Calibrated Decoders for Experimental Quantum Error Correction

Edward H. Chen[1,*] Theodore J. Yoder[2,†] Youngseok Kim,[2] Neereja Sundaresan,[2] Srikanth Srinivasan[2]
Muyuan Li,[2] Antonio D. Córcoles,[2] Andrew W. Cross,[2] and Maika Takita[2]

[1]*IBM Quantum, Almaden Research Center, San Jose, California 95120, USA*
[2]*IBM Quantum, T.J. Watson Research Center, Yorktown Heights, New York 10598, USA*

Arbitrarily long quantum computations require quantum memories that can be repeatedly measured without being corrupted. Here, we preserve the state of a quantum memory, notably with the additional use of flagged error events. All error events were extracted using fast, midcircuit measurements and resets of the physical qubits. Among the error decoders we considered, we introduce a perfect matching decoder that was calibrated from measurements containing up to size-four correlated events. To compare the decoders, we used a *partial* postselection scheme shown to retain ten times more data than *full* postselection. We observed logical errors per round of $2.2 \pm 0.1 \times 10^{-2}$ (decoded *without* postselection) and $5.1 \pm 0.7 \times 10^{-4}$ (*full* postselection), which was less than the physical measurement error of 7×10^{-3} and therefore surpasses a pseudothreshold for repeated logical measurements.

Introduction.—Preparing and preserving logical quantum states is necessary for performing long quantum computations [1]. Because noise inevitably corrupts the underlying physical qubits, quantum error correction (QEC) codes have been designed to detect and recover from errors [2–6]. Significant efforts are currently focused on demonstrating capabilities that will be necessary for implementing practical QEC. An optimal choice of a code varies depending on the device and its noise properties [7]. Notable experimental implementations include NMR [8,9], ion traps [10–13], donors [14–16], quantum dots [17,18], and superconducting qubits [19–23]. Recent developments of high-fidelity midcircuit measurements and resets of superconducting qubits have enabled the preparation and repeated stabilization of logical states [24–26]; demonstrations of such quantum memories with enhanced lifetimes have been limited by, among other reasons, a combination of gate and measurement cross-talk.

One way to mitigate cross-talk [27] is to reduce the lattice connectivity [28,29]. Consequently, fault-tolerant operations require intermediary qubits; such qubits can be used to *flag* high-weight errors originating from low-weight errors [30,31]. In certain QEC codes and lattice geometries, flag qubits supply the information needed to extend the effective distance of a QEC code up to its intended distance, and thus enable maximal efficiency at detecting and correcting errors [32].

We demonstrated repeated error detection and correction of a $[\![4, 1, 2]\!]$ error-detecting topological stabilizer code on a heavy-hexagonal (HH) device designed to mitigate the limiting effects of cross-talk using flag qubits (Fig. 1). The combination of fast readout with reduced qubit connectivity improved, after postselecting on instances in which no errors were detected, logical errors per round when compared to the physical measurement error rate. A thorough analysis of this code led us to introduce a *partial* postselection scheme allowing us to discard ten times less data for comparing matching decoding algorithms. Compared against previously known decoding strategies on the entire dataset, we found that a decoder performed best with experimentally calibrated edge weights that account for the correlations between syndromes; we show that the computational cost of calibrating such a decoder scales linearly with the size of the syndrome extraction circuit for topological codes due to the local nature of the parity checks, and thus can be extended to larger distance codes even without any postselection. Furthermore, we showed that correlations between five or more syndromes can be eliminated by the application of a "deflagging" procedure. The minimal impact of deflagging on logical errors is an encouraging sign that this technique, and its extension to general flag-based codes, is a viable way to process flag outcomes in practice.

Theory of correlation analysis, decoding, deflagging, and postselection.—Active error correction involves decoding, using syndrome measurements, the errors that occurred in the circuit so that the proper corrections can be applied. We define error-sensitive events to be linear combinations of syndrome measurement bits that, in an ideal circuit, would be zero. Thus, a nonzero error-sensitive event indicates some error has occurred. For the HH code, there are two types of error-sensitive events defined as: (1) the difference of two subsequent measurements of the same stabilizer and (2) flag qubit measurements.

Error-sensitive events are depicted as nodes in a decoding graph with edges representing errors that are detected
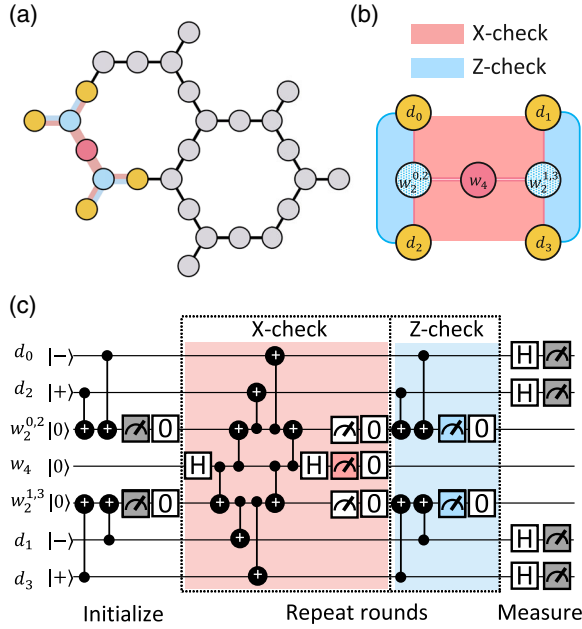
FIG. 1. (Color) (a) Experiments were performed on *ibmq_kolkata*, which had 27 qubits connected in a heavy hexagon (HH) topology. The seven qubits used for the [[4, 1, 2]] code are colored yellow, blue, and pink. (b) The code layout indicates a single weight-four, X stabilizer (pink), and two weight-two, Z stabilizers (blue) on the four data qubits (yellow) labeled d_i for integers "i" from 0 to 3. For the weight-two stabilizers, superscripts "0,2" ("1,3") indicate the data qubits on which they operate. The reduced connectivity of the graph is addressed by flag qubits (blue) alternating between (i) being used as weight-two stabilizers, and (ii) as intermediary qubits used to detect errors on the center, syndrome qubit (pink). (c) Circuit diagram for the code layout in (b) applied to an initial $|-\rangle_L$ logical state with repeated X- (pink), flag (white), and Z-check (blue) stabilizer measurements, together comprising a round, with midcircuit reset operations ("0") applied between rounds. In this illustration, the final measurement measures the four data qubits in the X basis due to the application of Hadamard gates.

by both events at their end points [Fig. 2(a) and [33]]. If the probability an edge occurs is P, then the edge is given weight $\log[(1 - P)/P]$. The decoding graph may also have a boundary node, so that an error detected by just one error-sensitive event can be represented as an edge from that event to the boundary node. In practice, there are also errors detected by more than two error-sensitive events that could be represented as *hyperedges* in a more general decoding *hypergraph*.

Given a set of nonzero error-sensitive events, minimum-weight perfect-matching (MWPM) finds paths of edges connecting pairs of those events with minimum total weight, and is a simple and effective decoding algorithm for a topological stabilizer code that only operates on a decoding *graph* [42], as opposed to a decoding *hypergraph*. While MWPM is computationally efficient, the analogous matching algorithm on a hypergraph is not, which limits the practicality of a decoding hypergraph.
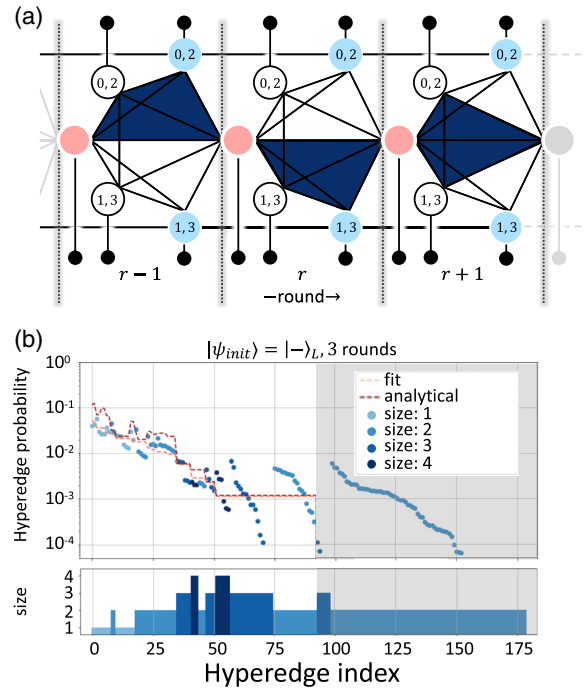

FIG. 2. (Color) (a) Decoder graph for the code layout depicted in Fig. 1(b). Syndrome measurements from weight-four (2) stabilizers are mapped to the pink (blue) nodes, and the weight-two flag measurements are mapped onto the white nodes. Identical to Fig. 1, "0,2" ("1,3") denotes the left- (right-) hand side of the code layout, with the colors of the circuits and syndrome measurement matching in both figures. For initial $|-/+\rangle_L$ states stabilized by the circuit in Fig. 1(c), there are three different possible size-four hyperedges within each round, each highlighted in dark blue across three consecutive rounds. The boundary nodes in black have, by definition, edges with weight "0" connecting them; rendering all boundary nodes to be effectively a single node for the purposes of the decoding process. (b) Applying the technique introduced in the main text to the experiment in Fig. 1(c), we estimated the correlation probabilities for all of the hyperedges shown in (a). The probabilities are sorted from largest to smallest based on the results from a least-squares fit using a six-parameter noise model. Points with darker colors represent hyperedges of greater sizes, as shown in the lower half of the plot. Hyperedges with indices greater than 93 (shaded gray) had no analytical expression, but were still experimentally adjusted to quantify the impact of computational leakage. The result of fitting the six-parameter noise model (fit, pink dash) agreed well with the analytical (red dash) curve generated using noise terms from simultaneous randomized benchmarking [33].

The effectiveness of MWPM depends crucially on edge weights in the decoding graph. We explored three strategies for setting these edge weights: (1) In the *uniform* approach, all edge weights were identical. (2) In the *analytical* approach, edge weights were individually calculated in terms of Pauli error rate parameters ρ_j, where the index j indicates one of the six errors being considered: CNOT gates, single-qubit gates, idle locations, initialization, resets, and measurements. The numerical values of the

parameters ρ_j can be chosen in several ways as discussed in the next section on *logical error measurements and decoding*. (3) In the *correlation* approach, we analyzed experimental data to determine a set of edge probabilities that are likely to have produced it. This approach involved first calculating the probabilities for all hyperedges in the decoding *hypergraph* before determining the edge probabilities used in the decoder *graph*. Importantly, this calibrated decoder graph, informed by the experimentally estimated hyperedge probabilities, can be used for any fault-tolerant protocol, and is not limited to the HH code or when postselection is employed.

The key idea of the estimation protocol is as follows: a hyperedge in the decoding hypergraph represents any of a number of Pauli faults in the circuit that are indistinguishable from one another because they each lead to the same set of nonzero error-sensitive events. If several faults occur together, the symmetric difference of their hyperedges is denoted S, the syndrome, or, in other words, the set of nonzero error-sensitive events that is observed. The probability that we observe a particular S is the probability that hyperedges occur in combination to produce S. Since this is related to the probability α_h of an individual hyperedge h occurring, we can learn α_h from many observations of S.

Realistically, the possible hyperedges are limited in size $|h|$ by locality of the circuits. In the $[\![4,1,2]\!]$ code, we found that hyperedges are limited to sizes four or less. Finding α_h in practical time begins by considering local clusters and then adjusting local estimates recursively from size-four hyperedges down to size one and two (Fig. 2 and [33]). Only size-one and -two edges are required for MWPM, but ignoring larger hyperedges can result in nonphysical, negative size-one correlations. Another way we explored decoding strategies was to consider analyzing only a subset of all data. By Pauli tracing, we classified edges in the decoding graph into three categories depending on whether its inclusion in the minimum-weight matching necessitated (1) flipping the logical measurement, (2) not flipping the logical measurement, or (3) is ambiguous [33]. The ambiguous case occurs specifically for error-detecting codes, like the $[\![4,1,2]\!]$ code presented here, because some errors result in the decoder having to choose between two equally probable corrections.

Using these classifications for edges in the decoder graph, we explored three degrees of postselection. The most conservative approach, using *full* postselection, involved discarding all results showing any nonzero error-sensitive event; this approach was the only one in which further decoding cannot be done. In the opposite regime, *without* postselection, all results were kept and any ambiguous edges in the MWPM were treated without flipping the logical measurement; here, logical error rates could have been improved by decoding but was not strictly needed. Finally, the intermediate regime involved a *partial*
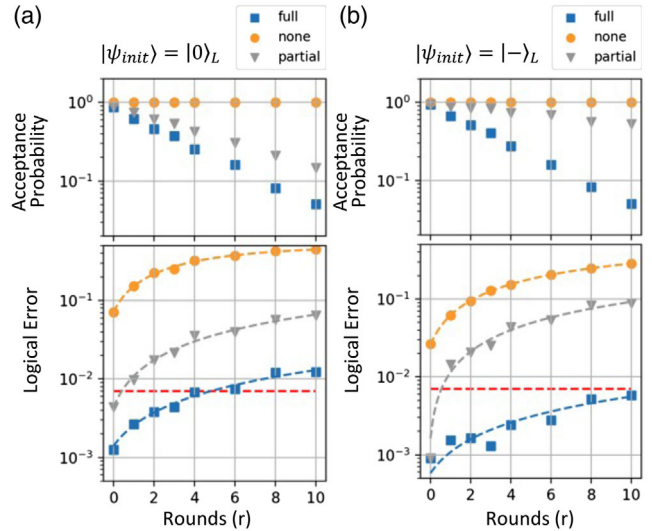

FIG. 3. (Color) (top) Fraction of total results used for the logical states [(a) $|0\rangle_L$, (b) $|-\rangle_L$] as the number of stabilizer rounds were repeated from 0 to 10 times when *full* (blue squares), *none* (yellow circles), or *partial* (gray triangles) postselection analysis was used. For *partial* postselection, the analytical decoder was used to exclude ambiguous shots. (bottom) The corresponding logical errors versus number of rounds. The dashed red lines indicate the pseudothreshold as determined by the best (average) physical measurement errors of 7×10^{-3} (7.7×10^{-3}).

postselection scheme whereby results were only discarded if the MWPM algorithm highlighted an ambiguous edge; here, decoding had to be done so that results with ambiguous edges that were highlighted could be discarded.

Logical error measurements and decoding.—Fitting the adjusted hyperedge probabilities to analytical expressions produces approximate estimates for the six-noise parameters in the error correcting experiments [Fig. 2(b)]. These noise estimates were found to be in good agreement with benchmarks based on simultaneous randomized benchmarking. Experiments were performed on four logical states ($|-/+\rangle_L$ and $|0/1\rangle_L$) each of which was stabilized up to ten rounds to extract a logical error per round of stabilizers (Fig. 3). This logical error varied depending on the analysis method.

For the *full* postselection scheme, the logical error for some rounds fell below the best and average physical initialization and measurement errors—a hallmark of being below a so-called pseudothreshold for fault-tolerant quantum computing. Fitting the decay curves resulted in inferred logical errors per round of $6.4 \pm 1.3 \times 10^{-4}$ for $|-/+\rangle_L$, and $11 \pm 1 \times 10^{-4}$ for $|0/1\rangle_L$.

If *none* of the instances of the experiment were discarded, then the logical error remained consistently above the pseudothreshold. In this analysis without any postselection and without decoding, we inferred logical errors per round of $40.4 \pm 0.2 \times 10^{-3}$ for $|-/+\rangle_L$, and $102 \pm 2 \times 10^{-3}$ for $|0/1\rangle_L$.
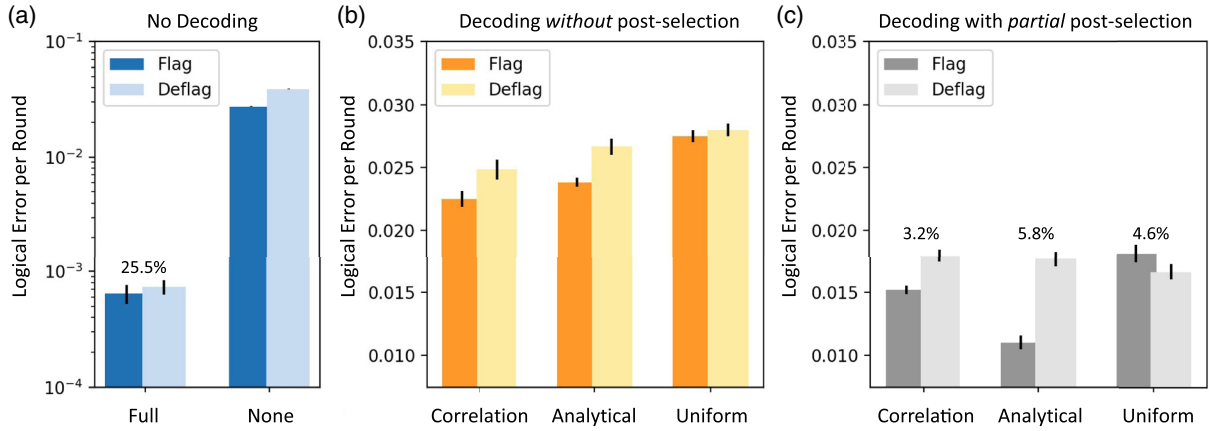
FIG. 4. (Color) Logical errors per round initially in $|-/+\rangle_L$ states under various analysis methods with acceptance probability per round labeled above. Results varied depending on whether the flag events were directly used for decoding (*flag*) or indirectly used for decoding using a *deflagging* procedure. (a) Logical error per round for *full* and *none* postselection methods. 25.5% of the counts were rejected with each round for the *full* postselection scheme. (b) Comparison between errors using three decoder graphs on data *without* postselection. (c) Comparison between errors using three decoder graphs on data with *partial* postselection. The approximate percentage of counts rejected for each stabilizer round are indicated above each bar.

Recalling that the $[[4, 1, 2]]$ is an error detecting code, we used the syndrome outcomes from each stabilizer round to perform a *postfacto* logical correction in software. Discarding instances where ambiguous edges were highlighted by the decoder allowed us to apply the *partial*, in contrast with the *full*, postselection scheme. With this scheme, significantly more instances of the experimental runs remained, resulting in inferred logical errors per round of $10.7 \pm 0.7 \times 10^{-3}$ for $|-/+\rangle_L$, and $6.2 \pm 0.3 \times 10^{-3}$ for $|0/1\rangle_L$.

Within the *none* and *partial* postselection schemes, we were able to compare the performance of three different instances of decoders (Fig. 4). The most generic decoder assumes there was no known noise model for the underlying physical system. Such a *uniform* decoder graph, in which every edge of the decoder graph was given equal weight, was expected to perform better than no decoding at all; but, was expected to be worse than any other graph whose edges were informed by some knowledge of the underlying noise. For instance, by selecting a simple, Pauli noise model, analytical expressions for the edge weights were calculated and led to improved logical error rates. Alternatively, if no assumptions were made about the noise, then edge weights were populated by the experimentally calibrated, correlation probabilities described earlier. We found that, as expected, such a correlation decoder graph indeed corrected for logical errors more effectively than the uniform decoding strategy and compared well with the analytical method [Fig. 4(b)]. However, when the *partial* postselection scheme was used, this trend no longer held since an analytical decoder with noise parameters from simultaneous randomized benchmarking outperformed the correlation analysis [Fig. 4(c)].

While the correlation analysis should, in principle, contain complete information about all of the noise in our experiments, its implementation is expected to become more computationally costly when applied to codes at larger distances. We simplified the decoder graph and thus the number and size of hyperedges needed in the correlation analysis, by feeding-forward information from each round of flag measurements. This procedure, known as "deflagging" [33], allowed us to eliminate all 30 of the size-four hyperedges in an experiment with ten rounds of stabilizer measurements without a significant increase in the logical error per round (Fig. 4). Furthermore, the logical errors were mostly preserved compared to results without the deflagging procedure.

Naïvely extending the HH code to distance-three would result in size-five hyperedges arising in the decoder hypergraph. However, when deflagging is applied, we found that there were no longer any size-five hyperedges, and the number of size-four hyperedges reduced from $148r - 12$ to $60r - 12$, where $r \geq 1$ is the number of rounds. Since the computational resources scale exponentially with the largest weight hyperedge in a graph, we expect that the deflagging procedure will provide a dramatic reduction in the computational resources needed to carry out the correlation analysis for codes beyond distance-three.

Conclusions.—Experimentally preparing and repeatedly stabilizing a logical quantum state, with error rates nearly ten times smaller than the lowest physical measurement error rate, is an important step towards executing larger, fault-tolerant circuits. The hexagonal lattice on which we demonstrated our findings can be extended to operate larger distance versions of the fault-tolerant HH code used here, for related codes [43–45], or even other codes using additional, bosonic degrees of freedom [46,47]. Although the distance-2 version was implemented on a subset of qubits within a hexagonal lattice, other topologies are also expected to benefit; for example, a heavy-square topology akin to the rotated surface code with added flag qubits [32]

or for distance-three demonstrations of the widely studied surface code [48,49]. Nevertheless, our probabilistic error correction methods and higher order error correlation analysis represents an approach for improving decoders for codes with or without flags within any device topology. We also demonstrated an effective use of flags to limit the extent of the correlations needed for efficient decoding. Our approach for extracting quantitative noise figures from the experiments creates a path to diagnose and reduce the logical errors per round of codes at larger distances.

As quantum computing devices become larger and less noisy, approaches such as ours may form the basis for efficiently decoding experimentally relevant errors. Other decoding strategies such as maximum-likelihood algorithms are known to scale unfavorably with code distances but may also benefit from our approach [50–52]. Eventually, decoders will need to be trained in real time [53], whereby logical operations could be interleaved with calibration circuits to periodically update the decoder graph's prior information with calibrated correlation probabilities. Previously studied bootstrapping techniques [26] coupled with the periodic recalibration of the correlation edges may eventually approach near-optimal decoding efficiencies, although the existence of an optimal strategy remains an open question.

[*]Cospokesperson.
ehchen@ibm.com
[†]Cospokesperson.
ted.yoder@ibm.com

[1] É. Gouzien and N. Sangouard, Factoring 2048-bit RSA Integers in 177 Days with 13 436 Qubits and a Multimode Memory, Phys. Rev. Lett. **127**, 140503 (2021).

[2] D. Aharonov and M. Ben-Or, Fault-tolerant quantum computation with constant error rate, SIAM J. Comput. **38**, 1207 (2008).

[3] E. Knill, R. Laflamme, and W. Zurek, Threshold accuracy for quantum computation, arXiv:quant-ph/9610011.

[4] D. Gottesman, Stabilizer Codes and Quantum Error Correction, Ph.D. Thesis, California Institute of Technology. Pasadena, California (2004).

[5] P. W. Shor, Scheme for reducing decoherence in quantum computer memory, Phys. Rev. A **52**, R2493 (1995).

[6] E. Knill and R. Laflamme, Theory of quantum error-correcting codes, Phys. Rev. A **55**, 900 (1997).

[7] P. Iyer and D. Poulin, A small quantum computer is needed to optimize fault-tolerant protocols, Quantum Sci. Technol. **3**, 030504 (2018).

[8] O. Moussa, J. Baugh, C. A. Ryan, and R. Laflamme, Demonstration of Sufficient Control for Two Rounds of Quantum Error Correction in a Solid State Ensemble Quantum Information Processor, Phys. Rev. Lett. **107**, 160501 (2011).

[9] J. Zhang, D. Gangloff, O. Moussa, and R. Laflamme, Experimental quantum error correction with high fidelity, Phys. Rev. A **84**, 034303 (2011).

[10] D. Nigg, M. Müller, E. A. Martinez, P. Schindler, M. Hennrich, T. Monz, M. A. Martin-Delgado, and R. Blatt, Quantum computations on a topologically encoded qubit, Science **345**, 302 (2014).

[11] N. M. Linke, M. Gutierrez, K. A. Landsman, C. Figgatt, S. Debnath, K. R. Brown, and C. Monroe, Fault-tolerant quantum error detection, Sci. Adv. **3**, e1701074 (2017).

[12] L. Egan, D. M. Debroy, C. Noel, A. Risinger, D. Zhu, D. Biswas, M. Newman, M. Li, K. R. Brown, M. Cetina, and C. Monroe, Fault-tolerant control of an error-corrected qubit, Nature (London) **598**, 1 (2021).

[13] C. Ryan-Anderson, J. G. Bohnet, K. Lee, D. Gresh, A. Hankin, J. P. Gaebler, D. Francois, A. Chernoguzov, D. Lucchetti, N. C. Brown, T. M. Gatterman, S. K. Halit, K. Gilmore, J. Gerber, B. Neyenhuis, D. Hayes, and R. P. Stutz, Realization of real-time fault-tolerant quantum error correction, Phys. Rev. X **11**, 041058 (2021).

[14] G. Waldherr, Y. Wang, S. Zaiser, M. Jamali, T. Schulte-Herbrüggen, H. Abe, T. Ohshima, J. Isoya, J. F. Du, P. Neumann, and J. Wrachtrup, Quantum error correction in a solid-state hybrid spin register, Nature (London) **506**, 204 (2014).

[15] M. H. Abobeih, Y. Wang, J. Randall, S. J. H. Loenen, C. E. Bradley, M. Markham, D. J. Twitchen, B. M. Terhal, and T. H. Taminiau, Fault-tolerant operation of a logical qubit in a diamond quantum processor, arXiv:2108.01646.

[16] C. D. Hill, M. Usman, and L. C. L. Hollenberg, An exchange-based surface-code quantum computer architecture in silicon, arXiv:2107.11981.

[17] X. Xue, M. Russ, N. Samkharadze, B. Undseth, A. Sammak, G. Scappucci, and L. M. K. Vandersypen, Computing with spin qubits at the surface code error threshold, arXiv:2107.00628.

[18] R. W. Andrews, C. Jones, M. D. Reed, A. M. Jones, S. D. Ha, M. P. Jura, J. Kerckhoff, M. Levendorf, S. Meenehan, S. T. Merkel, A. Smith, B. Sun, A. J. Weinstein, M. T. Rakher, T. D. Ladd, and M. G. Borselli, Quantifying error and leakage in an encoded Si/SiGe triple-dot qubit, Nat. Nanotechnol. **14**, 747 (2019).

[19] M. D. Reed, L. DiCarlo, S. E. Nigg, L. Sun, L. Frunzio, S. M. Girvin, and R. J. Schoelkopf, Realization of three-qubit quantum error correction with superconducting circuits, Nature (London) **482**, 382 (2012).

[20] J. Kelly *et al.*, State preservation by repetitive error detection in a superconducting quantum circuit, Nature (London) **519**, 66 (2015).

[21] A. D. Córcoles, E. Magesan, S. J. Srinivasan, A. W. Cross, M. Steffen, J. M. Gambetta, and J. M. Chow, Demonstration of a quantum error detection code using a square lattice of four superconducting qubits, Nat. Commun. **6**, 6979 (2015).

[22] D. Ristè, S. Poletto, M.-Z. Huang, A. Bruno, V. Vesterinen, O.-P. Saira, and L. DiCarlo, Detecting bit-flip errors in a logical qubit using stabilizer measurements, Nat. Commun. **6**, 6983 (2015).

[23] M. Takita, A. W. Cross, A. D. Córcoles, J. M. Chow, and J. M. Gambetta, Experimental Demonstration of Fault-Tolerant State Preparation with Superconducting Qubits, Phys. Rev. Lett. **119**, 180501 (2017).

[24] C. K. Andersen, A. Remm, S. Lazar, S. Krinner, N. Lacroix, G. J. Norris, M. Gabureac, C. Eichler, and A. Wallraff, Repeated quantum error detection in a surface code, Nat. Phys. **16**, 875 (2020).

[25] J. F. Marques, B. M. Varbanov, M. S. Moreira, H. Ali, N. Muthusubramanian, C. Zachariadis, F. Battistel, M. Beekman, N. Haider, W. Vlothuizen, A. Bruno, B. M. Terhal, and L. DiCarlo, Logical-qubit operations in an error-detecting surface code, Nat. Phys. **18**, 80 (2022).

[26] Google Quantum AI, Exponential suppression of bit or phase errors with cyclic error correction, Nature (London) **595**, 383 (2021).

[27] M. Takita, A. D. Córcoles, E. Magesan, B. Abdo, M. Brink, A. Cross, J. M. Chow, and J. M. Gambetta, Demonstration of Weight-Four Parity Measurements in the Surface Code Architecture, Phys. Rev. Lett. **117**, 210505 (2016).

[28] J. B. Hertzberg, E. J. Zhang, S. Rosenblatt, E. Magesan, J. A. Smolin, J.-B. Yau, V. P. Adiga, M. Sandberg, M. Brink, J. M. Chow, and J. S. Orcutt, Laser-annealing Josephson junctions for yielding scaled-up superconducting quantum processors, npj Quantum Inf. **7**, 129 (2021).

[29] IBM Quantum, High-fidelity superconducting quantum processors via laser-annealing of transmon qubits, arXiv: 2012.08475.

[30] R. Chao and B. W. Reichardt, Quantum Error Correction with Only Two Extra Qubits, Phys. Rev. Lett. **121**, 050502 (2018).

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

[32] C. Chamberland, G. Zhu, T. J. Yoder, J. B. Hertzberg, and A. W. Cross, Topological and Subsystem Codes on Low-Degree Graphs with Flag Qubits, Phys. Rev. X **10**, 011022 (2020).

[33] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevLett.128.110504 for (i) discussion of the Pauli tracer in Sec. D which does not include computational leakage, and results from simultaneous randomized benchmarking in Table S6; (ii) detailed derivations of the adjustment procedure in Sec. G; (iii) decoder graphs with classified edges labeled in color in Fig. S1; (iv) a detailed explanation of the deflagging procedure in Sec. F; and (v) further discussion of Refs. [23–26,28,32,34–41].

[34] S. Bravyi, S. Sheldon, A. Kandala, D. C. Mckay, and J. M. Gambetta, Mitigating measurement errors in multiqubit experiments, Phys. Rev. A **103**, 042605 (2021).

[35] IBM Quantum and Community, Qiskit: An open-source framework for quantum computing (2021), 10.5281/zenodo.2573505.

[36] S. Garion, N. Kanazawa, H. Landa, D. C. McKay, S. Sheldon, A. W. Cross, and C. J. Wood, Experimental implementation of non-clifford interleaved randomized benchmarking with a controlled-s gate, Phys. Rev. Research **3**, 013204 (2021).

[37] C. Gidney, Decorrelated depolarization (2020).

[38] R. Chao, M. E. Beverland, N. Delfosse, and J. Haah, Optimization of the surface code design for Majorana-based qubits, Quantum **4**, 352 (2020).

[39] A. Blais, R.-S. Huang, A. Wallraff, S. M. Girvin, and R. J. Schoelkopf, Cavity quantum electrodynamics for superconducting electrical circuits: An architecture for quantum computation, Phys. Rev. A **69**, 062320 (2004).

[40] A. D. Córcoles, M. Takita, K. Inoue, S. Lekuch, Z. K. Minev, J. M. Chow, and J. M. Gambetta, Exploiting Dynamic Quantum Circuits in a Quantum Algorithm with Superconducting Qubits, Phys. Rev. Lett. **127**, 100501 (2021).

[41] N. Sundaresan, I. Lauer, E. Pritchett, E. Magesan, P. Jurcevic, and J. M. Gambetta, Reducing unitary and spectator errors in cross resonance with optimized rotary echoes, PRX Quantum **1**, 020318 (2020).

[42] O. Higgott, PyMatching: A PYTHON package for decoding quantum codes with minimum-weight perfect matching, arXiv:2105.13082.

[43] M. B. Hastings and J. Haah, Dynamically generated logical qubits, Quantum **5**, 564 (2021).

[44] C. Gidney, M. Newman, A. Fowler, and M. Broughton, A fault-tolerant honeycomb memory, Quantum **5**, 605 (2021).

[45] J. R. Wootton, Hexagonal matching codes with 2-body measurements, arXiv:2109.13308.

[46] S. Puri, A. Grimm, P. Campagne-Ibarcq, A. Eickbusch, K. Noh, G. Roberts, L. Jiang, M. Mirrahimi, M. H. Devoret, and S. M. Girvin, Stabilized Cat in a Driven Nonlinear Cavity: A Fault-Tolerant Error Syndrome Detector, Phys. Rev. X **9**, 041009 (2019).

[47] B. M. Terhal, J. Conrad, and C. Vuillot, Towards scalable bosonic quantum error correction, Quantum Sci. Technol. **5**, 043001 (2020).

[48] S. Krinner, N. Lacroix, A. Remm, A. D. Paolo, E. Genois, C. Leroux, C. Hellings, S. Lazar, F. Swiadek, J. Herrmann, G. J. Norris, C. K. Andersen, M. Müller, A. Blais, C. Eichler, and A. Wallraff, Realizing repeated quantum error correction in a distance-three surface code, arXiv:2112.03708.

[49] Y. Zhao *et al.*, Realizing an error-correcting surface code with superconducting qubits, arXiv:2112.13505.

[50] S. Bravyi, M. Suchara, and A. Vargo, Efficient algorithms for maximum likelihood decoding in the surface code, Phys. Rev. A **90**, 032326 (2014).

[51] S. Bravyi and A. Cross, Doubled color codes, arXiv: 1509.03239.

[52] B. Heim, K. M. Svore, and M. B. Hastings, Optimal circuit-level decoding for surface codes, arXiv:1609.06373.

[53] P. Das, A. Locharla, and C. Jones, LILLIPUT: A lightweight low-latency lookup-table based decoder for near-term quantum error correction, arXiv:2108.06569.