Addressable fault-tolerant universal quantum gate operations for high-rate lift-connected surface codes

Josias Old,[1,2,3,*] Juval Bechar,[1,2] Markus Müller,[1,2] and Sascha Heußen[3,†]

[1] *Institute for Quantum Information, RWTH Aachen University, Aachen, Germany*
[2] *Institute for Theoretical Nanoelectronics (PGI-2), Forschungszentrum Jülich, Jülich, Germany*
[3] *neQxt GmbH, Cologne, Germany*

Quantum low-density parity check (qLDPC) codes are among the leading candidates to realize error-corrected quantum memories with low qubit overhead. Potentially high encoding rates and large distance relative to their block size make them appealing for practical suppression of noise in near-term quantum computers. In addition to increased qubit-connectivity requirements compared to more conventional topological quantum error correcting codes, qLDPC codes remain notoriously hard to compute with. In this work, we introduce a construction to implement all Clifford quantum gate operations on the recently introduced lift-connected surface (LCS) codes [1]. These codes can be implemented in a 3D-local architecture and achieve asymptotic scaling $[[n, \mathcal{O}(n^{1/3}), \mathcal{O}(n^{1/3})]]$. In particular, LCS codes realize favorable instances with small numbers of qubits: For the $[[15, 3, 3]]$ LCS code, we provide deterministic fault-tolerant (FT) circuits of the logical gate set $\{\overline{H}_i, \overline{S}_i, \overline{C_i X_j}\}_{i,j\in(0,1,2)}$ based on flag qubits. By adding a procedure for FT magic state preparation, we show quantitatively how to realize an FT universal gate set in $d = 3$ LCS codes. Numerical simulations indicate that our gate constructions can attain pseudothresholds in the range $p_{\text{th}} \approx 4.8 \cdot 10^{-3} - 1.2 \cdot 10^{-2}$ for circuit-level noise. The schemes use a moderate number of qubits and are therefore feasible for near-term experiments, facilitating progress for fault-tolerant error corrected logic in high-rate qLPDC codes.

I. INTRODUCTION

Quantum error correcting (QEC) codes provide a means to practically realize fault-tolerant (FT) quantum computation (QC) by overcoming the limitations that are placed by noisy qubits and imperfect physical quantum gate operations performed on them. The most prominent QEC codes, like topological surface or color codes, typically have vanishing *rate*, meaning that the relative amount of logical information they encode approaches zero upon increased protection against noise [2, 3]. High-rate quantum low-density parity check (qLDPC) codes emerged as promising candidates for improved memory capabilities while also reducing the physical qubit overhead compared to, for instance, the paradigmatic surface code [4–9]. Since the actual goal is FTQC, there is a need for QEC codes that, while achieving high rate and small memory overhead, also allow one to perform FT logical quantum gate operations.

The simplest way to construct a FT logical gate is a *transversal* implementation, where gates only act on single (or bounded number of) disjoint data qubits within a block [10]. This ensures that not too many faults occur on the data qubits and prevents faults from spreading uncontrollably. There are, however, only very few error-correcting codes that support transversal Clifford gates. Most notably, the full Clifford group is transversal on 2D topological color codes [3]. If an error correcting code has a certain structure or symmetries, unitary implementations of gates that are not strictly transversal can still be fault-tolerant. For instance, the logical Hadamard gate can be realized transversally up to qubit permutation in the presence of a so-called ZX-duality [11–14]. In high-rate codes such gates are often global; they act on all logical qubits of a block and are not able to address individual logical qubits [15].

A well-established method of achieving addressability of logical qubits is through teleportation-based gates using joint logical measurements [16]. The main challenge of this approach is to realize FT logical measurements and has been tackled using lattice-surgery approaches [17]. Generalizations of the original surface code lattice-surgery employ special but rather involved LDPC ancillary states that enable a decomposition of logical operators into low-weight measurements [18–26].

Between the previous two approaches lies a third strategy to realize FT logical gates. It is based on unitary logical gates implemented with non-FT circuits. Through additional means, the circuits are then rendered FT subsequently. Examples include flag measurements [27], concatenation or intermediate stabilizer measurements in the pieceable fault-tolerance framework [28]. This is the approach we follow in our work. In order to achieve universality, it is important to note that no error-correcting code can implement a *universal* gate set using only transversal operations [29]. Implementing a FT universal gate can be challenging. A widely used technique relies on magic state injection. There, good quality magic states are used as a non-stabilizer resource to feed into a gate teleportation circuit that itself again only contains Clifford gates but facilitates the logical non-Clifford gate [30, 31].

Recent experimental implementations of QEC have been focusing mostly on demonstrating increased qubit

* j.old@fz-juelich.de
† s.heussen@neqxt.org

lifetimes via quantum memory experiments in small code instances [32–38]. Primitives for a universal set of quantum gate operations have been demonstrated with topological color codes, which are especially favorable in this regard [34, 39–43].

The main goal of this work is to construct circuits for *addressable* and *fault-tolerant* logical quantum gates in *small* instances of *high-rate* qLDPC codes, which can be realized in near-term experiments. A particular class of qLDPC codes based on a quasi-cyclic lifted product construction are lift-connected surface (LCS) codes [1]. They consist of sparsely-interconnected copies of 2D surface code patches and can be implemented in a 3D-local fashion. The $[\![15, 3, 3]\!]$ LCS code encodes 3 logical qubits into 15 physical qubits and can correct any single Pauli error. Figure 1 a) depicts a visualization of the code. It is made up of 3 interconnected copies of distance $d = 2$ error-detecting surface codes. Compared to other qLDPC code constructions, in particular asymptotically good [44, 45] or bivariate bicycle codes [8], LCS codes fall short with respect to asymptotic parameters: $[\![n, k, d]\!]$ of LCS codes asymptotically scale like disjoint copies of surface codes. They offer, however, a well defined constructive family of QEC codes with attractive small to intermediate qubit number members. In a memory setting, these have been shown to achieve similar logical error rates as surface codes while requiring up to four times fewer physical qubits [1].

In this work, we systematically construct all quantum gate operations required to span the logical Clifford group in LCS codes. We show such a gate operation, a logical Hadamard gate, acting on one of the three logical qubits, in Fig. 1 b). For the $[\![15, 3, 3]\!]$ code, the logical gate set $\{\overline{H}, \overline{S}, \overline{CX}\}$ is rendered FT via adding a small number of *flag* qubits to our construction. Our flag-FT Clifford gates are fully deterministic and no postselection on mid-circuit measurement results is required. By also explicitly providing an FT magic state that can facilitate a logical $\pi/4$-rotation, we complete the universal gate set *Clifford+T*.

This work is structured as follows. In Sec. II, we introduce the necessary background on error correcting codes and fault tolerance. Sec. III discusses the synthesis of circuits implementing logical Clifford gates targeting arbitrary (pairs of) logical qubits within LCS codes. All gates that we discover this way respect a round-robin structure. Subsequently in Sec. IV, we lever insights from Ref. [46] to add flag qubits to our round-robin gates that effectuate our FT Clifford gate set. The performance of our FT gate set, including FT magic state preparation, under circuit-level noise, is scrutinized in numerical simulations in Sec. V. We provide conclusions and an outlook on future work in Sec. VI.

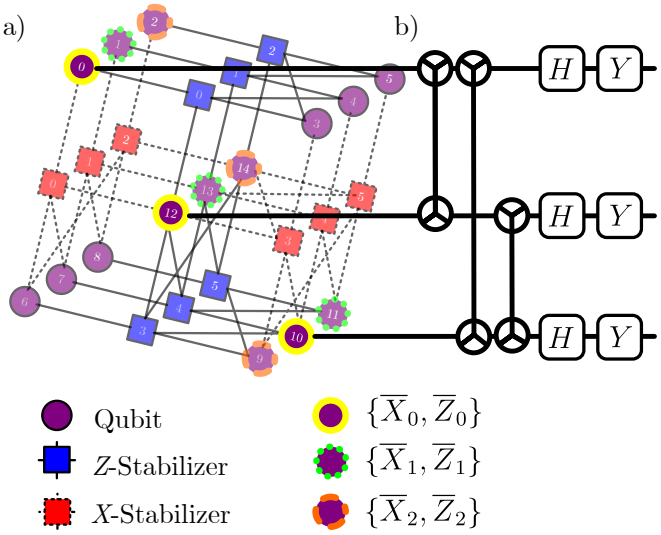

FIG. 1. a) Tanner graph of $[\![15, 3, 3]\!]$ Lift-connected surface code (LCS) and round-robin logical $\overline{H}$ gate. Circles represent data qubits, blue and solid squares the Z-stabilizers and red and dashed squares the X-stabilizers. This LCS code corresponds to three interconnected copies of $d = 2$ surface codes. We indicate the support of the three logical $\overline{X}$- and logical $\overline{Z}$-operators by a yellow solid, green dotted and orange dashed border, respectively. b) For one logical operator we also show the targeted round-robin logical $\overline{H}$ gate. It consists of all-to-all YCY and transversal physical H and Y gates on qubits in the support of the logical operator. In the main text, we explain how to render this logical gate circuit fault-tolerant.

II. ERROR CORRECTED QUANTUM CIRCUITS AND FAULT TOLERANCE

We consider circuits that can correct errors by implementing a stabilizer code $\mathcal{C}$ [10, 47]. A quantum error correcting code encoding k logical qubits is defined by a generating set of stabilizer operators $\mathcal{S} = \{S_i\}_{i=0}^{n-k-1} \subseteq \mathcal{P}^{\otimes n} \setminus \{-I\}$, an Abelian subgroup of the n-qubit Pauli group. The code space is defined as the set of $+1$ eigenstates of all stabilizer generators. Encoded Pauli operators are generated by the normalizer of $\mathcal{S}$ in $\mathcal{P}^{\otimes n}$, $\overline{\mathcal{P}} = \mathcal{N}_{\mathcal{P}^{\otimes n}}(\mathcal{S}) \setminus \mathcal{S}$. These operators preserve the stabilizer group, but they are not stabilizers. Therefore, they act nontrivially in the codespace. The minimum weight, i.e. number of non-identity Pauli terms, of any logical operator is the *distance* d of the code. A QEC code with distance d can correct any error of weight up to $t = \lfloor \frac{d-1}{2} \rfloor$.

We adapt a circuit-centric perspective on error correction. Typically, the realization of an error-correcting code induces constraints on the possible measurement outcomes of the corresponding circuit. As an example, measuring a stabilizer operator on the n physical qubits used to encode k logical qubits always yields a $+1$ measurement outcome in the absence of noise. Faults in the execution of the circuit can flip the measurement outcomes to -1. This perspective is formalized in the frame-

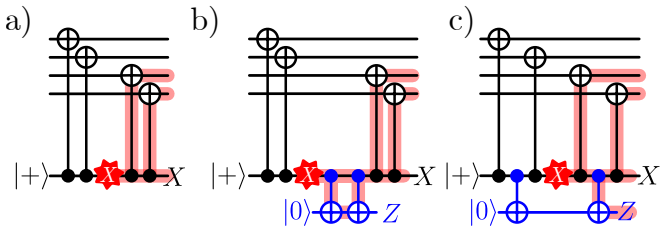

FIG. 2. Flagging a 4-body Pauli-X measurement to catch correlated errors. a) An X-fault on the ancilla qubit after the second entangling CX propagates to a weight-2 XX error on the data qubits. b) We insert an identity measurement using an additional flag qubit. This does not change the original measurement, and as such, does not detect the fault. c) Commuting one gate through the second CX can catch the X-fault as it now only propagates to the ancilla once and flips the flag-measurement.

works of spacetime codes [48], detector error models [49] or Pauli webs and Pauli flows in tensor networks [50, 51]. A matrix D encodes the constraints on the measurement outcomes $\mathbf{m} \in \mathbb{F}_2^{n_\mathrm{m}}$ such that in the noise-free execution of the circuit, the *detector outcomes* $\mathbf{d} \in \mathbb{F}_2^{n_\mathrm{D}} = D\mathbf{m}$ are trivial, $\mathbf{d} = \mathbf{0}$. The (sets of) deterministic measurements are typically referred to as *detectors*.

We define a circuit-level noise model by assigning Pauli noise channels to each faulty location $\ell \in \mathcal{L}$. We refer to a realization of a non-identity term of a noise channel as a (elementary) fault. A *fault path* $F = (P \subseteq \mathcal{P}^{\otimes n}, L \subseteq \mathcal{L})$ of order $w_F = |F|$ is a collection of elementary faults at w_F locations in the circuit. In a uniform noise model with parameter p, any order w_F fault path occurs with probability $\mathcal{O}(p^{w_F})$. We denote the propagation of the fault path to the error E on the data qubits at the end of the circuit by $E(F) \in \mathcal{P}^{\otimes n}$. We call the *syndrome* of the error its commutation relations with the stabilizer generators of the code, $\mathbf{s}(E) = (\langle E, S_i\rangle)_{i=0}^{n-k-1}$. Here $\langle P, P'\rangle = 0$ if $[P, P'] = 0$ and else 1.

A *flag* is an auxiliary qubit that is employed to detect faults that result in correlated errors on data qubits [27, 52]. To that end, the flag ancilla implements a projective measurement of the identity, typically using two consecutive controlled-Pauli (CP) gates, $(CP)^2 = I$. This flag measurement does not change the (noise-free) action of the circuit, but adds an additional detector to the circuit. Commuting one of the gates to another location in the circuit again does not change the action of the circuit. It, however, allows for faults $P' \neq P$ in between the flag gates to propagate to the ancilla and subsequently be detected by the measurement of the flag. We show this principle in Fig. 2 using a 4-body X-measurement.

By detecting more faults, flags can turn circuits *fault-tolerant* (FT). Loosely speaking, an encoded circuit is (t-) FT if it is possible to correct for any number of up to t faults. At the same time, faults occurring during the circuit should not propagate uncontrollably in order to prevent accumulation of errors. In Fig. 3, we illustrate

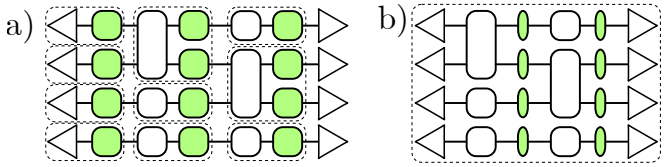

FIG. 3. Two notions of fault-tolerant error-correction circuits we consider in this work. We consider a Clifford circuit with encoded state preparation ($\triangleleft$), logical single- and two qubit gates ($\square$) and final measurements ($\triangleright$). We denote error correction cycles by a green box. a) Traditional Gadget Fault Tolerance. Each gadget is independently decoded (dashed rectangle). For overall fault-tolerance, this requires a strictly fault-tolerant implementation of the gates with a fully fault-tolerant round of error correction after each logical gate. b) In *Algorithmic Fault Tolerance*, as much information as possible is included in the decoding. For the circuits considered, this can be all detector outcomes of the circuit, including final measurement. This can lead to reduced requirements on overhead of error correction cycles while staying fault tolerant, as indicated by thinner squares. For *transversal algorithmic fault tolerance*, this can be as low as $\Theta(1)$ rounds of syndrome measurements per logical gate [59].

two notions of fault-tolerant error correction.

On one side, Fig. 3 a), each physical location is replaced by a fault-tolerant *gadget* that consists of encoded versions of the gates with subsequent fault-tolerant error correction cycles. This has been introduced for concatenated distance-3 codes in Refs. [53, 54]. To achieve fault-tolerance for the full circuit, each circuit element (state preparation, gate or measurement) has to be individually fault-tolerant. There are numerous ways to achieve that, e.g. using encoded ancillae (cat states for Shor-style, logical Pauli states for Steane- and Knill-type EC), $\Theta(d)$ rounds of syndrome measurements or flag-based approaches [2, 55–58]. Encoded gates are individually decoded and, e.g., using Monte Carlo simulations, an effective logical error rate can be estimated. For a fault-tolerant implementation using a distance d code, the logical error rate scales as $p_L \propto p^{t+1}$ in the low-p regime.

On the other side, Fig. 3 b), decoding is performed using as much information as possible. For Clifford circuits, this can be all detector outcomes of the circuit. Typically, such simulations show lower logical error rates, mainly for two reasons. The first is that correlations of errors between gadgets can be accounted for. Also, for a final single-qubit measurement data- and measurement-errors are equivalent. In a setting of purely transversal gates, it is also possible to use more compact syndrome measurement circuits ($\Theta(1)$ rounds per logical operation), while keeping fault-tolerance in the sense of suppression of all errors up until order p^t. This is referred to as *transversal algorithmic fault-tolerance* in Ref. [59].

In the following, we define fault-tolerant gadgets for $d = 3$ similar to Refs. [54, 60]. We then show fault-tolerant error correction protocols using flags, similar to Ref. [58] and define the corresponding protocols for logi-

cal gates.

The workhorse for these definitions is the property of *distinguishability* of faults, which relaxes traditional proofs considering solely the weights of errors [54, 58, 60]. We say that the set of all fault paths of order up to t, $\mathcal{F}^{(t)} = \{F : w_F \leq t\}$, is *distinguishable* if for all pairs of fault paths $F_i, F_j \in \mathcal{F}^{(t)}$ at least one of the following distinguishability criteria is satisfied:

1. $\mathbf{d}(F_i) \neq \mathbf{d}(F_j)$:
   The fault paths lead to different detector outcomes.

2. $E(F_i) \sim_{\mathcal{S}} E(F_j)$:
   The fault paths propagate to stabilizer-equivalent errors, i.e. the errors have the same syndrome but only differ by an element of the stabilizer group.

3. $\mathbf{s}(E(F_i)) \neq \mathbf{s}(E(F_j))$:
   Their errors have different syndromes and can therefore be distinguished by a noise-free round of stabilizer measurements.

For a circuit with a distinguishable fault set, a subsequent round of noise-free syndrome measurements allows one to correct all fault paths in $\mathcal{F}^{(t)}$, up to stabilizer-equivalence. We refer to the set of propagated errors of that fault set as $\mathcal{E}^{(t)}$.

A fault-tolerant error correction gadget for a code with distance $d = 2t+1 = 3$ has to be constructed such that if no more than $t = 1$ fault occurs, an ideal decoder identifies the correct logical state. A simple, but not optimized, protocol that achieves this is shown in Fig. 4. We denote by $\{S_i\}$ a measurement of all stabilizer generators using single auxiliary qubits. By a superscript f, we denote a measurement of all stabilizer generators such that the circuit has a distinguishable fault set $\mathcal{F}^{(1)}$. Typically, a *flagged* stabilizer measurement can achieve that [58].

In the $d = 3$ case ($t = 1$), any non-zero detector outcome of a first flagged syndrome measurement circuit $\{S_i\}^f$ indicates that (at least) one fault has occurred. Then, no more fault will occur in first order $\mathcal{O}(p)$ and we can re-measure the stabilizers without flags to obtain the correct syndrome. If all detector outcomes are 0, the distinguishability of all faults in the flagged syndrome measurement circuit ensures that no uncorrectable fault occurred during the gate. In that case, no subsequent action is necessary.

For logical gates, the fault-tolerant gadget acting on $k' \leq k$ encoded qubits $\{i\}$ has to fulfill similar conditions: if no more than 1 fault occurs, an ideal decoder doesn't induce a logical error. In contrast to Ref. [54], we do not require a bound on spread of error weight within one code-block for fault-tolerance - in our case this is handled by the distinguishability of faults, similar to Ref. [60]. In Fig. 5, we show a simple (but not optimal) protocol to construct a fault-tolerant gate for $d = 3$, $t = 1$ (*1-fault-tolerant*). Again, if the fault set $\mathcal{F}^{(1)}$ is distinguishable in the flagged gate G^f, then a following noise-free round of syndrome extraction would correct all these faults. The

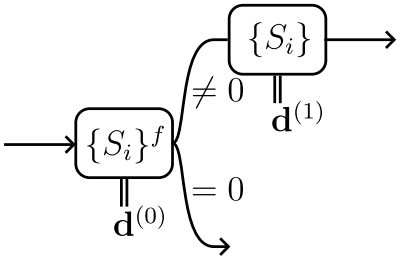

FIG. 4. 1-Fault-tolerant error-correction protocol. Start by measuring the stabilizer generators with flags $\{S_i\}^f$, resulting in detector outcomes $\mathbf{d}^{(0)}$. If any detector has non-trivial outcome, we know some fault has occurred. A subsequent (non-flagged) stabilizer generator measurement acts as a noise-free round of stabilizer measurements in order $\mathcal{O}(p)$, allowing for unique identification of all $\mathcal{O}(p)$ faults. If all detectors have trivial outcome, the distinguishability of $\{S_i\}^f$ ensures that no or no uncorrectable fault has occurred. In that case, nothing has to be done.

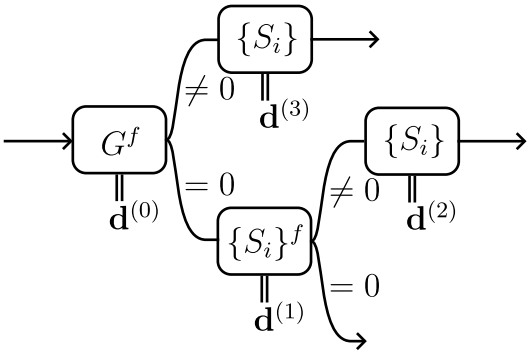

FIG. 5. 1-Fault-tolerant gate protocol. Start by applying the (flagged) unitary gate G^f that has a distinguishable fault set. If any of the detector outcomes $\mathbf{d}^{(0)}$ is non-trivial, a fault has occurred and a non-flagged round of stabilizer measurement is sufficient to correct all faults in order $\mathcal{O}(p)$. For a trivial detector outcome, we append a round of FT error correction, cf. protocol in Fig. 4.

protocol therefore consists of applying the (flagged) unitary gate G^f that has a distinguishable fault set. If any of the detector outcomes $\mathbf{d}^{(0)}$ is non-trivial, a fault has occurred and a non-flagged round of stabilizer measurement is sufficient to correct all errors from fault paths of order $\mathcal{O}(p)$. For a trivial detector outcome, we append a round of FT error correction, cf. Fig. 4.

Both protocols can be decoded by generating a lookup table for each branch, i.e., by simulating each $\mathcal{O}(p)$ fault path and recording the detector outcomes and propagated errors. The FTEC protocol therefore has two, the FT gate protocol three lookup tables. In a simulation, we apply these corrections and then verify whether the residual error is still correctable. For that, we again use a lookup table. When constructing this final lookup table, it is important to include all propagated errors in $\mathcal{E}^{(t)}$ that have trivial detector outcomes. Otherwise, errors of weight $> t$ that could be corrected because of

distinguishability might be misidentified.

In the following sections, we construct circuits that implement the full logical Clifford group for any LCS code, i.e., the set $\langle \overline{S}_i, \overline{H}_i, \overline{C_i X_j} \rangle$ for $i, j \in \{1, \ldots, k\}$. We then introduce flags that lead to distinguishable fault sets for $d = 3$ LCS codes such that the gates can be used in the protocols we introduced above. Finally, we benchmark the logical gates in the code family member with the lowest physical qubit count, the $[\![15, 3, 3]\!]$-LCS code.

III. ROUND-ROBIN GATES IN LCS CODES

In this section, we explain the synthesis of our gates starting from a general decoding-encoding circuit. In the 2-qubit (logical) gate case, this construction leads to the round-robin gates introduced in Refs. [28, 61]. As such, the circuits implementing the gates are not fault-tolerant and require the flag and stabilizer measurements shown in the next Sec. IV. In LCS codes, logical $\overline{X}$- and $\overline{Z}$-operators can be chosen to act on the same physical qubits [1]. This implies that the logical operators act on an *odd* number of qubits to ensure anticommutation. In this section, we focus on the $d = 3$ case, for which we construct a fault-tolerant version in the next section. We refer to the appendix D for the general (non-FT) construction for arbitrary distance.

Encoded logical operators can be understood in terms of their action on the stabilizers and logical Pauli operators of the error correcting code: A logical gate $\overline{G}$ transforms the (canonical, i.e. uniquely anticommuting) logical Pauli operators of logical qubit i in the same way physical Pauli operators are transformed by the physical gate. A logical Hadamard gate, e.g., has to act as $\overline{H}_i \overline{X}_i \overline{H}_i = \overline{Z}_i$ and $\overline{H}_i \overline{Y}_i \overline{H}_i = -\overline{Y}_i$. Additionally, the logical gate has to preserve the code space, i.e. keep the stabilizer group invariant, $\overline{G} \mathcal{S} \overline{G}^\dagger = \mathcal{S}$.

We start by constructing circuits that implement a gate G acting on a single logical qubit. In a first step, shown in Fig. 6 a), the canonical pair of logical operators $\overline{X}$ and $\overline{Z}$ is contracted onto a physical qubit ℓ using a unitary decoding circuit U_D. This circuit also has to map all stabilizers such that they have no support on the qubit ℓ. Then, the gate can be applied on the physical level, ensuring the correct transformation of the (now single-qubit) logical Pauli operators, and the invariance of all stabilizers. The re-encoding circuit $U_E = U_D^\dagger$ maps the correctly transformed single-qubit logical Pauli operators back to the full distance logical operators of the code and restores the original stabilizers.

Independent of the actual implementation of U_D, this logical gate is not fault-tolerant: a fault in the application of the physical gate is undetectable by construction, since no stabilizer has overlap with the designated qubit ℓ. This physical error propagates through the re-encoding circuit to a logical error. Fault-tolerance can be achieved using concatenation, i.e. encoding the physical qubits in a code that support a fault-tolerant implementation of the gate G [28]. This, however, comes at the cost of a large physical qubit overhead.

We now present an alternative path to fault-tolerance, shown in Fig. 6 b), that relies on the structure of the decoding circuit U_D. On a high level, we propagate G through the re-encoding circuit to the end of the circuit. This results in an adapted circuit $\tilde{U}_{DE}$ followed by a transformed $\tilde{G}$.

Valid de- and encoding circuits are fan-out CX gates on the common support of logical $\overline{X}$- and $\overline{Z}$-operators [62],

$$U_D = \bigotimes_{L \in \mathrm{supp}_G} \prod_{q \in \mathrm{supp}_L \setminus \ell} C_c X_\ell \prod_{q \in \mathrm{supp}_L \setminus \ell} C_\ell X_c, \tag{1}$$

shown in Fig. 6c). One can verify that these indeed fulfill the conditions spelled out above: The full overlap logical operators propagate to the designated qubit ℓ. Since stabilizers commute with the logical operators, they have even overlap with the fan-out gates and have no support on qubit ℓ after the gate. We show this circuit with the propagation of Pauli operators for a logical $\overline{H}$ gate in a $d = 3$ code in Fig. 7 a). Next, we propagate the central bottleneck single-qubit Clifford gate G to the end of the circuit. Applying circuit identities, we obtain circuits $\tilde{U}_{DE}$ that consist of all-to-all *Pauli-controlled-Pauli* gates (aPCP), followed by a transversal $\tilde{G} = G^{\otimes n_G}$, shown in Fig. 6 b) and d). We denote the aPCP gates as

$$PCP'_Q := \prod_{i,j \in \{Q\}} P_i C P'_j. \tag{2}$$

with physical Pauli-controlled-Pauli gates that are defined as $PCP' = \frac{I-P}{2} \otimes I + \frac{I+P}{2} \otimes P'$. In the following circuits, we draw the PCP, with $P \in \{X, Y, Z\}$ as

XCX :  , YCY :  , ZCZ :  , (3)

which should not be confused with notation for Mølmer-Sørensen gates of Ref. [63]. We also sometimes drop the Z for the control qubit and write $CP \equiv ZCP$.

Similar to before, we now investigate the action of the Clifford aPCP gates on Pauli operators with even (stabilizers) or full (logical operators) overlap. Note that the aPCP gates act on an odd number of qubits $\{Q\}$, $|\{Q\}| = 1 \mod 2$. Pauli operators that have even overlap with $\{Q\}$ and anticommute with the controls and targets P, change their Pauli types,

$$XCX_Q Y^{\otimes 2k} XCX_Q = Z^{\otimes 2k}, \tag{4}$$

$$YCY_Q X^{\otimes 2k} YCY_Q = Z^{\otimes 2k}, \tag{5}$$

$$ZCZ_Q X^{\otimes 2k} ZCZ_Q = Y^{\otimes 2k}. \tag{6}$$

On operators with full overlap, the aPCP gates act proportionally to the identity,

$$PCP_Q \tilde{P}^{\{Q\}} PCP_Q = (-1)^{\frac{|\{Q\}|-1}{2}} \tilde{P}^{\{Q\}}, \tag{7}$$

$$\text{for} \quad P \in \{X, Y, Z\}, \tilde{P} \in \{X, Y, Z\} \setminus P. \tag{8}$$

We can now combine these actions with transversal gates to generate the full single-qubit logical Clifford group: Consider e.g. the YCY_Q gate that swaps $X \leftrightarrow Z$ on stabilizers overlapping the logical operators. A transversal Hadamard H_Q fixes that, $Z \leftrightarrow X$ such that the stabilizer group is invariant. The logical operators undergo $\overline{X} \xrightarrow{YCY_Q} (-1)^{\frac{|\{Q\}|-1}{2}}\overline{X} \xrightarrow{H_Q} (-1)^{\frac{|\{Q\}|-1}{2}}\overline{Z}$ and equivalently for $\overline{X}$. A transversal Y_Q if $\frac{|\{Q\}|-1}{2} = 1 \bmod 2$ fixes the final phase. We again show this for a Hadamard gate in a $d = 3$ code, together with the propagation of Pauli operators in Fig. 7 b). Together with the targeted $\overline{S}$ gate (ZCZ-gates and transversal $S_{\mathrm{supp}(L)}$) in Fig. 8, this generates the single-qubit Clifford group.

For logical operators of weight d, the decoding approach requires $4(d-1)$ two-qubit gates resulting in a circuit depth $4d-3$. The round-robin approach using the aPCP gates needs more, i.e. $\frac{d(d-1)}{2}$ two-qubit gates. Because the aPCP gates can be parallelized, the circuit depth is reduced to $d+1$. With respect to fault-tolerance, the round-robin construction shows another advantage: the central gate location which could already lead to a logical error in $\mathcal{O}(p)$ is removed. This opens the possibility to introduce flags to turn the circuit FT.

Before showing these flag construction, let us add the final ingredient to generate the n-logical qubit Clifford group: a targeted two-qubit *intra*-block entangling Clifford gate. We follow the same approach as for single-qubit gates: We decode two logical qubits onto two physical qubits ℓ_1, ℓ_2. We apply the desired entangling gates, e.g. $C_{\ell_1}X_{\ell_2}$. Then, both qubits are re-encoded into the code. Propagating the entangling gate G through the encoding circuit results in the well-known round-robin gate of Ref. [28]. We show the targeted logical $\overline{CX}$ in Fig. 8 c). In the same way, arbitrary logical PCP can be constructed. Note that the *inter*-block $\overline{CX}$ between two code blocks is transversal since LCS codes are CSS [1]. This is, however, not a targeted $\overline{CX}$ but acts on *all* logical qubits of each block simultaneously.

We stress, again, that the round-robin gate constructions get rid of the bottleneck in the middle of the decode-gate-encode approach. There are still, however, single faults that lead to an uncorrectable error in first order. Consider, e.g., an XX-fault on the two qubits of the last YCY gate in the logical $\overline{H}$-gate depicted in Fig. 9 a). It propagates to an IZZ error at the end of the circuit. This is not correctable because the fault is indistinguishable from a single-qubit X-fault on the first qubit: They both flip the same detectors, and are not stabilizer equivalent but multiply to the logical operator. In the next section, we show how to carefully construct stabilizer- and flag measurements to make these logical gate circuits fault-tolerant.

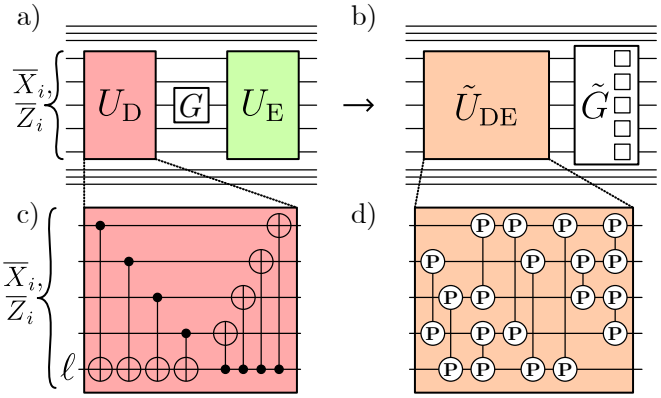

FIG. 6. General construction of the round-robin gates. a) A gate G acting on n_G logical qubits can be implemented by a unitary decoding circuit U_D acting on the support of the logical operators, mapping them to n_G physical qubits. This is followed by an application of the gate and a re-encoding $U_E = U_D^\dagger$. Since G acts on a physical level, this location is considered a bottleneck in the sense that any error on that gate is undetectable and propagates to a logical error through $\mathcal{U}_E$. Here we show a single-qubit gate. b) To circumvent this bottleneck, we commute G through the encoding circuit to the end. This results in an adapted circuit $\tilde{U}_{DE}$ followed by a transformed $\tilde{G}$. If G is a single-qubit Clifford gate, $\tilde{G}$ is transversal. c) A valid choice for a unitary decoding circuit U_D consists of fan-out CX gates, mapping the logical $\overline{X}$ and $\overline{Z}$ operator onto a single physical qubit ℓ. d) Commuting single-qubit Clifford gates through the re-encoding circuit results in a $\tilde{U}_{DE}$ that is an all-to-all Pauli-controlled-Pauli gate, i.e. a PCP gate is applied to every pair of qubits in the support of the logical $\overline{X}$ and $\overline{Z}$ operator.

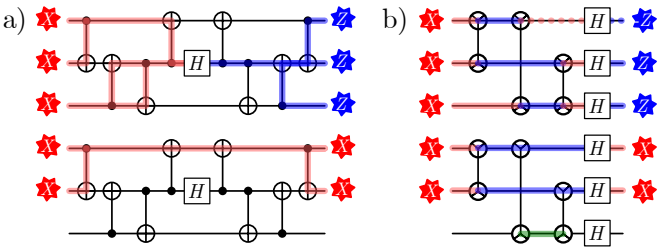

FIG. 7. Propagation of Pauli operators through the Hadamard gate on three qubits. We follow the convention X, Y, Z = red, green, blue. A contribution of -1 to a global phase is indicated by a dotted highlight. a) In the decoding approach, the logical operators (top) are contracted to the bottleneck location, transformed, and expanded back to the full support. Stabilizer generators with even overlap (bottom) propagate around the bottleneck and stay invariant. b) In the round-robin scheme, the logical operators (top) stay the same through the all-to-all YCYs, but pick up a phase of -1 each time two different Pauli operators (here Z_1X_3) propagate through a Pauli-controlled-Pauli (here Y_1CY_3) gate. They change their type upon acting with the transversal Hadamard. Stabilizer generators with even overlap (bottom) retain their support after the all-to-all YCYs, but change their Pauli type $X \leftrightarrow Z$. The transversal Hadamard fixes this to render the stabilizer group invariant.

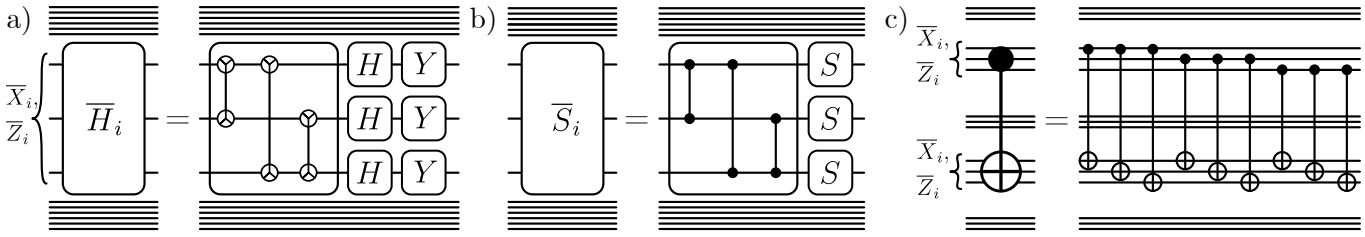

FIG. 8. Targeted logical Clifford Gates in $d = 3$ LCS codes. a) The logical $\overline{H}$ gate consists of all-to-all YCY-gates followed by transversal H and Y-gates on the support of the logical operator. b) The logical $\overline{S}$ gate uses all-to-all ZCZ-gates followed by transversal S-gates on the support of the logical operator. c) The logical intrablock $\overline{C_iX_j}$ gate is a round-robin, i.e., all-control-to-all-target CX.

IV. FLAG-FT GATE CONSTRUCTIONS

In what follows, we present our detailed constructions of the FT universal gate set Clifford+T on $[\![15, 3, 3]\!]$-LCS codes. We specify stabilizer generators and a logical operator basis in appendix A. We first consider making single-qubit gates FT. Together with our construction of the FT CX gate, which we discuss subsequently, the FT Clifford group is generated. Finally, we provide two FT magic state preparation schemes that allow us to complete the FT universal gate set via gate teleportation.

A. FT single-qubit gates

As seen in the previous section, the round-robin gates are not fault-tolerant. This implies that there are still pairs of locations that lead to undetectable logical errors. Using the $\overline{H}$-gate as a guiding example, we explain how flag constructions based on Ref. [46] render the fault set $\mathcal{F}^{(1)}$ distinguishable. The flagged gate can then be used in the Flag-FT protocol described above.

The general strategy is to first flag all two qubit gates to catch correlated errors like the XX-fault described above. Then, all remaining problematic faults are equivalent to single-qubit faults before the gates. These are detected by inserting stabilizer measurements.

For the bare logical gate, we find via computer-aided inspection that 66 out of 1953 possible order-p^2 faults lead to an undetected logical error. 48 of these contain correlated 2-qubit faults, like the XX fault of Fig. 9 a), that are not equivalent to single-qubit faults before the gate (i.e. $XI, YI, ZI, IX, IY, IZ, XZ, ZX, XY, YX$).

All these correlated two-qubit faults on data qubits after a YCY-gate can be caught using two auxiliary qubits as flags, cf. Ref. [46]: for the logical Hadamard gate, the YCY gates are flagged using a Y-flag on one qubit i.e. a flag implementing the measurement of identity using CY-gates. This Y-flag catches X- and Z-faults. On the other qubit, a Z-flag catches YY-faults. We show the same fault-pair from before, that is now detected by flag measurements in Fig. 9 b).

There are still faults that are undetected by flags. Despite the much larger number of fault locations (44850 fault pairs), still only 68 of those lead to an undetectable logical error. These fault pairs contain terms at circuit locations at or equivalent to X- or Y- faults at the input of the circuit. We show corresponding locations in Fig. 9 b). The reason for these faults not being detected is that they occur before the flags and therefore propagate onto the ancilla twice. The only chance to identify these remaining faults is to insert stabilizer measurements after the first gate of the initial flags. The stabilizers have to be chosen such that they overlap and anticommute with these initial faults. For the $\overline{H}$-gate, the faults are Pauli X and Z - the stabilizer operators therefore have to have a Pauli-Y term at these locations. For the logical $\overline{H}_0$, on the logical operator supported on qubits 0, 10 and 12, we choose to measure the stabilizer $S^{(0)} = \prod_{i=0}^{5} S_X^{(i)} S_Z^{(i)} = Y_0Y_1Y_2Y_{12}Y_{13}Y_{14}$ after the first gate of the Y-flag for gate Y_0CY_{10}, and $S^{(1)} = S_X^{(3)} S_Z^{(3)} = X_3Z_6Y_9Y_{10}Y_{12}Y_{14}$ after the first gate of the Y-flag for gate $Y_{10}CY_{12}$. We show this final circuit with a distinguishable fault set in Fig. 9 c). We show the full circuit for the logical $\overline{S}$ in the appendix B and tabulate the measured stabilizers in Tab. IV of appendix B 2.

B. FT CX gate

For the logical $\overline{CX}$ gate in the bare round-robin construction, 28 out of 16290 fault pairs lead to an undetected logical error. By flagging physical control and target qubits, 36 out of 69378. Note that because the $\overline{CX}$ has a Z-control and an X-target, the Z- and X-flags contain no extra gate due to anti-commutation. Also, we find that we don't have to flag each gate separately, but one flag spanning the first two controls and targets of each qubit is sufficient. Again, including stabilizer measurements allows one to distinguish the remaining, incoming faults such that $\mathcal{F}^{(1)}$ is distinguishable. For a $\overline{C_0X_1}$, on the logical operators supported on qubits $(0, 10, 12)$ and $(1, 11, 13)$ respectively, the measured stabilizers are $S_Z^{(0)} = Z_0Z_3Z_4Z_{12}$, $S_Z^{(3)} = Z_6Z_9Z_{10}Z_{12}Z_{14}$ and $S_X^{(1)} = X_1X_7X_8X_{13}$, $S_X^{(5)} = X_5X_9X_{11}X_{13}X_{14}$. We show the full circuit in Fig. 10. Circuits for other combinations of logical controls and targets have the identical

gate	n(entangling gates) bare	flagged(nFT)	flagged+EC(FT)	n(flag qubits)
$\overline{H}$	3	18	75	6
$\overline{S}$	3	18	72	6
$\overline{CX}$	9	21	75	6

TABLE I. Resource requirements for the logical Clifford gates. We show the number of entangling gates for the bare, non-FT gates and the flagged non-FT logical gates (without intermediate stabilizer measurements). There, the single-qubit logical gates require less compared to the two-qubit logical gate. For the FT gates, we include the entangling gates required for the intermediate stabilizer measurements, as well as a set of stabilizers to complete a generating set. The $\overline{H}$ and the $\overline{CX}$ gate now both require 75 entangling gates. The $\overline{S}$ gate needs three entangling gates less because the stabilizer generators measured have less weight compared to the $\overline{H}$ gate. All flagged protocols employ at most 6 additional flag qubits.

structure while measuring different stabilizer generators. We tabulate those in Tab. IV in appendix B 2.

The total overhead of the FT logical gates depends on their embedding within an FT protocol. In particular, within the gadget-FT protocols of Sec. II the number of gates depends on the executed circuit branch. As a reference, we consider the FT flagged gate and enough stabilizer measurements to ensure a generating set has been measured during the gate. We summarize the number of entangling gates and flag qubits required for the single- and two-logical qubit gates in Tab. I. Notably, all logical gates use the same number of flag qubits. The larger number of entangling gates during the logical gate in a $\overline{CX}$ is offset by the smaller number required for flags. This results in very similar numbers for logical single- and two-qubit Clifford gates.

C. FT Magic state preparation

In order to complete the FT universal gate set Clifford+T for the $[\![15,3,3]\!]$ LCS code, we see two possible routes to take. We require either a flag-FT T-gate circuit or a routine to fault-tolerantly prepare a magic state which can then facilitate the FT T-gate via gate teleportation [34, 53, 64]. Using the same approach as introduced in the previous sections, we arrive at a (non-FT) logical T-gate that uses non-Clifford two-qubit gates, shown in appendix D, Fig. 18 d). These cannot be flagged with conventional methods, placing a roadblock on the first route to a direct, unitary, flag-FT T-gate. We therefore opt for the second route and implement two strategies for FT magic state preparation [63–65]. We use the fact that

$$\overline{H}_i|\overline{\psi}_1,\overline{\psi}_2,...,\overline{H}_i,...,\overline{\psi}_k\rangle = +1|\overline{\psi}_1,\overline{\psi}_2,...,\overline{H}_i,...,\overline{\psi}_k\rangle \tag{9}$$

for any $i \in \{1,k\}$ and logical states $\overline{\psi}_i$. We slightly abuse notation to write in short form $\overline{H}|\overline{H}\rangle = +1|\overline{H}\rangle$, with the logical Hadamard operator

$$\overline{H} = \frac{\overline{X}+\overline{Z}}{\sqrt{2}} \tag{10}$$

to prepare the H-type magic state

$$|\overline{H}\rangle = \cos\left(\frac{\pi}{8}\right)|\overline{0}\rangle + \sin\left(\frac{\pi}{8}\right)|\overline{1}\rangle \tag{11}$$

on the i-th logical qubit. The first approach is a deterministic protocol that repeatedly measures the logical Hadamard operator on one of the logical qubits. Thereby, a noisy magic state is projected onto the $+1$ eigenstate of $\overline{H}$ [64]. A repeat-until-success variant of this protocol allows one to only measure the logical Hadamard operator once, followed by a QEC cycle for detection of residual, otherwise uncorrectable errors. In case that all physical measurements yield $+1$, the magic state preparation succeeds and is guaranteed to be fault-tolerant. The state is discarded in case any measurement yields -1 and the state preparation protocol is repeated [65]. A quantitative comparison of these two variants has previously been given with a focus on the Steane code in Ref. [63].

Here, we realize *addressable* FT magic state preparation within one LCS codeblock and retain the flexibility to choose either the deterministic or repeat-until-success variant for practical implementation. Therefore, every physical gate in the logical Hadamard circuit of Fig. 8a) is equipped with an additional control that connects to a measurement qubit. Only one flag qubit is required to detect faults that can otherwise propagate to undetectable errors. The full FT circuit construction is given on the physical-qubit level in appendix B 3.

V. NUMERICAL PERFORMANCE ANALYSIS

We conduct an extensive numerical study on the performance of our FT gates. A single-parameter circuit-level depolarizing noise model is employed, i.e., every noisy operation acting on n qubits is modeled by the ideal version followed by an n-qubit depolarizing channel of strength p,

$$\mathcal{E}_n(\rho) = (1-p)\rho + \frac{p}{4^n-1}\sum_{i=1}^{4^n-1} P_n^{(i)}\rho P_n^{(i)} \tag{12}$$
$$\text{with} \quad P_n^{(i)} \in \{\{I,X,Y,Z\}^n \setminus I^{\otimes n}\}.$$

We use the same depolarizing noise channel after (before) initializations (measurements) of qubits and assume idling locations as noise-free.

We simulate the two approaches discussed in Sec. II: the algorithmically fault-tolerant circuits and the conventionally fault-tolerant protocols.

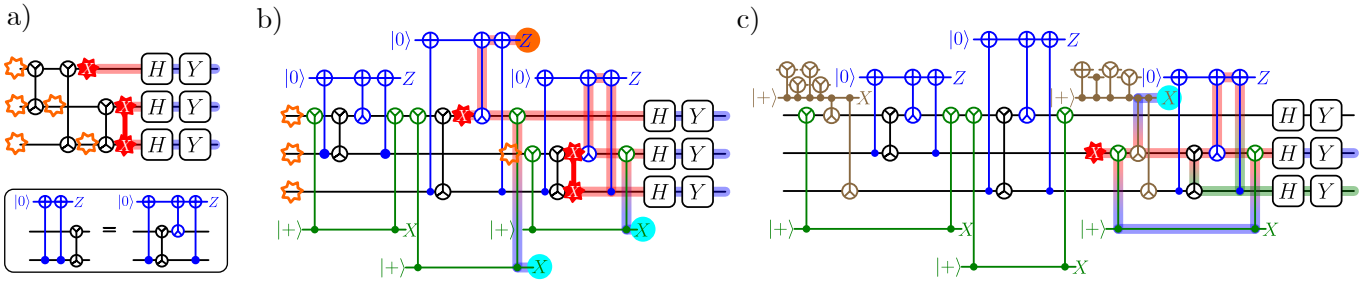

FIG. 9. Flagging the single-qubit logical H-gate to ensure a distinguishable order $\mathcal{O}(p)$ fault-set $\mathcal{F}^{(1)}$. a) The unflagged $\overline{H}$-gate has numerous fault pairs that propagate to an undetected logical error. They occur at almost any location, indicated by orange stars. We also draw such a fault, a combination of a correlated XX-fault after the last YCY gate and a X-fault on the first qubit. b) We wrap Y- and Z-flags around all YCY gates. Since the Z- flag does not commute with the YCY gate, propagating it through results in an XCY gate, reflecting the propagating of a Pauli-Z through a YCY gate. We show this in the lower left corner box. This catches all correlated faults on these gates. In particular, the fault pair from the previous example is now detected. The remaining fault-pairs that lead to undetectable logical errors always include fault that are equivalent to faults on the input. We draw exemplary such locations by orange stars. c) The remaining faults handled by inserting FT stabilizer measurements after the first gate of Y flags (brown). The stabilizers to measure are chosen such that they overlap with the logical operator. We draw an X-fault that has previously been undetected but is now caught by the stabilizer measurement.

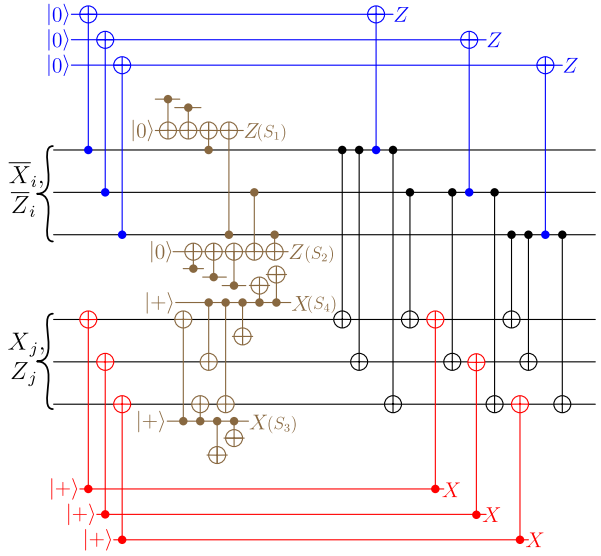

FIG. 10. Flagging the two-qubit logical $\overline{CX}$-gate (black) to ensure a distinguishable order $\mathcal{O}(p)$ fault-set $\mathcal{F}^{(1)}$. This can be achieved by wrapping the first two controls and targets of each physical qubit of the round-robin $\overline{CX}$ with X- and Z-flags (red and blue). To catch the remaining incoming faults, two Z- and X-stabilizer measurements (brown) are inserted after each flag's first gate.

A. Algorithmic fault-tolerance – Memory Experiments

We simulate the Clifford circuits using `stim` [66]. For the algorithmic fault-tolerant protocol, we initialize the logical Pauli state $|\overline{000}\rangle$ of the $[\![15,3,3]\!]$ LCS code without noise and then apply noise-free logical gates to prepare arbitrary logical Pauli eigenstates. This is to focus on the performance of the gates, not including state preparation and measurement (SPAM) errors[1]. For a logical single-qubit gate on the first logical qubit, these are the 6 states $|\overline{0}\rangle, |\overline{1}\rangle, |\overline{+}\rangle, |\overline{-}\rangle, |\overline{+i}\rangle, |\overline{-i}\rangle$. For a logical two-qubit gate, we prepare all 36 combinations of these on two logical qubits. We choose to prepare the logical qubit(s) untouched by the logical gate in $|\overline{0}\rangle$. We add a layer of single-qubit depolarizing noise channels, followed by the flagged gate circuits, cf. Fig. 9 c), Fig. 10 and Fig. 13. We append one round of noisy stabilizer measurements – the expected number of QEC rounds required per logical gate for algorithmic fault tolerance [59]. In this round, we do not need to measure a full generating set of stabilizers and we omit measuring those that have already been measured during the execution of the gate. If we measured a product of stabilizers (e.g. $S_X^{(3)}S_Z^{(3)}$ for the logical $\overline{H}$ gate), we omit one of the factors (e.g. $S_Z^{(3)}$). This ensures that a complete set of stabilizer generators is measured during the protocol. We finally measure all stabilizer generators and adequate logical operators without noise. For example for $\overline{H}_i$, if the logical qubit is prepared in an eigenstate of logical $\overline{X}_i$, we measure $\overline{Z}_i$. For the logical $\overline{CX}$, if the control is a $\overline{X}$-eigenstate, and target in $\overline{Z}$, (e.g. $|\overline{+0}\rangle$) we measure control and target jointly in $\overline{XX}$ and $\overline{ZZ}$. We tabulate all combinations of input bases and logical operators measured in appendix B 2 in Tab. V. We annotate all flag- and stabilizer measurement as *detectors* and the measurements of logical operators as *observables*. This allows us to verify the fault-tolerance of these circuits by calling `.search_for_undetectable_logical_errors(...)` on the noisy `stim` circuit, which returns the circuit distance

[1] Note that an FT state preparation can be achieved using rounds of (noisy) stabilizer measurement, or flag verification methods as e.g. shown in Refs. [65, 67–69].

$d_{\text{circ}} = 3$ for all circuits. This distance is the number of elementary faults required to flip an observable, without flipping any detector. When sampling, we get a set of *detector* and *observable* measurement outcomes. We decode the detector vector using the `tesseract` decoder [70], which returns a prediction for the values of the observables. We then count a logical error if the predicted values do not match the measured values of the observable/logical operators.

For the results, we average over all input states with corresponding logical measurements as described above, as well as over all possible logical (control and) target qubits. As a reference, we also simulate one round of noisy error-correction without any logical gates ($\overline{EC}$). We show the results in Fig. 11 a). As an additional FT-check, we fit a function $p_L = ap^\nu$ to the 5 lowest data points. All logical error rates show a scaling in accordance with the expected fault-tolerant scaling $p_L \propto p^2$.

To estimate a pseudothreshold, i.e. a point where the logical gate starts to outperform the physical gate, we assume that each of the three physical qubits fails with probability $p_i, i \in [1,2,3]$. The total failure probability that any qubit fails then is $p_{\text{fail}} = 1-(1-p_1)(1-p_2)(1-p_3)$. In our general single-parameter noise model, we consider an upper ($p_{\text{th}}^{(+)}$) and lower limit ($p_{\text{th}}^{(-)}$): first, we assume that any of three physical qubits fails with probability p, such that $p_{\text{fail}} = 1-(1-p)^3$. Since in practice, idling qubits fail with some factor $p_{\text{idle}} = \lambda p$, we also look at the limit $\lambda \to 0$ where $p_{\text{fail}} = 1-(1-p)^{n_g}$.

We obtain the pseudothresholds with uncertainties using linear interpolation of the data points upper and lower bounded by the respective statistical errors. The upper limits are in the range of $p_{\text{th}}^{(+)} \approx 4.8 \cdot 10^{-3} - 1.02 \cdot 10^{-2}$. We summarize the fit parameters in Tab. II and the pseudothresholds in Tab. III.

FT type	gate	a	ν
Alg.	$\overline{EC}$	200 ± 10	1.994 ± 0.008
	$\overline{H}$	710 ± 50	2.036 ± 0.009
	$\overline{S}$	530 ± 20	2.023 ± 0.007
	$\overline{CX}$	340 ± 20	2.06 ± 0.02
Gadget	$\overline{EC}$	6000 ± 3000	2.07 ± 0.06
	$\overline{H}$	5300 ± 800	2.01 ± 0.02
	$\overline{S}$	3700 ± 700	1.97 ± 0.02
	$\overline{CX}$	7000 ± 1000	2.04 ± 0.02

TABLE II. Fit-parameters of logical gates obtained from a least-squares fit of the lowest 5 data points to $p_L = ap^\nu$. They show the expected fault-tolerant scaling $\nu \approx 2$.

FT type	gate	$p_{\text{th}}^{(-)}$	$p_{\text{th}}^{(+)}$
Alg.	$\overline{EC}$	$(4.6 \pm 0.01) \cdot 10^{-3}$	$(1.22 \pm 0.02) \cdot 10^{-2}$
	$\overline{H}$	$(1.8 \pm 0.01) \cdot 10^{-3}$	$(4.8 \pm 0.02) \cdot 10^{-3}$
	$\overline{S}$	$(2.14 \pm 0.03) \cdot 10^{-3}$	$(5.99 \pm 0.09) \cdot 10^{-3}$
	$\overline{CX}$	$(6.71 \pm 0.06) \cdot 10^{-3}$	$(1.020 \pm 0.007) \cdot 10^{-2}$
Gadget	$\overline{EC}$	$(3.1 \pm 0.2) \cdot 10^{-4}$	$(9.5 \pm 0.3) \cdot 10^{-4}$
	$\overline{H}$	$(1.9 \pm 0.1) \cdot 10^{-4}$	$(5.88 \pm 0.09) \cdot 10^{-4}$
	$\overline{S}$	$(2.1 \pm 0.1) \cdot 10^{-4}$	$(6.93 \pm 0.07) \cdot 10^{-4}$
	$\overline{CX}$	$(3.93 \pm 0.08) \cdot 10^{-4}$	$(5.91 \pm 0.09) \cdot 10^{-4}$

TABLE III. The pseudothresholds, upper bounds $p_L(p_{\text{th}}^{(+)}) = 1-(1-p_{\text{th}})^3$, are in the order of $10^{-3} - 10^{-2}$ for the algorithmic and 10^{-4} for the gadget fault-tolerance. Lower bounds $p_L(p_{\text{th}}^{(-)}) = 1-(1-p_{\text{th}})^{n_G}$ depend on the number of qubits the gates act on.

B. Conventional gadget-FT Clifford gate simulations

The conventional gadget fault-tolerant protocol starts with a noise-free initialization, followed by a layer of single-qubit depolarizing noise channels on the data qubits. We then apply the full, noisy 1-FT gate protocol, cf. Fig. 5. We decode using lookup tables for each circuit branch as described above. We apply the corrections before simulating a perfect round of error correction that puts the state back to the codespace. A logical error occurs if the residual error anti-commutes with *any* logical operator of the code. We show results for logical $\overline{S}$, $\overline{H}$ and $\overline{CX}$ gate, averaged over all possible logical targets in Fig. 11 b). For reference, we show the logical error rate of the bare error correction ($\overline{EC}$) protocol of Fig. 4. We again fit a function $p_L = ap^\nu$ to the data and find good accordance with the expected fault-tolerant scaling $p_L \propto p^2$ for all logical gates. We summarize the fitting parameters in Tab. II and upper and lower bounds on pseudothresholds as described above in Tab. III. For all the gates, the upper bounds lie in the range $p_{\text{th}}^{(+)} \approx 5.8 - 7 \cdot 10^{-4}$.

C. FT magic state preparation

Numerical statevector simulations are performed using a modified version of `PECOS` [71]. Here, we use the logical fidelity

$$F_{\overline{H}} = \frac{1}{2}\left(1 + \left(\frac{\langle \overline{X}\rangle + \langle \overline{Z}\rangle}{\sqrt{2}}\right)\right) \tag{13}$$

of the logical magic state $|\overline{H}\rangle$ to determine logical failure rates $p_L = 1 - F_{\overline{H}}$ by evaluating $\langle \overline{X}\rangle$ and $\langle \overline{Z}\rangle$.

Logical failure rates of (gadget-)FT magic state preparation are showing the expected p^2-scaling in Fig. 12 in the low-p regime. For the deterministic variant, given in Fig. 12 a), we observe a breakeven point as high as approx. $8 \cdot 10^{-4}$ when comparing to three physical qubits. This point shifts to a slightly smaller value of about $3 \cdot 10^{-4}$ when comparing to a single physical qubit. Both values are very similar to the ones reported above for the gadget-level FT Clifford gates (see Fig. 11 b)). Remarkably, the analogous curve for the repeat-until-success FT

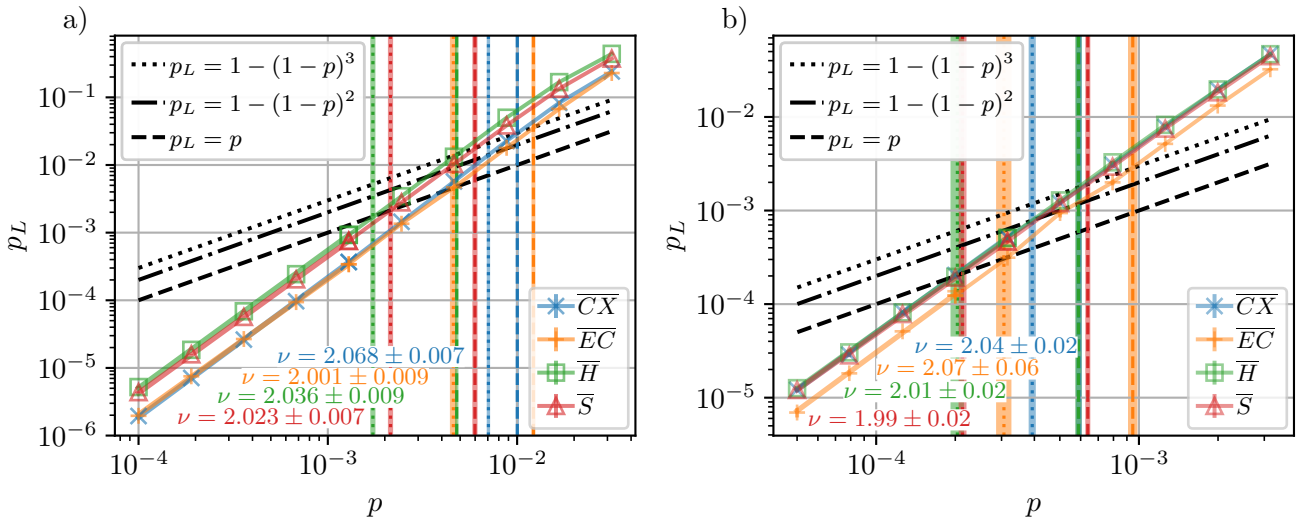

FIG. 11. a) Flag-FT logical Clifford gates in the algorithmic fault-tolerance protocol. The logical error rate of this protocol depends on the input state and corresponding final logical measurement. We show the average logical error probability over all combinations of input states and logical operators addressed. All logical gates, $\overline{H}$, $\overline{S}$ and $\overline{CX}$, show fault-tolerant scaling ($\nu \approx 2$ for a least-squares fit to $p_L = ap^\nu$). We show lower and upper bounds on pseudothresholds as described in the main text. The breakeven point compared to the failure rate of three physical qubits are in the range of $p_{\text{th}}^{(+)} \approx 4 - 6.5 \cdot 10^{-3}$. As reference, we show the logical error rate of one round of noisy stabilizer measurements. b) Flag-FT logical gate gadgets within the conventional fault-tolerance protocol. This protocol is independent of the input state. We show the average logical error probability over all combinations of logical operators addressed. All logical gates, $\overline{H}$, $\overline{S}$ and $\overline{CX}$, show fault-tolerant scaling ($\nu \approx 2$ for a least-squares fit to $p_L = ap^\nu$). We show lower and upper bounds on pseudothresholds as described in the main text. The breakeven point compared to the failure rate of three physical qubits are in the range of $p_{\text{th}}^{(+)} \approx 5.8 - 7 \cdot 10^{-4}$. All error bars are standard Monte-Carlo error bars and can be smaller than the line-width. Pseudothresholds are obtained from the intersection of a linear interpolation of data points, as described in the main text and detailed in appendix C. As reference, we show the logical error rate of the FT EC protocol (cf. Fig. 4).

magic state preparation in Fig. 12 b) reveals that an advantage over physical qubits can be expected for the entire range of physical error rates $p \leq 10^{-2}$ considered. However, this advantage comes at the price of increasing the required average number of preparation trials until the state is accepted. While the average acceptance rate at $p = 10^{-2}$ is as low as approx. 18% (similar to the value reported in Ref. [34]), at the realistically attainable value of $p = 10^{-3}$ already about 85% of trials are accepted. The left-most data point in Fig. 12 b) corresponds to an average acceptance rate of approx. 95% (see Ref. [63] for an analysis how such values translate to an average expected number of trials until success).

The FT magic state can then be used as an input state to a standard gate teleportation gadget to facilitate the action of a FT T-gate on an arbitrary logical target state [47]. The gate teleportation circuit itself consists of FT Clifford gates only. Yet, there could be different variants to employ: One option would be to prepare the FT magic state on one (or more) of the k logical qubits of one LCS block and teleport the T-gate at will to any other qubit within the block or between logical qubits of different code blocks. Alternatively, one may also prepare a full block of k FT magic states in parallel and teleport all at once via a standard transversal $\overline{CX}$ gate to another LCS code block.

D. Discussion

In the following, we summarize the resource requirements for the logical gate circuits. In terms of physical qubit overhead, the flagged single- and two-qubit logical gates on $d = 3$ codes require six flag qubits. For the single-qubit gates, the flag measurements can be scheduled such that at most two flag qubits are used at the same time, reusing them after measurement and reset. For the stabilizer measurements, the same qubits used for regular QEC cycles can be used. In terms of gate overhead, the flag circuits of single-qubit gates employ a total of 15 additional entangling gates, five per flagged two-qubit gate of the unflagged circuit. For the two-qubit $\overline{CX}$ gate, only 12 additional entangling gates are required. In total, the implementation of all FT Clifford gates and FT QEC cycles require at most 33 qubits, such that the gates are attractive candidates for near-term implementation in medium-scale devices.

The FT magic state preparation making use of the unflagged logical $\overline{H}$ gate only requires two additional qubits: one acting as measurement qubit, one as a flag for the measurement, cf. Fig. 14. However, three-qubit gates are necessary to measure the logical $\overline{H}$ operator. Additionally, the deterministic FT magic state preparation needs

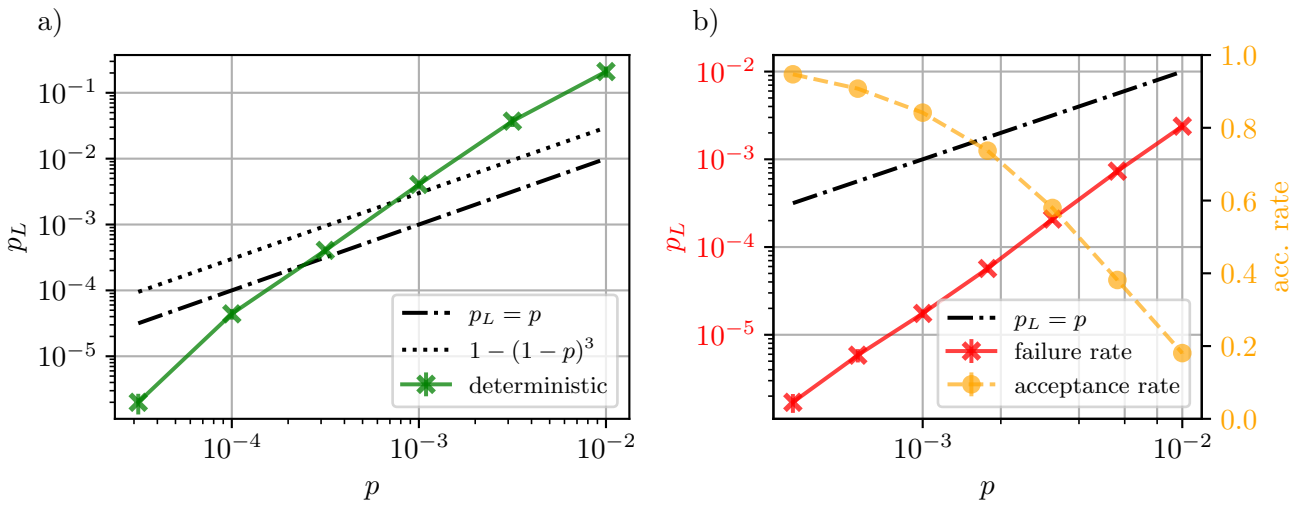

FIG. 12. a) Failure rate of deterministic FT magic state preparation (green). Breakeven points for outperforming physical qubits (black) with logical magic states lie at similar values as for FT Clifford gates. b) Repeat-until-success FT magic state preparation. While logical failure rates (red) are much lower than physical error rates for all p-values of interest, this advantage comes at the price of an exponentially decreasing acceptance rate (orange). Below $p = 10^{-3}$ acceptance rates are feasibly close to unity.

at most five repetitions. The non-deterministic protocol offers a tradeoff between high-quality magic states and the cost of post-selection.

Next, we put the results of our circuit-level simulations into context. We find pseudothresholds of approximately $2 - 7 \cdot 10^{-4}$ for the gadget-FT and of approximately $2 - 6.5 \cdot 10^{-3}$ for the algorithmic FT simulations, cf. Tab. III. With respect to the benchmark, i.e. one round of noisy error correction, we find that performing a flagged gate during one such round, lowers the logical performance only modestly. Most notably, the performance of a $\overline{CX}$ gate in a memory experiment is only slightly worse than that of a bare error correction cycle. In the gadget-FT protocol, all gates show a similar logical error rate.

Compared to other pseudothresholds reported in the literature, our pseudothresholds are competitive. Ref. [72], e.g., reports a pseudothreshold of $p_{\text{th}} \approx 4.5 \cdot 10^{-5}$ for a Hadamard gate in a concatenated 105-qubit code. In Ref. [24], the authors report a logical Clifford gate pseudothreshold of $1.85 \cdot 10^{-4}$ per round for a logical randomized benchmarking experiment. The surgery-based protocols of Ref. [25] for a $[\![42, 6, 4]\!]$ multi-cycle qLDPC code have been simulated with a phenomenological noise model, under which they report a pseudothreshold of $p_{\text{th}}^{(-)} \approx 7 \cdot 10^{-3}$. It is reasonable to expect a worse performance under a circuit-level noise model. Ultimately, although not directly comparable because of the different noise models and figures of merit, our pseudothresholds are in similar orders of magnitude, underlining the near-term competitiveness of flag-based protocols.

Next, we ask whether we can expect a finite pseudothreshold scaling these protocols to larger distances. Using a fault-tolerant protocol, the logical error rate scales as $p_L = ap^{t+1} + \mathcal{O}(p^{t+2})$ for small physical error rates. Here, a is the prefactor that is determined by the number of uncorrectable faults in the lowest (i.e. $t+1$) order. In general, a depends on the code and the circuit considered. For our $d = 3$ code, the fit-parameter a (cf. Tab. II) therefore is an estimate for that number and is determined by how many $\mathcal{O}(p^2)$ faults lead to a logical failure. In the algorithmic fault-tolerance protocol, this number is substantially lower compared to the gadget fault-tolerance. This is because the final single-qubit measurements that act as a "noise-free" round of stabilizer-measurement are included in a correlated decoding. In particular, this enables more compact protocols: no branching is required (cf. flagged gadget protocol) to acquire confidence in the values of the stabilizers. The physical failure rate scales as $p_{\text{fail}} = 1-(1-p)^k \sim kp$ as $p \to 0$. For the LCS codes and assuming $k = d$, it follows

$$p_{\text{th}} = \left(\frac{k(d)}{a(d)}\right)^{\frac{1}{t+1}} = \left(\frac{d}{a(d)}\right)^{\frac{2}{d+1}}. \tag{14}$$

Whether the pseudothreshold is finite or goes to zero is determined by the functional dependence of the prefactor $a(d)$ on the distance d. If a scales at most exponentially in d, that is $a \sim d^\beta e^{\alpha d}$, we can see from Eq. (14) that p_{th} approaches a non-zero constant value in the limit $d \to \infty$. Additional "back-of-the-envelope" estimations are provided in appendix E.

VI. CONCLUSIONS AND OUTLOOK

High-rate qLDPC codes have received major attention in the QEC community recently because of their

favorable properties as fault-tolerant quantum memories. Yet, performing FT computation directly within the code space is a pending question for many of such codes.

In this work, we have given an analysis of how to realize the FT universal gate set Clifford+T in lift-connected surface codes. Focusing on near-term implementations, we put the $[\![15, 3, 3]\!]$ member of the LCS code family center-stage. We start from a general prescription on how to construct all unitary Clifford gates for *any* member of the code family that is based on the underlying lifted-product structure of the code definition. Then, we employ flag qubit constructions and FT magic state preparation to ensure fault tolerance of our universal gate set against $t = 1$ arbitrary Pauli fault. A rigorous analysis of the performance of our circuits under a general depolarizing noise model reveals competitive breakeven points: We find that, for a strict gadget-level fault tolerance definition, an advantage over physical gates can be expected around physical error rates of 10^{-4}. These values drastically improve to well above 10^{-3} when adopting the notion of algorithmic fault tolerance, which allows one to decode the full experiment including the final single-qubit data measurements.

The obvious way forward is to generalize the flag construction used for $d = 3$ Clifford gates to higher-distance round-robin gates. Given that generalizations for error-correction circuits have been devised in Ref. [58], we expect similar constructions to also work for logical gates.

Three-qubit-gates, involved in the FT magic state preparation, could be made natively available in state-of-the-art or near-term quantum processors, for instance, based on trapped ions or neutral atoms [9, 73–76]. Instead decomposing these gates into sequences of single- and two-qubit gates may lead to a decrease in logical fidelity or, in the worst case, increase circuit depth to prohibitive levels for some platforms. As an alternative to directly realize an FT T-gate, we conjecture that specialized flag-like constructions for a unitary T-gate circuit that catch both X- and Z-faults may be conceivable.

DATA AVAILABILITY

All data shown in this manuscript is available at https://doi.org/10.5281/zenodo.17574057 (Ref. [77]).

CODE AVAILABILITY

Software code used in this project is available from the corresponding authors upon reasonable request.

AUTHOR CONTRIBUTIONS

JO, JB and SH devised all gate constructions. JO and SH performed numerical simulations and wrote the manuscript with contributions from all authors. MM and SH supervised the project.

ACKNOWLEDGMENTS

We gratefully acknowledge funding from the German Federal Ministry of Research, Technology and Space (BMFTR) as part of the Research Program Quantum Systems, research project 13N17317 ("SQale"). We furthermore acknowledge support by the European Union's Horizon Europe research and innovation programme under Grant Agreement No. 101114305 ("MILLENION-SGA1" EU Project). This research is also part of the Munich Quantum Valley (K-8), which is supported by the Bavarian state government with funds from the Hightech Agenda Bayern Plus. We additionally acknowledge support by the BMFTR project MUNIQC-ATOMS (Grant No. 13N16070). MM and JO acknowledge support by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy "Cluster of Excellence Matter and Light for Quantum Computing (ML4Q) EXC 2004/1" 390534769. The authors gratefully acknowledge the computing time provided to them at the NHR Center NHR4CES at RWTH Aachen University (Project No. p0020074). This is funded by the Federal Ministry of Education and Research and the state governments participating on the basis of the resolutions of the GWK for national high performance computing at universities.

[1] J. Old, M. Rispler, and M. Müller, *Lift-connected surface codes*, Quantum Sci. Technol. **9**, 045012 (2024).

[2] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, *Topological quantum memory*, J. Math. Phys. **43**, 4452–4505 (2002).

[3] H. Bombin and M. A. Martin-Delgado, *Topological quantum error correction with optimal encoding rate*, Phys. Rev. A **73**, 062303 (2006).

[4] J.-P. Tillich and G. Zémor, *Quantum LDPC Codes With Positive Rate and Minimum Distance Proportional to the Square Root of the Blocklength*, IEEE Trans. Inf. Theory **60**, 1193–1202 (2013).

[5] D. Gottesman, *Fault-tolerant quantum computation with constant overhead*, Quantum Inf. Comput. **14**, 1338–1372 (2014).

[6] N. P. Breuckmann and J. N. Eberhardt, *Quantum Low-Density Parity-Check Codes*, PRX Quantum **2**, 040101 (2021).

[7] Q. Xu, et al., *Constant-overhead fault-tolerant quantum computation with reconfigurable atom arrays*, Nat. Phys.

20, 1084–1090 (2024).
[8] S. Bravyi, et al., *High-threshold and low-overhead fault-tolerant quantum memory*, Nature **627**, 778–782 (2024).
[9] L. Pecorari, S. Jandura, G. K. Brennen, and G. Pupillo, *High-rate quantum LDPC codes for long-range-connected neutral atom registers*, Nat. Commun. **16**, 1–9 (2025).
[10] D. Gottesman, *Stabilizer codes and quantum error correction* (1997), arXiv:quant-ph/9705052.
[11] A. O. Quintavalle, P. Webster, and M. Vasmer, *Partitioning qubits in hypergraph product codes to implement logical gates*, Quantum **7**, 1153 (2023).
[12] N. P. Breuckmann and S. Burton, *Fold-Transversal Clifford Gates for Quantum Codes*, Quantum **8**, 1372 (2024).
[13] J. N. Eberhardt and V. Steffan, *Logical operators and fold-transversal gates of bivariate bicycle codes* (2024), arXiv:2407.03973.
[14] H. Sayginel, S. Koutsioumpas, M. Webster, A. Rajput, and D. E. Browne, *Fault-Tolerant Logical Clifford Gates from Code Automorphisms*, PRX Quantum **6**, 030343 (2025).
[15] Z. Liang and Y.-A. Chen, *Self-dual bivariate bicycle codes with transversal Clifford gates* (2025), arXiv:2510.05211.
[16] T. A. Brun, Y.-C. Zheng, K.-C. Hsu, J. Job, and C.-Y. Lai, *Teleportation-based Fault-tolerant Quantum Computation in Multi-qubit Large Block Codes* (2015), arXiv:1504.03913.
[17] D. Horsman, A. G. Fowler, S. Devitt, and R. Van Meter, *Surface code quantum computing by lattice surgery*, New J. Phys. **14**, 123011 (2012).
[18] L. Z. Cohen, I. H. Kim, S. D. Bartlett, and B. J. Brown, *Low-overhead fault-tolerant quantum computing using long-range connectivity*, Sci. Adv. **8** (2022).
[19] D. J. Williamson and T. J. Yoder, *Low-overhead fault-tolerant quantum computation by gauging logical operators* (2024), arXiv:2410.02213.
[20] A. Cowtan, *SSIP: automated surgery with quantum LDPC codes* (2024), arXiv:2407.09423.
[21] B. Ide, M. G. Gowda, P. J. Nadkarni, and G. Dauphinais, *Fault-Tolerant Logical Measurements via Homological Measurement*, Phys. Rev. X **15**, 021088 (2025).
[22] A. Cross, Z. He, P. Rall, and T. Yoder, *Improved QLDPC surgery: Logical measurements and bridging codes* (2024), arXiv:2407.18393.
[23] Z. He, A. Cowtan, D. J. Williamson, and T. J. Yoder, *Extractors: QLDPC architectures for efficient Pauli-based computation* (2025), arXiv:2503.10390.
[24] A. J. Malcolm, et al., *Computing efficiently in QLDPC codes* (2025), arXiv:2502.07150.
[25] N. Baspin, L. Berent, and L. Z. Cohen, *Fast surgery for quantum LDPC codes* (2025), arXiv:2510.04521.
[26] Q. Xu, et al., *Batched high-rate logical operations for quantum LDPC codes* (2025), arXiv:2510.06159.
[27] R. Chao and B. W. Reichardt, *Quantum Error Correction with Only Two Extra Qubits*, Phys. Rev. Lett. **121**, 050502 (2018).
[28] T. J. Yoder, R. Takagi, and I. L. Chuang, *Universal Fault-Tolerant Gates on Concatenated Stabilizer Codes*, Phys. Rev. X **6**, 031039 (2016).
[29] B. Eastin and E. Knill, *Restrictions on Transversal Encoded Quantum Gate Sets*, Phys. Rev. Lett. **102**, 110502 (2009).
[30] D. Gottesman and I. L. Chuang, *Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations*, Nature **402**, 390–393 (1999).
[31] S. Bravyi and A. Kitaev, *Universal quantum computation with ideal Clifford gates and noisy ancillas*, Phys. Rev. A **71**, 022316 (2005).
[32] L. Egan, et al., *Fault-tolerant control of an error-corrected qubit*, Nature **598**, 281–286 (2021).
[33] C. Ryan-Anderson, et al., *Realization of Real-Time Fault-Tolerant Quantum Error Correction*, Phys. Rev. X **11**, 041058 (2021).
[34] L. Postler, et al., *Demonstration of fault-tolerant universal quantum gate operations*, Nature **605**, 675 (2022).
[35] S. Krinner, et al., *Realizing repeated quantum error correction in a distance-three surface code*, Nature **605**, 669 (2022).
[36] Y. Zhao, et al., *Realization of an Error-Correcting Surface Code with Superconducting Qubits*, Phys. Rev. Lett. **129**, 030501 (2022).
[37] G. Quantum AI et al., *Quantum error correction below the surface code threshold*, Nature **638**, 920 (2024).
[38] L. Postler, et al., *Demonstration of fault-tolerant Steane quantum error correction*, PRX Quantum **5**, 030326 (2024).
[39] C. Ryan-Anderson, et al., *Implementing fault-tolerant entangling gates on the five-qubit code and the color code* (2022), arXiv:2208.01863.
[40] N. Lacroix, et al., *Scaling and logic in the colour code on a superconducting quantum processor*, Nature **645**, 614–619 (2025).
[41] P. Sales Rodriguez, et al., *Experimental demonstration of logical magic state distillation*, Nature **645**, 620–625 (2025).
[42] L. Daguerre, R. Blume-Kohout, N. C. Brown, D. Hayes, and I. H. Kim, *Experimental Demonstration of High-Fidelity Logical Magic States from Code Switching*, Phys. Rev. X **15**, 041008 (2025).
[43] S. Dasu, et al., *Breaking even with magic: demonstration of a high-fidelity logical non-Clifford gate* (2025), arXiv:2506.14688.
[44] N. P. Breuckmann and J. N. Eberhardt, *Balanced product quantum codes*, IEEE Transactions on Information Theory **67**, 6653–6674 (2021).
[45] P. Panteleev and G. Kalachev, *Asymptotically good quantum and locally testable classical LDPC codes*, in *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing* (2022) pp. 375–388.
[46] R. Chao and B. W. Reichardt, *Fault-tolerant quantum computation with few qubits*, npj Quantum Information **4**, 42 (2018).
[47] M. A. Nielsen and I. L. Chuang, *Quantum computation and quantum information: 10th anniversary edition* (Cambridge University Press, 2010).
[48] N. Delfosse and A. Paetznick, *Spacetime codes of Clifford circuits* (2023), arXiv:2304.05943.
[49] P.-J. H. S. Derks, A. Townsend-Teague, A. G. Burchards, and J. Eisert, *Designing fault-tolerant circuits using detector error models*, Quantum **9**, 1905 (2025).
[50] H. Bombin, D. Litinski, N. Nickerson, F. Pastawski, and S. Roberts, *Unifying flavors of fault tolerance with the ZX calculus*, Quantum **8**, 1379 (2024).
[51] J. C. Magdalena de la Fuente, et al., XYZ *Ruby Code: Making a Case for a Three-Colored Graphical Calculus for Quantum Error Correction in Spacetime*, PRX Quantum **6**, 010360 (2025).

[52] T. J. Yoder and I. H. Kim, *The surface code with a twist*, Quantum **1**, 2 (2017).

[53] E. Knill, R. Laflamme, and W. H. Zurek, *Resilient quantum computation: error models and thresholds*, Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences **454**, 365–384 (1998).

[54] P. Aliferis, D. Gottesman, and J. Preskill, *Quantum accuracy threshold for concatenated distance-3 codes*, Quantum Info. Comput. **6**, 97–165 (2006).

[55] A. M. Steane, *Active Stabilization, Quantum Computation, and Quantum State Synthesis*, Phys. Rev. Lett. **78**, 2252–2255 (1997).

[56] P. Shor, *Fault-tolerant quantum computation*, in *Proceedings of 37th Conference on Foundations of Computer Science* (1996) pp. 56–65.

[57] D. P. DiVincenzo and P. Aliferis, *Effective Fault-Tolerant Quantum Computation with Slow Measurements*, Phys. Rev. Lett. **98**, 020501 (2007).

[58] C. Chamberland and M. E. Beverland, *Flag fault-tolerant error correction with arbitrary distance codes*, Quantum **2**, 53 (2018), 1708.02246v3.

[59] H. Zhou, et al., *Low-overhead transversal fault tolerance for universal quantum computation*, Nature **646**, 303–308 (2025).

[60] T. Tansuwannont and D. Leung, *Achieving Fault Tolerance on Capped Color Codes with Few Ancillas*, PRX Quantum **3**, 030322 (2022).

[61] T. Jochym-O'Connor and R. Laflamme, *Using Concatenated Quantum Codes for Universal Fault-Tolerant Quantum Gates*, Phys. Rev. Lett. **112**, 010505 (2014).

[62] F. Green, S. Homer, C. Moore, and C. Pollett, *Counting, fanout, and the complexity of quantum ACC* (2001), arXiv:quant-ph/0106017.

[63] S. Heußen, et al., *Strategies for a practical advantage of fault-tolerant circuit design in noisy trapped-ion quantum computers*, Phys. Rev. A **107**, 042422 (2023).

[64] C. Chamberland and A. W. Cross, *Fault-tolerant magic state preparation with flag qubits*, Quantum **3**, 143 (2019).

[65] H. Goto, *Minimizing resource overheads for fault-tolerant preparation of encoded states of the Steane code*, Scientific Reports **6**, 19578 (2016).

[66] C. Gidney, *Stim: A fast stabilizer circuit simulator*, Quantum **5**, 497 (2021).

[67] T. Peham, L. Schmid, L. Berent, M. Müller, and R. Wille, *Automated Synthesis of Fault-Tolerant State Preparation Circuits for Quantum Error-Correction Codes*, PRX Quantum **6**, 020330 (2025).

[68] L. Schmid, T. Peham, L. Berent, M. Müller, and R. Wille, *Deterministic Fault-Tolerant State Preparation for Near-Term Quantum Error Correction: Automatic Synthesis Using Boolean Satisfiability*, in *2025 Design, Automation & Test in Europe Conference (DATE)* (2025) pp. 1–7.

[69] R. Zen, et al., *Quantum Circuit Discovery for Fault-Tolerant Logical State Preparation with Reinforcement Learning*, Phys. Rev. X **15**, 041012 (2025).

[70] L. A. Beni, O. Higgott, and N. Shutty, *Tesseract: A search-based decoder for quantum error correction* (2025), arXiv:2503.10988.

[71] C. Ryan-Anderson, *PECOS: Performance estimator of codes on surfaces*, `https://github.com/PECOS-packages/PECOS` (2019).

[72] C. Chamberland, T. Jochym-O'Connor, and R. Laflamme, *Thresholds for Universal Concatenated Quantum Codes*, Phys. Rev. Lett. **117**, 010501 (2016).

[73] J. D. Arias Espinoza, K. Groenland, M. Mazzanti, K. Schoutens, and R. Gerritsma, *High-fidelity method for a single-step N-bit Toffoli gate in trapped ions*, Phys. Rev. A **103**, 052437 (2021).

[74] S. J. Evered, et al., *High-fidelity parallel entangling gates on a neutral-atom quantum computer*, Nature **622**, 268–272 (2023).

[75] K. Sahay, et al., *Fold-transversal surface code cultivation* (2025), arXiv:2509.05212.

[76] J. Old, S. Tasler, M. J. Hartmann, and M. Müller, *Fault-tolerant stabilizer masurements in surface codes with three-qubit gates* (2025), arXiv:2506.09029.

[77] J. Old and S. Heußen, *Data for Addressable FT Universal Quantum Gates for LCS Codes: v1.0.0* (2025).

[78] A. M. Meier, B. Eastin, and E. Knill, *Magic-state distillation with the four-qubit code* (2012), arXiv:1204.4221.

Appendix A: LCS codes

We investigate the $(\ell, L) = (1, 3)$-LCS codes with parameters $[\![n, k, d]\!] = [\![15, 3, 3]\!]$ with stabilizer generators

$$
\begin{aligned}
S_X^{(0)} &= X_0X_6X_7X_{12}\\
S_X^{(1)} &= X_1X_7X_8X_{13}\\
S_X^{(2)} &= X_2X_6X_8X_{14}\\
S_X^{(3)} &= X_3X_9X_{10}X_{12}X_{14}\\
S_X^{(4)} &= X_4X_{10}X_{11}X_{12}X_{13}\\
S_X^{(5)} &= X_5X_9X_{11}X_{13}X_{14}\\
S_Z^{(0)} &= Z_0Z_3Z_4Z_{12}\\
S_Z^{(1)} &= Z_1Z_4Z_5Z_{13}\\
S_Z^{(2)} &= Z_2Z_3Z_5Z_{14}\\
S_Z^{(3)} &= Z_6Z_9Z_{10}Z_{12}Z_{14}\\
S_Z^{(4)} &= Z_7Z_{10}Z_{11}Z_{12}Z_{13}\\
S_Z^{(5)} &= Z_8Z_9Z_{11}Z_{13}Z_{14}.
\end{aligned}
$$

The logical Pauli operators can be chosen as

$$
\begin{aligned}
\overline{X}_0 &= X_0X_{10}X_{12}\\
\overline{X}_1 &= X_1X_{11}X_{13}\\
\overline{X}_2 &= X_2X_9X_{14}\\
\overline{Z}_0 &= Z_0Z_{10}Z_{12}\\
\overline{Z}_1 &= Z_1Z_{11}Z_{13}\\
\overline{Z}_2 &= Z_2Z_9Z_{14}.
\end{aligned}
$$

For more details, we refer to Ref. [1].

Appendix B: Circuits

1. Logical $\overline{S}$ gate

In Fig. 13, we show the circuit for the flagged, targeted logical $\overline{S}_i$ gate. The stabilizers required to be measured are tabulated in Tab. IV.

2. Stabilizer operators measured

In Tab. IV, we tabulate the stabilizer operators measured in each logical gate to identify the incoming faults.

3. FT magic state preparation

In this section we discuss our adaptation of FT magic state preparation from Refs. [63–65] to the [[15, 3, 3]] LCS code. The scheme relies on repeatedly projecting a noisy logical state onto the eigensystem of a logical Hadamard operator [78]. By carefully choosing our physical circuits, we can ensure that under the general depolarizing noise model of Eq. (12) for all circuit components, no logical failure can occur in $\mathcal{O}(p)$, even though we make use of three-qubit gates in order to measure the logical Hadamard operator.

The FT magic state preparation scheme is depicted on the logical level in Fig. 14 a). The flagged measurement circuit, shown on the physical-qubit level in Fig. 14 b), is always used in the first round. It is also used in the second run if previously no physical measurement was flipped neither in M_H nor in the EC block. Otherwise an unflagged circuit can be used to measure the logical Hadamard operator. Only a single flag qubit is required to detect all dangerous faults, i.e., single faults that could potentially incur logical operators after performing EC. This simplification, compared to the 4-flag protocol given for the Steane code in Ref. [64], can be attributed to the fact that, here, we only measure a weight-3 operator in-

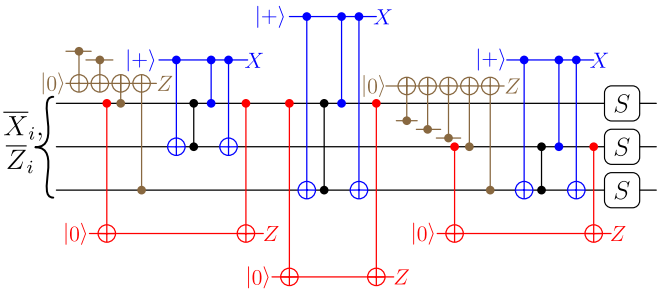

FIG. 13. Circuit for the flagged, targeted logical $\overline{S}_i$ gate. The unflagged gate contains ZCZ-gates. Therefore, one uses X- and Z-flags to flag correlated faults and measures Z-stabilizers for input faults.

TABLE IV. Stabilizer operators measured in gate protocols. Logical single-qubit gates require two, logical two-qubit gates four stabilizer measurements. The ordering corresponds to the ordering in the circuits given in Figs. 9,10,13. The maximum weight of any stabilizer operator measured is 6.

gate	logical	stabilizers
$\overline{S}$	0	$S_Z^{(0)}$, $S_Z^{(3)}$
	1	$S_Z^{(1)}$, $S_Z^{(4)}$
	2	$S_Z^{(2)}$, $S_Z^{(5)}$
$\overline{H}$	0	$\prod_{i=0}^{2} S_Z^{(i)} S_X^{(i)} = Y_0Y_1Y_2Y_{12}Y_{13}Y_{14}$, $S_X^{(3)}S_Z^{(3)}$
	1	$S_X^{(1)}S_Z^{(1)}$, $S_X^{(4)}S_Z^{(4)}$
	2	$\prod_{i=0}^{2} S_Z^{(i)} S_X^{(i)} = Y_0Y_1Y_2Y_{12}Y_{13}Y_{14}$, $S_X^{(5)}S_Z^{(5)}$
$\overline{CX}$	0, 1	$S_Z^{(0)}$, $S_Z^{(3)}$, $S_X^{(1)}$, $S_X^{(5)}$
	1, 0	$S_Z^{(1)}$, $S_Z^{(5)}$, $S_X^{(0)}$, $S_X^{(3)}$
	0, 2	$S_Z^{(0)}$, $S_Z^{(3)}$, $S_X^{(2)}$, $S_X^{(5)}$
	2, 0	$S_Z^{(2)}$, $S_Z^{(3)}$, $S_X^{(0)}$, $S_X^{(4)}$
	1, 2	$S_Z^{(1)}$, $S_Z^{(5)}$, $S_X^{(2)}$, $S_X^{(3)}$
	2, 1	$S_Z^{(2)}$, $S_Z^{(5)}$, $S_X^{(1)}$, $S_X^{(4)}$

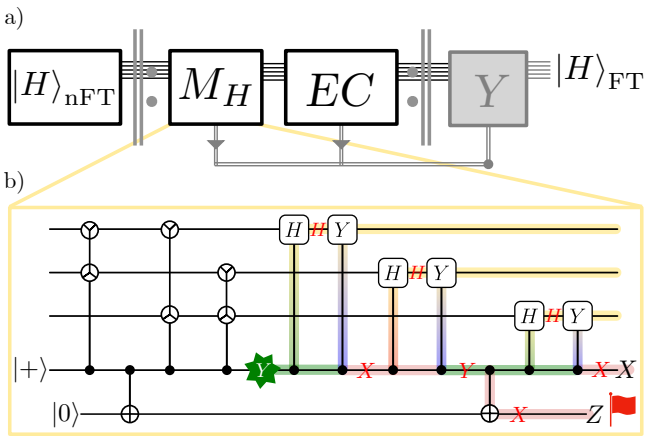

FIG. 14. Fault-tolerant magic state preparation circuit. (a) Logical blocks to execute the protocol either in a deterministic way or as a repeat-until-success scheme. Only one execution of the three blocks is required per trial of repeat-until-success. To deterministically prepare a single magic state fault-tolerantly, measurement of the logical Hadamard operator M_H and FT QEC need to be repeated until the measurement bit can be inferred unambiguously (see text). In case the FT measurement yields -1, a logical Y-flip is applied. (b) Circuit to measure a targeted logical Hadamard operator on three physical data qubits, on which logical operators $\overline{X}_i, \overline{Z}_i$ have their support. A single flag qubit catches otherwise undetected dangerous faults such as the one indicated by the green star. Such fault can manifest as either logical X or Z on the data qubits while leaving the measurement qubit unflipped.

stead of the weight-7 logical Hadamard operator of the Steane code. There, several combinations of high-weight Pauli error combinations must be rendered distinguishable by a suitable flag construction. An example of a dangerous fault and its propagation to all three relevant data qubits is sketched in Fig. 14 b).

Note that there are other high-weight faults in the circuit that will always be detected either by the measurement qubit or by the subsequent EC block. As an example, let us consider correlated weight-2 faults on the three-qubit controlled-YCY gates. There are two cases to distinguish:

1. The propagated fault after the third controlled-YCY gate is still weight-2. Then, it will not be detected by the measurement qubit since it has overlap 2 with the bitwise controlled-H and the bitwise controlled-Y gates. A single X/Z-fault before the target qubit of a controlled-Y (controlled-H) will (not) cause a Z-error on the control qubit. A single Y-fault before the target qubit of a controlled-H (controlled-Y) will (not) cause a Z-error on the control qubit. In any case, two Z-flips are incurred on the measurement qubit so that the measurement in the X-basis will not flip. However, the EC block will detect such an error of weight 2.

2. The propagated fault has become weight-3 after the third controlled-YCY gate. We argue that this is always a mix of X- and Z-operators and not both of them can be of weight 3. Therefore the propagated error can always be detected by the subsequent EC block. To see this, consider a fault ZIZ on the data qubits after the second controlled-YCY gate. This fault will be mapped to ZYZ (and a Z on the measurement qubit) by the subsequent controlled-YCY gate, which, despite being a logical Z, also contains a detectable X-component. A similar argument holds for mixed X- and Z-faults.

Now we discuss our measurement repetition conditions. In case two consecutive executions of M_H yield the same measurement results, we take said measurement result as the FT value if, additionally, either the syndrome measured in the EC block is trivial or we have performed four repetitions of M_H already. To see why the latter is necessary (compared to the typical $d = 3$ repetitions in FT operator measurements), consider the case where four measurements of $\overline{H}$ have resulted in the sequence $\{+1, +1, -1, -1\}$. This sequence could be caused by a weight-2 fault occurring in the second measurement round, which collapses onto a logical operator in EC and therefore flips the measurement result in the third round. Note that we assume that, for the above sequence, the third measurement collapses a state $\overline{P}|\overline{H}\rangle$ where P could be any X, Y or Z to $|-\overline{H}\rangle$. To distinguish this case from a simple measurement fault, the fourth repetition is necessary, which, here, will unambiguously reveal that indeed the orthogonal magic state $|-\overline{H}\rangle = \overline{Y}|\overline{H}\rangle$ is on the data qubits.

Appendix C: Additional Simulations and Simulation Details

1. Pseudothresholds

In Fig. 15 we show detailed data from which we obtain the pseudothresholds. We simulate around the expected crossing points and linearly interpolate the two closest points using the mean values, and a maximum and minimum estimate based on the error on the mean values. From these, we find a mean intersection, as well as an upper and a lower bound using root finding.

2. Non-averaged data

In Fig. 16, we show the raw data for the algorithmic fault-tolerance simulations before averaging. For all logical gates, if the input state is an eigenstate of the logical gate, it shows a lower logical error rate. This is a result of the primary logical errors induced by the circuits. If the all-to-all Pauli-controlled-Pauli gate (aPCP)

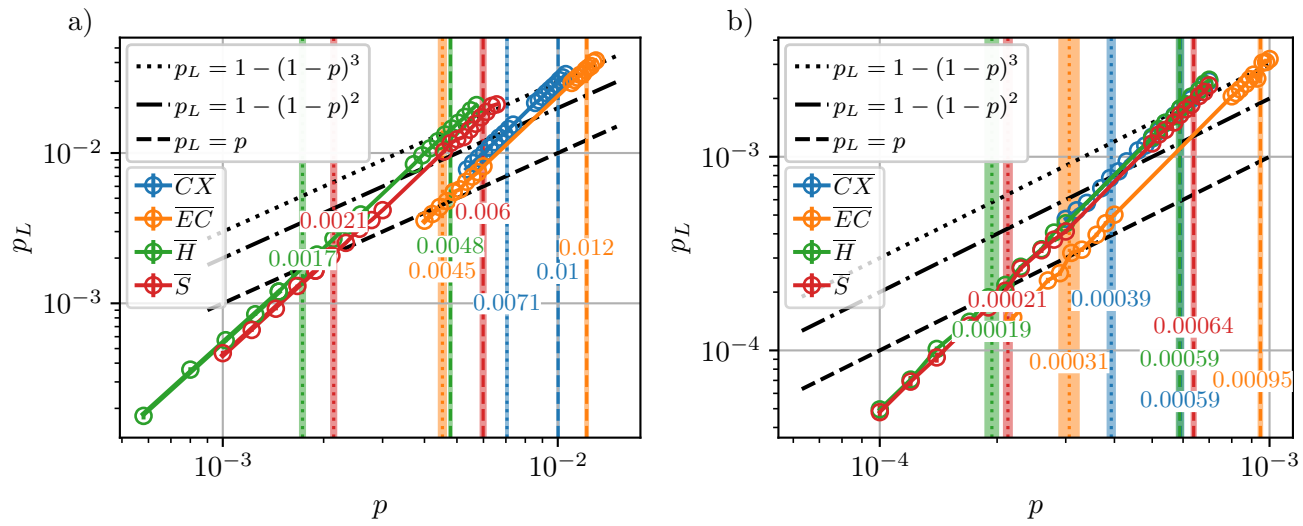

FIG. 15. Details on pseudothresholds. a) In algorithmic FT, the pseudothresholds are in the order 10^{-3}. b) In gadget FT, they are lower, $\sim 10^{-4}$.

part of the logical gate consists of Paulis P, then there is a bias towards logical P-errors because of the Pauli propagation. If an input state is an eigenstate of such a logical operator, it does not induce a logical error. For the $\overline{H}$ gate, shown in Fig. 16 a), these are Pauli-Y input-states because $H|+i/-i\rangle \propto |+i/-i\rangle$. For the $\overline{S}$ gate, shown in Fig. 16 b), these are Pauli-Z input-states because $S|0/1\rangle \propto |0/1\rangle$.

For the logical $\overline{CX}$, Fig. 16 c), the n logical errors are biased towards of X-type logical errors on the target and Z-type logical errors on the control. Here, this is because an X-type logical error on the control propagates to the target. If the input state on the control, however, is in the X-basis and the input state on the target is in the Z-basis, these errors don't induce a logical error. This results in the lower logical error rate for numerical experiments $\overline{C_0X_1}|{+}00\rangle$ and $\overline{C_0X_2}|{+}00\rangle$.

Fig. 17 shows the non-averaged data for the gadget fault-tolerance simulations. As they are independent of input logical states and the gates are symmetric on different (sets of) logical qubits, they show the expected same performance over all (combinations of) logical targets.

3. Logical operators measured in memory experiments

In Tab. V, we show the input state dependent logical operators measured in the memory experiments.

TABLE V. Logical operators measured perfectly to obtain logical failure rates of memory experiments. Depending on the input eigenstate, different logical operators are deterministic and can therefore be measured. For single qubit gates, these are the regular transformation of Paulis through conjugation, so e.g. if a logical qubit is initialized in $|\overline{+}\rangle$, i.e. an $\overline{X}$ eigenstate, the measurement after the $\overline{H}$ gate is $\overline{Z} = \overline{H}\ \overline{XH}$. For the logical two-qubit gate, if the control commutes with the input on the control, e.g. $\overline{Z}$ for a logical $\overline{CX}$, the logical measurement corresponds to the input eigenstate. For anti-commutation, e.g. $\overline{X}$ or $\overline{Z}$ for the logical $\overline{CX}$, we measure the joint $\overline{XX}$ and $\overline{ZZ}$ operators on the control and the target logical qubits.

gate	input eigenstate control	input eigenstate target	logical measurement control	logical measurement target
$\overline{S}$		$\overline{X}$ $\overline{Y}$ $\overline{Z}$		$\overline{Y}$ $\overline{X}$ $\overline{Z}$
$\overline{H}$		$\overline{X}$ $\overline{Y}$ $\overline{Z}$		$\overline{Z}$ $\overline{Y}$ $\overline{X}$
$\overline{CX}$	$\overline{Z}$ $\overline{X},\overline{Y}$	$\overline{X}$ $\overline{Y}$ $\overline{Z}$ $\overline{X}$ $\overline{Y}$ $\overline{Z}$	$\overline{Z}$	$\overline{X}$ $\overline{Y}$ $\overline{Z}$ $\overline{XX},\overline{ZZ}$

Appendix D: General Logical Operator Circuits

In this section, we provide the general, non-FT construction of circuits for logical gates LCS codes with $d > 3$. We show the circuits exemplarily for $d = 5$ in Fig. 18 a)-c). In general, for a code of distance d and logical $\overline{X}$ and $\overline{Z}$ operators supported on qubits $\mathcal{L}$, the single-qubit logical gate circuits are given by

$$\overline{H} = Y_{\mathcal{L}}^{\frac{d-1}{2} \bmod 2} H_{\mathcal{L}} Y C Y_{\mathcal{L}}, \tag{D1}$$

$$\overline{S} = S_{\mathcal{L}} Z C Z_{\mathcal{L}}. \tag{D2}$$

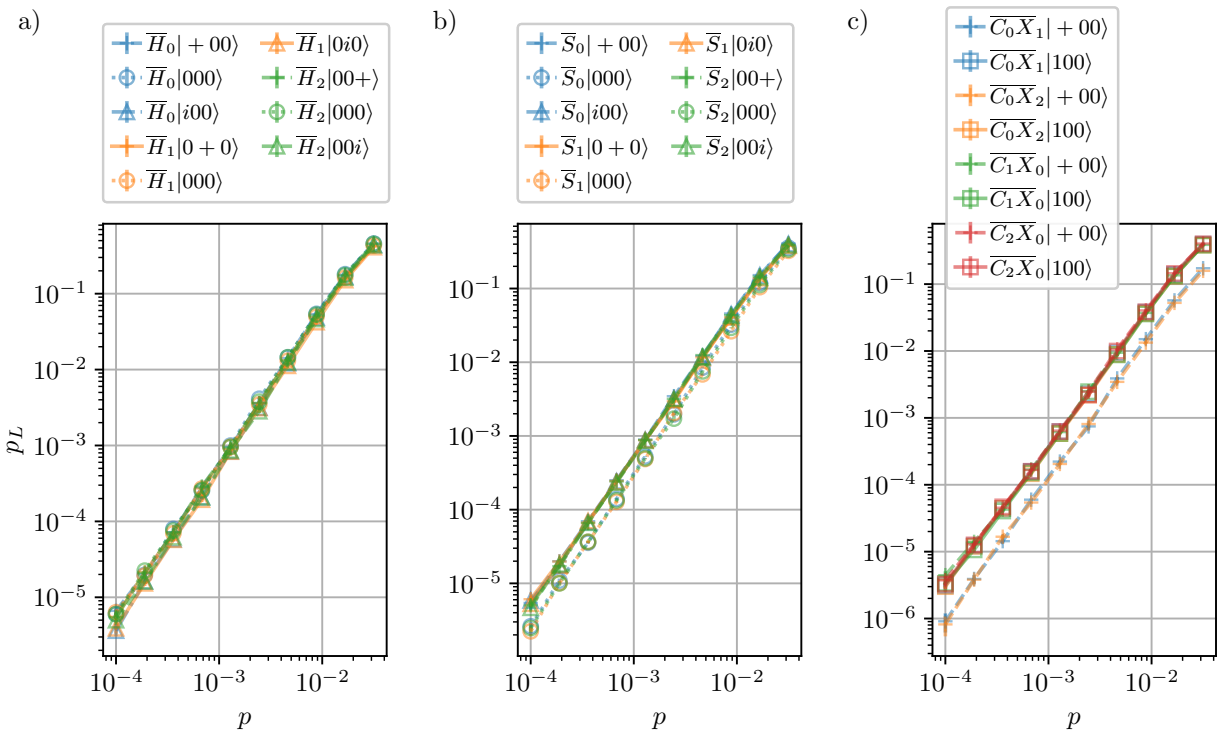

FIG. 16. Flag-FT gate circuits within the algorithmic fault-tolerance framework. For the logical $\overline{H}$ (a) and $\overline{S}$ (b) gates, we show individual logical error probabilities for all combinations of logical operators addressed and half of the considered input states. The corresponding Pauli operator flipped input states show the same logical error rates, e.g. $|-00\rangle$ has the same logical error rates as $|+00\rangle$. As explained in the text, Pauli Y-eigenstates perform best for the logical $\overline{H}$ and Pauli Z-eigenstates perform best for the logical $\overline{S}$. For the logical $\overline{CX}$ (c), we only show two representative input states. Here, if the experiment has Z-basis input states on the control and X-basis on the target, the logical error rate is lower as explained in the text.

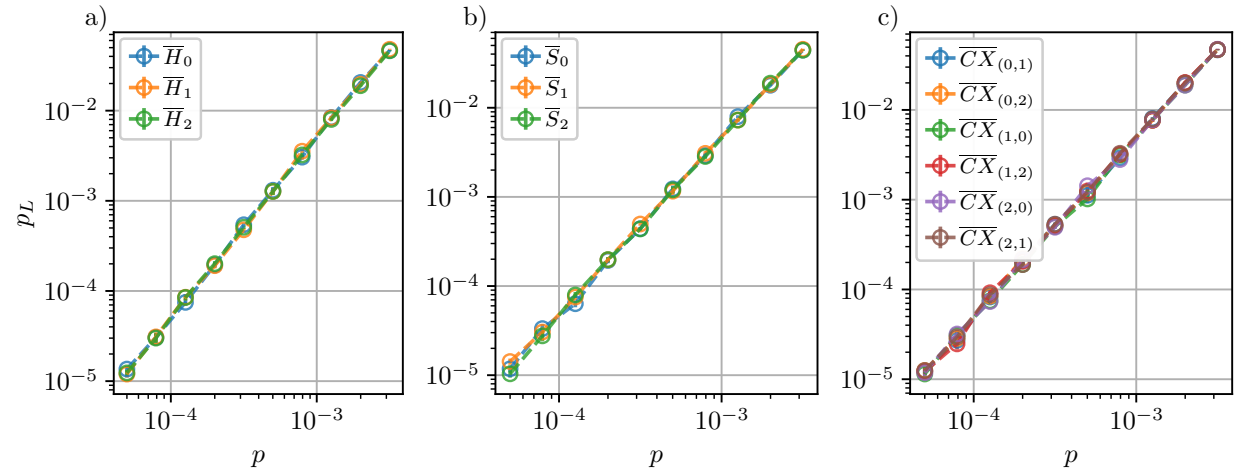

FIG. 17. Flag-FT gate gadgets within the conventional fault-tolerance protocol. This protocol is independent of the input state. We show the logical error probability for all combinations of logical operators addressed. All logical gates, $\overline{H}$ (a), $\overline{S}$ (b) and $\overline{CX}$ (c), have the same logical error rates within errorbars.

Here, we use the notation from the main text, i.e. a subscript $\mathcal{L}$ on a single-qubit gates refers to its transversal implementation on the set of qubits $\mathcal{L}$. For the two-qubit gates, this is the all-to-all Pauli-controlled-Pauli on qubits $\mathcal{L}$. A two-qubit logical $\overline{CX}$ from qubits supported on $\mathcal{L}_i$ and $\mathcal{L}_j$ is given by

$$\overline{C_iX_j} = \prod_{\substack{c\in\mathcal{L}_i,\\ t\in\mathcal{L}_j}} C_cX_t. \tag{D3}$$

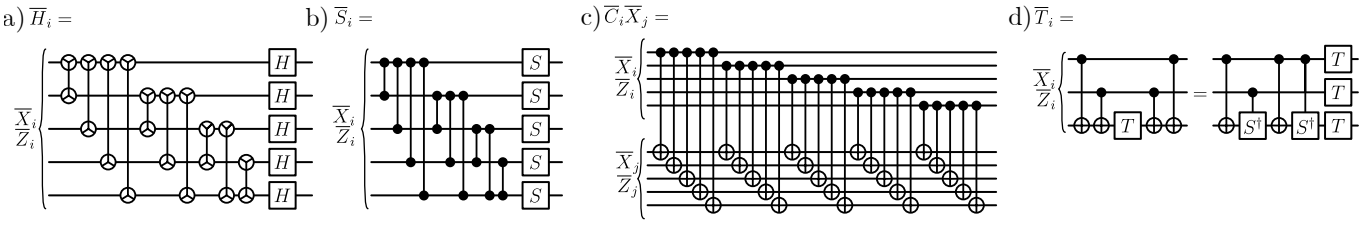

FIG. 18. a) – c) Non-FT circuits for logical gates in LCS codes of distance $d = 5$. For the logical $\overline{H}$ gate (a), the transversal part consists only of Hadamard gates. Distances $d = 3, 7, 11, \ldots$ require another layer of transversal Y-gates to fix a phase, as described in the main text for general all-to-all Pauli-controlled-Pauli gates. d) A logical T-gate, here for $d = 3$. The same decode-reencode approach can be used, and commuting the T-gate past the CX gates on the right results in a circuit with controlled-$S^\dagger$ gates.

In Fig. 18 d), we show a logical $\overline{T}$ gate on a distance $d = 3$ code. We used the same decode-reencode approach outlined in the main text. Commuting the T-gate through results in a circuit with controlled-$S^\dagger$ gates. It remains an open question whether/how a flag construction can render this circuit fault tolerant.

Appendix E: Scaling up round-robin flag-FT gates

From our discussion in Sec. V D and particularly our analysis of Eq. (14), it is clear that our round-robin flag-FT Clifford gate constructions will retain a finite pseudothreshold when scaling up to larger distances as long as we can assure that the leading-order coefficient of logical failure rates a scales at most exponentially with the code distance,

$$\text{Eq.(14)} \implies$$

$$p_{\text{th}} = e^{2\frac{\ln(d)}{d+1}} e^{-2\frac{\ln(a(d))}{d+1}} \tag{E1}$$

$$\xrightarrow{d\to\infty} \begin{cases} \text{const.} & \text{if} \quad a \sim d^\beta e^{\alpha d} \\ \sim e^{-d} \to 0 & \text{if} \quad a \sim d^\beta e^{\alpha d^\gamma}, \gamma > 1. \end{cases} \tag{E2}$$

We now give a (rather pessimistic) estimation on the functional behavior of a. Let us assume that the number of dangerous fault locations for an unflagged circuit scales like the number of physical gates, which is the worst case. For a round-robin construction, the number of gates scales like $d(d+1)/2 \sim d^2$. It is reasonable to expect the number of stabilizers that have to be measured as part of the FT gate to scale at most linearly in d because, in the worst case, we need to find one distinct stabilizer for each of the d physical qubits in the support of a logical operator. The same applies to the number of flag qubits required for such stabilizer measurements for arbitrary distance, as proposed in Ref. [58]. Note that each stabilizer's weight is bounded by 6 and independent of d since LCS codes are LDPC; this only contributes to the prefactor but not to the scaling behavior with distance.

The arbitrary-distance flag-FT protocol of Ref. [58] requires $\mathcal{O}(d^2)$ repetitions of syndrome measurements. With the number of fault locations scaling at most like d^2, there are at most $\mathcal{O}\left(\binom{d^2}{t+1}\right)$ total uncorrectable fault paths. Taking a Stirling-type approximation of this binomial coefficient, we conclude that the coefficient a of the leading order contribution to the logical failure rate may scale as badly as $\mathcal{O}\big((de)^d\big)$. From this estimation, we find a super-exponential scaling of $a(d)$ in the limit of large distances. This may indeed prevent a finite pseudothreshold in our scheme. The number of uncorrectable faults is at most allowed to scale linearly with d, instead of quadratic, for p_{th} to remain finite.

While this estimate may seem discouraging, we wish to point out two aspects of putting LCS codes to use. As this code family is not asymptotically good for large numbers of qubits and high distances anyway, we should not be deterred by this result on scaling up flagged gates. The regime of interest for practical implementations currently lies in the small- to medium-distance range. Recent results, such as Ref. [37], suggest that a scale-up to very large distances may not be required after all. Furthermore, we stress that our initial assumption that *all* fault locations lead to logical failure is overly pessimistic, as previous works, such as Ref. [58], indicate that flag constructions typically act to only make a small number of problematic faults distinguishable.