



OPEN

# Quantum error mitigation by Pauli check sandwiching

Alvin Gonzales<sup>1✉</sup>, Ruslan Shaydulin<sup>3</sup>, Zain H. Saleem<sup>2</sup> & Martin Suchara<sup>2,4</sup>

We describe and analyze an error mitigation technique that uses multiple pairs of parity checks to detect the presence of errors. Each pair of checks uses one ancilla qubit to detect a component of the error operator and represents one layer of the technique. We build on the results on extended flag gadgets and put it on a firm theoretical foundation. We prove that this technique can recover the noiseless state under the assumption of noise not affecting the checks. The method does not incur any encoding overhead and instead chooses the checks based on the input circuit. We provide an algorithm for obtaining such checks for an arbitrary target circuit. Since the method applies to any circuit and input state, it can be easily combined with other error mitigation techniques. We evaluate the performance of the proposed methods using extensive numerical simulations on 1850 random input circuits composed of Clifford gates and non-Clifford single-qubit rotations, a class of circuits encompassing most commonly considered variational algorithm circuits. We observe average improvements in fidelity of 34 percentage points with six layers of checks.

Hardware errors or noise arising from qubit imperfections such as unwanted interactions with the environment limit the power of near-term quantum technologies. Since these devices lack the necessary number of qubits and error rates to perform quantum error correction<sup>1–3</sup>, error mitigation is required in order to increase the fidelity of computations. In this work we investigate error mitigation that uses a small number of ancillas to suppress the effect of errors. Various error mitigation techniques have been developed, such as zero-noise extrapolation<sup>4–6</sup>, which uses different error rates to reduce the error in the measurement of an observable; probabilistic error cancellation<sup>4</sup>, which uses an ensemble of known noisy circuits to approach the correct expectation value; dynamical decoupling<sup>7–9</sup>, which uses timed control sequences to suppress interactions of the target quantum system with its environment; readout error mitigation<sup>10</sup>, which uses classical postprocessing techniques to mitigate measurement errors; and symmetry verification<sup>11–14</sup>, which verifies symmetries in computational problems of interest and discards erroneous computations.

Protocols that improve measurements of an observable have applications in problems such as the estimation of the ground state energy of a given Hamiltonian<sup>15</sup>. In contrast, protocols that improve fidelity generally apply to any problem. A main feature of many techniques aimed at improving measurements of an observable or reducing readout error is that they have no quantum overhead; in other words, they require no extra qubits or quantum operations (gates). Thus, these techniques are ideal for the noisy intermediate-scale quantum (NISQ) era<sup>16</sup> because current state-of-the-art NISQ devices contain few qubits, typically fewer than 50, and a limited number of gate operations because of fast decoherence times.

As quantum technology develops, error mitigation schemes must adapt and take advantage of improvements in qubit count and quality<sup>17,18</sup>. Qubit count and error rates vary widely depending on the underlying qubit technology. Additionally, many of the current error mitigation techniques such as dynamical decoupling and probabilistic error cancellation require intricate tailoring of the protocol to the noise. Thus, they typically require the added overhead in costly quantum tomography<sup>19</sup>.

In this work we theoretically and numerically study a quantum error mitigation technique inspired by stabilizer codes, that aims at improvement in quantum state fidelity. We build on the results of<sup>20</sup>, where they first explored the scheme of sandwiching a circuit between pairs of parity checks. Note that they refer to the pairs of checks as extended flag gadgets, inspired by the work in<sup>21,22</sup>. Our research puts this parity check scheme on a firm theoretical foundation and numerically demonstrates its efficacy on a wide variety of quantum circuits. The main contributions of our work are: (1) extending the analysis to greater than two layers of checks, (2) establishing the theoretical limits of the technique, which culminates in the unit fidelity result of Theorem 1, (3) providing parity checks in Propositions 1 and 2 that saturate this fidelity bound and hence answers an open question in<sup>20</sup>

<sup>1</sup>Intelligence Community Postdoctoral Research Fellowship Program, Argonne National Laboratory, Lemont, IL, USA. <sup>2</sup>Mathematics and Computer Science Division, Argonne National Laboratory, Lemont, IL, USA. <sup>3</sup>Global Technology Applied Research, JPMorgan Chase, New York, NY, USA. <sup>4</sup>Amazon Web Services, Amazon, Seattle, WS, USA. ✉email: agonza@siu.edu

regarding optimal checks to use, (4) providing a protocol that efficiently determines Pauli parity check pairs that can be used for a given input circuit, and (5) providing numerical simulations for a wide variety of random input circuits consisting of varying qubit count, CNOT count, non-Clifford gate count, and layer count.

The error mitigation scheme that we study in this paper at its basic level of one layer uses one ancilla and two controlled unitary operations, which we refer to as checks. The parity checks sandwich the input circuit. Consequently, the error operator is conjugated between two controlled parity matrices. We measure the ancilla and postselect the state on the measurement outcomes. The net effect of the checks and the postselection is a transformed error map, where terms of the error map that anticommute with the checks are eliminated in the postselected state. The performance of this technique, measured by the improvement in quantum state fidelity, improves with the depth of the input circuit. Furthermore, this scheme is tunable, meaning that the number of layers and ancillary qubits used can be set by the user.

This protocol shares some similarity to symmetry verification, which also uses stabilizer-style parity checks to improve the fidelity of the quantum state and requires no knowledge of the noise. However, unlike symmetry verification, which requires input states to be restricted to a specific eigenspace, this scheme places no restriction on the input state. Thus, the technique applies to subcircuits directly and can be easily combined with other error mitigation methods.

In Theorem 1, we prove that in a restricted scenario where the noise does not affect the checks (see Fig. 4a,b) there exist checks such that the postselected state is noiseless and the fidelity reaches unity. We provide an example of a randomly generated Clifford circuit with added checks that saturates this fidelity bound.

We also investigate the performance of the scheme with numerical simulations in a more realistic setting where the checks are also noisy. The numerical simulations consist of 1850 (unmitigated) randomly generated five- and ten-qubit circuits composed of Clifford + arbitrary diagonal unitary gates. Our technique shows an average fidelity gain of 34 percentage points for random input circuits consisting of 40 CNOTs with six layers of (noisy) checks; see Fig. 10a,b. The increase in fidelity comes at a cost of a lower probability of postselecting on the ancillas' measurement outcomes. We also provide Clifford simulations that give intuition that this technique will perform well for deep circuits.

This paper is organized as follows. In “Background” we review relevant background and provide definitions that are used in the paper. In “Pauli sandwich error mitigation protocol: single layer” we provide the single-layer protocol. In “Errors detected by the Pauli sandwich” we describe the theoretical foundation of the technique. In “Pauli sandwich error mitigation protocol: multiple layers” we provide the full multilayer scheme. In “Upper bounds on fidelity and required number of checks” we prove Theorem 1 and provide bounds on the number of layers required to reach unit fidelity for the restricted scenario where the noise does not affect the checks. In “General errors and hardware considerations” we discuss how our results apply in general settings. In “Protocol for finding checks quickly” we introduce techniques for finding checks quickly by using a precalculated table of commutation rules that eliminates the need to perform matrix multiplication. In “Numerical results” we give the results of our numerical simulations. In “Conclusions” we discuss our results and possible areas for future work.

## Background

We begin with definitions and notation. For a detailed introduction to modeling of noise in quantum computation the reader is referred to Chapter 8 of<sup>23</sup>.

The most general evolution of an open quantum system is given by a dynamical map<sup>24</sup>

$$\mathcal{E} : \rho_S \rightarrow \rho'_S, \quad (1)$$

where  $\rho_S$  and  $\rho'_S$  are elements of the system Hilbert space  $\mathcal{H}_S$ .  $\mathcal{H}_S$  is a subspace of the system and environment Hilbert space  $\mathcal{H}_{SE}$ , where  $S$  is the system and  $E$  is the environment.

In the case of an initially unentangled system and environment, the map is completely positive and trace preserving. It can be derived by taking the partial trace of the global unitary evolution and yields the operator sum representation

$$\begin{aligned} \mathcal{E}(\rho_S) &= \text{tr}_E(U_{SE}\rho_S \otimes \rho_E U_{SE}^\dagger) \\ &= \sum_i E_i \rho_S E_i^\dagger, \end{aligned} \quad (2)$$

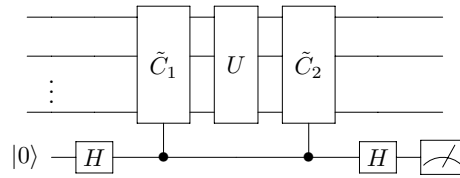
where  $U_{SE}$  is a unitary acting across the system and environment<sup>25,26</sup>. The operators  $E_i$  in Eq. (2) are commonly called Kraus operators<sup>26</sup>. A map is completely positive (CP) if it maps all positive operators to positive operators when extended by the identity map to arbitrary higher dimensions<sup>25</sup>, namely,

$$\mathcal{E} \otimes \mathcal{I}_n(\rho) \geq 0 \quad \forall n, \rho, \quad (3)$$

where  $\mathcal{I}_n$  is the  $n$ -dimensional identity map and  $\rho$  is a density matrix. This extension to higher dimensions is required to ensure that when the input state is part of a higher-dimensional state, the output of the map is still positive. A map is trace preserving if

$$\sum_i E_i^\dagger E_i = \mathbb{I}. \quad (4)$$

Maps that do not satisfy Eq. (3) are called not completely positive (NCP) maps. NCP maps play a significant role in non-Markovian evolutions<sup>27</sup>, where the evolution of the state is often not decomposable into a sequence of completely positive maps. NCP maps have the form



**Figure 1.** Overview of the one-layer version of the PCS scheme.  $U$  represents the gates of the computation and acts across  $n$  compute qubits.  $U$  is sandwiched between two controlled unitaries comprising  $\tilde{C}_1$  and  $\tilde{C}_2$  that satisfy Eq. (11). The ancilla is the bottom qubit. The measurement is performed in the  $\{|0\rangle, |1\rangle\}$  basis. The measurement outcome one is discarded, and zero is kept.

$$\mathcal{E}(\rho) = \sum_i \eta_i E_i \rho E_i^\dagger, \tag{5}$$

where  $\eta_i = \pm 1$  and at least one  $\eta_i = -1$  exists<sup>25</sup>.

In this paper we use fidelity as a figure of merit. For two quantum states  $\rho$  and  $\omega$ , fidelity is defined as

$$F(\rho, \omega) = \left( \text{tr} \sqrt{\sqrt{\rho} \omega \sqrt{\rho}} \right)^2. \tag{6}$$

Fidelity is symmetric with regard to its inputs.

Let  $U$  denote the unitary operation implemented by the target circuit that we want to error mitigate. Let  $\rho'$  be the density matrix representing the ideal, noiseless output of the target circuit,  $\rho_n$  be the density matrix of the noisy quantum state produced by the target circuit, and  $\rho_m$  be the density matrix of the noisy error-mitigated state produced by the error mitigated target circuit.

We denote the fidelity of the noisy state before error mitigation as  $F_n = F(\rho_n, \rho')$ , and the fidelity of the state after application of error mitigation as  $F_m = F(\rho_m, \rho')$ .

We define the fidelity gain (improvement due to the technique) as

$$F_m - F_n. \tag{7}$$

We say that the method “detects all errors” when  $F_m = 1$  or equivalently when the error map  $\mathcal{E}$  on the postselected state is identity, in other words, when all the Kraus operators of  $\mathcal{E}$  are proportional to identity.

Next we describe the noise model used in our numerical simulations. The depolarizing channel for dimension  $d$  is

$$\mathcal{D}_p(\rho) = (1 - p)\rho + p \frac{\mathbb{I}}{d}, \tag{8}$$

where  $0 \leq p \leq 1$ <sup>23</sup>. For the numerical simulations of noisy circuits the only noiseless gates are a measurement gate or the input state, which is generated by a random circuit. Otherwise, we apply the single-qubit depolarizing channel after each single-qubit gate and the two-qubit depolarizing channel after every two-qubit gate. Throughout this paper, we set the two-qubit error rate to ten times the single-qubit error rate, an assumption that roughly corresponds to noise observed in current NISQ systems<sup>28,29</sup>.

### Methods

In “Pauli sandwich error mitigation protocol: single layer” and “Errors detected by the Pauli sandwich” we describe the single-layer Pauli Check Sandwiching (PCS) technique and show that this protocol leads to a transformation of the error map. In “Pauli sandwich error mitigation protocol: multiple layers” and “Upper bounds on fidelity and required number of checks” we describe the multilayer protocol and prove that we can reach a fidelity of one between a noisy-mitigated circuit and a noiseless circuit when the error map is restricted to a subset of qubits. We also provide a small number of checks that achieve this fidelity. In “General errors and hardware considerations” we investigate how our techniques apply in a general setting. In “Protocol for finding checks quickly” and “Numerical results” we give the results of numerical experiments across 1850 unmitigated random circuits.

**Pauli sandwich error mitigation protocol: single layer.** We begin by describing the simplest version of the Pauli Check Sandwiching technique that consists of a single pair of parity checks sandwiching the computation (one “layer”). Figure 1 shows a graphical view of the protocol. The unitary operation  $U$  represents the gates of the computation. The bottom qubit is the single ancilla introduced by this scheme, and we commonly refer to the  $n$  qubits above as the compute or computation qubits.

Let  $C_2$  ( $C_1$ ) be a controlled unitary with control on the ancilla that applies  $\tilde{C}_2$  ( $\tilde{C}_1$ ) on the compute target qubits. Mathematically,

$$C_1 = \tilde{C}_1 \otimes |1\rangle\langle 1| + \mathbb{I} \otimes |0\rangle\langle 0| \tag{9}$$

$$C_2 = \tilde{C}_2 \otimes |1\rangle\langle 1| + \mathbb{I} \otimes |0\rangle\langle 0|. \tag{10}$$

This scheme also requires that

$$\tilde{C}_2 U \tilde{C}_1 = U. \tag{11}$$

Before continuing, we make an important distinction between two protocols: the *efficient PCS protocol* and the *general PCS protocol*. For the *efficient PCS protocol*, we restrict  $\tilde{C}_1$  and  $\tilde{C}_2$  to be elements of the  $n$ -qubit Pauli group  $\mathcal{P}_n$ , where

$$\mathcal{P}_n = \{\mathbb{I}, x, y, z\}^{\otimes n} \times \{\pm 1, \pm i\}. \tag{12}$$

These added conditions are partly due to the difficult problem of determining the optimal circuit that implements  $\tilde{C}_1$  from a given  $\tilde{C}_2$  and  $U$ . Note that  $\tilde{C}_2$  and  $\tilde{C}_1$  can be much more general and still satisfy Eq. (11). Thus, there are no additional constraints on the checks in the *general PCS protocol*.

In the *general PCS protocol* and for a given  $U$ , any unitary  $\tilde{C}_2$  can be used because in Eq. (11) we can always pick  $\tilde{C}_2$  and solve for  $\tilde{C}_1$ . We note in the text if a result holds for a specific case. If no statement is made, then the result holds for both scenarios.

The single-layer protocol is as follows.

1. Initialize the ancilla to  $|0\rangle$  and apply a Hadamard gate. Perform  $C_1$  with the control on the ancilla qubit and target on the compute qubits.
2. Perform  $U$  on the compute qubits.
3. Perform  $C_2$  with the control on the ancilla qubit and target on the compute qubits.
4. Apply a Hadamard gate to the ancilla. Measure the ancilla in the  $\{P_0 = |0\rangle\langle 0|, P_1 = |1\rangle\langle 1|\}$  basis, and discard the results where the outcome is  $P_1$ . We keep the result where the outcome is  $P_0$ .

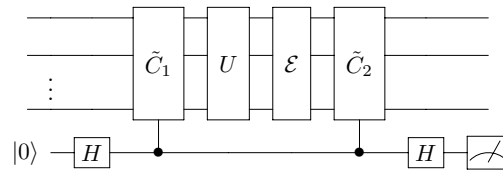
**Errors detected by the Pauli sandwich.** We now consider the effect of this scheme on an error map  $\mathcal{E}$  acting on the compute qubits after  $U$ , as shown in Fig. 2a. Let  $\mathcal{E}(\rho) = \sum_i E_i \rho E_i^\dagger$ . Then the postselected output state of the protocol is

$$\rho_m = \frac{\sum_i \left[ (\tilde{C}_2 E_i \tilde{C}_2^\dagger + E_i) U \rho U^\dagger (\tilde{C}_2 E_i^\dagger \tilde{C}_2^\dagger + E_i^\dagger) \right]}{\text{tr} \left( \sum_i \left[ (\tilde{C}_2 E_i \tilde{C}_2^\dagger + E_i) U \rho U^\dagger (\tilde{C}_2 E_i^\dagger \tilde{C}_2^\dagger + E_i^\dagger) \right] \right)}. \tag{13}$$

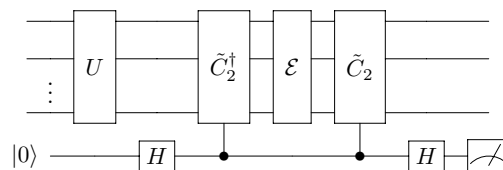
As shown in Fig. 2b, this protocol transforms the error map. We can write the postselected state given in Eq. (13) in terms of a new error map  $\mathcal{E}'$ ,

$$\rho_m = \frac{\mathcal{E}'(U \rho U^\dagger)}{\text{tr}[\mathcal{E}'(U \rho U^\dagger)]}, \tag{14}$$

where  $\mathcal{E}'$  has Kraus operators

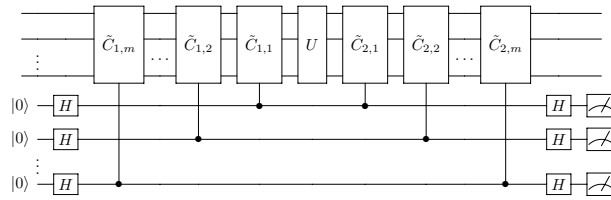


(a)  $\mathcal{E}$  is a noise map.



(b) Equivalent noisy circuit using Eq. (11). The gates after  $U$  along with the postselection on the ancilla can be seen as the transformed error map.

**Figure 2.** Noisy single-layer scheme.



**Figure 3.** Multilayer scheme. There are  $n$  compute qubits,  $m$  layers, and  $m$  ancillas. The second index in the controlled unitaries represents the layer. Each layer uses one ancilla and two checks. The checks sandwich the input circuit.

$$E'_i = \frac{\tilde{C}_2 E_i \tilde{C}_2^\dagger + E_i}{2} \tag{15}$$

and the factor of  $1/2$  comes from multiplying Eq. (13) by a convenient form of one, namely,  $(1/4)/(1/4)$ . We can now observe the power of this error mitigation technique. The error operators  $E_i$  can be expanded in the Pauli basis. Thus, let

$$E_i = \sum_{\tilde{\sigma}_j \in \mathcal{P}_n} \alpha_{ij} \tilde{\sigma}_j, \tag{16}$$

where  $\tilde{\sigma}_j$  is an element of the Pauli group and  $\alpha_{ij} = \text{tr}(E_i \tilde{\sigma}_j) / (2^n)$  is a complex constant. Let  $\tilde{C}_2 \in \mathcal{P}_n$ . Each  $\tilde{\sigma}_j$  term in the expansion of  $E_i$  either commutes or anticommutes with  $\tilde{C}_2$ . Substituting Eq. (16) into Eq. (15), we see that the  $\tilde{\sigma}_j$  elements that anticommute with  $\tilde{C}_2$  are eliminated and

$$E'_i = \sum_{\tilde{\sigma}_j \in \mathcal{P}'_n} \alpha_{ij} \tilde{\sigma}_j, \tag{17}$$

where  $\mathcal{P}'_n$  is the Pauli group excluding the elements that anticommute with  $\tilde{C}_2$ .

The effect of the protocol on the error map shares some similarities to that of twirling<sup>30–32</sup>. In twirling, the twirling set  $T$  is used to conjugate the error map:

$$\begin{aligned} & \left| \frac{1}{|T|} \sum_{V \in T} V \mathcal{E}(V^\dagger \rho V) V^\dagger \right. \\ & \left. = \frac{1}{|T|} \sum_{i, (V \in T)} V E_i V^\dagger \rho V E_i^\dagger V^\dagger. \right. \end{aligned} \tag{18}$$

Usually, twirling is performed by using the Pauli or Clifford group as the twirling set. When twirling is performed with a suitable set, it transforms the noise into a Pauli channel. However, the PCS scheme is in some sense more powerful since it completely eliminates the contribution of anticommuting Pauli terms.

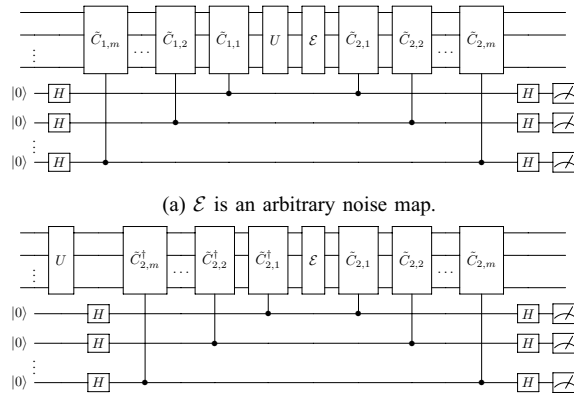
**Pauli sandwich error mitigation protocol: multiple layers.** The suppression of errors from anticommuting Pauli terms in the postselected state can be enhanced by introducing multiple layers of the single-layer error mitigation technique. A graphical view of how this scheme works is given in Fig. 3. There are  $m$  layers with each layer consisting of controlled operations  $C_{1,k}$  and  $C_{2,k}$ , where the second index represents the layer, and one ancilla corresponding to each layer. Each pair of  $C_{1,k}$  and  $C_{2,k}$  satisfies

$$\tilde{C}_{2,k} U \tilde{C}_{1,k} = U. \tag{19}$$

The multilayer scheme generalizes the single-layer scheme and is performed as follows.

1. Initialize the ancillas to  $|0\rangle$ , and perform Hadamard gates on the ancillas. Perform  $C_{1,k}$  with control on the  $k$ th ancilla qubit and target on the compute qubits.
2. Perform  $U$  on the compute qubits.
3. Perform  $C_{2,k}$  with control on the  $k$ th ancilla qubit and target on the compute qubits.
4. Perform Hadamard gates on the ancillas. Measure all the ancillas in the  $\{P_0 = |0\rangle\langle 0|, P_1 = |1\rangle\langle 1|\}$  basis, and discard the results where at least one of the outcomes is  $P_1$ . We keep the result where all the outcomes are  $P_0$ .

**Upper bounds on fidelity and required number of checks.** Now let us consider a noise map  $\mathcal{E}(\rho) = \sum_i E_i \rho E_i^\dagger$  acting after  $U$  on a subset of qubits as shown in Fig. 4a. From Eq. (15), in the expansion of  $E_i$  in the Pauli basis, we know that the  $k$ th layer eliminates Pauli terms that anticommute with  $\tilde{C}_{2,k}$ . This immediately



(a)  $\mathcal{E}$  is an arbitrary noise map.  
 (b) Equivalent circuit using Eq. (19). Reminiscent of the single-layer scheme, the gates after  $U$  along with the postselection on the ancillas can be seen as the transformed error map.

**Figure 4.** Noisy multilayer scheme.

leads to the observation that we can detect all errors under the noise model shown in Fig. 4a, which we prove in the following theorem. Theorem 1 holds in general for the *general PCS protocol* and it holds for the *efficient PCS protocol* when the checks are in the Pauli group, in other words, when  $U$  is Clifford. Note that we discuss why these results hold for NCP errors as well later at the start of “General errors and hardware considerations”.

**Theorem 1** (Unit Fidelity) *If errors are restricted to act only on the compute qubits, for any noisy unitary quantum circuit  $U$  acting on  $n$  compute qubits, there exist checks (see Fig. 4a) such that the fidelity between the post selected state and a noiseless run (noiseless execution of  $U$  only) reaches one.*

**Proof** First, note that if the error map  $\mathcal{E}(\rho) = \sum_i E_i \rho E_i^\dagger$  is the identity map, then the fidelity between the output  $\rho_m$  of the error-mitigated circuit and the output  $\rho'$  of a circuit with only  $U$  (a noiseless run) is

$$F(\rho_m, \rho') = 1. \tag{20}$$

This directly follows from Eq. (19). Thus, if we can transform all the Kraus operators  $E_i$  of the error  $\mathcal{E}$  to identity in the postselected state, then we have the result.

Notice that from Eq. (19), Fig. 4a is equivalent to Fig. 4b and the error map is conjugated by multiple layers of checks. Expanding the error in the Pauli basis, we have

$$E_i = \sum_{\tilde{\sigma}_j \in \mathcal{P}_n} \alpha_{ij} \tilde{\sigma}_j, \tag{21}$$

where  $\tilde{\sigma}_j$  is an element of the Pauli group  $\mathcal{P}_n$  and  $\alpha_{ij} = \text{tr}(E_i \tilde{\sigma}_j) / (2^n)$  is a complex constant. Let  $\tilde{C}_{2,i} \in \mathcal{P}_n, \forall i$ .

We now make the results given in “Errors detected by the Pauli sandwich” recursive. First, we label the check layers from 1 to  $m$  starting with the innermost layer. Then, Eq. (15) can be written recursively as

$$E_i^{(k)} = \frac{\tilde{C}_{2,k} E_i^{(k-1)} \tilde{C}_{2,k}^\dagger + E_i^{(k-1)}}{2}, \tag{22}$$

where  $(k)$  represents the layer and  $E_i^{(0)}$  is the initial error Kraus operation. This leads to the recursive form of Eq. (17),

$$E_i^{(k)} = \sum_{\sigma_j \in G_n^{(k)}} \alpha_{ij} \tilde{\sigma}_j, \tag{23}$$

where  $G_n^{(k)}$  is the Pauli group excluding the elements that anticommute with  $\{\tilde{C}_{2,1}, \tilde{C}_{2,2}, \dots, \tilde{C}_{2,k}\}$ . Letting  $k$  equal the size of  $\mathcal{P}_n$  (excluding global phases), namely,  $4^n$ , we get  $E_i^{(k)} = \alpha_i I$ . The  $\alpha_i$  is a constant that cancels out under renormalization, and the result follows.  $\square$

Before proceeding, we need to clarify the implications of Theorem 1. In that theorem, if we satisfy the conditions, we will have unit fidelity in the postselected state. However, the probability of postselecting is

$$P(\bar{0}) = \text{tr}(\mathcal{E}^{(m)}(U\rho U^\dagger)), \tag{24}$$

where the Kraus operators of  $\mathcal{E}^{(m)}$  are given by Eq. (23). In Eq. (23) the checks will eliminate all the Pauli terms that are nonidentity. Thus, we see that if all the Kraus operators of the error map are traceless, in other words, contain no identity term in their expansion in the Pauli basis, all the Kraus operators for the error map in the postselected state will be the zero matrix, and the probability of postselecting is zero. This makes sense because

we are not correcting errors, but mitigating errors by post selecting outcomes. The theorem holds trivially in this scenario because there is no post selected state.

Moreover, from Fig. 4a and Theorem 1, it seems that we can set  $\mathcal{E}(\rho) = U^\dagger \rho U$ , which eliminates  $U$ , and use only the checks for the implementation of the circuit. While this is certainly true, we must consider the postselection probability. If  $U$  is traceless, the probability of postselecting is zero.

Next we provide a small number of  $C_2$  checks that can reach unit fidelity in the setting of Theorem 1. The following results given in Propositions 1 and 2 are for the *general PCS protocol*. Propositions 1 and 2 hold for the *efficient PCS protocol* given that the checks are in the Pauli group, in other words,  $U$  is Clifford. Propositions 1 and 2 hold for the noise model given in Fig. 4a. For arbitrary weight-one Kraus errors, that is, the Kraus error operators,  $E_i$  act only on a single qubit; there exist two layers, where  $\tilde{C}_{2,1}$  and  $\tilde{C}_{2,2}$  are max weight, and we reach unit fidelity in the postselected state.

**Proposition 1** (Weight-One Kraus Operators: Two layers of max weight checks are sufficient) *For the noise model given in Fig. 4a and for all  $\mathcal{E}$  consisting of only weight-one  $E_i$ , there exist two layers of checks such that we have unit fidelity in the postselected state. The  $C_2$  part of the checks requires a total of  $2n$  cnot gates, where  $n$  is the number of compute qubits.*

**Proof** Each of the single-qubit errors can be expanded in terms of the single-qubit Pauli gates. Thus,

$$E_{i,k} = \sum_j \alpha_{i,j} \sigma_{j,k}, \quad (25)$$

where  $k$  is the qubit it is acting on,  $\sigma_j$  is a Pauli matrix or identity, and  $\alpha_{i,j}$  is a complex constant. Let our checks be

$$\tilde{C}_{2,1} = x^{\otimes n} \quad (26)$$

and

$$\tilde{C}_{2,2} = z^{\otimes n}. \quad (27)$$

These checks are inspired by the parity checks used in Shor's code<sup>33</sup>. The  $\tilde{C}_{2,1}$  consist of tensors of Pauli X and anticommutes with Pauli Y and Pauli Z errors in Eq. (25). The  $\tilde{C}_{2,2}$  consist of tensors of Pauli Z and anticommutes with Pauli X errors in Eq. (25). From Theorem 1, the anticommuting terms in the error operators are suppressed. Thus, these two layers of checks are sufficient to reach fidelity one.  $\square$

The checks given in Proposition 1 can detect all errors  $\mathcal{E}$  that consist of weight-one Kraus operators  $E_i$ . This class of errors contains error maps that are more general than just single-qubit error maps. For example,  $E_1$  can act on qubit one, and  $E_2$  can act on qubit two.  $E_1$  and  $E_2$  are weight-one errors, but the overall map affects multiple qubits.

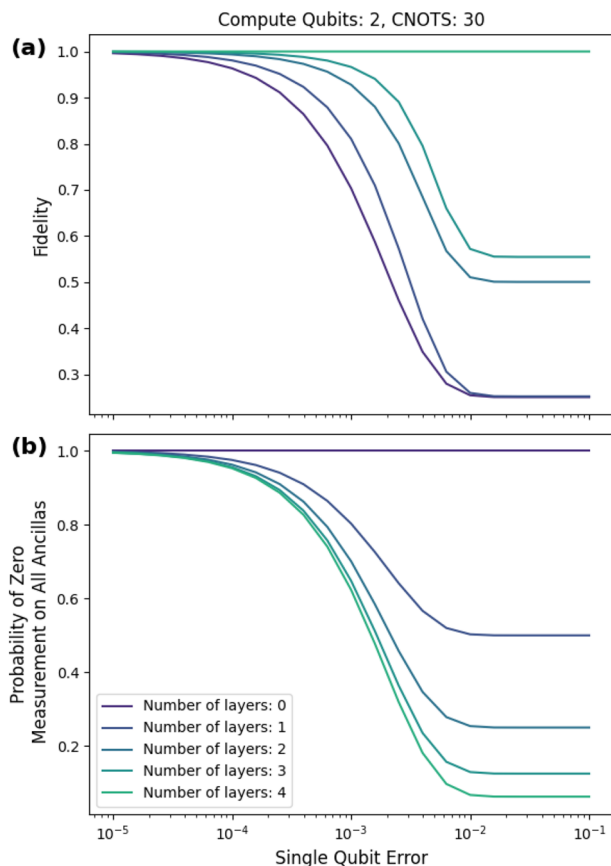
**Remark** At least two layers are necessary to reach fidelity one in Proposition 1. To see this, we need only show that a single layer is insufficient for arbitrary weight-one errors. Consider a circuit with only one compute qubit. For an arbitrary single-layer scheme, let  $\tilde{C}_2 = W$  be the check. Then let the error map be  $E = W$ . The check and the error do not anticommute so the error map in the postselected state is not identity. Thus, a single layer is insufficient to detect all weight-one errors; at least two layers are necessary. Proposition 1 shows that we can always saturate this lower bound on the number of required checks.

We can also reach fidelity one for arbitrary weight errors for the error model given in Fig. 4a with a small number of weight-one  $\tilde{C}_{2,k}$ . These checks are generators of the Pauli group and require  $2n$  layers, but the  $C_2$  components of the checks require the same number of cnot gates as in Proposition 1. Thus, generally at the cost of more ancillas, we can detect all errors on the postselected state. Consider two weight-one  $\tilde{C}_2$  checks of  $\sigma_1$  and  $\sigma_3$  on the  $k$ th compute qubit. All Pauli group elements that are nonidentity on the  $k$ th qubit anticommute with either  $\sigma_1$  or  $\sigma_3$ . This leads to the following small set that can reach fidelity one.

**Proposition 2** (Any Error:  $2n$  number of weight-one checks are sufficient) *For the noise model given in Fig. 4a and arbitrary errors, let  $n$  be the number of compute qubits. Then there exist  $2n$  number of distinct (ignoring the global phase) weight-one  $\tilde{C}_{2,k}$  such that we have unit fidelity in the postselected state.*

**Proof** Let the  $k$ th compute qubit have two layers acting on it with  $\tilde{C}_{2,r}^{(k)} = \sigma_1^{(k)}$  and  $\tilde{C}_{2,l}^{(k)} = \sigma_3^{(k)}$ . All Pauli group elements that are nonidentity on the  $k$ th qubit anticommute with at least one of the checks. Thus, this eliminates all Pauli terms in the expansion of the error Kraus operators that do not have identity on the  $k$ th qubit for the postselected state. We repeat these checks for the other compute qubits. The same argument holds in general for  $\{\sigma_i^{(k)}, \sigma_j^{(k)} | i \neq j\}$  and the result follows.  $\square$

Figure 5 shows an example of a random Clifford circuit consisting of two compute qubits and 30 CNOT gates that gives unit fidelity for the postselected state. This matches the prediction of Theorem 1. We use a Clifford circuit to guarantee that we can get the desired checks with the *efficient PCS protocol*. We use the checks provided in Proposition 2. The two checks on each compute qubit are  $\sigma_1^{(k)}$  and  $\sigma_2^{(k)}$ , and we vary the number of layers



**Figure 5.** Example of checks that detect all errors. The upper bound on fidelity is saturated at four layers for this randomly generated Clifford input circuit consisting of two qubits and 30 CNOT gates. We use depolarizing noise for the given noise model in Fig. 4a. The two-qubit error rate is ten times the single-qubit error rate. The single qubit-error rate ranges from  $10^{-5}$  to  $10^{-1}$ . At  $10^{-1}$ , each CNOT gate (acting on the compute qubits only) is followed by a two-qubit maximal depolarizing channel. Regardless, the postselected state is noiseless, as predicted by Theorem 1.

from zero to four. Interestingly, at the single-qubit error of 0.1, the two-qubit depolarizing channel is maximally depolarizing, but the fidelity remains at one for the postselected state (as predicted).

The gain in fidelity comes at the cost of a lower probability of measuring all zeros for the ancillas. This trade-off is demonstrated in Fig. 5b. The probability of measuring all zeros  $p(\bar{0})$  drops to around 7% for this circuit at the high single-qubit error of 0.1. Note that the overhead in the number of runs is  $\frac{1}{p(\bar{0})}$ .

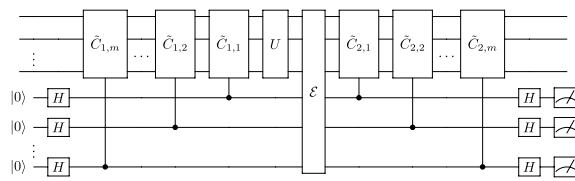
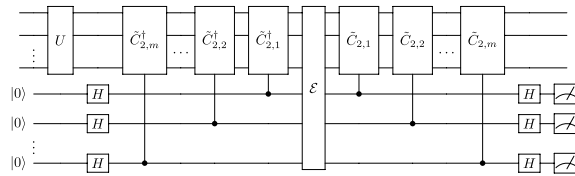
**General errors and hardware considerations.** In the preceding sections, we restricted  $\mathcal{E}$  to CP maps, but our results hold also for general linear Hermitian maps, which includes NCP maps. As previously mentioned, NCP maps play a major role in non-Markovian evolutions, where the maps tend to be non-CP divisible. NCP maps have a similar form to CP maps and are written as  $\mathcal{E}(\rho) = \sum_i \eta_i E_i \rho E_i^\dagger$ , where  $\eta_i = \pm 1$  and there exists at least one  $\eta_i = -1$ . The coefficients  $\eta_i$  are not used in any of our proofs, and thus the results hold.

Also, we restricted  $\mathcal{E}$  to act only on the compute qubits. Obviously this is a restricted case, and in physical systems the checks are noisy and the error map would generally act across all the qubits, as shown in Fig. 6a. In this situation, the checks still conjugate the error, as shown in Fig. 6b. Consequently, the technique is effective when  $\mathcal{E}$  is dominated by Kraus operators that mainly affect the compute qubits; that is, the majority of the noise is from  $U$ .

On non fully connected quantum computers, the parity checks may be difficult to perform with resulting minimal noise on the ancillas due to the need for swapping qubits. Thus, applications of this technique likely need to carefully map the circuit to the hardware to minimize the swaps between ancillas and compute qubits or execute the circuits on a fully connected device.

Since single-qubit gates introduce less noise than nonlocal gates, the Pauli group is a good candidate for the  $\tilde{\mathcal{C}}_2$  part of the checks. Furthermore, when  $U$  is a deep circuit, the noise it induces will generally act across multiple qubits. In this scenario, low-weight  $\tilde{\mathcal{C}}_2$  will act nontrivially on these errors. Thus, in general it is better to use low-weight checks in order to avoid introducing too many errors.



(a)  $\mathcal{E}$  is a general noise map that acts across all qubits.

(b) Using Eq. (19), the error map is still conjugated between the checks.

**Figure 6.** Noisy multilayer scheme.

Moreover, for some executions of this scheme, the postselection probability may be smaller than desired. The postselection probability can be increased by reducing the number of check layers.

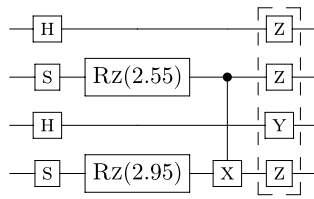
**Protocol for finding checks quickly.** While checks always exist for a given  $U$ , in practice it is difficult to directly compute  $\tilde{C}_1$  from Eq. (19) for a given  $\tilde{C}_2$ . Here we introduce our searching protocol for the *efficient PCS protocol* for determining the check pairs quickly and without matrix multiplication. Note that this protocol can fail to find any checks or may not find the desired number of checks. This can happen when the circuit contains many non-Clifford gates. We refer to the checks searching protocol as the *finding checks protocol*. For our implementation, we constrained  $\tilde{C}_1$  and  $\tilde{C}_2$  to be in the Pauli group. We leave the potential searching protocol of a non-Pauli  $\tilde{C}_1$  for future work.

The goal is to determine the gates comprising  $\tilde{C}_1$  from a given  $\tilde{C}_2 \in \mathcal{P}_n$  and a given  $U$ . Instead of performing matrix multiplication, we transpile the input circuit to an equivalent circuit that uses the gate set  $\{x, y, Rz, s, h, cnot\}$  and perform lookups of the commutation relations. This method applies to circuits consisting of Clifford + arbitrary diagonal gates, which is a universal gate set since diagonal gates contain the gate  $T$ .

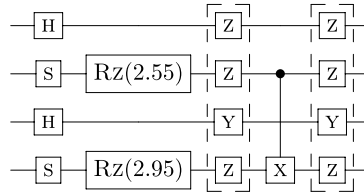
To determine the checks, we use the equality  $U_1 U_2 = U_2 (U_2^\dagger U_1 U_2) = U_2 U'_1$ , where  $U'_1 = U_2^\dagger U_1 U_2$  and  $U_1$  and  $U_2$  are unitary. We refer to this technique as “pushing”  $U_1$  through  $U_2$ . Figure 7 gives a visual example of the pushing of the  $\tilde{C}_2$  gates to determine  $\tilde{C}_1$ . Figure 8 is the completed error-mitigated circuit. This process is efficient since the cost of each lookup call is constant  $O(1)$ .

Algorithm 1 is the pseudocode for the main script for finding a desired number of Pauli check pairs. It iterates through the minimum weight Pauli checks first and terminates when a sufficient number of layers of checks have been found. The protocol focuses on using low weight checks to minimize the noise introduced by the checks as discussed previously in “General errors and hardware considerations”. The main script calls on Alg. 3 to see whether it is possible to push the current gate through. In Alg. 2 the lookup call is a preset table that has commutation relations. This symbolic “pushing” of Pauli gates through  $U$  works for all gates in the basis set except for  $Rz$ . For  $Rz$ , if the gate being pushed is not in  $\{z, I\}$ , which are operators that commute with an arbitrary diagonal gate, then we skip that Pauli group element.

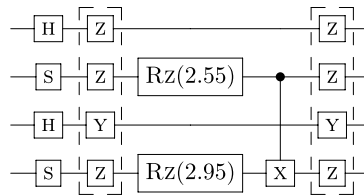
Note that mathematically, any  $\tilde{C}_2 \in \mathcal{P}_n$  can be used because  $\tilde{C}_1$  can be determined from Eq. (11). Thus, one should be able to expand the current algorithm to allow for finding of general  $\tilde{C}_1$ . This problem is nontrivial.



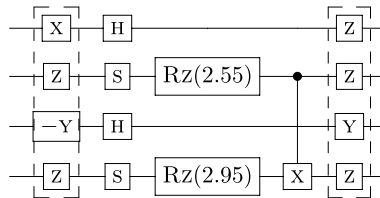
(a) We assign  $ZYZZ$  to  $\tilde{C}_2$



(b) We propagate one layer to the left and determine the intermediate check gate  $ZYZZ$ .

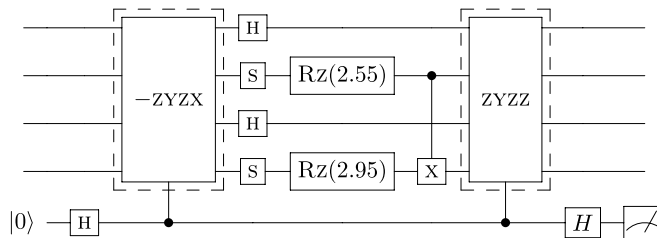


(c) We propagate the intermediate check gate one layer to the left and determine the intermediate check gate  $ZYZZ$ . Notice that the Z gate commutes with Rz.



(d) We propagate through the last layer and assign the result  $-ZYZX$  to the  $\tilde{C}_1$  gate.

**Figure 7.** Visual example of “pushing” the checks through the circuit. We start with  $\tilde{C}_2$  and find  $\tilde{C}_1$ . No multiplication is performed since the propagation of the  $\tilde{C}_2$  gate is determined through lookups of predetermined commutation relations.



**Figure 8.** Final error-mitigated circuit for the example described in Fig. 7.

**Algorithm 1** Main script: find pairs of Pauli checks

---

```

1:  $circ \leftarrow$  quantum circuit
2:  $paulis \leftarrow$  +1 phase Pauli group for N qubits.  $paulis$  is
   sorted by weight from smallest to largest
3:  $c_1$  is initialized to None
4:  $c_2$  is initialized to None
5:  $layersFound \leftarrow 0$ 
6:  $numberLayers \leftarrow$  number of layers to find for  $circ$ .
7: for  $pauli$  in  $paulis$  do
8:   if  $layersFound$  equals  $numberLayers$  then
9:     BREAK  $\triangleright$  Found the necessary number of layers.
10:  end if
11:   $op2 \leftarrow pauli$ 
12:  for  $op1$  and  $index$  in  $circ$  do
13:    if CANCONTINUE( $op1$ ,  $op2$ ) then
14:       $op2 \leftarrow$  PUSH( $op1$ ,  $op2$ )
15:    else
16:      BREAK  $\triangleright$  This check attempt didn't work so
       break out and try the next Pauli.
17:    end if
18:    if  $index$  is the last index then
19:       $layersFound \leftarrow layersFound + 1$ 
20:       $c_1 \leftarrow$  append  $op2$ 
21:       $c_2 \leftarrow$  append  $pauli$ 
22:    end if
23:  end for
24: end for

```

---

**Algorithm 2** Push Pauli gate left

---

```

1: Input:
2:  $op1$ : The current gate that we need to pass through.
3:  $op2$ : The gate being pushed through gate.
4: Output:
5: The gate after it is pushed through  $op1$ . Should be
   equivalent to  $op1^\dagger * op2 * op1$ 
6: function PUSH( $op1$ ,  $op2$ )
7:   if  $op1$  is X then
8:     return lookup  $X(op2)X$ 
9:   else if  $op1$  is Y then
10:    return lookup  $Y(op2)Y$ 
11:   else if  $op1$  is H then
12:    return lookup  $H(op2)H$ 
13:   else if  $op1$  is S then
14:    return lookup  $S^\dagger(op2)S$ 
15:   else if  $op1$  is RZ then
16:    return  $op1 \triangleright op1$  will commute with RZ from the
       previous check performed in the function CanContinue
17:   else if  $op1$  is CNOT then
18:    return lookup  $CX(op2)CX$ 
19:   end if
20: end function

```

---

**Algorithm 3** Checks if it is possible to pass the gate through

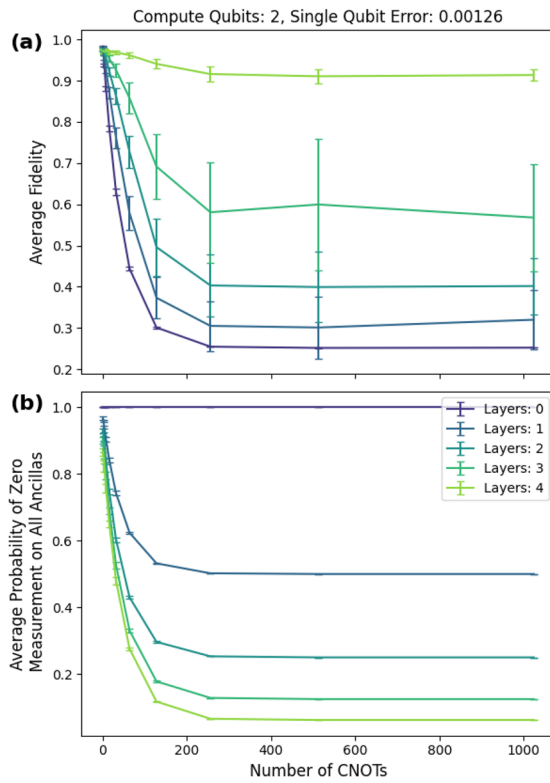
---

```

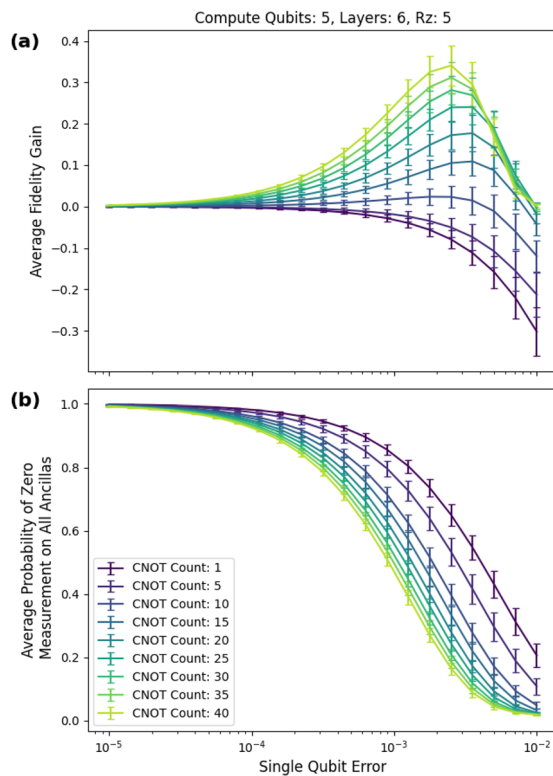
1: Input:
2:  $op1$ : The current gate that we need to pass through.
3:  $op2$ : The gate being pushed through gate.
4: Output:
5: True if the gate can be pushed through and false if not.
6: function CANCONTINUE( $op1$ ,  $op2$ )
7:   return  $op1$  is not RZ or ( $op2$  is I or Z)
8: end function

```

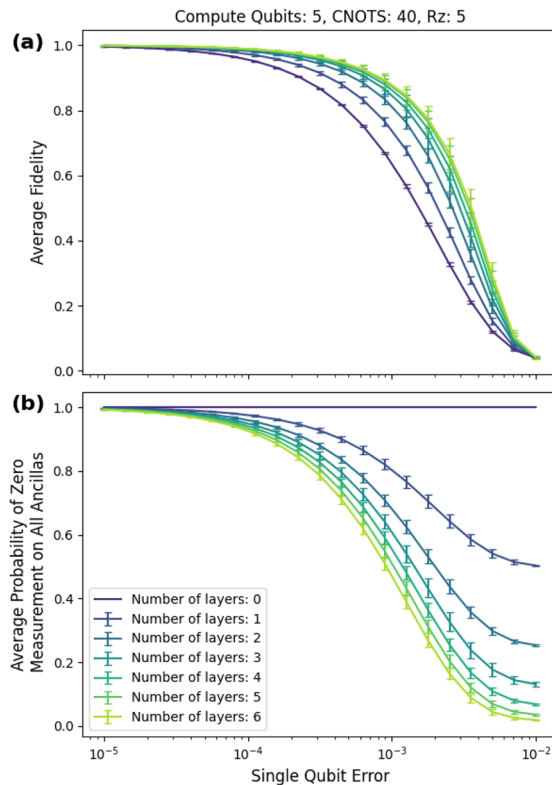
---



**Figure 9.** Two-qubit Clifford simulation with single-qubit error of 0.00126 (0.0126 two-qubit error). Note that while these input circuits can be optimized to use  $O(n^2/\log(n))$  CNOT gates (i.e.,  $O(2)$  CNOT gates), these simulations provide intuition that the protocol is suitable for deep circuits.



**Figure 10.** Six layers. (a) The peak average fidelity gain of 34 percentage points occurred at a single-qubit error of approximately 0.00251 (0.0251 two-qubit error) and 40 CNOTs. (b) The probability of postselecting decreases with increasing error rate and increasing CNOT count.



**Figure 11.** (a) Average fidelity for layers zero to six. At the peak fidelity gain, the nonmitigated circuit has about 33% fidelity, and the six-layer mitigated circuit has about 67% fidelity. (b) Probability of postselecting.

**Numerical results.** The analytical results presented above assume perfect checks. Here we numerically investigate the scheme in a more realistic setting where most gates are noisy, including those involved in the parity checks (the only gates that are not noisy are measurements and the circuit that generates the random input state).

Intuitively, given a Clifford circuit and using the *efficient PCS protocol*, we should be able to perform long computations with high fidelity. For a given Clifford circuit, we can keep the  $C_2$  checks constant and independent of the depth of  $U$ . Thus, the noise induced by our  $C_2$  checks should be relatively constant.

The  $C_1$  checks depend on  $U$ , but they are elements of the Pauli group and hence limited in size and complexity. Therefore, the noise induced by the  $C_1$  checks should also be limited and independent of the depth of  $U$ .

We demonstrate this intuition on simulations consisting of 550 randomly generated Clifford circuits with two compute qubits. Note that these Clifford simulations provide only intuition that the protocol is suitable for deep circuits because Clifford circuits can in general be optimized to use  $O(n^2/\log(n))$  CNOT gates, where  $n$  is the number of qubits<sup>34</sup>. Thus, two-qubit Clifford circuits can be optimized to be shallow. It may be possible to prove this performance on Clifford circuits with higher qubit counts.

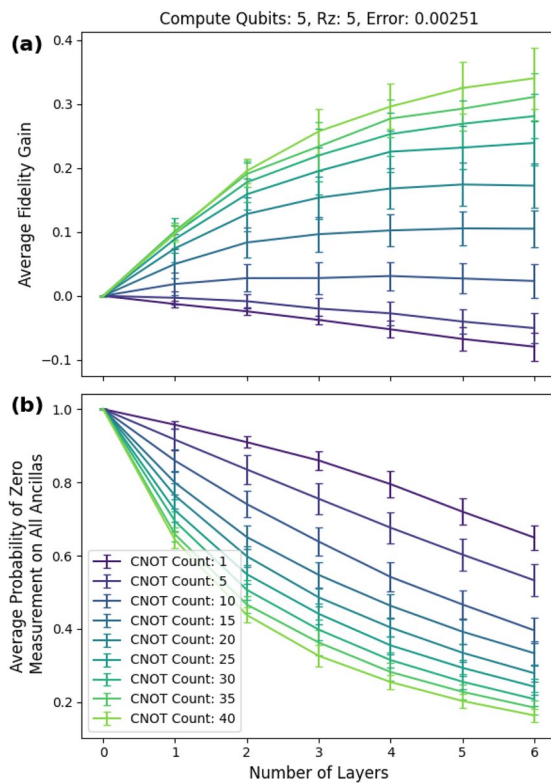
We considered random circuits with CNOT counts that varied from 1, 2, 4,  $\dots$ , 1,024. For each CNOT count we generated 50 random circuits, and we used single-qubit depolarizing noise of 0.0126 (0.0126 two-qubit noise). This lies within the range of current noise levels found in state of the art quantum computers<sup>29,35</sup>. We used four layers of checks; the form of the checks was provided in Proposition 2. As shown in Fig. 9a, we maintained an average fidelity  $F_m$  for the postselected state of greater than 90% for circuits consisting of up to 1,024 CNOT gates. The average fidelity of the unmitigated circuits drops to 25% at 256 CNOTs. Note that this comes at the cost of a lower postselection rate of 6.25% as shown in Fig. 9b.

For optimized Clifford circuits, we would likely not want to use all the checks from Proposition 2 because we would probably exceed the CNOT count of the input circuit. Still, as shown in Fig. 9a, fewer layers can produce significant fidelity improvement.

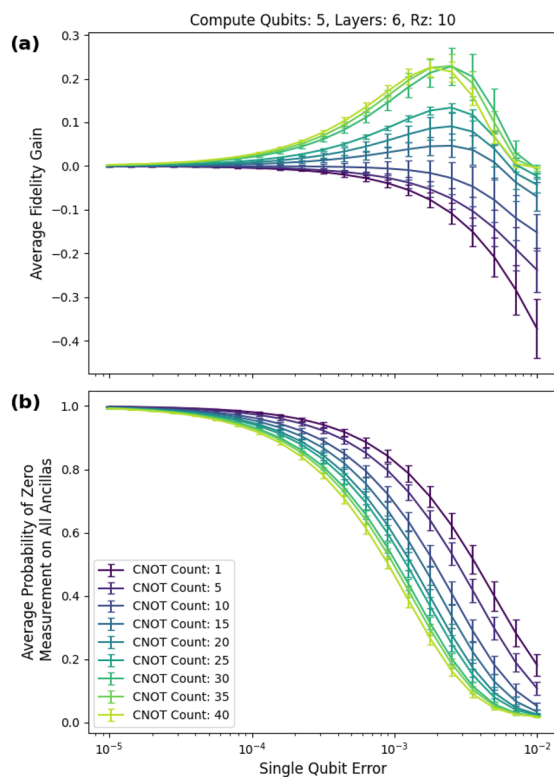
These simulations establish the general trend that fidelity is positively correlated with the number of layers up to some value. We suspect that these results also hold for general (non-Clifford) circuits.

We also randomly generated 1850 input circuits consisting of Clifford + arbitrary diagonal unitary gates. Of these, 1,350 input circuits consist of five qubits with CNOT counts of  $\{1, 5, \dots, 40\}$ ; 500 input circuits consist of ten qubits with CNOT counts of  $\{1, 5, \dots, 40, 80\}$ . We varied the single-qubit error from  $10^{-5}$  to  $10^{-2}$  with 21 equally spaced points in log scale.

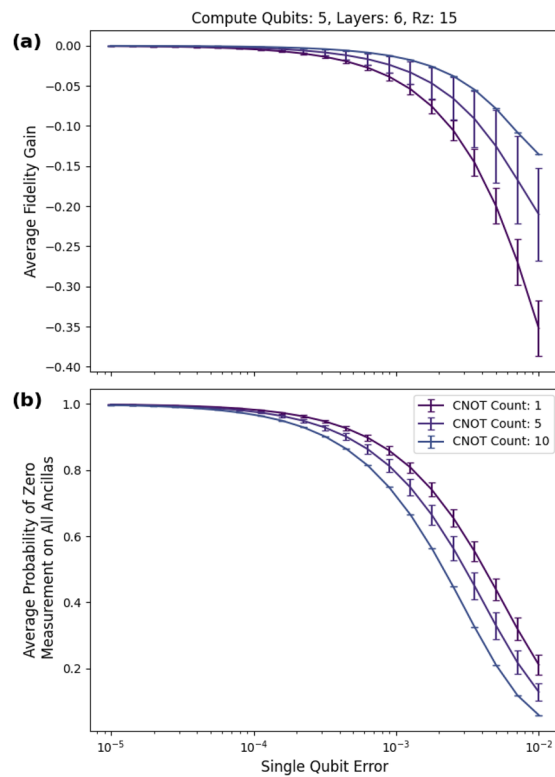
For the ten-qubit circuits we also generated circuits with CNOT gate counts of 80 to match the max CNOT count to qubit ratio of the five qubit case. Each random circuit was generated first as a random Clifford gate, which we truncated to reach the desired CNOT count. Next, we inserted RZ gates with random rotation angles



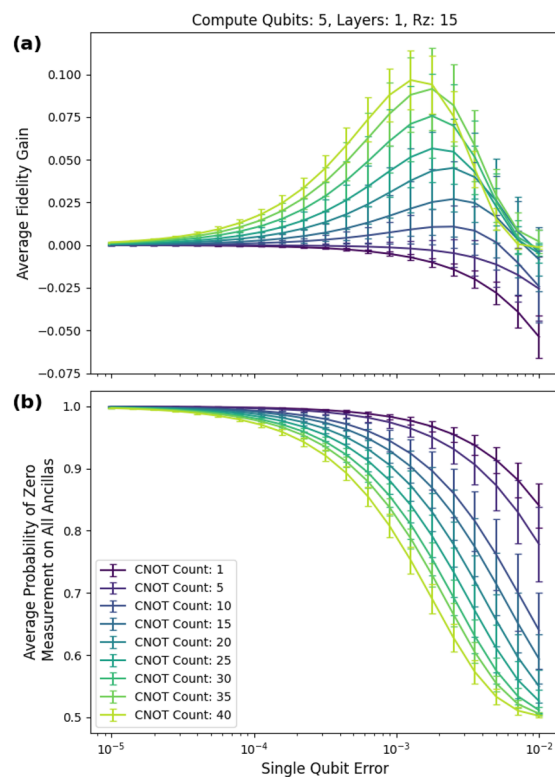
**Figure 12.** (a) Average fidelity gain vs number of layers. (b) Probability of postselecting vs number of layers. The single-qubit error is fixed at approximately 0.00251 (0.0251 two-qubit error).



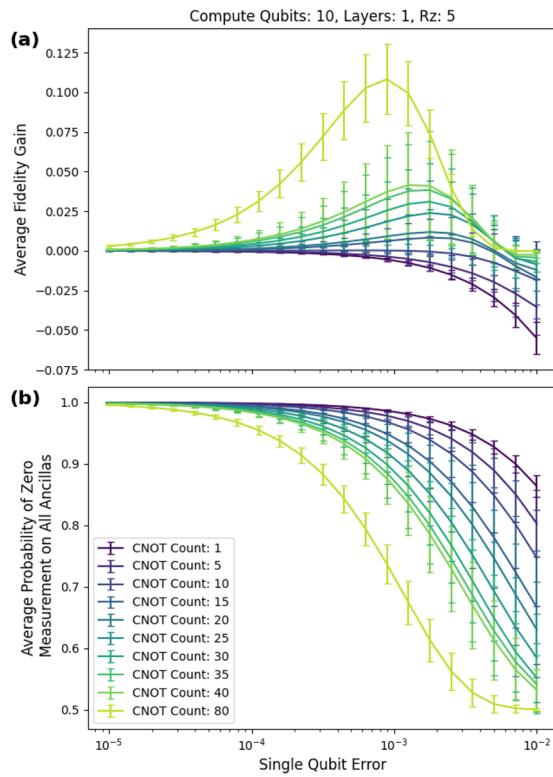
**Figure 13.** Five-qubit circuits with 10 RZ gates. (a) The max fidelity gain is about 25 percentage points. (b) Probability of postselecting vs single-qubit error.



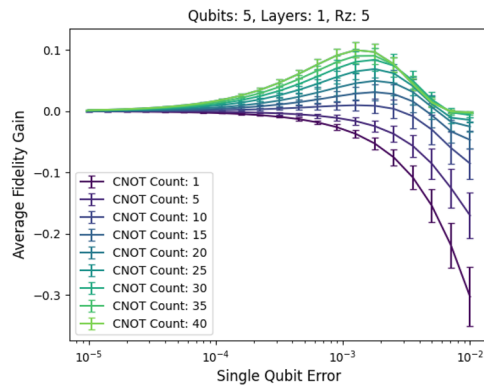
**Figure 14.** Five-qubit circuits with 15 RZ gates. After 10 CNOT gates, we cannot find circuits with six layers.



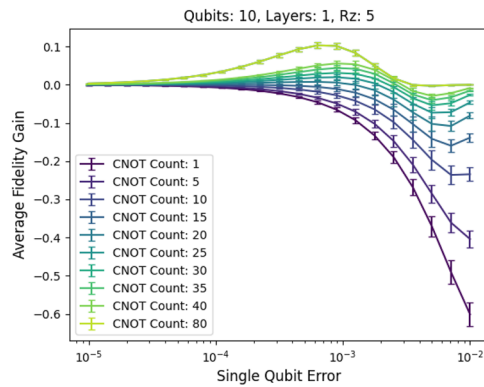
**Figure 15.** Five-qubit circuits with 15 RZ gates. (a) At one layer of checks, the peak fidelity gain is about 10 percentage points.



**Figure 16.** Ten-qubit circuits with 5 RZ gates. We used a single layer of low weight checks. The 80 CNOT count case matches the CNOT count to qubit ratio of the five-qubit case with 40 CNOT gates. (a) The peak fidelity gain is about ten percentage points. It occurs at a single-qubit error of about 0.000891 (0.00891 two-qubit error).



(a) One layer using max weight checks for five compute qubits.



(b) One layer using max weight checks for ten compute qubits.

**Figure 17.** Max weight checks. The high-weight checks introduce a lot of noise compared with the low-weight method, as shown in the large negative fidelity at a high single-qubit error rate.



and random locations in the circuit. We used RZ gate counts of {5, 10, 15}. Each RZ value for five qubits consists of 450 circuits. This covers a large class of variational quantum eigensolver and quantum approximate optimization algorithm circuits<sup>36</sup>.

We achieved an average peak fidelity gain  $F_m - F_n$  of 34 percentage points for five-qubit circuits with a CNOT gate count of 40, five RZ gates, and six layers of checks, as shown in Fig. 10a. For input circuits with a low CNOT count, the fidelity gain is negative because the checks introduce more errors than they eliminate in the post selected state. The average postselection probability is given in Fig. 10b.

We also give in Fig. 11a a plot that breaks down this peak fidelity gain. At the peak fidelity gain, the nonmitigated circuit has about 33% fidelity, and the six-layer mitigated circuit has about 67% fidelity.

As shown in Fig. 11a, the mitigated circuits perform significantly and consistently better than the unmitigated circuits. Even for lower-layer counts such as two, the average fidelity gain reached 20 percentage points. Fig. 11b gives the corresponding post selection probabilities and demonstrates that we have significant control over the probabilities by changing the number of layers.

Each additional layer increased the average fidelity provided enough circuit depth. We show this in more detail in Fig. 12a, where we fixed the single-qubit error rate to 0.00251 (0.0251 two-qubit error) the value that gave the peak fidelity gain in Fig. 10a.

Circuits with more than six layers may result in even better performance, but the amount of fidelity gained decreases with subsequent layers. Figure 12b shows the corresponding post selection probabilities and the minimum post selection probability is about 16%.

As the number of RZ (non-Clifford) gates increases, the number of possible low weight  $\tilde{C}_2$  checks for the *efficient PCS protocol* decreases, and consequently the fidelity gain decreases. As shown in Fig. 13a, at an RZ gate count of 10, the peak fidelity gain is about 25%.

As shown in Fig. 14a, at an RZ gate count of 15, we cannot find six layers of checks for random circuits with 20 CNOT gates or higher. Interestingly, as shown in Figs. 10b, 13b and 14b, the post selection curves are relatively unchanged.

Using one layer of checks, we have a peak fidelity gain of about 10 percentage points at 40 CNOT gates, as shown in Fig. 15a. Figure 15b shows the corresponding post selection probabilities.

For the ten-qubit case, as shown in Fig. 16a, we achieved a fidelity gain of about ten percentage points. This occurred at a CNOT count to qubit ratio of eight, which matches the scenario of the peak fidelity gain in the five-qubit case. The peak fidelity gain occurred at a single-qubit error of about 0.000891 (0.00891 two-qubit error). Figure 16b shows the corresponding post selection probabilities.

The preceding simulations focus on using low-weight checks first. We now analyze the performance of high-weight checks. As shown in Fig. 17a,b, while the high-weight checks do give a boost in fidelity, they introduce significant amounts of noise compared to the low-weight checks.

## Conclusions

The quantum error mitigation technique we have studied in this work is novel because (1) it has an adjustable quantum overhead for any input circuit, (2) by adjusting the number of layers of check operators, the technique allows controlling of the post-selection probability and the error from the error mitigation protocol, (3) the method can be applied repeatedly and at any location in the circuit, and works for arbitrary input states, and (4) in the setting of Theorem 1, we prove that we can achieve unit fidelity provided that we use a sufficient number of layers.

We prove in Theorem 1 that if the error is restricted to the compute qubits (see Fig. 4a), there exist checks such that the fidelity for the postselected state reaches unity. We also give a small number of  $\tilde{C}_2$  checks that reach unit fidelity in this scenario in Propositions 1 and 2.

In Eq. (19),  $\tilde{C}_2$  is chosen and  $\tilde{C}_1$  can be directly determined through our *finding checks protocol* given in “Numerical results”. This algorithm determines the pairs of checks without matrix multiplication. Instead, we perform lookups of predetermined commutation relations. One limitation of our *finding checks protocol* is that we are able to find only  $\tilde{C}_1$  that are in the Pauli group. This limitation does not exist for the *general PCS protocol*.

The main limitation of the proposed approach is the need to obtain the checks  $\tilde{C}_1$  and  $\tilde{C}_2$ , with cost exponential in the number of qubits in the subcircuit. This cost can be reduced to exponential in the number of non-Clifford gates (and only polynomial in the number of qubits) by leveraging the extended stabilizer formalism<sup>37</sup>.

The performance of the protocol is tested through extensive numerical simulations on random circuits consisting of 550 Clifford and 1850 non-Clifford circuits. We used the Clifford simulations to provide intuition that the technique is suitable for deep circuits.

For the non-Clifford circuits, we used five- and ten-qubit circuits. We use the difference between the fidelity of the mitigated circuit and the fidelity of the unmitigated circuit as a figure of merit. Under depolarizing noise, the simulations reached an average fidelity gain of 34 percentage points for circuits consisting of five qubits, 40 CNOTs, and six low-weight  $\tilde{C}_2$  checks (see Figs. 10a and 10b). It is possible that more layers will provide further boosts in fidelity. The single-qubit noise ranged from  $10^{-5}$  to  $10^{-1}$ . This coincides with current noise levels found in superconducting quantum computers<sup>35</sup>.

In<sup>38</sup>, the authors derive an error mitigation scheme based on symmetry verification, which they call the spatio-temporal stabilizer (STS) technique. The STS technique shares many similarities with the PCS scheme, as first introduced in<sup>20</sup>, and when there is only one pair of checks, STS is the PCS scheme. An important difference is that when there are multiple pairs of checks, layers are allowed to be partly nested in the STS technique. For example, a possible STS execution is layer one and layer two act on the same compute qubits, but layer two begins before layer one has ended and layer one ends before layer two. Since the STS method also allows the standard layering of checks in PCS, our results also hold for the STS technique.

We also note that while the results of this research are presented in the context of quantum computing, the theoretical results hold in general for settings where the user intends to implement an ideal known unitary  $U$  on a quantum state. This follows because we placed no restrictions on the unitary. The performance of the scheme in other settings needs to be investigated. Also, since the protocol places no restriction on the input state, one can apply the mitigation technique on subcircuits and easily combine it with other methods. Splitting a large circuit into subcircuits for finding checks or combining the protocol with other techniques have not been studied. Determining the optimal number of check layers also needs to be further investigated.

Moreover, the best type of checks to use may be non-Pauli in the *general PCS protocol*. This is likely true given some knowledge of the dominant noise. One potential line of investigation is to use the controls in dynamical decoupling protocols as the  $\tilde{C}_2$  parity checks<sup>9,39</sup>.

### Data availability

The data presented in this paper is available online at [https://github.com/alvinquantum/noise\\_mitigation\\_symmetry](https://github.com/alvinquantum/noise_mitigation_symmetry).

### Code availability

The code used for numerical experiments in this work is available online at [https://github.com/alvinquantum/noise\\_mitigation\\_symmetry](https://github.com/alvinquantum/noise_mitigation_symmetry).

Received: 4 November 2022; Accepted: 12 January 2023

Published online: 06 February 2023

### References

- Knill, E., Laflamme, R. & Zurek, W. H. Resilient quantum computation: Error models and thresholds. *Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci.* **454**(1969), 365–384 (1998).
- Aharonov, D. & Ben-Or, M. Fault-tolerant quantum computation with constant error rate. arXiv preprint [arXiv:quant-ph/9906129](https://arxiv.org/abs/quant-ph/9906129) (1999).
- Kitaev, A. Y. *Quantum Error Correction with Imperfect Gates* 181–188 (Springer US, 1997).
- Temme, K., Bravyi, S. & Gambetta, J. M. Error mitigation for short-depth quantum circuits. *Phys. Rev. Lett.* **119**, 180509 (2017).
- Kandala, A. *et al.* Error mitigation extends the computational reach of a noisy quantum processor. *Nature* **567**(7749), 491–495 (2019).
- Giurgica-Tiron, T., Hindy, Y., LaRose, R., Mari, A. & Zeng, W. J. Digital zero noise extrapolation for quantum error mitigation. In *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*, 306–316, (2020)
- Viola, L. & Lloyd, S. Dynamical suppression of decoherence in two-state quantum systems. *Phys. Rev. A* **58**, 2733–2744 (1998).
- Byrd, M. S. & Lidar, D. A. Empirical determination of dynamical decoupling operations. *Phys. Rev. A* **67**, 012324 (2003).
- Tripathi, V. *et al.* Suppression of crosstalk in superconducting qubits using dynamical decoupling. arXiv preprint [arXiv:2108.04530](https://arxiv.org/abs/2108.04530) (2021).
- Nachman, B., Urbaneck, M., de Jong, W. A. & Bauer, C. W. Unfolding quantum computer readout noise. *npj Quant. Inf.* **6**(1), 84 (2020).
- Bonet-Monroig, X., Sagastizabal, R., Singh, M. & O'Brien, T. E. Low-cost error mitigation by symmetry verification. *Phys. Rev. A* **98**, 062339 (2018).
- McArdle, S., Yuan, X. & Benjamin, S. Error-mitigated digital quantum simulation. *Phys. Rev. Lett.* **122**(18), 180501 (2019).
- Shaydulin, R. & Galda, A. Error mitigation for deep quantum optimization circuits by leveraging problem symmetries. arXiv preprint [arXiv:2106.04410](https://arxiv.org/abs/2106.04410) (2021).
- Cai, Z. Quantum error mitigation using symmetry expansion. *Quantum* **5**, 548 (2021).
- Peruzzo, A. *et al.* A variational eigenvalue solver on a photonic quantum processor. *Nat. Commun.* **5**(1), 4213 (2014).
- Preskill, J. Quantum computing in the NISQ era and beyond. *Quantum* **2**, 79 (2018).
- Gambetta, J., Faro, I. & Wehden, K. IBM's roadmap for building an open quantum software ecosystem. <https://research.ibm.com/blog/quantum-development-roadmap> (Accessed 18 Apr 2022) (2021)
- Chapman, P. Scaling IonQ's quantum computers: The roadmap. <https://ionq.com/posts/december-09-2020-scaling-quantum-computer-roadmap> (Accessed 18 Apr 2022) (2020).
- Chuang Isaac, L. & Nielsen, M. A. Prescription for experimental determination of the dynamics of a quantum black box. *J. Mod. Opt.* **44**(11–12), 2455–2467 (1997).
- Debrooy, D. M. & Brown, K. R. Extended flag gadgets for low-overhead circuit verification. *Phys. Rev. A* **102**(5), 052409 (2020).
- Chao, R. & Reichardt, B. W. Quantum error correction with only two extra qubits. *Phys. Rev. Lett.* **121**, 050502 (2018).
- Zhou, H. & Byrd, G. T. Quantum circuits for dynamic runtime assertions in quantum computation. *IEEE Comput. Architect. Lett.* **18**(2), 111–114 (2019).
- Nielsen, M. A. & Chuang, I. L. *Quantum Computation and Quantum Information* (Cambridge University Press, 2011).
- Sudarshan, E. C. G., Mathews, P. M. & Rau, J. Stochastic dynamics of quantum-mechanical systems. *Phys. Rev.* **121**, 920–924 (1961).
- Choi, M.-D. Completely positive linear maps on complex matrices. *Linear Algebra Appl.* **10**(3), 285–290 (1975).
- Kraus, K., Böhm, A., Dollard, J. D. & Wootters, W. H. *States, Effects, and Operations: Fundamental Notions of Quantum Theory, Lecture Notes in Physics* (Springer, 1983).
- Milz, S., Kim, M. S., Pollock, F. A. & Modi, K. Completely positive divisibility does not mean Markovianity. *Phys. Rev. Lett.* **123**, 040401 (2019).
- Patel, T., Potharaju, A., Li, B., Roy, R. B. & Tiwari, D. Experimental evaluation of nisq quantum computers: Error measurement, characterization, and implications. In *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis* 1–15 (2020).
- Zhang, S. *et al.* Error-mitigated quantum gates exceeding physical fidelities in a trapped-ion system. *Nat. Commun.* **11**(1), 587 (2020).
- Cai, Z. & Benjamin, S. C. Constructing smaller Pauli twirling sets for arbitrary error channels. *Sci. Rep.* **9**(1), 11281 (2019).
- Bennett, C. H., DiVincenzo, D. P., Smolin, J. A. & Wootters, W. K. Mixed-state entanglement and quantum error correction. *Phys. Rev. A* **54**, 3824–3851 (1996).
- Bennett, C. H. *et al.* Purification of noisy entanglement and faithful teleportation via noisy channels. *Phys. Rev. Lett.* **76**, 722–725 (1996).
- Shor, P. W. Scheme for reducing decoherence in quantum computer memory. *Phys. Rev. A* **52**, R2493–R2496 (1995).
- Patel, K. N., Markov, I. L. & Hayes, J. P. Optimal synthesis of linear reversible circuits. *Quantum Inf. Comput.* **8**(3), 282–294 (2008).

35. Maldonado, T. J., Flick, J., Krastanov, S. & Galda, A. Error rate reduction of single-qubit gates via noise-aware decomposition into native gates. *Sci. Rep.* **12**(1), 6379 (2022).
36. Farhi, E., Goldstone, J. & Gutmann, S. A quantum approximate optimization algorithm. arXiv preprint [arXiv:1411.4028](https://arxiv.org/abs/1411.4028) (2014).
37. Bravyi, S. *et al.* Simulation of quantum circuits by low-rank stabilizer decompositions. *Quantum* **3**, 181 (2019).
38. Xiong, Y., Chandra, D., Ng, S. Xin. & Lajos, H. Circuit symmetry verification mitigates quantum-domain impairments. arXiv preprint [arXiv:2112.13904](https://arxiv.org/abs/2112.13904) (2021).
39. Khodjasteh, K. & Lidar, D. A. Performance of deterministic dynamical decoupling schemes: Concatenated and periodic pulse sequences. *Phys. Rev. A* **75**(6), 062310 (2007).

## Acknowledgements

This research was supported in part by an appointment to the Intelligence Community Postdoctoral Research Fellowship Program at Argonne National Laboratory, administered by Oak Ridge Institute for Science and Education through an interagency agreement between the U.S. Department of Energy and the Office of the Director of National Intelligence. This work was supported in part by the U.S. Department of Energy (DOE), Office of Science, National Quantum Information Science Research Centers, Office of Advanced Scientific Computing Research AIDE-QC and FAR-QC projects, and by the Argonne LDRD program under contract number DE-AC02-06CH11357. Z.S. thanks Q-NEXT for supporting this project. We gratefully acknowledge the computing resources provided on Bebop, a high-performance computing cluster operated by the Laboratory Computing Resource Center at Argonne National Laboratory.

## Disclaimer

This paper was prepared for information purposes with contributions from the Global Technology Applied Research group of JPMorgan Chase. This paper is not a product of the Research Department of JPMorgan Chase or its affiliates. Neither JPMorgan Chase nor any of its affiliates make any explicit or implied representation or warranty and none of them accept any liability in connection with this paper, including, but not limited to, the completeness, accuracy, reliability of information contained herein and the potential legal, compliance, tax or accounting effects thereof. This document is not intended as investment research or investment advice, or a recommendation, offer or solicitation for the purchase or sale of any security, financial instrument, financial product or service, or to be used in any way for evaluating the merits of participating in any transaction.

## Author contributions

A.G. performed the theoretical analysis with feedback from R.S. and Z.H.S. R.S. developed the “finding checks quickly” protocol. Z.H.S. suggested using the Pauli expansion of Kraus operators in the theory. A.G. wrote the code, performed the numerical simulations, and wrote the manuscript with input from R.S., Z.H.S., and M.S. R.S. structured multiple sections of the manuscript. R.S., Z.H.S., and M.S. developed the numerical experiments with input from A.G. R.S., M.S., A.G., and Z.H.S. found numerical optimizations.

## Funding

The submitted manuscript has been created by UChicago Argonne, LLC, Operator of Argonne National Laboratory (“Argonne”). Argonne, a U.S. Department of Energy Office of Science laboratory, is operated under Contract No. DE-AC02-06CH11357. The U.S. Government retains for itself, and others acting on its behalf, a paid-up nonexclusive, irrevocable worldwide license in said article to reproduce, prepare derivative works, distribute copies to the public, and perform publicly and display publicly, by or on behalf of the Government. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan. <http://energy.gov/downloads/doe-public-access-plan>.

## Competing interests

The authors declare no competing interests.

## Additional information

**Correspondence** and requests for materials should be addressed to A.G.

**Reprints and permissions information** is available at [www.nature.com/reprints](http://www.nature.com/reprints).

**Publisher’s note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2023