Active readout-error mitigation

Rebecca Hicks, Bryce Kobrin, Christian W. Bauer, and Benjamin Nachman

Active Readout Error Mitigation

Rebecca Hicks[#],[1, *] Bryce Kobrin[#],[1, 2, †] Christian W. Bauer,[2, ‡] and Benjamin Nachman[2, §]

[1]*Physics Department, University of California, Berkeley, Berkeley, CA 94720, USA*
[2]*Physics Division, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA*
(Dated: December 17, 2021)

Mitigating errors is a significant challenge for near term quantum computers. One of the most important sources of errors is related to the readout of the quantum state into a classical bit stream. A variety of techniques have been proposed to mitigate these errors with post-hoc corrections. We propose a complementary scheme to actively reduce readout errors on a shot-by-shot basis by encoding single qubits, immediately prior to readout, into multi-qubit states. The computational resources of our technique are independent of the circuit depth and are compatible with the error rates and connectivity of many current devices. We analyze the potential of our approach using two types of error-correcting codes and, as a proof of principle, demonstrate an 80% improvement in readout error on the IBMQ Mumbai quantum computer.

I. INTRODUCTION

Quantum computers are promising tools to solve many computationally intractable scientific and industrial problems. However, existing noisy intermediate-scale quantum (NISQ) computers [1] suffer from significant errors that must be mitigated before obtaining useful results. These errors can be categorized as initialization, state preparation, or readout errors. Initialization errors, which often arise from thermal noise, lead to residual entropy in the starting state of the system. State preparation errors result from mis-calibrated quantum gates and unintended couplings with the environment. The focus of this paper is readout errors, which result from decoherence during measurement and from the overlapping support between the measured physical quantities that correspond to the $|0\rangle$ and $|1\rangle$ states. Readout errors can be significant and even dominate the error budget for relatively shallow quantum circuits.

A number of strategies have recently been proposed for mitigating readout errors which rely on classical, post-processing techniques. A particularly common scheme involves measuring a response matrix R_{ij} = Pr(measure i|true state is j) and performing regularized matrix inversion on the output distribution [2–6]. However, this scheme requires characterizing, at worst, an exponentially large matrix, and inverting the response matrix may suffer from numerical instabilities [2]. More fundamentally, such passive methods can only reduce the bias from readout errors on average, rather than correcting individual errors on a shot-by-shot basis [7–11]. Thus, they cannot be applied to tasks where obtaining individual output states—not expectation values—is desired, including random circuit sampling [12, 13], prime

factorization [14], and high energy physics simulations [15].

A more powerful strategy for correcting individual errors, during but not limited to readout, is based on the celebrated framework of quantum error correction (QEC). The core idea behind QEC is to embed the logical state into non-local degrees of freedom, such that local errors that affect only a few degrees of freedom can be isolated and corrected [16–18]. To utilize this protection for quantum processing, operations on the logical state must also be encoded and, crucially, be able to withstand a low rate of physical errors; this is the hallmark of *fault-tolerant* quantum computation. While individual aspects of error correction / fault tolerance have been demonstrated on small-scale experiments [19–29], performing full fault-tolerant computation imposes stringent technical requirements which are infeasible for near-term devices [30, 31]. Moreover, unless all components of a computation are implemented fault-tolerantly, the encoding is likely to have little effect on or even worsen the computational
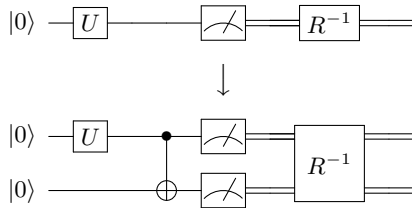

FIG. 1. A schematic diagram of the active readout error mitigation protocol for a single qubit and a quantum circuit represented by the unitary operation U. (Top) In passive readout error mitigation, the qubit is read out directly and classical post-processing (box with R^{-1}) is applied to reduce the bias in readout errors. (Bottom) In active error correction, the qubit is encoded before readout in a non-local state. The encoding circuit shown implements the 2-qubit repetition code, which enables error detection but not correction. If desired, passive readout error mitigation may be applied in combination with the active strategy, by post-processing the output of the encoded circuit (box with R^{-1}).

_____

* rebecca_hicks@berkeley.edu
† bkobrin@berkeley.edu
‡ cwbauer@lbl.gov
§ bpnachman@lbl.gov
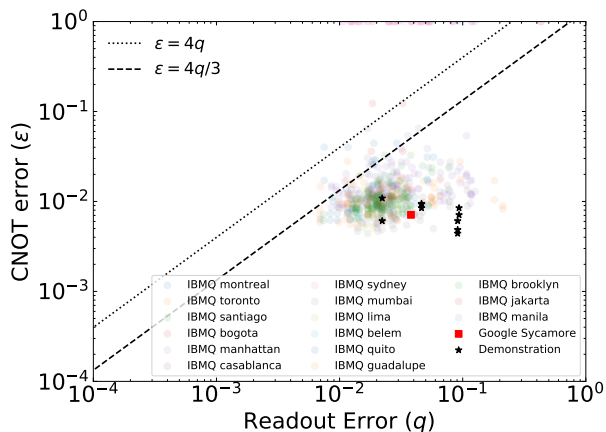[#] These authors contributed equally.

FIG. 2. A scatter plot of the CNOT error rates (ϵ) versus the readout error rates (q) across many IBMQ quantum computers. The dotted line indicates the threshold below which active readout error mitigation is effective when error *detection* is performed, while the dashed line indicates the analogous threshold for error *correction* with the 3-qubit repetition code (see Section III). The points labelled 'Demonstration' correspond to the five qubits on IBMQ Mumbai used in Sec. V. For the Sycamore device, we consider a decomposition of a CNOT gate into a CZ gate and two single-qubit Hadamard gates, whose individual error rates are estimated to be 0.41% and 0.15%, respectively [32].

outcome compared to a direct, unencoded implementation. Finding loopholes to this "all-or-nothing" outlook remains an essential task for bridging the gap towards full fault-tolerant computation.

In this paper, we propose and analyze an active error mitigation technique that targets readout errors only. The essence of our approach is to encode each qubit immediately before readout into a multi-qubit logical state, as exemplified in Fig. 1. A general feature of this strategy is a tradeoff between suppressing readout noise and introducing errors during the encoding circuit. Our strategy is thus most effective on devices whose intrinsic readout errors dominate over entangling gate errors, a situation that arises among many existing quantum computers (Fig. 2). Moreover, the resource requirement of the encoding in terms of ancilla qubits and gate overhead is independent of circuit depth, making it well suited for near-term applications. To this end, we implement our protocol on an IBMQ quantum computer and demonstrate almost an order of magnitude reduction in readout errors.

The rest of the paper is organized as follows. In Section II, we introduce the active readout error mitigation protocol after briefly reviewing existing forms of readout error mitigation. In Section III and IV, we present analytical results and numerical simulations based on simple error models. In Section V, we demonstrate the experimental performance of our strategy on a superconducting quantum computer. In Section VI, we conclude with a brief summary and outlook on error mitigation strategies.

II. METHOD

Consider a quantum system which is characterized in the computational basis by the frequencies $t_i = \Pr(\text{true state in } i) = \text{Tr}\left[\rho \left|i\right\rangle\!\left\langle i\right|\right]$, where $i \in \mathbb{Z}_2^{n_{\text{qubits}}}$. The goal of quantum readout is to accurately and efficiently estimate t_i using a series of measurements on identically prepared quantum systems. This task becomes non-trivial in the presence of readout errors, since these lead to biases in the measured frequencies $m_i = \Pr(\text{measure } i)$ compared to the true frequencies t_i.

A widely used strategy to correct these biases is to characterize the response matrix R and estimate the true state frequencies as $\hat{t} = R^{-1}m$. As an illustrative example, consider a single qubit quantum state with $t_0 = 1 - t_1 = p$ and suppose that it undergoes symmetric readout noise with error probability q, i.e. the response matrix is given by $R_{10} = R_{01} = q$ and $R_{00} = R_{11} = 1 - q$. If a series of measurements is performed, the expected frequency in the 0 state is $\mathbb{E}[m_0] = \lambda$ and its variance is $\text{Var}[m_0] = \lambda(1 - \lambda)$ for $\lambda = p + q(1 - 2p)$. Performing matrix inversion allows one to correct the average bias. In particular, applying

$$\hat{t} = \begin{pmatrix} 1 - q & q \\ q & 1 - q \end{pmatrix}^{-1} \begin{pmatrix} m_0 \\ m_1 \end{pmatrix} \qquad (1)$$

leads to the correct expected value of $\hat{t}$, i.e. $\mathbb{E}[\hat{t}] = (p, 1 - p)$.

However, this passive approach towards error mitigation is limited in two fundamental ways. First, the *variance* of $\hat{t}$ typically remains larger than that of the noise-free case; indeed, in our simple example, the variance increases by an $\mathcal{O}(q)$ factor:

$$\begin{aligned}
\text{Var}[\hat{t}_0] &= \text{Var}[m_0] - 2q\text{Cov}[m_0, m_1] + \mathcal{O}(q^2) \\
&= \lambda(1 - \lambda)(1 + 2q) + \mathcal{O}(q^2) \qquad (2) \\
&= \text{Var}[t_0] + q\left[1 - 2p(1 - p)\right] + \mathcal{O}(q^2),
\end{aligned}$$

where we have used that $\text{Cov}[m_0, m_1] = -m_0 m_1$ for a multinomial distribution. Correspondingly, the number measurements required to estimate $\hat{t}_i$ to a desired precision, which is proportional to $\text{Var}[\hat{t}_i]$, also increases with the readout error rate.

Second, and more significantly, we have assumed that m_i can be estimated using a large number of measurements relative to the size of the vector space. This is generally the case when computing the expectations of quantum observables, where the relevant vector space is that of the observable and not the full Hilbert space. But there are many problems of interest where the goal is to sample a relatively sparse number of output states rather, than to estimate statistical properties of the output distributions; for such problems, passive correction methods are fundamentally inapplicable.

Active readout error mitigation overcomes both of these limitations by effectively reducing the readout error q on a shot-by-shot basis. The main idea is to take a circuit where the physical qubits are the logical qubits

and encode the qubits before they are measured into a bigger multiqubit array. This encoding is analogous to conventional strategies for quantum error correction, but with two important distinctions. First, in the case of readout error mitigation, one is only concerned with bit-flip errors in the computational basis since phase-flip errors do not affect the measured distribution. Second, the encoding is performed *after* state preparation rather than at the beginning of the circuit. These simplifications allow us to circumvent the significant space and gate overhead typically associated with full quantum error correction.

The simplest version of active readout error mitigation is based on the two-qubit repetition code. As depicted in Fig. 1, each qubit is entangled with a unique partner qubit using a single CNOT gate (though other fully entangling gates could also be employed). Without errors, the measured outcomes are either 00 or 11, whereas with a single bit-flip error, the measured outcomes become 01 or 10; single qubit readout errors can thus be detected but not corrected. A natural extension of this encoding is to introduce a second ancilla qubit and entangle it with the original qubit (Fig. 3). This forms a three-qubit repetition code and allows for the correction of single qubit readout errors — i.e. by taking the majority vote among the three qubit — or the detection of two-qubit readout errors. We henceforth refer to these encodings as the (2,1) and (3,1) codes, where the notation (n, k) indicates that n physical qubits are required to encode k logical qubits.

By design, these encodings offer substantial protection against readout errors; nevertheless, they remain susceptible to certain gate errors that occur during the encoding circuit. For example, the single CNOT gate in the two-qubit encoding may lead to a correlated bit-flip error on *both* of the qubits, resulting in a spurious measurement outcome despite error detection. In general, if the average two-qubit error rate is ϵ, one expects an effective readout error rate of $q_{\mathrm{eff}} \approx \alpha\epsilon$, where α is an order-one constant that depends on specific protocol (e.g. two-qubit vs. three-qubit, and error detection vs. correction), as well as the error model for the entangling gates. We confirm this scaling for a symmetric depolarizing noise model via analytical results in Sec. III and numerical simulations in Sec. IV. Active readout error mitigation is thus beneficial
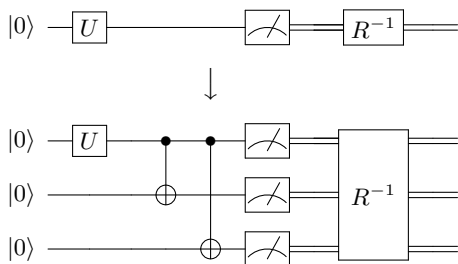
Encoding	(n,k)	Det. or cor.?	Eff. error rate q-dependence	α	Discarded measurements?
Repetition code	(2,1)	det.	$\sim q^2$	1/4	Yes
	(3,1)	det.	$\sim q^3$	1/4	Yes
	(3,1)	cor.	$\sim q^2$	3/4	No
Hamming code	(7,4)	det.	$\sim q^3$	1/4	Yes
	(7,4)	cor.	$\sim q^2$	7/8	No
	(8,4)	det.	$\sim q^4$	1/4	Yes
	(8,4)	hybrid	$\sim q^3$	1/4	Yes

TABLE I. Summary of the encoding schemes presented in this work. The notation (n, k) indicates that n physical qubits are required to encode k logical qubits. The effective error rate Q_{eff} [defined in Eq. (13)] scales non-linearly with the nominal readout error rate q, and is linearly proportional to the CNOT error rate ϵ with a susceptibility α given by $Q_{\mathrm{eff}}/k \approx \alpha\epsilon$.
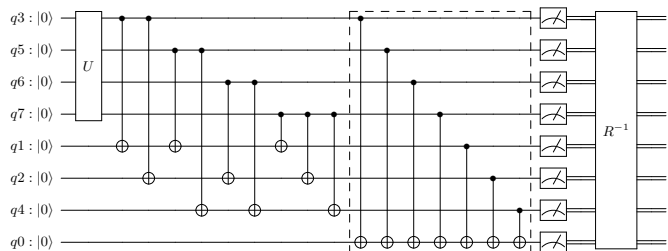

FIG. 4. Illustration of the encoding for the Hamming (8,4) code. The first four qubits (indexed q3, q5, q6, q7) contain logical state information while the remaining qubits are the parity bits. For the Hamming (7,4) code we omit the last parity bit (q0) and all gates connected to it, i.e. those within the black dashed box.

whenever the two-qubit error rate is lower than the intrinsic readout error rate q. In practice, this condition is met by many existing quantum devices, as depicted in Fig. 2 for Google Sycamore and a variety of IBMQ quantum computers.

Generalizing our strategy, one may consider encoding circuits for implementing arbitrary *classical* error correction codes. To do so, one would add ancilla qubits and entangle them with the original qubits to generate a classical code in the computational basis, i.e. each bitstring on the original qubits is mapped to an encoded bitstring on the full set of qubits. To understand the tradeoffs of using increasingly complex codes, we compare two families of error encoding schemes: the aforementioned repetition codes and two versions of the Hamming code—the (7,4) and (8,4) codes—illustrated in Fig. 4. We test the performance of these codes via numerical simulations in Sec. IV and summarize their key differences in Table I. In particular, we find that both types of codes offer comparable levels of error mitigation, and the more important factor for determining the effective error rate is whether error detection or error correction is performed.

FIG. 3. An illustration of the repetition (3,1) code, which enables both error detection and error correction.

III.   ANALYTICAL RESULTS

A.   Two-qubit repetition code

We begin by analyzing the simplest realization of active readout error mitigation: protecting a single qubit via the two-qubit encoding depicted in Fig. 1. Since each qubit is encoded separately, this represents an excellent model of a full quantum circuit, ignoring cross talk. Suppose that after applying U, the quantum state (for the first qubit) is

$$\rho = |\psi\rangle\langle\psi|, \quad |\psi\rangle = \sqrt{p}\,|0\rangle + \sqrt{1-p}\,|1\rangle, \qquad (3)$$

where, without loss of generality, we have chosen the phase to be real and positive. Moreover, as in the previous section, suppose that readout errors are described by a symmetric bit-flip channel with $\Pr(1 \to 0) = q$ and $\Pr(0 \to 1) = q$ for all qubits. Finally, we model the dominant error channel affecting the CNOT gate as a two-qubit depolarizing noise channel:

$$\rho \mapsto (1-\epsilon)\text{CNOT}\,\rho\,\text{CNOT} + \frac{\epsilon}{4}I_4, \qquad (4)$$

where I_4 is the identity matrix. Note that this last assumption, while widely used in theoretical analysis, is not essential to our qualitative results [33].

Based on this setup, we can compare the rate of readout errors with and without the encoding. In the nominal case with no encoding, the measurement frequencies are given by

$$\begin{aligned} m_0 &= p + (1-2p)q \\ m_1 &= (1-p) - (1-2p)q \end{aligned} \qquad (5)$$

where $m_i = \text{Tr}\,[\rho\,|i\rangle\langle i|]$. For the encoded circuit (bottom of Fig. 1), we instead have four output states with frequencies

$$\begin{aligned} m_{00} &\approx p(1-2q-\epsilon) + q^2 + \frac{\epsilon}{4} \\ m_{01} &\approx q - q^2 + \frac{\epsilon}{4} \\ m_{10} &\approx q - q^2 + \frac{\epsilon}{4} \\ m_{11} &\approx (1-p)(1-2q-\epsilon) + q^2 + \frac{\epsilon}{4}, \end{aligned} \qquad (6)$$

where we have neglected $\mathcal{O}(\epsilon q)$ terms and higher. In post-selection, we keep only the two symmetric states, $|\bar{0}\rangle = |00\rangle$ and $|\bar{1}\rangle = |11\rangle$, whose normalized frequencies are

$$\begin{aligned} m_{\bar{0}} &= \frac{m_{00}}{m_{00}+m_{11}} \approx p + \left(\frac{\epsilon}{4} + q^2\right)(1-2p) \\ m_{\bar{1}} &= \frac{m_{11}}{m_{00}+m_{11}} \approx (1-p) - \left(\frac{\epsilon}{4} + q^2\right)(1-2p). \end{aligned} \qquad (7)$$

Comparing to the nominal case (Eq. 5), the final distribution is characterized by an effective error rate,

$q_{\text{eff}} \approx \epsilon/4 + q^2$, which is independent of p. Thus, for $q^2 \ll \epsilon$, the encoding effectively lowers the readout error rate by a factor of $\sim 4q/\epsilon$.

Physically, this leading order dependence on ϵ may be understood through a simple error analysis. From a stochastic perspective, the depolarizing channel corresponds to introducing a random two-qubit Pauli error, each with probability $\epsilon/16$. There are 4 errors which could lead to a undetected measurement error, i.e. XX, XY, YX, and YY; thus, the total effective error rate is $\epsilon/4$. Additional mechanisms that result in an undetected error correspond to (*i*) two independent measurement errors or (*ii*) a CNOT error followed by a measurement error. However, these are subleading with probability $\mathcal{O}(q^2)$ and $\mathcal{O}(q\epsilon)$, respectively.

A few additional remarks are in order. First, as discussed in the previous section, it is important to consider not only the average bias in the measurment outcome but also the number of repetitions required to achieve a desired precision. For the unencoded circuit, the required repetitions is directly proportional to the variance of the measurement outcome distribution (Eq. 2). However, in the two-qubit encoding, a fraction of the repetitions are discarded (i.e. those in the 01 and 10 states), thereby contributing to the total repetitions but not to the output distribution. Thus, there is a competition between the improved bias offered by the encoding and the repetition overhead of the discarded states; whether or not this competition favors the encoding depends quantitatively on the desired observable, error model, and initial state.

Second, it is instructive to consider the case of non-symmetric readout errors. If we directly generalize our scheme to a non-symmetric error channel, $\Pr(1 \to 0) = q(1 + \kappa)$ and $\Pr(0 \to 1) = q(1 - \kappa)$, we find that the normalized probabilities, $m_{\bar{0}}$ and $m_{\bar{1}}$, exhibit a systematic bias compared to the true probabilities at linear order in $q|\kappa|$. Notably, this bias does not result from measurement errors that flip between $|00\rangle \leftrightarrow |11\rangle$, but rather from the fact that one of these states may be more likely to decay into the non-symmetric sector and therefore discarded during post-selection [34].

An *ad hoc* solution to eliminate this bias to perform error detection as before but, in a random half of the repetitions, apply an extra X gate immediately before the measurement. In doing so, the resulting distribution is *rebalanced*, or rendered symmetric with respect to the initial state [35]. The system can then be modeled as having an effective measurement rate, $\Pr(1 \to 0) = \Pr(0 \to 1) = 1/2[\Pr(1 \to 0) + \Pr(0 \to 1)] = q$, regardless of the asymmetry parameter α. Of course, this scheme assumes that the X gate itself introduces negligible errors compared to readout errors or two-qubit errors, which is reasonable assumption for many experimental platforms.

As we discuss next, an alternative approach to mitigate the effects of asymmetric noise—as well as eliminate the discarded measurements—is to perform error *correction* on the encoding qubit.

B. Three-qubit repetition code

By adding a third qubit to the encoded circuit, as shown in Fig. 3, one can realize the repetition (3,1) code whose output states without errors are 000 and 111. With this code, two bit-flip errors can be detected, while single bit-flip errors can be corrected by taking the "majority vote" among the three qubits. To understand the tradeoffs between error detection and detection, we analyze the three-qubit encoding circuit under the previous error model, i.e. symmetric readout errors $(\Pr(1 \to 0) = \Pr(0 \to 1) = q)$ and two-qubit depolarizing errors (at rate ϵ) following each CNOT gate.

We begin by computing the final measurement distribution up to order $\mathcal{O}(q\epsilon)$ and $\mathcal{O}(q^3)$:

$$
\begin{aligned}
m_{000} &\approx \left(1 - 3q + 3q^2 - \frac{7\epsilon}{4}\right) p + \frac{\epsilon}{4} \\
m_{001} &\approx \frac{\epsilon}{4} p + pq + (1 - 3p)q^2 \\
m_{010} &\approx \frac{\epsilon}{4}(2 - p) + pq + (1 - 3p)q^2 \\
m_{011} &\approx \frac{\epsilon}{4}(1 - p) + (1 - p)q - (2 - 3p)q^2 \\
m_{100} &\approx \frac{\epsilon}{4} p + pq + (1 - 3p)q^2 \\
m_{101} &\approx \frac{\epsilon}{4}(1 + p) + (1 - p)q - (2 - 3p)q^2 \\
m_{110} &\approx \frac{\epsilon}{4}(1 - p) + (1 - p)q - (2 - 3p)q^2 \\
m_{111} &\approx \left(1 - 3q + 3q^2 - \frac{7\epsilon}{4}\right)(1 - p) + \frac{\epsilon}{4}
\end{aligned}
\tag{8}
$$

For error detection, we proceed in analogy with the two-qubit code by computing the normalized frequencies in the logical code subspace:

$$
\begin{aligned}
m_{\bar{0}}^{\det} &= \frac{m_{000}}{m_{000} + m_{111}} \approx p + \frac{\epsilon}{4}(1 - 2p) \\
m_{\bar{1}}^{\det} &= \frac{m_{111}}{m_{000} + m_{111}} \approx (1 - p) - \frac{\epsilon}{4}(1 - 2p).
\end{aligned}
\tag{9}
$$

The effective error rate is thus $q_{\text{eff}}^{\det} = \epsilon/4 + \mathcal{O}(q^3)$ and exhibits the same leading order dependence on ϵ as the two-qubit code. This implies that the three-qubit encoding is subject to failure under the same number of individual gate errors, as we confirm by direct examination of the encoding circuit (see Appendix C). In contrast, the q^2 terms have vanished since three independent readout errors are required to produce an undetectable logical error.

We next simulate error correction by assigning the subspace of states with two or more 0's as $\left|\bar{0}\right\rangle$, and the remaining states as $\left|\bar{1}\right\rangle$. This results in a final output distribution

$$
\begin{aligned}
m_{\bar{0}}^{\text{cor}} &= m_{000} + m_{001} + m_{010} + m_{100} \\
&\approx p + 3\left(\frac{\epsilon}{4} + q^2\right)(1 - 2p) \\
m_{\bar{1}}^{\text{cor}} &= m_{111} + m_{110} + m_{101} + m_{110} \\
&\approx (1 - p) - 3\left(\frac{\epsilon}{4} + q^2\right)(1 - 2p),
\end{aligned}
\tag{10}
$$

characterized by an effective error rate, $q_{\text{eff}}^{\text{cor}} \approx 3(\epsilon/4 + q^2)$. Notably, the dependence on ϵ is enhanced by a factor of 3 compared to error detection compared with either the 2-qubit or 3-qubit code. This difference arises from the increased number of gate errors that can lead to a falsely *corrected* output state. For example, a single bit-flip on the first qubit after the first CNOT gate can *propagate* across the second CNOT gate into a two-qubit error, thereby corrupting the measurement outcome (see Appendix C).

Despite the less favorable effective error rate, we emphasize that error correction offers two main advantages compared to error detection, owing to the fact that none of the measurements are discarded during post-processing. First, the total number of events kept is larger, leading to reduced statistical uncertainties (see Appendix A). More importantly, contrary to error detection schemes, error correction is intrinsically robust against asymmetries in the readout error channels. For example, if the first qubit was subject to an asymmetric readout channel, $\Pr(1 \to 0) = q(1 + \kappa)$ and $\Pr(0 \to 1) = q(1 - \kappa)$ with $\kappa > 0$, then population would be transferred from $|000\rangle \to |100\rangle$ more frequently than from $|111\rangle \to |011\rangle$, leading to a bias in error detection schemes. But neither of these processes affect the sums, $m_{\bar{0}}$ and $m_{\bar{1}}$, at leading order in q, implying that this bias is eliminated in error correction.

IV. SIMULATIONS

A. Repetition codes

To demonstrate active error mitigation with realistic error rates, we perform numerical simulations on the encoded circuits shown in Fig. 1 and Fig. 3. As before, we consider two types of errors: (*i*) two-qubit depolarizing errors with rate ϵ applied after each CNOT gate (Eq. 4), and (*ii*) symmetric readout errors on each qubit with rate q. We implement the noisy circuits using the software package Cirq [36] and measure the outcome distribution directly as $m_i = \text{Tr}\left[\rho \left|i\right\rangle\!\left\langle i\right|\right]$.

For the two-qubit code, we perform error *detection* by discarding the probability of measuring the qubits in the odd parity subspace. For the three-qubit code, we perform error *correction* on the outcome distribution via majority vote (see Eq. 10). Subsequently, we characterize
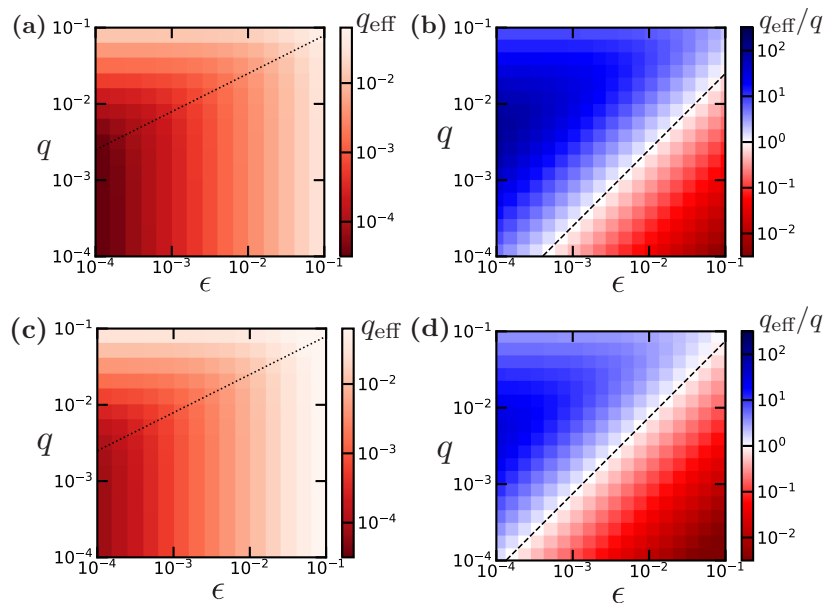
FIG. 5. Effective readout error, q_{eff}, as a function of the bare readout error rate, q, and the two-qubit error rate, ϵ. (a,b) Readout mitigation is performed via error detection with the two-qubit repetition code. (c,d) Analogous results for error correction with the three-qubit repetition code. In the left column, q_{eff} is plotted directly and exhibits a weak dependence on q for $\epsilon \gtrsim 16q^2$ (dotted line). In the right column, q_{eff} is normalized by the bare readout error rate, q. The dashed lines indicate the predicted thresholds for which error mitigation is advantageous, i.e. (b) $\epsilon = 4q$ for error detection, and (d) $\epsilon = \frac{4}{3}q$ for error correction. The results for q_{eff} are independent of the initial state parameter, p.

the effective error rate q_{eff} using

$$m_{\bar{0}} = p + (1 - 2p)q_{\text{eff}}$$
$$m_{\bar{1}} = (1 - p) - (1 - 2p)q_{\text{eff}} \tag{11}$$

where $m_{\bar{0}/\bar{1}}$ are the error-mitigated probabilities, and p is the initial state parameter defined in Eq. 3.

In Fig. 5, we depict the effective error rate for the two encodings as a function of ϵ and q. In both cases, for $\epsilon \gg q^2$, q_{eff} scales linearly with ϵ and has a negliglible dependence on q. This is consistent with our leading order analysis and indicates that higher order effects arise primarily from two-qubit readout errors with probability $\mathcal{O}(q^2)$. For a quantitative comparison with the unencoded readout error rate, we next plot the ratio q_{eff}/q in Fig. 5(b). For low error rates, this ratio crosses unity at $\epsilon = 4q$ for error detection and $\epsilon = 4q/3$ for error correction, in ageement with our predictions; small deviations from these trends become noticeable as $\epsilon, q \rightarrow 1$.

Note that although the simulations were performed with a specific value of p, the effective error rate q_{eff} is in fact independent of the initial state. This is a consequence of using *symmetric* noise channels for the readout and gate error and would change for a state-dependent noise model, e.g. for readout errors where $\Pr(1 \rightarrow 0) \neq \Pr(0 \rightarrow 1)$.

B. Comparison with the Hamming code

We now compare the performance of the repetition codes to a more complex encoding circuit based on the

Hamming (7,4) code, which encodes 4 logical qubits into 7 physical qubits (Fig. 4). The Hamming code is thus more space-efficient encoding for 4 logical qubits than either repetition code; in fact, even larger Hamming codes reduce the ratio of physical to logical qubits further, quickly approaching an optimal ratio of one (see Appendix B). Note, however, that implementing a Hamming code generally requires each logical qubit to be entangled with several ancillary qubits, which requires SWAP gates on systems with limited connectivity.

Like the three-qubit repetition code, the 7-qubit Hamming code enables the correction of single-qubit errors and the detection of two-qubit errors. This similarity arises because both codes have a *code distance* of three, which refers to the minimum number of local bit-flips (or Hamming distance) that connects two logical states. Indeed, it is well known that a classical code with code distance d enables the detection of up to $d-1$ independent errors and the correction of up to $d/2$ errors.

A circuit for implementing the Hamming (7,4) code is shown in Fig. 4. The first four qubits contain the logical state, while the remaining three qubits are ancilla qubits required for the encoding. Without errors, the state of each ancilla qubit is equal to the parity of a certain set of logical qubits. In contrast, if a single bit-flip error occurs (on either the logical bits or the ancilla bits), one or more of these conditions will be violated. A convenient way to check for errors is by multiplying the output state s by
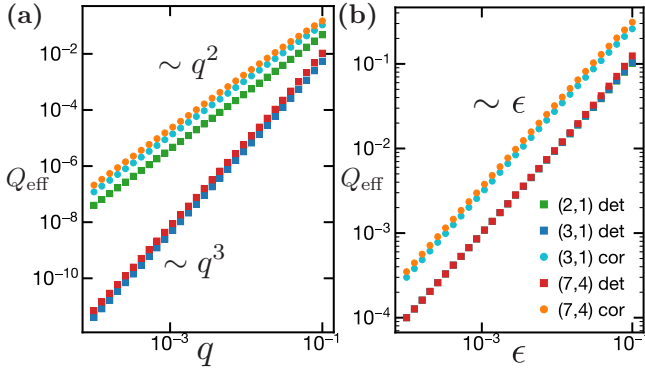
FIG. 6. Total readout error rate, Q_{eff}, for various error mitigation strategies. (a) Q_{eff} exhibits a power-law scaling as a function the bare readout error rate q, with an exponent determined by the code distance and whether error correction or error detection is performed. (b) Q_{eff} depends linearly on the CNOT gate error rate, ϵ. The linear prefactors α, defined by $Q_{\text{eff}}/k = \alpha\epsilon$, are summarized in Table I.

the parity check matrix:

$$H = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}_{[7,4]} , \tag{12}$$

where the ordering matches the physical layout of the first seven qubits in Fig. 4. Each element for which the bitwise product Hs is equal to 1 (modulo 2) indicates that an error has occurred on either the parity bit or the associated set of logical qubits. Based on these checks, one can uniquely determine the location of a *single* bit-flip error or detect the occurence of up to two bit-flip errors (see Appendix B).

To compare the performance of the Hamming (7,4) code to the repetition codes, we perform noisy simulations based each encoding and apply error either error correction or detection to the output distribution. We estimate the total error rate after error mitigation by computing

$$Q_{\text{eff}} = 1 - \frac{\sum_i R_{ii}}{2^k}, \tag{13}$$

where R_{ii} are the diagonal elements of the response matrix, i.e. the probabilities of measuring the correct state. This generalizes our previous error rate q_{eff} (Eq. 13); indeed, $Q_{\text{eff}} = q_{\text{eff}}$ in the case of a single qubit.

We begin by simulating the encoded circuits with readout errors only, i.e. taking $\epsilon = 0$. As shown in Fig. 6(a), the total error rate for both the Hamming (7,4) code and the repetition (3,1) code scales as $\sim q^2$ when using error correction and $\sim q^3$ when using error detection. These trends follow directly from having a code distance of $d = 3$, which implies that 2 or 3 independent bit-flip errors are required for a logical errors in the two respective cases. Similarly, the total error rate for the two-qubit repetition code scales as $\sim q^2$, since its code distance is $d = 2$.
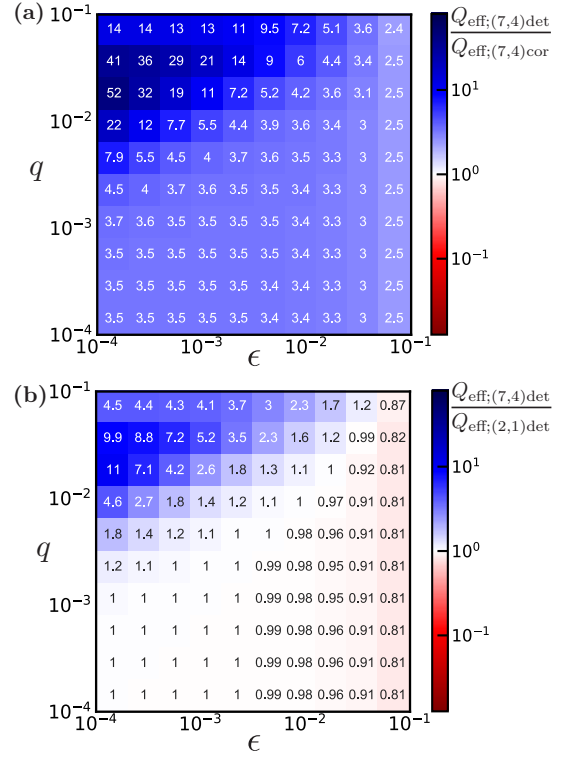

FIG. 7. Comparison of the effective error rate Q_{eff} between various active error mitigation strategies. (a) Ratio of Q_{eff} of the (7,4) code with error detection to that of the (7,4) code with error correction. (b) Analogous ratio of the (7,4) code with error detection to the (2,1) code with error detection.

We next isolate the effect of gate errors by plotting the error rate as a function of ϵ [Fig. 6(b)]. In contrast to the dependence on readout errors, all encoding schemes lead to the same general linear scaling, i.e. $Q_{\text{eff}} \approx 4\alpha\epsilon$ (we include a factor of 4 for the number of logical qubits). This indicates that a *single* gate error during the encoding can propagate into logical errors on the measurement outcomes, as we had shown previously for the repetition codes. More specifically, for all cases of error detection, we observe the same prefactor $\alpha = 1/4$; whereas, for error correction, α is an order-one factor larger. In fact, we previously derived $\alpha = 3/4$ for a single qubit encoded with the (3,1) code (see Eq. (10)). Similarly, we can determine $\alpha = 7/8$ for the Hamming (7,4) code by counting the relevant error channels (see Appendix C).

We now probe the effects of both readout and gate errors through two direct comparisons. First, we compare the 7-qubit encoding circuit using both error detection and error correction [Fig. 7(a)]. For the most part, their total error rates differ by a factor of ~ 3 due to their respective scalings with ϵ; this ratio increases further when $\epsilon \ll q$, owing to the favorable scaling with q offered by error detection. Second, we compare error detection using the 7-qubit encoding and the 2-qubit encoding [Fig. 7(a)]. Analogous with the previous case, the error rates are nearly equivalent for the two encodings except when $\epsilon \ll q$,
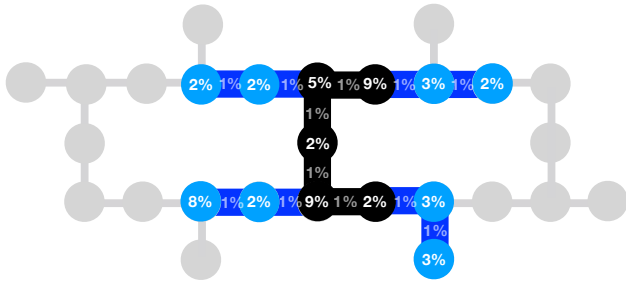
FIG. 8. A diagram of the IBMQ Mumbai computer layout, where circles represent qubits and links represent qubits that are connected. The qubits used for the measurement presented in Fig. 10 are colored in black (logical qubits) and blue (coding qubits). The readout errors are reported in the circles and the two qubit errors are reported on links connecting the circles. Error rates are reported from August 19, 2021.

in which case the scaling with q becomes relevant.

Finally, in Appendix D, we analyze an extended (8,4) version of the Hamming code which includes an extra ancilla qubit compared to the (7,4) code and has an increased code distance of $d = 4$. Interestingly, there are two natural circuits that lead to the same classical encoding and differ in the number of entangling gates. While one would naively expect the circuit with the fewest gates to have the lowest error rate, we find the opposite to be true. This highlights the importance of designing encoding circuits that are robust not only to readout errors, but also to gate errors that occur during the encoding circuit.

V. EXPERIMENTAL DEMONSTRATION

This section demonstrates the experimental performance of the repetition code on the IBMQ Mumbai quantum computer. This computer has 27 qubits total, arranged in a pattern depicted in Fig. 8. To demonstrate the active readout error correction protocol, we construct a 5 qubit sub-computer consisting of the five filled black circles in Fig. 8 (corresponding to qubits 12-16 in the computer's labeling scheme). Due to the adjacency map of connected qubits, we are unable to encode all qubits, without adding extra SWAP gates. Instead, the first (top right filled black circle in Fig. 8), second, fourth and fifth (counter-clockwise from the first) are encoded with the (3,1) repetition code to improve the readout errors.

In Fig. 9, the effective readout error rates for the four qubits are compared under three different scenarios. First, we measure the readout error rate with the encoding circuit but *without* performing error mitigation (i.e. we discard the measurements of the ancilla bits). As expected, this leads to an increase in the readout error rate relative to that of the nominal circuit. Indeed, the observed increase of $\sim 1\%$ is consistent with the independently

measured rate of depolarizing errors (Fig. 8).

Second, we measure the effective error rate after performing either error detection or correction. With either scheme, we observe a substantial improvement in the error rate, e.g. dropping by a factor of five in the first two qubits compared to the unencoded qubits. This indicates that the suppression of readout errors due to the encoding outweighs the errors introduced by the entangling gates and is consistent with the relatively large readout error rates for these qubits (Fig. 8).

A global picture of the subcomputer performance is illustrated in Fig. 10. Even though only four of the five qubits are encoded, the probability for a prepared state to be correctly measured increases from $\sim 75\%$ to more than 90% on average.

On other quantum computers, the number and location of qubits that may be encoded for readout mitigation will be determined by the device connectivity. This is a relatively minor concern for problems where only a small number of qubits are read out at a time (e.g. when measuring few-body correlation functions), and thus only a few additional qubits are required for the readout encoding. These ancilla qubits may be spare nearby qubits on the device or, for greater space efficiency, they may be qubits that are initially involved in the computation and repurposed directly before readout, i.e. by resetting to the ground state.

A more challenging situation occurs when measuring *all* the qubits simultaneously is required, e.g. as in random circuit sampling. Indeed, with current two-dimensional grid architectures, applying our scheme to all qubits is only well suited to computational tasks performed on a one-dimensional (or quasi-one-dimensional) subset of qubits, such that a neighboring row (or few rows) of qubits may be used for the readout encoding. Looking forward, we envision that improvements in hardware connectivity will overcome this limitation. For example, one may consider designing a two-dimensional device with two sublattices of qubits: one sublattice containing the computational qubits and the other sublattice containing ancilla qubits. The computational qubits would be connected to each other as in current devices but also to a single ancilla qubit in the opposite sublattice for implementing the (2,1) code. Notably, as the ancilla qubits would only involved in the readout operations, they would not affect the implementation or performance of the main computational task (assuming negligible cross-talk).

VI. CONCLUSIONS

In this work, we proposed a scheme for active readout error mitigation based on encoding the output state of a quantum circuit via a classical error correcting code. We showed that this approach generally provides significant readout improvement on devices whose bare readout error rate is comparable or larger than the error rate of entangling gates.
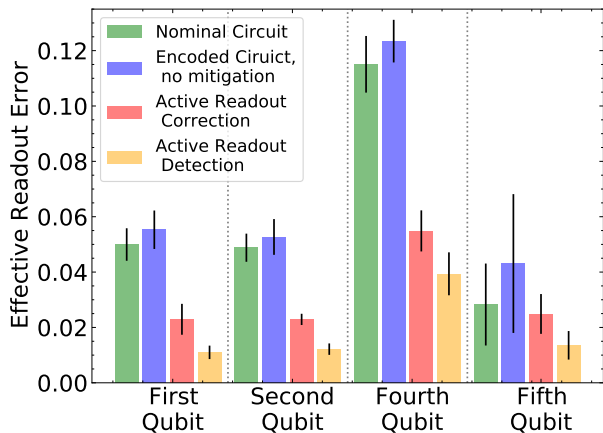
FIG. 9. The effective readout error for the first (top right filled black circle in Fig. 8), second, fourth and fifth (counterclockwise from the first) are encoded with the (3,1) repetition code to improve the readout errors. For each qubit, the left (green) bar corresponds to the nominal circuit without any additional CNOTs, the blue represents the circuit with the additional qubits and CNOTs without using them for correction, the red shows the results with the active readout error correction, and the orange shows the results with active readout error detection. The bar height is the average over all $2^5 = 32$ initial states and the error bar represents the standard deviation.

More specifically, we introduced two forms of encoding (the repetition code and the Hamming code) and analyzed the tradeoffs between error detection and error correction. Which scheme gives the optimal results depends on the criteria that are most important to satisfy. Error correction schemes are favorable if a large asymmetry in the readout error exists and it can not be mitigated through rebalancing techniques. On the other hand, error detection schemes have a smaller overall effective readout error. Among the error detection schemes, one should choose a specific encoding depending relative size of readout and CNOT gate errors, as well as how many ancillary qubits are available. If readout errors dominate over gate errors, then correction schemes with a code distance of 3, namely the (7,4) Hamming code and the (3,1) repetition code are preferable. On the other hand, if gate errors dominate, then all detection codes are more or less equal. Finally, the (7,4) Hamming code requires the least amount of ancillary qubits for a given number of physical qubits (however the encoding circuit involves non-local entangling gates).

With any of these implementations, active readout mitigation offers a few general advantages compared to passive error mitigation techniques. First, active error mitigation does not require characterizing a device's noise parameters (e.g. response matrix), which can involve an exponential overhead and be subject to temporal drifts; nor does it suffer from pathologies that can occur when applying post-processing (e.g. matrix inversion) on specific error models. Second, by removing errors on a shot-by-shot basis, our approach is more applicable to tasks where sam-

pling individual states is desired and, for general tasks, can be more effective at reducing not only the average measurement bias but also its variance. Crucially, active and passive strategies can also be combined to maximize the readout fidelity; i.e. by actively encoding the readout qubits and implementing error detection / correction, followed by post-processing the effective output distribution using passive correction techniques.

Lastly, in contrast to active *gate* error correction schemes, active readout error correction is fully compatible with near-term quantum hardware. Indeed, the protocol we have presented in this paper only requires local encoding whereby additional qubits need only have a small number of connections with entangling gates. Active readout error mitigation thus provides a practical intermediate step on the long path towards realizing full quantum error correction.

Note added: After this work had been completed, we became aware of a recent study [37] by Günther et. al. which also introduces active readout correction. Aside from the core concept, the two papers are complementary in their analysis and proposed implementations.

CODE AND DATA

The code and data for this paper can be found at https://github.com/LBNL-HEP-QIS/activereadouterrors.
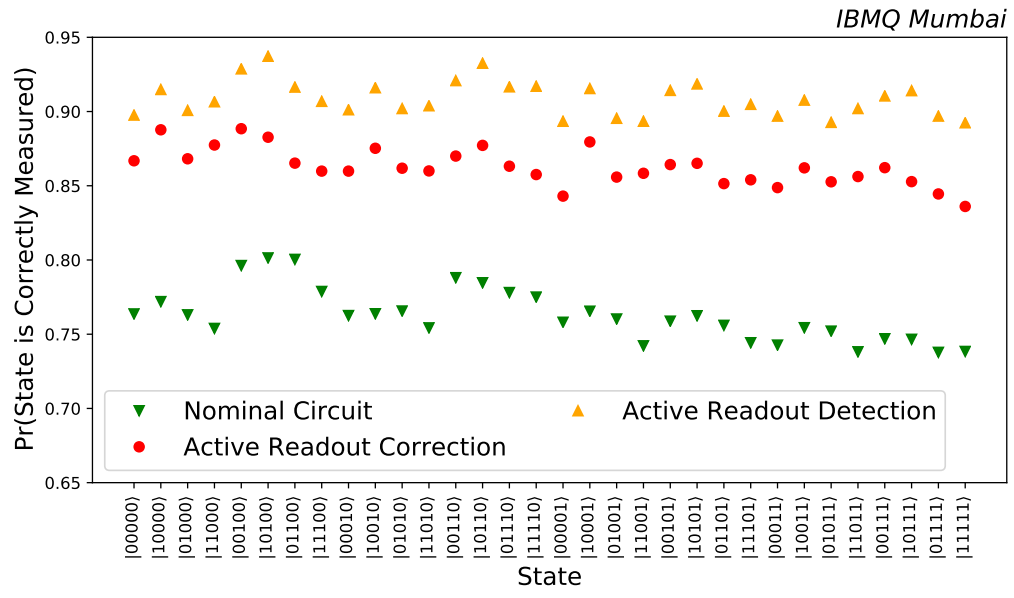
ACKNOWLEDGMENTS

FIG. 10. The diagonal elements of the response matrix measured on IBMQ Manhattan for an effective five logical qubit setup (see Fig. 8) with and without active readout mitigation. Due to the layout of IBMQ Manhattan, only two of the qubits have readout error correction and one has readout error detection. The other two qubits are not encoded.

[1] J. Preskill, Quantum Computing in the NISQ era and beyond, Quantum **2**, 79 (2018).

[2] M. R. Geller and M. Sun, Efficient correction of multiqubit measurement errors, arXiv:2001.09980 [quant-ph] (2020), arXiv: 2001.09980.

[3] C. Song, K. Xu, W. Liu, C.-p. Yang, S.-B. Zheng, H. Deng, Q. Xie, K. Huang, Q. Guo, L. Zhang, *et al.*, 10-qubit entanglement and parallel logic operations with a superconducting circuit, Physical review letters **119**, 180511 (2017).

[4] M. Gong, M.-C. Chen, Y. Zheng, S. Wang, C. Zha, H. Deng, Z. Yan, H. Rong, Y. Wu, S. Li, F. Chen, Y. Zhao, F. Liang, J. Lin, Y. Xu, C. Guo, L. Sun, A. D. Castellano, H. Wang, C. Peng, C.-Y. Lu, X. Zhu, and J.-W. Pan, Genuine 12-qubit entanglement on a superconducting quantum processor, Physical Review Letters **122**, 110501 (2019), arXiv: 1811.02292.

[5] K. X. Wei, I. Lauer, S. Srinivasan, N. Sundaresan, D. T. McClure, D. Toyli, D. C. McKay, J. M. Gambetta, and S. Sheldon, Verifying Multipartite Entangled GHZ States via Multiple Quantum Coherences, Physical Review A **101**, 032343 (2020), arXiv: 1905.05720.

[6] K. E. Hamilton, T. Kharazi, T. Morris, A. J. McCaskey, R. S. Bennink, and R. C. Pooser, Scalable quantum processor noise characterization (2020), arXiv:2006.01805 [quant-ph].

[7] W. d. J. C. B. B. Nachman, M. Urbanek, Unfolding Quantum Computer Readout Noise., npj Quantum Information **6**, 10.1038/s41534-020-00309-7 (2020), arXiv:1910.01969 [hep-ph].

[8] S. Y. Yanzhu Chen, Maziar Farahzad and T.-C. Wei, Detector tomography on ibm 5-qubit quantum computers and mitigation of imperfect measurement, arXiv (2019), arXiv:1904.11935 [quant-ph].

[9] Z. Z. Filip B. Maciejewski and M. Oszmaniec, Mitigation of readout noise in near-term quantum devices by classical post-processing based on detector tomography, arXiv (2019), arXiv:1907.08518 [quant-ph].

[10] F. Arute *et al.*, Quantum approximate optimization of non-planar graph problems on a planar superconducting processor (2020), arXiv:2004.04197 [quant-ph].

[11] S. S. Tannu and M. K. Qureshi, Mitigating measurement errors in quantum computers by exploiting state-dependent bias, in *Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture*, MICRO '52 (Association for Computing Machinery, New York, NY, USA, 2019) p. 279–290.

[12] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. Brandao, D. A. Buell, *et al.*, Quantum supremacy using a programmable superconducting processor, Nature **574**, 505 (2019).

[13] Y. Wu, W.-S. Bao, S. Cao, F. Chen, M.-C. Chen, X. Chen, T.-H. Chung, H. Deng, Y. Du, D. Fan, *et al.*, Strong quantum computational advantage using a superconducting quantum processor, arXiv preprint arXiv:2106.14734 (2021).

[14] P. W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, SIAM review **41**, 303 (1999).

[15] B. Nachman, D. Provasoli, W. A. de Jong, and C. W. Bauer, Quantum algorithm for high energy physics simulations, Physical Review Letters **126**, 062001 (2021).

[16] D. Gottesman, An introduction to quantum error correction and fault-tolerant quantum computation, (2009), arXiv:0904.2557 [quant-ph].

[17] B. M. Terhal, Quantum error correction for quantum memories, Reviews of Modern Physics **87**, 307 (2015).

[18] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition*, 10th ed. (Cambridge University Press, New York, NY, USA, 2011).

[19] M. Urbanek, B. Nachman, and W. de Jong, Quantum error detection improves accuracy of chemical calculations on a quantum computer, (2019), arXiv:1910.00129 [quant-ph].

[20] J. R. Wootton and D. Loss, Repetition code of 15 qubits, Phys. Rev. A **97**, 052313 (2018).

[21] R. Barends, J. Kelly, A. Megrant, A. Veitia, D. Sank, E. Jeffrey, T. C. White, J. Mutus, A. G. Fowler, B. Campbell, Y. Chen, Z. Chen, B. Chiaro, A. Dunsworth, C. Neill, P. O'Malley, P. Roushan, A. Vainsencher, J. Wenner, A. N. Korotkov, A. N. Cleland, and J. M. Martinis, Superconducting quantum circuits at the surface code threshold for fault tolerance, Nature **508**, 500 (2014).

[22] J. Kelly, R. Barends, A. G. Fowler, A. Megrant, E. Jeffrey, T. C. White, D. Sank, J. Y. Mutus, B. Campbell, Y. Chen, Z. Chen, B. Chiaro, A. Dunsworth, I.-C. Hoi, C. Neill, P. J. J. O'Malley, C. Quintana, P. Roushan, A. Vainsencher, J. Wenner, A. N. Cleland, and J. M. Martinis, State preservation by repetitive error detection in a superconducting quantum circuit, Nature **519**, 66 (2015).

[23] 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).

[24] 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).

[25] J. Roffe, D. Headley, N. Chancellor, D. Horsman, and V. Kendon, Protecting quantum memories using coherent parity check codes, Quantum Sci. Technol. **3**, 035010 (2018).

[26] C. Vuillot, Is error detection helpful on IBM 5Q chips?, Quantum Inf. Comput. **18**, 0949 (2018).

[27] D. Willsch, M. Willsch, F. Jin, H. De Raedt, and K. Michielsen, Testing quantum fault tolerance on small systems, Phys. Rev. A **98**, 052348 (2018).

[28] R. Harper and S. T. Flammia, Fault-tolerant logical gates in the IBM Quantum Experience, Phys. Rev. Lett. **122**, 080504 (2019).

[29] Z. Chen *et al.*, Exponential suppression of bit or phase flip errors with repetitive error correction (2021), arXiv:2102.06132 [quant-ph].

[30] E. T. Campbell, B. M. Terhal, and C. Vuillot, Roads towards fault-tolerant universal quantum computation, Nature **549**, 172 (2017).

[31] C. Gidney and M. Ekerå, How to factor 2048 bit rsa integers in 8 hours using 20 million noisy qubits, Quantum **5**, 433 (2021).

[32] F. Arute, K. Arya, R. Babbush, *et al.*, Quantum supremacy using a programmable superconducting processor, Nature **574**, 505–510 (2019).

[33] In particular, other two-qubit error models would simply modify the linear prefactor α (e.g. see Table I), which is determined by the relative likelihood of bit-flip errors occurring on *both* qubits. Furthermore, additional single qubit errors occuring *after* the CNOT gate would not contribute to q_{eff} at leading order, as two independent errors would be necessary to change the logical state.

[34] We note that in the highly asymmetric case, $\kappa \approx 1$, it may be advantageous to *correct* errors by assigning 01 and 10 to the logical $\hat{1}$ state, instead of discarding them. This reduces the measurement overhead associated with post-selection; however, it does not remove the $\mathcal{O}(\epsilon)$ error from two qubit flips in the encoding circuit.

[35] R. Hicks, C. W. Bauer, and B. Nachman, Readout rebalancing for near-term quantum computers, Physical Review A **103**, 10.1103/physreva.103.022407 (2021).

[36] Cirq, a python framework for creating, editing, and invoking noisy intermediate scale quantum (nisq) circuits, https://github.com/quantumlib/Cirq.

[37] J. M. Günther, F. Tacchino, J. R. Wootton, I. Tavernelli, and P. K. Barkoutsos, Improving readout in quantum simulations with repetition codes, arXiv preprint arXiv:2105.13377 (2021).

Appendix A: Readout variance with error mitigation

While we have mostly focused on the bias of quantum observables in the presence of readout errors, it is also important to consider the change in variance under active error mitigation, as this determines the number of measurements required to estimate an observable to a desired degree of precision. To this end, let us revisit the single qubit example presented in Section III B. We recall that, without any error mitigation, the distribution of measurements characterized by the expected frequencies $\mathbb{E}[m_0] = 1 - \mathbb{E}[m_1] = \lambda$ for $\lambda = p + q(1 - 2p)$ and the variance $\mathrm{Var}[m_0] = \mathrm{Var}[m_1] = \lambda(1 - \lambda)$. In practice, we care about the estimator of the frequencies after N measurements, e.g. $\theta_0 = N_0/N$ where N_0 is the number of measurements in the 0 state. The variance of the estimator is given by

$$
\begin{aligned}
\mathrm{Var}[\theta_0] &= \mathrm{Var}[m_0]/N \\
&= \frac{1}{N}\lambda(1 - \lambda) \\
&= \frac{1}{N}\left[p(1 - p) + q(1 - 2p)^2\right] + \mathcal{O}(q^2),
\end{aligned}
\tag{A1}
$$

which increases as a function of the error rate.

Let us now compare to case of active readout error *detection*. As we found in the main text, the bias in the distribution of the two logical states, $\bar{0}$ and $\bar{1}$, is renormalized by the effective error rate: $\mathbb{E}[m_{\bar{0}}] = 1 - \mathbb{E}[m_{\bar{1}}] = \lambda_{\mathrm{eff}}$ with $\lambda_{\mathrm{eff}} = p + q_{\mathrm{eff}}(1 - 2p) + \mathcal{O}(q^2)$. However, the estimator of these frequencies depends on two competing factors: the change in bias and the total number of measurements that are kept during post-selection. Specifically, we have $\theta_{\bar{0}} = N_{00}/(N_{00} + N_{11}) = N_{00}/(N(1 - 2q))$. This leads to a variance

$$
\begin{aligned}
\mathrm{Var}[\theta_{\bar{0}}] &\approx \frac{1}{N}\lambda_{\mathrm{eff}}(1 - \lambda_{\mathrm{eff}})(1 + 2q) \\
&\approx \frac{1}{N}\left[p(1 - p) + q_{\mathrm{eff}}(1 - 2p)^2 + 2qp(1 - p)\right],
\end{aligned}
\tag{A2}
$$

where we have dropped $\mathcal{O}(q^2, q_{\mathrm{eff}}^2)$ terms. Comparing with Eq. (A1), we find that whether or not error detection improves the estimator variance depends on both the state parameter p and the ratio q_{eff}/q.

In contrast, when error *correction* is performed, no measurements are discarded. Hence, the variance of the estimator for error correction depends only on q_{eff}:

$$
\begin{aligned}
\mathrm{Var}[\theta_{\bar{0}}] &= \frac{1}{N}\lambda_{\mathrm{eff}}(1 - \lambda_{\mathrm{eff}}) \\
&= \frac{1}{N}\left[p(1 - p) + q_{\mathrm{eff}}(1 - 2p)^2\right] + \mathcal{O}(q_{\mathrm{eff}}^2).
\end{aligned}
\tag{A3}
$$

This is strictly lower than unmitigated variance [Eq. (A1)] whenever $q_{\mathrm{eff}} < q$ and, for certain parameters, can be lower than variance with error detection [Eq. (A2)].

Finally, we note that the change in variance with or without error mitigation is generally small compared to intrinsic quantum uncertainty, i.e. $p(1 - p)$. For this reason, the more significant benefits offered by active error mitigation may be its application to problems where passive readout mitigation methods are unsuitable, e.g. when a response matrix cannot be accurately determined or the output of individual shots is required.

Appendix B: Review of Hamming codes

In this appendix, we briefly describe the family of classical error correction codes known as Hamming codes. The 7-qubit code considered in this work is a member of this family; in fact, the three-qubit repetition code is a member as well (though, for clarity, we have chosen not to use this terminology). Each Hamming code has a code distance of $d = 3$, implying that a single bit-flip error can be corrected and up to two bit-flip errors can be detected. Specifically, for any integer $r \geq 2$, a Hamming code can be constructed to encode $2^r - r - 1$ logical bits into $2^r - 1$ physical bits ($r = 2$ is the 3-qubit repetition code; $r = 3$ is the 7-qubit code). Thus, as r increases, the ratio of physical to logical bits quickly approaches one.

The general procedure for constructing a Hamming code is as follows. One begins by enumerating the $2^r - 1$ physical bits starting from $l = 1, 2, ..., 2^r - 1$ and assigning the bits with index 2^i for $i = 0, 1, \ldots, r - 1$ as parity bits and the rest as logical bits. The role of each parity bit is to store the parity of a particular set of logical bits. In particular, for the ith parity bit, this set includes all logical bits whose index l as a binary string satisfies $l_i = 1$ (i.e. the ith bit of l is equal to 1). For example, in the (7,4) code, the $l = 1$ parity bit stores the parity of the logical bits with index 3, 5, and 7.

To utilize the encoding, one checks if the state of each parity bit matches the measured parity of the corresponding set of logical bits; this is accomplished by multiplying the measurement outcome by the parity check matrix, as shown in Eq. (12) for the (7,4) code. If only a single bit-flip error has occurred, the index of the error is given by the binary string whose bits are set to 1 for each parity check that is violated. For example, if the $l = 1$ and $l = 2$ parity checks are violated, it indicates that an error occurred on the logical bit with index $l = 3$; while if only the $l = 1$ parity check is violated, it indicates that this parity bit itself had an error. However, if more than one bit-flip errors occur, the errors can no longer be correctly identified (though they can still be detected for up to two bit-flip errors); indeed, attempting to perform error correction would lead to a spurious final state.

Finally, we note that, for each Hamming code, one can form an extended Hamming code by adding a single extra parity bit, denoted with the label $l = 0$, which stores the parity of *all* the physical bits.. The effect of this extra parity bit is to increase the code distance from $d = 3$ to $d = 4$. We provide a detailed analysis of the Hamming (8,4) code in Appendix D.
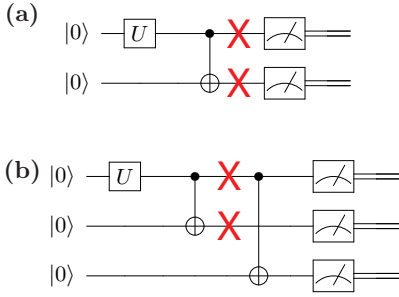
FIG. 11. Schematic of the single undetectable error channel for (a) the (2,1) code, and (b) the (3,1) code. The red crosses indicate the location of a bit-flip error (i.e. a Pauli X- or Y-type error).
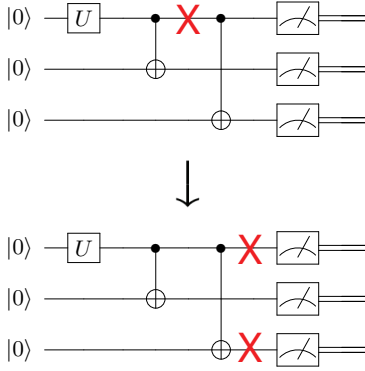

FIG. 12. Example of an uncorrectable error for the (3,1) code. The error occurs on a single site (top), but propagates to a second site (bottom) due to the second CNOT gate.

Appendix C: Analysis of gate errors

A key feature of active readout error correction is that the effective readout error rate remains linearly susceptible to gate errors during the encoding circuit, i.e. $Q_{\text{eff}}/k \approx \alpha\epsilon$, when $\epsilon, q \ll 1$. This is because single gate errors can lead to correlated bit-flip errors in the final measured outcomes. As we discussed in Section III, the prefactor α can be derived by summing all error channels that lead to a logical error, weighted by their individual likelihood (e.g. $\epsilon/16$ for a two-qubit symmetric depolarizing channel). Note that the definition of a logical error depends both on the encoding and also whether one is performing error detection or correction, as in the latter case it suffices to falsely *identify* an error.

We previously identified the relevant error channels for the (2,1) and (3,1) codes. For the (2,1) code, they consist of correlated errors that flip both of the physical bits, i.e. Pauli errors of the form XX, XY, YX, or YY [Fig. 11]. The (3,1) code with error detection is directly analagous: logical errors require all of the physical bits to be flipped and are caused only by correlated errors following the first CNOT gate. However, for the (3,1) code with error
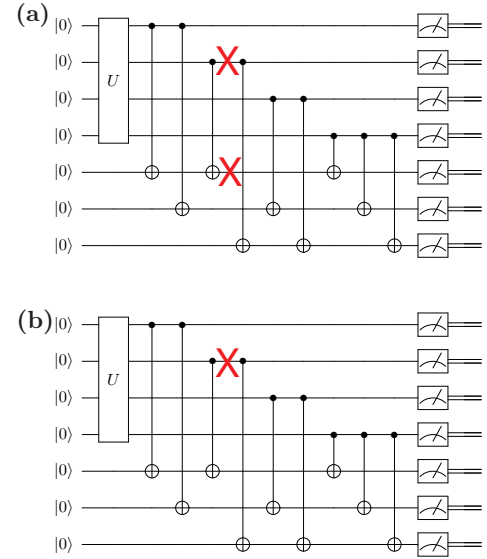

FIG. 13. Example of (a) an undetectable and (b) an uncorrectable error for the (7,4) code.

correction, logical errors require only two out of the three qubits to be flipped; this can occur by a two-qubit error on the second CNOT gate or single qubit error after the first CNOT gate (i.e. X or Y error) which propagates into a two-qubit error, as depicted in Fig. 12. Adding together these channels, we determine the values for α presented in Table I and confirmed numerically in Fig. 6.

We now extend this logic to analyze the Hamming (7,4) code. In order to have a fully *undetectable* logical error, all parity checks must remain unviolated, requiring that a bit-flip error occurs on one of the logical bits and *all* of the parity bits associated with it. As with repetition codes, this situation can only arise through a correlated two-qubit error after the first CNOT gate acting on one of the logical bits [Fig. 13(a)]. Indeed, such an error is equivalent to having a bit flip on the logical qubit itself *before* the encoding circuit, making it clearly undetectable. On the other hand, having a falsely *identified* logical error, requires only that a spurious combination of parity checks to be violated. This can occur in two possible ways. First, a two-qubit bit-flip error can occur after *any* of the CNOT gates (including the undetectable error mentioned above). Second, a single-qubit bit-flip error can occur after one CNOT gate and propogate into a multi-qubit error owing to a later CNOT gate [Fig. 13(b)]. For this, the single-qubit error must occur on the control bit of a subsequent CNOT gate.

This understanding allows us to directly count the number of relevant error channels for the (7,4) code. Referring to Fig. 4, there are a total of 9 CNOT gates which could directly lead to a two-qubit error, while there are 4 gates for which a single qubit error could later propogate. We mutiply these by the number of distinct Pauli errors per gate (4 two-qubit errors and 2 single-qubit errors), and weight them by their individual likelihood of occuring. This

yields a total error rate of $Q_{\text{eff}} = \frac{\epsilon}{16}(4*9 + 2*4) = 4*\frac{7}{2}$, which agrees precisely with the numerical results presented in Fig. 6.

We can further generalize this counting argument to predict the total error rate for an arbitrary Hamming code, denoted a $(2^r - 1, 2^r - r - 1)$ code . To do so, we note that the layout CNOT gates is given by the binary representation of each logical bit. Thus, the number of CNOT gates connected to a logical bit is equal to the number of 1s in the binary string, and the number of gates that could lead to a propagating error is equal to one fewer (i.e. only excludes the last gate). Summing each of the logical bits, the total number of CNOT gates is

$$\sum_{m=2}^{r} \binom{r}{m} k = r2^{r-1} - r$$
$$= (r-1)2^{r-1}$$

and the total number of propagating errors is

$$\sum_{m=2}^{r} \binom{r}{m}(m-1) = r2^{r-1} - r - (2^r - r).$$
$$= (r-1)2^{r-2}$$

This yields a total error rate

$$Q_{\text{eff}} = \frac{\epsilon}{16}\left[4(r-1)2^{r-1} + 2(r-1)2^{r-1}\right],$$

which, for $r \gg 1$, we can approximate as

$$Q_{\text{eff}} \approx \frac{\epsilon}{16}\left(4r2^{r-1} + 2r2^{r-1}\right).$$

Recalling that the number of logical qubits is $k \approx 2^r$, we can express the error rate as $Q_{\text{eff}}/k \approx \alpha\epsilon$ with $\alpha \approx 3r/16 \approx 3/16 \log_2 n$. Incidentally, we could also have arrived at this result by realizing that the number of CNOT gates per logical qubit is sharply peaked at $r/2$, corresponding to the typical number of 1's in a random binary string.

We conclude that, when performing error correction, the error rate per qubit increases logarithmically with the total number of qubits. This is in contrast to the *constant* error rate per qubit, i.e. $\alpha = 1/4$, when performing error detection. While the exact error rates depended on our model of 2-qubit depolarizing noise, we expect these trends to hold for arbitrary noise models.

Appendix D: Hamming (8,4) code

As mentioned in Appendix B, an extended version of each Hamming code may be obtained by including an additional parity bit that stores the parity of all logical bits. This increases the code distance from $d = 3$ to $d = 4$, enabling the detection of up to three bit-flip errors and the correction of only single bit-flip, as in the non-extended version. Notably, this implies that two-qubit errors can
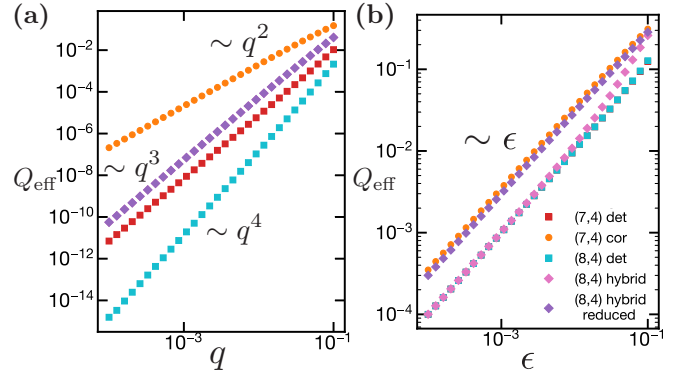

FIG. 14. Total readout error rate, Q_{eff}, for different error mitigation strategies, analogous to Fig. 6. Note the "hybrid" label for the (8,4) code indicates that single bit-flip errors are corrected, while others errors are detected and discarded. The "reduced" circuit for the (8,4) code is shown in Fig. 15.

be *distinguished* from single-qubit errors; in contrast, in the non-extended version, two-qubit errors can always be misidentified as a single-qubit error. Thus, a common usage for the code is to perform a hybrid between error correction and error detection, i.e. one corrects ostensibly single-qubit errors and discards multi-qubit errors.

We consider the implementation of this code using the encoding circuit shown in Fig. 4. The corresponding parity check matrix is given by:

$$H = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}_{[8,4]} . \quad (D1)$$

If the first parity check is violated, it suggests that a single bit-flip error has occurred, and the remaining parity checks can locate the error in the same way as the non-extended version (of course, the same scenario could occur with three bit-flip errors). On the other hand, if the first parity check succeeds, but one of the other checks fail, it indicates that multiple bit-flip errors have occurred, but because the location of these cannot be determined, the measurement is discarded.

In Fig. 14, we test the performance of this hybrid approach via noisy simulations with the same noise channels as in main text. We first verify that the total error rate scales as $Q_{\text{eff}} \sim q^3$ with respect to readout noise, as expected from the fact that three bit-flip errors can be misidentified as a single bit-flip error. More surprisingly, the linear susceptibility with respect to gate errors is $Q_{\text{eff}}/k = \alpha\epsilon$ with $\alpha = 1/4$, the same as other codes performing pure error detection. While this suggest that the hybrid approach is more robust to gate errors, a close inspection of the circuit reveals that all gate errors that plagued the (7,4) code with error correction are not *correctable* by the (8,4) code; rather they contribute to the number of discarded measurements. More specifically, such errors propagate during the last round of CNOT gates

(shown in the box of Fig. 4), resulting in least two bit-flip errors on the measurement outcome, which cannot be corrected. Indeed, we verify that the number of discarded measurements for the (8,4) code is equivalent to that of the (7,4) code with error detection when only gate errors are included. Thus, utilizing the (8,4) code with the hybrid error correction / detection strategy is practically equivalent to utilizing the (7,4) code with error detection in terms of handling gate errors (as well as readout errors).

Finally, we note that a simpler circuit can, in fact, be utilized to implement the (8,4) code. The reason is that the parity bits redundantly stores the parity of certain combinations of logical bits, so one can obtain the parity of all the physical bits by interacting with only a partial set of the bits. The reduced circuit is shown in Fig. 15 and consists of only 3 CNOT gates acting on the last parity bit rather than the original 7 gates. By performing noisy simulations, we verify that this reduced circuit has the same scaling with respect to readout errors as the full circuit. However, the simplified circuit exhibits a *worse* performance with respect to gate errors; in particular, we find $\alpha = 3/4$. The explanation for this counter-intuitive

result is that, in the simplified circuit, many of gate errors that occur in the non-extended part of the circuit do not propagate during the final round of CNOT gates. As a result, many such errors are misidentified as single bit-flip errors and lead to a spurious final state, similar to the (7,4) code performing error correction. In contrast, for the full circuit shown in Fig. 4, these same errors always propagate and are flagged as multi-qubit errors which are discarded.
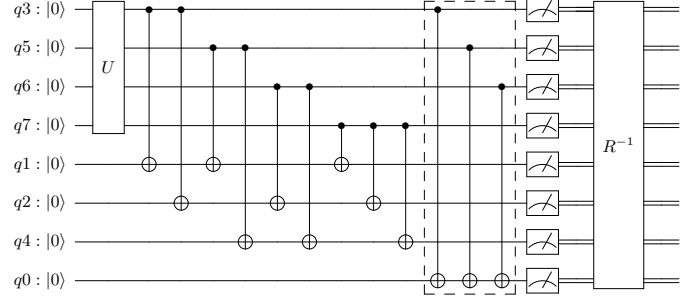

FIG. 15. Reduced circuit for encoding the Hamming (8,4) code.