

Factory-based fault-tolerant preparation of quantum polar codes encoding one logical qubitAshutosh Goswami^{1,*}, Mehdi Mhalla^{2,†} and Valentin Savin^{3,‡}¹*QMATH, Department of Mathematical Sciences, University of Copenhagen, Universitetsparken 5, 2100 Copenhagen, Denmark*²*University of Grenoble Alpes, CNRS, Grenoble INP, LIG, F-38000 Grenoble, France*³*University of Grenoble Alpes, CEA-Leti, F-38054 Grenoble, France*

(Received 21 January 2024; accepted 11 June 2024; published 12 July 2024)

A fault-tolerant way to prepare logical code states of \mathcal{Q}_1 codes, i.e., quantum polar codes encoding one qubit, was recently proposed. The fault tolerance therein is guaranteed by an error detection gadget, where, if an error is detected during the preparation, one entirely discards the preparation. Due to error detection, the preparation is probabilistic and its success rate, referred to as the *preparation rate*, decreases rapidly with the code length, preventing the preparation of code states of large code-lengths. In this paper, to improve the preparation rate, we consider a factory preparation of \mathcal{Q}_1 code states, where one attempts to prepare several copies of \mathcal{Q}_1 code states in parallel. Using an extra scheduling step, we can avoid discarding the preparation entirely every time an error is detected, hence, achieving an increased preparation rate in turn. We further provide a theoretical method to estimate preparation and logical error rates of \mathcal{Q}_1 codes, prepared using factory preparation, which is shown to tightly fit the Monte Carlo simulation-based numerical results. Therefore, our theoretical method is useful for providing estimates for large code lengths, where Monte Carlo simulations are practically not feasible. Our numerical results, for a circuit-level depolarizing noise model, indicate that the preparation rate increases significantly, especially for large code-length N . For example, for $N = 256$, it increases from 0.02% to 27% for a practically interesting physical error rate $p = 10^{-3}$. Remarkably, a \mathcal{Q}_1 code with $N = 256$ achieves logical error rates around 10^{-11} and 10^{-15} for $p = 10^{-3}$ and $p = 3 \times 10^{-4}$, respectively. This corresponds to an improvement of about three orders of magnitude compared to a surface code with similar code length and minimum distance, thus showing the promise of the proposed scheme for large-scale fault-tolerant quantum computing.

DOI: [10.1103/PhysRevA.110.012438](https://doi.org/10.1103/PhysRevA.110.012438)**I. INTRODUCTION**

Polar codes are known for their excellent error correction performance in both classical and quantum communication settings [1–5]. In particular, they achieve the symmetric capacity of any binary-input discrete memoryless classical channel, and achieve the symmetric coherent information of any quantum channel. They also come equipped with a fast classical decoding algorithm (log-linear complexity), which can readily be adapted for Pauli channels, using a syndrome-based decoding approach [2,3].

Despite their excellent error correction performance and nice algebraic and structural properties, polar codes remain largely unexplored for fault-tolerant quantum computation (FTQC). The main reason is due to the high weight of their stabilizer group generators, which prevents fault-tolerant state preparation and error correction from being implemented by repeated syndrome measurements. This contrasts with conventional approaches based on topological, or more generally, quantum low-density parity-check (LDPC) codes, allowing the implementation of a fault-tolerant quantum memory through repeated syndrome measurements, with errors being

detected by the difference between syndromes measured in consecutive rounds [6–8].

Recently, a fault-tolerant procedure to prepare logical states of \mathcal{Q}_1 codes, that is, CSS quantum polar codes encoding one logical qubit, was proposed in [9]. Combined with Steane error correction [10,11], this preparation procedure provides an alternative and promising approach for building a fault-tolerant quantum memory.

The preparation in [9] is measurement-based, where a set of N qubits are first initialized in the Pauli Z basis and then two qubit Pauli measurements are recursively applied on them. To achieve fault tolerance, the preparation is aided by an error detection gadget, which detects errors at each level of recursion. For error-detection-aided preparation, it has been explicitly proven that the preparation is fault-tolerant in the sense that the weight of the error in the prepared state does not exceed the number of components that fail during the procedure. The fault tolerance therein has been further confirmed by numerical simulations, revealing practically interesting pseudothreshold values for small \mathcal{Q}_1 codes of length $N \leq 64$ qubits, and showing the promise of the proposed approach to fault-tolerant error correction.

However, the preparation of \mathcal{Q}_1 code-states in [9] is probabilistic due to the error detection gadget. If the gadget detects an error at some recursion level, one declares a preparation failure and discards the prepared state. Hence, one may need to restart the preparation from the beginning several times

*Contact author: akg@math.ku.dk†Contact author: mehdi.mhalla@univ-grenoble-alpes.fr‡Contact author: valentin.savin@cea.fr

before a \mathcal{Q}_1 code is successfully prepared. The preparation rate, i.e., the rate of successful preparation, decreases rapidly with the code length and approaches zero as the code length increases, hence preventing the preparation of large \mathcal{Q}_1 code states.

In this paper, to improve the rate of successful preparation, we consider a factory preparation of \mathcal{Q}_1 code states, where one attempts to prepare several copies of polar code states in parallel, using the measurement-based preparation with error detection from [9]. Taking advantage of the recursive nature of the preparation, we introduce an extra scheduling step at some recursion levels, so that even if errors are detected we may proceed to the next level of recursion. In other words, we may not need to restart the entire procedure from the beginning every time an error is detected. Therefore, the factory preparation may provide better preparation rates compared to the preparation in [9].

In addition, we conduct a thorough theoretical analysis of the proposed factory-based preparation approach, constituting one of the most significant contributions of the paper. To this end, we define the notions of rough and smooth errors depending on whether or not the error flips one of the measurement outcomes in the measurement-based preparation. Using rough and smooth errors, we provide theoretical estimates of the preparation rate, as well as the probability of X and Z errors on the prepared state. The latter are used to estimate the logical error rates of the \mathcal{Q}_1 codes under Steane error correction, by using density evolution, as in [[9], Appendix E.4]. Our theoretical estimates are further substantiated by Monte Carlo simulations. For the circuit level depolarizing noise model, we observe that our theoretical estimate of the preparation rate fits well the Monte Carlo simulation for code lengths $N = 64$ and $N = 256$. Further, for the code length $N = 64$, our theoretical estimate of the logical error rate matches the logical error rate obtained based on Monte Carlo simulation. Therefore, we use our theoretical estimates to obtain logical error rates corresponding to small physical error rates and large code lengths, where Monte Carlo simulation is not practically feasible.

For the circuit level depolarizing noise model, our numerical results show that the factory preparation significantly improves the preparation rate of \mathcal{Q}_1 code states compared to [9]. In particular, for the physical error rate $p = 10^{-3}$, the preparation rate increases from 47% to 70% for a \mathcal{Q}_1 code state of length $N = 64$ and from 0.02% to 27% for a \mathcal{Q}_1 code of length $N = 256$. The improvement for $N = 256$ is quite significant as the preparation rate of 27% is practically feasible and implies a qubit overhead only by a factor of 4. We further included numerical results on the logic error rates of \mathcal{Q}_1 codes, using Steane error correction that incorporates our factory preparation of \mathcal{Q}_1 code states. The \mathcal{Q}_1 -code of length $N = 256$ achieves a logical error rate of 10^{-11} and 10^{-15} for physical error rate of 10^{-3} and 3×10^{-4} , respectively. A comparison with a surface code of similar length and minimum distance is also provided, further reinforcing the promise of polar codes for fault-tolerant quantum computation.

The paper is organized as follows. In Sec. II, we review \mathcal{Q}_1 codes and the measurement-based preparation of \mathcal{Q}_1 code states with error detection from [9]. In Sec. III, we describe our factory preparation, in Sec. IV, we provide theoretical estimates of the preparation rate and of the Pauli error

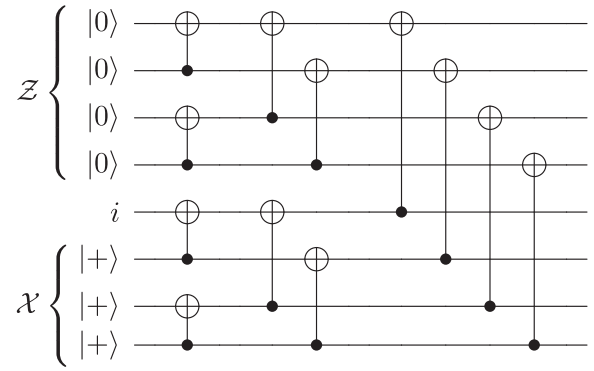


FIG. 1. An example of the encoding of \mathcal{Q}_1 codes: The figure shows the encoding of the code $\mathcal{Q}_1(N = 2^3, i = 5)$, with frozen states $|\mathbf{u}\rangle_{\mathcal{Z}} = |0, 0, 0, 0\rangle$ and $|\bar{\mathbf{v}}\rangle_{\mathcal{X}} = |+, +, +\rangle$.

probabilities on the prepared state, and in Sec. V we present our numerical results regarding the factory preparation and comparison with surface codes. Finally, in Sec. VI, we conclude with some perspectives and future directions.

II. PRELIMINARIES

A. \mathcal{Q}_1 codes

Here, we briefly review \mathcal{Q}_1 codes, which are CSS quantum polar codes that encode one logical qubit (for a review of CSS quantum polar codes see [[9], Sec. II]).

The quantum polar transform Q_N , where $N = 2^n$, with $n \geq 0$, is the unitary operation on N qubits that operates in the computational basis as the classical polar transform P_N . Precisely, for any $\mathbf{u} = (u_1, \dots, u_N) \in \{0, 1\}^N$, we define $Q_N|\mathbf{u}\rangle = |P_N\mathbf{u}\rangle$, where $P_N = \begin{pmatrix} 1 & \\ & 1 \end{pmatrix}^{\otimes n}$. Hence, Q_N can be realized by recursively applying the quantum CNOT gate, transversely, on subblocks of 2^k qubits, for $k = 0, \dots, n - 1$ (see Fig. 1).

Let $\mathcal{S} = \{1, \dots, N\}$ denote an N -qubit quantum system. For a \mathcal{Q}_1 code, a position $i \in \mathcal{S}$ is chosen to encode the logical information. Given the index i , the set of indices preceding i , i.e., $\mathcal{Z} := \{1, \dots, i - 1\}$, are frozen in a Z basis state $|\mathbf{u}\rangle_{\mathcal{Z}}$, $\mathbf{u} \in \{0, 1\}^{i-1}$. Further, the set of indices succeeding i , i.e., $\mathcal{X} := \{i + 1, \dots, N\}$, are frozen in an X basis state, $|\bar{\mathbf{v}}\rangle_{\mathcal{X}}$, where $\mathbf{v} \in \{0, 1\}^{N-i}$ and we use the notation $|\bar{0}\rangle := |+\rangle$, and $|\bar{1}\rangle := |-\rangle$.

Therefore, the logical code state, denoted by $|\tilde{\phi}\rangle_{\mathcal{S}}$, is given by $|\tilde{\phi}\rangle_{\mathcal{S}} = Q_N(|\mathbf{u}\rangle_{\mathcal{Z}} \otimes |\phi\rangle_i \otimes |\bar{\mathbf{v}}\rangle_{\mathcal{X}})$. In the following, we shall denote by $\mathcal{Q}_1(N, i)$ the \mathcal{Q}_1 code of length N , with information position $i \in \mathcal{S} = \{1, \dots, N\}$.

It is worth emphasizing that the error correction performance of a \mathcal{Q}_1 code greatly depends on the choice of the information position i . For depolarizing channels, the information position providing the best error correction performance, depending on the code length N , was determined by using the density evolution in [9].

Shor- \mathcal{Q}_1 codes. When the information position $i \in \mathcal{S}$ is a power of 2, i.e., $i = 2^k$, $0 \leq k \leq n$, the corresponding \mathcal{Q}_1 code is a Shor code [[9], Theorem 1]. The subfamily of Shor codes obtained from \mathcal{Q}_1 codes are referred to as Shor- \mathcal{Q}_1 codes. A Shor- \mathcal{Q}_1 code in general has inferior error correction

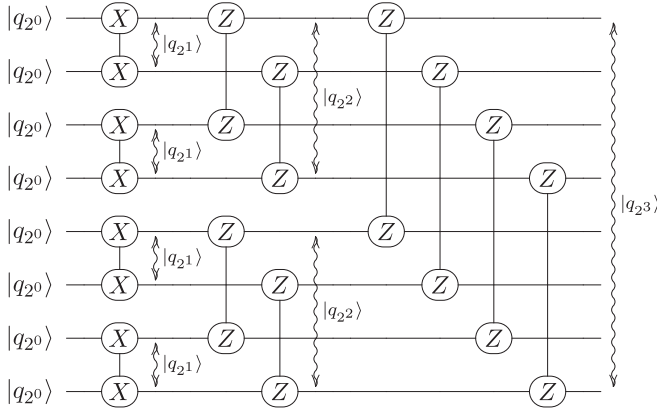


FIG. 2. An example of Procedure 1: The figure shows the measurement-based preparation of $|q_N\rangle_{\mathcal{S}}$ in (1), with $N=2^3$, $i(n)=7$, that is, $|0\rangle_L$ corresponding to $\mathcal{Q}_1(N=2^3, i=7)$. Here, slightly flattened circles connected by a vertical wire denote either an $X \otimes X$ or a $Z \otimes Z$ measurement on the corresponding qubits, and $|q_{2^k}\rangle$ are equivalent \mathcal{Q}_1 states of length 2^k ($|q_{2^0}\rangle$ is a Pauli Z basis state).

performance compared to a \mathcal{Q}_1 code of the same code length and minimum distance as the successive cancellation (SC) decoding of \mathcal{Q}_1 codes is able to decode beyond the minimum distance [9].

B. Measurement-based preparation of \mathcal{Q}_1 code states

In this section, we summarize the measurement-based preparation from [9] and discuss briefly its fault tolerance, under a circuit-level Pauli noise model.

Consider a $\mathcal{Q}_1(N, i)$ code, with $N=2^n$ and $i \in \mathcal{S} = \{1, \dots, N\}$. We consider logical Z and X states of the code $\mathcal{Q}_1(N, i)$, hence the information position is also frozen in either the Z or X basis, accordingly. Therefore, a $\mathcal{Q}_1(N, i)$ code state has the following form:

$$|q_N\rangle_{\mathcal{S}} := \mathcal{Q}_N(|\mathbf{u}, \bar{\mathbf{v}}\rangle_{\mathcal{S}}) = \mathcal{Q}_N(|\mathbf{u}\rangle_{\mathcal{Z}(n)} \otimes |\bar{\mathbf{v}}\rangle_{\mathcal{X}(n)}), \quad (1)$$

where $\mathcal{Z}(n) = \{1, \dots, i(n)\}$ and $\mathcal{X}(n) = \{i(n)+1, \dots, N\}$, where

$$i(n) = \begin{cases} i, & \text{for } Z \text{ logical state,} \\ i-1, & \text{for } X \text{ logical state.} \end{cases}$$

Therefore, $i(n)$ simply represents the length of Z type frozen set after the n th level of recursion. When no confusion is possible, we may simply write $|q_N\rangle$ instead of $|q_N\rangle_{\mathcal{S}}$. Finally, \mathcal{Q}_1 states defined by the same value of $i(n)$ are considered equivalent, regardless of the corresponding frozen values \mathbf{u}, \mathbf{v} . Note that the equivalent \mathcal{Q}_1 states are defined by the same stabilizer generators, up to sign factors [[9], Lemma 2].

1. Recursive measurement-based preparation without noise

Any \mathcal{Q}_1 code state can be prepared using the following measurement-based procedure [[9], Theorem 2] (see also Fig. 2).

Procedure 1 (Measurement-based Preparation [11]).

Consider $|q_N\rangle_{\mathcal{S}}$ from (1) and let $b_1 \dots b_n$ be the binary representation of $i(n)-1$, with b_n being the most

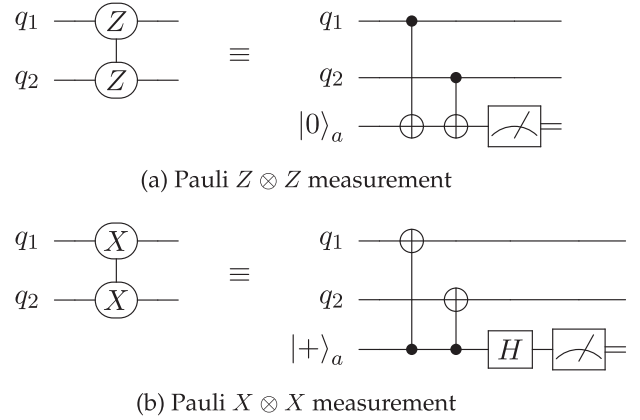


FIG. 3. Two-qubit Pauli measurements: Panels (a),(b) provide quantum circuits implementing Pauli $Z \otimes Z$ and $X \otimes X$ measurements, respectively.

significant bit, i.e., $i(n)-1 = \sum_{k=1}^n b_k 2^{k-1}$. Then, the measurement-based procedure to prepare $|q_N\rangle_{\mathcal{S}}$ is carried out in $n+1$ steps, as follows.

- (0) First, $\mathcal{S} = \{1, \dots, N\}$ is initialized in a Pauli Z basis state $|\mathbf{u}\rangle_{\mathcal{S}}$, for some $\mathbf{u} \in \{0, 1\}^N$.
- (1 \rightarrow n) Then, two-qubit Pauli measurements are recursively applied for n levels. The recursion is the same as the recursion of the quantum polar transform (see Fig. 1), except that each CNOT gate is replaced by either Pauli $X \otimes X$ or $Z \otimes Z$ measurement. Precisely, if $b_k = 0$ (or, $b_k = 1$), we apply Pauli $X \otimes X$ (or, $Z \otimes Z$) measurements at the k th recursion level, $k = 1, \dots, n$.

Note that a Pauli Z basis state can be considered as a \mathcal{Q}_1 code state of length 2^0 . Therefore, the first step of Procedure 1, i.e., the initialization in a Pauli Z basis, corresponds to the zeroth level of recursion, where one prepares 2^n copies of $|q_{2^0}\rangle$. After any k th level of recursion, $1 \leq k \leq n$, Procedure 1 prepares 2^{n-k} equivalent code states $|q_{2^k}\rangle$, with $i(k)-1 = \sum_{k=1}^k b_k 2^{k-1}$ (see Fig. 2). In particular, each $|q_{2^k}\rangle$ is produced by applying transversal Pauli $X \otimes X$ or $Z \otimes Z$ measurements on two equivalent $|q_{2^{k-1}}\rangle$ from the $(k-1)$ th level of recursion [[9], Lemma 1].

2. Recursive measurement-based preparation with noise

We consider the standard implementation of Pauli $Z \otimes Z$ and $X \otimes X$ measurements, using an ancilla qubit, as depicted in Fig. 3. Then, the measurement-based procedure consists of the following basic components: qubits initialization in either the Pauli X or Z basis, CNOT gates, measurements in the Pauli X or Pauli Z basis. It is easy to see that the total number of components in the preparation of a \mathcal{Q}_1 code state of length N , denoted here by C_N , is given by [9]

$$C_N = N(1 + 2 \log_2 N). \quad (2)$$

Noise model. We further assume that each component fails independently with some probability p , referred to as the physical error rate, according to a circuit level depolarizing noise model as follows [12].

(1) A noisy initialization in Pauli Z (or X) basis corresponds to the perfect initialization, followed by an X (or Z) error on the initialized qubit, with probability p .

(2) A noisy CNOT gate corresponds to applying the perfect CNOT gate, followed by a two-qubit depolarizing channel, with probability p . Precisely, after the perfect CNOT, any one of the following 15-two qubit Pauli errors $I \otimes X, I \otimes Y, I \otimes Z, X \otimes I, X \otimes X, X \otimes Y, X \otimes Z, Z \otimes I, Z \otimes X, Z \otimes Y, Z \otimes Z, Y \otimes I, Y \otimes X, Y \otimes Y, Y \otimes Z$ may happen, with total probability p (probability $p/15$ each).

(3) A noisy Pauli Z (or X) basis measurement corresponds to first applying a Pauli X (or Z) error on the qubit to be measured with probability p , then doing the perfect Pauli Z (or X) measurement.

The measurement-based preparation given in Procedure 1 is not fault-tolerant by itself under the above noise model. The reason is that the outcomes of transversal Pauli $Z \otimes Z$ or Pauli $X \otimes X$ measurements, which are needed to determine the values corresponding to the frozen \mathcal{Z} and \mathcal{X} sets, get error corrupted, thus leading to a wrong determination of frozen values [9].

To achieve fault tolerance, an error detection gadget is incorporated into the measurement-based procedure. Taking advantage of the redundancy in the transversal Pauli measurements, the error detection gadget detects errors at each level of recursion.

Error detection gadget. Here, we briefly present the error detection gadget for the case of Pauli $Z \otimes Z$ measurements (see [[9], Procedure 2] for more details). For $K = 2^k$, consider two equivalent \mathcal{Q}_1 code states $|q_{K/2}^1\rangle_{\mathcal{S}_1} = \mathcal{Q}_{K/2}|\mathbf{u}_1, \overline{\mathbf{v}_1}\rangle$ and $|q_{K/2}^1\rangle_{\mathcal{S}_2} = \mathcal{Q}_{K/2}|\mathbf{u}_2, \overline{\mathbf{v}_2}\rangle$, where $\mathbf{u}_1, \mathbf{u}_2 \in \{0, 1\}^{i(k-1)}$, and $\mathbf{v}_1, \mathbf{v}_2 \in \{0, 1\}^{K-i(k-1)}$. Consider first the noiseless scenario. The result of the transversal Pauli- $Z \otimes Z$ measurements on these polar code states is a codeword of a classical polar code as follows:

$$\mathbf{m} = P_{\frac{K}{2}}(\mathbf{u}', \mathbf{x}) \in \{0, 1\}^{\frac{K}{2}}, \quad (3)$$

where $\mathbf{u}' = \mathbf{u}_1 \oplus \mathbf{u}_2 \in \{0, 1\}^{i(k-1)}$ and $\mathbf{x} \in \{0, 1\}^{\frac{K}{2}-i(k-1)}$ is a random unknown vector and $P_{\frac{K}{2}}$ is the classical polar transform. After measurements, the state of the joint system $\mathcal{S} = \mathcal{S}_1 \cup \mathcal{S}_2$ is a \mathcal{Q}_1 state

$$|q_K\rangle_{\mathcal{S}} = \mathcal{Q}_K|\mathbf{u}', \mathbf{x}, \mathbf{u}_2, \overline{\mathbf{v}_1 \oplus \mathbf{v}_2}\rangle_{\mathcal{S}}, \quad (4)$$

with $i(k) = i(k-1) + K/2 > K/2$, and where \mathbf{x} is determined from the measurement outcome \mathbf{m} in (3) by, $\mathbf{x} = P_{\frac{K}{2}}(\mathbf{e}_X)|_{\mathcal{X}(k-1)}$, i.e., the subvector of $P_{\frac{K}{2}}(\mathbf{m}) \in \{0, 1\}^{K/2}$ corresponding to indices in the set $\mathcal{X}(k-1)$.

For the noisy scenario, the measurement outcome gives a noisy codeword of the classical polar code instead of (3), as follows:

$$\mathbf{m} = P_{\frac{K}{2}}(\mathbf{u}', \mathbf{x}) \oplus \mathbf{e}_X. \quad (5)$$

The error detection gadget determines the syndrome of the error term \mathbf{e}_X in the measurement outcome \mathbf{m} as, $P_{\frac{K}{2}}(\mathbf{e}_X)|_{\mathcal{Z}(k-1)} = P_{\frac{K}{2}}(\mathbf{m})|_{\mathcal{Z}(k-1)} \oplus \mathbf{u}'$. If the syndrome is zero, we proceed as in the noiseless case. If the syndrome is not zero, we report a component failure and discard the prepared state. For example, in Fig. 2, if an error is detected in one of

the $|q_2^2\rangle$ prepared at the second level of recursion, we discard the other prepared state even if no error is detected in it and restart the procedure from the beginning.

Therefore, one needs to repeat the preparation until a preparation succeeds without an error detection. The preparation rate is defined as

$$p_{\text{prep}} = \lim_{R \rightarrow \infty} \frac{t}{R}, \quad (6)$$

where t is the number of successful preparations out of R independent preparation attempts.

Finally, it is worth noting that some errors may not be detected by the gadget and hence will remain on the successfully prepared states. It has been shown that the successfully prepared state is fault-tolerant in the sense that the errors in the prepared state do not exceed the number of component failures [[9], Theorem 3].

III. FACTORY PREPARATION OF POLAR CODE STATES

Note that the rate of the preparation p_{prep} in (6) decreases as the code length N increases. Intuitively, this is because the number of components increases with respect to N , as given in (2), thus resulting in an increased expected number of failures, and as a consequence a higher probability of error detection. Numerical results in [9] suggest that the p_{prep} decreases rapidly as N increases, hence, prohibiting the preparation of \mathcal{Q}_1 code states of larger code lengths in a practical scenario.

In this section, to improve the preparation rate, we consider a factory preparation, where several \mathcal{Q}_1 code states of length N are prepared in parallel. We modify [[9], Procedure 2], so that at some intermediate levels of recursion, we only discard \mathcal{Q}_1 code states where an error is detected, and keep all the successfully prepared intermediate states and continue the preparation for the next levels of recursion. As this allows to avoid restarting the preparation from the beginning every time an error is detected, we may achieve better preparation rate than [[9], Procedure 2].

Below we describe our factory preparation in detail.

Let $\mathcal{S}_T := \{1, \dots, TN\}$, $T \geq 1$ be a set of TN qubits, on which we want to prepare several copies of a \mathcal{Q}_1 code state of length $N = 2^n$, i.e., $|q_N\rangle$, with a given $i(n)$ value, so that $b_1 \cdots b_n \in \{0, 1\}^n$ is the binary expansion of $i(n) - 1$. We will refer to T as the *size of the factory*, as T is the maximum number of copies of $|q_N\rangle$ that can be produced. Further, consider the following ordered set:

$$n_{\text{sch}} := \{i_1, i_2, \dots, i_{|n_{\text{sch}}|}\} \subseteq \{1, \dots, n\}, \quad (7)$$

such that $0 < i_1 < i_2 < \dots < i_{|n_{\text{sch}}|}$, and $i_{|n_{\text{sch}}|} = n$. We shall refer to n_{sch} as the *scheduling set* and the elements in n_{sch} as the *scheduling recursion levels*.

We denote by $B_{i \rightarrow j}$ the recursion levels from $i + 1, \dots, j$ of Procedure 1, with respect to the binary string b_{i+1}, \dots, b_j . For the particular case $B_{0 \rightarrow j}$, we also include the initialization of the data qubits. We shall also assume that $B_{i \rightarrow j}$ incorporates the error detection gadget, as explained in Sec. II B 2, and detailed in [[9], Procedure 2]. If an error is detected at one of the recursion levels of $B_{i \rightarrow j}$, we declare a preparation failure and discard the prepared state, hence no output is produced.

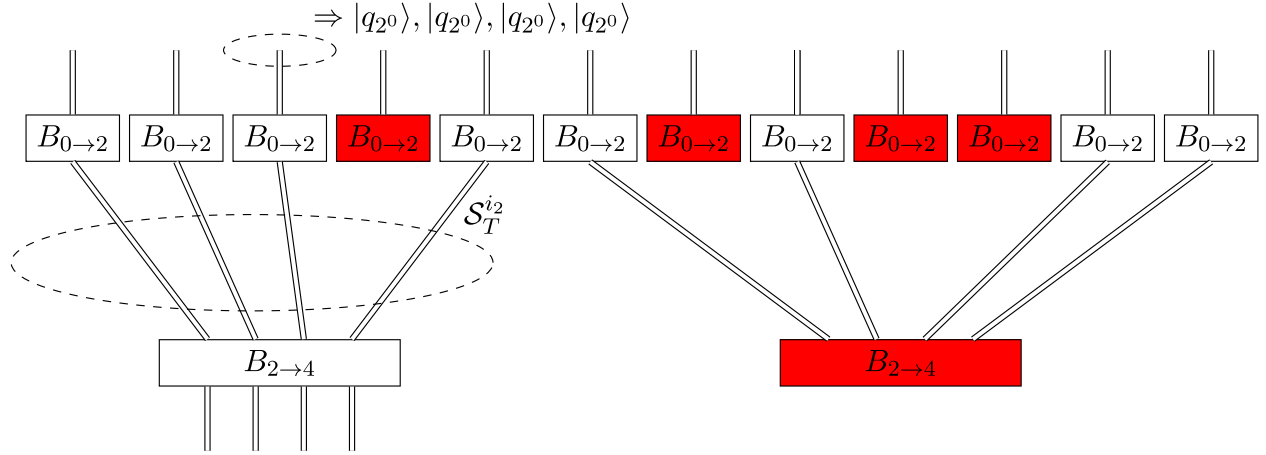


FIG. 4. An example of Procedure 2: The figure illustrates the factory preparation for $N = 16$, $T = 3$ and $n_{\text{sch}} = \{2, 4\}$, successfully preparing one copy of $|q_N\rangle$. The factory preparation proceeds from top to bottom. Each double line denotes a set of four qubits and the box $B_{i \rightarrow j}$, $0 \leq i < j \leq n$ denotes the preparation using [[9], Procedure 2] between $(i + 1)$ th and j th level of recursion, and a red box $B_{i \rightarrow j}$ signifies that an error was detected, and therefore, no output is produced. First, we initialize qubits in \mathcal{S}_T (i.e., the set of all qubits) in the Pauli Z basis. Then, we split the set \mathcal{S}_T in groups containing four qubits, which is denoted by the double wires in the figure. Then, on each group, we attempt to prepare the code state $|q_{2^2}\rangle$, by applying $B_{0 \rightarrow 2}$ (note that for the analysis in Sec. IV, we actually consider the initialization step as part of $B_{0 \rightarrow 2}$). Further, we split the successful prepared states into groups, each containing four copies of $|q_{2^2}\rangle$. On each group, we attempt to prepare $|q_{2^4}\rangle$, by applying $B_{2 \rightarrow 4}$.

If no error is detected during $B_{i \rightarrow j}$, we successfully prepare a \mathcal{Q}_1 code state of length 2^j (as shown in Fig. 4).

Our factory preparation is as follows (see also Fig. 4).

Procedure 2 (Factory Preparation). Consider the set of qubits $\mathcal{S}_T = \{1, \dots, TN\}$, $T \geq 1$, $N = 2^n$, $n > 0$, and the scheduling set $n_{\text{sch}} \subseteq \{1, \dots, n\}$ according to (7). Then, the factory preparation consists of the following steps.

(i) We first split \mathcal{S}_T in groups, each containing 2^{i_1} qubits. We then apply $B_{0 \rightarrow i_1}$ on each group.

(ii) To prepare several copies of $|q_{2^n}\rangle$, we then recursively apply the preparation $B_{i_k \rightarrow i_{k+1}}$, $1 \leq k < |n_{\text{sch}}|$, as follows.

After any scheduling recursion level i_k , $1 \leq k < |n_{\text{sch}}|$, if more than 2^{n-i_k} copies of $|q_{2^{i_k}}\rangle$ are successfully prepared, we do the following.

We split the set of successfully prepared code states $|q_{2^{i_k}}\rangle$ into groups, each containing $2^{i_{k+1}-i_k}$ copies of $|q_{2^{i_k}}\rangle$. We then attempt to prepare $|q_{2^{i_{k+1}}}\rangle$ by applying $B_{i_k \rightarrow i_{k+1}}$ on each group.

Otherwise, if less than 2^{n-i_k} copies of $|q_{2^{i_k}}\rangle$ are successfully prepared, we declare a preparation failure and discard entirely the factory preparation.

Note that, after a scheduling recursion level i_k , we need to have at least 2^{n-i_k} copies of $|q_{2^{i_k}}\rangle$ to be able to prepare at least one copy of $|q_{2^n}\rangle$. Therefore, if the number of successfully prepared state is less than 2^{n-i_k} , we discard the factory preparation. In other words, the factory preparation is successful if we have at least 2^{n-i_k} successfully prepared copies of $|q_{2^{i_k}}\rangle$, after all the scheduling recursion levels i_k , $1 \leq k \leq |n_{\text{sch}}|$.

Finally, note that for $n_{\text{sch}} = \{n\}$, the factory preparation corresponds to applying $B_{0 \rightarrow n}$, hence it is the same as the preparation [[9], Procedure 1] with error detection.

Preparation rate and error probabilities of factory preparation

Consider the factory preparation of $|q_N\rangle$ from Procedure 2, with respect to some $T \geq 1$ and $n_{\text{sch}} \subseteq \{1, \dots, n\}$. Suppose

we run the factory preparation R times, successfully preparing $t_R \geq 0$ copies of $|q_N\rangle$ in total. Then, we define the preparation rate of the factory preparation, denoted by $p_{\text{fact}}^{T, n_{\text{sch}}}$, as follows:

$$p_{\text{fact}}^{T, n_{\text{sch}}} := \lim_{R \rightarrow \infty} \frac{t_R}{RT}. \quad (8)$$

Note that for $T = 1$, $p_{\text{fact}}^{T, n_{\text{sch}}}$ is equal to p_{prep} from (6), i.e., the preparation rate of [[9], Procedure 2]. Another interesting case is when $T \rightarrow \infty$, for which we define

$$p_{\text{fact}}^{n_{\text{sch}}} := \lim_{T \rightarrow \infty} p_{\text{fact}}^{T, n_{\text{sch}}}. \quad (9)$$

Let $e_X^j, e_Z^j \in \{0, 1\}^N$ be the X and Z errors, respectively, on the $j^{\text{th}} \in \{1, \dots, t_R\}$ successfully prepared state $|q_N\rangle$. Then, the average X and Z error probabilities, denoted respectively by p_X^{prep} and p_Z^{prep} , are as follows:

$$p_X^{\text{prep}} = \lim_{R \rightarrow \infty} \frac{1}{t_R N} \sum_{j=1}^{t_R} \text{wt}(e_X^j), \quad (10)$$

$$p_Z^{\text{prep}} = \lim_{R \rightarrow \infty} \frac{1}{t_R N} \sum_{j=1}^{t_R} \text{wt}(e_Z^j), \quad (11)$$

where $\text{wt}(e_X^j)$ is the Hamming weight of $e_X^j \in \{0, 1\}^N$. From now on, we refer to p_X^{prep} and p_Z^{prep} as the X and Z preparation error probabilities, respectively. The preparation error probabilities may be used to estimate the logical error rate of \mathcal{Q}_1 code under Steane's error correction, with the help of the density evolution technique as in [9].

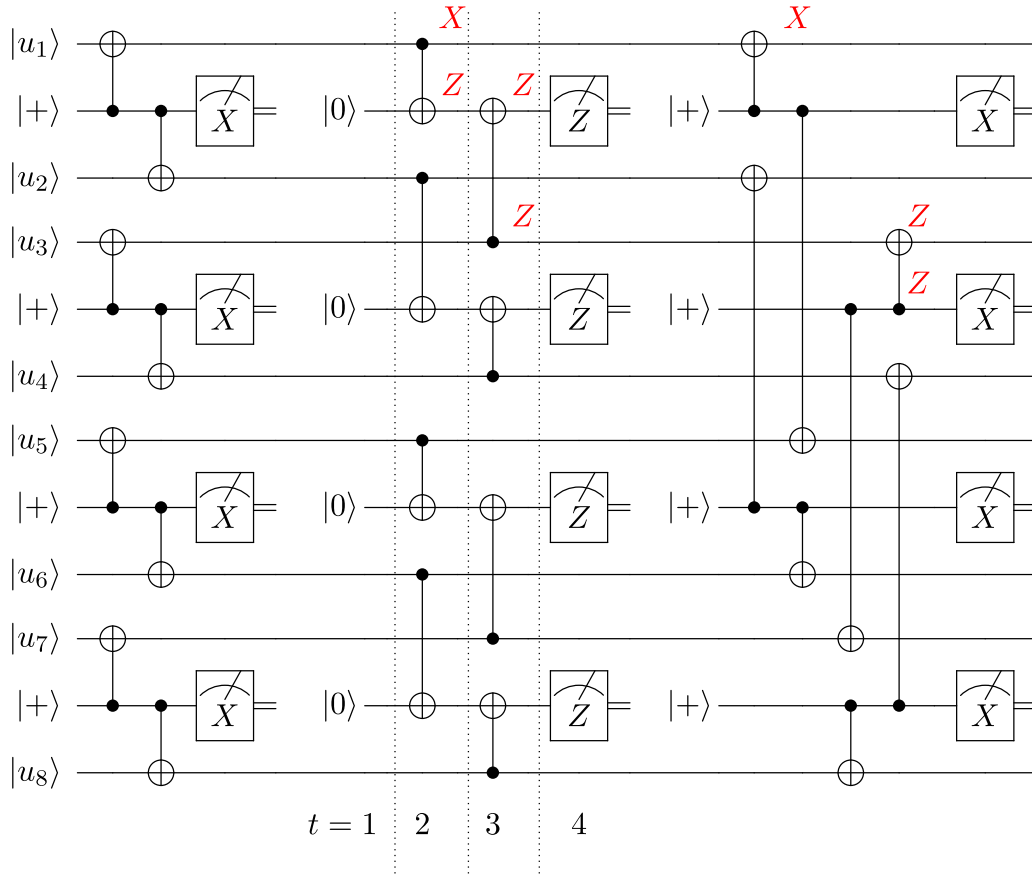


FIG. 5. An example of rough error: The figure shows a faulty measurement-based preparation (Procedure 1) of $|0_L\rangle$ corresponding to the code $\mathcal{Q}_1(N=8, i(n)=3)$, where Pauli $Z \otimes Z$ and Pauli $X \otimes X$ measurements are implemented according to the circuits in Fig. 3. A CNOT failure at $k=2, t=2$ produces the nontrivial error $X \otimes Z$, which flips the measurement outcome of a Pauli X measurement at $k=3, t=4$. Therefore, the produced error is a rough error.

IV. THEORETICAL ESTIMATES OF PREPARATION RATE AND ERROR PROBABILITIES

In this section, we define the notions of rough and smooth¹ errors for the measurement-based preparation, and then using them, we provide theoretical estimates of the preparation rate $p_{\text{fact}}^{\text{sch}}$ from (9) and preparation error probabilities p_X^{prep} and p_Z^{prep} from (10) and (11), respectively.

A. Rough and smooth errors with respect to $B_{i \rightarrow j}$

Recall that $B_{i \rightarrow j}$ corresponds to the recursion levels $i+1, \dots, j$ of the measurement-based procedure with respect to the binary string b_{i+1}, \dots, b_j , where $b_k=0$ signifies a recursion level with Pauli $X \otimes X$ measurements and $b_k=1$ a recursion level with Pauli $Z \otimes Z$ measurements. For $i=0$, $B_{i \rightarrow j}$ also includes initialization in Pauli Z basis at the zeroth level of recursion. Further, $B_{i \rightarrow j}$ takes as input 2^{j-i} copies of $|q_{2^i}\rangle$, and produces as output a copy of $|q_{2^j}\rangle$ if no error is detected.

¹The terms “smooth” and “rough” are used to distinguish X and Z boundaries in the topological quantum code literature. We emphasize that our definition is unrelated to the one used for topological codes.

Consider the set of data qubits $\mathcal{S}_{i \rightarrow j} := \{1, \dots, 2^j\}$ that are input to $B_{i \rightarrow j}$, and let $\mathcal{A}_{i \rightarrow j} := \{1, \dots, 2^{j-1}\}$ be the set of ancilla qubits used to implement the Pauli $Z \otimes Z$ and Pauli $X \otimes X$ measurements in $B_{i \rightarrow j}$. Further, let $\mathcal{C}_{i \rightarrow j}$ be the set of all components corresponding to $B_{i \rightarrow j}$. Note that $\mathcal{C}_{i \rightarrow j}$ consists of $T_{i \rightarrow j} = (j-i)2^{j-1}$ two qubit Pauli measurements. Therefore, it consists of $T_{i \rightarrow j}$ initializations and measurements of the ancilla qubits and $2T_{i \rightarrow j}$ CNOT gates between the data and ancilla qubits. If $i=0$, it also has 2^j initializations in the Pauli Z basis.

Any recursion level of $B_{i \rightarrow j}$ consists of the following four time steps (see also Fig. 5).

- ($t=1$) The ancilla qubits in $\mathcal{A}_{i \rightarrow j}$ are initialized in the Pauli Z or X basis.
- ($t=2$) The first CNOT gate (corresponding to all Pauli $X \otimes X$ or $Z \otimes Z$ measurements) are applied in parallel.
- ($t=3$) The second CNOT gate (corresponding to all Pauli $X \otimes X$ or $Z \otimes Z$ measurements) are applied in parallel.
- ($t=4$) The ancilla qubits in $\mathcal{A}_{i \rightarrow j}$ are measured in the Pauli Z or X basis.

We now define rough and smooth errors with respect to $B_{i \rightarrow j}$, as follows.

Definition 1 (Rough and smooth errors). Let $|\psi_{i \rightarrow j}^{k,t}\rangle$ be the quantum state corresponding to the joint system $\mathcal{S}_{i \rightarrow j} \cup \mathcal{A}_{i \rightarrow j}$, after a time step $t = 1, 2, 3, 4$ of the k th recursion level of $B_{i \rightarrow j}$ ($i < k \leq j$). We say a Pauli error P_e acting on $|\psi_{i \rightarrow j}^{k,t}\rangle$ is a rough error if it satisfies the following two conditions.

- (C.1) It is a nontrivial error in the sense that it not a stabilizer of the quantum state $|\psi_{i \rightarrow j}^{k,t}\rangle$.
- (C.2) It flips the outcome of at least one single qubit Pauli Z or X measurement at a recursion level k' , $k \leq k' \leq j$.

Further, we say that P_e is a smooth error if it satisfies the above condition (C.1), however, it does not satisfy the condition (C.2).

The condition (C.2) of Definition 1 is illustrated in Fig. 5 for $B_{0 \rightarrow 3}$, where the error $P_e = X \otimes Z$ happens at $k = 2, t = 2$. Recall that an X error propagates through the control of a CNOT to its target, while it simply passes through the target. An Z error propagates through the target of a CNOT gate to its control, while it simply passes through the control. Further, an error flips the outcome of a Pauli measurement if they anticommute with each other. The error P_e in Fig. 5, consisting of an X error on the first data qubit and a Z error on the first ancilla qubit, flips the outcome of an X measurement at $k = 3, t = 4$. Since P_e is not a stabilizer of $|\psi_{0 \rightarrow 3}^{2,2}\rangle$, it corresponds to a rough error.

The rough and smooth errors are related with the error detection gadget as follows. An error is detected only if it flips the outcome of at least one measurement. Hence, roughness is a necessary condition for error detection. However, it is not a sufficient condition as an error of large weight may flip several measurement outcomes and the error detection is limited by the minimum distance of the classical code. Nevertheless, due to the recursive nature of the preparation procedure, for an error to survive it should not be detected at any of the recursion levels that follow. Therefore, we expect that the rough errors will go undetected with small probabilities. This justifies the following assumption that we make for the estimation of the preparation rate and preparation error probabilities.

Assumption 1. Any rough error on the quantum state $|\psi_{i \rightarrow j}^{k,t}\rangle$, $i < k \leq j, t = 1, 2, 3, 4$, is detected by the error detection gadget at one of the recursion levels k, \dots, j .

We will further use the rough error probability and smooth error channel, defined below.

Definition 2 (Rough error probability). We define the rough error probability for a component $C \in \mathcal{C}_{i \rightarrow j}$, as the probability that C produces a rough error in $B_{i \rightarrow j}$.

Considering Assumption 1, the error that remains on the output of $B_{i \rightarrow j}$, when no error is detected, is due to the smooth errors. We define below the smooth error channel corresponding to a component.

Definition 3 (Smooth error channel). Let $L(H_{\mathcal{S}_{i \rightarrow j}})$ be the set of linear operators acting on $H_{\mathcal{S}_{i \rightarrow j}}$, the Hilbert space corresponding to $\mathcal{S}_{i \rightarrow j}$. Then, we define the smooth error channel corresponding to a component $C \in \mathcal{C}_{i \rightarrow j}$, denoted by $\mathcal{W}_C : L(H_{\mathcal{S}_{i \rightarrow j}}) \rightarrow L(H_{\mathcal{S}_{i \rightarrow j}})$, as the channel that acts on the output quantum state of $B_{i \rightarrow j}$, due to the smooth errors produced by the faults in C .

In the following, we denote by p_C and \mathcal{W}_C , the rough error probability and the smooth error channel associated with a component C , respectively. We further define $k_{i \rightarrow j}^{\min}$, as follows:

$$k_{i \rightarrow j}^{\min} := \min\{k \in \{i+1, \dots, j-1\} \mid b_{i'} = b_j, \forall k < i' \leq j\}. \quad (12)$$

In other words, $k_{i \rightarrow j}^{\min}$ is the minimum value in $\{i+1, \dots, j-1\}$, so that the recursion levels of $B_{i \rightarrow j}$ after $k_{i \rightarrow j}^{\min}$ consists of only one type of two-qubit Pauli measurements, given by the value of b_j .

We provide below p_C and \mathcal{W}_C for the initialization of data qubits (Lemma 1), initialization and measurement of ancilla qubits (Lemma 2), and CNOT gates (Lemma 3), considering the circuit level depolarizing noise model from Sec. II B 2, with the physical error rate p . The proofs of lemmas are given in Appendix.

Lemma 1. Consider the recursion levels corresponding to $B_{0 \rightarrow j}$, $j > 0$ and let C be an initialization component on a data qubit $q \in \mathcal{S}_{0 \rightarrow j}$, at the zeroth level of recursion. Then, p_C and \mathcal{W}_C are as follows:

$$p_C = \begin{cases} 0, & \text{if } \sum_{t=1}^j b_t = 0, \\ p, & \text{otherwise.} \end{cases} \quad (13)$$

$$\mathcal{W}_C = \begin{cases} I_{\mathcal{S}_{0 \rightarrow j} \setminus q} \otimes \mathcal{B}_q^p, & \text{if } \sum_{t=1}^j b_t = 0, \\ I_{\mathcal{S}_{0 \rightarrow j}}, & \text{otherwise,} \end{cases} \quad (14)$$

where $\mathcal{B}_q^{(p)}$ is a bit-flip channel, acting on q with the error probability p , i.e., $\mathcal{B}_q^{(p)}(\rho) := (1-p)\rho + pX\rho X$.

Lemma 2. For an initialization and measurement component on an ancilla qubit in $\mathcal{A}_{i \rightarrow j}$, $0 \leq i < j$, we have the following:

$$p_C = p, \quad (15)$$

$$\mathcal{W}_C = I_{\mathcal{S}_{i \rightarrow j}}. \quad (16)$$

Lemma 3. Consider a CNOT $_{q \rightarrow a}$ on the k th recursion level of $B_{i \rightarrow j}$, $0 \leq i < k \leq j$, acting between a data qubit $q \in \mathcal{S}_{i \rightarrow j}$ and an ancilla qubit $a \in \mathcal{A}_{i \rightarrow j}$. Then, depending on k , p_C is as follows:

$$p_C = \begin{cases} 8p/15, & \text{if } k = j, \\ 4p/5, & \text{if } k_{i \rightarrow j}^{\min} \leq k < j, \\ 14p/15, & \text{otherwise.} \end{cases} \quad (17)$$

Further, $\mathcal{W}_C = I_{\mathcal{S}_{i \rightarrow j} \setminus q} \otimes \mathcal{W}_q$, where \mathcal{W}_q is a quantum channel acting on q as follows:

$$\mathcal{W}_q = \begin{cases} \mathcal{D}_q^{(6p/15)}, & \text{if } k = j, \\ \mathcal{B}_q^{(2p/15)}, & \text{if } k_{i \rightarrow j}^{\min} \leq k < j, \text{ and } b_j = 0, \\ \mathcal{P}_q^{(2p/15)}, & \text{if } k_{i \rightarrow j}^{\min} \leq k < j, \text{ and } b_j = 1, \\ I_q, & \text{otherwise,} \end{cases} \quad (18)$$

where $\mathcal{D}_q^{(p)}$ is a depolarizing channel, acting on q with error probability p as $\mathcal{D}_q^{(p)}(\rho) := (1-p)\rho + \frac{p}{3}(X\rho X + Y\rho Y + Z\rho Z)$, $\mathcal{B}_q^{(p)}$ is a bit-flip channel, acting on q with the error

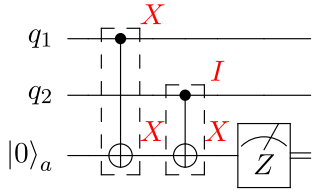


FIG. 6. Two rough errors together may produce a smooth error: The figure shows a faulty Pauli $Z \otimes Z$ measurement. The fault in $\text{CNOT}_{q_1 \rightarrow a}$ produces the error $X_{q_1} \otimes X_a$ and the fault in $\text{CNOT}_{q_2 \rightarrow a}$ produces the error $I_{q_2} \otimes X_a$. Both errors $X_{q_1} \otimes X_a$ and $I_{q_2} \otimes X_a$ are rough individually as they will flip the measurement outcome of the Pauli Z measurement on the ancilla qubit. However, when these errors are considered together, they do not flip the measurement outcome as the X errors on the ancilla qubit get canceled. Hence, the total error may not be a rough error. Note that after the measurement, the remaining error on data qubits is $X_{q_1} \otimes I_{q_2}$, which may flip a measurement outcome at one of the subsequent levels of recursion.

probability p as $\mathcal{B}_q^{(p)}(\rho) := (1 - p)\rho + pX\rho X$, and $\mathcal{P}_q^{(p)}$ is a phase-flip channel, acting on q with the error probability p as $\mathcal{P}_q^{(p)}(\rho) := (1 - p)\rho + pZ\rho Z$.

B. Success probability of $B_{i \rightarrow j}$

Considering Assumption 1, the success probability of $B_{i \rightarrow j}$ corresponds to the probability that a rough error is not produced at any of the recursion levels in $B_{i \rightarrow j}$. In this regard, we consider the following two cases.

- (1) The component failures at the previous recursion levels, i.e., $B_{0 \rightarrow i}$, producing smooth errors with respect to $B_{0 \rightarrow i}$.
- (2) The component failures at one of the recursion levels corresponding to $B_{i \rightarrow j}$.

Let $p^{(1)}$ and $p^{(2)}$ be the probability that a rough error with respect to $B_{i \rightarrow j}$ is not produced due to Point (1) and Point (2), respectively. In the following, we give approximations of $p^{(1)}$ and $p^{(2)}$, using which we may approximate the success probability of $B_{i \rightarrow j}$.

$$p_x^{0 \rightarrow i} = \begin{cases} 1 - (1 - p)(1 - 2p/15)^i, & \text{if } \sum_{t=1}^i b_t = 0, \\ 1 - (1 - 2p/15)^{(i - k_{0 \rightarrow i}^{\min}) + 1}, & \text{if } \sum_{t=1}^i b_t \neq 0, \text{ and } b_i = 0, \\ 2p/15, & \text{if } b_i = 1, \end{cases} \quad (20)$$

$$p_y^{0 \rightarrow i} = 2p/15, \quad (21)$$

$$p_z^{0 \rightarrow i} = \begin{cases} 2p/15, & \text{if } b_i = 0, \\ 1 - (1 - 2p/15)^{(i - k_{0 \rightarrow i}^{\min}) + 1}, & \text{if } b_i = 1. \end{cases} \quad (22)$$

As explained in Remark 1 and Fig. 6, a set of component failures may produce a smooth error even though individual component failures produce rough errors. Such smooth errors are not taken into account in Lemma 4, therefore, the channel $\mathcal{W}_{0 \rightarrow i}$ therein underestimates the actual errors on the output of $B_{0 \rightarrow i}$. However, as the probability of such an event is exponentially small in the number of component failures, we

We first consider Point (2). We may approximate $p^{(2)}$ directly from Lemmas 1, 2, and 3, as follows:

$$p^{(2)} \approx \prod_{C \in \mathcal{C}_{i \rightarrow j}} (1 - p_C). \quad (19)$$

Recall that p_C is the rough error probability associated with the component C . We have the following remark.

Remark 1. We note that the total error produced by component failures in a set $\mathcal{C}' \subseteq \mathcal{C}_{i \rightarrow j}$ (containing two or more components, i.e., $|\mathcal{C}'| \geq 2$) can be a smooth error, even though individual component failures in \mathcal{C}' produce rough errors (see Fig. 6). To get the actual value of $p^{(2)}$, we need to add in the right-hand side (RHS) of (19), the probabilities of the events for all the subsets $\mathcal{C}' \subseteq \mathcal{C}_{i \rightarrow j}$, $|\mathcal{C}'| \geq 2$, where each component in \mathcal{C}' produces a rough error individually but the total error is a smooth error. As the probability of such an event is upper bounded by $\prod_{C \in \mathcal{C}'} p_C$, we expect the RHS in (19) to be a good lower bound of $p^{(2)}$.

In the next paragraph, we give an approximation of $p^{(1)}$.

Rough error probability due to previous recursion levels. We first estimate the errors that remain on the input of $B_{i \rightarrow j}$ due to the smooth errors in previous recursion levels, i.e., $B_{0 \rightarrow i}$. We then estimate the probability that this remaining error is a rough error with respect to $B_{i \rightarrow j}$ (hence, giving an estimate of $p^{(1)}$).

To estimate the remaining error on the input of $B_{i \rightarrow j}$, we simply concatenate the smooth error channels corresponding to components in $\mathcal{C}_{0 \rightarrow i}$. The resulting channel after concatenation is a Pauli channel as given in Lemma 4, which is proven in Appendix.

Lemma 4. Let $\mathcal{W}_{0 \rightarrow i} : L(H_{\mathcal{S}_{0 \rightarrow i}}) \rightarrow L(H_{\mathcal{S}_{0 \rightarrow i}})$ be the quantum channel that corresponds to the concatenation of smooth error channels for all $C \in \mathcal{C}_{0 \rightarrow i}$, i.e., $\mathcal{W}_{0 \rightarrow i} = \mathcal{W}_{C_1} \circ \dots \circ \mathcal{W}_{C_{|\mathcal{C}_{0 \rightarrow i}|}}$. Then, $\mathcal{W}_{0 \rightarrow i} = \otimes_{q \in \mathcal{S}_{0 \rightarrow i}} \mathcal{W}_q$ (i.e., $\mathcal{W}_{0 \rightarrow i}$ acts independently and identically on qubits in $\mathcal{S}_{0 \rightarrow i}$), where \mathcal{W}_q is a Pauli channel, whose X , Y , and Z error probabilities are, respectively, upper bounded by $p_x^{0 \rightarrow i}$, $p_y^{0 \rightarrow i}$, and $p_z^{0 \rightarrow i}$, which are as follows:

expect $\mathcal{W}_{0 \rightarrow i}$ approximates well the actual channel acting on the output of $B_{0 \rightarrow i}$.

We now estimate the probability that the channel $\mathcal{W}_{0 \rightarrow i}$ in Lemma 4 produces a rough error with respect to $B_{i \rightarrow j}$. Note that an X (or Z) error on the qubit $q \in \mathcal{S}_{i \rightarrow j}$ at the input of $B_{i \rightarrow j}$, will flip the measurement outcome of the next Pauli $Z \otimes Z$ (or $X \otimes X$) measurement. Therefore, using (20), (21), and

(22), the probability of preexisting error on a qubit q to be rough in $B_{i \rightarrow j}$ is upper bounded as

$$p_{\text{pre}} = \begin{cases} p_y^{0 \rightarrow i} + p_z^{0 \rightarrow i}, & \text{if } \sum_{t=i+1}^j b_t = 0, \\ p_x^{0 \rightarrow i} + p_y^{0 \rightarrow i}, & \text{if } \sum_{t=i+1}^j b_t = j - i, \\ p_x^{0 \rightarrow i} + p_y^{0 \rightarrow i} + p_z^{0 \rightarrow i}, & \text{otherwise.} \end{cases} \quad (23)$$

Using (23) and the fact that there are 2^j qubits in $S_{i \rightarrow j}$, the probability that the preexisting errors do not produce a rough error with respect to $B_{i \rightarrow j}$ is given by

$$p^{(1)} \approx (1 - p_{\text{pre}})^{2^j}. \quad (24)$$

Note that (24) is an approximation due to the fact that several individual rough errors may be one smooth error when considered together, as previously explained.

Finally, from (24) and (19), the probability that no-rough error happens in $B_{i \rightarrow j}$ (in other words, the success probability of $B_{i \rightarrow j}$ using Assumption 1) can be approximated as follows:

$$p_{i \rightarrow j}^s \approx (1 - p_{\text{pre}})^{2^j} \prod_{C \in \mathcal{C}_{i \rightarrow j}} (1 - p_C). \quad (25)$$

C. Preparation rate of the factory preparation

Using (25), we now estimate the preparation rate for the asymptotically large factory size, that is, $T \rightarrow \infty$ as in (9). Suppose after the i_k^h , $i_k \in n_{\text{sch}}$ scheduling recursion level, we have total T_{i_k} successfully prepared \mathcal{Q}_1 code-states of length 2^{i_k} . As in Procedure 2, we split successfully prepared code states into $\lceil T_{i_k} / (2^{i_{k+1} - i_k}) \rceil$ groups, each group containing $2^{i_{k+1} - i_k}$ code states. We then apply $B_{i_k \rightarrow i_{k+1}}$ on each group to prepare \mathcal{Q}_1 code states of length $2^{i_{k+1}}$. Using (25) and the law of large numbers, we may estimate the number of prepared state after $B_{i_k \rightarrow i_{k+1}}$ as follows:

$$T_{i_{k+1}} \approx \lceil T_{i_k} / (2^{i_{k+1} - i_k}) \rceil p_{i_k \rightarrow i_{k+1}}^s. \quad (26)$$

Using (26), we have

$$p_{i_k \rightarrow i_{k+1}}^s \approx \frac{T_{i_{k+1}} (2^{i_{k+1} - i_k})}{T_{i_k}}. \quad (27)$$

Further, applying (27) recursively, we may get an estimate of the preparation rate in (9), as follows (we take $i_0 = 0$ and recall $i_{|n_{\text{sch}}|} = n$),

$$p_{\text{fact}}^{n_{\text{sch}}} = \frac{T_{i_{|n_{\text{sch}}|}}}{T} \quad (28)$$

$$= \prod_{k=0}^{i_{|n_{\text{sch}}|} - 1} \frac{T_{i_{k+1}} (2^{i_{k+1} - i_k})}{T_{i_k}} \quad (29)$$

$$\approx \prod_{k=0}^{i_{|n_{\text{sch}}|} - 1} p_{i_k \rightarrow i_{k+1}}^s, \quad (30)$$

where we use $T_{i_0} = T_0 = NT$ in the second equality, and approximation in the last line follows from (27). Therefore, from (30), the factory preparation rate for $T \rightarrow \infty$ is simply a multiplication of the success probabilities of the blocks $B_{i_k \rightarrow i_{k+1}}$, $k = 0, \dots, i_{|n_{\text{sch}}|} - 1$. Further, from (25) and (30), we

may roughly see as follows why factory preparation may improve the preparation rate compared to preparation in [9]. As noted before, preparation in [9] corresponds to $n_{\text{sch}} = \{n\}$, for which we get $p_{\text{fact}}^{n_{\text{sch}}} \approx p_{0 \rightarrow n}^s$ from (30). Further from (25), for $p_{0 \rightarrow n}^s$, we need to consider component failures in the set $\mathcal{C}_{0 \rightarrow n}$ for error detection, while for $\prod_{k=0}^{i_{|n_{\text{sch}}|} - 1} p_{i_k \rightarrow i_{k+1}}^s$ corresponding to a $n_{\text{sch}} = \{i_1, \dots, i_k, \dots, i_{|n_{\text{sch}}|}\} \supset \{n\}$, we need to consider component failures in the smaller set $\mathcal{C}_{0 \rightarrow i_1} \cup \dots \cup \mathcal{C}_{i_{k-1} \rightarrow i_k} \cup \mathcal{C}_{i_k \rightarrow i_{k+1}} \dots \cup \mathcal{C}_{i_{|n_{\text{sch}}| - 1} \rightarrow i_{|n_{\text{sch}}|}} \subset \mathcal{C}_{0 \rightarrow n}$.

D. Preparation error probabilities of the factory preparation

The remaining error on the final prepared state is due to the smooth errors in $B_{0 \rightarrow n}$, hence given by the quantum channel $\mathcal{W}_{0 \rightarrow n}$ as in Lemma 4. The action of $\mathcal{W}_{0 \rightarrow n}$ on each qubit $q \in \mathcal{S}_{0 \rightarrow j}$ corresponds to the Pauli channel, with the X , Y , and Z error probabilities $p_x^{0 \rightarrow n}$, $p_y^{0 \rightarrow n}$, and $p_z^{0 \rightarrow n}$ according to (20), (21), and (22), respectively. Therefore, the preparation X and Z error probabilities in (10) and (11), respectively, are given by

$$p_X^{\text{prep}} = p_x^{0 \rightarrow n} + p_y^{0 \rightarrow n}, \quad (31)$$

$$p_Z^{\text{prep}} = p_y^{0 \rightarrow n} + p_z^{0 \rightarrow n}. \quad (32)$$

V. NUMERICAL RESULTS

In this section, we present our numerical results regarding the factory preparation rate and logical error rates, using our theoretical estimates in Sec. IV as well as a Monte Carlo simulation, considering the circuit level depolarizing noise model from Sec. II B 2.

A. Preparation rate

For the Monte Carlo simulation of the preparation rate, we proceed as follows. We simulate R times the factory preparation according to Procedure 2, for a factory size T . Let t_i be the number of successfully prepared states for the i th instance of the factory preparation, $1 \leq i \leq R$. Then, we determine $p_{\text{fact}}^{T, n_{\text{sch}}}$ as follows:

$$p_{\text{fact}}^{T, n_{\text{sch}}} = \frac{1}{RT} \sum_{i=1}^R t_i. \quad (33)$$

We obtain the value of $p_{\text{fact}}^{T, n_{\text{sch}}}$ for T values $1, 2^1, \dots, 2^{10}$. For the fixed physical error rate $p = 10^{-3}$, the factory preparation rate $p_{\text{fact}}^{T, n_{\text{sch}}}$, with respect to T , is shown in Fig. 7(a) for $\mathcal{Q}_1(N = 64, i = 23)$ and $\mathcal{Q}_1(N = 256, i = 91)$, with scheduling sets $n_{\text{sch}} = \{2, 4, 6\}$ and $n_{\text{sch}} = \{2, 4, 6, 8\}$, respectively. The information positions are chosen according to [[9], Table II] for ignoring correlations.

We observe that $p_{\text{fact}}^{T, n_{\text{sch}}}$ increases with respect to T in the beginning, and then it saturates. The saturated value corresponds to the preparation rate with respect to $T \rightarrow \infty$, i.e., $p_{\text{fact}}^{n_{\text{sch}}}$ from (9). We take the saturated value to be the value of $p_{\text{fact}}^{T, n_{\text{sch}}}$ for $T = 2^{10}$.

The difference between the saturated value and the value of p_{prep} from (6) (i.e., $p_{\text{fact}}^{T, n_{\text{sch}}}$ for $T = 1$) is quite significant, especially for $N = 256$. In particular, for $N = 64, 256$, the value of p_{prep} is around 47%, 2%, respectively, while the saturated

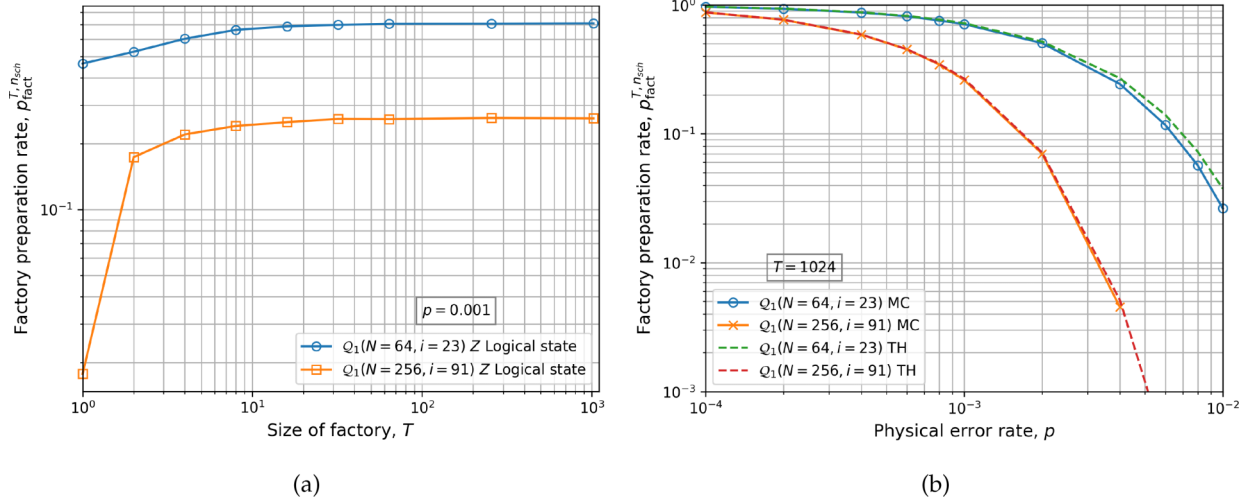


FIG. 7. Factory preparation rate: We consider codes $Q_1(N=64, i=23)$ and $Q_1(N=256, i=91)$, and scheduling sets $n_{\text{sch}} = \{2, 4, 6\}$ and $n_{\text{sch}} = \{2, 4, 6, 8\}$, respectively, for the factory preparation according to Procedure 2. Panel (a) shows factory preparation rate with respect to the size of the factory T for a fixed value of physical error rate $p = 10^{-3}$, obtained using a Monte Carlo simulation. Panel (b) shows factory preparation rate with respect to the physical error rate p for a fixed value of the factory size $T = 1024$, obtained using a Monte Carlo (MC) simulation and also using theoretical (TH) estimates from Sec. IV C.

value is around 70%, 27%. Therefore, the factory preparation provides significant improvement in the preparation rate compared to [[9], Procedure 2]. Further, the saturation happens rather quickly, for example the value of $p_{\text{fact}}^{T, n_{\text{sch}}}$ for $T = 8$ is already quite close to its value for $T = 1024$ for both $N = 64, 256$. This means that we do not need a large factory size to get the increased preparation rate of the factory preparation.²

We further obtain the value of $p_{\text{fact}}^{n_{\text{sch}}}$ using our theoretical estimate in (25) and (30). In Fig. 7(b), we present both the Monte Carlo and theoretical values of $p_{\text{fact}}^{n_{\text{sch}}}$ with respect to the physical error rate p . We observe that the curves corresponding to Monte Carlo simulation and theoretical estimates are very close, thus validating that our method of theoretical estimation is a good approximation of reality.

B. Logical error rate

In this section, we estimate the logical error rates of Q_1 codes, using the Steane error correction, which uses the ancilla code states (logical $|0\rangle$ and $|+\rangle$ states), prepared by the factory preparation according to Procedure 2.

To do so, we use the density evolution technique as in [[9], Appendix E.4]. Precisely, using the X and Z error probabilities on the prepared states, given by (31) and (32), we first estimate the input error probability for the two decoders used within the Steane error correction procedure, as described in [[9], Appendices E.3 and E.4] [see also Eqs. (E10) and (E11) therein]. Then, we use density evolution to estimate the output error probability of the two decoders, from which we determine the

X and Z logical error rates, P_X^L and P_Z^L [[9], Eq. (E12)], and then the (total) logical error rate $P_e^L = P_X^L + P_Z^L - P_X^L P_Z^L$.

In Fig. 8, we present the logical error rate versus physical error rate curves for $N = 64, 256$. For $N = 64$, we include the density evolution based curves as well as the Monte Carlo simulation based curves from [[9], Fig. 4]. We observe that the density evolution curves virtually superimpose the Monte Carlo curves, therefore, substantiating our theoretical method of estimating preparation error probabilities.

For $N = 256$, we only include the density evolution curves, as the logical error rates are very small to be simulated using the Monte Carlo simulation, and also the preparation rates are comparatively smaller. As expected, the Q_1 codes perform much better than the Shor- Q_1 codes. Note that the

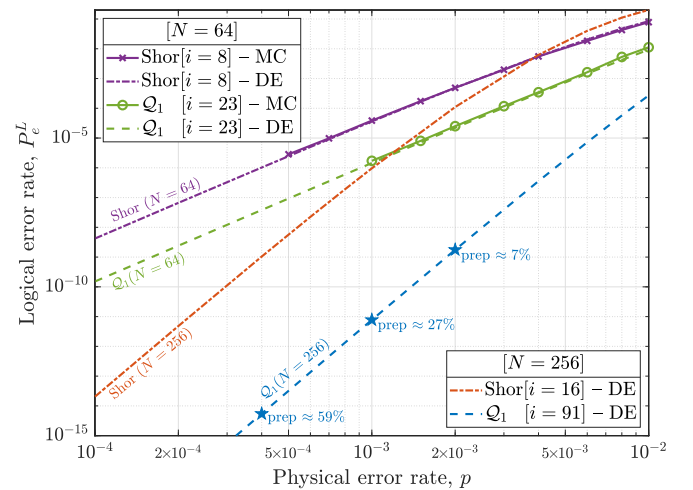


FIG. 8. Logical error rates of codes $Q_1(N=64, i=23)$ and $Q_1(N=256, i=91)$, using Steane error correction, where ancilla code states are prepared using the factory preparation in Procedure 2. The curves are obtained either based on a Monte Carlo (MC) simulation or a density evolution (DE) method.

²Note that the inverse of the preparation rate contributes to the qubit overhead of the preparation, hence, the factory preparation reduces significantly the qubit overhead of Q_1 code-state preparation. In particular, for $N = 256, p = 10^{-3}$, it reduces the qubit overhead by a factor of around 13.

TABLE I. Preparation and logical error rates for $N = 1024$.

p	0.001	0.0004	0.0002	0.0001
$p_{\text{fact}}^{\text{sch}}$	0.5%	12%	35%	59%
P_e^L	4.08×10^{-22}	2.04×10^{-28}	3.23×10^{-33}	2.69×10^{-38}

performance of the \mathcal{Q}_1 code for $N = 64$ is better than the performance of the Shor- \mathcal{Q}_1 code for $N = 256$, down to a physical error rate $p = 10^{-3}$. Remarkably, the \mathcal{Q}_1 code for $N = 256$ achieves logical error rates around 10^{-11} and 10^{-15} for physical error rates 10^{-3} and 3×10^{-4} , respectively, which is very promising for the practical large-scale fault-tolerant quantum computation [12].

Finally, for $N = 1024$, our numerical results are given in Table I. For a practically interesting range of physical error rates $p \in [10^{-4}, 10^{-3}]$, we observe that \mathcal{Q}_1 code of length $N = 256$ is the best choice due to good preparation rate and sufficiently low logical error rates. Increasing the code length may be useful for physical error rates above 10^{-3} , however, in this case the $N = 1024$ code is penalized by its poor preparation rate.

C. Comparison with the surface code

In Fig. 9, we compare the error correction performance of \mathcal{Q}_1 and Shor \mathcal{Q}_1 codes of length $N = 256$ and minimum distance $d = 16$, with that of a surface code with minimum distance $d = 15$, assuming a circuit level depolarizing noise model. For the surface code, the simulation results are taken from [12]. Note that taking $d = 16$ for the surface code only increases the code length N , but not the error correction performance. For $d = 15$, the code length reported in [12] is $N = 421$, however, it can be reduced to $N = 225$, by considering a rotated variant of the surface code. The logical error rate of the surface code is simulated in [12] down to a physical error rate $p = 5 \times 10^{-3}$. We extrapolated the logical error rate for lower physical error rates using $P_e^L = c(p/p_{th})^{\frac{d+1}{2}}$, as

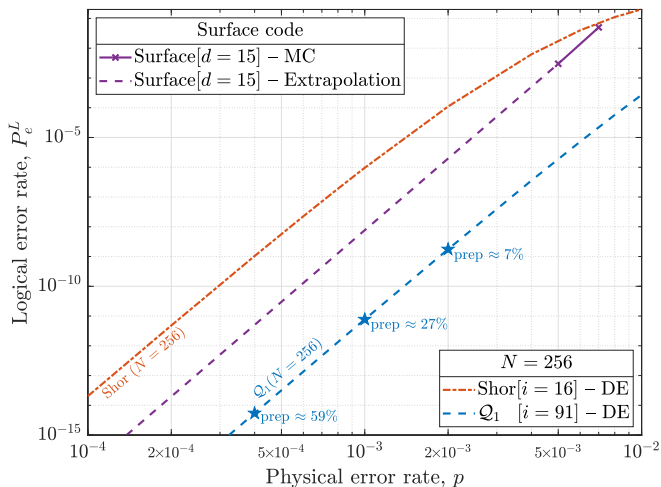


FIG. 9. Comparison of logical error rates of \mathcal{Q}_1 ($N = 256$, $i = 16$) and \mathcal{Q}_1 ($N = 256$, $i = 91$), with that of the surface code, with code length $N = 225$, and minimum distance $d = 15$ from [12].

proposed in [12], where $p_{th} = 0.0057$ is the reported surface code threshold.

We can see that the \mathcal{Q}_1 code outperforms the surface code by about three orders of magnitude. For example, for practically interesting physical error rates $p = 10^{-3}$ and $p = 5 \times 10^{-4}$, the corresponding logical error rates for the surface code are around 10^{-8} and 5×10^{-11} , while for \mathcal{Q}_1 codes are around 10^{-11} and 5×10^{-14} . This is an encouraging result for \mathcal{Q}_1 codes and shows that they are of independent interest in the context of fault tolerant quantum computing.

However, we should consider the above comparison carefully, as error correction procedures are different for \mathcal{Q}_1 and surface codes. While Steane error correction is natural for \mathcal{Q}_1 codes, generator measurement-based error correction is natural for surface codes due to their small weight generators. On the one hand, Steane error correction is advantageous in the sense that one round of syndrome extraction is enough for error correction, while for generator measurement-based error correction, several rounds of syndrome extraction are needed. Therefore, error correction for polar codes can be faster than for surface codes. On the other hand, generator measurement-based error correction is advantageous in the sense that ancilla qubits can be intercalated between data qubits and are directly reusable after each round of syndrome extraction. However, Steane error correction needs a separate ancilla factory running, to produce ancilla states needed for error correction. Ancilla qubits used in a round of error correction will be moved back to the factory so that they are reused.

Finally, we note that the factory preparation (see also Fig. 5) requires distant CNOT gates, i.e., interaction between nonneighboring qubits, as opposed to the surface codes. Although distant CNOT gate is possible on some potential quantum systems such as ion traps [13,14], there are ways to circumvent this for quantum systems with local interaction constraint such as by applying swap gates [15,16] or physically moving qubits around [17–21].

Recently, quantum error correcting codes such as small surface and color codes, as well as three-dimensional codes, have been implemented on a reconfigurable quantum architecture (based on Rydberg atoms), with storage, entangling, and readout zones [17,18]. Qubits therein are moved around within a zone or between zones to achieve long-range connectivity. Moreover, it is worth noticing that the three-dimensional $[[8,3,2]]$ code implemented in [17] is actually very similar to a length-8 polar code, both codes using hypercube connectivity. As \mathcal{Q}_1 codes provide better error correction performance exploiting distant operations, they are naturally suited to this kind of architecture. Moreover, the reconfigurable quantum architecture allows to move blocks of qubits in parallel, realizing transversal CNOT gate on two code blocks in parallel. As our preparation is based on recursively applying transversal measurements between two blocks of qubits, it may be implemented in a similar way.

VI. DISCUSSION

We proposed a factory preparation of \mathcal{Q}_1 code states, which is shown to be a useful extension of the measurement based preparation in [9], providing much better preparation rates comparatively. Its better preparation rate is owed to a

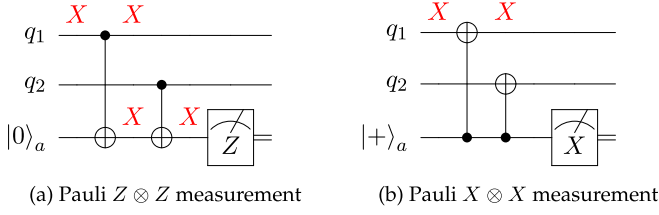


FIG. 10. Propagation of initialization errors through Pauli $Z \otimes Z$ and $X \otimes X$ measurement.

scheduling step, which makes clever use of the \mathcal{Q}_1 code states prepared at the intermediate levels of recursion. Thanks to the factory preparation, we are able to prepare code states of lengths $N = 256, 1024$, with reasonably high preparation rates for a practically interesting physical error rate range $10^{-4} - 10^{-3}$. Further, it is shown that for $N = 256, 1024$ and a physical error rate in the range $10^{-4} - 10^{-3}$, the \mathcal{Q}_1 code achieves a logical error rate below 10^{-15} , which is currently estimated to be the required logical error rate for large-scale fault-tolerant quantum computation [12]. Due to a higher preparation rate, smaller number of qubits, and also achieving a sufficiently low logical error rate, the code length $N = 256$ is the best \mathcal{Q}_1 code in this physical error range.

To estimate the preparation rate and logical error rates of \mathcal{Q}_1 codes, we used a theoretical framework based on the new notions of smooth and rough errors. It is shown that estimates based on our theoretical framework fit well the estimates obtained using Monte Carlo simulations, therefore substantiating the accuracy of the theoretical framework. We note that our notions of smooth and rough errors are not particular to \mathcal{Q}_1 codes, and therefore, as a natural future direction, it would be interesting to analyze other fault-tolerant protocols using these notions, especially the ones based on error detection [22,23].

ACKNOWLEDGMENT

This work was supported by the QuantERA grant EQUIP, by the French Agence Nationale de la Recherche, ANR-22-QUA2-0005-01.

APPENDIX: PROOFS OF LEMMAS 1, 2, 3, AND 4

1. Proof of Lemma 1

Lemma 1 follows from the fact that the initialization at the zeroth level of recursion produces an X error, which flips the outcome of a Pauli $Z \otimes Z$, while it does not have any effect on Pauli $X \otimes X$ measurement as shown in Fig. 10.

Proof of Lemma 2. Consider the circuit implementing Pauli $Z \otimes Z$ measurement from Fig. 3. Note that a failure in initialization on the ancilla qubit produces an X error on it, which will flip the outcome of the next Pauli Z measurement. A failure in Pauli Z measurement produces an X error just before the measurement, hence it will also flip the measurement outcome. Therefore, the measurement outcome will be flipped with probability 1 if an initialization or a measurement error happens on the ancilla qubit. Therefore, it follows that

$$p_C = p, \quad (\text{A1})$$

$$\mathcal{W}_C = I. \quad (\text{A2})$$

Similarly, it can be proven for Pauli $X \otimes X$ measurement.

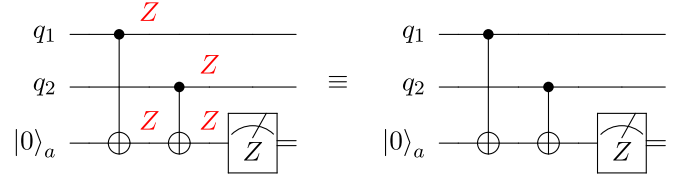


FIG. 11. A fault in $\text{CNOT}_{q_1 \rightarrow a}$ leading to a stabilizer error.

a. Proof of Lemma 3

We prove below Lemma 3 for the CNOT gate applied at the time step $t = 2$ of a recursion level in $B_{i \rightarrow j}$, with Pauli $Z \otimes Z$ measurements. It can be similarly seen that Lemma 3 also holds for $t = 3$, and as well as for the CNOT gates applied in a Pauli $X \otimes X$ measurement.

Consider a Pauli $Z \otimes Z$ measurement applied at some k th recursion level corresponding to $B_{i \rightarrow j}$. We shall denote the data qubits on which Pauli $Z \otimes Z$ measurement acts on by q_1, q_2 and the ancilla qubit by a , as in Fig. 11. Below, we classify the errors produced by the first cnot gate into rough and smooth errors, as per Definition 1, and then, using the set of rough and smooth errors, we compute p_C and \mathcal{W}_C for $C = \text{CNOT}_{q_1 \rightarrow a}$.

First, note that if a failure in C produces the error $Z_{q_1} \otimes X_a$, it propagates as $Z_{q_1} \otimes Z_{q_2}$ after the measurement, which is a stabilizer of the quantum state corresponding to the joint system $q_1 q_2$, and hence this error can be ignored (see Fig. 11). Further, if a failure in C produces an X or Y error on the ancilla qubit a , it will flip the measurement outcome of the corresponding Pauli $Z \otimes Z$ measurement, hence such an error is a rough error. Precisely, the following errors produce an X or Y error on the ancilla

$$I_{q_1} \otimes X_a, X_{q_1} \otimes X_a, Z_{q_1} \otimes X_a, Y_{q_1} \otimes X_a, I_{q_1} \otimes Y_a, X_{q_1} \otimes Y_a, Z_{q_1} \otimes Y_a, Y_{q_1} \otimes Y_a. \quad (\text{A3})$$

The remaining errors, i.e., $X_{q_1} \otimes I_a, Y_{q_1} \otimes I_a, Z_{q_1} \otimes I_a, I_{q_1} \otimes Z_a, X_{q_1} \otimes Z_a, Y_{q_1} \otimes Z_a$ do not flip the outcome of the measurement and propagate to the following errors after the measurement (up to the stabilizer $Z_{q_1} \otimes Z_{q_2}$, similar to Fig. 11), $X_{q_1} \otimes I_{q_2}, Y_{q_1} \otimes I_{q_2}, Z_{q_1} \otimes I_{q_2}, Z_{q_1} \otimes I_{q_2}, X_{q_1} \otimes I_{q_2}, Y_{q_1} \otimes I_{q_2}$, respectively. Hence, the remaining errors only act nontrivially on the qubit q_1 , and they correspond to a depolarizing channel, with error probability $6p/15$ as follows, $\mathcal{D}_{q_1}^{(6p/15)}(\rho) := (1 - \frac{6p}{15})\rho + \frac{2p}{15}(X\rho X + Y\rho Y + Z\rho Z)$. Some or all of the remaining errors may also be rough errors depending on whether they flip a measurement outcome at one of the next recursion levels. We have the following four cases in order.

If k is the last recursion level, i.e., $k = j$. In this case, p_C is simply given by the errors that flip the measurement outcome of the corresponding Pauli $Z \otimes Z$ measurement at the k th recursion level. Therefore $p_C = 8p/15$. Further, \mathcal{W}_C is the remaining noise channel after the k th recursion level. Therefore, $\mathcal{W}_C = I_{S_{i \rightarrow j} \setminus q_1} \otimes \mathcal{D}_{q_1}^{(6p/15)}$.

If $k_{i \rightarrow j}^{\min} \leq k < j$ with $b_j = 0$ (hence, only Pauli $X \otimes X$ measurements after the k th recursion level.) In this case, the remaining Z error after the k th level of recursion will flip a Pauli $X \otimes X$ measurement at the $(k + 1)$ th level of recursion,

while the remaining X errors will not flip any measurement outcome. This implies that the depolarizing channel $\mathcal{D}_{q_1}^{(6p/15)}$ transforms into a bit-flip channel, denoted by $\mathcal{B}_{q_1}^{(2p/15)}$, acting on q_1 with error probability $2p/15$. Therefore, $p_C = 12p/15$ and $\mathcal{W}_C = I_{\mathcal{S}_{i \rightarrow j} \setminus q_1} \otimes \mathcal{B}_{q_1}^{(2p/15)}$.

If $k_{i \rightarrow j}^{\min} \leq k < j$ with $b_j = 1$ (hence, only Pauli $Z \otimes Z$ measurements after the k th recursion level.) In this case, the remaining X error after the k th level of recursion, will flip a Pauli $Z \otimes Z$ measurement at the $(k+1)$ th level of recursion, while the remaining Z errors will not flip any measurement outcome. This implies that the depolarizing channel $\mathcal{D}_{q_1}^{(6p/15)}$ transforms into a phase-flip channel, denoted by $\mathcal{P}_{q_1}^{(2p/15)}$, acting on q_1 with error probability $2p/15$. Therefore, $p_C = 12p/15$ and $\mathcal{W}_C = I_{\mathcal{S}_{i \rightarrow j} \setminus q_1} \otimes \mathcal{P}_{q_1}^{(2p/15)}$.

If $k < k_{i \rightarrow j}^{\min}$. In this case, we have both Pauli $Z \otimes Z$ and $X \otimes X$ measurements afterwards, hence both the remaining X and Z errors after the k th recursion level will be detected in one of the next recursion levels. Therefore, the depolarizing channel $\mathcal{D}_{q_1}^{(6p/15)}$ transforms into the identity channel. Hence, $p_C = 14p/15$ and $\mathcal{W}_C = I_{\mathcal{S}_{i \rightarrow j} \setminus q_1} \otimes I_{q_1}$.

In summary, we have the following for p_C and \mathcal{W}_C :

$$p_C = \begin{cases} 8p/15, & \text{if } k = j, \\ 4p/5, & \text{if } k_{i \rightarrow j}^{\min} \leq k < j, \\ 14p/15, & \text{otherwise.} \end{cases} \quad (\text{A4})$$

Further, $\mathcal{W}_C = I_{\mathcal{S}_{i \rightarrow j} \setminus q} \otimes \mathcal{W}_q$, where \mathcal{W}_q is a quantum channel acting on q as follows:

$$\mathcal{W}_q = \begin{cases} \mathcal{D}_q^{(6p/15)}, & \text{if } k = j, \\ \mathcal{B}_q^{(2p/15)}, & \text{if } k_{i \rightarrow j}^{\min} \leq k < j, \text{ with } b_j = 0, \\ \mathcal{P}_q^{(2p/15)}, & \text{if } k_{i \rightarrow j}^{\min} \leq k < j, \text{ with } b_j = 1, \\ I_q, & \text{otherwise.} \end{cases} \quad (\text{A5})$$

b. Proof of Lemma 4

From Lemma 1, the smooth error channel corresponding to initialization at the zeroth level of recursion (with respect to $k_{0 \rightarrow i}^{\min}$) is the identity, except when $\sum_{t=1}^i b_t = 0$ (it implies that $k_{0 \rightarrow i}^{\min} = 1$), it is a bit-flip channel on the corresponding qubit at the output of $B_{0 \rightarrow i}$, as in (14). Further, from Lemma 2, the smooth error channel corresponding to initialization and measurement of the ancilla qubit is always the identity. Furthermore, from Lemma 3, the smooth error channel corresponding to CNOT gates for recursion levels $k < k_{0 \rightarrow i}^{\min}$ is also the identity. For $k \geq k_{0 \rightarrow i}^{\min}$, it is either a depolarizing, bit or phase-flip channel as given in (18), acting on the data qubit on which the CNOT gate acts. Hence, from (14) and (18), and noting that one CNOT gate acts on a data qubit $q \in \mathcal{S}_{i \rightarrow j}$ at each recursion level (see Fig. 5), it follows that the total smooth channel on each qubit $q \in \mathcal{S}_{0 \rightarrow i}$ is an (i.i.d.) Pauli channel as follows:

$$\mathcal{W}_q = \begin{cases} \mathcal{B}_q^{(p)} \circ (\mathcal{B}_q^{(2p/15)} \circ \dots \circ \mathcal{B}_q^{(2p/15)}) \circ \mathcal{D}_q^{(6p/15)}, & \text{if } \sum_{t=1}^i b_t = 0, \\ (\mathcal{B}_q^{(2p/15)} \circ \dots \circ \mathcal{B}_q^{(2p/15)}) \circ \mathcal{D}_q^{(6p/15)}, & \text{if } \sum_{t=1}^i b_t \neq 0 \text{ and } b_i = 0, \\ (\mathcal{P}_q^{(2p/15)} \circ \dots \circ \mathcal{P}_q^{(2p/15)}) \circ \mathcal{D}_q^{(6p/15)}, & \text{if } b_i = 1. \end{cases} \quad (\text{A6})$$

By simplifying (A6), we can see that \mathcal{W}_q is a Pauli channel, whose X , Y , and Z error probabilities are, respectively, upper bounded by $p_x^{0 \rightarrow i}$, $p_y^{0 \rightarrow i}$, and $p_z^{0 \rightarrow i}$, which are given by

$$p_x^{0 \rightarrow i} = \begin{cases} 1 - (1-p)(1-2p/15)^i, & \text{if } \sum_{t=1}^i b_t = 0, \\ 1 - (1-2p/15)^{(i-k_{0 \rightarrow i}^{\min})+1}, & \text{if } \sum_{t=1}^i b_t \neq 0 \text{ and } b_i = 0, \\ 2p/15, & \text{if } b_i = 1. \end{cases} \quad (\text{A7})$$

$$p_y^{0 \rightarrow i} = 2p/15, \quad (\text{A8})$$

$$p_z^{0 \rightarrow i} = \begin{cases} 2p/15, & \text{if } b_i = 0, \\ 1 - (1-2p/15)^{(i-k_{0 \rightarrow i}^{\min})+1}, & \text{if } b_i = 1. \end{cases} \quad (\text{A9})$$

We note that $p_y^{0 \rightarrow i}$ is equal to the Y error probability. Furthermore, $p_x^{0 \rightarrow i}$ is equal to the X error probability for $b_i = 1$, and $p_z^{0 \rightarrow i}$ is equal to the Z error probability, when $b_i = 0$.

- [1] E. Arikan, Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels, *IEEE Trans. Inf. Theory* **55**, 3051 (2009).
- [2] F. Dupuis, A. Goswami, M. Mhalla, and V. Savin, Polarization of quantum channels using Clifford-based channel combining, *IEEE Trans. Inf. Theory* **67**, 2857 (2021).
- [3] J. M. Renes, F. Dupuis, and R. Renner, Efficient polar coding of quantum information, *Phys. Rev. Lett.* **109**, 050504 (2012).

- [4] J. M. Renes and M. M. Wilde, Polar codes for private and quantum communication over arbitrary channels, *IEEE Trans. Inf. Theory* **60**, 3090 (2014).
- [5] M. M. Wilde and S. Guha, Polar codes for degradable quantum channels, *IEEE Trans. Inf. Theory* **59**, 4718 (2013).
- [6] H. Bombin and M. A. Martin-Delgado, Topological quantum distillation, *Phys. Rev. Lett.* **97**, 180501 (2006).

- [7] N. P. Breuckmann and J. N. Eberhardt, Quantum low-density parity-check codes, *PRX Quantum* **2**, 040101 (2021).
- [8] A. Y. Kitaev, Fault-tolerant quantum computation by anyons, *Ann. Phys. (NY)* **303**, 2 (2003).
- [9] A. Goswami, M. Mhalla, and V. Savin, Fault-tolerant preparation of quantum polar codes encoding one logical qubit, *Phys. Rev. A* **108**, 042605 (2023).
- [10] A. M. Steane, Active stabilization, quantum computation, and quantum state synthesis, *Phys. Rev. Lett.* **78**, 2252 (1997).
- [11] A. M. Steane, Fast fault-tolerant filtering of quantum code-words, [arXiv:quant-ph/0202036](https://arxiv.org/abs/quant-ph/0202036).
- [12] 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).
- [13] L. Postler, S. Heußen, I. Pogorelov, M. Rispler, T. Feldker, M. Meth, C. D. Marciniak, R. Stricker, M. Ringbauer, R. Blatt *et al.*, Demonstration of fault-tolerant universal quantum gate operations, *Nature (London)* **605**, 675 (2022).
- [14] C. Ryan-Anderson, J. G. Bohnet, K. Lee, D. Gresh, A. Hankin, J. P. Gaebler, D. Francois, A. Chernoguzov, D. Lucchetti, N. C. Brown, T. M. Gatterman, S. K. Halit, K. Gilmore, J. A. Gerber, B. Neyenhuis, D. Hayes, and R. P. Stutz, Realization of real-time fault-tolerant quantum error correction, *Phys. Rev. X* **11**, 041058 (2021).
- [15] H. Asai, S. Iizuka, T. Mogami, J. Hattori, K. Fukuda, T. Ikegami, K. Kato, H. Oka, and T. Mori, Device structure and fabrication process for silicon spin qubit realizing process-variation-robust swap gate operation, *Jpn. J. Appl. Phys.* **62**, SC1088 (2023).
- [16] A. Sigillito, M. Gullans, L. Edge, M. Borselli, and J. Petta, Coherent transfer of quantum information in a silicon double quantum dot using resonant swap gates, *npj Quantum Inf.* **5**, 110 (2019).
- [17] D. Bluvstein, S. J. Evered, A. A. Geim, S. H. Li, H. Zhou, T. Manovitz, S. Ebadi, M. Cain, M. Kalinowski, D. Hangleiter *et al.*, Logical quantum processor based on reconfigurable atom arrays, *Nature (London)* **626**, 58 (2023).
- [18] D. Bluvstein, H. Levine, G. Semeghini, T. T. Wang, S. Ebadi, M. Kalinowski, A. Keesling, N. Maskara, H. Pichler, M. Greiner *et al.*, A quantum processor based on coherent transport of entangled atom arrays, *Nature (London)* **604**, 451 (2022).
- [19] J. M. Boter, J. P. Dehollain, J. P. G. van Dijk, Y. Xu, T. Hensgens, R. Versluis, H. W. L. Naus, J. S. Clarke, M. Veldhorst, F. Sebastiano, and L. M. K. Vandersypen, Spiderweb array: A sparse spin-qubit array, *Phys. Rev. Appl.* **18**, 024053 (2022).
- [20] B. Jadot, P.-A. Mortemousque, E. Chanrion, V. Thiney, A. Ludwig, A. D. Wieck, M. Urdampilleta, C. Bäuerle, and T. Meunier, Distant spin entanglement via fast and coherent electron shuttling, *Nat. Nanotechnol.* **16**, 570 (2021).
- [21] I. Seidler, T. Struck, R. Xue, N. Focke, S. Trellenkamp, H. Bluhm, and L. R. Schreiber, Conveyor-mode single-electron shuttling in *si/sige* for a scalable quantum computing architecture, *npj Quantum Inf.* **8**, 100 (2022).
- [22] R. Chao and B. W. Reichardt, Quantum error correction with only two extra qubits, *Phys. Rev. Lett.* **121**, 050502 (2018).
- [23] E. Knill, Scalable quantum computing in the presence of large detected-error rates, *Phys. Rev. A* **71**, 042322 (2005).