

Lattice surgery with Bell measurements: Modular fault-tolerant quantum computation at low entanglement cost

Trond Hjerpekjøn Haug,^{*} Timo Hillmann, Anton Frisk Kockum, and Raphaël Van Laer[†]

*Department of Microtechnology and Nanoscience,
Chalmers University of Technology, 412 96 Gothenburg, Sweden*

(Dated: October 16, 2025)

Modular architectures are a promising approach to scaling quantum computers to fault tolerance. Small, low-noise quantum processors connected through relatively noisy quantum links are capable of fault-tolerant operation as long as the noise can be confined to the interface. Finding protocols that implement the quantum links between modules as efficiently as possible is essential because inter-module entanglement is challenging to produce at a similar rate and fidelity as local entanglement. We introduce a protocol for lattice surgery on surface codes in which all non-local operations are Bell measurements. The protocol simultaneously confines the link noise and requires only half as many module-crossing gates as previously proposed protocols. To mitigate distance-reducing hook errors, we introduce a strategy of alternating the gate sequence between rounds of syndrome measurement, which prevents multiple hooks from simultaneously aligning with a logical operator in the code. We evaluate our protocol's performance when two logical qubits on separate modules are prepared in a logical Bell state. Circuit-level simulations under depolarizing noise show that the logical error suppression for a given entanglement rate between modules is consistently stronger compared to the best-performing alternative protocols for a wide range of link noise, with a typical $\sim 40\%$ entanglement resource saving for a constant logical error rate. Our approach to protocol design is applicable to any quantum circuit that must be divided across processor modules and can therefore guide development of resource-efficient modular quantum computation beyond the surface code.

I. INTRODUCTION

Quantum computation and communication offer new opportunities that are believed to go beyond what can be achieved classically [1–4]. However, to unlock these capabilities, protecting quantum information from the noise of physical hardware components is necessary. This need has motivated the development of quantum error correction and fault tolerance, which introduce redundancy in the storage and manipulation of quantum information to protect it against noise.

Scaling up the number of physical components to meet the demands of fault tolerance creates challenges for hardware platforms currently in development. For example, superconducting qubits [5, 6] are currently cryogenically cooled and controlled using cables connected to room-temperature electronics. Increasing the number of qubits, therefore, increases the demand for cooling power from cryostats, which often is in short supply [7, 8]. Moreover, fabrication imperfections and crosstalk limit the precision of qubit operations, and the probability of critical defects increases rapidly with system size [9]. Other platforms, e.g., neutral atoms [10] and trapped ions [11], do not suffer in the same way from fabrication variations but are similarly susceptible to crosstalk. Furthermore, the requirement for chambers with ultra-high vacuum conditions ($\sim 10^{11}$ torr) to accommodate the qubits in these platforms poses a significant obstacle to scalability [12].

Modular architectures have been proposed as a solution to overcome the challenges faced by monolithic systems [13–18]. In such architectures, small processors with low levels of noise can be connected through quantum links that allow them to operate as a single entity in a network configuration. Depending on the size of each module, it has been found that the noise suffered by the links can be about an order of magnitude higher than local levels without significantly compromising the logical error suppression of surface codes [19]. The resilience to noise at the interface suggests that focus can partly shift from solving the scaling challenges in monolithic architectures to establishing relatively error-prone quantum links between modules. To prevent bottlenecks, the bandwidth of these inter-module quantum links should be comparable to the internal clock rate of the quantum processor modules. Links can be short-range interconnects that support direct gates between modules [20–22] or long-range connections for distribution of entanglement using itinerant photons [23–29].

In this work, we consider a modular architecture of quantum processing units (QPUs) where each unit supports at least one surface-code logical qubit. The QPUs are connected through quantum links that enable either direct gates or the distribution of Bell pairs between modules. An example architecture is shown in Fig. 1(a). Operations on logical qubits between separate modules are performed using lattice surgery [30, 31]—a popular method for executing logical operations on topological codes like the surface code [32, 33].

We define a “patch” as the group of physical qubits that collectively support one surface-code logical qubit. By merging two or more patches at their boundaries, we

^{*} trond.haug@chalmers.se

[†] raphael.van.laer@chalmers.se

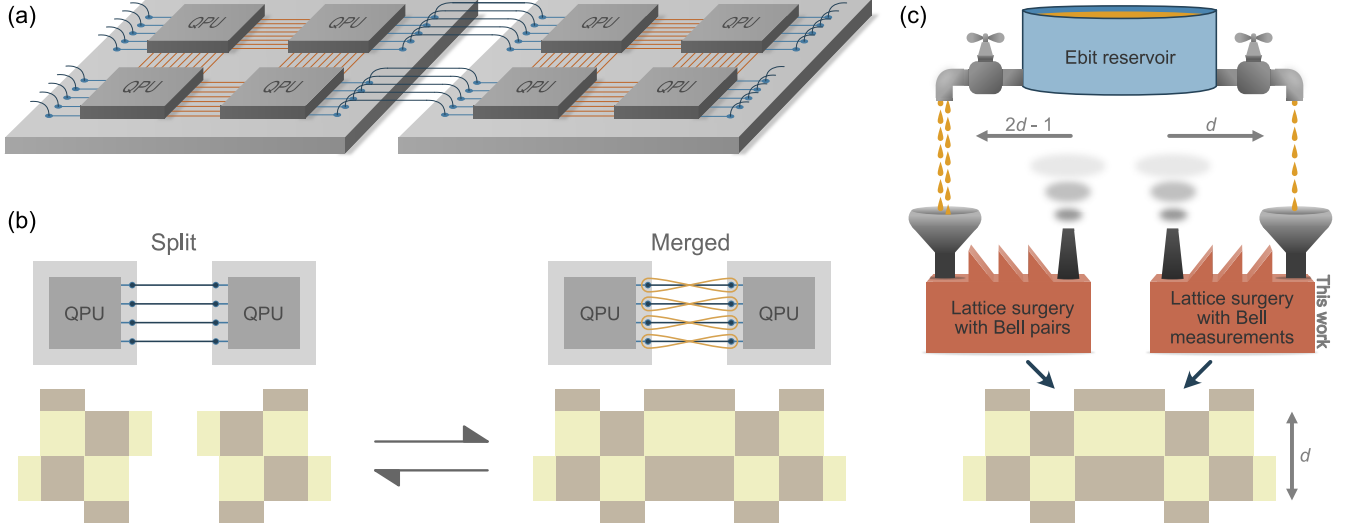


FIG. 1. Modular quantum-computing architectures and lattice surgery on surface codes. (a) An example modular architecture for a superconducting quantum computer with four quantum processing units (QPUs) connected by direct links (orange) through an interposer chip. Each QPU is also equipped with photonic connections (blue) that support entanglement distribution with itinerant photons to a QPU on another module. (b) Lattice surgery with surface-code logical qubits on separate modules connected by quantum links. Dark (light) squares represent $X(Z)$ -stabilizers of the surface code. When patches are merged, physical qubits on separate modules interact by consuming entanglement that must be established between modules over the quantum links. (c) The rate of entanglement consumption depends on the protocol that is used for the surgery. Previous protocols have required $2d - 1$ Bell pairs per round of stabilizer measurement in a code of distance d . Each Bell pair has one ebit of entanglement [34]. We introduce a protocol for lattice surgery based on Bell measurements, which only requires d ebits per round of syndrome measurement.

can measure a product of logical operators that are encoded within the patches. This merging of patches entangles the physical qubits from one patch with those of the other patch [Fig. 1(b)]. While the patches are merged, products of Pauli operators on each patch can be inferred from the measurement outcomes of the new stabilizers with support on both patches. After a product of logical operators has been measured fault-tolerantly, the merged patches are split so that logical qubits are free to take part in the next step of the algorithm. Splitting is performed by measuring a subset of the physical qubits in the merged patch.

Recent studies of rotated surface codes in modular architectures have introduced protocols for merging patches across an interface [35, 36]. A protocol called “cat state gadget” in Ref. [35] and “measurement teleportation” in Ref. [36] uses Bell pairs to measure stabilizers at the interface. It can be implemented with only d Bell pairs per round of syndrome measurements, where d is the distance of the surface code, at the cost of cutting the effective code distance at the interface in half. This distance reduction, as well as the noise propagation from the Bell pairs to data qubits, implies that the protocol is impractical. Full distance in the protocol can be restored by increasing the number of Bell pairs per round to $2d - 1$. Distribution of high-fidelity Bell pairs between processor modules is currently a difficult and slow process [29, 37–39], which suggests that Bell pairs are a valu-

able resource that should be used sparingly. However, to our knowledge, there is no protocol for inter-module lattice surgery with an effective distance at the interface larger than $(d + 1)/2$ which requires less than $2d - 1$ Bell pairs per round of syndrome measurement.

Here, we develop such a protocol based on Bell measurements [Fig. 1(c)]. A Bell measurement is a projective measurement of two qubits onto a Bell state. It measures the observables X_1X_2 and Z_1Z_2 simultaneously, where X_i and Z_i are Pauli operators on the i^{th} qubit. We refer the reader to Appendix A for details on Bell measurements and their implementation with quantum circuits. Bell measurements are ubiquitous in discrete-value optical quantum technologies, where they are often referred to as fusion gates [40]. The fusion-based model of quantum computation [41] employs Bell measurements on entangled resource states. The key insight is that Bell measurements—like multibody entangling measurements on Bell pairs [42]—create the entanglement that is required by the code, as well as the syndrome information to detect errors [43]. We use this insight to construct a gate-based protocol for lattice surgery between modules in which Bell measurements replace the Bell pairs used in Refs. [35, 36]. We also introduce a technique for controlling distance-reducing errors at the interface by alternating the syndrome-measurement circuit between rounds of syndrome measurement. Finally, using a realistic circuit-level noise model, we compare the perfor-

mance of the Bell-measurement protocol to a benchmark protocol from Ref. [35]. For physically reasonable parameters, we find that using Bell measurements reduces the required entanglement rate between modules by $\sim 40\%$ compared to the benchmark protocol for a constant logical error rate. Our results demonstrate that the Bell-measurement protocol achieves a stronger logical error suppression than previously proposed protocols when access to high-fidelity inter-module entanglement is limited.

This paper is organized as follows. In Section II, we introduce the benchmark and Bell-measurement protocols using the language of ZX calculus. We explain in detail the effects of using Bell measurements for lattice surgery on rotated surface codes in a circuit-based implementation. In Section III, we address the problem of distance reduction at the interface, which requires an alternating syndrome-measurement circuit to mitigate. Numerical studies of the performance of the Bell-measurement protocol are presented in Section IV and compared to the benchmark protocol under the same levels of noise. We conclude and present an outlook for the Bell-measurement protocol in Section V.

II. PROTOCOL DESIGN AND EFFECTS OF BELL MEASUREMENTS

A well-designed protocol for inter-module lattice surgery should have the following properties. (i) The protocol should ensure that noise from operations at the interface remains confined to the interface. Two-qubit gates applied locally can propagate errors that originate at the noisy interface to the bulk. This complicates error tracking and could lead to a reduction in the effective code distance at the interface. (ii) Operations at the interface should not interfere with operations in the bulk. For example, if the gate sequence at the interface for a round of syndrome measurement is significantly longer than the corresponding sequence in the bulk, the added circuit depth slows down the logical speed of the processor and allows more noise to accumulate before each syndrome measurement. (iii) If direct gates between physical qubits are not supported by the architecture and access to high-fidelity Bell pairs is limited, the protocol should involve as few teleported gates as possible at each round of syndrome measurement. Each teleported gate consumes one Bell pair that must be established between modules before the gate can be applied.

One possible protocol is to keep the same gate sequence at the interface as in the bulk. Each weight-4 stabilizer at the interface is measured using four CX or CZ gates between the data qubits and a single ancilla qubit. The ancilla qubit can be in either of the two modules. Since two data qubits will be in the module without the ancilla qubit, two of the four gates must be applied across an interface and will therefore introduce elevated noise. This protocol avoids distance-reducing errors, also known as hook errors, and does not interfere with the gate schedule

in the bulk. However, the noisy interface-crossing gates will increase the error rate on data qubits at the interface. Such errors can subsequently propagate through local two-qubit gates. If the architecture does not allow the interface-crossing gates to be implemented directly, the protocol needs $2d - 1$ Bell pairs per round of syndrome measurement to teleport the gates. We refer to this protocol as the benchmark protocol. It was found to be the protocol that achieved the lowest logical error rate of those studied in Ref. [35].

A protocol for lattice surgery based on Bell measurements cannot easily be understood in terms of a code at a single instance in time. It is more insightful to take a holistic view of the structure that emerges as the code evolves through time [44], also referred to as the code in space-time. In this picture, outcomes from Bell measurements at different points in space and time are combined to form detectors. A detector is a set of binary measurement outcomes $D = \{m_i\}$ whose parity is deterministic in the absence of errors [45]. An error that anticommutes with a measurement used to form a detector will flip the detector's parity. In the surface code, a single-qubit X or Z error flips the parity of at most two detectors. A Y error is equivalent to a combined X - and Z error and thus flips the parity of four detectors.

Identifying detectors in the surface code is not trivial when teleported gates and Bell measurements are used. We therefore employ the language of ZX calculus [46], which is useful for reasoning about error correction and fault tolerance [47, 48]. We use the convention that red vertices represent X-spiders and green vertices represent Z-spiders. Each spider is associated with a set of stabilizer generators. For an n -legged Z-spider, the generators are $\{X_1 X_2 \cdots X_n, Z_1 Z_i\}$ for all positive integers $i \leq n$. Analogously, the stabilizer generators associated with an n -legged X-spider are $\{Z_1 Z_2 \cdots Z_n, X_1 X_i\}$. Detectors are identified in a ZX diagram when the stabilizer generators associated with the spiders combine to form the identity. A ZX diagram that produces the surface-code detectors for phase flips is shown in Fig. 2(a). Each Z-spider (green) has a stabilizer $X_1 X_2 X_3 X_4$, and each X-spider (red) has a stabilizer $X_1 X_2$. Since each Z-spider is connected to an X-spider on all four legs and the network forms a closed cell, the stabilizers for the full structure combine to the identity. If a Z error occurs on one of the qubits used to construct the ZX diagram in Fig. 2(a), this corresponds to inserting a two-legged Z-spider with a π phase in the diagram. The addition of this spider changes the stabilizer product of the full structure from I to $-I$. This change manifests as a flipped measurement outcome, which changes the parity of measurements in the detector.

The transformation rules of ZX calculus allow spiders to be split and merged, which produces new but equivalent ZX diagrams. Equivalent diagrams have the same stabilizers and therefore represent the same detectors. When an interface needs to be introduced that separates two modules, we use this freedom to modify the ZX dia-

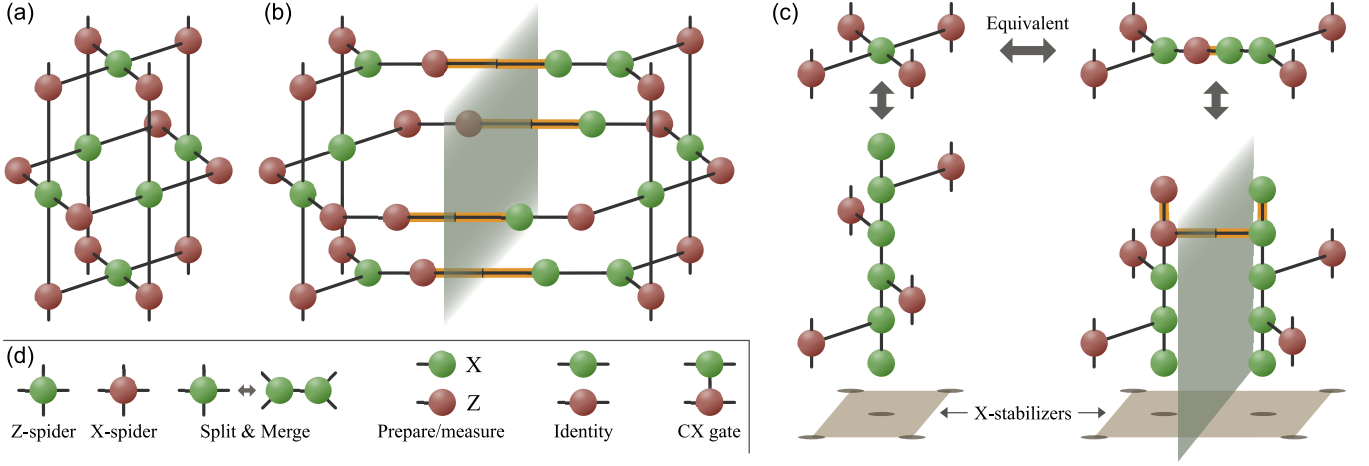


FIG. 2. ZX diagrams for detection of phase flips in the surface code. (a) The simplest ZX diagram in which all spiders are connected to spiders of the opposite type. (b) At the interface between two modules, we split spiders in the diagram so that the number of legs crossing the interface is four. We then insert two-legged spiders of opposite type on each side of the interface to illustrate that links between the two sides of the interface can be implemented with CX gates as part of a Bell measurement. These links are highlighted in orange for clarity. (c) The ZX diagrams can be further expanded to give intuition about how they are implemented with quantum circuits on physical qubits. On the left is an expansion that produces the quantum circuit for measuring surface-code X-stabilizers in the bulk. On the right is the corresponding expansion for measuring the same stabilizer at the interface. (d) Elementary building blocks for ZX diagrams and their interpretations in terms of operations on physical qubits.

gram as shown in Fig. 2(b). Spiders at the interface are split and new, phaseless two-legged spiders—which are equivalent to the identity—of opposite type are inserted on either side of the interface. The result is a diagram where four edges crossing the interface connect spider nodes of opposite type.

The ZX diagrams in Fig. 2(a)-(b) can be interpreted as a series of operations on qubits through time. We associate time with the vertical direction in the diagrams. In this picture, vertical connections between spider nodes represent qubit worldlines, and horizontal connections represent gates between qubits. Specifically, each horizontal connection between an X-spider and a Z-spider represents a CX gate between qubits. According to our design criteria, a protocol should minimize the number of gates that cross the interface between modules. Detectors like the one shown in Fig. 2(b) have four connections across the interface, but since each connection is part of two detectors, the average number of interface-crossing gates per detector is two. This is half as many as the benchmark protocol, and we conjecture that it is, in fact, optimal.

In Fig. 2(c), we take the top part of the surface-code ZX diagrams and apply further spider splits to show how the diagram can be compiled into a quantum circuit. In the bulk, the diagram is compiled into the usual circuit for measuring stabilizers in the surface code: an ancilla qubit is prepared in the X basis, then undergoes CX gates with its neighboring data qubits before it is measured in the X basis. At the interface, the diagram is compiled into a circuit with two ancilla qubits that undergo CX gates with neighboring data qubits and an

interface-crossing CX gate between the ancillae. The ZX diagrams do not specify the direction time flows, i.e., whether the interface-crossing CX gate should be applied before or after the local CX gates. In the diagram on the left in Fig. 2(c), it does not matter which way time flows since the diagram is invariant with respect to this choice. At the interface, however, the choice gives two different interpretations. If time is chosen to flow downward in Fig. 2(c) so that the interface-crossing gate is applied before local gates, this produces the circuit in the measurement-teleportation protocol from Ref. [36]. If time is chosen to flow upward so that local gates are applied before the interface-crossing CX gate, this produces a circuit where the final operation on the ancilla qubits is a Bell measurement. Both circuits implement the surface code, but the Bell-measurement protocol is preferred because noise from the interface cannot propagate to the bulk.

Bell measurements produce twice as many measurement outcomes as single-ancilla measurements. This increases the number of measurements in detectors at the interface. Detectors in the bulk of the surface code are formed from only two measurements: one measurement outcome from each of two subsequent rounds of syndrome measurement. Detectors at the interface are formed from one measurement outcome from each of the four Bell measurements necessary to construct the ZX diagram in Fig. 2(b). An error at the interface will flip the outcome of the Bell measurement where the error occurred. Two detectors that share a measurement outcome which can be flipped by a single error are said to share a *fault mechanism*. In the syndrome graph, detectors correspond to

vertices and fault mechanisms correspond to edges. Bell measurements introduce two new edges for each vertex in the syndrome graph. The valence of the syndrome graph at the interface is eight under phenomenological noise, compared to six for the bulk. The new edges are in the same plane as the interface between modules; they are diagonal with both a space- and a time-like direction. Edges from Bell measurements decrease the code distance at the interface if they are allowed to form uninterrupted chains in the syndrome graph. This is similar to how hook errors under circuit-level noise form diagonal edges in the syndrome graph [36]. The orientation of diagonal edges due to hook errors is set by the order of gates in the syndrome-measurement circuit. This is true for diagonal edges due to Bell measurements as well. Hence, the choice of syndrome-measurement circuit effectively determines the distance of the surface code at the interface between modules.

III. ALTERNATING SYNDROME-MEASUREMENT CIRCUITS

The syndrome-measurement circuit for rotated surface codes is designed to avoid propagation of errors through two-qubit gates that would reduce the distance of the code. The order of two-qubit gates between ancilla qubits and data qubits is required to follow a pattern in which the last two gates cannot propagate an error on the ancilla qubit to data qubits aligned parallel to a logical operator [49]. This restricts the viable choices of gate sequence to two, both of which follow a Z-N pattern across a stabilizer. We use CX gates for X-stabilizer measurements and CZ gates for Z-stabilizer measurements for clarity. One of the two distance-preserving measurement sequences is illustrated in Fig. 3. We assign the label A to this gate sequence. The alternative sequence is obtained by swapping the numbers $1 \leftrightarrow 3$ and $2 \leftrightarrow 4$ for CX gates in the bulk, and $1 \leftrightarrow 2$ and $3 \leftrightarrow 4$ for CZ gates in the bulk. We assign the label B to this gate sequence.

The Bell-measurement protocol for lattice surgery does not require any changes to the standard syndrome-measurement circuit in the bulk. The gate sequence at the interface is chosen to match the circuit used to measure syndromes in the bulk, as shown in Fig. 3. Interface-crossing CX gates that implement Bell measurements are applied after the local gates. The gate sequence is symmetric about the interface to avoid qubits idling unnecessarily. Note that the Bell measurement can be executed immediately after the local gates have been applied. The gate sequence at the interface that matches the alternative sequence in the bulk is obtained by swapping the numbers $1 \leftrightarrow 3$ for local CX gates and $1 \leftrightarrow 2$ for local CZ gates on plaquettes with an interface-crossing CX gate in Fig. 3.

The full syndrome graph of a surface code under circuit-level noise is a three-dimensional graph that accounts for all Pauli errors. An example of a syndrome

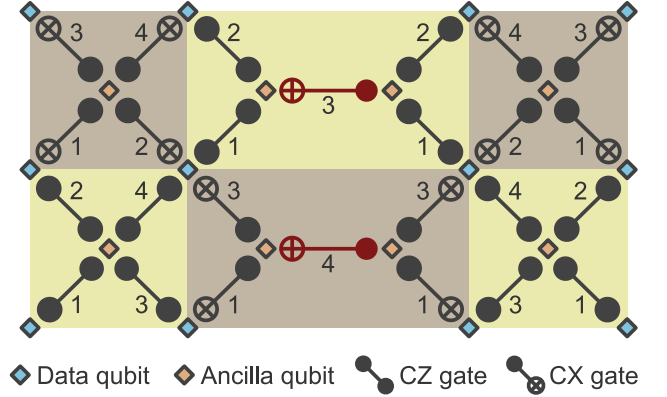


FIG. 3. Syndrome-measurement circuits for a rotated surface code. Light (dark) plaquettes represent Z(X)-stabilizers. The syndrome-measurement circuit in the absence of an interface is shown on the square plaquettes. A zig-zag pattern is chosen to avoid X errors propagating vertically and Z errors propagating horizontally. The gate sequence in the Bell-measurement protocol is shown on rectangular plaquettes. Gates crossing the interface are colored red. The gate sequence must be alternated between rounds to avoid distance-reducing hook errors at the interface, as described in the main text.

graph produced by the circuit in Fig. 3 is provided in Appendix C. To aid understanding of the effect of Bell measurements on the syndrome graph, we will first assume that all local operations are perfect so that edges due to local errors in the syndrome graph can be removed. We are left with a syndrome graph where all edges represent Bell measurements at the interface. Since the interface has one spatial and one temporal dimension, the syndrome graph becomes two-dimensional.

To allow for efficient decoding of surface codes, one needs to represent Y errors as combinations of X and Z errors. In that case, under standard noise model assumptions, the syndrome graph splits into two disconnected graphs, also known as the primal and dual syndrome graphs. Typically, all edges in the primal (dual) syndrome graph represent bit-flip (phase-flip) errors. In the remaining discussion, we consider only the primal syndrome graph. All properties of the primal syndrome graph are shared by the dual. The primal syndrome graph at the interface between two modules for a distance-7 rotated surface code is shown in Fig. 4(a). The gate sequence in a round of syndrome measurements determines the orientation of the diagonal edges in the syndrome graph. Repeating the same gate sequence every round leads to the syndrome graph shown at the bottom of Fig. 4(a). Using alternating gate sequences between rounds produces the syndrome graph on the top. The syndrome graphs are overlaid on the surface-code detectors at the interface. Paths on the syndrome graph that connect two disjoint X -type boundaries lead to undetectable logical bit flips. If a repeated gate schedule is

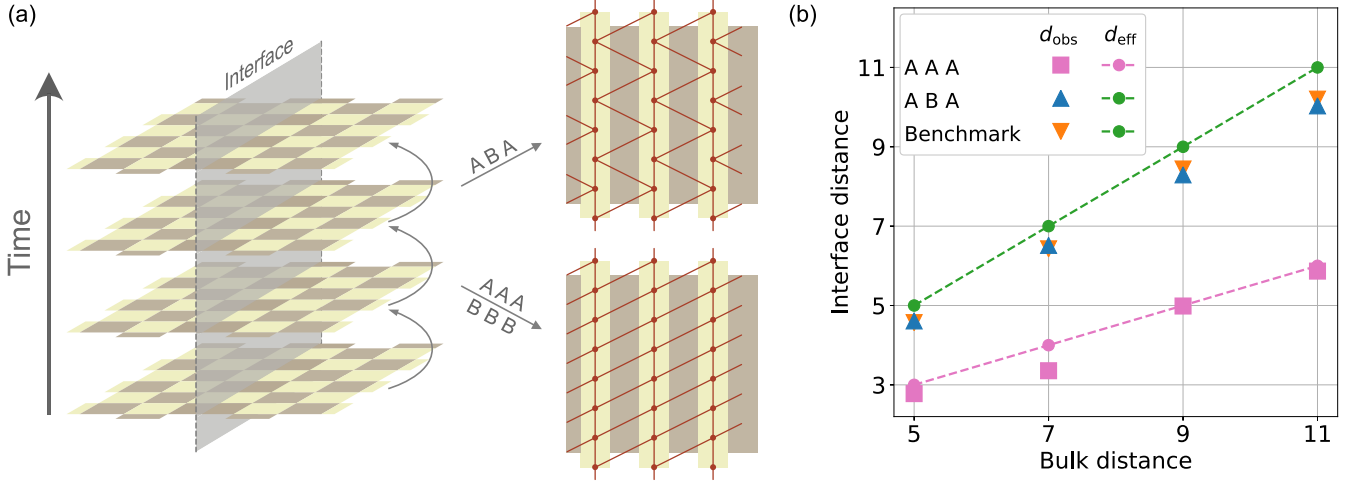


FIG. 4. Effect of alternating the gate sequence between rounds of syndrome measurement. Local gates and measurements are noiseless and gates crossing the interface are noisy. (a) There are two possible choices for the gate sequence that do not cause hook errors in the bulk, which we label A and B. Applying the same gate sequence for each round of syndrome measurement (AAA/BBB) produces a syndrome graph with parallel diagonal edges. The parallel orientation of the edges represents physical errors aligning with the logical operator of the code to reduce the effective distance of the code at the interface. Alternating the gate sequence between rounds of syndrome measurement (ABA) mitigates this effect by redirecting chains of physical errors. The resulting syndrome graph has a zig-zag pattern of diagonal edges. (b) Simulation of the effective code distance using repeating and alternating gate sequences for syndrome measurement. Repeating the gate sequence every round of syndrome measurement effectively halves the distance of the code. Alternating gate sequences between rounds increases the effective code distance at the interface to $0.9d$, which matches the benchmark protocol.

used, such a path consists of $(d+1)/2$ edges. Alternating the gate sequence changes the direction of half of the diagonal edges so that they form a zig-zag pattern. This alternation breaks the paths with $(d+1)/2$ edges. The result is that the shortest path connecting two boundaries contains d edges, which implies that the alternating gate sequence is distance-preserving. We stress that this is under the assumption that the bulk is noiseless.

Alternating the gate sequence between rounds of syndrome measurement will not increase the effective code distance if Bell pairs are used to measure stabilizers at the interface, as in Refs. [35, 36]. Noise originating from Bell-state preparation inevitably propagates to all data qubits near the interface as gates are applied, regardless of the chosen gate sequence.

To confirm the different effective distances of the repeating and alternating gate sequences in the Bell-measurement protocol, we simulate the syndrome-measurement circuit with interface noise only using Stim [50]. For code sizes up to $d = 11$, we use Stim to search for the shortest error chain that produces a logical error with trivial syndrome. The length of this error chain is the effective distance d_{eff} of the code. We find that $d_{\text{eff}} = d$ for the alternating gate sequence, and $d_{\text{eff}} = (d+1)/2$ for the repeating gate sequence.

In addition to searching for the shortest undetectable logical error, we run Monte Carlo simulations of our circuits to determine the observed effective distance of the

code, which we define by

$$p_L \sim \left(\frac{p}{p_{\text{th}}} \right)^{\frac{d_{\text{obs}}+1}{2}}, \quad (1)$$

where p_L is the logical error rate, p_{th} is the threshold error rate, and p is the physical error rate. The observed effective distance can be extracted from the simulated $p_L(p)$ for a fixed distance as

$$\frac{\partial \log p_L}{\partial \log p} = \frac{d_{\text{obs}}+1}{2}. \quad (2)$$

We approximate the left-hand side of Eq. (2) as the difference in logical error rate when $p = 0.1\%$ and $p = 1\%$. To determine the logical error rate, we use a minimum-weight perfect-matching decoder [51] to predict the logical state after d rounds of syndrome measurement. Simulation results are shown in Fig. 4(b). Repeating the gate sequence between rounds halves the effective distance at the interface compared to the bulk, as expected. For the simulated code distances, the observed effective distance at the interface when gate sequences are alternated is $d_{\text{obs}} = 0.9d$. We also run a simulation using the benchmark protocol that produces no in-plane diagonal edges at the interface. We find no difference in effective distance at the interface between the alternating gate sequence in the Bell-measurement protocol and the benchmark protocol. The deviation of the observed effective distance from the length of the shortest error chain is likely due to finite-size effects, as Eq. (1) is only accurate in the low (physical) error rate regime [33].

The effective distance at the interface due to errors in the Bell measurements alone will determine the logical error rate in the limit of vanishing local noise. However, significant local noise will be present in any implementation of surface codes on physical hardware. When a syndrome graph for the surface code is constructed with noise both in the bulk and at the interface, we find that the shortest error chain that causes an undetectable logical error is $3d/4$. This reduction is a result of errors from local two-qubit gates that affect both a data qubit and a qubit used for Bell measurement. To determine how strong this effect is, we resort to numerical studies in Section IV.

It is possible to restore the full distance at the interface by performing parity measurements on the ancilla qubits before the Bell measurements. The parity measurements flag errors that contribute to the reduction in distance. However, interface-crossing parity measurements also introduce additional noisy gates and interfere with the gate schedule in the bulk. If gates across the interface have to be teleported by consuming a Bell pair, adding parity measurements to the syndrome-measurement circuit will increase the entanglement-generation rate required to support the protocol.

IV. PERFORMANCE ANALYSIS

As an example application of the Bell-measurement protocol, we choose the task of establishing a logical Bell state shared by two separate modules. If the modules are too far apart to execute CX or CZ gates between modules directly, these gates are teleported by consuming one ebit [34] in the form of a pre-shared physical Bell pair. For a code distance d , this consumes d physical Bell pairs at the interface when the Bell-measurement protocol is used. The corresponding detector diagram in spacetime for code distance $d = 3$ is shown in Fig. 5. Two logical qubits are initially prepared in the state $|0_{L1}\rangle|0_{L2}\rangle$. A logical Bell state is obtained after measurement of the operator $X_{L1}X_{L2}$, which is performed by merging the two patches for d rounds of syndrome extraction and then splitting the merged patch. The logical measurement outcome is revealed by combining the physical measurement outcomes of the new stabilizers in the merged region of the instrument [30, 31]. When Bell measurements are used to merge the patches at the interface, the measurement outcomes also determine the value of the operator $Z_{L1}Z_{L2}$, even in the absence of noise, as the states are initially in $|0_{L1}\rangle|0_{L2}\rangle$. This is similar to how the outcome of measurements in a qubit teleportation determine the Pauli correction that must be applied to the teleported qubit to recover the state.

We simulate the full circuit needed to construct the instrument under depolarizing noise using Stim [50]. Details about the error model are provided in Appendix B. We then sample errors from the circuit that prepares logical Bell states as shown in Fig. 5 and decode the

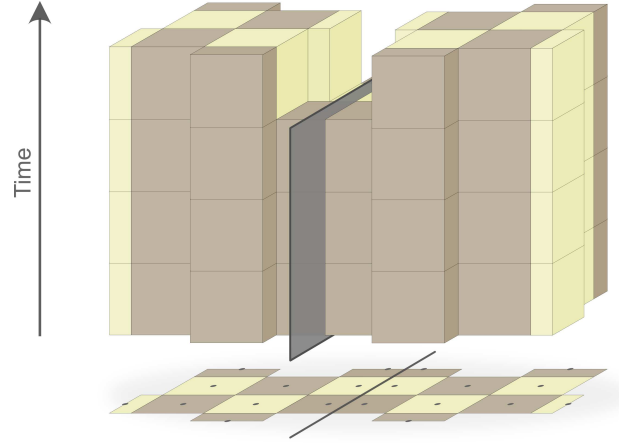


FIG. 5. Preparation of a Bell state across an interface using a rotated surface code and lattice surgery. Dark boxes represent X-detectors and light boxes represent Z-detectors. At the bottom is a projection of the fault-tolerant channel onto a surface, where the physical layout of ancilla qubits used to extract syndrome information is shown. The interface between the two modules is represented by a dark sheet which bisects the channel.

syndrome using a minimum-weight perfect-matching decoder [51]. For comparison, we also simulate the circuit with the benchmark protocol that applies the standard surface-code gate sequence across the interface. The Bell-measurement protocol uses this sequence to measure stabilizers in the bulk (Fig. 3). The local error rate p_{loc} is the error rate of operations that are executed within a module. These operations are physical qubit state preparation and measurement (SPAM), CX and CZ gates between nearest-neighbor physical qubits, and Hadamard gates. The link error rate p_{link} is the error rate of operations that are executed across the interface between modules. All such operations are either CX or CZ gates between two physical qubits in separate modules.

We find that under uniform depolarizing noise $p_{\text{loc}} = p_{\text{link}}$, the error threshold is $p_{\text{th}} = 0.53\%$ for the benchmark protocol, while the Bell-measurement protocol has a threshold at $p_{\text{th}} = 0.52\%$. The logical error rate at threshold is three percentage points higher for the Bell-measurement protocol. Next, we set the local physical error rate to $p_{\text{loc}} = 0.1\%$ as we sweep the link error rate p_{link} from 0.1% to 20% . We find an error threshold at $p_{\text{link}} = 13.5\%$ for the benchmark protocol vs. $p_{\text{link}} = 17.4\%$ for the Bell-measurement protocol. Details about the data from the threshold simulations are presented in Appendix D. Using a teleported Bell measurement instead of a direct Bell measurement lowers the threshold value to $p_{\text{link}} = 16.9\%$. Unlike the Bell-measurement protocol, the benchmark protocol benefits from teleportation of gates when the interface noise is high. Errors on Bell pairs used for teleporting gates in Z-stabilizer measurements cannot propagate to data qubits, which reduces the noise experienced by the data

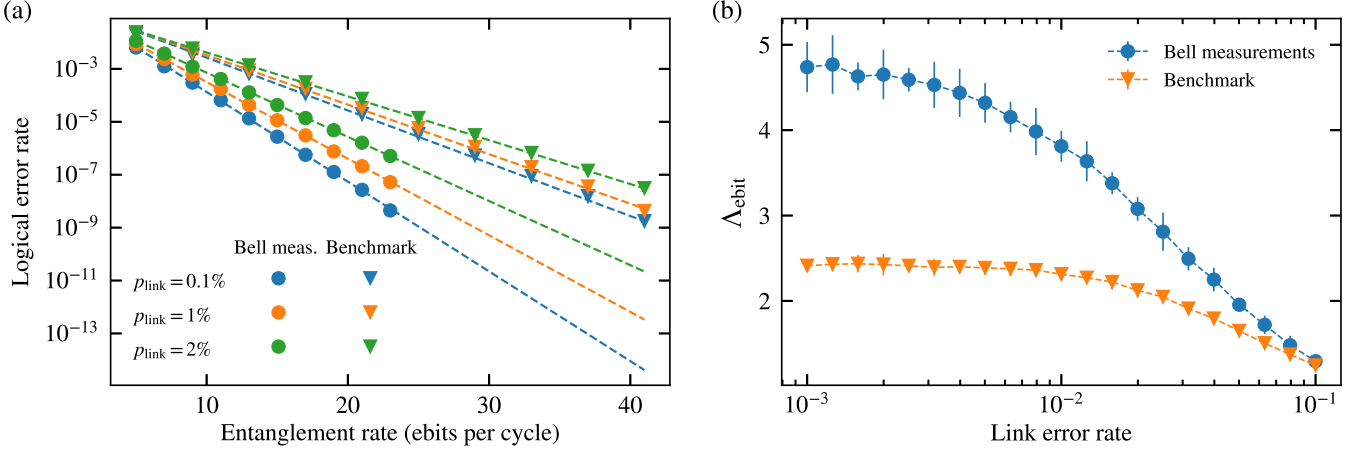


FIG. 6. Sub-threshold performance comparison between the benchmark protocol and the Bell-measurement protocol we develop in this work. (a) Logical error rate of the prepared Bell state as a function of the entanglement rate between modules. Dashed lines represent exponential fits to the simulated data. (b) Logical error suppression per ebit as a function of link error rate for the benchmark protocol and the Bell-measurement protocol. Markers represent the average over distances up to 19; error bars show the standard deviation.

qubits close to the interface. This is reflected in a link-error threshold for the teleported benchmark protocol at $p_{\text{link}} = 19.2\%$.

We now consider the case where the link noise is in the region $p_{\text{link}} \in [0.1\%, 10\%]$. This is the near-term feasible region where the link noise is below threshold for all protocols, yet still equal to or higher than the bulk noise. When high-fidelity inter-module entanglement is scarce, the Bell-measurement protocol is expected to achieve a stronger logical error suppression than the benchmark protocol because it requires only d ebits, or Bell pairs, per round of syndrome measurement. This efficient use of ebits allows the Bell-measurement protocol to effectively double the distance at the interface relative to the benchmark protocol. In Fig. 6(a), we plot the achievable error rate of a logical Bell state as a function of the available entanglement rate between modules with the benchmark protocol and the Bell-measurement protocol, respectively, for different levels of link noise. We find that the Bell-measurement protocol benefits from its low consumption of ebits, with a saving of $\sim 40\%$ relative to the benchmark protocol to achieve the same logical error rate. For example, by extrapolating our simulated error rates at $d \leq 23$ to higher distances, we find that to reach a logical error rate of 10^{-12} with a link error rate of 0.1% (1%) would require 35 (39) Bell pairs per round of syndrome measurement with the Bell-measurement protocol using a distance-35(39) surface code, compared to 57 (61) Bell pairs using a distance-29(31) surface code for the benchmark protocol.

The suppression of the logical error rate with code distance is exponential. It is captured by the suppression factor $\Lambda_d = p_L(d)/p_L(d+2)$, which represents the drop in logical error rate when the code distance is increased by two. Similarly, we define the suppression factor Λ_{ebit} as the reduction in logical error rate when the number

of ebits available per syndrome-measurement cycle increases by two. We plot the simulated value of Λ_{ebit} as a function of the link error rate p_{link} in Fig. 6(b). The Bell-measurement protocol offers stronger logical error suppression than the benchmark protocol for any level of interface noise in our simulations when access to ebits is limited. The suppression relative to the benchmark protocol is stronger for lower levels of interface noise and increases with the size of the code.

We also consider the case where gates between modules are applied directly without teleportation. The simulated logical error rates as a function of code distance are shown in Fig. 7(a). When $p_{\text{link}} > 3.4\%$, the Bell-measurement protocol outperforms the benchmark protocol despite the reduced distance at the interface. We believe this is due to the fact that the Bell-measurement protocol eliminates error propagation to qubits away from the interface, whereas the benchmark protocol applies noisy gates directly to data qubits close to the interface that take part in subsequent noise-propagating two-qubit gates. By decreasing the link noise below 1% , the logical error rate approaches a floor set by the error rate in the bulk for both protocols. This logical error floor is approximately a factor two higher for the Bell-measurement protocol. Adding parity checks to recover the full distance at the interface does not lower this floor, which suggests that the higher error floor is a consequence of the extra gates and measurements needed to perform Bell measurements, or the added edges in the syndrome graph at the interface that result from using Bell measurements, or a combination of these two effects.

We plot the simulated values of Λ_d for the benchmark and Bell-measurement protocols in Fig. 7(b). The benchmark protocol benefits significantly from teleportation of gates when $p_{\text{link}} > 1\%$. This agrees with the elevated threshold we found for the benchmark protocol and is due

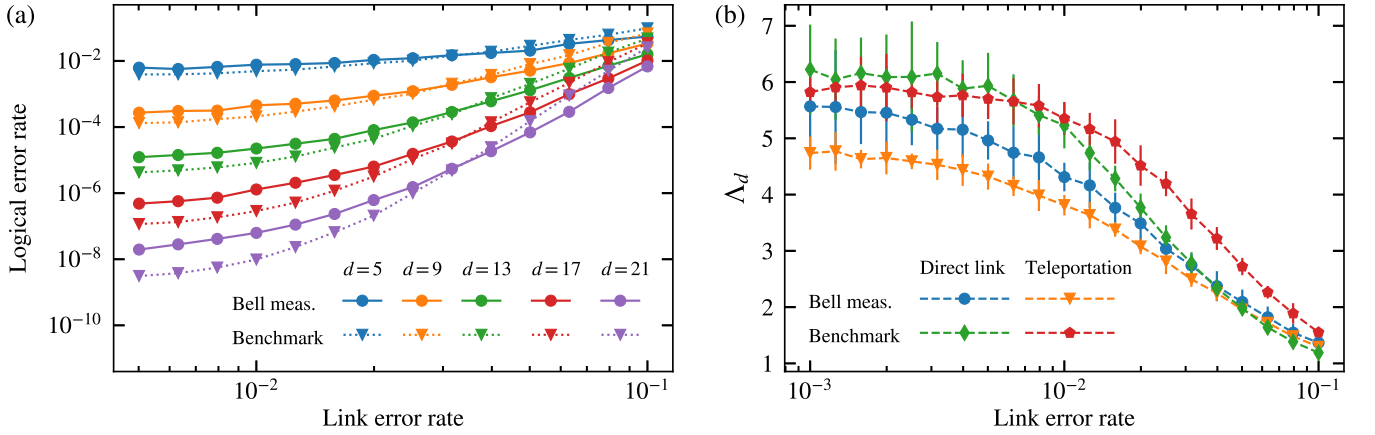


FIG. 7. Sub-threshold performance when entanglement between modules is not a scarce resource. (a) Logical error rate of the prepared Bell state as a function of link noise when gates are applied directly across the interface. (b) Logical error suppression per added distance as a function of link error rate for the syndrome-measurement circuits in this work. Markers represent the average over distances up to 19; error bars show the standard deviation.

to noise confinement. The Bell-measurement protocol is maximally noise-confining by design. The difference between direct and teleported gates is therefore negligible for this protocol at high levels of interface noise. As the link error rate decreases towards the bulk error rate, the relative importance of noise confinement decreases. The overhead associated with teleportation means noise suppression for the teleportation protocols is not as strong as the direct link protocols for link noise levels close to that of the bulk. Suppressing logical errors by increasing the distance at low levels of interface noise is not as effective with the Bell-measurement protocol compared to the benchmark protocol. As a result, the Bell-measurement protocol typically requires an increase in distance at the interface by $\sim 20\%$ relative to the benchmark protocol to reach the same logical error rate. This increased distance is the reason the Bell-measurement protocol saves only $\sim 40\%$ on ebits compared to the benchmark protocol.

V. CONCLUSION AND OUTLOOK

We have introduced a protocol for performing lattice surgery with Bell measurements at the interface between processor modules in a distributed quantum computer using the rotated surface code. The protocol is based on the insight that Bell measurements and Bell pairs are interchangeable in measurement-based quantum computing. The Bell measurements provide two main advantages compared to Bell pairs used in other protocols. First, they strongly confine the elevated noise at the interface. Second, a protocol based on Bell measurements requires only d ebits for gate teleportation for each round of syndrome measurement in a rotated surface code, compared to $2d-1$ for established protocols. This makes it particularly attractive for architectures that use gate teleportation between modules. The performance of the protocol

shows that it is competitive over a wide range of noise levels. Due to its effective use of ebits, the Bell-measurement protocol offers a stronger logical error suppression per consumed ebit than alternative protocols, albeit at the cost of an increased number of physical qubits to achieve the same distance. As an example, we find a 40% reduction in entanglement rate needed to prepare a logical Bell state with an error rate of 10^{-12} between two modules with local physical error rate at 0.1% and link error rates up to 1%. However, there is still room for improvement because interface errors can combine with local errors to form undetectable logical errors of length $3d/4$ such that the physical qubit overhead to reach the same effective distance is increased in comparison to alternative schemes that require more shared entanglement, such as the protocols in Refs. [35, 36].

We expect that the availability of high-fidelity Bell pairs will be a limiting factor for modular architectures for the foreseeable future, particularly for modules in separate cryostats or vacuum chambers. Optically distributed entanglement of superconducting circuits in separate dilution refrigerators is yet to be demonstrated experimentally. Platforms such as trapped ions and neutral atoms have optical transitions that can be used to generate photon-mediated entanglement, but even here the rate of ebit generation has not exceeded 200 Hz [38]. We expect the Bell-measurement protocol to accelerate the development of modular, fault-tolerant quantum computers by effectively halving the entanglement-generation rate required to support inter-module lattice surgery. Our design principles can be applied to any quantum circuit that is distributed across processor modules, which can share only a limited amount of entanglement, thus cutting the cost of implementing such circuits in modular architectures.

The effective distance of the rotated surface code we study in this paper depends on the gate sequence over

two rounds of syndrome measurements. We have introduced a technique that alternates gate sequences between subsequent rounds to avoid distance-reducing errors at the interface between modules. This technique could have applications in the implementation of other quantum low-density parity-check codes that suffer from distance-reducing hook errors.

Future work could look to break the trade-off between low ebit consumption and full effective code distance that still exists in the protocol presented here. Another natural extension of this work is to study the impact of missing Bell pairs on code performance, as it is likely that hardware responsible for heralding Bell pairs will occasionally fail due to the probabilistic nature of heralding. Varying the noise model to identify regimes where different protocols provide a particular advantage over others could provide further insight.

ACKNOWLEDGMENTS

We thank Moritz Lange and Mats Granath for helpful discussions. T.H. acknowledges the financial support from the Chalmers Excellence Initiative Nano and the Knut and Alice Wallenberg Foundation through the Wallenberg Centre for Quantum Technology (WACQT). A.F.K. acknowledges support from the Swedish Foundation for Strategic Research (grant numbers FFL21-0279 and FUS21-0063), the Horizon Europe programme HORIZON-CL4-2022-QUANTUM-01-SGA via the project 101113946 OpenSuperQPlus100, and from the Knut and Alice Wallenberg Foundation through the Wallenberg Centre for Quantum Technology (WACQT). R.V.L. acknowledges support from the Knut and Alice Wallenberg foundation through the Wallenberg Centre for Quantum Technology (WACQT), from the European Research Council via Starting Grant 948265, and from the Swedish Foundation for Strategic Research via grant FFL21-0039. Numerical simulations were performed using resources at the Chalmers Centre for Computational Science and Engineering (C3SE) under project 2025/1-6.

Appendix A: Bell measurements

Bell measurements are joint measurements on two qubits in the Bell basis, which are the four Bell states that are simultaneous eigenstates of the Pauli operators X_1X_2 and Z_1Z_2 . The Bell basis is a complete basis for the Hilbert space of two qubits. The quantum circuit and corresponding ZX diagram for a Bell measurement are shown in Fig. 8. The circuit for a Bell measurement is simply the inverse of the circuit that prepares a Bell state between two qubits. Bell measurements can be implemented deterministically in platforms with access to entangling gates, e.g. superconducting qubits, atoms and

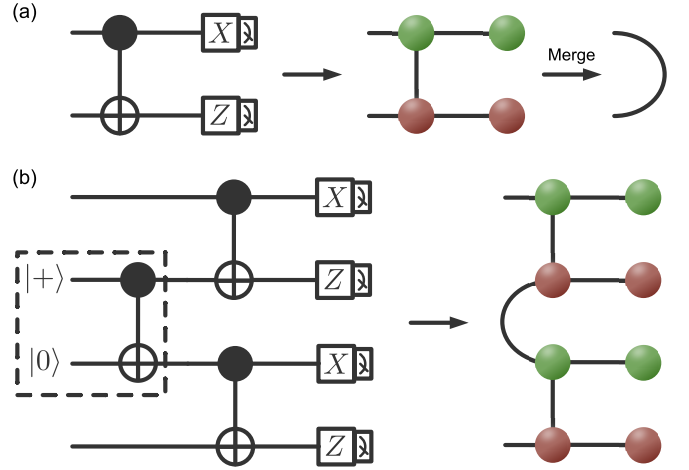


FIG. 8. Bell-measurement circuits and their representation as ZX diagrams. (a) A standard Bell measurement on two qubits can be implemented with one CX gate followed by single-qubit measurements. The control qubit is measured in the X basis to determine the observable X_1X_2 , and the target qubit is measured in the Z basis to determine the observable Z_1Z_2 . (b) A teleported Bell measurement requires an initial preparation of a Bell pair (dashed box).

ions, because the circuit in Fig. 8 distinguishes all four Bell states.

In the ZX diagram language, Bell measurements and Bell states can be represented by a single piece of wire, as illustrated in Fig. 8(a) by converting the circuit for a Bell measurement into the corresponding ZX diagram using the building blocks in Fig. 2(d). This representation is particularly helpful when we are looking for possible implementations of fault-tolerant protocols described by ZX diagrams. Interfaces between modules can be modeled as planes cutting through the diagrams, and wires that cross planes can be interpreted as Bell measurements on qubits between modules. The ZX calculus allows spiders to be split and merged to find planes that intersect as few wires as possible. These planes represent optimal divisions of an operation described by the ZX diagram across modules with the respect to the amount of entanglement that is needed to implement it. We implement this strategy on the ZX diagram that describes the surface code in this work, but the approach is applicable to any quantum circuit that must be divided among processor modules that can share a limited amount of entanglement.

Appendix B: Error model

We use a primarily depolarizing error model for the circuit-level simulations. Each single-qubit gate (including the identity gate) is followed by a depolarizing channel which, with probability p , applies one Pauli operator $P \in \{X, Y, Z\}$ chosen at random. That is, the operator X is applied with probability $p/3$, the operator Z is ap-

plied with probability $p/3$, or the operator Y is applied with probability $p/3$. The probability of no error is then $1 - p$. Each two-qubit gate is followed by a two-qubit depolarizing channel which, with probability p , applies $P_A \otimes P_B \neq I \otimes I$ where $P_{A,B} \in \{X, Y, Z, I\}$. That is, each of the Pauli product operators have a probability $p/15$ of being applied, and the probability of no error is $1 - p$. All measurements are done in the Z basis; measurements in the X basis are implemented with a Hadamard followed by a Z-basis measurement. Each measurement outcome can return the wrong result with probability set by the state-preparation and measurement (SPAM) error rate. All qubits are reset to $|0\rangle$ after measurement and a bit-flip error may occur after the reset. The error model and noise levels are summarized in Table I.

When a gate needs to be teleported, we first prepare a Bell pair between the two modules using perfect operations. The resulting state is

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle). \quad (\text{B1})$$

Ideal preparation is followed by a two-qubit depolarizing channel with noise strength p_{link} . After the channel, the state shared between the modules is

$$\rho = \sum_{i=0}^{15} K_i |\Phi^+\rangle \langle \Phi^+| K_i^\dagger, \quad (\text{B2})$$

where $\{K_i\} = \{\sqrt{1 - p_{\text{link}}} I_A \otimes I_B, \sqrt{p_{\text{link}}/15} P_A \otimes P_B\}$ are the Kraus operators for the depolarizing channel. The fidelity of the resulting Bell pair is

$$F = \langle \Phi^+ | \rho | \Phi^+ \rangle = 1 - p_{\text{link}} + \frac{3p_{\text{link}}}{15} = 1 - \frac{4p_{\text{link}}}{5}. \quad (\text{B3})$$

Appendix C: Syndrome graph

A syndrome graph is a graphical representation of the error syndromes obtained from the detectors of an error-correcting code. Detectors are represented by vertices in the graph, and faults that can cause a detector to flip correspond to edges incident on a vertex. Under circuit-level noise, the syndrome graph for the surface code is a (2+1)-dimensional spacetime graph where edges represent gate errors, idling errors, or errors on qubits during state preparation or measurement.

The structure of the syndrome graph depends on the gate sequence used to measure detectors. The circuit that we use to extract error syndromes in this work produces a syndrome graph of the form shown in Fig. 9. The syndrome graph is generally an undirected simple graph where each vertex has degree twelve. However, the introduction of an alternative gate sequence at the interface produces vertices close to the interface that have degree eleven. The gate sequence is symmetric about the interface, and hence the syndrome graph becomes symmetric about the interface as well.

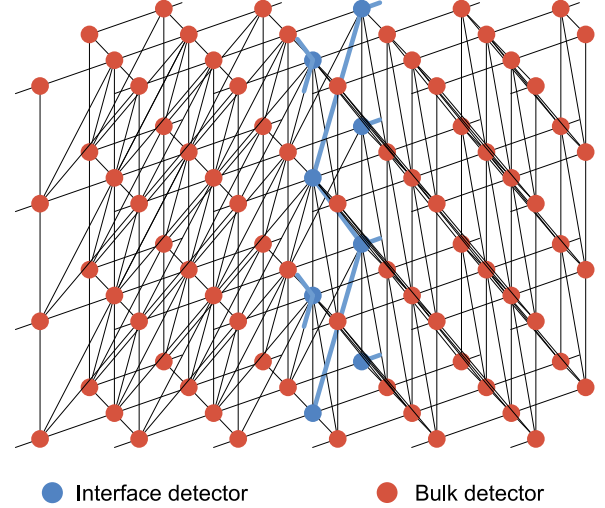


FIG. 9. Full syndrome graph of two merged distance-5 rotated surface codes under circuit-level noise with an interface between modules in the center. Note that time-like boundaries are not included. Edges that correspond to faults at the interface are highlighted.

Appendix D: Threshold simulations

The results from simulations identifying the thresholds for the benchmark and Bell-measurements protocols under uniform noise are plotted in Fig. 10(a)-(c). We find that the logical error rate close to the threshold is slightly higher for the Bell-measurement protocol than the benchmark protocol, and the threshold value is slightly lower. This is expected due to the added complexity and local reduction in effective distance due to the Bell measurements, which are advantageous only when there is a noisy interface between modules.

We also consider a situation where the local error rate in both modules is set to $p_{\text{loc}} = 0.1\%$ but the link noise is elevated. This is a much more realistic scenario with near-term hardware constraints. The simulation results are shown in Fig. 10(d)-(f). As explained in the main text, the Bell-measurement protocol is more resilient to high link noise than the benchmark protocol when gates are applied across the interface without teleportation. The fact that the Bell measurements fully confine the link noise to the ancilla qubits at the interface outweighs the reduction in effective distance. This is no longer the case when gates are teleported. Gate teleportation protects the data qubits from some of the link noise. Because the Bell-measurement protocol is intrinsically fully noise-confining, performance is slightly compromised by the overhead required for teleportation.

Operation	Idling	Hadamard	CX/CZ	SPAM	Interface CX/CZ
Stim instruction	DEPOLARIZE1	DEPOLARIZE1	DEPOLARIZE2	X_ERROR	DEPOLARIZE2
Uniform noise	p_{loc}	p_{loc}	p_{loc}	p_{loc}	$p_{\text{link}} = p_{\text{loc}}$
Elevated link noise	0.1 %	0.1 %	0.1 %	0.1 %	p_{link}

TABLE I. Summary of the error model used in the simulations.

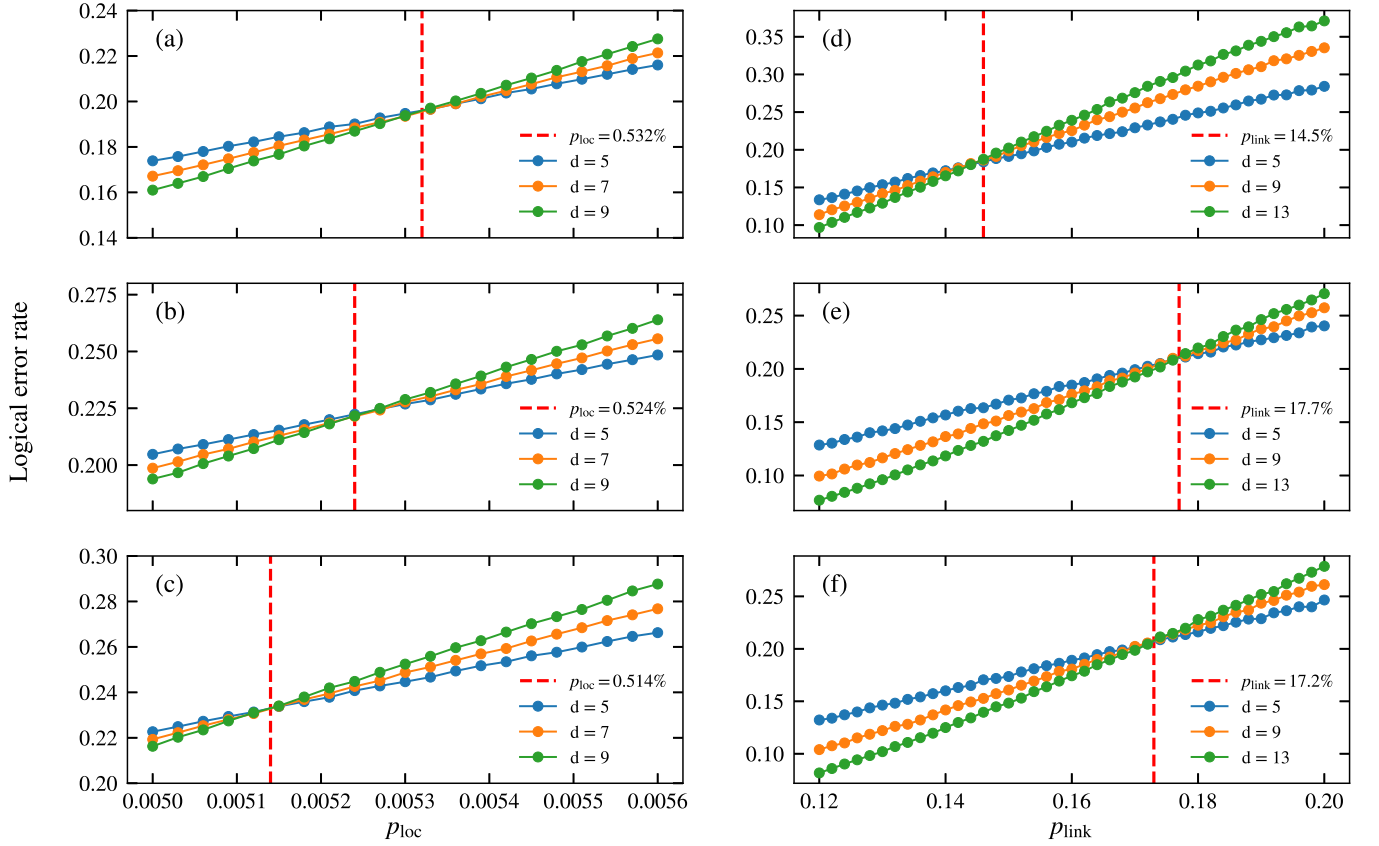


FIG. 10. Data from threshold simulations. (a) The benchmark protocol with direct link under uniform noise. (b) The Bell-measurement protocol with direct link under uniform noise. (c) The Bell-measurement protocol with teleported gates under uniform noise. (d) The benchmark protocol with direct link under elevated link noise. (e) The Bell-measurement protocol with direct link under elevated link noise. (f) The Bell-measurement protocol with teleported gates under elevated link noise.

-
- [1] P. W. Shor, *SIAM Journal on Computing* **26**, 1484 (1997).
- [2] N. Gisin and R. Thew, *Nature Photonics* **1**, 165 (2007).
- [3] A. M. Dalzell, S. McArdle, M. Berta, P. Bienias, C.-F. Chen, A. Gilyén, C. T. Hann, M. J. Kastoryano, E. T. Khabiboulline, A. Kubica, G. Salton, S. Wang, and F. G. S. L. Brandão, Quantum algorithms: A survey of applications and end-to-end complexities (2023), [arXiv:2310.03011](https://arxiv.org/abs/2310.03011).
- [4] H.-Y. Huang, S. Choi, J. R. McClean, and J. Preskill, The vast world of quantum advantage (2025), [arXiv:2508.05720](https://arxiv.org/abs/2508.05720).
- [5] X. Gu, A. F. Kockum, A. Miranowicz, Y.-X. Liu, and F. Nori, *Physics Reports* **718–719**, 1 (2017).
- [6] A. Blais, A. L. Grimsmo, S. M. Girvin, and A. Wallraff, *Reviews of Modern Physics* **93**, 025005 (2021).
- [7] S. Krinner, S. Storz, P. Kurpiers, P. Magnard, J. Heinsoo, R. Keller, J. Lütolf, C. Eichler, and A. Wallraff, *EPJ Quantum Technology* **6**, 1 (2019).
- [8] M. Mohseni, A. Scherer, K. G. Johnson, O. Wertheim, M. Otten, N. A. Aadit, Y. Alexeev, K. M. Bresniker, K. Y. Camsari, B. Chapman, S. Chatterjee, G. A. Dagnew, A. Esposito, F. Fahim, M. Fiorentino, A. Gajjar, A. Khalid, X. Kong, B. Kulchytsky, E. Kyoseva, R. Li, P. A. Lott, I. L. Markov, R. F. McDermott, G. Pedretti, P. Rao, E. Rieffel, A. Silva, J. Sorebo, P. Spentzouris, Z. Steiner, B. Torosov, D. Venturelli, R. J. Visser, Z. Webb, X. Zhan, Y. Cohen, P. Ronagh, A. Ho, R. G. Beausoleil, and J. M. Martinis, How to Build a Quantum Supercomputer: Scaling from Hundreds to Millions of Qubits (2025), [arXiv:2411.10406](https://arxiv.org/abs/2411.10406).
- [9] S. Kosen, H.-X. Li, M. Rommel, R. Rehammar, M. Caputo, L. Grönberg, J. Fernández-Pendás, A. F. Kockum, J. Biznárová, L. Chen, C. Križan, A. Nylander, A. Osman, A. F. Roudsari, D. Shiri, G. Tancredi, J. Govenius, and J. Bylander, *PRX Quantum* **5**, 030350 (2024).
- [10] T. Strohm, K. Wintersperger, F. Dommert, D. Basilewitsch, G. Reuber, A. Hoursanov, T. Ehmer, D. Vodola, and S. Luber, Ion-Based Quantum Computing Hardware: Performance and End-User Perspective (2024), [arXiv:2405.11450](https://arxiv.org/abs/2405.11450).
- [11] K. Wintersperger, F. Dommert, T. Ehmer, A. Hursanov, J. Klepsch, W. Maurer, G. Reuber, T. Strohm, M. Yin, and S. Luber, *EPJ Quantum Technology* **10**, 32 (2023).
- [12] C. Monroe and J. Kim, *Science* **339**, 1164 (2013).
- [13] L. Jiang, J. M. Taylor, A. S. Sørensen, and M. D. Lukin, *Physical Review A* **76**, 062323 (2007).
- [14] C. Monroe, R. Raussendorf, A. Ruthven, K. R. Brown, P. Maunz, L.-M. Duan, and J. Kim, *Physical Review A* **89**, 022317 (2014).
- [15] N. H. Nickerson, Y. Li, and S. C. Benjamin, *Nature Communications* **4**, 1756 (2013).
- [16] N. H. Nickerson, J. F. Fitzsimons, and S. C. Benjamin, *Physical Review X* **4**, 041041 (2014).
- [17] Y. Li and S. C. Benjamin, *Physical Review A* **94**, 042303 (2016).
- [18] H. Aghaee Rad, T. Ainsworth, R. N. Alexander, B. Altieri, M. F. Askarani, R. Baby, L. Banchi, B. Q. Baragiola, J. E. Bourassa, R. S. Chadwick, I. Charania, H. Chen, M. J. Collins, P. Contu, N. D’Arcy, G. Dauphinais, R. De Prins, D. Deschenes, I. Di Luch, S. Duque, P. Edke, S. E. Fayer, S. Ferracin, H. Ferretti, J. Gefaell, S. Glancy, C. González-Arciniegas, T. Grainger, Z. Han, J. Hastrup, L. G. Helt, T. Hillmann, J. Hundal, S. Izumi, T. Jaeken, M. Jonas, S. Kocsis, I. Krasnokutska, M. V. Larsen, P. Laskowski, F. Laudenbach, J. Lavoie, M. Li, E. Lomonte, C. E. Lopetegui, B. Luey, A. P. Lund, C. Ma, L. S. Madsen, D. H. Mahler, L. Mantilla Calderón, M. Menotti, F. M. Miatto, B. Morrison, P. J. Nadkarni, T. Nakamura, L. Neuhaus, Z. Niu, R. Noro, K. Papirov, A. Pesah, D. S. Phillips, W. N. Plick, T. Rogalsky, F. Rortais, J. Sabines-Chesterking, S. Safavi-Bayat, E. Sazhaev, M. Seymour, K. Rezaei Shad, M. Silverman, S. A. Srinivasan, M. Stephan, Q. Y. Tang, J. F. Tasker, Y. S. Teo, R. B. Then, J. E. Tremblay, I. Tzitrin, V. D. Vaidya, M. Vasmer, Z. Vernon, L. F. S. S. M. Villalobos, B. W. Walshe, R. Weil, X. Xin, X. Yan, Y. Yao, M. Zamani Abnili, and Y. Zhang, *Nature* **638**, 912 (2025).
- [19] J. Ramette, J. Sinclair, N. P. Breuckmann, and V. Vuletić, *npj Quantum Information* **10**, 58 (2024).
- [20] A. Gold, J. P. Paquette, A. Stockklauser, M. J. Reagor, M. S. Alam, A. Bestwick, N. Didier, A. Nersisyan, F. Oruc, A. Razavi, B. Scharmann, E. A. Sete, B. Sur, D. Venturelli, C. J. Winkleblack, F. Wudarski, M. Harburn, and C. Rigetti, *npj Quantum Information* **7**, 142 (2021).
- [21] X. Wu, H. Yan, G. Andersson, A. Anferov, M.-H. Chou, C. R. Conner, J. Grebel, Y. J. Joshi, S. Li, J. M. Miller, R. G. Povey, H. Qiao, and A. N. Cleland, *Physical Review X* **14**, 041030 (2024).
- [22] M. Mollenhauer, A. Irfan, X. Cao, S. Mandal, and W. Pfaff, A high-efficiency plug-and-play superconducting qubit network (2024), [arXiv:2407.16743](https://arxiv.org/abs/2407.16743).
- [23] N. Roch, M. E. Schwartz, F. Motzoi, C. Macklin, R. Vijay, A. W. Eddins, A. N. Korotkov, K. B. Whaley, M. Sarovar, and I. Siddiqi, *Physical Review Letters* **112**, 170501 (2014).
- [24] B. Hensen, H. Bernien, A. E. Dréau, A. Reiserer, N. Kalb, M. S. Blok, J. Ruitenbergh, R. F. L. Vermeulen, R. N. Schouten, C. Abellán, W. Amaya, V. Pruneri, M. W. Mitchell, M. Markham, D. J. Twitchen, D. Elkouss, S. Wehner, T. H. Taminiau, and R. Hanson, *Nature* **526**, 682 (2015).
- [25] C. Dickel, J. J. Wesdorp, N. K. Langford, S. Peiter, R. Sagastizabal, A. Bruno, B. Criger, F. Motzoi, and L. DiCarlo, *Physical Review B* **97**, 064508 (2018).
- [26] Y. Zhong, H.-S. Chang, A. Bienfait, E. Dumur, M.-H. Chou, C. R. Conner, J. Grebel, R. G. Povey, H. Yan, D. I. Schuster, and A. N. Cleland, *Nature* **590**, 571 (2021).
- [27] L. D. Burkhardt, J. D. Teoh, Y. Zhang, C. J. Axline, L. Frunzio, M. Devoret, L. Jiang, S. Girvin, and R. Schoelkopf, *PRX Quantum* **2**, 030321 (2021).
- [28] D. Main, P. Drmota, D. P. Nadlinger, E. M. Ainley, A. Agrawal, B. C. Nichol, R. Srinivas, G. Araneda, and D. M. Lucas, *Nature* **638**, 383 (2025).
- [29] A. Almanakly, B. Yankelevich, M. Hays, B. Kannan, R. Assouly, A. Greene, M. Gingras, B. M. Niedzielski, H. Stickler, M. E. Schwartz, K. Serniak, J. I.-j. Wang, T. P. Orlando, S. Gustavsson, J. A. Grover, and W. D. Oliver, *Nature Physics* **21**, 825 (2025).
- [30] D. Horsman, A. G. Fowler, S. Devitt, and R. V. Meter,

- New Journal of Physics **14**, 123011 (2012).
- [31] D. Litinski, *Quantum* **3**, 128 (2019).
 - [32] A. Yu. Kitaev, *Annals of Physics* **303**, 2 (2003).
 - [33] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, *Journal of Mathematical Physics* **43**, 4452 (2002).
 - [34] C. H. Bennett, H. J. Bernstein, S. Popescu, and B. Schumacher, *Physical Review A* **53**, 2046 (1996).
 - [35] M. A. Shalby, R. Wang, D. Sedov, and L. P. Pryadko, Optimized noise-resilient surface code teleportation interfaces (2025), [arXiv:2503.04968](#).
 - [36] H. Jacinto, E. Gouzien, and N. Sangouard, Network Requirements for Distributed Quantum Computation (2025), [arXiv:2504.08891](#).
 - [37] P. Magnard, S. Storz, P. Kurpiers, J. Schär, F. Marxer, J. Lütolf, T. Walter, J.-C. Besse, M. Gabureac, K. Reuer, A. Akin, B. Royer, A. Blais, and A. Wallraff, *Physical Review Letters* **125**, 260502 (2020).
 - [38] L. J. Stephenson, D. P. Nadlinger, B. C. Nichol, S. An, P. Drmota, T. G. Ballance, K. Thirumalai, J. F. Goodwin, D. M. Lucas, and C. J. Ballance, *Physical Review Letters* **124**, 110501 (2020).
 - [39] K. Heya, T. Phung, M. Malekakhlagh, R. Steiner, M. Turchetti, W. Shanks, J. Mamin, W.-S. Lu, Y. P. Kandel, N. Sundaresan, and J. Orcutt, Randomized benchmarking of a high-fidelity remote CNOT gate over a meter-scale microwave interconnect (2025), [arXiv:2502.15034](#).
 - [40] D. E. Browne and T. Rudolph, *Physical Review Letters* **95**, 010501 (2005).
 - [41] S. Bartolucci, P. Birchall, H. Bombín, H. Cable, C. Dawson, M. Gimeno-Segovia, E. Johnston, K. Kieling, N. Nickerson, M. Pant, F. Pastawski, T. Rudolph, and C. Sparrow, *Nature Communications* **14**, 912 (2023).
 - [42] F. Verstraete and J. I. Cirac, *Physical Review A* **70**, 060302 (2004).
 - [43] B. W. Walshe, B. Q. Baragiola, H. Ferretti, J. Gefaell, M. Vasmer, R. Weil, T. Matsuura, T. Jaeken, G. Pantaleoni, Z. Han, T. Hillmann, N. C. Menicucci, I. Tzitrin, and R. N. Alexander, *Physical Review Letters* **134**, 100602 (2025).
 - [44] T. Hillmann, G. Dauphinais, I. Tzitrin, and M. Vasmer, Single-shot and measurement-based quantum error correction via fault complexes (2024), [arXiv:2410.12963 \[quant-ph\]](#).
 - [45] M. McEwen, D. Bacon, and C. Gidney, *Quantum* **7**, 1172 (2023).
 - [46] J. van de Wetering, ZX-calculus for the working quantum computer scientist (2020), [arXiv:2012.13966](#).
 - [47] C. Gidney, A Pair Measurement Surface Code on Pentagons (2023), [arXiv:2206.12780](#).
 - [48] H. Bombin, D. Litinski, N. Nickerson, F. Pastawski, and S. Roberts, *Quantum* **8**, 1379 (2024).
 - [49] Y. Tomita and K. M. Svore, *Physical Review A* **90**, 062320 (2014).
 - [50] C. Gidney, *Quantum* **5**, 497 (2021).
 - [51] O. Higgott and C. Gidney, *Quantum* **9**, 1600 (2025).