x(m_A) = \gamma' \langle \phi_{\mathsf{link}} | \overline{U}_A^\dagger (\overline{\Pi}_A^x \Omega_A) \overline{U}_A (I + S_B) | \phi_{\mathsf{link}} \rangle, \tag{39}$$

where $\gamma' = 2^{-|B|}\gamma$. Next we note that ϕ_{link} is stabilized by $S_A S_B$ since the latter coincides with the product of all link operators L_e (including ic_1c_2 and ic_3c_4) and ϕ_{link} is stabilized by any link operator. Thus one can replace S_B by S_A in Eq. (39). However, since S_A commutes with $\overline{U}_A$ and $S_A \Omega_A = \Omega_A$, we arrive at

$$p_t^x(m_A) = 2\gamma' \langle \phi_{\mathsf{link}} | \overline{U}_A^\dagger (\overline{\Pi}_A^x \Omega_A) \overline{U}_A | \phi_{\mathsf{link}} \rangle. \tag{40}$$

Using the identity

$$\overline{\Pi}_A^x \Omega_A = \prod_{u \in A} \frac{1}{2}(I + m_u \overline{X}_u) \frac{1}{2}(I + m_u \overline{X}_u S_u)$$

one finally gets

$$p_t^x(m_A) = 2^{t+1-n} \gamma \| G_{2t} G_{2t-1} \cdots G_2 G_1 \phi_{\mathsf{link}} \|^2 \tag{41}$$

where G_{2a-1}, G_{2a} are operators acting non-trivially only on the subset of modes Γ_a, namely

$$G_{2a-1} = \frac{1}{2}(I + m_a \overline{X}_a) e^{i\eta_a \overline{Z}_a},$$

and

$$G_{2a} = \frac{1}{2}(I + m_a \overline{X}_a S_a).$$

Noting that $\overline{Z}_a$, $\overline{X}_a$ and $\overline{X}_a S_a$ have the form $ic_p c_q$ for some $p, q \in \Gamma_a$, see Eqs. (1,2), one concludes that Eq. (41) includes only FLO gates.

The above arguments also apply to the case $B = \emptyset$, that is, $t = n$. The only difference is that now Eq. (39) has no term S_B and thus Eq. (41) becomes

$$p^x(m) = \gamma \| G_{2n} G_{2n-1} \cdots G_2 G_1 \phi_{\mathsf{link}} \|^2. \tag{42}$$

Let us discuss the cost of sampling m from $p^x(m)$. First, observe that any single-qubit marginal state of ψ_L is maximally mixed since the surface code has no stabilizers of weight one. Since the same is true about the state $U|\psi_L\rangle$, one can pick the first measurement outcome m_1 at random from the uniform distribution. Suppose we have already sampled $m_1, \ldots, m_{t-1}$ and the simulator's current state is

$$|\phi_{t-1}\rangle = G_{2t-2} G_{2t-3} \cdots G_2 G_1 |\phi_{\mathsf{link}}\rangle. \tag{43}$$

Plugging Eqs. (41,42) into Eq. (33) gives

$$p_t^x(m_t | m_1, \ldots, m_{t-1}) = \frac{\kappa_t \| G_{2t} G_{2t-1} \phi_{t-1} \|^2}{\| \phi_{t-1} \|^2}, \tag{44}$$

where $\kappa_t = 2$ for $t < n$ and $\kappa_n = 1$. The conditional probability of the outcome $m_t = 1$ can be computed by simulating two more FLO gates G_{2t}, G_{2t-1} starting from the state ϕ_{t-1} which takes time $O(n^2)$. Once the conditional probability is computed, one can sample m_t by tossing a suitably biased coin. This produces the next syndrome m_t together with the next state ϕ_t. After n iterations one gets the desired sample m from $p^x(m)$. Since the above algorithm uses $O(n)$ FLO gates, the simulation runtime scales as $O(n^3)$.

As before (see Supplementary Note 2), we can reduce the runtime to $O(n^2)$ by exploiting the fact that the initial state ϕ_{link} has a product form and by observing that once a pair of modes have been measured, it can be removed from the simulator. Indeed, consider some intermediate step t and let j be the column of the lattice that contains t-th qubit. Then all modes in the columns $j+2,\ldots,d$ can be grouped into unentangled pairs located on edges such that each pair is stabilized by the respective link operator L_e. Such unentangled pairs do not need to be loaded into the simulator. Likewise, all modes in the columns $1,\ldots,j-2$ have already been measured and can be removed from the simulator. Thus at any given time step the number of "active" modes that needs to be simulated is only $O(n^{1/2})$. Accordingly, the cost of simulating a single FLO gate is at most $O(n)$. Since the number of gates is $O(n)$, the total simulation cost is $O(n^2)$.

Remark 1: The same reasoning shows that the probability $p^x(m)$ of any given outcome m can be computed up to the normalizing coefficient γ in time $O(n^2)$ by simulating the FLO circuit defined in Eq. (42). Furthermore, the normalizing coefficient γ depends only on n (but not on the rotation angles η_a).

Remark 2: Below we shall also use a slightly modified version of the above algorithm where the initial state ψ_L is an eigenvector of the logical-Y operator. The modified version is exactly the same as above except that the initial state ϕ_{link} in Eq. (36) is defined as

$$|\phi_{\mathsf{link}}\rangle\langle\phi_{\mathsf{link}}| = \frac{1}{2}(I - ic_1c_3)\frac{1}{2}(I + ic_2c_4)\prod_{e\in E}\frac{1}{2}(I + L_e). \tag{45}$$

This corresponds to initializing the unpaired modes in the logical-Y state.

Computing the logical rotation angle

It remains to show how to compute the logical rotation angle θ_s for a given syndrome s. Let us first initialize the logical qubit in the X-basis, that is, we choose $|\psi_L\rangle = |+_L\rangle$. Let ϕ_s be the final logical state defined in Eq. (28). Define logical amplitudes

$$A_s^+ = \langle +_L|\phi_s\rangle = \cos(\theta_s) \tag{46}$$

and

$$A_s^- = \langle +_L|Z_L|\phi_s\rangle = i\sin(\theta_s). \tag{47}$$

Here we used Lemma 4. Then

$$\tan^2(\theta_s) = \left|\frac{A_s^-}{A_s^+}\right|^2. \tag{48}$$

Using Eq. (28) and the identity $\Pi_s = C_s\Pi_0C_s$ one gets

$$\tan^2(\theta_s) = \frac{|\langle +_L|Z_LC_sU|+_L\rangle|^2}{|\langle +_L|C_sU|+_L\rangle|^2}. \tag{49}$$

Let $U_+ = C_sU$ and $U_- = Z_LC_sU$. By definition, $U_\pm$ are products of single-qubit Z rotations:

$$U_\pm = \prod_{u=1}^n e^{i\eta_u^\pm Z_u}.$$

Let us expand the logical state $|+_L\rangle$ in the Z-basis:

$$|+_L\rangle = |\mathcal{L}|^{-1/2}\sum_{x\in\mathcal{L}}|x\rangle,$$

where $\mathcal{L}$ is the set of basis states $x\in\{0,1\}^n$ that obey Z-stabilizers of the surface code (we do not need an explicit formula for $\mathcal{L}$). Since $U_\pm$ is diagonal in the Z-basis,

$$\langle +_L|U_\pm|+_L\rangle = 2^{n/2}|\mathcal{L}|^{-1/2}\langle +^{\otimes n}|U_\pm|+_L\rangle. \tag{50}$$

Substituting this into Eq. (48) gives

$$\tan^2(\theta_s) = \frac{|\langle +^{\otimes n}|U_-|+_L\rangle|^2}{|\langle +^{\otimes n}|U_+|+_L\rangle|^2} \equiv \frac{p_-}{p_+}. \tag{51}$$

Since $U_\pm$ is a tensor product of Z-rotations, $p_\pm$ is a special case of the probability $p^x(m)$ defined in Eq. (32) with $m_u = 1$ for all u. We have already shown that one can compute $\gamma^{-1} p_\pm$ in time $O(n^2)$, where γ depends only on n, see Remark 1 at the end of the previous section. Thus $\tan^2(\theta_s) = p_-/p_+$ can be computed in time $O(n^2)$.

Next let us initialize the logical qubit in the Y-basis state: $|\psi_L\rangle = |Y_L\rangle$, where $|Y\rangle \equiv (|0\rangle + i|1\rangle)/\sqrt{2}$. Define logical amplitudes

$$B_s^+ = \langle +_L|\phi_s\rangle = e^{i\pi/4}(\cos(\theta_s) + \sin(\theta_s)) \tag{52}$$

and

$$B_s^- = \langle +_L|Z_L|\phi_s\rangle = e^{-i\pi/4}(\cos(\theta_s) - \sin(\theta_s)). \tag{53}$$

The same arguments as above show that

$$\tan^2(\theta_s - \pi/4) = \frac{|\langle +^{\otimes n}|U_-|Y_L\rangle|^2}{|\langle +^{\otimes n}|U_+|Y_L\rangle|^2} \equiv \frac{q_-}{q_+}. \tag{54}$$

Since $U_\pm$ is a product of Z-rotations, $q_\pm$ is a special case of the probability $p^x(m)$ defined in Eq. (32) with $m_u = 1$ for all u and the initial state ψ_L chosen as a logical Y-basis state. We have already shown that one can compute $q_\pm$ in time $O(n^2)$ see Remarks 1,2 at the end of the previous section.

Thus $\tan^2(\theta_s - \pi/4) = q_-/q_+$ is computable in time $O(n^2)$. Combining Eqs. (51,54) gives

$$\cos(2\theta_s) = \frac{p_+ - p_-}{p_+ + p_-} \quad \text{and} \quad \sin(2\theta_s) = \frac{q_+ - q_-}{q_+ + q_-}. \tag{55}$$

This determines the logical rotation angle modulo π.

SUPPLEMENTARY NOTE 4: FERMIONIC LINEAR OPTICS

Here we state some necessary facts on simulation of fermionic linear optics. The material of this note is based on Refs. [2–4]. Let $\mathcal{P}_n$ be the group generated by single-qubit Pauli operators X_j, Y_j, Z_j with $j = 1, \ldots, n$. Define Majorana operators $c_1, \ldots, c_{2n} \in \mathcal{P}_n$ such that $c_1 = Y_1$, $c_2 = X_1$,

$$c_{2j-1} = Z_1 \cdots Z_{j-1} Y_j \quad \text{and} \quad c_{2j} = Z_1 \cdots Z_{j-1} X_j \tag{56}$$

for $2 \le j \le n$. They obey commutation rules

$$c_p c_q = -c_q c_p \quad \text{for } p \ne q. \tag{57}$$

More generally, given a bit string $x \in \{0,1\}^{2n}$, define a Majorana monomial $c(x) = c_1^{x_1} c_2^{x_2} \cdots c_{2n}^{x_{2n}}$. Then

$$c(x)c(y) = (-1)^{|x \cdot y|} c(y)c(x) \tag{58}$$

whenever at least one of the strings x,y has even weight. Any n-qubit operator can be uniquely expressed as a linear combination of the 4^n Majorana monomials $c(x)$.

Suppose ρ is a (mixed) n-qubit state. The *covariance matrix* of ρ is a real anti-symmetric matrix M of size $2n \times 2n$ defined by

$$M_{p,q} = \begin{cases} \mathrm{Tr}(i c_p c_q \rho) & \text{if } p \ne q \\ 0 & \text{if } p = q \end{cases} \tag{59}$$

A state ρ is called *Gaussian* iff ρ is a linear combination of only even-weight Majorana monomials $c(x)$, and the expectation value of $c(x)$ on ρ can be computed from the covariance matrix M using Wick's theorem. For example, we require that

$$-\mathrm{Tr}(c_p c_q c_r c_s \rho) = M_{p,q} M_{r,s} - M_{p,r} M_{q,s} + M_{p,s} M_{q,r} \tag{60}$$

for all $p \neq q \neq r \neq s$. More generally, we require that

$$\mathrm{Tr}(i^{|x|/2} c(x)\rho) = \mathrm{Pf}(M[x]) \tag{61}$$

for all even-weight $x \in \{0,1\}^{2n}$, where $M[x]$ is a submatrix of M including only rows and columns p with $x_p = 1$ and Pf denotes the Pfaffian. In the present paper we only use the special case Eq. (60). To summarize, a Gaussian state can be fully specified by its covariance matrix.

It is well known that FLO gates defined in the methods section preserve the class of Gaussian states. Thus a quantum circuit composed of FLO gates acting on some initial Gaussian state can be efficiently simulated if we know how to update the covariance matrix under the action of each gate. These update rules are stated below.

Consider a pair of modes $p, q \in [1, 2n]$ and let

$$\Lambda = (1/2)(I + ic_p c_q). \tag{62}$$

Note that Λ is a projector. It describes a post-selective parity measurement for the pair of modes p, q with the measurement outcome $+1$ (note that the outcome -1 can be obtained simply by exchanging p and q).

Fact 1. *If ρ is a Gaussian state with covariance matrix M then $\rho' = \Lambda\rho\Lambda/\mathrm{Tr}(\Lambda\rho)$ is a Gaussian state with a covariance matrix M' that can be computed in time $O(n^2)$ by the following algorithm:*

```
function MEASURE(M, p, q)
    λ ← (1/2)(1 + M_{p,q})
                                          ▷ probability of the outcome +1
    if λ ≠ 0 then
        K ← p-th column of M
        L ← q-th column of M
        M' ← M + (2λ)^{-1}(K L^T − L K^T)
        Set to zero rows and columns p, q of M'
        M'_{p,q} ← 1
        M'_{q,p} ← −1
        return (λ, M')
    end if
end function
```

We note that the final matrix M' has a block structure such that the modes p, q are not coupled to any other modes. Thus one can remove rows and columns p, q from M' without losing any information.

Next let us discuss two-mode rotations

$$U = \exp(\gamma c_p c_q). \tag{63}$$

Here $\gamma \in [0, \pi)$ is the rotation angle. One can check that U is a unitary operator such that the action of U in the Heisenberg picture is

$$\begin{aligned} U^\dagger c_p U &= \cos(2\gamma) c_p + \sin(2\gamma) c_q, \\ U^\dagger c_q U &= -\sin(2\gamma) c_p + \cos(2\gamma) c_q, \\ U^\dagger c_r U &= c_r, \quad \text{if } r \notin \{p, q\}\ . \end{aligned} \tag{64}$$

We shall describe the transformation Eq. (64) by an orthogonal matrix $R \in SO(2n)$ such that

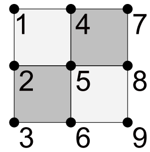
$$U^\dagger c_t U = \sum_{s=1}^{2n} R_{t,s} c_s, \qquad 1 \le t \le 2n.$$

Fact 2. *If ρ is a Gaussian state with a covariance matrix M then $\rho' = U\rho U^\dagger$ is a Gaussian state with a covariance matrix $M' = RMR^T$ that can be computed in time $O(n)$.*

Finally, the class of Gaussian states is closed under the tensor product operation.

Fact 3. *Let ρ_i be a Gaussian state of n_i qubits with a covariance matrix M_i. Here $i = 1, 2$. Then $\rho_1 \otimes \rho_2$ is a Gaussian state of $n_1 + n_2$ qubits with a covariance matrix*

$$M_1 \oplus M_2 \equiv \begin{bmatrix} M_1 & 0 \\ 0 & M_2 \end{bmatrix}. \tag{65}$$



Supplementary Figure 7. The ordering of qubits used in the simulation of the syndrome measurement: Qubits are ordered column by column (see Supplementary Note 3).

All our algorithms work by simulating a sequence of two-mode parity measurements and two-mode rotations starting from a certain simple initial Gaussian state. To describe these initial states we need two more facts.

Fact 4. *Let ψ be a pure single-qubit state with a Bloch vector $\vec{b} = (b^x, b^y, b^z)$. Let $\overline{\psi}$ be the $C4$-encoding of ψ defined in Eqs. (1,2). Then $\overline{\psi}$ is Gaussian with a covariance matrix*

$$M = \begin{bmatrix} 0 & b^x & -b^y & b^z \\ -b^x & 0 & b^z & b^y \\ b^y & -b^z & 0 & b^x \\ -b^z & -b^y & -b^x & 0 \end{bmatrix}. \tag{66}$$

We note that Fact 4 does not apply to mixed states: Computing the expectation value of the $C4$-stabilizer $ic_1c_2c_3c_4$ using Wick's theorem reveals that a Gaussian state with covariance matrix of the form (66) only is $C4$-encoded if $\|\vec{b}\| = 1$. As a corollary, any tensor product of single-qubit states encoded by the $C4$-code is a Gaussian state whose covariance matrix is a direct sum of 4×4 matrices defined in Eq. (66).

By analogy with stabilizer states, certain Gaussian states can be defined by an abelian group of Pauli stabilizers. Namely, suppose $\sigma : [2n] \to [2n]$ is a permutation.

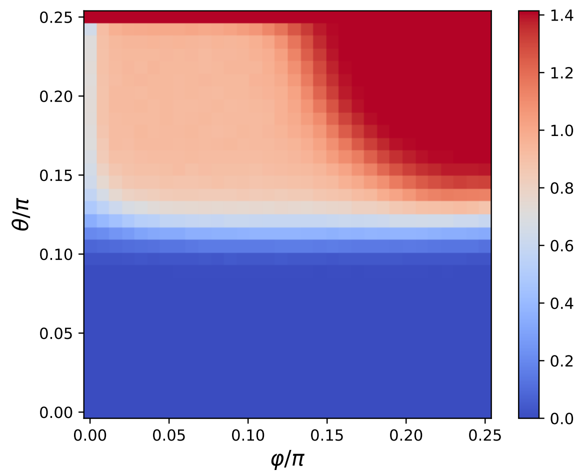
Fact 5. *There exists a unique Gaussian state $\rho = |\phi\rangle\langle\phi|$ such that $ic_{\sigma(2j-1)}c_{\sigma(2j)}|\phi\rangle = |\phi\rangle$ for all $j = 1, \ldots, n$. The state ρ has a covariance matrix*

$$M_{r,s} = \sum_{j=1}^{n} \delta_{r,\sigma(2j-1)}\delta_{s,\sigma(2j)} - \delta_{r,\sigma(2j)}\delta_{s,\sigma(2j-1)}. \tag{67}$$

The above facts are sufficient to simulate any quantum circuit composed of FLO gates, as defined in the methods section, and provide all details necessary for implementation of our algorithms A and B.

SUPPLEMENTARY REFERENCES

[1] Austin G. Fowler, Ashley M. Stephens, and Peter Groszkowski, "High-threshold universal quantum computation on the surface code," Phys. Rev. A **80**, 052312 (2009).
[2] Barbara M. Terhal and David P. DiVincenzo, "Classical simulation of noninteracting-fermion quantum circuits," Phys. Rev. A **65**, 032325 (2002).
[3] Sergey Bravyi, "Lagrangian representation for fermionic linear optics," Quant. Inf. Comp. **5**, 216–238 (2004).
[4] Sergey Bravyi and Robert Koenig, "Classical simulation of dissipative fermionic linear optics," Quant. Inf. Comp. **12**, 925 (2012)



Supplementary Figure 8. Preparation of the logical basis state $|+_L\rangle$ by performing syndrome measurements on the initial product state $(\exp(i\varphi X)\exp(i\theta Z)|+\rangle)^{\otimes n}$. The color represents the logical error rate P^L defined in Eq. (24). The dark blue and red regions represent a "fault-tolerant" phase where the final logical state is close to $|+_L\rangle$ and $|0_L\rangle$ respectively. Here we consider a fixed code distance $d = 39$. The plot was generated by sampling 5,000 syndromes for each pair (θ, φ).