Quantum Error Mitigation as a Universal Error Reduction Technique: Applications from the NISQ to the Fault-Tolerant Quantum Computing Eras

Yasunari Suzuki,[1,2,*,§] Suguru Endo[1,2,†,§] Keisuke Fujii,[3,4,5] and Yuuki Tokunaga[1,‡]

[1] *NTT Computer and Data Science Laboratories, NTT Corporation, Musashino 180-8585, Japan*

[2] *JST, PRESTO, 4-1-8 Honcho, Kawaguchi, Saitama 332-0012, Japan*

[3] *Graduate School of Engineering Science, Osaka University, 1-3 Machikaneyama, Toyonaka, Osaka 560-8531, Japan*

[4] *Center for Quantum Information and Quantum Biology, Institute for Open and Transdisciplinary Research Initiatives, Osaka University, Japan*

[5] *Center for Emergent Matter Science, RIKEN, Wako, Saitama 351-0198, Japan*

In the early years of fault-tolerant quantum computing (FTQC), it is expected that the available code distance and the number of magic states will be restricted due to the limited scalability of quantum devices and the insufficient computational power of classical decoding units. Here, we integrate quantum error correction and quantum error mitigation into an efficient FTQC architecture that effectively increases the code distance and *T*-gate count at the cost of constant sampling overheads in a wide range of quantum computing regimes. For example, while we need 10^4 to 10^{10} logical operations for demonstrating quantum advantages from optimistic and pessimistic points of view, we show that we can reduce the required number of physical qubits by 80% and 45% in each regime. From another perspective, when the achievable code distance is up to about 11, our scheme allows executing 10^3 times more logical operations. This scheme will dramatically alleviate the required computational overheads and hasten the arrival of the FTQC era.

I. INTRODUCTION

Quantum computers are believed to be capable of implementing several tasks such as factoring and Hamiltonian simulations, in exponentially smaller computational times than those of classical computers [1,2]. However, quantum systems generally interact with their environments, which leads to physical errors in the system that may destroy their quantum advantages. Since the physical error rates of quantum computers are still much higher than those of classical computers, it is vital to suppress these errors. As a solution, fault-tolerant quantum computing (FTQC) using quantum error-correcting codes has been studied

[3–7]. The long-term FTQC allows executing conventional quantum algorithms such as Hamiltonian simulation algorithms [8]. According to the current state-of-the-art resource estimations [9,10], the logical quantum operation count will be in the order of 10^{10} to observe clear quantum advantages based on the computational complexity theory.

Towards the realization of the long-term FTQC, we experience several intermediate regimes as shown in Fig. 1 because high-level encoding is not allowed due to restrictions of quantum resources such as qubit and magic state count [5,7]. Since quantum error correction (QEC) requires massive classical computation for repetitive error estimations, the available code distance would also be strongly limited in the near future [15–17]. As quantum technologies become mature, computational quantum supremacy [13] will be achieved in the logical space. We refer to the intermediate regime from the realization of logical quantum supremacy to the demonstration of long-term applications as an early FTQC regime. The number of physical qubits will go beyond one thousand in this region, and we anticipate that more than about 10^4 reliable logical operations on 10^2 logical qubits are available. Even at the beginning of the early FTQC regime, we may observe

_____

* yasunari.suzuki.gz@hco.ntt.co.jp

† suguru.endou.uc@hco.ntt.co.jp

‡ yuuki.tokunaga.bf@hco.ntt.co.jp

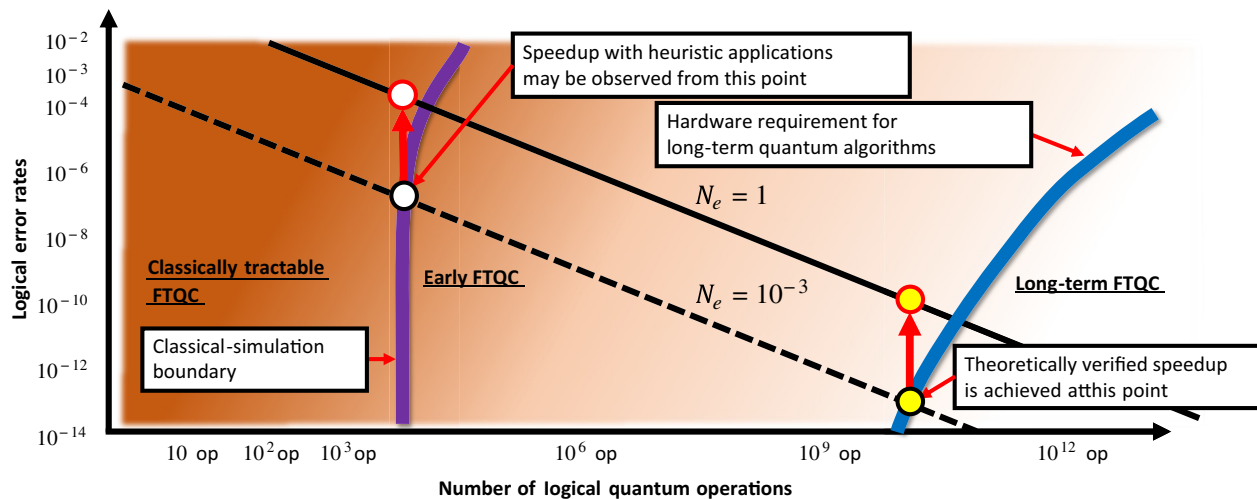§ Y.S and S.E contributed to this work equally.

FIG. 1. Schematic picture representing the transitional period from the classically tractable FTQC regime towards the realization of the long-term FTQC. We are now in the classically tractable FTQC era due to the lack of physical qubits [11,12]. In the figure, the purple line indicates the hardware requirement for performing classically intractable tasks with a realistic time whereas the blue line corresponds to the requirement for demonstrating quantum advantages with conventional long-term quantum algorithms. To estimate these lines, we refer to the quantum supremacy experiments [13] and the existing state-of-the-art resource estimation [9,10,14]. The early FTQC regime is defined as a region between these lines. In the main text, we assume that the number of error events during FTQC N_e is required to be smaller than 10^{-3}, which is shown as the dotted black line. Our technique allows for FTQCs with the number of error events in the order of unity $N_e \sim 1$, which is shown as a solid black line, to execute applications that originally require a much smaller error-event count. For example, at the beginning and the end of the early FTQC regime, our technique allows simulating applications (white and yellow circles with black rims) with the relaxed hardware requirement (white and yellow circles with red rims).

a quantum speedup with heuristic quantum algorithms, for example, with the variational quantum eigensolver [18–20].

In this paper, to realize efficient and high-accuracy quantum computation in the early FTQC era, we propose a novel framework of FTQC, where QEC and quantum error mitigation (QEM) are combined on an equal footing. While QEM has been considered to be an alternative error-minimization technique for noisy intermediate-scale quantum (NISQ) devices due to its low hardware overhead at the expense of the sampling cost, we show that, by integrating probabilistic error cancellation [21,22] into the FTQC framework, we can mitigate all the dominant types of errors in the logical space. We also note that our scheme can efficiently mitigate Pauli errors by virtually updating the quantum states with a classical memory called the Pauli frame [5]. In the conventional QEM formalism, the sampling cost of QEM increases exponentially with the number of physical error events [23,24]. Therefore, the sampling overheads of QEM become unrealistic in NISQ computing when the number of physical operations increases for a fixed error rate per quantum gate; and the number of error events that QEM can efficiently suppress is limited to the order of unity. In our framework, the sampling cost of QEM increases exponentially with the number of *logical error events* in the encoded space. Note that we can tune the number of logical error events by adjusting several parameters such as the code

distance, distillation levels, and precision of approximations for Solovay-Kitaev decomposition. Thus, it is highly likely that we can find regions where the QEM techniques are the most effective, i.e., the number of logical error events is the order of unity. Accordingly, we can relax the hardware requirement with constant sampling overheads. Even after the scalable FTQC is realized, taking QEM into account, we can optimize quantum computation by allocating computation resources at will to perform even more efficient quantum computing.

We need to overcome several fundamental difficulties for applying QEM in the logical space because the costs and restrictions of logical operations and dominant sources of errors are different from the NISQ formalism. We resolve them in the affirmative by giving a solution one by one. For example, solutions to major problems are as follows. In FTQC, logical Clifford operations and Pauli measurements can be efficiently applied while non-Clifford operations are costly because it involves a number of T-gate injection, distillation, and teleportation procedures [5,7]. These logical operations are affected by three types of logical errors: logical errors in each elementary gate operation due to restricted code distances, noise in non-Clifford logical gates deriving from shortage of magic state distillation processes, and errors induced in the Solovay-Kitaev decomposition [25,26]. We call the first two logical errors decoding errors and the last one approximation errors. We discuss what types of errors are present

when implementing logical operations, and provide a hierarchical way to mitigate noisy and costly operations with clean and less costly ones. To detect and correct physical errors during computation, we store the estimated errors in the Pauli frame instead of physically applying recovery operations [5]. This means that actual physical states are almost never in the code space. We provide concrete procedures for a universal set of logical operations incorporating QEM, which are compatible with the Pauli frame. To apply probabilistic error cancellation, we need a good characterization of the noise model to construct QEM operations. We show that decoding errors can be efficiently characterized with gate-set tomography [27,28] on the code space. Note that the approximation errors of the Solovay-Kitaev algorithm can be characterized efficiently on classical computers. Finally, while probabilistic error cancellation is a QEM technique to mitigate errors in the algorithms for calculating the expectation values, many FTQC algorithms are sampling algorithms using the phase estimation [9,10,29]. We show that probabilistic error cancellation is compatible with the phase estimation algorithm. See Appendix H for details.

We perform resource estimation of FTQC under realistic scenarios with and without QEM, and we show that our scheme can dramatically alleviate the required computational overheads in FTQC. We assume that the mean number of logical error events N_e is required to reach $N_e = 10^{-3}$, and the sampling overhead by QEM is restricted to a reasonable level, i.e., within 10^2 times greater samples for achieving a certain accuracy. We expect at least 10^4 logical operations are required to demonstrate classically intractable applications. In this case, the required number of qubits is reduced to approximately one fifth with QEM compared to the original qubit count. We also expect that 10^{10} logical operations are at least necessary to perform conventional long-term applications. The required number of qubits is reduced to 55% in this regime. From another perspective, our scheme can be used for increasing the number of available logical operations when the available code distance is strongly restricted. The lifetime of current superconducting qubits is about up to 1 ms, and a cycle of error estimations during FTQC must be sufficiently faster than the lifetime, i.e., about 1 μs [15,30]. To cope with this strong restriction, an efficient implementation of classical error-decoding architectures has been studied. According to the recent state-of-the-art proposals [15–17], the available code distance would be limited up to about 11 in the near future even with simplified decoding algorithms. When the available code distance is limited up to 11, our scheme enables 10^3 times more logical operations with the same hardware requirement. Thus, our technique can clearly accelerate the realization of applications in early and long-term FTQC regimes. This improvement is illustrated by red arrows in Fig. 1. It is also worth noting that, to the best of our knowledge, these are

the first examples where the performance of useful quantum algorithms with clear quantum advantages is enhanced via QEM under realistic conditions since QEM has been investigated for near-term heuristic quantum algorithms dependent on numerical optimization.

This paper is organized as follows. In Sec. II, we review probabilistic error cancellation and the architecture of fault-tolerant quantum computing. In Sec. III, we describe how to evaluate decoding errors and approximation errors. Then we show our novel FTQC architecture with an analytical argument of the cost of QEM and explain the effect of model estimation errors. In Sec. IV, we numerically analyze the sampling cost of QEM for decoding errors and approximation errors and demonstrate that we can effectively increase the code distance and the number of T gates via QEM even when there are finite estimation errors. Finally, we conclude our paper with a discussion in Sec. V.

II. PRELIMINARIES

A. Quantum error mitigation and probabilistic error cancellation

Quantum processors are affected by a number of physical noise sources, which should be mitigated to obtain correct results. Here, for simplicity, we assume that the gate errors are Markovian, i.e., the noise process $\mathcal{N}$ for a gate is totally independent of other gate errors. In this case, we have

$$\rho_{\text{out}} = \mathcal{N}_{N_G} \circ \mathcal{U}_{N_G} \circ \mathcal{N}_{N_G-1} \circ \mathcal{U}_{N_G-1} \cdots \mathcal{N}_1 \circ \mathcal{U}_1(\rho_{\text{in}}),$$

$$(1)$$

where ρ_{out} and ρ_{in} are the output and input quantum states, $\mathcal{U}_k$ and $\mathcal{N}_k$ denote the ideal and noisy part of the process of the kth gate, and N_G is the number of gates. To ensure correct computations, it is necessary to mitigate the effect of $\mathcal{N}_k, (k = 1, 2, \ldots, N_G)$ and obtain

$$\rho_{\text{out}}^{\text{ideal}} = \mathcal{U}_{N_G} \circ \mathcal{U}_{N_G-1} \cdots \circ \mathcal{U}_1(\rho_{\text{in}}). \quad (2)$$

QEM has been proposed as a method for suppressing errors without encoding, and it is useful especially for NISQ devices with a restricted number of qubits [21,22,31]. Generally speaking, QEM methods recover not the ideal density matrix $\rho_{\text{out}}^{\text{ideal}}$ itself, but rather the ideal expectation value of an observable $\langle \hat{M} \rangle_{\text{ideal}} = \text{Tr}(\rho_{\text{out}}^{\text{ideal}} \hat{M})$ via classical postprocessing. Note that QEM is not a scalable technique because it needs exponentially increasing circuit runs with the number of error events in the quantum circuit [21,22].

Now let us explain the concept of probabilistic error cancellation with which we can eliminate a bias from the expectation value of the observables completely given the complete information on the noise model [21,22]. (Later, we use this method to suppress errors in FTQC.) First, we identify the noise map $\mathcal{N}$ via either process or gate-set

tomography [27,28], and calculate the inverse $\mathcal{N}^{-1}$. Then, by finding a set of processes $\{\mathcal{B}_i\}$ such that $\mathcal{N}^{-1} = \sum_i \eta_i \mathcal{B}_i$ where $\eta_i \in \mathbb{R}$ and $\sum_i \eta_i = 1$, we have

$$\mathcal{U} = \mathcal{N}^{-1}\mathcal{N}\mathcal{U}$$
$$= \sum_i \eta_i \mathcal{B}_i \mathcal{N}\mathcal{U}. \qquad (3)$$

Note that arbitrary operations can be represented as linear combinations of tensor products of single-qubit Clifford operations and Pauli measurements [22]. Here, we can rewrite Eq. (3) as

$$\mathcal{U} = \gamma_Q \sum_i q_i \, \text{sgn}(\eta_i) \mathcal{B}_i \mathcal{N}\mathcal{U}, \qquad (4)$$

where $\gamma_Q = \sum_i |\eta_i|$, $q_i = |\eta_i|/\gamma_Q$, $\gamma_Q \geq 1$ and $\text{sgn}(\eta_i)$ is a parity, which takes ± 1, corresponding to the operation $\mathcal{B}_i$. We refer to γ_Q as the QEM cost because it is related to the sampling overhead.

Now let us suppose that we measure an observable $\hat{M}$ and obtain

$$\langle \hat{M} \rangle_{\mathcal{U}} = \gamma_Q \sum_i q_i \langle \hat{\mu}_i^{\text{eff}} \rangle. \qquad (5)$$

Here, $\hat{\mu}_i^{\text{eff}} = \text{sgn}(\eta_i)\hat{m}_i$, and $\hat{m}_i$ is a measurement outcome for a process $\mathcal{B}_i\mathcal{N}\mathcal{U}$. We generate the process $\mathcal{B}_i$ with a probability q_i and multiply the corresponding parity with the measurement result, which is denoted as $\hat{\mu}^{\text{eff}}$. Then, the expectation value of the random variable $\hat{\mu}^{\text{mit}} = \gamma_Q \hat{\mu}^{\text{eff}}$ approximates the error-free expectation value $\langle \hat{M} \rangle_{\mathcal{U}}$. Note that since $\text{Var}[\hat{\mu}^{\text{mit}}] = \gamma_Q^2 \text{Var}[\hat{\mu}^{\text{eff}}]$ and a measurement outcome without QEM, which we denote $\hat{\mu}^{\text{nmit}}$ has a similar variance, the variance of the error-mitigated value is approximately amplified as $\Gamma_Q = \gamma_Q^2$. Therefore, we need to have Γ_Q times more samples to achieve a similar accuracy before applying QEM.

In practice, we use probabilistic error cancellation for each gate in quantum circuits. The ideal process for the entire quantum circuit is described as $\prod_{k=1}^{N_G} \mathcal{U}_k$. Denoting $\mathcal{U}_k = \gamma_Q^{(k)} \sum_{i_k} q_{i_k} \text{sgn}(\eta_{i_k}) \mathcal{B}_{i_k} \mathcal{N}_k \mathcal{U}_k$, we have

$$\prod_{k=1}^{N_G} \mathcal{U}_k = \prod_{k=1}^{N_G} \gamma_Q^{(k)} \sum_{i_1 i_2 \cdots i_{N_G}} \prod_{k=1}^{N_G} q_{i_k} \prod_{k=1}^{N_G} \text{sgn}(\eta_{i_k}) \prod_{k=1}^{N_G} \mathcal{B}_{i_k} \mathcal{N}_k \mathcal{U}_k. \qquad (6)$$

From Eq. (6), we can see that, in each gate, a process $\mathcal{B}_{i_k}$ is generated with probability q_{i_k}, and the product of parities $\prod_{k=1}^{N_g} \text{sgn}(\eta_{i_k})$ is multiplied with the measurement results to obtain the outcome $\hat{\mu}^{\text{eff}}$. This procedure is repeated, and the product of the mean of the outcomes $\langle \hat{\mu}^{\text{eff}} \rangle$

and $\gamma_Q^{\text{tot}} = \prod_{k=1}^{N_g} \gamma_Q^{(k)}$ approximates the correct expectation value. Note that here γ_Q^{tot} is the QEM cost for the entire quantum circuit. Let us assume the cost for each gate is uniform and can be approximated as $\gamma_Q^{(k)} = \gamma_Q = 1 + a\varepsilon$ with a and ε being a positive constant value and the effective error rate, respectively. Now the QEM cost and sampling overhead can be approximated as $\gamma_Q^{\text{tot}} \simeq e^{a\varepsilon N_G} = e^{(\gamma_Q-1)N_G}$ and $\Gamma_Q^{\text{tot}} = (\gamma_Q^{\text{tot}})^2$, which increase exponentially with the mean number of error events in the quantum circuit εN_G. Note that for $\varepsilon N_G = O(1)$ and $\varepsilon \to 0$, since $\varepsilon^k N_G = 0$ $(k \geq 2)$, the QEM cost can be exactly described as $\gamma_Q^{\text{tot}} = e^{(\gamma_Q-1)N_G}$.

B. Fault-tolerant quantum computing

1. Stabilizer formalism

In the framework of FTQC, one prepares a redundant number of physical qubits and performs quantum computing in a code space defined as a subspace of the whole Hilbert space. By repetitively performing quantum error detection and correction, we can protect the logical qubits defined in the code space against physical errors. The state of the logical qubits is manipulated in a fault-tolerant manner with a set of logical operations.

The stabilizer formalism [3,32] is the most standard way to construct quantum error-correcting codes. Here, supposing that we construct k logical qubits with n physical qubits, a 2^k-dimensional code space $\mathcal{C}$ is specified with a subgroup of n-qubit Pauli operators called the stabilizer group. Let the n-qubit Pauli group be

$$\mathcal{G}_n = \{\pm 1, \pm i\} \times \{I, X, Y, Z\}^{\otimes n}, \qquad (7)$$

where I is the identity operator and $X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$ are Pauli operators. The set of Pauli operators $\mathcal{S} \subset \mathcal{G}_n$ is called a stabilizer group if $\mathcal{S}$ is a commutative subgroup, the number of elements in $\mathcal{S}$ is 2^{n-k}, and $-I \notin \mathcal{S}$. We denote the $(n-k)$ generator set of a stabilizer group as $\mathcal{G} = (g_1, \ldots, g_{n-k})$. The code space $\mathcal{C}$ is defined as an eigenspace with $+1$ eigenvalues for all the operators in the stabilizer group, i.e., $\mathcal{C} = \{|\psi\rangle \mid \forall s_i \in \mathcal{S}, s_i |\psi\rangle = |\psi\rangle\}$. In the code space, we can introduce a logical basis as $\{|0\rangle_L, |1\rangle_L\}^{\otimes k}$ and logical Pauli operators as $\{I_L, X_L, Y_L, Z_L\}^{\otimes k}$. The code distance d is defined as the minimum number of physical qubits on which an arbitrary logical operator, except the logical identity $I_L^{\otimes k}$, acts.

During a quantum computation, physical errors that occur in the encoded state are detected by using $(n-k)$ Pauli measurements $P_s = (1/2)[I + (-1)^s g_i]$ for $s \in \{0, 1\}$. These measurements are called stabilizer measurements and their binary outcomes s are called syndrome

values. The original state is restored by applying appropriate feedback operations that are estimated from the syndrome values. These stabilizer measurements are performed repeatedly during a computation. One repetition of the stabilizer measurements is called a code cycle of fault-tolerant quantum computing. If the effective error probability per physical qubit during a cycle is smaller than a certain threshold, we can estimate the Pauli operator that restores the original state with an exponentially small failure probability with the code distance d. Since the required number of physical qubits n increases polynomially with the code distance d in typical quantum error-correcting codes, we can exponentially decrease the error probability of logical qubits with a polynomial qubit overhead.

2. Logical operations

We must not only correct physical errors but also update the logical quantum state for performing quantum computation. To this end, a universal set of logical operations should be performed in a fault-tolerant manner. According to the Solovay-Kitaev theorem [25,26], we can approximate arbitrary one- and two-qubit gates with a finite set of local operations. For example, the Hadamard gate $H = (1/\sqrt{2}) \left(\begin{smallmatrix} 1 & 1 \\ 1 & -1 \end{smallmatrix} \right)$, controlled-not (CNOT) gate $\Lambda = |0\rangle \langle 0| \otimes I + |1\rangle \langle 1| \otimes X$, and T gate $T = \exp[i(\pi/8)Z]$ form a universal gate set. Several logical operations can be performed by transversally operating the same one- or two-qubit operations on physical qubits. Since transversal operations constantly increase the effective physical error rate per qubit during a cycle, we can fault tolerantly achieve transversal logical operations. However, it is known that there is no stabilizer code for which the set of transversal gates is universal [33]. Thus, we need an additional technique to achieve fault-tolerant and universal quantum computing. The most promising solution is to create a quantum state called a magic state and perform nontransversal logical operations with gate teleportation [5]. For example, $|A\rangle_L = TH |0\rangle_L = (1/\sqrt{2})(e^{i\pi/8} |0\rangle_L + e^{-i\pi/8} |1\rangle_L)$ is a typical magic state and T-gate operations can be performed by consuming this state. This magic state encoded in a logical qubit can be constructed with a process called magic state injection. While the infidelity of a magic state created by magic state injection is generally larger than the logical error rate, we can create a high-fidelity magic state from several noisy magic states by using another quantum error-correcting code implemented on the logical space, which is called magic state distillation. Since the application of T gates requires a longer time than the other operations, the number of T gates is the dominant factor affecting the computation time of FTQC.

Although we can estimate a Pauli operation for recovery from syndrome values, we do not directly apply it immediately after estimation. Instead, we store the Pauli operations that should be applied to the physical qubits
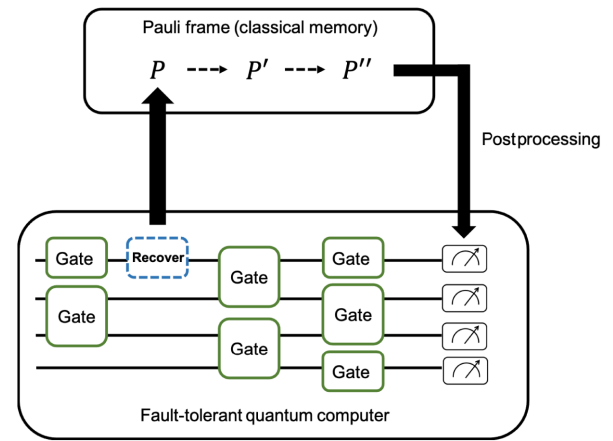

FIG. 2. Schematic figure of the Pauli frame. The recovery operations are not physically applied to quantum computers but rather are stored in the Pauli frame and efficiently updated after each Clifford gate operation. The measurement outcomes are flipped depending on the state of the Pauli frame.

for recovery in a classical memory called the Pauli frame [5,34]. The stored operations will be taken into account when the logical measurements are performed; the outcome of a logical measurement is flipped according to the Pauli frame. A schematic figure is shown in Fig. 2. In the above construction of logical operations, the whole process, except for magic state injection, consists only of Clifford operations and Pauli channels in the code space. Since a Pauli operator conjugated by a Clifford operator is also a Pauli operator, we can always track a recovery operator as a Pauli operator during a computation. In addition, when we can apply a logical Pauli operator to a quantum state, we can perform it simply by updating the Pauli frame, since a logical Pauli operator is a transversal physical Pauli operation. As far as classical computers are reliable, this operation is effectively noiseless.

III. QUANTUM ERROR MITIGATION FOR FAULT-TOLERANT QUANTUM COMPUTING

In this section, we discuss how to integrate QEM into the FTQC architecture. Here, we consider two types of errors in FTQC: decoding errors due to failures in the error estimation and insufficiency of magic state distillation and approximation errors in the Solovay-Kitaev decomposition. In Sec. A, we explain how these errors in FTQC can be modeled. In Sec. B, we discuss how these errors can be canceled and evaluate their QEM costs. Probabilistic error cancellation requires the errors to be estimated in advance. In Sec. C, we also discuss the effect of estimation errors on probabilistic error cancellation and the characterization efficiency.

A. Errors in fault-tolerant quantum computing

1. Decoding error

Here, we describe noise due to the failures of error estimation in elementary logical operations, i.e., stabilizer measurements and magic state distillation. The first obstacle to applying probabilistic error cancellation to FTQC is how to characterize an effective map of noise due to the failures of error estimation. If we suppose that the physical errors can be modeled as a stochastic physical Pauli map and assume that there are no errors on the ancillary qubits for syndrome measurements, we can define a logical noise map for decoding errors that is Markovian and a logical stochastic Pauli map. Yet, these assumptions do not hold in practice. Nevertheless, here we assume that we can define an effectively Markovian logical error map for each logical operation and also assume that this noise map is a stochastic logical Pauli map. It is known that even if noise is unitary, a noise map in a logical space of surface codes can be well approximated as stochastic Pauli noise when the code distance is sufficiently large [35]. Furthermore, the remaining coherent errors can be canceled by using pulse optimization techniques. Thus, it is reasonable to suppose that the decoding errors due to the failure of error estimations in surface codes are almost stochastic Pauli errors. In addition, we numerically verify that we can regard the decoding errors as Markovian errors even in the presence of measurement errors. See Appendix F for details. While we mainly describe and analyze the decoding errors in the surface codes, a similar idea can be applied to the decoding errors due to insufficient magic state distillation. As for the logical noise map on a prepared magic state due to insufficient magic state distillation, we can twirl the noise map by logical Clifford operations, and it can also be assumed to be a stochastic Pauli noise.

Under the above assumptions, we can describe a noise map for a l-qubit logical operation $\mathcal{N}_{\text{dec}}$ as the following stochastic Pauli noise:

$$\mathcal{N}_{\text{dec}}(\rho) = \sum_{g \in \{I_L, X_L, Y_L, Z_L\}^{\otimes l}} p_g g \rho g^{\dagger}, \tag{8}$$

where $p_g \in \mathbb{R}$, $\sum_g p_g = 1$ and $p_g \geq 0$. The sum of probabilities of nonidentity logical operations is called the logical error probability p_{dec}, i.e., $p_{\text{dec}} = \sum_{g \neq I^{\otimes l}} p_g$. It is known that when the physical error rate p is smaller than a value called the threshold p_{th}, the effective logical error probability decreases exponentially with respect to the code distance d. For the effective logical error probability per syndrome-measurement cycle of surface codes p_{cyc}, it decreases as

$$p_{\text{cyc}} \simeq C_1 \left(C_2 \frac{p}{p_{\text{th}}} \right)^{(d+1)/2}, \tag{9}$$

where C_1, C_2 are constants [36]. While the constant values depend on the details of the error correction schemes, $C_1 \simeq 0.13$ and $C_2 \simeq 0.61$ are expected in a typical construction of surface codes and the noise model [36,37]. Suppose that a logical operation requires m cycles; then, the logical error probability for the logical operation can be approximated as p_{dec} as

$$p_{\text{dec}} = 1 - (1 - p_{\text{cyc}})^m \simeq m p_{\text{cyc}}. \tag{10}$$

Note that the number of cycles per logical gate increases at most linearly with the code distance d.

In order to apply probabilistic error cancellation, we need to know the logical error probabilities $\{p_g\}$ in advance. While we can estimate $\{p_g\}$ by using gate-set tomography in the logical space, the estimations are not exact. The effect of estimation errors is discussed in Sec. C, while the efficiency of our proposal, including noise characterization, is discussed in Appendix C.

2. Approximation error

Since we are only allowed to use a limited set of logical operations for achieving fault tolerance, we need to decompose an arbitrary unitary gate into a sequence of available gates. Any unitary operator can be decomposed into a product of CNOT gates and single-qubit gates. Thus, we need to approximate single-qubit gates with a given gate set to the desired accuracy. By using the improved Solovay-Kitaev algorithm [38], given a universal gate set such as $\{T, H, S\}$ and the single-qubit gate U to be approximated, we can construct an approximated gate $\tilde{U}$, which satisfies $\varepsilon = \|\tilde{U} - U\|$ to an arbitrary accuracy ε as a sequence of given gate set with length $\tilde{O}[\log(\varepsilon^{-1})]$ with $\|\cdot\|$ being an operator norm. The error of approximated map is given by

$$\mathcal{N}_{\text{SK}}(\rho) = \tilde{U} U^{\dagger} \rho U \tilde{U}^{\dagger}. \tag{11}$$

Since this decomposition involves only single-qubit operations, this error channel can be efficiently and exactly evaluated in advance.

B. Quantum error mitigation for fault-tolerant quantum computing

1. Overview of our framework

Here, we show that decoding errors and approximation errors can be mitigated with probabilistic error cancellation. When we insert recovery operations for probabilistic error cancellation, it is assumed that the noise level of the recovery operations for QEM is much lower than that of the error-mitigated gates. In NISQ computing, for example, it is reasonable to assume that the error probabilities of two-qubit gates are much larger than those of single-qubit gates and measurements; therefore, the errors of two-qubit

gates can be mitigated by using single-qubit recovery operations. However, this is not a reasonable assumption in FTQC, since the operations that are noisy and time consuming are different from those of NISQ architecture. More concretely, even Clifford operations involving only one logical qubit suffer decoding errors.

Here, we show an architecture of FTQC that implements QEM with small overheads. The keys are the two significant properties of FTQC architecture: logical Pauli operations are error-free and instantaneous due to the Pauli frame, and the noise map of the decoding errors can be assumed as stochastic Pauli noise. Thanks to these properties, we can mitigate errors in all the elementary logical operations simply by updating the Pauli frame. This means the error-mitigated Clifford operations and Pauli measurements are available for computation. Because they form a complete basis for mitigating arbitrary errors [22], we can mitigate approximation errors due to the Solovay-Kitaev decomposition. Since the approximation errors can be exactly known in advance, an unbiased estimator free from approximation error can be obtained, as explained in Sec. 3.

To make our QEM procedure work, the accuracy and efficiency of the decoding error estimation are vital. We show that the decoding errors can be estimated with gate-set tomography under an appropriate choice of the gauge, considering state-preparation and measurement errors. We also show that the cost of gate-set tomography is acceptable compared with the main computation of FTQC for estimating expectation values in Sec. C. In this section, we further show a refined gate-set tomography suited to our framework that significantly improves the estimation for logical Clifford gates.

2. Quantum error mitigation for decoding errors

We can express the inverse channel of the nonuniform depolarizing channel Eq. (8) as a linear combination of Pauli operations. Thus, we can express the inverse channel as

$$\mathcal{N}_{\text{dec}}^{-1}(\rho) = \sum_{g \in \{I_L, X_L, Y_L, Z_L\}^{\otimes l}} \eta_g g \rho g^{\dagger}$$

$$= \gamma_{\text{dec}} \sum_{g \in \{I_L, X_L, Y_L, Z_L\}^{\otimes l}} q_g \, \text{sgn}(\eta_g) g \rho g^{\dagger}. \quad (12)$$

Refer to Appendix B for a concrete expression of each coefficient η_g, γ_{dec}, and q_g. Thus, we can suppress the errors by applying probabilistic error cancellation only with Pauli operators after the decoding processes. The QEM cost for decoding errors in the entire circuit can be expressed as $\gamma_{\text{dec}}^{\text{tot}} = \prod_{k=1}^{N_{\text{dec}}} \gamma_{\text{dec}}^{(k)}$, where N_{dec} is the number of logical gates, and $\gamma_{\text{dec}}^{(k)}$ is a QEM cost of the kth operation.
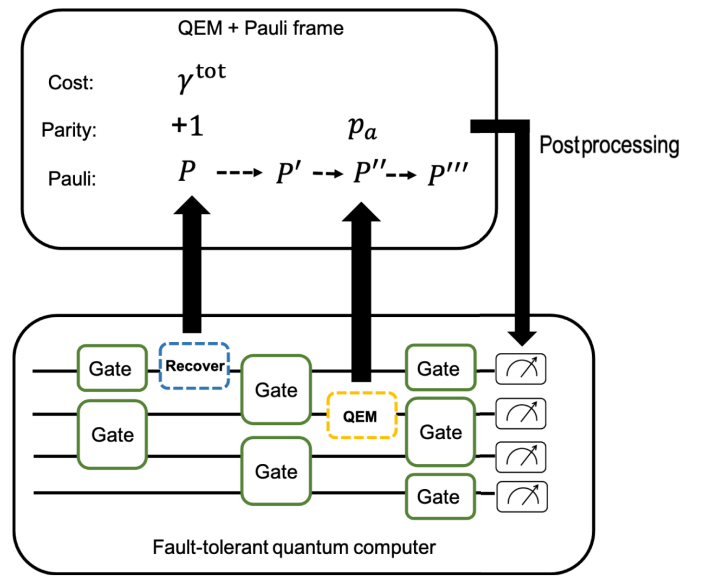

FIG. 3. Schematic figure for the Pauli frame incorporating QEM. If a QEM recovery operation is a Pauli operation, it is not directly applied to the quantum computer but rather the Pauli frame is updated instead. The parity is also updated in accordance with the generated recovery operations of QEM. Here, we denote the parity corresponding to the QEM recovery operation as p_a in the figure. If a QEM recovery operation is not a Pauli operator, it is performed physically. Then measurement outcomes are then postprocessed depending on the Pauli frame, parity, and QEM cost.

Note that probabilistic error cancellation usually applies the recovery operations of QEM immediately after the noisy gates [21,22]; however, because we perform only logical Pauli operations as the recovery operations for decoding errors, they can be done simply by updating the Pauli frame instead of directly applying them after noisy gates. Finally, the measurement result is postprocessed according to the state of the Pauli frame, the parity corresponding to the applied recovery operations, and the QEM cost. Thus, unlike in probabilistic error cancellation for NISQ devices, the logical noise due to decoding errors can be mitigated without any additional noise due to the recovery operation. A schematic figure is shown in Fig. 3. Note that the information on the QEM cost and the parity is used only when the final measurement result is obtained; the outcome of a destructive logical Pauli measurement is flipped depending only on the state of the Pauli frame. Whether we can mitigate decoding errors of complicated logical operations such as magic state preparation, gate teleportation, and adaptive Clifford gates by simply updating the Pauli frame is not trivial; therefore we provide a concrete procedure for actual devices and Pauli frames in Appendix E.

In the case of decoding errors in surface codes, by approximating the QEM cost to the first order of the logical

error, we have

$$\gamma_{\text{dec}} \simeq 1 + 2p_{\text{dec}}. \tag{13}$$

Refer to Appendix B for details. Under the assumption that the logical error rate is the same for all the logical operations and $p_{\text{dec}}N_{\text{dec}} = O(1)$ with $p_{\text{dec}} \rightarrow +0$, the QEM cost $\gamma_{\text{dec}}^{\text{tot}}$ for the entire quantum circuit can be shown to be exactly equal to $e^{2p_{\text{dec}}N_{\text{dec}}}$ on the basis of the argument in Sec. A. Thus, by using Eqs. (9) and (10), we obtain

$$\gamma_{\text{dec}} - 1 = 2mC_1 \left(C_2 \frac{p}{p_{\text{th}}} \right)^{(d+1)/2}, \tag{14}$$

which results in the total QEM sampling overhead

$$\Gamma_L^{\text{tot}} = e^{2(\gamma_{\text{dec}}-1)N_{\text{dec}}} = \exp\left[4mC_1N_{\text{dec}} \left(C_2 \frac{p}{p_{\text{th}}} \right)^{(d+1)/2} \right]. \tag{15}$$

Notice that Eq. (15) clearly shows a trade-off relationship between the sampling overhead and the code distance, i.e., the number of physical qubits.

Finally, although we mainly focus on the surface code architecture in this paper, our scheme is also available for typical stabilizer-based error-correcting codes such as color codes [39] and concatenated Steane or Shor codes [3,40,41] because we assume only basic properties of stabilizer codes, i.e., Pauli recovery operations are estimated from syndrome measurements and non-Clifford operations are applied by generating and consuming magic states. Therefore, we can choose suitable quantum error-correcting codes according to the hardware connectivity.

3. Quantum error mitigation for approximation errors

Unlike decoding errors due to the failure of error correction, we cannot describe errors due to the Solovay-Kitaev decomposition as stochastic Pauli errors. Nevertheless, we can still apply probabilistic error cancellation with negligible overheads. Denote $\mathcal{N}_{\text{SK}}(\rho) = \tilde{U}U\rho(\tilde{U}U)^\dagger$; we invert this approximation error by

$$\mathcal{N}_{\text{SK}}^{-1} = \sum_i \eta_i \mathcal{B}_i^{(L)}$$
$$= \gamma_{\text{SK}} \sum_i q_i \, \text{sgn}(\eta_i) \mathcal{B}_i^{(L)}, \tag{16}$$

where $\{\mathcal{B}_i^{(L)}\}$ denotes recovery operations in the logical space. Note that we can represent any map as a linear combination of Clifford operations and Pauli channels [22], and thus, we do not need T gates for mitigating approximation errors. Recovery operations are randomly chosen and applied immediately after each single-qubit logical operation if they are not Pauli operations. In the case of Pauli

operations, we can again use the Pauli frame, and physical operations on quantum computers are not required, in a similar vein to QEM for decoding errors. Since a single-qubit logical unitary operation consists of several repetitions of Clifford gates and T-gate teleportation, the insertion of the recovery operation for probabilistic error cancellation negligibly increases the length of the quantum circuit. In the numerical simulations described in the next section, we verify that the QEM costs can be approximated with the following equation:

$$\gamma_{\text{SK}} - 1 = \beta_1 e^{-\beta_2 N_T}, \tag{17}$$

where β_1 and β_2 are constants dependent on the quantum gate and N_T is the number of available T gates.

The QEM cost due to approximation errors can also be represented as $\gamma_{\text{SK}}^{\text{tot}} = \prod_{k=1}^{N_{\text{SK}}} \gamma_{\text{SK}}^{(k)}$, where N_{SK} is the total number of recovery operations for mitigating approximation errors in the quantum circuit with the cost $\gamma_{\text{SK}}^{(k)}$ corresponding to the kth recovery operation. By assuming that the cost does not depend on gates, we have the following QEM sampling overhead:

$$\Gamma_{\text{SK}}^{\text{tot}} \simeq \exp\left(2\beta_1 N_{\text{SK}} e^{-\beta_2 N_T} \right). \tag{18}$$

This shows there is a trade-off relationship between the sampling overhead and the number of available T gates.

C. Effect of estimation errors of the noise map

1. Effect of estimation errors on expectation values

While approximation errors can be exactly determined in advance, decoding errors have to be characterized. Since the decoding error probabilities are small, it is unavoidable that the characterization will contain finite and non-negligible estimation errors. Thus, we need to care about QEM with estimation errors and the efficiency of the characterization of decoding errors.

Let us discuss how estimation errors affect the performance of QEM. Given a perfect characterization of the noise model $\mathcal{N}_k$ for the kth gate, we can realize the inverse operation $\mathcal{N}_k^{-1}$ with probabilistic error cancellation to achieve $\mathcal{N}_k^{-1}\mathcal{N}_k = \mathcal{I}$. If we obtain an incorrect estimation for the error process $\mathcal{N}_k' \neq \mathcal{N}_k$, it leads to an estimation error $\Delta\mathcal{N}_k \equiv \mathcal{N}_k'^{-1}\mathcal{N}_k \neq \mathcal{I}$.

Now, denoting the ideal process of the kth gate as $\mathcal{U}_k$, the difference of the error-mitigated process and the error-free process for the entire quantum circuit can be described by the diamond norm:

$$\left\| \prod_{k=1}^{N_G} \Delta\mathcal{N}_k \mathcal{U}_k - \prod_{k=1}^{N_G} \mathcal{U}_k \right\|_\diamond \leq \Delta\varepsilon N_G, \tag{19}$$

where we use the fact that the diamond norm is subadditive and we denote $\Delta\varepsilon = \max_k \|\Delta\mathcal{N}_k - \mathcal{I}\|_\diamond$. Similarly,

the discrepancy of the noisy and ideal process can be upper bounded as $\| \prod_{k=1}^{N_G} \mathcal{N}_k \mathcal{U}_k - \prod_{k=1}^{N_G} \mathcal{U}_k \|_\diamond \le \varepsilon N_G$, where $\varepsilon = \max_k \| \mathcal{N}_k - \mathcal{I} \|_\diamond$. Because the deviation of the expectation values of an observable M for two processes $\mathcal{E}_1$ and $\mathcal{E}_2$ with the input state ρ can be described as $\delta M = \mathrm{Tr}\{M[\mathcal{E}_1(\rho) - \mathcal{E}_2(\rho)]\} \le \|M\| \|\mathcal{E}_1 - \mathcal{E}_2\|_\diamond$ [42], where $\| \cdot \|$ is an operator norm, we have $\delta M_{\mathrm{QEM}} \le \|M\| \Delta \varepsilon N_G$ and $\delta M_{\mathrm{noise}} \le \|M\| \varepsilon N_G$. Here, δM_{QEM} and $\delta M_{\mathrm{noise}}$ are the deviation of the observable with and without error mitigation.

Thus, we can see that QEM is beneficial when we can achieve $r < 1$ for

$$\Delta \varepsilon = r \varepsilon. \qquad (20)$$

Note that this discussion does not include sampling errors; i.e., δM is the error of the expectation value given infinite samples.

2. Efficiency of characterization of decoding errors

As a cause of model estimation errors, when we use gate-set tomography to characterize the noise model for decoding errors, we need to consider state preparation and measurement (SPAM) errors and the finite statistical error arising from an insufficient number of samples. It has been shown that the effect of SPAM errors can be eliminated in the case of probabilistic error cancellation based on gate-set tomography [22]. While the general choice of the gauge is not compatible with the Pauli frame, we can modify the scheme of gate-set tomography so that this method is compatible with QEM with the Pauli frame. Refer to Appendix C for details.

To achieve an accuracy r given in Eq. (20), we need to perform $N_{\mathrm{GST}} = O[(r\varepsilon)^{-2}]$ samplings for gate-set tomography [28,43]. Here, we show this efficiency is acceptable compared with the main part of FTQC, i.e., the time required for gate-set tomography corresponds to $O(r^{-2}n_q N_G)$ runs of the whole quantum logical circuits to obtain expectation values, where n_q is the number of logical qubits. Let the time for a single run of the logical circuit of FTQC be τ. The depth of logical quantum circuit is estimated as $O(N_G n_q^{-1})$, and the time per gate can be roughly approximated as $\tau_{\mathrm{gate}} = O(\tau n_q N_G^{-1})$. Then, the time for gate-set tomography can be estimated as $\tau_{\mathrm{GST}} = O(\tau_{\mathrm{gate}} N_{\mathrm{GST}}) = O[\tau N_G^{-1} n_q (r\varepsilon)^{-2}]$. In a situation where QEM is useful, we have $\varepsilon N_G = O(1)$ [44]. Thus, we can conclude that to use QEM to decrease the logical error rate p_{dec} to rp_{dec} by QEM, we need gate-set tomography as a precomputation that takes $\tau_{\mathrm{GST}} = O(\tau N_G n_q r^{-2})$, which is $\tau_{\mathrm{GST}}/\tau = O(N_G n_q r^{-2})$ times longer than a single circuit run of FTQC. The numbers of logical gates N_G and logical qubits n_q are expected to grow polynomially with the problem size, and FTQC circuits will be repeated on the order of $O(r^{-2})$ to make the statistical fluctuation of expectation values smaller than the reduced bias. Accordingly,

while the estimation costs of the noise map cause another overhead to FTQC depending on the required accuracy, it is performed with a time that grows polynomially with the problem size and without requiring additional physical qubits. We remark that when we assume the noise properties of the quantum devices are uniform, we can perform the sampling for gate-set tomography in parallel. If we use all the logical qubits for characterization, the time for gate-set tomography is reduced to $\tau_{\mathrm{GST}} = O(\tau N_G r^{-2})$. Note that, in the scenario that we can fully parallelize the sampling procedure, i.e., when we have $O[(r\varepsilon)^{-2}]$ distinct uniform quantum gates, we have $\tau_{\mathrm{GST}} = O(\tau N_G^{-1})$.

To further make the characterization of noise more efficient, we propose an improved gate-set tomography for decoding errors of the Clifford process that is fast and compatible with the Pauli frame. See Appendix C for the details of this scheme. The number of measurements N_{GST} is reduced to $N_{\mathrm{GST}} = O(r^{-2}\varepsilon^{-1})$, which makes the costs of precomputation $O(n_q r^{-2})$. Thus, as long as r is not too small, the time for characterization is expected to be relatively short. While our efficient gate-set tomography cannot be applied to the characterization of the T-gate preparation, several ways to reduce the costs for estimating errors of T gates can be considered. Since the error of logical T gate depends on physical T gate and the process of injection and distillation is constructed by a few T-gate circuit, there may be an efficient way to numerically estimate the noise of logical T gate from the characterization of physical T gate and efficient simulation for quantum circuits dominated by Clifford gates [45]. There may be a way to mitigate T-gate errors by temporally expanding the code distance or increasing the distillation depth for T gate. The cost of gate-set tomography might be also reduced by utilizing long-sequence GST [43], i.e., repeating several T gates to amplify a small error rate to a large value. Reference [46] shows that if decoding errors of logical Clifford gates are negligible, one can reliably twirl the noise of T gate and perform efficient process tomography on that by repeating T gates. Nevertheless, it is still an open problem whether there exists a more efficient gate-set tomography on the logical space with imperfect logical Clifford gates.

3. Effective increase in code distance by quantum error mitigation under estimation errors

We can regard that QEM effectively increases the code distance. Suppose that we can effectively achieve an r times smaller logical error rate $p_{\mathrm{eff}} = rp_{\mathrm{dec}}$ via QEM. Since the logical error rate is roughly approximated with the code distance as $p_{\mathrm{dec}}(d) = p(p/p_{\mathrm{th}})^{(d-1)/2}$, QEM effectively achieves a larger code distance d' where $p_{\mathrm{eff}} = p_{\mathrm{dec}}(d')$ without increasing the number of physical qubits. The effective increase in the code distance via QEM can be

derived as

$$d' - d = 2\frac{\ln r}{\ln \left(p/p_{\text{th}} \right)}. \qquad (21)$$

Therefore, by setting $r = (p/p_{\text{th}})^x$, we can effectively increase the code distance by $2x$. Note that, as discussed in the previous sections, we need $\exp[O(N_{\text{dec}}p_{\text{dec}})] = \exp[O(1)]$ times more repetitions to achieve the same precision as the error-free case. It is worth noting that we can increase the number of available logical qubits via QEM. If we are allowed to use a fixed number of physical qubits, the decrease in the code distance indicates that we can allocate more logical qubits; therefore, we can convert the code distance into the number of logical qubits.

IV. NUMERICAL ANALYSIS

We numerically evaluate how well error mitigation suppresses the qubit overhead in FTQC. (See Appendix G for the detailed settings and the definitions of the terms used in the numerical analysis.)

A. Quantum error mitigation for decoding errors

1. Cost analysis

We evaluate the performance of QEM on decoding errors occurring during logical operations, where we assume FTQC with surface codes and lattice surgery. (See Appendix D for details about surface codes.) For simplicity, we assume a single-qubit depolarizing noise model for each data and measurement qubit at the beginning of each cycle, which corresponds to a phenomenological noise model [47,48]. To determine the failure probability of decoding with faulty syndrome-measurement cycles, we further assume perfect syndrome measurements in the 0th and dth cycles. Then, we check whether any logical Pauli errors occur during d cycles. The recovery operations are estimated from the syndrome values by using the minimum-weight perfect-matching decoder [49,50]. We evaluate the logical error probabilities of Pauli X, Y, Z and compute the QEM cost for d cycles according to Eq. (B5). Despite our assumption of perfect syndrome measurements of the 0th and dth cycle, we expect that the numerical results are asymptotically equivalent to those without the assumption when d is sufficiently large.

The logical error probabilities of Pauli X, Y, Z of a single logical qubit for several code distances are calculated. The sum of their probabilities are plotted according to physical error rates in Fig. 4(a). The logical error probability exponentially decreases according to the code distance when the physical error probability p is smaller than a threshold value. Figure 4(b) plots the logical error probabilities around the threshold value, which is around $p_{\text{th}} = 0.044$.

We compute the QEM costs for decoding errors γ_{dec} corresponding to d cycles and different code distances and

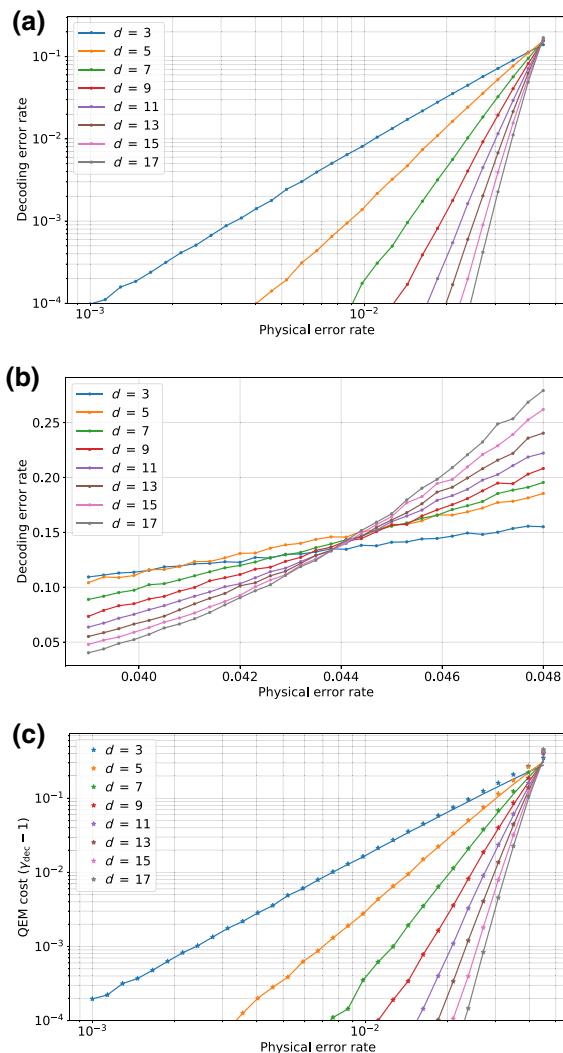
FIG. 4. Logical error probability and QEM cost for d-cycle syndrome measurements plotted against the physical error rates at several code distances. (a) Logical error probability as a function of physical error rate. (b) The same figure enlarged around the threshold value. (c) QEM costs for d-cycle idling operations. The first-order approximations from Eq. (13) are shown as solid lines.

compare them with the first-order approximation shown in Eq. (13). The numerical results are plotted in Fig. 4(c). In this figure, the solid lines correspond to the approximation of the QEM cost in Eq. (13). We can see that the QEM costs decay exponentially depending on the code distances and show a threshold behavior like the logical error probabilities. They coincide well when the physical error rate is sufficiently small.

2. Performance analysis

Next, we examine the performance of QEM on decoding errors in large-scale quantum circuits with a 100-qubit logical random Clifford circuit with 100 layers. We remark

that since a linear combination of Clifford operations can represent arbitrary quantum operations, it is sufficient to demonstrate the performance of QEM for Clifford operations [51]. We simultaneously apply randomly generated single-qubit Clifford gates to each layer, and then we apply 50 CNOT gates to two randomly chosen qubits. We can simulate these protocols efficiently by using an efficient algorithm for stabilizer circuits [32,52]. As an observable, we choose a Pauli operator whose measurement outcome is always unity for the final state vector if there are no physical errors; in other words, the final state is a $+1$ eigenstate of the chosen observable. We assume a nonuniform single-qubit depolarizing logical error in the form of Eq. (8) for each layer. The logical error probabilities of depolarizing channels are determined according to the numerical results of the last section. We choose $p = 0.01$ and obtain the logical error probabilities by extrapolation. The estimated logical error probabilities are summarized in Table I. Without QEM, the final state converges to a highly mixed state due to physical errors. Thus, it is expected that expectation values decay to zero. By employing QEM, they are taken back to unity, sacrificing statistical accuracy and requiring a greater number of experiments accordingly.

TABLE I. Estimated logical error probability for code distances with $p = 0.01$ and $p_{\text{th}} \sim 0.044$.

Code distance d	p_{X_L}, p_{Z_L}	p_{Y_L}
5	1.80×10^{-4}	1.96×10^{-6}
7	1.39×10^{-5}	4.11×10^{-8}
9	1.08×10^{-6}	8.64×10^{-10}
11	8.35×10^{-8}	1.81×10^{-11}

Note that while the required number of cycles for Clifford operations scales linearly with the code distance, the actual number of cycles and logical error probability per logical gate are dependent on the Clifford operations. In particular, logical CNOT gates with lattice surgery may induce correlated logical Pauli errors on multiple logical qubits. Nevertheless, we use a simplified error model, since we expect this evaluation captures the basic properties of QEM performance.

We numerically perform a series of 10^4 experiments, each of which computes an expectation value from 10^4 single-shot measurements. The results are shown in Fig. 5(a), while data around the ideal expectation value are shown in Fig. 5(b). Figure 5(c) shows the mean value of
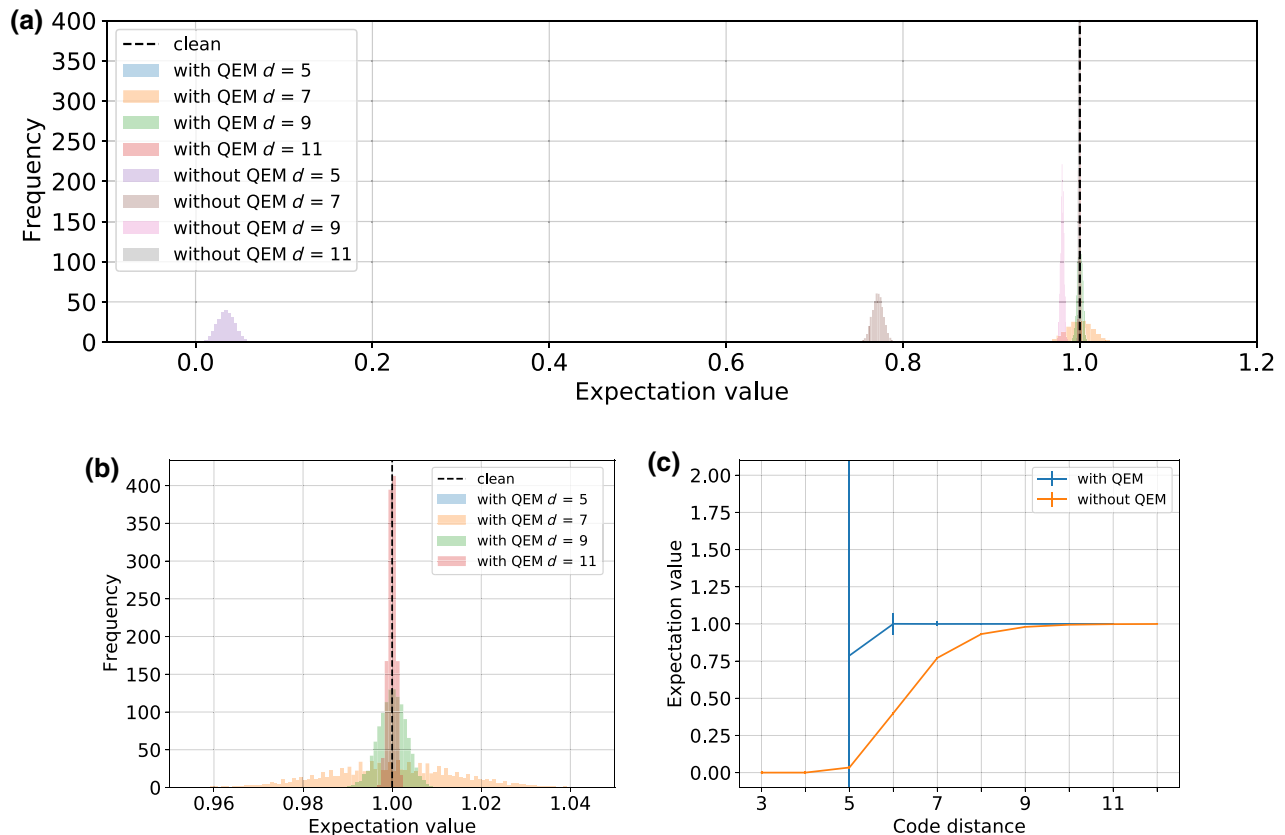

FIG. 5. (a) Histogram of expectation values for 100-qubit random Clifford circuits. (b) Histogram of expectation values with QEM for 100-qubit random Clifford circuits. (c) Sample averages and standard deviation of 100-qubit random Clifford circuits are plotted as a function of code distances.
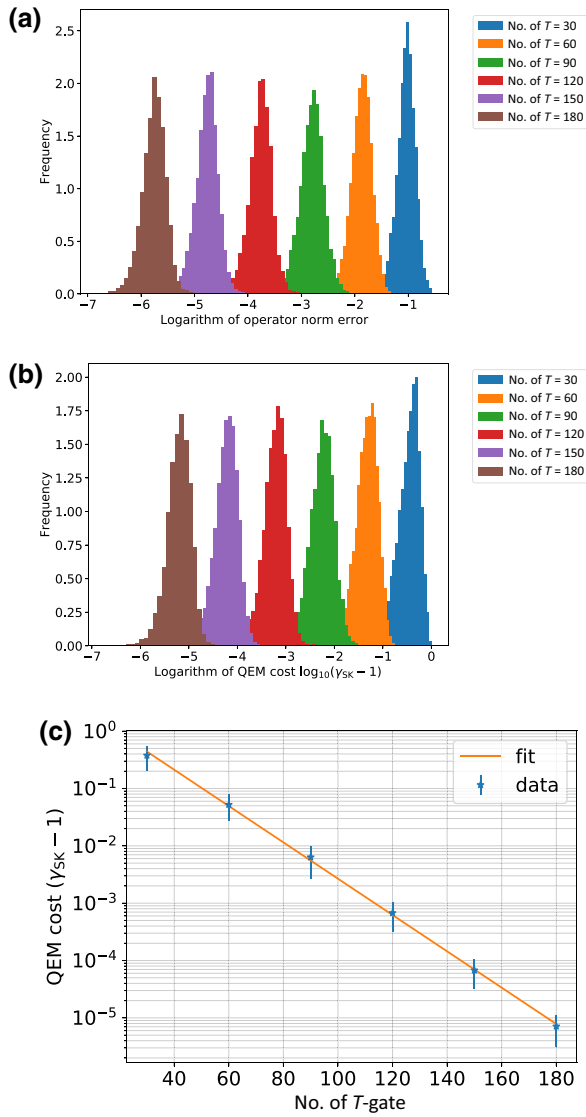
(a)

(b)

(c)

FIG. 6. Approximation error and QEM cost of improved Solovay-Kitaev method calculated for Haar-random unitary operations. Each color corresponds to the maximum count of T gates. (a) Histogram of approximation errors due to the Solovay-Kitaev decomposition. (b) Histogram of QEM costs. (c) Histogram of QEM costs for approximation errors as a function of the number of allowed T gates.

10^4 samples for each logical error probability together with its standard deviation (the error bar). We can see that there is a large bias in the expectation value without QEM, but no bias when the QEM technique is employed, while its standard deviation is amplified. The standard deviation of the expectation value for $d = 5$ is 14.2, and thus, it is not visible in the histogram because it is too large. The mean number of Pauli errors in the whole quantum circuit is 3.6 for $d = 5$ and 0.28 for $d = 7$. Thus, as explained in Sec. A, QEM is useful when the number of Pauli errors in the circuit is less than unity. These results show that the QEM
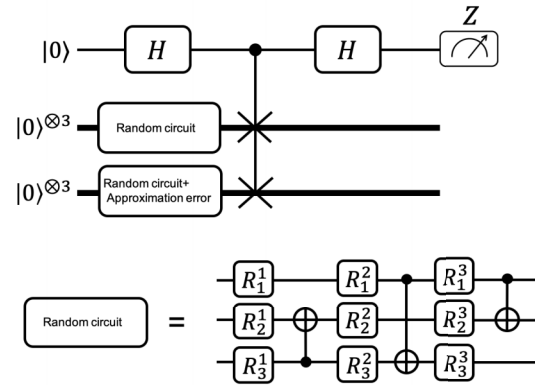

FIG. 7. Schematic figure of simulated seven-qubit SWAP test circuit. We calculate the overlap of randomly generated states and the approximation of the generated state by using the Solovay-Kitaev algorithm. The random circuit is composed of three layers, each of which consists of random single-qubit rotation gates and CNOT gates acts on two randomly chosen qubits.

technique is effective for large-scale quantum computing and it enables us to increase the effective code distance.

B. Quantum error mitigation for approximation errors

1. Cost analysis

Next, we study the performance of QEM when the Solovay-Kitaev decomposition is used. Since the actual QEM cost γ_{SK} depends on the target unitary operator, we draw a sample of unitary operations U from a Haar-measure random distribution μ_H. Then, we decompose a unitary gate in the form of $U = R_Z(\theta_1)\sqrt{X}R_Z(\theta_2)\sqrt{X}R_Z(\theta_3)$, where $\sqrt{X} = HSH$ is a Clifford operation. We use the improved Solovay-Kitaev algorithm from Ref. [38]. This algorithm enables us to approximate an arbitrary Pauli-Z rotation $R_Z(\theta) = \exp[i(\theta/2)Z]$ with an operator $\tilde{U}$, which is described as a sequence of Clifford operations and T gates. We set the maximum count of T gates for each decomposition for three Pauli-Z rotations to check the trade-off relation between the T-gate count and the approximation accuracy. Figure 6(a) shows the histogram of errors evaluated with operator norm $||U - \tilde{U}||$. As expected, there is an exponential decrease in the approximation errors.

Next, we calculate the QEM cost by using Eq. (16). Figure 6(b) shows the histogram of QEM costs γ_{SK}, and Fig. 6(c) plots the QEM cost versus the number of allowed T gates. We can see that $\gamma_{SK} - 1$ exponentially decreases according to the number of T gates, and its variance also decreases exponentially. We fit the QEM cost γ_{SK} with Eq. (17), and obtain $\beta_1 = 3.9(5)$ and $\beta_2 = 0.072(1)$.
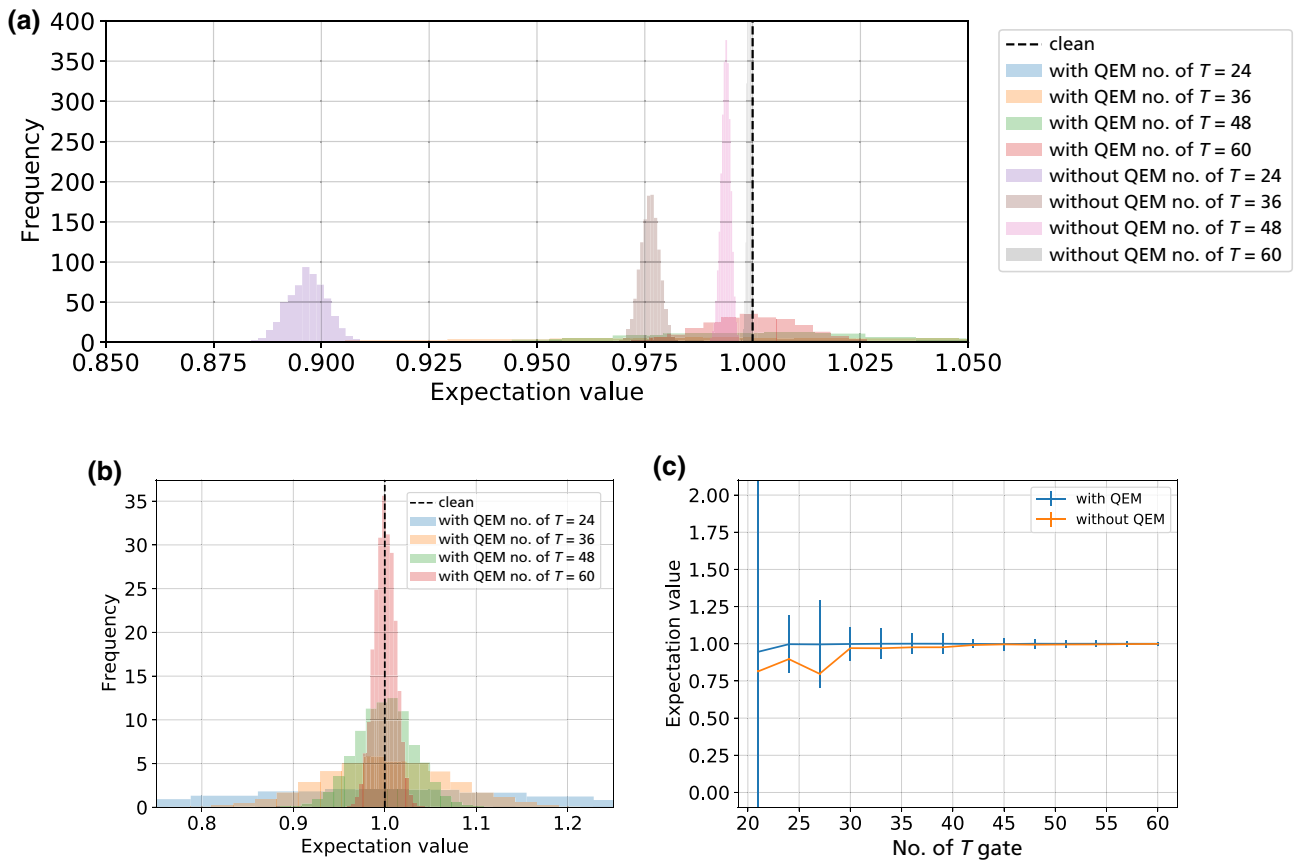
FIG. 8. Histogram of expectation values of SWAP test with approximation errors. (a) Histogram of expectation values. (b) The same figure enlarged around the ideal expectation value. (c) Sample averages and standard deviation as a function of the allowed number of T gates.

2. Performance analysis

Next, we evaluate the performance of QEM for approximation errors due to the Solovay-Kitaev decomposition in a simulation of a SWAP test circuit with seven qubits. A SWAP test circuit evaluates the overlap of two input states ρ and σ as $\text{Tr}[\rho\sigma]$ by measuring ancilla qubits [53]. We set one of the input states to the ideal state and the other to the state affected by approximation errors. A schematic diagram is shown in Fig. 7.

The ideal state is generated by using random quantum circuits composed of three layers. In each layer, random single-qubit unitary operations are simultaneously applied; then a CNOT gate acts on two randomly chosen qubits. The same random quantum circuit is applied to the approximate state by applying the Solovay-Kitaev decomposition to each single-qubit rotation. In this case, if there are no approximation errors, we necessarily obtain $+1$ as measurement outcomes since the input states are the same; hence the expectation value is also $+1$ for the Pauli-Z operator of the ancilla qubit. On the other hand, the expectation value becomes smaller than unity when the inner product is reduced by approximation errors. Since approximation errors cannot be treated in the framework of stabilizer

simulation, we simulate the quantum circuits by directly updating the state vector after each gate.

We numerically perform a series of 10^4 experiments and compute the expectation values from 10^4 single-shot measurements. The number of allowed T gates in each Solovay-Kitaev decomposition for single-qubit unitary operations is varied from 24 to 60. Figure 8(a) shows the results, while Fig. 8(b) shows the data around the ideal expectation value. Moreover, Fig. 8(c) shows the mean value of 10^4 samples for each logical error probability together with its standard deviation (the error bar). Note that standard deviation with 21 T gates is 4.65. We can see that our QEM technique successfully remove bias from the expectation value in the noisy cases. Compared with the QEM cost for decoding errors, we obtain a larger QEM cost in this case. This is consistent with the results reported in Ref. [22] that indicates the QEM cost for unitary errors tends to be larger than that for stochastic errors.

This problem may be alleviated by performing several Solovay-Kitaev decompositions with the same accuracy, constructing randomizing approximation errors, and removing the coherent component of the noise. Note that with a sufficiently large sample size, our QEM technique
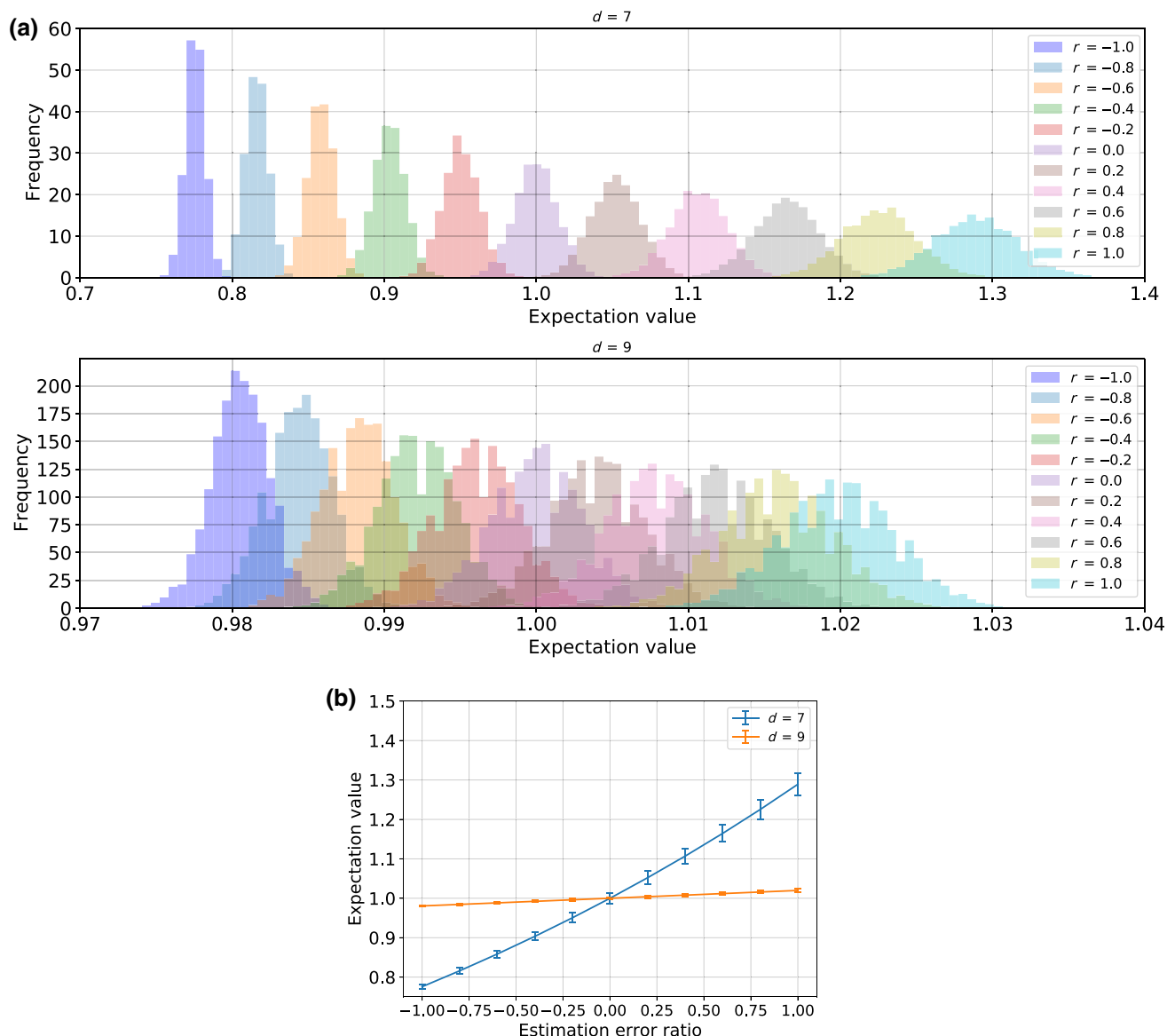
FIG. 9. Histogram of expectation values with estimation errors. (a) Histogram of expectation values. (b) Sample averages and standard deviation as a function of estimation accuracy.

enables the effective number of T gates to be increased by inserting additional Clifford gates and Pauli channels and conducting repeated sampling, with negligible additional hardware requirements.

C. Quantum error mitigation with estimation errors

Probabilistic error cancellation assumes that the noise maps to be canceled are known in advance. While we can determine the approximation error of the Solovay-Kitaev decomposition within the numerical precision, it is hard to exactly characterize the noise maps of the decoding errors because GST is affected by finite sampling, as discussed in Sec. C. In this section, we numerically evaluate the

performance of our framework in the case of finite estimation errors.

For the benchmarks, we choose the same quantum circuit, noise model, and observable as in Sec. 2. We evaluate the expectation value for a 100-qubit noisy Clifford circuit that is unity if there is no noise. A nonuniform depolarizing channel where Pauli X, Y, Z occurs with probabilities (p_x, p_y, p_z) is inserted after each gate. We use the probabilities obtained from the simulation of surface codes shown in Table I. The point of difference from the previous simulations is that these probabilities are over- or underestimated as $[(1+r)p_x, (1+r)p_y, (1+r)p_z]$ $(r > -1)$ in probabilistic error cancellation. Here, $r = 0$ corresponds to an exact characterization and perfect error mitigation, while $r = -1$

corresponds to FTQC without error mitigation. As in the discussion in Sec. C, we expect the distribution is such that its mean value is the same as in the simulation of the mitigated noise model $(|r|p_x, |r|p_y, |r|p_z)$ and its variance is approximately Γ, which is the overhead of QEM determined by the estimated noise model $[(1 + r)p_x, (1 + r)p_y, (1 + r)p_z]$. The histogram in the case of finite estimation errors parametrized by r for $d = 7$ and $d = 9$ are plotted in Fig. 9(a), and the mean values and standard deviations are plotted as a function of r in Fig. 9(b). We can see that the residual bias increases exponentially with $|r|$. With infinite samples, QEM is beneficial when r is sufficiently smaller than unity. Comparing the cases of under-estimation $r < 0$ and overestimation $r > 0$ with the same absolute value $|r|$, we find that overestimation ($r > 0$) has a larger variance than that of underestimation ($r < 0$), while there is similar amount of bias in the expectation values. Since QEM prefers underestimation to overestimation, we conclude that characterization methods with a weighted penalty may lead to a further improvement in QEM.

D. Practical utility of quantum error mitigation for FTQC

While we have shown our method effectively decreases the hardware requirement of FTQC with the examples of random Clifford and swap-test circuits, the practical availability of QEM in the region where FTQC is used for useful applications with a quantum advantage is also vital. In this section, we discuss the enhancement of computation accuracy in this regime with our protocol.

We estimate how many quantum logical Clifford operations N_G are required in the practical region from the existing resource estimation. Note that there are other noise sources, such as imperfect T-gate preparation, that can be counted as the overhead of logical operations in distillation processes, and thus we count the effects of them as decoding errors. We consider two scenarios for evaluation; an optimistic one and a realistic one. An optimistic scenario is the case of lightweight applications mainly for showing a quantum advantage, i.e., an application with the minimum possible N_G that cannot be simulated with the existing classical computers. We refer to the analysis of Refs. [13,54] to estimate the maximum problem size tractable with the existing classical computers. According to them, we expect that quantum circuits with depth 100 and 100 logical qubits are sufficient to go beyond the limitation of classical simulation and that we can achieve this with $N_G \sim 10^4$. The other scenario is the ground-energy estimation of spin models and chemical molecules since the quantum simulation is expected as one of the most resource-efficient applications whose quantum advantage is well studied. We pick an expected number of gates from recent the

state-of-the-art resource estimation in Refs. [9,10] that discuss the cost for calculating ground-state energy. These papers utilize quantum simulation with Trotterization [55] and qubitization [56] as a subroutine, respectively. Trotterization is a method to simulate quantum systems by approximating Hamiltonian dynamics with Trotter decomposition [55], and qubitization is a recently proposed method that constructs the state after time evolution by repetitive applications of Grover-like iterations [56]. According to Tables 1 and 2 in Ref. [9] and Table IV in Ref. [10], approximately 10^8 T gates are required to simulate a Hubbard model that is hard to simulate with classical computers. Note that while these algorithms use phase-estimation sampling to obtain the ground energy as binary digits and are not a procedure to estimate expectation values, we can still apply QEM to these algorithms with small overheads. This is because these problems can be translated into a series of decision problems. See Appendix H for a detailed explanation. Considering there is an overhead of executing magic state distillation and so on, we expect that $N_G = 10^{10}$ is required as a pessimistic estimation.

Next, we consider how QEM can reduce the effect of decoding errors of logical operations during FTQC. Without QEM, a logical error probability p_L must be much smaller than the inverse of the number of logical operations N_G. In other words, the mean number of logical errors $N_G p_L$ must satisfy $N_G p_L = O(1)$. Suppose the allowed mean number of errors without QEM for satisfying the required accuracy as N_e, which becomes small as required accuracy becomes strict. On the other hand, with QEM, the bias of expectation values caused by logical errors whose mean number is below unity can be mitigated with $e^{O(1)}$, i.e., about $e^4 \sim 55$, sampling overheads according to Eq. (15). Thus, the required logical error rates are relaxed from $p_L \sim N_e/N_G$ to $p_L \sim 1/N_G$. The effective error rates of elementary logical operations such as logical Clifford gates decrease exponentially with the code distance. When we focus on the advantage of QEM in the logical space, we can estimate the relations between the problem size N_G and code distances d in terms of the required mean number of errors without QEM N_e as shown in Fig. 10(a). In this figure, the required code distance for reliable accuracy is plotted as a function of the number of logical gates required for executing an algorithm for several cases of N_e. For the calculation, we use Eq. (9) with $p/p_{\text{th}} = 0.1$. The number of physical qubits per logical qubit scales as $O(d^2)$ in the case of two-dimensional topological codes. We plot how the number of physical qubits are reduced with QEM $d_{\text{mit}}^2/d_{\text{nmit}}^2$ in Fig. 10(b), where d_{mit} and d_{nmit} are the required code distance with and without QEM, respectively. While the improvement of the code distance d is constant, the impact of resource reduction depends on the expected technologies and the size of the problems of interest.
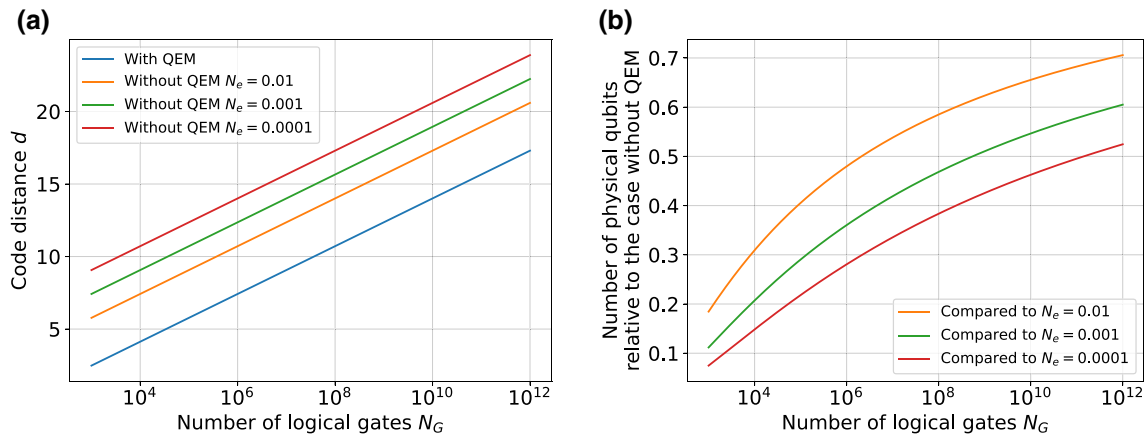
FIG. 10. (a) The relation between the required number of gates for executing algorithm N_G and the required code distance for encoding the information of qubit d. The relation is plotted for several numbers of allowed logical error counts during FTQC N_e. (b) The ratio between the number of required physical qubits per logical qubit with and without QEM.

When we consider applications in the early FTQC era with $N_G \sim 10^4$, according to Fig. 10(a), the required code distance is reduced from about 9 to 4 if $N_e = 10^{-3}$ and the required number of qubits is reduced to 21%. This advantage becomes large when the required accuracy N_e becomes strict. Even when we take the cost for distillation and lattice surgery into account, FTQC with 100 logical qubits with the code distance 4 is estimated to require about 10^4 physical qubits with QEM, which can be the promising milestone to show the computational supremacy on the logical space. While this advantage becomes comparably small when we see promising long-term applications in $N_G \sim 10^{10}$, the code distance is still reduced from 19 to 14, which suppresses the number of physical qubits to 55%. Thus, we can conclude that in both cases, our proposal is expected to drastically alleviate the hardware requirement in practice.

It should be noted that the reduction of the code distance is vital not only for the reduction of the number of physical qubits but also for relaxing the requirement of error decoding architectures. To estimate occurred Pauli errors on physical qubits during FTQC, we need classical peripherals for decoding with sufficiently small latency for stabilizer measurement cycles. However, recent results show the tractable size of realistic implementation can decode up to about code distance 11 [15–17]. While this value may be improved in the future, it is obvious that the performance of decoding units is another restriction of FTQC. When we assume tractable code distance is limited to 11, we can use at most 10^5 logical gates with $N_e = 10^{-3}$, which is just around the limitation of the classical simulation. On the other hand, the use of QEM can increase the available logical gates to 10^8. Thus, our proposal can be the key to push the performance of FTQC from a classically simulatable region to the quantum supremacy regime.

Although we discuss only the reduction of code distance, we can similarly reduce the effective required number of T gates. While the effective increase of the resource is also constant as in the case of code distance, this can be used not only for reducing the total number of generated T gates but also for mitigating the bias of generation throughput of magic states during the FTQC protocol. Since magic state generation succeeds probabilistically, the number of available magic states per time unit statistically fluctuates and cannot be estimated in advance. We expect our method can exempt these kinds of difficult runtime schedulings and make long-time execution of FTQC reliable. Furthermore, in the case of distributed FTQC, logical CNOT gates between distant nodes require entanglement distribution and distillation, which also typically succeed probabilistically. Thus, QEM can be used for reducing a wide range of difficulties in FTQC.

V. DISCUSSION

We have described a method to effectively decrease errors in FTQC by performing QEM in the logical space. In the case of decoding errors due to insufficient code distances and magic state distillation, we can perform QEM with small modification on quantum circuits. In particular, owing to the Pauli frame, we can perform QEM without implementing any physical operations if decoding errors are stochastic Pauli maps, while QEM operations may induce additional errors when we implement it physically on general quantum circuits. In regard to the approximation errors due to the Solovay-Kitaev decomposition, we cannot always use the Pauli frame because QEM employs not only Pauli operations but also Clifford operations. Since Clifford operations can be efficiently performed in FTQC and the number of decoding processes for error correction is much larger than the gate count in

Solovay-Kitaev algorithms, this overhead is negligible. We have verified the trade-off between sampling costs of QEM and the code distance and the number of T gates. Furthermore, we have estimated the sampling cost of gate-set tomography to obtain the decoding noise map to the required accuracy, and clarified that our approach enables quantum computing corresponding to more than the achievable code distance. We have numerically compared the computation with and without QEM on FTQC with the same sampling number and have shown the advantage of QEM even under the existence of finite estimation errors. We also have estimated the required resources with and without QEM in the early FTQC era, and have shown that the required number of physical qubits can be suppressed up to tens of percent.

It should be noted that this is the first result to clearly show QEM can dramatically improve the computation accuracy for useful applications under realistic assumptions as we have shown by using the example of quantum simulation on FTQC. This is because the computational advantage of typical NISQ algorithms such as variational algorithms are empirically assumed and the required runtime of the algorithm has not been revealed; yet the accuracy of FTQC algorithms can be estimated reliably depending on the complexity of the problem. Accordingly, the usefulness of quantum error mitigation can be clearly discussed in the FTQC scenario.

Another important aspect of providing the theory and implementation of QEM for FTQC rather than NISQ computing is as follows. While it is known that QEM is the most effective when the mean number of errors during computation is in the order of unity [22], this criterion is not necessarily satisfied in the NISQ regime depending on the problem size. On the other hand, because we are allowed to tune code distances, magic state distillation levels, and the number of T gates per Solovay-Kitaev decomposition in FTQC, it is highly likely that we can satisfy this criterion. Thus, the main drawback of QEM, the exponential growth of the sampling overheads, can be circumvented; therefore we can find the highly practical regimes where QEM can help enhance the computation accuracy. Accordingly, we can conclude that the theory of QEM on FTQC is more versatile than that for NISQ computing.

As mentioned in the main text, promising applications of our method are quantum phase estimation algorithms and Hamiltonian simulation algorithms for investigating quantum many-body dynamics. There are algorithmic errors in the Trotter decomposition [8] and recently proposed methods such as Taylor series [57] and quantum signal processing [56,58]. In Refs. [59,60], it is shown that such algorithmic errors can be mitigated by employing extrapolation. Since algorithmic errors can be controlled by changing the simulation accuracy, this technique can also be naturally incorporated in an FTQC scenario. Thus, the dominant errors in FTQC can be compensated via QEM.

The first generation of FTQC may not be sufficiently large for naively solving large and useful problems. While the architecture of distributed quantum computing is the most straightforward approach to increase the total number of qubits, it requires interconnections between quantum nodes, which induces additional overheads for entanglement distillation. Thus, we sometimes cannot use the sufficient number of distilled entanglements for distributed FTQC. In this context, techniques developed in the NISQ era [61,62] for solving larger problems with small NISQ computers may also be useful in the middle-term FTQC. Our work is the first proposal that makes the best of a technique tailored for NISQ devices in the context of FTQC.

Finally, we should discuss the difference between our scheme and a similar work that combines QEM with the quantum error correction proposed by Ref. [63]. Their method considers implementing quantum error correction for NISQ devices via classical postprocessing in the case that experimentalists cannot implement stabilizer measurements because of the limited connectivity and large error rates of NISQ devices. Although their method enables the state to be projected to the code space via quantum subspace expansion [64], logical errors cannot be fully eliminated. On the other hand, our scheme assumes FTQC can be performed but the number of qubits and T gates cannot be increased infinitely. The remarkable advantage of our method is that we can fully eliminate the decoding errors and approximation errors by using a greater number of measurements at negligible hardware overhead, given the good characterization of the noise model.

Note added.—Recently, three relevant works appeared that have a similar concept that quantum error mitigation is incorporated in fault-tolerant quantum computing to relax the hardware requirement at the cost of sampling overheads [46,65,66]. Reference [65] shows quantum error mitigation for encoded qubits but they focus on concatenated codes rather than topological codes. Reference [46] uses quantum error mitigation for implementing T gate without magic state distillation and shows efficient characterization methods for the errors of T gate under the assumption that logical Clifford operations are perfect. Reference [46] also discusses how to relax the costs for

implementing T gates by using the concepts of robustness of magic.

Compared to these works, we emphasize that our framework considers a different scenario where logical Clifford operations are imperfect, and thus FTQC suffers from decoding errors of logical Clifford procedure, logical noise on prepared magic states, and insufficient magic state supply. This difference makes our framework versatile in the early FTQC era. While we provided consistent analysis of gate-set tomography with noisy logical Clifford gates, we have refined the treatment of the noise of magic state preparation in gate-set tomography, motivated by Refs. [46,66].

APPENDIX A: PAULI TRANSFER MATRIX

Any quantum map can be represented as a matrix called a Pauli transfer matrix (PTM). Suppose we perform a quantum process $\mathcal{E}$ on n qubits that maps an n-qubit density matrix to another density matrix. We denote the set of n-qubit Pauli operators as $\mathcal{P}^{(n)} = \{I, X, Y, Z\}^{\otimes n}$. The set of n-qubit Pauli operators forms a basis of $4^n \times 4^n$ matrices and the elements are mutually orthonormal, $(1/d)\mathrm{Tr}[P_i P_j] = \delta_{ij}$, for $d = 2^n$. The PTM representation of process $\mathcal{E}$ is defined with the Pauli basis as follows:

$$\mathcal{M}(\mathcal{E})_{ij} = \frac{1}{d}\mathrm{Tr}[P_j \mathcal{E}(P_i)], \tag{A1}$$

where P_i is the ith element of $\mathcal{P}^{(n)}$. Note that since any physical process maps a self-adjoint operator to another self-adjoint operator, all the elements of the Pauli transfer matrix are real values. The density matrix can also be represented as a column vector with the Pauli basis:

$$|\rho\rangle\rangle_i = \mathrm{Tr}[P_i \rho]. \tag{A2}$$

Note that an element of the vector corresponding to $P_i = I$ is the trace of ρ. A measurement of an observable O can be mapped to a row vector,

$$\langle\langle O|_j = \frac{1}{d}\mathrm{Tr}[O P_j]. \tag{A3}$$

Note that the PTM representation satisfies

$$\mathrm{Tr}[O\mathcal{E}(\rho)] = \langle\langle O|\mathcal{M}(\mathcal{E})|\rho\rangle\rangle. \tag{A4}$$

There are several important properties of the Pauli transfer matrix representation. Given a composite map $\mathcal{E} = \mathcal{E}_A \circ \mathcal{E}_B$, the PTM representation is $\mathcal{M}(\mathcal{E}_B \circ \mathcal{E}_A) = \mathcal{M}(\mathcal{E}_B)\mathcal{M}(\mathcal{E}_A)$. Due to the linearity of the PTM representation, for the map $\mathcal{E}' : \rho \mapsto \sum_k q_k \mathcal{E}_k(\rho)$, we have $\mathcal{M}(\mathcal{E}') = \sum_k q_k \mathcal{M}(\mathcal{E}_k)$. When no confusion is possible, we simply represent the Pauli transfer matrix $\mathcal{M}(\mathcal{E})$ as $\mathcal{E}$.

APPENDIX B: COEFFICIENTS FOR QUASIPROBABILITY DECOMPOSITION

To perform probabilistic error cancellation on an arbitrary l-qubit noise map $\mathcal{N}$, we need to decompose the inverse of the noise map $\mathcal{N}^{-1}$ into a linear combination of physical quantum processes. According to Ref. [22], any TPCP map can be represented as a linear combination of Clifford operations and Pauli channels. This is because the set of Pauli transfer matrices of l-qubit Clifford operations and Pauli channels forms a basis of l-qubit Pauli transfer matrices. Here, we introduce the following 16 operators:

$$\mathcal{B} = \left\{I, \sigma_j, \frac{I + i\sigma_j}{\sqrt{2}}, \frac{I + \sigma_j}{2}, \frac{\sigma_j + \sigma_{j+1}}{\sqrt{2}}, \frac{\sigma_j + i\sigma_{j+1}}{2}\right\}, \tag{B1}$$

where $j \in \{1, 2, 3\}$, $(\sigma_1, \sigma_2, \sigma_3) = (X, Y, Z)$, and $\sigma_4 = \sigma_1$. The Pauli transfer matrices of $\mathcal{B}^{\otimes l}$ comprise a complete basis of l-qubit Pauli transfer matrices; in other words, any quantum map can be represented as a linear combination of Clifford operations and Pauli channels. Since the application of non-Clifford operations requires complicated processes such as magic state injection, distillation, and teleportation, this property is vital to performing probabilistic error cancellation on an arbitrary noise map in FTQC.

Specifically, when the noise can be modeled as stochastic Pauli errors, we can cancel it only with Pauli operations. This is because the set of Pauli transfer matrices of Pauli operations forms a basis of diagonal l-qubit Pauli transfer matrices. In FTQC, since we can perform logical Pauli operations only by updating the Pauli frame, which is stored in a classical memory, we can cancel stochastic logical Pauli noise without acting on the actual quantum device. Suppose that the noise model is described by a Pauli transfer matrix acting on l qubits:

$$\mathcal{N}_{\mathrm{Pauli}} = (1 - p_{\mathrm{err}})\mathcal{I} + \sum_{g \neq I^{\otimes l}} p_g \mathcal{P}_g, \tag{B2}$$

where $\mathcal{I}$ is the identity map, $\mathcal{P}_g$ is the Pauli transfer matrix of Pauli operator g, p_g ($g \in \{I, X, Y, Z\}^{\otimes l}$) is the probability at which the Pauli error g occurs, and $p_{\mathrm{err}} = \sum_{g \neq I^{\otimes l}} p_g$. This noise can be canceled with the following map:

$$\mathcal{N}_{\mathrm{Pauli}}^{-1} = \sum_g \eta_g \mathcal{P}_g, \tag{B3}$$

where

$$\eta_g = 4^{-l} \sum_{g'} \frac{c(g, g')}{\sum_{g''} p_{g''} c(g', g'')}. \tag{B4}$$

Note that $c(g, g')$ is a function of two Pauli operators such that $c(g, g') = 1$ if $gg' = g'g$ and $c(g, g') = -1$ otherwise.

In the case of single-qubit Pauli noise, the coefficients and QEM cost γ_Q can be explicitly expressed as follows:

$$\eta_I = \frac{1}{4}\left(1 + \frac{1}{1-2(p_Y+p_Z)} + \frac{1}{1-2(p_Z+p_X)} + \frac{1}{1-2(p_X+p_Y)}\right),$$

$$\eta_X = \frac{1}{4}\left(1 + \frac{1}{1-2(p_Y+p_Z)} - \frac{1}{1-2(p_Z+p_X)} - \frac{1}{1-2(p_X+p_Y)}\right),$$

$$\eta_Y = \frac{1}{4}\left(1 - \frac{1}{1-2(p_Y+p_Z)} + \frac{1}{1-2(p_Z+p_X)} - \frac{1}{1-2(p_X+p_Y)}\right),$$ (B5)

$$\eta_Z = \frac{1}{4}\left(1 - \frac{1}{1-2(p_Y+p_Z)} - \frac{1}{1-2(p_Z+p_X)} + \frac{1}{1-2(p_X+p_Y)}\right),$$

$$\gamma_Q = \sum_{g\in\{I,X,Y,Z\}} |\eta_g| = \frac{1}{2}\left(-1 + \frac{1}{1-2(p_Y+p_Z)} + \frac{1}{1-2(p_Z+p_X)} + \frac{1}{1-2(p_X+p_Y)}\right).$$

Next, we show that the first-order approximation of the QEM cost for stochastic Pauli noise is Eq. (13). Consider an unphysical map,

$$\mathcal{N}'_{\text{Pauli}} \equiv (1 + p_{\text{err}})\mathcal{I} - \sum_{g\neq I^{\otimes l}} p_g \mathcal{P}_g.$$ (B6)

We can easily show

$$\|\mathcal{N}'_{\text{Pauli}}\mathcal{N}_{\text{Pauli}} - \mathcal{I}\| \leq 4p_{\text{err}}^2,$$ (B7)

where $\|\cdot\|$ denotes the operator norm for the Pauli transfer matrix. Here, we can see that $\mathcal{N}'_{\text{Pauli}}$ is an approximation of the inverse map $\mathcal{N}_{\text{Pauli}}^{-1}$ up to first order, and we have $\mathcal{N}'_{\text{Pauli}} = \mathcal{N}_{\text{Pauli}}^{-1} + O(p_{\text{err}}^2)$. Accordingly, the QEM cost can be approximated as

$$\gamma_Q \approx p_{\text{err}} + \sum_g p_g = 1 + 2p_{\text{err}},$$ (B8)

when $p_{\text{err}} \ll 1$.

APPENDIX C: PROBABILISTIC ERROR CANCELLATION WITH GATE-SET TOMOGRAPHY

In this section, we explain how probabilistic error cancellation can be implemented using the result of gate-set tomography. Furthermore, we also show this method is compatible with the Pauli frame.

1. Gate-set tomography

Suppose that our goal is to characterize the gate set $\{\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_{N_s}\}$, which involves at most N qubits. To implement gate-set tomography, we measure

$$\tilde{\mathcal{G}}_{ij} = \langle\langle O_i|\mathcal{G}|\rho_j\rangle\rangle,$$ (C1)

where $\mathcal{G}$ is one of the gates in the gate set and $\langle\langle O_i|$ and $|\rho_j\rangle\rangle$ are linearly independent 4^N observables and states. Note that the measurement results are generally noisy because of SPAM errors. Let O_i^0 and $\rho_j^{(0)}$ be error-free observables and states. Here, we set $O_i^{(0)} \in \{I, X, Y, Z\}^{\otimes N}$ and $\rho_j^{(0)} \in \{|0\rangle, |1\rangle, |+\rangle, |+_y\rangle\}^{\otimes N}$ with $|+\rangle = (1/\sqrt{2})(|0\rangle + |1\rangle)$ and $|+_y\rangle = (1/\sqrt{2})(|0\rangle + i|1\rangle)$. Note that $\langle\langle I|$ corresponds to a trivial measurement whose outcome is always unity. By inserting the identity operator $\mathcal{I} = \sum_k |O_k^{(0)}\rangle\rangle\langle\langle O_k^{(0)}|$, we obtain

$$\tilde{\mathcal{G}}_{ij} = \sum_{kk'} A_{ik}^{(\text{out})} \mathcal{G}_{kk'} A_{k'j}^{(\text{in})},$$ (C2)

where $A_{ik}^{(\text{out})} = \langle\langle O_i|O_k^{(0)}\rangle\rangle$, $A_{k'j}^{(\text{in})} = \langle\langle O_{k'}^{(0)}|\rho_j\rangle\rangle$, and $\mathcal{G}_{kk'} = \langle\langle O_k^{(0)}|\mathcal{G}|O_{k'}^{(0)}\rangle\rangle$. Thus, we have

$$\tilde{\mathcal{G}} = A^{(\text{out})}\mathcal{G}A^{(\text{in})}.$$ (C3)

Note that $A^{(\text{out})}$ and $A^{(\text{in})}$ are affected by SPAM errors, which cannot be experimentally measured because they cannot be separated from each other.

In addition, we replace $\mathcal{G}$ with the identity operation to obtain

$$g = A^{(\text{out})}A^{(\text{in})}.$$ (C4)

In a typical scenario of gate-set tomography, the estimation of the process is represented as

$$\mathcal{G}^{\text{est}} = Bg^{-1}\tilde{\mathcal{G}}B^{-1}$$
$$= BA^{(\text{in})-1}\mathcal{G}A^{(\text{in})}B^{-1},$$ (C5)

with B being an arbitrarily chosen invertible matrix. Denoting the error-free $A^{(\mathrm{in})}$ matrix as $A^{(\mathrm{in})(0)}$, a feasible choice is to set $B = A^{(\mathrm{in})(0)}$ when the initialization error is small. We estimate the initial state and measurement as follows:

$$
\begin{aligned}
|\rho_j^{\mathrm{est}}\rangle\rangle &= B_{\bullet,j} = BA^{(\mathrm{in})-1}|\rho_j\rangle\rangle, \\
\langle\langle O_i^{\mathrm{est}}| &= (gB^{-1})_{i,\bullet} = \langle\langle O_i|A^{(\mathrm{in})}B^{-1},
\end{aligned}
\tag{C6}
$$

which can be computed from the B and g matrices. Now, by implementing the same procedure for $\mathcal{G}_k(k = 1, 2, \dots N_s)$ and assuming identical SPAM errors for each experiment, we can estimate $\mathcal{G}_k^{\mathrm{est}} = BA^{(\mathrm{in})-1}\mathcal{G}A^{(\mathrm{in})}B^{-1}$. Although the estimated gate set, initial states, and measurements may differ from the true ones $\{|\rho_j\rangle\rangle, \langle\langle O_i|, \mathcal{G}_k\}$ due to SPAM errors, they give the correct expectation value for the gate-set sequence as follows:

$$
\langle\langle O_i^{\mathrm{est}}| \prod_k \mathcal{G}_k^{\mathrm{est}}|\rho_j^{\mathrm{est}}\rangle\rangle = \langle\langle O_i| \prod_k \mathcal{G}_k|\rho_j\rangle\rangle.
\tag{C7}
$$

Throughout this paper, we refer to the transformation due to SPAM errors and the B matrix as the gauge and denote G_a. In the aforementioned case, $G_a = BA^{(\mathrm{in})-1}$.

Note that the choice of the gauge is crucial because it affects the set of required QEM operations; on the other hand, the QEM operations are restricted to Pauli operations because of the use of the Pauli frame. Therefore, we need to carefully choose the gauge so that only Pauli operations are used for QEM. We propose to estimate the gate set, initial states, and measurements as follows:

$$
\begin{aligned}
\mathcal{G}_k^{\mathrm{est}} &= \tilde{\mathcal{G}}g^{-1} = A^{(\mathrm{out})}\mathcal{G}A^{(\mathrm{out})-1}, \\
|\rho_j^{\mathrm{est}}\rangle\rangle &= g_{\bullet,j} = A^{(\mathrm{out})}|\rho_j\rangle\rangle, \\
\langle\langle O_i^{\mathrm{est}}| &= \langle\langle O_i^{(0)}| = \langle\langle O_i|A^{(\mathrm{out})-1}.
\end{aligned}
\tag{C8}
$$

The above formalism corresponds to the case with gauge $G_a = A^{(\mathrm{out})}$. Later, we see that this choice of gauge is compatible with QEM with the Pauli frame in the presence of stochastic Pauli errors.

2. Probabilistic error cancellation

Here, we discuss how to derive quasiprobabilities in order to implement probabilistic error cancellation based on the results of gate-set tomography. In addition, we show that this method is compatible with the Pauli frame. Let us assume that we obtain the estimations of the gate set, initial states, and measurements as follows:

$$
\begin{aligned}
\mathcal{G}_k^{\mathrm{est}} &= G_a\mathcal{G}_kG_a^{-1}, \\
|\rho_j^{\mathrm{est}}\rangle\rangle &= G_a|\rho_j\rangle\rangle, \\
\langle\langle O_i^{\mathrm{est}}| &= \langle\langle O_i|G_a^{-1}.
\end{aligned}
\tag{C9}
$$

Let us also assume that gate-set tomography is applied to the basis operations for QEM and estimates are obtained

as $\mathcal{B}_l^{\mathrm{est}} = G_a\mathcal{B}_lG_a^{-1}$. Here, we denote the ideal operation as $\mathcal{U}_k$. We attempt to invert the estimated noise process, i.e., $\mathcal{N}_k^{\mathrm{est}-1} = \mathcal{U}_k\mathcal{G}_k^{\mathrm{est}-1} = \sum_l q_l\mathcal{B}_l^{\mathrm{est}}$. In reality, what we implement via probabilistic error cancellation corresponds to $\mathcal{N}_k^{\mathrm{mit}-1} = \sum_l q_l\mathcal{B}_l = G_a^{-1}\mathcal{U}_kG_a\mathcal{G}_k^{-1}$. Therefore, the error-mitigated gate can be expressed as $\mathcal{G}_k^{\mathrm{mit}} = \mathcal{N}_k^{\mathrm{mit}-1}\mathcal{G}_k = G_a^{-1}\mathcal{U}_kG_a$. Similarly, we try to construct the ideal initial states and measurements, i.e., $|\rho_j^{(0)}\rangle\rangle = \sum_{l'} q_{l'}\mathcal{B}_{l'}^{\mathrm{est}}|\rho_j^{\mathrm{est}}\rangle\rangle$ and $\langle\langle O_i^{(0)}| = \sum_{l''} q_{l''}\langle\langle O_i^{\mathrm{est}}|\mathcal{B}_{l''}^{\mathrm{est}}$, which actually correspond to $|\rho_j^{\mathrm{mit}}\rangle\rangle = G_a|\rho^{(0)}\rangle\rangle$ and $\langle\langle O_i^{\mathrm{mit}}| = \langle\langle O_i^{(0)}|G_a^{-1}$, respectively. Accordingly, we obtain the error-mitigated expectation value for the sequence of quantum gates,

$$
\langle\langle O_i^{\mathrm{mit}}| \prod_k \mathcal{G}_k^{\mathrm{mit}}|\rho_j^{\mathrm{mit}}\rangle\rangle = \langle\langle O_i^{(0)}| \prod_k \mathcal{U}_k|\rho_j^{(0)}\rangle\rangle.
\tag{C10}
$$

Now, let us discuss the compatibility of this method with the Pauli frame. When using the Pauli frame for QEM, only Pauli operations are allowed. This indicates that the solution of

$$
\mathcal{N}_k^{\mathrm{est}-1} = \sum_l q_l\mathcal{B}_l^{\mathrm{est}},
\tag{C11}
$$

contains only Pauli operators. Here, we choose the gauge $G_a = A^{(\mathrm{out})}$. Note that in the presence of stochastic Pauli measurement errors, $A^{(\mathrm{out})}$ can also be described by a diagonal matrix corresponding to a stochastic Pauli error. We can easily check that the Pauli operations are invariant under this gauge and that $\mathcal{N}_k^{\mathrm{est}-1}$ is also described by stochastic Pauli noise when the gates suffer stochastic Pauli errors. Thus, only Pauli operators are required for QEM. Similarly, we can also show only Pauli operations are required to realize $|\rho_j^{(0)}\rangle\rangle = \sum_{l'} q_{l'}\mathcal{B}_{l'}^{\mathrm{est}}|\rho_j^{\mathrm{est}}\rangle\rangle$ and $\langle\langle O_i^{(0)}| = \sum_{l''} q_{l''}\langle\langle O_i^{\mathrm{est}}|\mathcal{B}_{l''}^{\mathrm{est}}$ for the state preparation in the presence of stochastic Pauli errors. Therefore, this method is fully compatible with the Pauli frame.

3. Efficiency of gate-set tomography for decoding errors

In this section, we discuss the efficiency of gate-set tomography. Here, we assume that the noise of elementary logical operations is modeled as stochastic Pauli noise. Then, the noise map is of the form $\mathcal{N} = \sum_{g \in \{I,X,Y,Z\}^{\otimes l}} p_g g\rho g$, where $p_g = O(p_L)$. We evaluate the number of required samplings to estimate p_g within a precision of $(1 \pm r)p_g$.

When we estimate each element of $\tilde{\mathcal{G}}$ and g with the standard error δ, it is enough to perform $N_{\mathrm{GST}} = O(\delta^{-2})$ samplings. Suppose we have obtained $\tilde{\mathcal{G}}$ and g with a small statistical fluctuation, as $\tilde{\mathcal{G}} + \Delta\tilde{\mathcal{G}}$ and $g + \Delta g$, respectively, where each element of $\Delta\tilde{\mathcal{G}}$ and Δg is in the order of $O(\delta)$. We can show that the gate-set tomography estimation in Eq. (C8) can be performed with small

standard errors. $|\rho_j^{\text{est}}\rangle\rangle$ can be directly obtained from g, and $\langle\langle O_i^{\text{est}}|\rangle$ can be obtained without any statistical errors. Then, we obtain $\mathcal{G}^{\text{est}}$ as $\mathcal{G}^{\text{est}} = (\tilde{\mathcal{G}} + \Delta\tilde{\mathcal{G}})(g + \Delta g)^{-1} \sim \tilde{\mathcal{G}}g^{-1} + \Delta\tilde{\mathcal{G}}g^{-1} - \tilde{\mathcal{G}}g^{-1}\Delta g g^{-1}$. Thus, the elements of $\mathcal{G}^{\text{est}}$ have the same order of standard error. The noise map of $\mathcal{G}^{\text{est}}$ can be obtained as $\mathcal{N} = \mathcal{G}^{\text{est}}(\mathcal{G}^{(0)})^{-1}$, where $\mathcal{G}^{(0)}$ is an error-free gate. Thus, the logical error probabilities can be also estimated with the standard error $O(\delta)$. Let p_g^{est} for $g \in \{I, X, Y, Z\}^{\otimes N}$ be the estimated logical error. Then, to achieve an estimation accuracy of the form $p_g^{\text{est}} = (1 + r)p_g$, its standard error must be smaller than rp_g. Thus, the number of samples required to achieve an accuracy factor r in the above form is given by $N_{\text{GST}} = O[(rp_L)^{-2}]$. Note that when we evaluate the noise map of magic state preparation, we perform state tomography for the magic state twirled with Clifford gates, which also requires $O[(rp_L)^{-2}]$ samplings for the same accuracy.

Next, we show that the required number of samplings can be improved to $N_{\text{GST}} = O(r^{-2}p_L^{-1})$ when the target gate of gate-set tomography is a logical Clifford gate. When there is no noise, every element of $\tilde{\mathcal{G}}$ and g is obtained as the results of Pauli measurements for a stabilizer state. Each element of $\tilde{\mathcal{G}}$ and g is zero if the measured state is not an eigenstate of an observable, and ± 1 otherwise. Since the noise is modeled as stochastic Pauli noise, these elements are zero even in the presence of noise. Thus, we do not need to perform sampling for such elements. When the elements are ± 1 without noise, the elements with noise become $\pm(1 - 2\mu)$, which is obtained as the mean value of a random variable $\hat{m}$ such that $\hat{m} = \pm 1$ with probability $1 - \mu$ and $\hat{m} = \mp 1$ with probability μ, where $\mu = O(p_L)$. Supposing that N_{GST} samplings are performed on this distribution, $\pm(1 - 2\mu)$ can be estimated with the standard error $O\left(\sqrt{p_L N_{\text{GST}}^{-1}}\right)$. Therefore, to estimate each element of $\tilde{\mathcal{G}}$ and g with the standard error δ, it is enough to perform $N_{\text{GST}} = O(\delta^{-2}p_L)$ samplings. This means when we need to estimate logical error probabilities with the accuracy $\delta = rp_g$, we need to only perform $N_{\text{GST}} = O(r^{-2}p_L^{-1})$ samplings.

APPENDIX D: SURFACE CODE AND LATTICE SURGERY

While the scope of our proposal is not limited to a specific architecture of FTQC, as an example, let us consider FTQC with surface codes and lattice surgery. Surface code [49,67] is one of the most promising quantum error-correcting codes for integrated devices such as superconducting qubits. This is because surface code has a large threshold value, its stabilizer measurements can be done in a short and constant depth, and it requires physical qubits that are allocated in a two-dimensional grid and interact
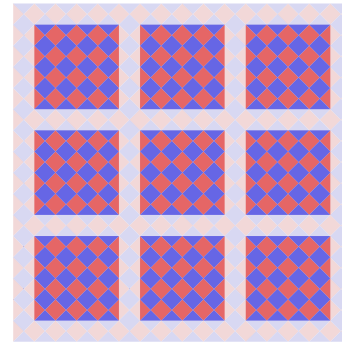

FIG. 11. Schematic diagram of nine logical qubits for $d = 5$ surface code. The vertices of the blue and red squares correspond to physical qubits. Each boldly colored patch consisting of blue and red squares represents a single logical qubit. In each logical qubit, the bold red (blue) squares represent a Pauli-Z (Pauli-X) stabilizer operator acting on their vertices.

only with the nearest-neighboring ones. An array of logical qubits is shown in Fig. 11. There are nine boldly colored patches in the figure, each of which corresponds to a logical qubit. The data qubits are allocated on the vertices of the boldly colored red and blue squares. The red (blue) squares in each patch correspond to a stabilizer operator, which acts on their vertices as a Pauli-Z (Pauli-X) operator. The width of each boldly colored patch d is equivalent to the code distance. Thus, we use $O(d^2)$ physical qubits per logical qubit for surface codes with code distance d. Note that while we can reduce the number of physical qubits by using rotated surface codes [6], we use the surface codes shown in Fig. 11 for the sake of a simple numerical simulation.

Meanwhile, lattice surgery [6,7] is a method to increase the number of logical qubits and perform logical two-qubit gates with physical qubits in a planer topology. When we prepare logical qubits as patches, we can perform a Hadamard gate transversally. Although a logical CNOT gate is also a transversal in surface codes, we cannot perform it fault tolerantly in planer topology. Instead, we implement multiqubit Pauli measurements by merging and splitting patches corresponding to logical qubits. By using multiqubit Pauli measurements and feed-forward operations, we can indirectly perform logical two-qubit Clifford gates fault tolerantly. For nontransversal operations, we can perform them by injecting, distilling, and performing gate teleportation with magic states, $|A\rangle$ and $|Y\rangle = SH|0\rangle$ where $S = \exp[i(\pi/4)Z]$. Using these magic states and the technique of gate teleportation, we can indirectly apply nontransversal operations such as S gate and T gate. While we can perform an S gate using $|Y\rangle$ without consuming it, a magic state $|A\rangle$ is consumed whenever a T gate is performed. Therefore, the number of required magic states for T gate is a dominant factor in the execution time.

Although the above-mentioned strategy of FTQC was used in the main text, there are several other possible strategies choosing codes and logical operations that improve efficiency and feasibility. For instance, we can also construct logical qubits as defect pairs in a single large patch and perform two-qubit Clifford gates with braiding [5]. We can also use concatenated Steane codes or color codes if we can perform CNOT operations more flexibly. An S gate can be achieved with code deformation [68] instead of magic state injection and distillation. We can choose CCZ gate as a magic state instead of a T gate [69]. We can estimate the recovery operations by using algorithms that achieve small latencies through the use of threshold degradation [15,48]. In any case, our method is general, and we expect that it is practical.

APPENDIX E: CONCRETE PROCESS FOR LOGICAL OPERATIONS

In this section, we explicitly show how our framework works by updating the quantum state of experimental devices and the Pauli frame, i.e., a classical memory for storing and updating Pauli operators for obtaining error-corrected results via postprocessing of measurement outcomes. Note that there is latency in collecting sufficient information to correct quantum states since there are errors in syndrome measurements and we cannot instantaneously perform feedback operations for error correction; the required recovery Pauli operations need to be continuously updated in the Pauli frame classically because unitary Clifford operators are performed for computation while storing the outcomes of the syndrome measurements. Finally, we obtain the error-corrected results from the outcomes of postprocessed measurements based on the state of the Pauli frame.

First, we describe a typical construction of FTQC without QEM as studied in Refs. [7,37]. We reformulate it with the superoperator representation so that we can introduce notations that will simplify the description of our framework. Then, we explain the FTQC framework incorporating both QEC and QEM. In this framework, the decoding errors are assumed to be not negligible but can be mitigated by probabilistic error cancellation in the logical space.

In FTQC, the process of making syndrome measurements on all the logical qubits is called a *code cycle*. For reliable decoding, we need to wait for several cycles depending on the code distance before processing the next logical operation. Here, we call the unit of latency for the slowest logical operation (i.e., the maximum number of cycles that are required before being ready for the next logical operation) a *step* and assume that in each logical operation, every logical qubit waits until all the logical qubits become ready for the next logical operation. This leads to a simple definition of the states of the quantum

devices and the classical memory at a certain step. We should emphasize that this unit is introduced only for the sake of illustration and that our scheme can be applied to asynchronous processing of logical operations.

1. FTQC architecture without QEM

At the tth step of FTQC without QEM, we have to update two components: a quantum device of which the state is $|\rho_{\mathrm{dev}}^{(t)}\rangle\rangle$ and the Pauli frame $\mathcal{P}_{\mathrm{PF}}^{(t)}$. Let us denote the Pauli transfer matrix of an actual quantum process that suffers noise until the tth step as $\mathcal{U}_{\mathrm{dev}}^{(t)}$ and an ideal initial state as $|\rho_0\rangle\rangle$. Then, the state at the tth step can be represented as $|\rho_{\mathrm{dev}}^{(t)}\rangle\rangle = \mathcal{U}_{\mathrm{dev}}^{(t)}|\rho_0\rangle\rangle$. Note that $\mathcal{U}_{\mathrm{dev}}^{(t)}$ is the superoperator of a trace-preserving and completely positive map, but it is not a unitary process, since the actual process requires several intermediate measurements. Denote the ideal operations until the tth step as $\tilde{\mathcal{U}}^{(t)}$. Here, we aim to have

$$\mathcal{P}_{\mathrm{PF}}^{(t)}\mathcal{U}_{\mathrm{dev}}^{(t)} = \tilde{\mathcal{U}}^{(t)}, \tag{E1}$$

for an arbitrary step t. Note that since several successive syndrome values are required to calculate $\mathcal{P}_{\mathrm{PF}}^{(t)}$, the Pauli frame at the tth step turns out with a latency in cycles that is at least proportional to the code distance [37].

We focus on a simplified universal set of logical operations, including preparation of logical $|0_L\rangle\rangle$ and $|A_L\rangle\rangle$, logical Clifford operations including logical Pauli operations, logical single-qubit Pauli-Z measurements, and logical gate teleportation for performing a logical T gate. While these logical operations can be divided into more basic logical operations such as merge-and-split operations in lattice surgery [6], we use this set for simplicity. In the following, we illustrate how the physical states and the Pauli frame are updated in each case.

a. Preparation of logical states

There are two types of initialization in FTQC: preparation of a logical $|0_L\rangle\rangle$ state and a logical magic state $|A_L\rangle\rangle$ implemented in the surface code. Here, we describe an operation to add a clean logical qubit:

$$\begin{aligned} \mathcal{Q}_0 |\rho\rangle\rangle &:= |\rho\rangle\rangle \otimes |0_L\rangle\rangle, \\ \mathcal{Q}_A |\rho\rangle\rangle &:= |\rho\rangle\rangle \otimes |A_L\rangle\rangle. \end{aligned} \tag{E2}$$

Let us first explain the procedure for preparing $|0_L\rangle$. We join n physical qubits to a system, where n is the number of qubits used to construct a logical qubit. The initial state of the joined data qubits can be any random state. We measure all the joined data qubits in the Pauli-Z basis. If there is no error in this measurement, we obtain a computational basis $|\mathbf{x}\rangle$, where $\mathbf{x} \in \{0, 1\}^n$ is n-bit outcomes. We suppose a Pauli operator $P_{\mathbf{x}} = \bigotimes_i X^{x_i}$; then we find

$P_{\mathbf{x}}|\mathbf{x}\rangle = |0\rangle^{\otimes n}$. The state $|0\rangle^{\otimes n}$ is the $+1$ eigenstate of all the Z-stabilizer operators and the logical Pauli-Z operator, but not an eigenstate of any X-stabilizer operator. We perform X-stabilizer measurements to project $|0\rangle^{\otimes n}$ to the code space. If the state is projected to the $+1$ eigenspace for all the X-stabilizer operators, the state becomes the $+1$ eigenstate of all the stabilizer operators and the logical Pauli-Z operator, which is the definition of $|0_L\rangle$. If any of them is -1, we can find a Pauli-Z operator $P_{\mathbf{z}}$, which anticommutes with all the X-stabilizer operators with -1 outcome and commutes with the other stabilizer operators. Accordingly, we have that $P_{\mathbf{z}}P_{\mathbf{x}}|\psi\rangle$, where $|\psi\rangle$ is the state after X-stabilizer measurements on $|\mathbf{x}\rangle$, is equal to $|0\rangle_L$. The procedure to add $|0_L\rangle$ is as follows: suppose that $\mathcal{U}_{\mathrm{dev}}^{(t+1)}$ is a sequence to join n qubits; measure all of them in the Pauli-Z basis, and perform X-stabilizer measurements. The Pauli frame $\mathcal{P}_{\mathrm{PF}}^{(t+1)}$ is the tensor product of $\mathcal{P}_{\mathrm{PF}}^{(t)}$ and the superoperator of $P_{\mathbf{z}}P_{\mathbf{x}}$. In practice, Pauli errors may occur during the above measurements. Since the error rates are expected to be below the threshold value, they can be reliably detected in the succeeding stabilizer measurements and corrected by updating $P_{\mathbf{z}}P_{\mathbf{x}}$. See Sec. c for the update of the Pauli frame with latency.

The other initialization is the preparation of the logical magic state $|A_L\rangle$, which is, as detailed later, used for performing non-Clifford gates fault tolerantly via gate teleportation. In order to use magic states for gate teleportation, the infidelity of the logical magic state must be comparable to or smaller than the required logical error rate. On the other hand, since $|A_L\rangle$ is not an eigenstate of the logical Pauli-Z operator or logical Pauli-X operator, we cannot use the same method we used above for preparing $|0_L\rangle$. Instead, a logical magic state $|A_L\rangle$ with sufficient fidelity for gate teleportation can be constructed as follows: (I) create a noisy magic state $|A_L\rangle$ whose code distance is a small constant d_s, (II) expand the code distance from d_s to d fault tolerantly, and (III) create a clean magic state from several noisy magic states with magic state distillation. If the fidelity of noisy magic states is sufficiently large and a magic state is sufficiently distilled, we can obtain a clean magic state together with its Pauli frame. See Refs. [7,37,70] for details on these procedures.

Thus, we can perform an update $\tilde{\mathcal{U}}^{(t+1)} = \mathcal{Q}\tilde{\mathcal{U}}^{(t)}$, where $\mathcal{Q} \in \{\mathcal{Q}_0, \mathcal{Q}_A\}$, by updating $\mathcal{U}_{\mathrm{dev}}^{(t)}$ and $\mathcal{P}_{\mathrm{PF}}^{(t)}$ as

$$\begin{aligned}\mathcal{U}_{\mathrm{dev}}^{(t+1)} &= (\mathcal{I} \otimes \mathcal{P}_{\mathrm{PF}}^{(Q)})\mathcal{Q}\mathcal{U}_{\mathrm{dev}}^{(t)}, \\ \mathcal{P}_{\mathrm{PF}}^{(t+1)} &= \mathcal{P}_{\mathrm{PF}}^{(t)} \otimes \mathcal{P}_{\mathrm{PF}}^{(Q)},\end{aligned} \quad (E3)$$

where $\mathcal{P}_{\mathrm{PF}}^{(Q)}$ turns out with a latency because of noisy syndrome measurements.

b. Logical Pauli operation

Logical Pauli operations are special cases of logical Clifford operations and can be performed much more easily than general Clifford operations because of the Pauli frame. When we perform a logical Pauli operation $\mathcal{P}^{(t)}$ at the tth step, we need to construct $\mathcal{P}_{\mathrm{PF}}^{(t+1)}$ and $\mathcal{U}_{\mathrm{dev}}^{(t+1)}$ such that $\mathcal{P}_{\mathrm{PF}}^{(t+1)}\mathcal{U}_{\mathrm{dev}}^{(t+1)} = \tilde{\mathcal{U}}^{(t+1)} = \mathcal{P}^{(t)}\tilde{\mathcal{U}}^{(t)}$. Since a logical Pauli operation is a product of physical Pauli operations, there are two ways to achieve this. One is to update only the Pauli frame as follows:

$$\begin{aligned}\mathcal{P}_{\mathrm{PF}}^{(t+1)} &= \mathcal{P}^{(t)}\mathcal{P}_{\mathrm{PF}}^{(t)}, \\ \mathcal{U}_{\mathrm{dev}}^{(t+1)} &= \mathcal{U}_{\mathrm{dev}}^{(t)}.\end{aligned} \quad (E4)$$

Note that this is an instantaneous and noiseless operation because it is processed only on the classical computer. We refer to this scheme as a Pauli operation by *software update*.

The other is to update the physical device instead of the Pauli frame.

$$\mathcal{P}_{\mathrm{PF}}^{(t+1)} = \mathcal{P}_{\mathrm{PF}}^{(t)}, \quad (E5)$$

$$\mathcal{U}_{\mathrm{dev}}^{(t+1)} = \mathcal{P}^{(t)}\mathcal{U}_{\mathrm{dev}}^{(t)}. \quad (E6)$$

Compared with the software update, this operation requires operation of the physical quantum devices, and hence, it may involve an error. We call this scheme a *hardware update*. Note that this procedure is a transversal single-qubit Pauli operation, so we expect that it negligibly increases the physical error rate per code cycle. Although the hardware update seems to have no advantage in a typical architecture of FTQC, logical Pauli operations by hardware update are required in the Pauli twirling of logical errors in FTQC, as described in Appendix F.

c. Logical Clifford operation

Unlike logical Pauli operations, Clifford operations require physical operations to be performed on quantum devices. Here, we denote the Pauli transfer matrix of physical errors at the tth step as $\mathcal{P}_{\mathrm{rec}}^{(t)}$. Note that since quantum states are projected to the code space with syndrome measurements, we expect that $\mathcal{P}_{\mathrm{rec}}^{(t)}$ can be approximated as a Pauli error. For the case that we cannot assume $\mathcal{P}_{\mathrm{rec}}^{(t)}$ to be a Pauli error, we can twirl it with stochastic Pauli operations by hardware update and project it to the stochastic Pauli errors. See Appendix F for details.

Pauli errors are detected via syndrome measurements. Furthermore, the recovery Pauli operation $\mathcal{P}_{\mathrm{rec}}^{(t)}$ can be estimated with a high success probability and the Pauli frame can be updated in the same way as Eq. (E4). If there are no measurement errors, the recovery operation for a certain cycle can be estimated only from the syndrome

values during the same cycle. However, when the syndrome measurements suffer physical errors, we need to idle the logical qubits for d cycles to collect sufficient syndrome measurement outcomes before starting the next Clifford operation to guarantee exponential decay of the logical error rate [37]. Therefore, the recovery operations turn out with a latency of at least d cycles. In practice, the latency is larger than d cycles since we also need postprocessing for signal discrimination and of the decoding algorithms on classical peripherals.

Suppose that we perform a logical Clifford operation $\mathcal{C}^{(t)}$ and that physical Pauli errors $\mathcal{P}_{\text{rec}}^{(t)}$ happen during the Clifford operation, which appear after the decoding. Here, we can ignore recovery failures when the code distance is sufficiently large. We aim to obtain $\mathcal{P}_{\text{PF}}^{(t+1)}$ and $\mathcal{U}_{\text{dev}}^{(t+1)}$ that satisfy $\mathcal{P}_{\text{PF}}^{(t+1)}\mathcal{U}_{\text{dev}}^{(t+1)} = \tilde{\mathcal{U}}^{(t+1)} = \mathcal{C}^{(t)}\tilde{\mathcal{U}}^{(t)}$. A logical Clifford operation updates the state of the quantum devices as follows:

$$\mathcal{U}_{\text{dev}}^{(t+1)} = \mathcal{P}_{\text{rec}}^{(t)}\mathcal{C}^{(t)}\mathcal{U}_{\text{dev}}^{(t)}. \tag{E7}$$

In addition, the Pauli frame is updated as follows:

$$\mathcal{P}_{\text{PF}}^{(t+1)} = \mathcal{C}^{(t)}\mathcal{P}_{\text{PF}}^{(t)}(\mathcal{C}^{(t)})^{-1}\mathcal{P}_{\text{rec}}^{(t)}. \tag{E8}$$

Since a logical Clifford operation is also a Clifford operation on physical qubits in the case of stabilizer codes, $\mathcal{P}_{\text{PF}}^{(t+1)}$ is a Pauli operator.

In practice, $\mathcal{P}_{\text{rec}}^{(t)}$ is expected to have a large latency since it requires postprocessing to estimate the recovery operation. Thus, another Pauli error might happen during the idling operations performed while waiting for the estimation of $\mathcal{P}_{\text{rec}}^{(t)}$. Nevertheless, the next logical Clifford operation or logical Pauli measurement can be performed before $\mathcal{P}_{\text{rec}}^{(t)}$ is estimated. As an example, consider the case in which $\mathcal{C}^{(t+1)}$ is processed before the last Pauli frame is updated. The Pauli frame of the next step is represented as

$$\begin{aligned}\mathcal{P}_{\text{PF}}^{(t+2)} &= \mathcal{P}_{\text{rec}}^{(t+1)}\mathcal{C}^{(t+1)}\mathcal{P}_{\text{PF}}^{(t+1)}(\mathcal{C}^{(t+1)})^{-1}\\&= [\mathcal{C}^{(t+1)}\mathcal{C}^{(t)}\mathcal{P}_{\text{PF}}^{(t)}(\mathcal{C}^{(t)})^{-1}(\mathcal{C}^{(t+1)})^{-1}]\\&\quad\times[\mathcal{C}^{(t+1)}\mathcal{P}_{\text{rec}}^{(t)}(\mathcal{C}^{(t+1)})^{-1}]\mathcal{P}_{\text{rec}}^{(t+1)}.\end{aligned} \tag{E9}$$

Since Pauli actions commute, we can update the part $\mathcal{C}^{(t+1)}\mathcal{P}_{\text{rec}}^{(t)}(\mathcal{C}^{(t+1)})^{-1}$ even in the next step. We can use a similar technique when the next operation is a logical Pauli measurement, which is explained in the next section. Therefore, as long as all the successive operations are Clifford operations or Pauli measurements, we can postpone the application of the recovery operators and updates of the Pauli frame. This technique is essential not only for improving the throughput of logical operations in FTQC but also for avoiding exponential growth in latency [15].

Note that in the lattice surgery scheme, logical CNOTs are performed by preparing a logical $|0_L\rangle\rangle$ state, making

logical Pauli measurements, and adaptive Pauli operations [7]. While none of them is a Clifford operation, we can consider them on the whole to constitute a unitary process. Adaptive Pauli operations can be postponed for the same reason as the updates of the Pauli frame.

d. Single-qubit logical Pauli measurement

We can perform a destructive single-qubit logical Pauli-Z measurement by making physical Pauli-Z measurements on all the data qubits of the target logical qubit. Note that logical Pauli measurements with another Pauli basis can be performed by swapping Z and X or combining the logical Pauli-Z measurement with single-qubit logical Clifford operations. The outcome of a logical measurement is calculated as the parity of the measurement outcomes of the data qubits. In other words, let n be the number of data qubits of the target logical qubit, and $\Pi_{\mathbf{x}} = \prod_i[I + (-1)^{x_i}Z_i]/2^n$ for $\mathbf{x} \in \{0,1\}^n$ be an operator that projects all the data qubits in a physical computational basis $|\mathbf{x}\rangle$. A single-bit outcome of a single-qubit logical measurement is calculated as a parity of the outcomes of physical measurements $\mathbf{x}$. We denote this function as $f : \{0,1\}^n \to \{0,1\}$. Therefore, we aim to estimate $\mathbf{x}$ and obtain $f(\mathbf{x})$ fault tolerantly.

When we perform physical Pauli-Z measurements, physical Pauli errors $\mathcal{P}_{\text{rec}}^{(t)}$ occur on the data qubits. Also, as explained in the section on logical Clifford operations, the Pauli frame $\mathcal{P}_{\text{PF}}^{(t)}$ does not have the same timing as the logical measurement. Even in this case, the recovery operation can be applied after the physical Pauli-Z measurements have been performed on the data qubits, since

$$\begin{aligned}\langle\langle\Pi_{\mathbf{x}}|\mathcal{P}_{\text{rec}}^{(t)}\mathcal{U}_{\text{dev}}^{(t)} &= \langle\langle\Pi_{\mathbf{x}\oplus\mathbf{y}}|\mathcal{P}_{\text{PF}}^{(t)}\mathcal{U}_{\text{dev}}^{(t)}\\&= \langle\langle\Pi_{\mathbf{x}\oplus\mathbf{y}}|\tilde{\mathcal{U}}^{(t)},\end{aligned} \tag{E10}$$

where $\oplus$ represents an elementwise XOR, and $\mathbf{y} \in \{0,1\}^n$ is a binary vector such that $y_i = 1$ if $\mathcal{P}_{\text{PF}}^{(t)}\mathcal{P}_{\text{rec}}^{(t)}$ acts on the ith data qubits as a Pauli-X or Y operator and $y_i = 0$ otherwise. We denote this function as $\mathbf{y} = \text{mask}(\mathcal{P}_{\text{PF}}^{(t)}\mathcal{P}_{\text{rec}}^{(t)})$.

Therefore, we obtain $\mathbf{x}$ as an outcome of the measurement on the uncorrected state $|\rho_{\text{dev}}^{(t)}\rangle\rangle$. Notice that $\mathbf{y}$ appears when the decoding process catches up with the code cycle of the logical measurements, and we can retrieve the error-corrected logical outcome as $f[\mathbf{x} \oplus \text{mask}(\mathcal{P}_{\text{PF}}^{(t)}\mathcal{P}_{\text{rec}}^{(t)})]$. The update rules can be written as

$$\mathcal{U}_{\text{dev}}^{(t+1)} = \frac{1}{p_{\mathbf{x}}}\langle\langle\Pi_{\mathbf{x}}|\mathcal{P}_{\text{rec}}^{(t)}\mathcal{U}_{\text{dev}}^{(t)}, \tag{E11}$$

$$\mathcal{P}_{\text{PF}}^{(t+1)} = \text{Dis}_M[\mathcal{P}_{\text{PF}}^{(t)}], \tag{E12}$$

where $p_{\mathbf{x}}$ is the probability with which we obtain $\mathbf{x}$ and $\text{Dis}_M[\cdot]$ is an operation to discard the Pauli frame corresponding to the measured logical qubits. In contrast

to the logical initialization, the update for the logical measurement will reduce the space of the target logical qubit.

Several things should be noted in regard to the decoding algorithm at the measurement timing. The Pauli frame can be represented in the form $(\bigotimes_i X^{x_i} Z^{z_i})$ for $x_i, z_i \in \{0, 1\}$. We call $\{x_i\}$ and $\{z_i\}$ as the X part and Z part of the Pauli frame, respectively. Since all the data qubits are directly measured in this logical operation, syndrome measurements are not performed in this code cycle. Instead, we calculate the values of the Z-stabilizer syndrome measurements without measurement errors as parities of **x** because we directly measure the data qubits without resorting to ancilla qubits. Since there is no effective measurement error in the Z-stabilizer syndrome measurements at the cycle of the logical measurement, an estimation of the X part of $\mathcal{P}_{\text{PF}}^{(t)} \mathcal{P}_{\text{rec}}^{(t)}$ can be converted into an instance of a graph problem called minimum-weight perfect matching [37]. On the other hand, the information required to construct a perfect-matching problem for estimating the Z part of the Pauli frame is lost by making direct Z-basis measurements. Nevertheless, this does not affect the computation since only the X part of the Pauli frame is relevant to determining $\mathbf{y} = \text{mask}(\mathcal{P}_{\text{PF}}^{(t)} \mathcal{P}_{\text{rec}}^{(t)})$.

e. Gate teleportation with magic state

For a universal FTQC, we need to perform non-Clifford operations on logical qubits encoded in surface code. To this end, in a typical scenario, we create a logical qubit prepared in $|A_L\rangle\rangle$ and perform T gate on the target system by consuming the magic state $|A_L\rangle\rangle$.

First, we describe gate teleportation without noise. Let $|\rho\rangle\rangle \otimes |A\rangle\rangle$ be the tensor product of the target system and a magic state. It is known that the following equation holds for an arbitrary ρ [37,71]:

$$
\begin{aligned}
(\mathcal{S}^{f(\mathbf{x})} &\otimes \langle\langle\Pi_{\mathbf{x}}|)\Lambda \mathcal{Q}_A |\rho\rangle\rangle \\
&= (\mathcal{S}^{f(\mathbf{x})} \otimes \langle\langle\Pi_{\mathbf{x}}|)\Lambda(|\rho\rangle\rangle \otimes |A_L\rangle\rangle) \\
&= \mathcal{T} |\rho\rangle\rangle\langle\langle\Pi_{\mathbf{x}}|+_L\rangle\rangle,
\end{aligned} \tag{E13}
$$

where Λ is the Pauli transfer matrix of a logical CNOT gate acting on $|\rho\rangle\rangle$ as a control and $|A_L\rangle\rangle$ as a target, $\mathcal{T}$ is the Pauli transfer matrix of the T gate, and $\mathcal{S}^{f(\mathbf{x})}$ is the Pauli transfer matrix of the adaptive S gate that is applied to the state if $f(\mathbf{x}) = 1$ (see the last section for the definition of the function f).

Using the above fact, we can construct the following update rules for the target system:

$$
\mathcal{U}_{\text{dev}}^{(t+1)} = \frac{1}{p_{\mathbf{x}}}(\mathcal{S}^{f(\mathbf{x}\oplus\mathbf{y})} \otimes \langle\langle\Pi_{\mathbf{x}}|)\mathcal{P}_{\text{rec}}^{(t)}\Lambda(\mathcal{I} \otimes \mathcal{P}_{\text{PF}}^{(Q)})\mathcal{Q}_A\mathcal{U}_{\text{dev}}^{(t)}
$$
$$\tag{E14}$$
$$
\mathcal{P}_{\text{PF}}^{(t+1)} = \mathcal{S}^{f(\mathbf{x}\oplus\mathbf{y})}\text{Dis}_Q[\Lambda(\mathcal{P}_{\text{PF}}^{(t)} \otimes \mathcal{P}_{\text{PF}}^{(Q)})\Lambda^{-1}\mathcal{P}_{\text{rec}}^{(t)}]\mathcal{S}^{-f(\mathbf{x}\oplus\mathbf{y})},
$$
$$\tag{E15}$$

where $\mathcal{P}_{\text{rec}}^{(t)}$ represents physical detectable errors,

$$
p_{\mathbf{x}} = \langle\langle\Pi_{\mathbf{x}\oplus\mathbf{y}}|+_L\rangle\rangle, \tag{E16}
$$

$$
\mathbf{y} = \text{mask}\{\text{Dis}_\rho[\Lambda(\mathcal{P}_{\text{PF}}^{(t)} \otimes \mathcal{P}_{\text{PF}}^{(Q)})\Lambda^{-1}\mathcal{P}_{\text{PF}}^{(t)}]\}, \tag{E17}
$$

and $\text{Dis}_Q[\cdot]$ ($\text{Dis}_\rho[\cdot]$) is an operation to discard the Pauli frame of the magic state (target state). Then, we can verify that the above update rule obeys Eq. (E4), as follows:

$$
\begin{aligned}
\mathcal{P}_{\text{PF}}^{(t+1)}\mathcal{U}_{\text{dev}}^{(t+1)} &= \frac{1}{p_{\mathbf{x}}}\mathcal{S}^{f(\mathbf{x}\oplus\mathbf{y})}\text{Dis}_Q[\Lambda(\mathcal{P}_{\text{PF}}^{(t)} \otimes \mathcal{P}_{\text{PF}}^{(Q)})\Lambda^{-1}\mathcal{P}_{\text{rec}}^{(t)}]\mathcal{S}^{-f(\mathbf{x}\oplus\mathbf{y})} \\
&\quad \times (\mathcal{S}^{f(\mathbf{x}\oplus\mathbf{y})} \otimes \langle\langle\Pi_{\mathbf{x}}|)\mathcal{P}_{\text{rec}}^{(t)}\Lambda(\mathcal{I} \otimes \mathcal{P}_{\text{PF}}^{(Q)})\mathcal{Q}_A\mathcal{U}_{\text{dev}}^{(t)} \\
&= \frac{1}{p_{\mathbf{x}}}(\mathcal{S}^{f(\mathbf{x}\oplus\mathbf{y})} \otimes \langle\langle\Pi_{\mathbf{x}}|)\text{Dis}_Q[\Lambda(\mathcal{P}_{\text{PF}}^{(t)} \otimes \mathcal{P}_{\text{PF}}^{(Q)})\Lambda^{-1}\mathcal{P}_{\text{rec}}^{(t)}]\mathcal{P}_{\text{rec}}^{(t)}\Lambda(\mathcal{I} \otimes \mathcal{P}_{\text{PF}}^{(Q)})\mathcal{Q}_A\mathcal{U}_{\text{dev}}^{(t)} \\
&= \frac{1}{p_{\mathbf{x}}}(\mathcal{S}^{f(\mathbf{x}\oplus\mathbf{y})} \otimes \langle\langle\Pi_{\mathbf{x}\oplus\mathbf{y}}|)[\Lambda(\mathcal{P}_{\text{PF}}^{(t)} \otimes \mathcal{P}_{\text{PF}}^{(Q)})\Lambda^{-1}\mathcal{P}_{\text{rec}}^{(t)}]\mathcal{P}_{\text{rec}}^{(t)}\Lambda(\mathcal{I} \otimes \mathcal{P}_{\text{PF}}^{(Q)})\mathcal{Q}_A\mathcal{U}_{\text{dev}}^{(t)} \\
&= \frac{1}{p_{\mathbf{x}}}(\mathcal{S}^{f(\mathbf{x}\oplus\mathbf{y})} \otimes \langle\langle\Pi_{\mathbf{x}\oplus\mathbf{y}}|)\Lambda\mathcal{Q}_A\mathcal{P}_{\text{PF}}^{(t)}\mathcal{U}_{\text{dev}}^{(t)} \\
&= \mathcal{T}\tilde{\mathcal{U}}^{(t)} = \tilde{\mathcal{U}}^{(t+1)}. \tag{E18}
\end{aligned}
$$

Note that we use the following property of the discard operation for an arbitrary Pauli operation $\mathcal{P}$:

$$
\langle\langle\Pi_{\mathbf{x}}| = \langle\langle\Pi_{\mathbf{x}\oplus\text{mask}(\mathcal{P})}|\mathcal{P} \tag{E19}
$$

$$
\mathcal{P} = \text{Dis}_Q[\mathcal{P}]\text{Dis}_\rho[\mathcal{P}]. \tag{E20}
$$

Note as well that **y** has latency due to the decoding process and the $(\mathcal{S}^{f(\mathbf{x})}$ and $\langle\langle\Pi_{\mathbf{x}}|)$ operations cannot be

performed simultaneously. Thus, there is an additional delay in determining $f(\mathbf{x} \oplus \mathbf{y})$, which is required for determining whether we perform $\mathcal{S}$ or not.

2. FTQC architecture with QEM

The failure probability of decoding and residual errors on the prepared magic states are not negligible when the code distance of quantum error correction is not enough and when we cannot perform sufficient magic state distillation. Here, let us consider the case that a decoding error $\mathcal{N}$ happens after each elementary logical operation. We assume that $\mathcal{N}$ is a stochastic logical Pauli channel that is characterized in advance. While these assumptions do not strictly hold in practice, we nonetheless expect that they hold with negligible errors in typical quantum devices. See Appendix F for a justification of this assumption. Note that even if there is a finite discrepancy because of this approximation, our scheme can decrease bias in the expectation values as long as the discrepancy is small. In this section, we show that probabilistic error cancellation can be integrated into the FTQC architecture. More concretely, we show that each logical operation in the previous section can be modified so that probabilistic error cancellation can remove the residual noise of QEC only with additional logical Pauli operations by software update.

Unlike in the NISQ scenario, FTQC operations are probabilistic, since intermediate measurements are involved in the gate teleportation of the magic states, and subsequent operations are adaptively chosen corresponding to the measurement outcomes. We need to determine QEM operations accordingly because decoding of the noise processes may change depending on the measurement outcomes. Here, let us denote the set of outcomes of intermediate measurements and QEM operations up to the tth step as $\mathbf{h}(t)$, with the corresponding Pauli frame and the state of the hardware denoted as $\mathcal{P}_{\mathrm{PF}}^{[\mathbf{h}(t)]}$ and $\mathcal{U}_{\mathrm{dev}}^{[\mathbf{h}(t)]}$. Note that we do not independently define the measurement outcomes and QEM operations since they affect each other. The probability that $\mathbf{h}(t)$ can be expressed as $p_{\mathrm{h}(t)}q_{\mathrm{h}(t)}$, where $p_{\mathrm{h}(t)}$ is the probability with which a certain measurement outcome of intermediate measurements is observed and $q_{\mathrm{h}(t)}$ is the probability with which a certain QEM operation is performed. Note that, these probabilities are functions of $\mathbf{h}(t)$. Furthermore, we denote the parity of the QEM operation (the product of the parities of the generated QEM operations) and the QEM cost at the tth step as $s^{[\mathbf{h}(t)]}$ and $\gamma_Q^{[\mathbf{h}(t)]}$.

In our framework, we can construct a consistent FTQC architecture incorporating QEM by satisfying the following equation:

$$\sum_{\mathbf{h}(t)} q_{\mathrm{h}(t)}p_{\mathrm{h}(t)}\gamma_Q^{[\mathbf{h}(t)]}s^{[\mathbf{h}(t)]}\mathcal{P}_{\mathrm{PF}}^{[\mathbf{h}(t)]}\mathcal{U}_{\mathrm{dev}}^{[\mathbf{h}(t)]} = \tilde{\mathcal{U}}^{(t)}. \tag{E21}$$

Here, we explain why satisfying the above equation is sufficient. If we can construct $\mathcal{U}_{\mathrm{dev}}^{[\mathbf{h}(t)]}$ as a product of physical processes, we can sample the density matrix $|\rho^{[\mathbf{h}(t)]}\rangle\rangle = \mathcal{P}_{\mathrm{PF}}^{[\mathbf{h}(t)]}\mathcal{U}_{\mathrm{dev}}^{[\mathbf{h}(t)]}|\rho_0\rangle\rangle$ with probability $q_{\mathrm{h}(t)}p_{\mathrm{h}(t)}$. Suppose the spectral decomposition of a given observable O is $O = \sum_i O_i \Pi_i$ ($O_i \in \mathbb{R}$). Sampling from the final state with positive operator-valued measure (POVM) elements $\{\Pi_i\}$ with weight $\gamma_Q^{[\mathbf{h}(t)]}s^{[\mathbf{h}(t)]}O_i$ gives

$$\sum_{\mathbf{h}(t),i} q_{\mathrm{h}(t)}p_{\mathrm{h}(t)}\gamma_Q^{[\mathbf{h}(t)]}s^{[\mathbf{h}(t)]}O_i\langle\langle\Pi_i|\rho^{[\mathbf{h}(t)]}\rangle\rangle$$

$$= \langle\langle O|\sum_{\mathbf{h}(t)} q_{\mathrm{h}(t)}p_{\mathrm{h}(t)}\gamma_Q^{[\mathbf{h}(t)]}s^{[\mathbf{h}(t)]}\mathcal{P}_{\mathrm{PF}}^{[\mathbf{h}(t)]}\mathcal{U}_{\mathrm{dev}}^{[\mathbf{h}(t)]}|\rho_0\rangle\rangle$$

$$= \langle\langle O|\tilde{\mathcal{U}}^{(t)}|\rho_0\rangle\rangle, \tag{E22}$$

which shows that QEM can recover an unbiased expectation value of the observables when Eq. (E21) holds.

a. Logical Pauli operation

Since logical Pauli operations by software update are instantaneous and noiseless, we do not need to perform error mitigation on them.

b. Logical Clifford operation

Suppose that we want to apply a Clifford operation $\mathcal{C}$ at the $(t+1)$th step; here, $\mathcal{C}$ is followed by not only detected physical Pauli errors $\mathcal{P}_{\mathrm{rec}}^{(t+1)}$ but also logical Pauli noise $\mathcal{N}$ reflecting a decoding failure with a non-negligible probability; in other words, the quantum device is updated as

$$\mathcal{U}_{\mathrm{dev}}^{[\mathbf{h}(t+1)]} = \mathcal{P}_{\mathrm{rec}}^{(t)}\mathcal{N}\mathcal{C}\mathcal{U}_{\mathrm{dev}}^{[\mathbf{h}(t)]}. \tag{E23}$$

While $\mathcal{P}_{\mathrm{rec}}^{(t)}$ is revealed with latency and canceled by updating the Pauli frame, $\mathcal{N}$ is not canceled and affects the expectation values without QEM. Here, we show that we can cancel $\mathcal{N}$ by applying QEM with a probabilistic update of the Pauli frame. Our goal is to find a set of update rules $(\gamma_Q^{[\mathbf{h}(t)]}, s^{[\mathbf{h}(t)]}, \mathcal{P}_{\mathrm{PF}}^{[\mathbf{h}(t)]}, \mathcal{U}_{\mathrm{dev}}^{[\mathbf{h}(t)]}) \mapsto (\gamma_Q^{[\mathbf{h}(t+1)]}, s^{[\mathbf{h}(t+1)]}, \mathcal{P}_{\mathrm{PF}}^{[\mathbf{h}(t+1)]}, \mathcal{U}_{\mathrm{dev}}^{[\mathbf{h}(t+1)]})$, which satisfies Eq. (E21) for

$$\tilde{\mathcal{U}}^{(t+1)} = \mathcal{C}\tilde{\mathcal{U}}^{(t)}. \tag{E24}$$

Since we know the stochastic logical Pauli noise $\mathcal{N}$ in advance, we can calculate the inverse of the noise map $\mathcal{N}^{-1} = \sum_i \eta_i \mathcal{P}_i$. Next, we decompose the nonzero coefficients η_i into $\eta_i = \gamma_Q \operatorname{sgn}(\eta_i)q_i$, where $\gamma_Q = \sum_i |\eta_i|$ and $q_i = |\eta_i|/\gamma_Q$. We randomly choose i with probability q_i, and the index i is appended to give $\mathbf{h}(t+1)$. We find that $q_{\mathrm{h}(t+1)} = q_i q_{\mathrm{h}(t)}$ and $p_{\mathrm{h}(t+1)} = p_{\mathrm{h}(t)}$. Then, we update the

state of the Pauli frame and the classical coefficients with the following update rules:

$$\mathcal{P}_{\mathrm{PF}}^{[\mathrm{h}(t+1)]} = \mathcal{C}\mathcal{P}_{\mathrm{PF}}^{[\mathrm{h}(t)]}\mathcal{C}^{-1}\mathcal{P}_i\mathcal{P}_{\mathrm{rec}}^{(t)}, \quad (E25)$$

$$s^{[\mathrm{h}(t+1)]} = \mathrm{sgn}(\eta_i)s^{[\mathrm{h}(t)]}, \quad (E26)$$

$$\gamma_Q^{[\mathrm{h}(t+1)]} = \gamma_Q\gamma_Q^{[\mathrm{h}(t)]}. \quad (E27)$$

Now, we can verify that Eq. (E21) is satisfied in the $(t+1)$th step if it is satisfied in the tth step.

$$\sum_{\mathbf{h}(t+1)} q_{\mathrm{h}(t+1)}p_{\mathrm{h}(t+1)}\gamma_Q^{[\mathrm{h}(t+1)]}s^{[\mathrm{h}(t+1)]}\mathcal{P}_{\mathrm{PF}}^{[\mathrm{h}(t+1)]}\mathcal{U}_{\mathrm{dev}}^{[\mathrm{h}(t+1)]}$$

$$= \sum_{\mathbf{h}(t)} q_{\mathrm{h}(t)}p_{\mathrm{h}(t)}\gamma_Q^{[\mathrm{h}(t)]}s^{[\mathrm{h}(t)]}\mathcal{C}\mathcal{P}_{\mathrm{PF}}^{[\mathrm{h}(t)]}\mathcal{C}^{-1}\left(\sum_i \eta_i\mathcal{P}_i\right)$$

$$\times \mathcal{N}\mathcal{C}\mathcal{U}_{\mathrm{dev}}^{[\mathrm{h}(t)]}$$

$$= \mathcal{C}\sum_{\mathbf{h}(t)} q_{\mathbf{h}(t)}p_{\mathbf{h}(t)}\gamma_Q^{[\mathrm{h}(h)]}s^{[\mathrm{h}(t)]}\mathcal{P}_{\mathrm{PF}}^{[\mathrm{h}(t)]}\mathcal{U}_{\mathrm{dev}}^{[\mathrm{h}(t)]}$$

$$= \mathcal{C}\tilde{\mathcal{U}}^{(t)} = \tilde{\mathcal{U}}^{(t+1)}. \quad (E28)$$

As we see later, we can construct a similar update rule for probabilistic error cancellation for a single-qubit logical Pauli measurement; therefore, we can inductively show that a decomposition specified with the update rule satisfies Eq. (E21).

c. Logical initialization and single-qubit logical Pauli measurement

When code distances and magic state distillation processes are insufficient, errors in the logical state preparation are not negligible. The noise maps for logical $|0\rangle$ state preparation can be assumed to be stochastic logical Pauli noise. While the noise map on magic state preparation due to insufficient distillation may not be approximated as a stochastic Pauli noise, it can be twirled by error-mitigated logical Clifford operations. Thus, these errors are assumed to be inserted stochastic logical Pauli noise maps just after initialization. We can consider that these preparations are ideal, and instead, there exists a virtual noisy idling operation just after initialization. We can mitigate these errors with the same update rules as in the case of logical Clifford operations by treating these errors as decoding errors. Similarly, the errors of single-qubit logical Pauli measurement can be considered to be a probabilistic logical bit flip just before the logical measurement; as such, they can be processed in the same manner.

d. Gate teleportation with magic state

To perform gate teleportation on the T gate, we can try the following process:

$$(\mathcal{I} \otimes \langle\langle\Pi_{\mathbf{x}}|)\Lambda\mathcal{Q}_A, \quad (E29)$$

and then perform $\mathcal{S}^{f\,(\mathbf{x})}$ depending on the measurement outcomes to indirectly perform $\mathcal{T}$ on the target system. However, in practice, what is performed until the measurement is made is

$$(\mathcal{I} \otimes \langle\langle\Pi_{\mathbf{x}}|)\mathcal{N}\mathcal{P}_{\mathrm{rec}}^{(t)}\Lambda(\mathcal{I} \otimes \mathcal{P}_{\mathrm{PF}}^{(Q)})\mathcal{Q}_A, \quad (E30)$$

where $\mathcal{P}_{\mathrm{rec}}^{(t)}$ means the physical errors caused by the logical magic state preparation, logical CNOT, and logical measurements. Compared with the case without QEM Eq. (E14), there is additional logical Pauli noise $\mathcal{N}$ that is caused by the failure of the decoding to estimate $\mathcal{P}_{\mathrm{rec}}^{(t)}$.

Since this procedure involves an intermediate measurement, we obtain the outcome $\mathbf{x}$ randomly. The procedure for the inverse decomposition is the same as that of a logical Clifford operation: we calculate the inverse of the noise map $\mathcal{N}^{-1} = \sum_i \eta_i\mathcal{P}_i$ and decompose each nonzero coefficient η_i into $\eta_i = \gamma_Q\,\mathrm{sgn}(\eta_i)q_i$, where $\gamma_Q = \sum_i |\eta_i|$ and $q_i = |\eta_i|/\gamma_Q$. Note that the measurement result $\mathbf{x}$ is added to $\mathbf{h}(t)$ together with the choice of QEM operation i to obtain $\mathbf{h}(t+1)$. Next, $q_{\mathrm{h}(t)}$ is updated to $q_{\mathrm{h}(t+1)} = q_iq_{\mathrm{h}(t)}$ and $p_{\mathrm{h}(t)}$ is updated to $p_{\mathrm{h}(t+1)} = p_{\mathbf{x}}p_{\mathrm{h}(t)}$, where $p_{\mathbf{x}}$ is the probability that the measurement outcome is $\mathbf{x}$. Accordingly, we can update the relevant elements with the following rules:

$$\mathcal{U}_{\mathrm{dev}}^{[\mathrm{h}(t+1)]} = \frac{1}{p_{\mathbf{x}}}(\mathcal{S}^{f\,(\mathbf{x}\oplus\mathbf{y})} \otimes \langle\langle\Pi_{\mathbf{x}}|)\mathcal{N}\mathcal{P}_{\mathrm{rec}}^{(t)}\Lambda$$

$$\times (\mathcal{I} \otimes \mathcal{P}_{\mathrm{PF}}^{(Q)})\mathcal{Q}_A\mathcal{U}_{\mathrm{dev}}^{[\mathrm{h}(t)]}, \quad (E31)$$

$$\mathcal{P}_{\mathrm{PF}}^{[\mathrm{h}(t+1)]} = \mathcal{S}^{f\,(\mathbf{x}\oplus\mathbf{y})}\mathrm{Dis}_A[\mathcal{R}^{[\mathbf{h}(t)]}]\mathcal{S}^{-f\,(\mathbf{x}\oplus\mathbf{y})}, \quad (E32)$$

$$s^{[\mathrm{h}(t+1)]} = \mathrm{sgn}(\eta_i)s^{[\mathrm{h}(t)]}, \quad (E33)$$

$$\gamma_Q^{[\mathrm{h}(t+1)]} = \gamma_Q\gamma_Q^{[\mathrm{h}(t)]}, \quad (E34)$$

where

$$\mathcal{R}^{[\mathbf{h}(t)]} = \Lambda(\mathcal{P}_{\mathrm{PF}}^{[\mathrm{h}(t)]} \otimes \mathcal{P}_{\mathrm{PF}}^{(Q)})\Lambda^{-1}\mathcal{P}_i\mathcal{P}_{\mathrm{rec}}^{(t)}, \quad (E35)$$

$$\mathbf{y} = \mathrm{mask}(\mathrm{Dis}_\rho[\mathcal{R}^{[\mathbf{h}(t)]}]). \quad (E36)$$

Note that an additional decoding error that depends on whether we perform $\mathcal{S}$ or not would occur after the delayed application of $\mathcal{S}^{f\,(\mathbf{x}\oplus\mathbf{y})}$. While this map is omitted for simplicity, we can perform probabilistic error cancellation on it in the same way.

With the above update rules, we can verify that Eq. (E21) is obeyed at the $(t+1)$th step. The product of $\mathcal{U}_{\mathrm{dev}}^{[\mathrm{h}(t+1)]}$ and $\mathcal{P}_{\mathrm{PF}}^{[\mathrm{h}(t+1)]}$ is evaluated as follows:

$$\mathcal{P}_{\mathrm{PF}}^{[\mathbf{h}(t+1)]}\mathcal{U}_{\mathrm{dev}}^{[\mathbf{h}(t+1)]} = \frac{1}{p_{\mathbf{x}}}\mathcal{S}^{f\,(\mathbf{x}\oplus\mathbf{y})}\mathrm{Dis}_A[\mathcal{R}^{[\mathbf{h}(t)]}]\mathcal{S}^{-f\,(\mathbf{x}\oplus\mathbf{y})}(\mathcal{S}^{f\,(\mathbf{x}\oplus\mathbf{y})}\otimes\langle\langle\Pi_{\mathbf{x}}|)\mathcal{N}\mathcal{P}_{\mathrm{rec}}^{(t)}\Lambda(\mathcal{I}\otimes\mathcal{P}_{\mathrm{PF}}^{(Q)})\mathcal{Q}_A\mathcal{U}_{\mathrm{dev}}^{[\mathbf{h}(t)]}$$

$$= \frac{1}{p_{\mathbf{x}}}(\mathcal{S}^{f\,(\mathbf{x}\oplus\mathbf{y})}\otimes\langle\langle\Pi_{\mathbf{x}}|)\mathrm{Dis}_A[\mathcal{R}^{[\mathbf{h}(t)]}]\mathcal{N}\mathcal{P}_{\mathrm{rec}}^{(t)}\Lambda(\mathcal{I}\otimes\mathcal{P}_{\mathrm{PF}}^{(Q)})\mathcal{Q}_A\mathcal{U}_{\mathrm{dev}}^{[\mathbf{h}(t)]}$$

$$= \frac{1}{p_{\mathbf{x}}}(\mathcal{S}^{f\,(\mathbf{x}\oplus\mathbf{y})}\otimes\langle\langle\Pi_{\mathbf{x}\oplus\mathbf{y}}|)\mathcal{R}^{[\mathbf{h}(t)]}\mathcal{N}\mathcal{P}_{\mathrm{rec}}^{(t)}\Lambda(\mathcal{I}\otimes\mathcal{P}_{\mathrm{PF}}^{(Q)})\mathcal{Q}_A\mathcal{U}_{\mathrm{dev}}^{[\mathbf{h}(t)]}$$

$$= \frac{1}{p_{\mathbf{x}}}(\mathcal{S}^{f\,(\mathbf{x}\oplus\mathbf{y})}\otimes\langle\langle\Pi_{\mathbf{x}\oplus\mathbf{y}}|)\mathcal{P}_i\mathcal{N}\Lambda\mathcal{Q}_A\mathcal{P}_{\mathrm{PF}}^{[\mathbf{h}(t)]}\mathcal{U}_{\mathrm{dev}}^{[\mathbf{h}(t)]}. \tag{E37}$$

Therefore, the expectation value is evaluated as follows:

$$\sum_{\mathbf{h}(t+1)} q_{\mathbf{h}(t+1)}p_{\mathbf{h}(t+1)}\gamma_Q^{[\mathbf{h}(t+1)]}s^{[\mathbf{h}(t+1)]}\mathcal{P}_{\mathrm{PF}}^{[\mathbf{h}(t+1)]}\mathcal{U}_{\mathrm{dev}}^{[\mathbf{h}(t+1)]}$$

$$= \sum_{\mathbf{h}(t)}\sum_{\mathbf{x}} q_{\mathbf{h}(t)}p_{\mathbf{h}(t)}\gamma_Q^{[\mathbf{h}(t)]}s^{[\mathbf{h}(t)]}(\mathcal{S}^{f\,(\mathbf{x}\oplus\mathbf{y})}\otimes\langle\langle\Pi_{\mathbf{x}\oplus\mathbf{y}}|)(\sum_i\eta_i\mathcal{P}_i)\mathcal{N}\Lambda\mathcal{Q}_A\mathcal{P}_{\mathrm{PF}}^{[\mathbf{h}(t)]}\mathcal{U}_{\mathrm{dev}}^{[\mathbf{h}(t)]}$$

$$= \sum_{\mathbf{h}(t)}\sum_{\mathbf{x}} q_{\mathbf{h}(t)}p_{\mathbf{h}(t)}\gamma_Q^{[\mathbf{h}(t)]}s^{[\mathbf{h}(t)]}(\mathcal{S}^{f\,(\mathbf{x}\oplus\mathbf{y})}\otimes\langle\langle\Pi_{\mathbf{x}\oplus\mathbf{y}}|)\Lambda\mathcal{Q}_A\mathcal{P}_{\mathrm{PF}}^{[\mathbf{h}(t)]}\mathcal{U}_{\mathrm{dev}}^{[\mathbf{h}(t)]}$$

$$= \mathcal{T}(\sum_{\mathbf{x}}\langle\langle\Pi_{\mathbf{x}\oplus\mathbf{y}}|+_L\rangle\rangle)\sum_{\mathbf{h}(t)}p_{\mathbf{x}}q_{\mathbf{h}(t)}p_{\mathbf{h}(t)}\gamma_Q^{[\mathbf{h}(t)]}s^{[\mathbf{h}(t)]}\mathcal{P}_{\mathrm{PF}}^{[\mathbf{h}(t)]}\mathcal{U}_{\mathrm{dev}}^{[\mathbf{h}(t)]}$$

$$= \mathcal{T}\tilde{\mathcal{U}}^{(t)} = \tilde{\mathcal{U}}^{(t+1)}. \tag{E38}$$

Therefore, we have verified that all the logical operations obey Eq. (E21).

APPENDIX F: ON THE NOISE MODEL OF THE DECODING ERRORS

In the main text, we assume that the decoding errors for elementary logical operations can be modeled as Markovian and stochastic Pauli noise obeying Eq. (8). Here, we justify this assumption. First, we determine whether an actual noise model of syndrome-measurement cycles of surface codes can be treated as Markovian or not. When the syndrome measurements may output an incorrect value, we need d consecutive syndrome values for reliably estimating the recovery operations. Since the quantum states are in the logical code space only after the recovery operations, the actual quantum states are not in the code space during FTQC. This makes it hard to evaluate the logical noise map for several cycles during FTQC, because the map does not take a logical state to another logical state, while we need to evaluate the logical noise map in advance in order to perform QEM on the code space. To avoid this problem, we assume that the following noise model can well approximate the actual noise model: suppose that we can perform perfect syndrome measurements in the Ldth cycle, where $L = 1, 2, \ldots$, and that we can perform recovery operations just after that. Then, the quantum state is in the logical code

space at the Ldth cycle. In this case, we can define a logical error map $\mathcal{M}_{\mathrm{dec}}$ from the $(L-1)d$th cycle to the Ldth cycle. Here, we assume that if we have a logical operation $\mathcal{U}$ requiring χd cycles, the logical map including effective decoding errors can be approximated with $\mathcal{M}_{\mathrm{dec}}^{\chi}\mathcal{U}$ when the code distance d is sufficiently large. If this assumption holds, we can cancel the noise map $\mathcal{M}_{\mathrm{dec}}^{\chi}$ by performing QEM on each logical operation. Although the actual χ depends on the logical operations, we assume that $\chi = 1$ for simplicity.

The following numerical analysis shows that this assumption holds at least when the physical errors are stochastic Pauli noise. Let $\mathcal{M}_{\mathrm{dec},c}$ be a noise map for c-cycle idling of a single logical qubit with code distance d, and let $\Lambda^{(c)}$ be the Pauli transfer matrix of $\mathcal{M}_{\mathrm{dec},c}$. Here, $\mathcal{M}_{\mathrm{dec},c}$ is a stochastic noise map since a stochastic Pauli error can only cause logical Pauli errors; thus, $\Lambda^{(c)}$ is a diagonal matrix. Our assumption can be rephrased as follows: there is an effective Pauli transfer matrix Λ_{eff} such that $\Lambda^{(c)} = \Lambda_{\mathrm{eff}}^c$ for sufficiently large c. Equivalently, we assume that each diagonal element decays exponentially to the number of cycles c. Since Λ_{00} is always unity for stochastic Pauli errors, we are interested in the other diagonal elements. Figure 12 plots the diagonal elements, except Λ_{00}, according to the number of cycles. We utilize the same settings as in Sec. 2, i.e., a depolarizing noise map with $p = 0.01$. Note that Λ_{33} and Λ_{11} are equal, since the

behavior of surface codes is symmetric for Pauli-X and Z errors. In the figure, the circles are the numerical results, and the dashed lines are fitting results with an exponentially decaying function. We use data points with $c > 20$ for the fitting. The results agree well with our assumption, except for the region where the cycle count c is around 1. Thus, we can conclude that if a physical error is stochastic Pauli noise, we can define an effective logical noise map for each logical operation.

Next, we discuss the case in which the physical noise cannot be modeled as stochastic Pauli noise. In practice, the noise of calibrated quantum operations is expected to be almost stochastic Pauli, since the unitary component of the physical errors can be canceled by echoing. In addition, it is expected that the noise map per code cycle in the code space of surface codes is well approximated as stochastic Pauli noise at a large code distance [35]. While the logical noise map on a prepared magic state also suffers the noise from magic state injection and distillation, this noise can be twirled to stochastic Pauli noise with error-mitigated logical Clifford operations. Nevertheless, there may be non-negligible coherence in quantum noise of logical Clifford gates. In this case, we can twirl the noise map per code cycle and remove the unitary component of the noise via logical Pauli operations with hardware update (see Appendix E for the definition of hardware update). Below, we show that we can perform twirling on logical noise caused by logical Clifford operations if logical Pauli operations with physical operations, i.e., logical Pauli operations without updating Pauli frame, can be performed with negligible error rates. Note that the logical error rates for logical Pauli operations are expected to be sufficiently smaller than those for logical Clifford operations. This is because logical Pauli operations can be performed with transversal single-qubit operations that are completed in a single cycle, and the errors caused by these operations are negligible compared with those caused by two-qubit operations for stabilizer measurements. Suppose that we perform a logical Clifford operation $\mathcal{C}$ and a logical noise map $\mathcal{M}_{\text{dec}}^{\chi}$ follows it. Further suppose that we perform twirling noise $\mathcal{M}_{\text{dec}}^{\chi}$ with a set of Pauli operators $\mathbb{S}$. The twirling process can be described as follows:

$$\left(\frac{1}{|\mathbb{S}|} \sum_{\mathcal{P} \in \mathbb{S}} \mathcal{P} \mathcal{M}_{\text{dec}}^{\chi} \mathcal{P}\right) \mathcal{C} = \frac{1}{|\mathbb{S}|} \sum_{\mathcal{P} \in \mathbb{S}} \mathcal{P} \mathcal{M}_{\text{dec}}^{\chi} \mathcal{C}(\mathcal{C}^{\dagger} \mathcal{P} \mathcal{C}),$$
$$(\text{F1})$$

where $\mathcal{P}$ is the superoperator of the logical Pauli operations. Since $\mathcal{C}^{\dagger} \mathcal{P} \mathcal{C}$ is a logical Pauli operation, $\mathcal{M}_{\text{dec}}^{\chi}$ can be twirled simply with logical Pauli operations. The same arguments hold for Pauli measurements and feedback operations dependent on their outcomes. Since all the elemental logical operations except for magic state injection are Clifford operations or Pauli channels, we can apply

Pauli twirling to most of the quantum operations in FTQC. Note that while logical Pauli operations for computation can be done by updating the Pauli frame, the logical Pauli operations for twirling require a physical implementation on quantum devices. This is because when we attempt to perform logical Pauli operations via the Pauli frame for twirling, we need to update the Pauli frame according to the actual logical operation $\mathcal{M}_{\text{dec}}^{\chi} \mathcal{C}$ (see Appendix E for the formalism of the Pauli frame in superoperator representation). However, since $\mathcal{M}_{\text{dec}}^{\chi}$ is not a stochastic Pauli noise, we cannot keep the Pauli frame as a Pauli operator; thus, we cannot continue tracking the frame as a Pauli operator. Thus, the Pauli twirling must be done physically.

APPENDIX G: DETAILS OF THE NUMERICAL ANALYSIS

In the numerical simulations for evaluating the decoding errors, we use a uniform depolarizing noise model, in which noise occurs on each physical qubit independently and acts as follows:

$$\mathcal{E}(\rho) = (1 - p)\rho + \frac{p}{3}(X \rho X + Y \rho Y + Z \rho Z). \quad (\text{G1})$$

This error acts on data qubits at the beginning of each cycle and on ancillary qubits just before the measurement. As indicated in the main text, we assume perfect syndrome measurements at the 0th and dth cycles, which guarantees that the quantum states at these cycles are in logical space with recovery operations regardless of whether the decoding is successful or not. Then, we evaluate the logical error probabilities during these cycles. To estimate the recovery operation, we use a minimum-weight perfect-matching decoder [49]. This decoder reduces the decoding problem to an instance of the minimum-weight perfect-matching problem. While this problem is NP hard when there are Pauli-Y errors, we can approximately solve it by using Edmonds' blossom algorithm [50]. It is known that surface codes show threshold behavior even with this approximation. We use the implementation in Ref. [72] for solving this problem. To estimate the logical error rate, we evaluate 10^5 samples for each data point in Fig. 4(b) and 10^6 samples for the other figures.

In the performance evaluation of error mitigation for decoding errors, we assume that the error channel for each logical gate is a nonuniform logical depolarizing channel obtained in the benchmark of the logical error probabilities in surface codes. Since there is no perfect syndrome measurement in practice, this assumption does not hold exactly. Nevertheless, this approximation is asymptotically correct, and thus, we use it to evaluate the performance of QEM in the case of logical errors.

For the simulation of the Clifford circuits, we use a stabilizer circuit simulator of which the memory allocations are optimized so that the updates for the actions
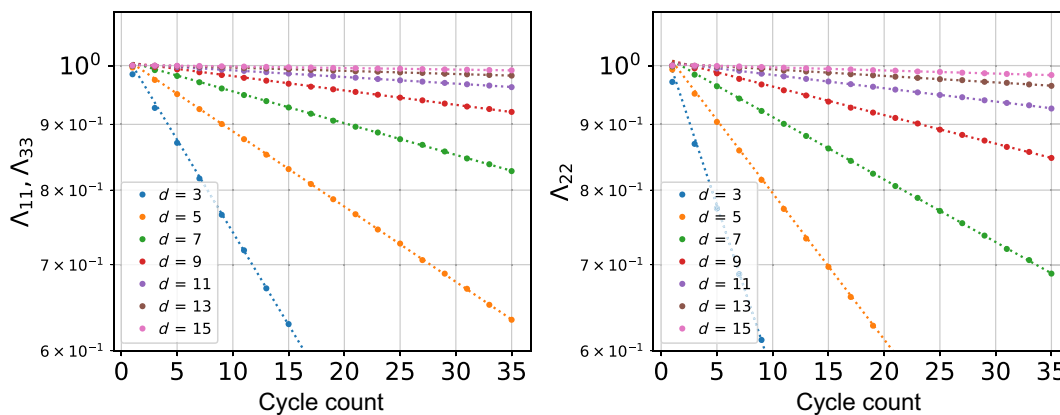
FIG. 12. Diagonal elements of Pauli transfer matrix for a noise map of a single logical qubit during c syndrome-measurement cycles plotted versus the number of code cycles. Each color corresponds to the code distance of a logical qubit. The circles are numerical results, and the dashed lines are fitting results with an exponentially decaying function.

of the Clifford operations become sequential. With this technique, the simulation of the stabilizer circuits that are dominated by Clifford operations rather than by Pauli measurements becomes hundreds of times faster than the existing stabilizer simulators [52].

For the Solovay-Kitaev algorithm, we use the method and implementation proposed in Ref. [38]. While we need to limit the allowed number of T gates, this method outputs a sequence of Clifford and T gates according to the allowed error rate ε. Thus, we search for the minimum error rate ε^* with which the algorithm outputs a sequence wherein the number of T gates is smaller than the allowed number of T gates. This search involves using a simple bisection method, and it is repeated until the accuracy reached 10^{-14}. For the simulation of the SWAP test circuits, we use Qulacs [73], which is a simulator for general noisy quantum circuits and is fast especially when a huge number of simulations have to be performed on small quantum circuits.

APPENDIX H: QUANTUM ERROR MITIGATION FOR DECISION PROBLEMS

Several important algorithms in FTQC, such as prime factoring and calculation of the ground-state energy, are a procedure to obtain the computational results via phase estimation sampling [9,30] and are not a procedure to calculate expectation values of observables. Since typical QEM techniques are designed to reduce the bias in the expectation value caused by noise, it is not clear whether QEM can be applied to such sampling algorithms. In this section, we show that QEM can be utilized for mitigating errors in a wider range of problems than calculating expectation values. In particular, we show that several promising long-term algorithms, such as ground-state energy estimation via phase estimation sampling and Shor's factoring algorithm, can be decomposed into a series of decision

problems, and show that QEM can be applied to each algorithm to solve the decision problem. To the best of our knowledge, this has not been mentioned in the context of QEM, while a similar concept has been known in the context of quasiprobability sampling for classically simulating quantum circuits [74,75].

First, we show that ground-energy estimation with phase estimation sampling [9,10] can be decomposed into a series of decision problems with a bisection method. In a quantum phase estimation routine, we prepare an initial state, and perform phase estimation based on quantum simulation. Then, we obtain a quantum state $|\psi\rangle = \sum_i \alpha_i |\tilde{E}_i\rangle |\psi_i\rangle$, where $|\tilde{E}_i\rangle$ is the energy of the ith eigenstate or its relevant value in binary representation, $|\psi_i\rangle$ is an eigenstate of the given Hamiltonian, and α_i corresponds to the overlap between the ith eigenstate and the initial state. The Pauli-Z basis measurement is performed on the first register of $|\psi\rangle$, and E_i is sampled with the probability $|\alpha_i|^2$.

Now, we apply a bisection method to estimate the ground-state energy. The subroutine outputs 1 if the sampled energy is smaller than a given parameter K and outputs 0 otherwise. Suppose that the prepared initial state has an overlap with the ground state larger than $1/\mathrm{poly}(n)$. If the ground energy is smaller than K, the subroutine outputs 1 with probability more than $1/\mathrm{poly}(n)$. Otherwise, the subroutine always outputs 0. We denote the process before the Pauli-Z measurements as $\mathcal{P}$, and the classical postprocessing after the measurements to determine 0 or 1 as f. Then, we can conclude that the procedure as the sampling of a bit-string x with the probability $\langle\langle \Pi_x | \mathcal{P} | 0 \rangle\rangle$, where Π_x is the POVM element on the first register, and outputs a single bit $f(x)$ with the classical postprocessing. When the quantum process $\mathcal{P}$ suffers from errors, we can apply QEM to $\mathcal{P}$ to mitigate errors. While there may exist a complicated classical postprocessing using a sampled bit string after the quantum process $\mathcal{P}$, the combination of sampling Π_x and classical postprocessing f can
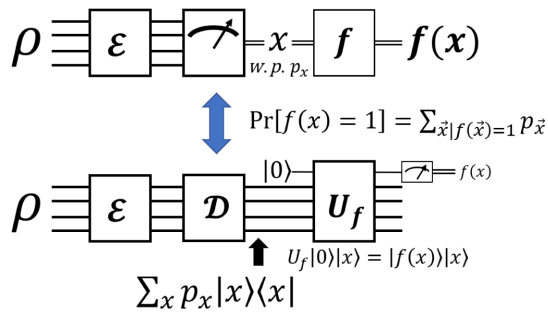
FIG. 13. The diagram of circuit conversion from the original decision problem to the fully quantum picture. Here, ρ is an n-qubit initial state, $\mathcal{E}$ is a noisy process, $\mathcal{D}$ is a perfect dephasing noise map, f is a classical binary function on an n-bit string x, and U_f is a quantum circuit that simulates f as $U|0\rangle|x\rangle = |f(x)\rangle|x\rangle$. See the main text for details.

be interpreted as a quantum process with strong dephasing noise. The evaluation of a decision bit $f(x)$ can also be interpreted as the evaluation of the Pauli-Z operator. Therefore, we can interpret the whole subroutine as a fully quantum process for calculating the expectation value, and we can apply QEM to the subroutine. The process of circuit conversion is shown in Fig. 13.

With this subroutine, we can solve the original ground-energy estimation with a bisection method. We assume the minimum and maximum possible energy $E_{\min}$ and $E_{\max}$ as variables, respectively. Then, the subroutine is called several times to check whether the ground energy is smaller than $E_{\mathrm{mid}} = (E_{\max} + E_{\min})/2$. If the ground energy is smaller than $(E_{\max} + E_{\min})/2$, the polynomial number of calls is enough to determine the inequality with high confidence. According to the output, we reduce a possible range of the ground energy $[E_{\min}, E_{\max}]$ by updating $E_{\max}$ or $E_{\min}$ with E_{mid}. Since the range is halved in each iteration, we need $O(\log \epsilon^{-1})$ iterations to achieve $E_{\max} - E_{\min} < \epsilon$.

When the number of logical errors in each iteration is $O(1)$ on average, the sampling overhead due to the application of QEM is also constant. While this conversion enables the application of QEM, this requires $O(\log \epsilon^{-1})$ times more iterations for binary search compared with the original algorithm. Nevertheless, since these iterations are independent computational tasks and the total number of logical operations and hardware requirements per single execution do not change. Therefore, we can conclude that QEM can be applied not only for the evaluation of the expectation values but also for long-term applications.

For a fair comparison, the following fact should also be noted. In the case of prime factoring, we can apply the same decomposition to the algorithm by finding the minimum nontrivial factor of a given integer. However, when the noise model is stochastic and the sampling overhead of QEM is constant, i.e., the mean number of logical errors during FTQC is $O(1)$, we can obtain a correct answer

for prime factoring with a small overhead without using QEM. Since prime factoring is in the NP class, we can efficiently check whether the submitted answer is correct or not. When the mean number of errors in FTQC is $O(1)$, i.e., the overhead of QEM is constant, a noiseless sampling occurs with a constant probability. This means we can obtain a correct nontrivial factor with constant sampling overheads even without QEM. Thus QEM is not effective for problems in the intersection of the BQP and NP class such as prime factoring. In other words, QEM is useful for problems satisfying the following two conditions: (1) problems are in the BQP class but not in the NP class, and (2) problems can be reduced to a series of decision problems. Useful algorithms for long-term applications, such as quantum simulation and the estimation of ground energy of spin models and molecules, satisfy these conditions.

[1] P. W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, SIAM Rev. **41**, 303 (1999).
[2] A. W. Harrow, A. Hassidim, and S. Lloyd, Quantum Algorithm for Linear Systems of Equations, Phys. Rev. Lett. **103**, 150502 (2009).
[3] M. A. Nielsen and I. Chuang, Quantum computation and quantum information (Cambridge University Press, Cambridge, England, 2010).
[4] D. A. Lidar and T. A. Brun, *Quantum Error Correction* (Cambridge University Press, New York, 2013).
[5] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, Surface codes: Towards practical large-scale quantum computation, Phys. Rev. A **86**, 032324 (2012).
[6] C. Horsman, A. G. Fowler, S. Devitt, and R. Van Meter, Surface code quantum computing by lattice surgery, New J. Phys. **14**, 123011 (2012).
[7] A. G. Fowler and C. Gidney, Low overhead quantum computation using lattice surgery, Preprint ArXiv:1808.06709 (2018).
[8] S. Lloyd, Universal quantum simulators, Science, **273**, 1073 (1996).
[9] I. D. Kivlichan, C. Gidney, D. W. Berry, N. Wiebe, J. McClean, W. Sun, Z. Jiang, N. Rubin, A. Fowler, A. Aspuru-Guzik, *et al.*, Improved fault-tolerant quantum simulation of condensed-phase correlated electrons via trotterization, Quantum **4**, 296 (2020).
[10] R. Babbush, C. Gidney, D. W. Berry, N. Wiebe, J. McClean, A. Paler, A. Fowler, and H. Neven, Encoding Electronic Spectra in Quantum Circuits with Linear t Complexity, Phys. Rev. X **8**, 041015 (2018).
[11] A. Erhard, H. P. Nautrup, M. Meth, L. Postler, R. Stricker, M. Stadler, V. Negnevitsky, M. Ringbauer, P. Schindler, H. J. Briegel, *et al.*, Entangling logical qubits with lattice surgery, Nature **589**, 220 (2021).
[12] G. Q. Ai, Exponential suppression of bit or phase errors with cyclic error correction, Nature **595**, 383 (2021).

[13] 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).

[14] For drawing the blue line, we assume that the ratio between physical error rates and the threshold is 0.1 and the quantum algorithms are run with $O(n^3)$ time complexity for the problem size n with the number of physical qubits being fixed.

[15] A. Holmes, M. R. Jokar, G. Pasandi, Y. Ding, M. Pedram, and F. T. Chong, Nisq+: Boosting quantum computing power by approximating quantum error correction, Preprint ArXiv:2004.04794 (2020).

[16] Y. Ueno, M. Kondo, M. Tanaka, Y. Suzuki, and Y. Tabuchi, Qecool: On-line quantum error correction with a superconducting decoder for surface code, Preprint ArXiv:2103.07526 (2021).

[17] P. Das, A. Locharla, and C. Jones, Lilliput: A lightweight low-latency lookup-table based decoder for near-term quantum error correction, Preprint ArXiv:2108.06569 (2021).

[18] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O'brien, A variational eigenvalue solver on a photonic quantum processor, Nat. Commun. **5**, 1 (2014).

[19] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow, and J. M. Gambetta, Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets, Nature **549**, 242 (2017).

[20] J. R. McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik, The theory of variational hybrid quantum-classical algorithms, New J. Phys. **18**, 023023 (2016).

[21] K. Temme, S. Bravyi, and J. M. Gambetta, Error Mitigation for Short-Depth Quantum Circuits, Phys. Rev. Lett. **119**, 180509 (2017).

[22] S. Endo, S. C. Benjamin, and Y. Li, Practical Quantum Error Mitigation for Near-Future Applications, Phys. Rev. X **8**, 031027 (2018).

[23] R. Takagi, S. Endo, S. Minagawa, and M. Gu, Fundamental limitations of quantum error mitigation, Preprint ArXiv:2109.04457 (2021).

[24] S. Wang, P. Czarnik, A. Arrasmith, M. Cerezo, L. Cincio, and P. J. Coles, Can error mitigation improve trainability of noisy variational quantum algorithms?, Preprint ArXiv:2109.01051 (2021).

[25] A. Y. Kitaev, Quantum computations: Algorithms and error correction, Russ. Math. Surv. **52**, 1191 (1997).

[26] C. M. Dawson and M. A. Nielsen, The Solovay-Kitaev algorithm, Preprint ArXiv:quant-ph/0505030 (2005).

[27] R. Blume-Kohout, J. K. Gamble, E. Nielsen, J. Mizrahi, J. D. Sterk, and P. Maunz, Robust, self-consistent, closed-form tomography of quantum logic gates on a trapped ion qubit, Preprint ArXiv:1310.4492 (2013).

[28] D. Greenbaum, Introduction to quantum gate set tomography, Preprint ArXiv:1509.02921 (2015).

[29] A. Y. Kitaev, Quantum measurements and the Abelian stabilizer problem, Preprint ArXiv:quant-ph/9511026 (1995).

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

[31] Y. Li and S. C. Benjamin, Efficient Variational Quantum Simulator Incorporating Active Error Minimization, Phys. Rev. X **7**, 021050 (2017).

[32] D. Gottesman, Stabilizer codes and quantum error correction, Preprint ArXiv:quant-ph/9705052 (1997).

[33] B. Eastin and E. Knill, Restrictions on Transversal Encoded Quantum Gate Sets, Phys. Rev. Lett. **102**, 110502 (2009).

[34] L. Riesebos, X. Fu, S. Varsamopoulos, C. G. Almudever, and K. Bertels, in *Proceedings of the 54th Annual Design Automation Conference 2017* (ACM, 2017), p. 1.

[35] S. Bravyi, M. Englbrecht, R. König, and N. Peard, Correcting coherent errors with surface codes, npj Quantum Inf. **4**, 1 (2018).

[36] N. C. Jones, R. Van Meter, A. G. Fowler, P. L. McMahon, J. Kim, T. D. Ladd, and Y. Yamamoto, Layered Architecture for Quantum Computing, Phys. Rev. X **2**, 031007 (2012).

[37] A. G. Fowler, A. C. Whiteside, and L. C. Hollenberg, Towards Practical Classical Processing for the Surface Code, Phys. Rev. Lett. **108**, 180501 (2012).

[38] N. J. Ross and P. Selinger, Optimal ancilla-free Clifford+T approximation of z-rotations, Preprint ArXiv:1403.2975 (2014).

[39] H. Bombin and M. A. Martin-Delgado, Topological Quantum Distillation, Phys. Rev. Lett. **97**, 180501 (2006).

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

[41] A. Steane, Multiple-particle interference and quantum error correction, Proc. R. Soc. London, Ser. A **452**, 2551 (1996).

[42] E. Campbell, Random Compiler for Fast Hamiltonian Simulation, Phys. Rev. Lett. **123**, 070503 (2019).

[43] E. Nielsen, J. K. Gamble, K. Rudinger, T. Scholten, K. Young, and R. Blume-Kohout, Gate set tomography, Preprint ArXiv:2009.07301 (2020).

[44] S. Endo, Z. Cai, S. C. Benjamin, and X. Yuan, Hybrid quantum-classical algorithms and quantum error mitigation, Preprint ArXiv:2011.01382 (2020).

[45] S. Bravyi and D. Gosset, Improved Classical Simulation of Quantum Circuits Dominated by Clifford Gates, Phys. Rev. Lett. **116**, 250501 (2016).

[46] C. Piveteau, D. Sutter, S. Bravyi, J. M. Gambetta, and K. Temme, Error mitigation for universal gates on encoded qubits, Preprint ArXiv:2103.04915 (2021).

[47] C. Wang, J. Harrington, and J. Preskill, Confinement-Higgs transition in a disordered gauge theory and the accuracy threshold for quantum memory, Ann. Phys. (N. Y.) **303**, 31 (2003).

[48] N. Delfosse and N. H. Nickerson, Almost-linear time decoding algorithm for topological codes, Preprint ArXiv:1709.06218 (2017).

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

[50] J. Edmonds, Paths, trees, and flowers, Can. J. Math. **17**, 449 (1965).

[51] A. Strikis, D. Qin, Y. Chen, S. C. Benjamin, and Y. Li, Learning-based quantum error mitigation, Preprint ArXiv:2005.07601 (2020).

[52] S. Aaronson and D. Gottesman, Improved simulation of stabilizer circuits, Phys. Rev. A **70**, 052328 (2004).

[53] A. K. Ekert, C. M. Alves, D. K. Oi, M. Horodecki, P. Horodecki, and L. C. Kwek, Direct Estimations of Linear and Nonlinear Functionals of a Quantum State, Phys. Rev. Lett. **88**, 217901 (2002).

[54] E. Pednault, J. A. Gunnels, G. Nannicini, L. Horesh, and R. Wisnieff, Leveraging secondary storage to simulate deep 54-qubit sycamore circuits, Preprint ArXiv:1910.09534 (2019).

[55] M. Suzuki, General theory of fractal path integrals with applications to many-body theories and statistical physics, J. Math. Phys. **32**, 400 (1991).

[56] G. H. Low and I. L. Chuang, Hamiltonian simulation by qubitization, Quantum **3**, 163 (2019).

[57] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, Simulating Hamiltonian Dynamics with a Truncated Taylor Series, Phys. Rev. Lett. **114**, 090502 (2015).

[58] G. H. Low and I. L. Chuang, Optimal Hamiltonian Simulation by Quantum Signal Processing, Phys. Rev. Lett. **118**, 010501 (2017).

[59] S. Endo, Q. Zhao, Y. Li, S. Benjamin, and X. Yuan, Mitigating algorithmic errors in a hamiltonian simulation, Phys. Rev. A **99**, 012334 (2019).

[60] A. C. Vazquez, R. Hiptmair, and S. Woerner, Enhancing the quantum linear systems algorithm using Richardson extrapolation, Preprint ArXiv:2009.04484 (2020).

[61] K. Mitarai and K. Fujii, Constructing a virtual two-qubit gate from single-qubit operations, Preprint ArXiv:1909.07534 (2019).

[62] K. Mitarai and K. Fujii, Overhead of the non-local-to-local channel decomposition by quasiprobability sampling, Preprint ArXiv:2006.11174 (2020).

[63] J. R. McClean, Z. Jiang, N. C. Rubin, R. Babbush, and H. Neven, Decoding quantum errors with subspace expansions, Nat. Commun. **11**, 1 (2020).

[64] J. R. McClean, M. E. Kimchi-Schwartz, J. Carter, and W. A. De Jong, Hybrid quantum-classical hierarchy for mitigation of decoherence and determination of excited states, Phys. Rev. A **95**, 042308 (2017).

[65] Y. Xiong, D. Chandra, S. X. Ng, and L. Hanzo, Sampling overhead analysis of quantum error mitigation: Uncoded vs. coded systems, IEEE Access **8**, 228967 (2020).

[66] M. Lostaglio and A. Ciani, Error mitigation and quantum-assisted simulation in the error corrected regime, Preprint ArXiv:2103.07526 (2021).

[67] S. B. Bravyi and A. Y. Kitaev, Quantum codes on a lattice with boundary, Preprint ArXiv:quant-ph/9811052 (1998).

[68] B. J. Brown, K. Laubscher, M. S. Kesselring, and J. R. Wootton, Poking Holes and Cutting Corners to Achieve Clifford Gates with the Surface Code, Phys. Rev. X **7**, 021029 (2017).

[69] C. Gidney and A. G. Fowler, Efficient magic state factories with a catalyzed $|CCZ\rangle$ to $2|T\rangle$ transformation, Quantum **3**, 135 (2019).

[70] Y. Li, A magic state's fidelity can be superior to the operations that created it, New J. Phys. **17**, 023037 (2015).

[71] C. J. Trout and K. R. Brown, Magic state distillation and gate compilation in quantum algorithms for quantum chemistry, Int. J. Quantum Chem. **115**, 1296 (2015).

[72] V. Kolmogorov, Blossom V: A new implementation of a minimum cost perfect matching algorithm, Math. Prog. Comput. **1**, 43 (2009).

[73] Y. Suzuki, Y. Kawase, Y. Masumura, Y. Hiraga, M. Nakadai, J. Chen, K. M. Nakanishi, K. Mitarai, R. Imai, S. Tamiya, *et al.*, Qulacs: A fast and versatile quantum circuit simulator for research purpose, Quantum **5**, 559 (2021).

[74] R. S. Bennink, E. M. Ferragut, T. S. Humble, J. A. Laska, J. J. Nutaro, M. G. Pleszkoch, and R. C. Pooser, Unbiased simulation of near-clifford quantum circuits, Phys. Rev. A **95**, 062337 (2017).

[75] S. Hakkaku, K. Mitarai, and K. Fujii, A sampling-based quasi-probability simulation for fault-tolerant quantum error correction on the surface codes under coherent noise, Preprint ArXiv:2105.04478 (2021).