IEEE Transactions on Quantum Engineering

Received 11 November 2024; revised 16 May 2025; accepted 16 May 2025; date of publication 23 May 2025; date of current version 23 June 2025.

Digital Object Identifier 10.1109/TQE.2025.3572764

Reducing Quantum Error Correction Overhead With Versatile Flag-Sharing Syndrome Extraction Circuits

PEI-HAO LIOU AND CHING-YI LAI (Senior Member, IEEE)

Institute of Communications Engineering, National Yang Ming Chiao Tung University, Hsinchu 30010, Taiwan

Corresponding author: Ching-Yi Lai (e-mail: cylai@nycu.edu.tw).

The work of Ching-Yi Lai was supported by the National Science and Technology Council of Taiwan under Grant 113-2221-E-A49-114-MY3, Grant 113-2119-M-A49-008, and Grant 114-2119-M-A49-006.

ABSTRACT Given that quantum error correction processes are unreliable, an efficient error syndrome extraction circuit should use fewer ancillary qubits, quantum gates, and measurements while maintaining low circuit depth, to minimize the circuit area, roughly defined as the product of circuit depth and the number of physical qubits. We propose to design parallel flagged syndrome extraction with shared flag qubits for quantum stabilizer codes. Versatile parallelization techniques are employed to minimize the required circuit area, thereby improving the error threshold and overall performance. Specifically, measurement outcomes across multiple rounds of syndrome extraction are integrated into a lookup table decoder, enabling parallelization of multiple stabilizer measurements with shared flag qubits. In addition, we introduce an adaptive technique to reduce the overhead from excessive syndrome extraction. We present flag-sharing and fully parallel schemes for the $[\![17, 1, 5]\!]$, $[\![19, 1, 5]\!]$ Calderbank–Shor–Steane (CSS) codes and the $[\![5, 1, 3]\!]$ non-CSS code, where the $[\![5, 1, 3]\!]$ implementation achieves the minimum known circuit area. Numerical simulations have demonstrated improved pseudothresholds for these codes by up to an order of magnitude compared to previous schemes in the literature.

INDEX TERMS Adaptive syndrome extraction, effective circuit area, fault-tolerant quantum computation (FTQC), flag-sharing syndrome extraction, quantum stabilizer codes, syndrome extraction (SE).

I. INTRODUCTION

Quantum computers have the potential to solve certain problems much more efficiently than classical computers by leveraging the properties of quantum entanglement and parallel computation. However, implementing large-scale quantum algorithms requires a significant number of qubits, the fundamental units of quantum computation. Qubits are highly susceptible to decoherence due to external environmental interference, which poses a significant challenge to reliable quantum computation [1], [2], [3], [4]. To achieve large-scale and reliable quantum computing, it is essential to find methods to protect qubits from the effects of decoherence or to develop more stable qubits. One promising approach to address this challenge is through fault-tolerant quantum computation (FTQC) [5], [6], [7], [8], [9], [10], [11], [12], [13].

FTQC uses quantum error correcting codes (QEC) [11], [14] to encode physical qubits into logical qubits, ensuring that errors can be corrected when they occur. When the error rate of physical gates is below a certain threshold, arbitrarily precise quantum computation can be achieved [5], [15], [16]. However, to perform active quantum error correction, a fault-tolerant (FT) syndrome extraction (SE) procedure is implemented, which can require a substantial number of ancilla qubits, two-qubit quantum gates, and single-qubit measurements, depending on the code under consideration [17]. In addition, because these quantum components are unreliable, repeated SE procedures might be necessary to obtain reliable recovery information, adding another layer of complexity to the process. The considerable resource demands of FTQC present a significant challenge for its implementation.

Many SE approaches have been presented, including the Shor, Steane, and Knill procedures [6], [11], [18], which require the preparation of specific ancillary states. For the Steane and Knill SE procedures, FT state preparations are typically performed in advance [19], [20]. For the Shor-style

SE, dealing with unreliable measurements often involves repeated measurements or using redundant syndrome measurements [21], [22]. Another approach is flagged SE [23], [24], [25], where a small number of ancilla qubits are designed to detect multiple error propagation events. This method utilizes adaptive lookup-table decoding to achieve fault-tolerant quantum error correction (FTQEC).

A Shor-style SE approach implements redundant syndrome measurements in a specific sequence [26]. Recently, adaptive SE has been shown to reduce the required measurements by using information across rounds for FT error correction [27]. Building on this, optimization tools now reduce SE space and time overhead, enhancing efficiency and integrating well with flagged SE methods [28]. In addition, a lookup table decoder has been introduced, allowing some stabilizers to be skipped if they commute with errors indicated by certain flagged qubits [29], [30].

It is crucial to reduce the resources required in flagged FTQC. Reichardt [31] designed an SE circuit for the $[\![7, 1, 3]\!]$ code that utilizes measurement qubits as flag qubits, enabling simultaneous measurement of multiple stabilizers and reducing circuit depth. Liou and Lai [32] introduced parallel SE circuits with shared flag qubits, which measure multiple stabilizers of the same type for Calderbank–Shor–Steane (CSS)-type stabilizer codes of distance three. Following this, Du et al. [33] developed parallel schemes using shared flag qubits for the $[\![17, 1, 5]\!]$ code [34] and the (6.6.6) triangular color code with parameters $[\![19, 1, 5]\!]$ [35], based on the decoding algorithm proposed in [25].

In this article, we propose an approach for deriving efficient SE circuits for general CSS codes by combining two established techniques: flag sharing and parallel stabilizer measurements. We investigate the combined performance of these techniques under detailed circuit-level noise models. This integrated approach reduces ancilla qubit usage through flag sharing, improves scheduling through generalized error syndrome analysis, and achieves lower circuit depth through various circuit constructions involving parallel measurements. To evaluate the practical efficiency of these circuits, we define the notion of effective circuit area, measured by the total number of two-qubit gates, ancilla preparations, measurements, and idle qubits in the circuit, scaled by their relative error rates. An SE circuit with a low effective circuit area is expected to offer improved fault tolerance and higher thresholds in realistic implementations.

Achieving good parallelism in SE to minimize the number of idle qubits can effectively reduce the effective circuit area. Traditional flag schemes utilize serial SE circuits to measure all the stabilizer generators [24], [25], [29]. We show that these multiple SE stages in the serial scheme can be parallelized by interleaving the controlled-NOT (CNOT) gates in their order on the data qubits, thus enabling depth reduction. Therefore, a serial flagged SE scheme for stabilizers of the same type implies the existence of a corresponding fully parallel flagged SE scheme. By carefully adjusting the order of the two-qubit gates, it is possible to obtain an SE scheme with small circuit depth.

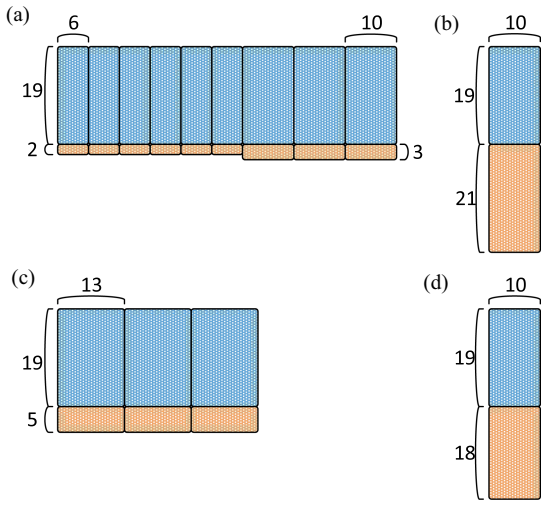

FIGURE 1. Versatile SE schemes for the $[\![19, 1, 5]\!]$ code are illustrated. (a) Serial $\mathbf{1}^T$ scheme. (b) Fully parallel $\mathbf{1}$ scheme. (c) Sequential [3 3 3] flag-sharing scheme. (d) Sequential [2; 2; 2; 1; 1; 1] flag-sharing scheme. The symbol $\mathbf{1}$ is understood as an all-ones column vector of appropriate dimension.

Using the technique of flag sharing can reduce the number of ancilla qubits and measurements at the cost of additional two-qubit gates and potentially more idle qubits. We provide design criteria for determining whether some stabilizer SE circuits can be merged using shared flags. Moreover, we show that a sequential flag-sharing SE scheme also implies the existence of a corresponding fully parallel version. When the measurement error rate dominates, it is preferable to reduce the number of measurements by using fewer flag qubits in SE.

Overall, versatile SE schemes for a given CSS code can be derived by grouping stabilizer measurements into blocks with shared flag qubits. These blocks may be executed either in parallel or sequentially. The details of these schemes will be introduced in Section III. We provide examples for the $[\![17, 1, 5]\!]$ and $[\![19, 1, 5]\!]$ codes, featuring low effective circuit areas. For instance, Fig. 1 illustrates four SE schemes for the $[\![19, 1, 5]\!]$ code, where each scheme is associated with a matrix. Since the $[\![19, 1, 5]\!]$ code has stabilizers of weight 6, a flagged SE using two flag qubits has a depth of 10. Both the fully parallel scheme in Fig. 1(b) and the flag-sharing scheme in Fig. 1(d) have been optimized to have a circuit depth of 10 for measuring the entire set of Z (X) stabilizers.

Extending the aforementioned results to non-CSS codes is more challenging since an SE stage of a stabilizer involves both X and Z components, leading to intricate error propagation events. However, we propose versatile sequential flag-sharing SE schemes for the $[\![5, 1, 3]\!]$ code, utilizing its cyclic structure [36].

In flagged FT QEC, error recovery is performed based on the error syndromes obtained from multiple rounds of SE. Typically, $O(d)$ (roughly between d and $2d$) rounds of

flagged or raw SE are sufficient for FT SE of a quantum code with distance d. However, by carefully analyzing the error correction decision tree [25], [37], it is possible to use fewer than d rounds of SE. In Section V-D, we illustrate that, for the cases considered, $d-1$ rounds are sufficient to perform FT SE. We use all measurement outcomes—of syndrome and flag qubits—to increase the number of distinguishable errors compared to a standard lookup-table-based approach. This effectively expands the syndrome space, allowing us to define a generalized error syndrome and construct a unified lookup table that incorporates both types of auxiliary qubit measurements. This approach has the potential to reduce the depth of SE schemes and simplify the lookup-table decoding process, which may assist in identifying circuits with smaller depths for the aforementioned codes.

We further propose an adaptive SE method tailored to reduce the overhead in the final stage of raw SE. The sequential flag measurement approach to handle high-weight error propagation events is initially introduced in [29], but it did not provide a generalized algorithmic framework. Then, a systematic algorithm for identifying stabilizer reduction opportunities during measurements was developed in [30]. Alternative methods, such as sequential Shor-style measurements [26], [27], dynamically adjust the number of required stabilizer measurements based on detected error locations. While Delfosse and Reichardt [26] focused on the theoretical framework, Tansuwannont et al. [27] supported it with experimental validation through simulation studies.

While previous approaches [26], [27], [29], [30] were applicable to sequential SE, where syndrome bits are generated sequentially, they cannot be directly applied to parallel SE, in which multiple syndrome bits are generated simultaneously. In particular, in our parallel SE schemes with shared flag qubits, the ordering of CNOT gates is crucial to enable parallel SE. However, it is not necessarily possible to further revise these schemes into adaptive schemes.

Building on insights from previous works and extending our SE schemes, we integrate adaptive SE techniques within a parallelized framework. Specifically, in the final round of raw SE, we can select a subset of stabilizers that yield syndromes capable of distinguishing errors that produced identical syndromes in earlier rounds, thus eliminating the need for stage-by-stage adaptation. This approach is particularly beneficial for FT designs with code distances greater than 3, which typically rely on multiround measurements. The key insight is that complete measurements are not required in every round, allowing for more efficient adaptive measurement strategies.

Adaptive schemes can improve performance by reducing the total number of measurements, especially in scenarios with low idle and measurement error rates.

Computer simulations on the various schemes are conducted under different error rate scenarios. Our results show that both our flag-sharing and fully parallel schemes outperform known schemes in the literature. The resources required for various schemes and their effective circuit areas are summarized in Table 2. Our simulations suggest that the performance of an SE scheme is closely related to its effective circuit area. While one might expect a fully parallel scheme to perform better due to minimized circuit depth and no additional two-qubit gates between different stabilizer measurements, our flag-sharing schemes demonstrate competitive or superior performances due to their lower effective circuit areas.

Using our schemes incorporating the aforementioned techniques, the $[\![17, 1, 5]\!]$ and $[\![19, 1, 5]\!]$ codes achieve pseudothresholds of 9×10^{-5} and 2×10^{-4}, respectively, under a standard error model where all components have the same error rates. These results outperform known schemes in the literature [25], [33]. In addition, our decision trees for distance-3 and distance-5 codes exhibit shorter depths compared to those presented in [25].

In particular, the pseudothresholds of the flag-sharing and fully parallel schemes for the $[\![5, 1, 3]\!]$ code are about 5×10^{-4}, assuming that all components have the same error rate. This makes the $[\![5, 1, 3]\!]$ code a competitive candidate for near-term experiments compared to the $[\![7, 1, 3]\!]$ or $[\![9, 1, 3]\!]$ codes, as the $[\![5, 1, 3]\!]$ code requires fewer qubits in total [38], [39], [40]. We also present simulations for various ratios of idle qubit fault to gate fault and measurement fault to gate fault, providing guidance on selecting the optimal scheme from the combinations of different techniques.

Recent experiments in quantum computing with atom arrays have focused on achieving parallel execution of entangling gates across multiple qubits. Evered et al. [41] demonstrated a neutral-atom-based quantum computer capable of high-fidelity entangling operations on multiple qubit pairs simultaneously. Bluvstein et al. [42] developed a logical quantum processor using reconfigurable atom arrays, enabling concurrent manipulation of multiple atoms with parallel control laser beams.

Google Quantum AI's studies on superconducting quantum systems [43], [44] have marked significant progress in FT computing, achieving logical error suppression through surface code implementations. These studies also identified critical error sources, including crosstalk and leakage, as key factors in enhancing scalability and system robustness. In the context of ion-trap platforms, Ryan-Anderson et al. [45] presented a real-time FTQEC scheme using a trapped-ion quantum processor. Employing the $[\![7, 1, 3]\!]$ color code, they encoded a single logical qubit into ten physical qubits and demonstrated repeated QEC cycles with real-time decoding and correction. The system maintained logical coherence through adaptive SE and Pauli frame tracking, establishing key operational benchmarks for scalable FTQC. Hilder et al. [46] also showcased FT parity readout on a shuttling-based trapped-ion quantum computer, highlighting scalable error correction protocols and robust logical qubit performance. Heußen et al. [37] investigated strategies for designing FT circuits under realistic noise models in static ion chains, providing insights into the feasibility and performance

tradeoffs of circuit-level FT schemes in practical implementations.

By leveraging parallel processing capabilities, our parallel SE technique aligns well with these recent advancements, offering complementary benefits for near-term quantum technologies.

The rest of this article is organized as follows. In Section II, we introduce the notion of FTQC and flagged SE. Moving on to Section III, we propose versatile SE schemes for CSS codes, using the $[\![17, 1, 5]\!]$ code as an illustration. In Section IV, we propose versatile SE schemes for the $[\![5, 1, 3]\!]$ non-CSS code. The QEC procedures for codes with distance 3 or 5 are discussed in Section V. Simulation results on the various SE schemes and their analysis are presented in Section VI. Finally, Section VII concludes this article.

II. STABILIZER CODES AND FT FLAGGED SE

A. STABILIZER CODES

We consider quantum computation with qubits using the computational basis $\{|0\rangle, |1\rangle\}$. The Pauli matrices $I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, $X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$, $Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$, and $Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$ form a basis for the linear operators on a single-qubit state space. Similarly, the collection of the n-fold tensor product of the Pauli matrices forms a basis for the linear operators on the n-qubit state space. For simplicity, let X_i denote the operator consisting of a Pauli matrix X on the ith qubit and identities on the other qubits. Z_i and Y_i are similarly defined. In addition, we may omit the tensor product symbol. Thus, an n-fold Pauli operator can be represented by its nonidentity components. For instance, $X \otimes Y \otimes Z \otimes I \otimes I$ will be represented as $X_1Y_2Z_3$. Moreover, we may use the notation X_T, where T is a set of positions, to indicate that an X is applied to each qubit in T. For example, $X_{1,2,3}$ represents $X_1X_2X_3$. The weight of an n-fold Pauli operator indicates the number of nonidentity elements it contains.

We consider quantum errors that are Pauli errors. Let $\mathcal{S}$ be an abelian subgroup of the n-fold Pauli operators with appropriate phases. Suppose that $\mathcal{S}$ has $n-k$ independent generators $g_1, \ldots, g_{n-k}$. Then, it defines a $[\![5, 1, 3]\!]$ stabilizer code $C(\mathcal{S})$, which is a 2^k-dimensional subspace of the n-qubit state space that is fixed by $\mathcal{S}$. Thus, $\mathcal{S}$ is called a stabilizer group and the operators in $\mathcal{S}$ are called stabilizers. Any nonidentity n-fold Pauli operators of weight less than d must either be in the stabilizer group or anticommute with some stabilizers [9]. Two Pauli operators either commute or anticommute with each other. The error syndrome of a Pauli error E with respect to the $n-k$ stabilizer generators $g_i \in \mathcal{S}$ is a binary string of length $n-k$, whose ith bit is 1 if E anticommutes with g_i, and 0, otherwise. The error syndrome of an error describes its commutation relations with the stabilizer generators. A code of distance d is capable of correcting $\lfloor \frac{d-1}{2} \rfloor$ errors.

Since we are working with binary error syndromes, it is convenient to define the binary measurement outcome $b \in \{0, 1\}$ for single-qubit X- or Z-basis measurements. Here, the measurement outcome b corresponds to the observed eigenvalue $(-1)^b$ of the X- or Z-operator.

A Pauli operator consisting entirely of X or Z matrices is said to be of X- or Z-type, respectively. A CSS code [47], [48] is a stabilizer code with some stabilizer generators of X-type and others of Z-type.

B. FT QUANTUM COMPUTATION

A quantum circuit comprises fundamental single-qubit and two-qubit quantum gates, along with state preparations and measurements. We will focus on qubit measurements in the X or Z basis, as well as the controlled-phase (CZ) gate and CNOT gate.

Errors in a quantum circuit can arise from imperfect quantum gates, idling qubit locations, noisy qubit measurements, and imperfect ancillary state preparation. Each of these sources will be referred to as a location. Let $\mathrm{CNOT}_{1,2} = |0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes X$, where the first and second qubits are called the control and target qubits, respectively. Similarly, let $\mathrm{CZ}_{1,2} = |00\rangle\langle 00| + |01\rangle\langle 01| + |10\rangle\langle 10| - |11\rangle\langle 11|$. Then, we have $\mathrm{CNOT}_{1,2}X_1 = X_1X_2\mathrm{CNOT}_{1,2}$, $\mathrm{CNOT}_{1,2}Z_2 = Z_1Z_2\mathrm{CNOT}_{1,2}$, $\mathrm{CZ}_{1,2}X_1 = X_1Z_2\mathrm{CZ}_{1,2}$, and $\mathrm{CZ}_{1,2}X_2 = Z_1X_2\mathrm{CZ}_{1,2}$. As a result, a single-qubit Pauli error occurring in one of the qubits involved in a CNOT or CZ gate may propagate to the other qubit.

In our theoretical analysis, we assume that each quantum gate, qubit state preparation, and qubit measurement take the same unit of time, though this may not hold true in physical implementations. A location fault denotes an event that a Pauli error occurs after a perfect single-qubit gate, two-qubit gate, ancilla preparation, an idle qubit location, or before a perfect qubit measurement in a circuit. The gate and error models considered in this article are defined as follows.

1) *Gate faults:* Independent depolarizing errors with rate p are introduced after perfect quantum gates in the circuit.
   a) Following each CNOT or CZ gate, one of the 15 nonidentity two-qubit Pauli operators is applied with probability $p/15$.
   b) Ancillary state preparations ($|0\rangle$, $|1\rangle$) are corrupted by X errors with probability $2p/3$, and ($|+\rangle$, $|-\rangle$) are corrupted by Z errors with probability $2p/3$.
2) *Idle qubit location faults:* Idle qubits are subject to depolarizing errors with rate γp, where $0 \leq \gamma \leq 1$.
3) *Measurement faults:* Independent depolarizing errors with rate βp are introduced before measurements, where β could be larger than 1.

We consider an FTQC scheme built upon a quantum code with distance d, aiming to correct up to $t = \lfloor \frac{d-1}{2} \rfloor$ location faults in a round of error correction. This approach achieves a logical error rate of $O(p^{t+1})$. While $s \leq t$ location faults occur with probability $O(p^s)$, these faults can—in a non-FT circuit—lead to uncorrectable Pauli errors with weight

greater than t. Specifically, we focus on the SE process in an FTQC scheme, which involves the interaction of data qubits of the quantum codeword with additional ancillary qubits for syndrome measurements, using CNOT and CZ gates.

If w location faults occur in a circuit, while the other locations remain noiseless, this is referred to as a *w-fault event*. If the evolution of a one-fault event results in a Pauli error of weight higher than one on the data qubits, this event is considered a high-weight error propagation event.

Definition 1 (FT Error Correction [15], [49]): An error correction protocol is *t-FT* if the following two conditions are both satisfied.

1) For any input codeword with a Pauli error of weight e_1, if e_2 faults occur during the protocol with $e_1 + e_2 \leq t$, then the output state is correctable using perfect decoding.
2) If e location faults occur during the protocol with $e \leq t$, regardless of the error in the input state, the output state will differ from any valid codeword by a Pauli error of at most weight e.

For nonadaptive SE procedures, a round of SE refers to a sequence of stages in a circuit designed to measure a complete set of stabilizer generators, where each stage extracts the error syndrome bits of one or more stabilizers. Since imperfect gates are used for SE, the syndrome measurement outcomes are not always reliable, and multiple rounds of SE are typically required to ensure that reliable error information can be extracted with high probability.

For a stabilizer code with minimum distance d, a total of $L = O(d)$ rounds of SE are sufficient to achieve FT SE [50], [51], [52]. However, unnecessary stabilizer measurements in a round of SE can be eliminated, leading to adaptive SE procedures. The collection of all measurement outcomes from more than one rounds of SE is referred to as a generalized error syndrome.

To perform error correction based on a given generalized error syndrome, one would like to find the most probable error event matching the syndrome. One approach is to construct a lookup table consisting of pairs of error syndromes and corresponding error operators for error recovery on the data qubits.

Definition 2: Two fault events are said to be degenerate if they generate the same residual Pauli errors on the data qubits up to a stabilizer.

Consequently, there is no need to distinguish between degenerate error events.

If two fault events are not degenerate, they should be treated differently. Furthermore, if two nondegenerate fault events produce the same generalized error syndrome, the resulting errors become indistinguishable, rendering error correction infeasible. Hence, for a set of nondegenerate fault events to be correctable, they must be distinguishable. This leads to the following proposition.

Proposition 1: An SE procedure for a code with minimum distance d is FT if any w-fault event, with $1 \leq w \leq \lfloor \frac{d-1}{2} \rfloor$, has a unique generalized error syndrome up to degeneracy.

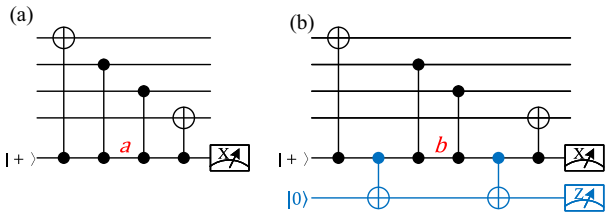

FIGURE 2. SE circuits for measuring a stabilizer $X_1Z_2Z_3X_4$. (a) Raw SE circuit. (b) Flagged syndrome extract circuit. The ancilla qubit initialized in $|0\rangle$ is a flag qubit. The blue part includes a pair of CNOT gates used to detect whether a high-weight error propagation event has occurred.

This implies that for any two correctable error patterns, their respective syndromes are distinct and distinguishable. If two error patterns were to produce the same syndrome, the errors would become indistinguishable, making error correction infeasible.

Since single-qubit and two-qubit gates dominate a quantum circuit, we adopt the following definition of error threshold with respect to the gate error rate.

Definition 3: The *error threshold* of a procedure is the value at which, if the physical gate error rate is below this threshold, the logical error rate of the procedure will be lower than the physical gate error rate.

To implement an FT quantum memory, the threshold of an FT error correction procedure is estimated. When implementing an FT logical CNOT gate across two logical codewords for a CSS code, this typically involves transversal CNOT gates along with four FT error correction procedures: two for the two codewords before the CNOT gates and two for the two codewords after the CNOT gates. As a result, the error threshold for implementing an FT logical CNOT gate can be estimated to be four times higher than that for implementing an FT quantum memory. For simplicity, we focus on simulating the estimation of the threshold for FT quantum memory.

C. FT SE WITH FLAG QUBITS

An SE circuit is used to determine the error syndrome of an occurred error. Fig. 2(a) illustrates an SE circuit for measuring the Pauli operator $X_1Z_2Z_3X_4$ on four data qubits, using an ancillary qubit in $|+\rangle$. This ancillary qubit is called a measurement qubit, and the measurement outcome corresponds to the syndrome bit associated with $X_1Z_2Z_3X_4$. This circuit accurately extracts the syndrome bit, assuming no gate or measurement errors. However, when a Pauli X error occurs at location a in Fig. 2(a), it propagates through a CZ and a CNOT gates, leaving a weight-2 error Z_3X_4 on the data qubits. This constitutes a high-weight error propagation event that may require additional attention.

On the contrary, the circuit in Fig. 2(b) introduces an ancillary qubit initialized to $|0\rangle$, referred to as a flag qubit (highlighted in blue), and includes an additional pair of CNOT

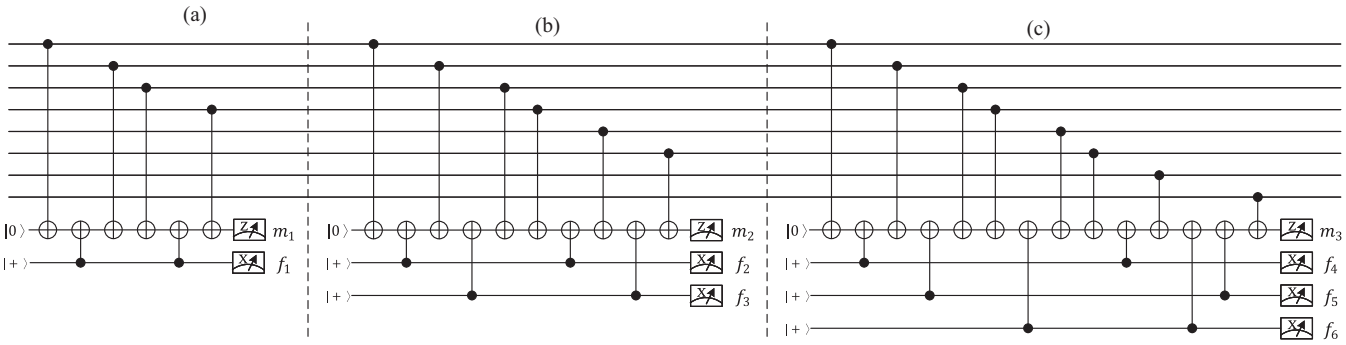

FIGURE 3. (a) Weight-4, (b) weight-6, and (c) weight-8 SE circuits using one, two, and three flag qubits, respectively. The measurement outcomes on the measurement qubits and flag qubits are denoted by m_i and f_j, respectively.

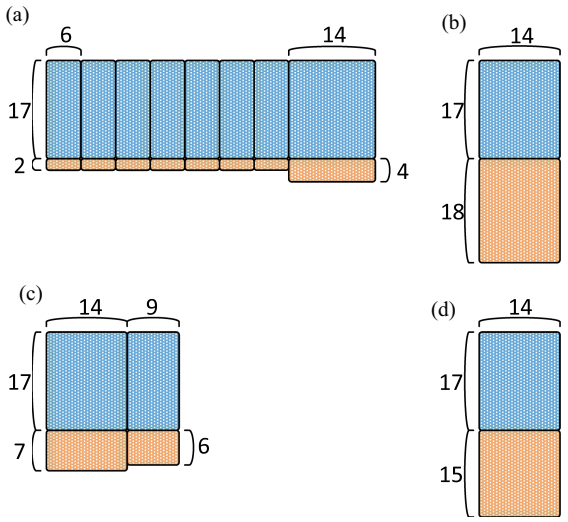

FIGURE 4. Various SE schemes for the $[\![17, 1, 5]\!]$ code are illustrated. (a) Serial [1 1 1 1 1 1 1 1] SE scheme. (b) Fully parallel [1; 1; 1; 1; 1; 1; 1; 1] scheme. (c) Sequential [4 4] flag-sharing scheme. (d) Sequential [2; 2; 2; 1; 1] flag-sharing scheme. A pair of a blue box and an orange box denote a stage of SE. The height of the blue box represents the number of data qubits of the code, and the height of the orange region indicates the number of ancilla qubits required. The width of the blue box indicates the circuit depth. Measurements and state preparations are not shown in these illustrations. Note that raw SE is not included in this figure.

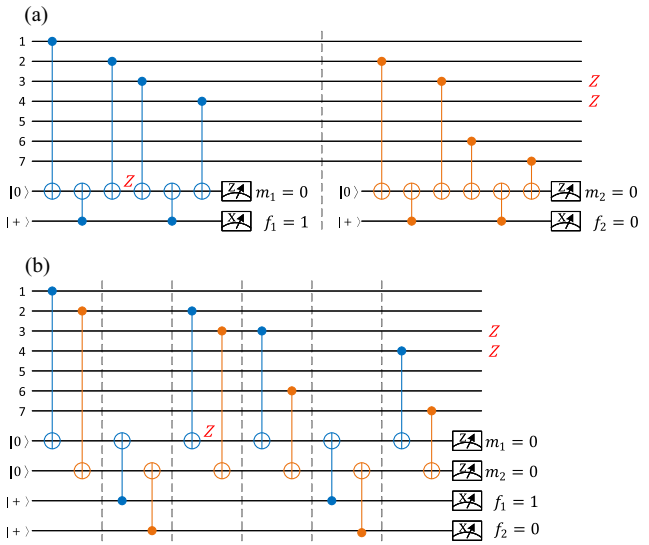

FIGURE 5. SE circuits for two weight-4 Z-type stabilizers. (a) Serial scheme. (b) Parallel scheme.

gates. When a Pauli X error occurs at location b, the measurement outcome on the flag qubit will be one, signaling a high-weight error propagation event [23], [24]. In this scenario, the occurrence of a high-weight error propagation event triggers the flag qubit, causing the flag to be raised. The circuit in Fig. 2(b) is termed a flagged SE circuit, while the circuit in Fig. 2(a) is termed an raw SE circuit.

For stabilizers of higher weight, additional flag qubits might be required for signaling high-weight error propagation events [25], [53]. A traditional flagged SE circuit [23], [24], [25] can be constructed by connecting a CNOT gate from the measurement qubit to a flag qubit after each CNOT gate from the data qubit to the measurement qubit. Fig. 3 illustrates the SE circuits for Z-type stabilizers of weights 4, 6, and 8, respectively, using one, two, and three flag qubits [25]. An SE circuit for a weight-w stabilizer will be referred to as a weight-w SE circuit.

In general, the number of flag qubits required in a flagged SE circuit depends on both the stabilizer weight-w and the code distance-d. For example, in the case of $d = 3$, $O(\log w)$ flag qubits are sufficient [54]. For $d = 5$, up to $O(w)$ flag qubits may be required [25]. More generally, to detect high-weight error propagation events in circuits designed for larger code distances, the number of flag qubits may scale as $O(f(d, w))$, where $f(d, w)$ reflects a joint dependence on both the stabilizer weight and the code distance [53].

In traditional flagged SE schemes, multiple rounds of SE are performed, and a decision tree is constructed with a lookup table at each leaf (cf. Figs. 11 and 12). Routing on the decision tree depends on the outcomes of the measurement qubits and flag qubits. A lookup table is a precomputed one-to-one function that outputs a correction operator on input an error syndrome from a complete (raw) SE circuit. Multiple lookup tables of potential errors can be constructed based

on the flag outcomes. However, the outcomes on the measurement qubits are often disregarded when some flag qubits rises in the same round of SE. Thus, we have the following proposition.

Proposition 2: In flagged SE, the outcomes of the measurement qubits and flag qubits in L rounds of flagged or raw SE can be collected to infer the most likely residual error on the data qubits.

Using additional error information from the measurement qubits helps design an SE circuit with a unique generalized error syndrome for each target error event. This technique is critical for our construction of flag-sharing schemes with shorter circuit depths later.

We defer the discussion of Proposition 2 to Section V-B, where we illustrate its utility using the example of the $[\![5, 1, 3]\!]$ code for clarity and simplicity.

In flagged SE, typically between d and $2d$ rounds of flagged or raw SE are sufficient for FT SE. By carefully examining the error correction decision tree, it is possible to use fewer than d rounds of SE on average, as we will show in Section V.

III. VERSATILE SE SCHEMES FOR CSS CODES

For a given independent set of stabilizer generators, the seminal flagged FTQC scheme adopts a serial implementation of SE, as illustrated in Fig. 3 [23], [24], which reduces the number of ancillary qubits required for FT SE. It is possible to extract multiple syndromes simultaneously using shared flag qubits [31], [32], [33], thus reducing the depth of SE.

To design a practically useful FT SE scheme, it is important to minimize sources of errors and keep these errors under control to achieve a good error threshold. We define the notion of circuit area as follows:

Definition 4: The effective circuit area of an SE circuit for a quantum code is defined as

$$1.6n_{\mathrm{g}} + n_{\mathrm{p}} + \beta n_{\mathrm{m}} + \gamma n_{\mathrm{i}}$$

where n_{g}, n_{p}, n_{m}, and n_{i} are the numbers of two-qubit gates, ancilla preparations, measurements, and idle qubits in $\mathcal{C}$, respectively.

Since there are 15 possible nonidentity CNOT faults, six of which result in a weight-1 error and nine of which result in a weight-2 error, we set the coefficient of n_g to $1 \times \frac{6}{15} + 2 \times \frac{9}{15} = 1.6$.

All other single-qubit gates produce only weight-1 errors, so their coefficients are set to 1.

Intuitively, an SE circuit with a low effective circuit area might have a better error threshold, as reducing the effective circuit area decreases the potential sources of error. When the idle qubit error rate is high, the error threshold degrades significantly. Therefore, achieving good parallelism in SE to minimize the number of idle qubit locations might be desirable. Moreover, when the measurement error rate dominates, it might be preferable to reduce the number of measurements by using fewer flag qubits in SE.

In this section, we propose versatile SE circuits to reduce the effective circuit area. These circuits are categorized into three types: 1) serial SE; 2) fully parallel SE; and 3) sequential flag-sharing SE.

Roughly speaking, an SE scheme can be associated with an integer matrix M, whose (i, j)th entry $M_{i,j}$ denotes that $M_{i,j}$ stabilizers are simultaneously measured using the ith set of flag qubits at stage j. The sum $s = \sum_{i,j} M_{i,j}$ equals the total number of stabilizers to be measured. By considering $M_{i,j}$ as an integer partition of s, versatile SE circuits can be designed. (For example, Fig. 5(a) is denoted as a [1 1] scheme, while Fig. 5(b) is a [2] scheme.)

To illustrate the concepts, we first use the $[\![17, 1, 5]\!]$ code as an example, which is defined with stabilizers

$$\begin{aligned}
g_1 &= Z_1Z_2Z_3Z_4 & g_2 &= Z_1Z_3Z_5Z_6\\
g_3 &= Z_5Z_6Z_9Z_{10} & g_4 &= Z_7Z_8Z_{11}Z_{12}\\
g_5 &= Z_9Z_{10}Z_{13}Z_{14} & g_6 &= Z_{11}Z_{12}Z_{15}Z_{16}\\
g_7 &= Z_8Z_{12}Z_{16}Z_{17} & g_8 &= Z_3Z_4Z_6Z_7Z_{10}Z_{11}Z_{14}Z_{15}\\
g_9 &= X_1X_2X_3X_4 & g_{10} &= X_1X_3X_5X_6\\
g_{11} &= X_5X_6X_9X_{10} & g_{12} &= X_7X_8X_{11}X_{12}\\
g_{13} &= X_9X_{10}X_{13}X_{14} & g_{14} &= X_{11}X_{12}X_{15}X_{16}\\
g_{15} &= X_8X_{12}X_{16}X_{17} & g_{16} &= X_3X_4X_6X_7X_{10}X_{11}X_{14}X_{15}
\end{aligned} \tag{1}$$

and logical operators

$$\bar{X} = X^{\otimes 17} \qquad \bar{Z} = Z^{\otimes 17}. \tag{2}$$

Since $g_1, \ldots, g_8$ are of Z-type and $g_9, \ldots, g_{16}$ are of X-type and they are equivalent up to a Hadamard transform, we focus on the design of SE for Z-type stabilizers.

Four SE schemes for the $[\![17, 1, 5]\!]$ code are presented in Fig. 4, excluding raw SE. We will explore these approaches in the following sections.

A. SERIAL SE

A traditional flag scheme utilizes a serial SE circuit to measure all the stabilizer generators [24], [25], [29], which is referred to as a $\mathbf{1}^T$ scheme. This approach helps reduce the number of ancillary qubits and measurements.

In the following calculation, we assume that for codes with $d = 5$, $\frac{1}{2}w$ flag qubits are required for each stabilizer of weight w, as suggested in [54]. For an n-qubit code with s stabilizer generators of mean weight w, a stage of SE circuit is of depth $w + 2(\frac{1}{2}w) = 2w$. Consequently, the $\mathbf{1}^T$ scheme involves roughly $2ws$ CNOT gates, $(s + \frac{1}{2}ws)$ state preparations, $(s + \frac{1}{2}ws)$ measurements, and $\frac{16}{5}sw(n + \frac{w}{2} + 1) - \frac{16}{5}sw$ idle qubits.

Thus, the effective circuit area is approximately

$$s + \frac{37}{10}sw + \beta\left(s + \frac{1}{2}sw\right)$$

$$-\gamma\left(\frac{16}{5}sw-\frac{16}{5}sw\left(n+\frac{1}{2}w+1\right)\right). \tag{3}$$

Further details are provided in Appendix A.

For the $[\![17, 1, 5]\!]$ code, seven of the Z-type stabilizers have weight 4, while the remaining one has weight 8. Thus, we have a [1 1 1 1 1 1 1 1] scheme, as shown in Fig. 4(a), where the first seven stages follow from Fig. 3(a) and the last stage is from Fig. 3(c).

This scheme introduces many idle qubits, making it suitable for scenarios with a low idle qubit error rate.

B. FULLY PARALLEL SE

In order to reduce the number of idle qubits, we propose a fully parallel SE scheme, denoted as the **1** scheme, which transforms the sequential $\mathbf{1}^T$ SE scheme into a parallel implementation with all of its flag qubits, without sharing any flag qubits.

Two stages of SE circuits in a $\mathbf{1}^T$ scheme can be parallelized by interleaving their gates, enabling simultaneous execution of operations as shown in Fig. 5.

Lemma 1: Suppose that a quantum code allows for FT SE of r Z-type (or X-type) stabilizers using a serial flagged SE scheme with L rounds of SE. Then, the r stages of SE in the serial scheme can be parallelized by interleaving the CNOTs in their order on the data qubits.

Proof: Consider a code of distance d and $t=\lfloor\frac{d-1}{2}\rfloor$.

Let us consider each 1-fault event in the **1** scheme, which has a corresponding 1-fault event in the original $\mathbf{1}^T$ scheme. Such an example is shown in Fig. 5(a) and (b).

An important observation is that in the flagged SE circuit in Fig. 3, a state preparation or measurement fault will not propagate. Only the one-fault events leaving a Z error on the measurement qubit will propagate back to the data qubits, triggering only the following X measurement qubits but no flag qubits. Now, if we interleave the SE circuits, as shown in Fig. 5(b), we obtain a similar result where a state preparation or measurement fault will not propagate. Thus, a state preparation or measurement fault in the **1** scheme will have the same generalized error syndrome as its corresponding one-fault event in the original $\mathbf{1}^T$ scheme.

Now, consider a one-fault high-weight error propagation event in the **1** scheme, which triggers some flag qubits and propagates some Z errors into the data qubits. Again, these Z errors on the data qubits will trigger only the following X measurement qubits but no flag qubits. It can be observed that a corresponding one-fault high-weight error propagation event in the $\mathbf{1}^T$ scheme will trigger corresponding flag qubits and propagate the same Z errors to the data qubits. Thus, these two one-fault high-weight error propagation events will have the same generalized error syndrome. Such an example is shown in Fig. 5.

By the assumption that the $\mathbf{1}^T$ scheme is FT, any t-fault event has a unique generalized error syndrome. Therefore, the **1** scheme is FT. □

By interleaving the CNOTs in their order on the data qubits, we can ensure the proper functionality of the SE scheme. Note that we do not claim that a fully parallel scheme has the same depth as a single stabilizer measurement.

More generally, we have the following corollary.

Corollary 1: Given an FT $\mathbf{1}^T$ SE scheme for s Z-type (or X-type) stabilizers, one can derive an M SE scheme for any binary matrix M such that $\sum_{i,j}M_{i,j}=s$. In particular, an FT $\mathbf{1}^T$ SE scheme can be transformed into a **1** SE scheme while maintaining fault tolerance.

It is straightforward to see that a **1** scheme would have the lowest circuit depth and possibly the lowest effective circuit area among possible binary matrices M with $\sum_{i,j}M_{i,j}=s$. We will focus on this variation in the following discussion.

Remark 1: The circuit depth of the **1** scheme is not necessarily the depth of its longest stage. This is because the CNOT order for each data qubit is preserved as in the original $\mathbf{1}^T$ scheme in the described procedure. One may reorder the CNOTs to reduce the circuit depth, but it is essential to ensure that any n-fault event has its unique generalized error syndrome. Consequently, it is important to include the outcomes of the measurement qubits as a part of the generalized error syndrome by Proposition 2. (See Section V-B for a detailed discussion on the $[\![5, 1, 3]\!]$ code.)

Consider r stabilizers of mean weight w, with the highest weight being $2w$. Assuming that the circuit depth is $3w$, the effective circuit area is about $(r+\frac{24}{5}w+\frac{1}{2}rw)+\beta(r+\frac{1}{2}rw)-\gamma(\frac{24}{5}w-(\frac{24}{5}w(n+r+\frac{1}{2}rw)))$.

If $r=s/2$, the effective circuit area of two **1** SE schemes for Z- and X-type stabilizers is about

$$\left(s+\frac{48}{5}w+\frac{1}{2}sw\right)+\beta\left(s+\frac{1}{2}sw\right)-\gamma\left(\frac{48}{5}w-\frac{48}{5}w\left(n+\frac{1}{2}s+\frac{1}{4}sw\right)\right). \tag{4}$$

A detailed derivation of (4) is provided in Appendix B. For $\gamma=0$ and $\beta=1$, (3) and (4) are similar. At $\gamma=\beta=1$, the dominant terms in (3) and (4) are $\frac{16}{5}sw(n+\frac{1}{2}w+1)$ and $\frac{48}{5}w(n+\frac{s}{2}+\frac{sw}{4})$, respectively. The second term is smaller than the first when $s\geq 4$ and $w\leq n-2$.

Thus, when sufficient ancillary qubits are available, the **1** scheme may have a smaller effective circuit area than the $\mathbf{1}^T$ scheme if at least four stabilizers are measured simultaneously. This observation is further supported in the simulation section.

For the $[\![17, 1, 5]\!]$ code, we adopt the $\mathbf{1}^T$ SE scheme for the Z-type stabilizers in (1) to construct a **1** SE scheme, as shown in Fig. 4(b). By Lemma 1, we can directly parallelize the $\mathbf{1}^T$ SE scheme to generate the **1** SE scheme. The detailed circuit of Fig. 4(b) is depicted in Fig. 21 in Appendix C, where the circuit depth is 14, which is the same as the length of the longest stage in the $\mathbf{1}^T$ scheme. The flagged SE circuit for the weight-8 Z-type stabilizer dominates the circuit depth, which is 14 according to Fig. 3. There are ten flag qubits in total, of which seven are for the seven weight-4 stabilizers.

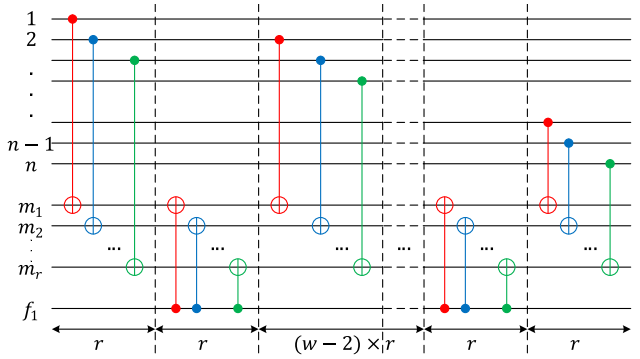

FIGURE 6. Parallel flagged SE circuit with r stabilizers sharing one flag qubit. This circuit extracts the syndrome g_1 (red), g_2 (blue), . . ., and g_r (green). It is capable of detecting weight-1 error events through a pair of CNOT operations between measurement qubits and the flag qubit. The solid black lines below the diagram indicate the total number of CNOT gates used within this interval. Ancillary state preparations and measurements have been omitted for clarity.

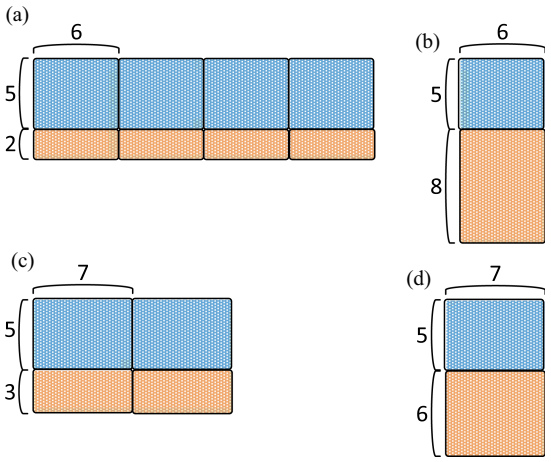

FIGURE 7. Various SE schemes for the ⟦5, 1, 3⟧ code are illustrated. (a) Serial [1 1 1 1] SE scheme. (b) Fully parallel [1; 1; 1; 1] scheme. (c) [2 2] flag-sharing scheme. (d) $\begin{bmatrix} 2 \\ 2 \end{bmatrix}$ flag-sharing scheme.

C. SEQUENTIAL FLAG-SHARING SE

This section will introduce how parallelism in SE can be achieved using versatile flag-sharing techniques, generalizing the ideas in [31], [32], and [33], to reduce the circuit depth while conserving the number of flag qubits.

Similar to the discussion in Lemma 1, we can merge multiple stages of SE circuits in a serial scheme by flag sharing to achieve parallelism as well. A parallel flag-sharing SE circuit using one flag qubit for distance-3 codes is shown in Fig. 6. However, since fewer flag qubits are used, these stages can be merged only if any t-fault event has its unique generalized error syndrome up to degeneracy. Thus, the number of stages that can be merged is restricted in a flag-sharing scheme.

Consider an $[\![n, k, d]\!]$ code with s Z-type stabilizers. Suppose that there is a serial SE scheme of s stages for the Z-type stabilizers $g_1, \ldots, g_s$. It is possible to design a sequential $[r_1\, r_2\, \cdots\, r_\ell]$ SE scheme for s stabilizers of Z-type, where r_i stabilizers are measured using a set of flag qubits at stage i and $\sum_i r_i = s$. By ensuring that any t-fault event in an SE procedure has its unique generalized error syndrome, we can achieve a t-FT SE procedure. Following a similar idea, a [4 4] SE scheme for the ⟦17, 1, 5⟧ was proposed in [33], as shown in Fig. 4(c).

Similar to the proof of Lemma 1, we have the following lemma that transforms a general sequential flag-sharing SE scheme into a parallel flag-sharing SE scheme.

Lemma 2: Suppose that a quantum code allows for FT SE of r Z-type (or X-type) stabilizers using a serial $[r_1\, r_2\, \cdots\, r_\ell]$ SE scheme with L rounds of SE. Then, this $[r_1\, r_2\, \cdots\, r_v]$ SE scheme can be transformed into an $[r_1; r_2; \cdots; r_\ell]$ SE scheme while maintaining fault tolerance.

Remark 2: Again, the circuit depth of the $[r_1; r_2; \cdots; r_\ell]$ scheme is not necessarily the depth of its longest stage due to the constrained CNOT order on each data qubit in the described procedure, as mentioned in Remark 1. One may try to reorder the CNOTs to reduce the circuit depth while ensuring that any n-fault event has its unique generalized error syndrome.

According to Lemma 2, we first construct a [2 2 2 1 1] SE scheme for the Z-type stabilizers of the ⟦17, 1, 5⟧ code and then transform it into a [2; 2; 2; 1; 1] SE scheme by Proposition 2, as shown in Fig. 4(d). This scheme uses a total of 15 ancilla qubits, and the circuit depth is 14. Moreover, it has the smallest effective area among the SE schemes in Fig. 4. The details are provided in Fig. 22 in Appendix C.

IV. VERSATILE SE SCHEMES FOR NON-CSS CODES

In the previous section, we studied various SE schemes for CSS codes. Extending those results to non-CSS codes is more challenging since an SE stage of a stabilizer may involve the interleaving of multiple CNOT and CZ gates, leading to intricate error propagation events. For instance, an error on the measurement qubit might propagate to a data qubit through a CZ (or CNOT) gate and then propagate back to another measurement qubit through a CNOT (or CZ) gate.

A. SE SCHEMES FOR THE ⟦5, 1, 3⟧ CODE

Nonetheless, we propose versatile SE schemes for the ⟦5, 1, 3⟧ code, as shown in Fig. 7. We consider the following five stabilizers:

$$\begin{aligned} g_1 &= X_1Z_2Z_3X_4I_5 \\ g_2 &= X_2Z_3Z_4X_5I_1 \\ g_3 &= X_3Z_4Z_5X_1I_2 \\ g_4 &= X_4Z_5Z_1X_2I_3 \\ g_5 &= X_5Z_1Z_2X_3I_4. \end{aligned} \tag{5}$$

Note that these stabilizer generators are written in the form $XZZXI$, with their labeling cyclically shifted. Measuring any four of them one by one, as shown in Fig. 7(a), constitutes a

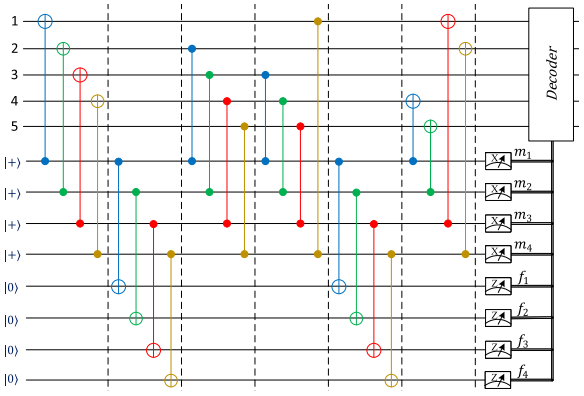

FIGURE 8. Fully parallel [1; 1; 1; 1] SE scheme for the ⟦5, 1, 3⟧ code, which extracts the syndrome bits of g_1, g_2, g_3, and g_4. Each color marks an SE circuit of a stabilizer.

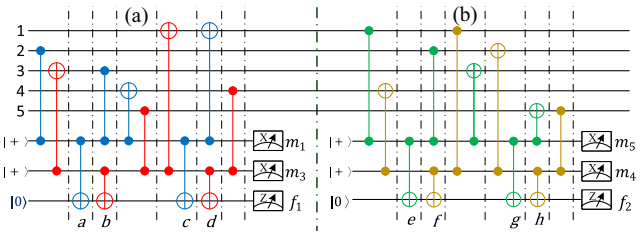

FIGURE 9. [2 2] flag-sharing SE scheme for the ⟦5, 1, 3⟧ code, which extracts the syndrome bits of g_1 (blue) and g_3 (red) in part (a), and g_5 (green) and g_4 (yellow) in the part (b). CNOTs labeled from *a* to *h* are paired to couple a measurement qubit and the flag qubit, while sharing the same flag qubit.

[1 1 1 1] SE scheme. Each flag SE circuit stage is of depth 6 and uses two ancilla qubits.

Similar to Corollary 1, we can construct a fully parallel [1; 1; 1; 1] scheme for the ⟦5, 1, 3⟧ code, as shown in Fig. 7(b), and the details shown in Fig. 8. This [1; 1; 1; 1] scheme has the smallest circuit depth of 6, but uses a total eight ancilla qubits, of which four are flag qubits. A key observation is that the ⟦5, 1, 3⟧ code is a cyclic code with stabilizer generators given in (5). When these five stabilizers are arranged as a 5×5 matrix, it becomes apparent that each column contains the same type of entry—X, Z, or I—but with distinct qubit numbers. As a result, we can organize all CNOTs or all CZs at each step of the circuit, as demonstrated in Fig. 8. In addition, a high-weight error propagation event will not propagate further. Specifically, the preexisting high-weight error propagation events from the nonparallelized circuit do not generate additional high-weight error propagation events after parallelization.

Building upon our previously proposed flag-sharing scheme [32], two stabilizers can be simultaneously measured using three ancilla qubits, with one being a shared flag qubit, resulting in a [2 2] structure as shown in Fig. 7(c), and the details shown in Fig. 9. While the depth of each SE circuit stage increases to 7, the total circuit depth decreases to 14, compared to the circuit depth of 24 in the serial [1 1 1 1] scheme.

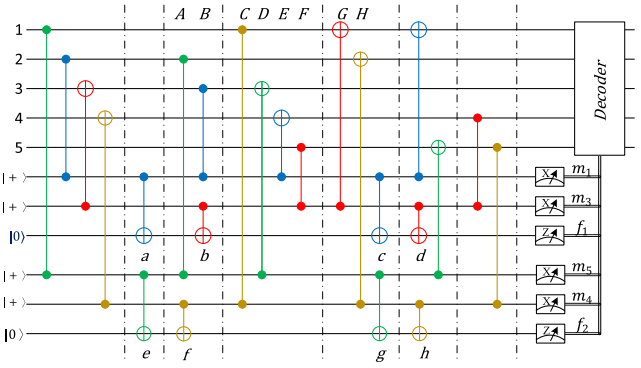

FIGURE 10. $\begin{bmatrix}2\\2\end{bmatrix}$ SE scheme for the ⟦5, 1, 3⟧ code, which extracts the syndrome bits of g_1 (blue), g_3 (red), g_5 (green), and g_4 (yellow) using two flag qubits.

Following the idea of Lemma 2, we can transform the [2 2] SE scheme into a $\begin{bmatrix}2\\2\end{bmatrix}$ SE scheme, as shown in Fig. 7(d), where two sets of shared flag SE circuits are used to concurrently measure four stabilizers. The details of the scheme are provided in Fig. 10, which has the smallest effective circuit area among the four schemes in Fig. 7, using only six ancillary qubits and six measurements.

We remark that g_1, g_5, g_3, and g_4 are measured in the [2 2] and $\begin{bmatrix}2\\2\end{bmatrix}$ SE schemes. By leveraging the outcomes of the measurement qubits when a flag rises and using the subsequent round of complete raw SE by Proposition 2, we are able to distinguish a broader range of errors and construct lookup tables. Consequently, we can construct the $\begin{bmatrix}2\\2\end{bmatrix}$ scheme with circuit depth 7, resulting in an effective circuit area lower than the fully parallel [1; 1; 1; 1] scheme. However, if g_1, g_2, g_3, and g_4 are measured instead, we only find a $\begin{bmatrix}2\\2\end{bmatrix}$ scheme with circuit depth 8. The construction of the lookup table for the $\begin{bmatrix}2\\2\end{bmatrix}$ SE scheme will be discussed in Section V-B after we introduce the QEC procedure for the ⟦5, 1, 3⟧ code.

V. QUANTUM ERROR CORRECTION WITH FLAG QUBITS

This section provides QEC procedures for codes with distances 3 or 5, with potential extensions for codes with higher distances.

A. DECISION TREE FOR DISTANCE-3 CODES

The flagged QEC rules for a distance-three quantum code are as follows. Initially, a complete flagged SE circuit is implemented with outcomes on the measurement qubits m and on the flag qubits f. If all measurement outcomes are zeros, the QEC procedure stops. Otherwise, nonzero measurement outcomes indicate the presence of a faulty event, and a complete raw SE circuit is performed, yielding outcomes m'. Error recovery is then performed based on all measurement outcomes m, f, and m' using a predetermined lookup table.

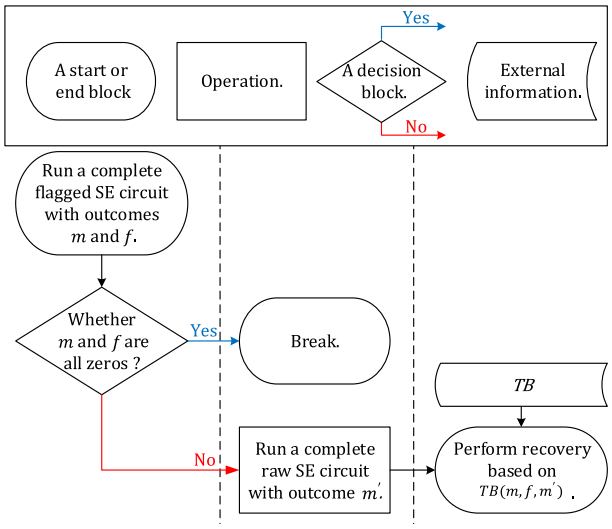

FIGURE 11. Decision tree for distance-3 codes. Each building block is listed in the top box. A rounded square represents the start or end of a procedure. A solid square represents an operation. A diamond indicates a decision with two outcomes, where the red line denotes the "no" event, and the blue line denotes the "yes" event. The last shape represents external data, such as the precomputed lookup table *TB*. Note that the content of the decision block, "whether *m* and *f* are all zeros?" should be interpreted as "Are all measurement qubit outcomes *m* and flag qubit outcomes *f* from the current round equal to zero?"

If no flag qubits are raised ($f = 0$), it suggests that there are no high-weight error propagation events, and standard syndrome decoding proceeds based on the measured error syndrome m' from the complete raw SE circuit. However, if any flag qubit is raised, a high-weight error correction is chosen after obtaining m'. This is done by exhaustively enumerating the one-fault events and ensuring that they have distinct generalized error syndromes up to degeneracy. Therefore, we can construct a lookup table indexed by m, f, and m'.

Fig. 11 depicts the decision tree for distance-3 codes.

B. CONSTRUCTING THE LOOKUP TABLE FOR THE [2; 2] SE SCHEME FOR THE ⟦5, 1, 3⟧ CODE

In this section, we demonstrate how to construct lookup tables for flagged SE schemes, using the $\begin{bmatrix}2\\2\end{bmatrix}$ scheme for the $[\![5, 1, 3]\!]$ code as an example. We will illustrate the effects of generalized error syndromes by utilizing the outcomes of measurement qubits and flag qubits, as suggested by Proposition 2.

Let us focus on the $\begin{bmatrix}2\\2\end{bmatrix}$ SE scheme in Fig. 10. Let $m_{1534} = m_1m_5m_3m_4 \in \{0, 1\}^4$ and $f = f_1f_2 \in \{0, 1\}^2$ be the measurement outcomes on the measurement qubits and flag qubits in a $\begin{bmatrix}2\\2\end{bmatrix}$ SE circuit. Let $m'_{1534} = m'_1m'_5m'_3m'_4 \in \{0, 1\}^4$ be the measurement outcomes of a complete raw SE circuits of stabilizers g_1, g_5, g_3, and g_4.

Consider Fig. 10. The CNOT and CZ gates labeled by the uppercase letters A–H and lowercase letters a–h indicate possible locations, where one-fault events may trigger flag qubits and generate high-weight error propagation events. Since the SE circuit for the $[\![5, 1, 3]\!]$ code has both CNOT and CZ gates, both X and Z errors will propagate through the circuit. Thus, we need to account for the effects of Y errors. We consider the following errors (which are applied to the qubits in the order they appear in the circuit, from top to bottom):

1) *A–H:* $IX, XX, ZX, YX, IY, XY, ZY, YY$;
2) *a, b, e, f:* XI, XX, YI, YX;
3) *c, d, g, h:* XZ, XY, YZ, YY.

We have identified a total of 96 high-weight error propagation events. For the lookup table to be effective, each high-weight error propagation event needs to have a unique syndrome, considering degeneracy. A traditional flagged QEC (cf. [32, Algorithm 1]) using the outcomes of the flag qubits (f) and a complete raw SE (m') would generate at most 2^{2+4} distinct syndromes, which is insufficient.

By leveraging Proposition 2, we can construct a lookup table of 2^{10} syndromes using m, f, and m'. Lookup tables for these high weight error propagation events are provided in Appendix D

Example 1: In Fig. 10, consider the (green) CZ at location A, which produces a Pauli error XY, denoted as A_{XY}. This results in a high-weight error $X_{2,3,5}$, triggering f_2. Using a complete raw SE circuit, the outcomes would be 0111.

Next, consider an error C_{YX} following the (yellow) CZ at location C, resulting in a high-weight error $Y_1X_2Z_5$, triggering f_2. Again, a complete raw SE circuit generates the syndrome bits 0111.

It can be observed that these two distinct high-weight error propagation events lead to the same syndrome. Consequently, a standard lookup table cannot differentiate between these two errors. However, leveraging the information from the first round of SE circuit allows us to expand the lookup table. For A_{XY}, $m_{1534}f_{12}$ is 010101, while for C_{YX}, it is 101001. Therefore, the generalized error syndrome for A_{XY} is updated to 0101010111, and for C_{YX}, it is updated to 1010010111. This enables the distinction between these two high-weight error propagation events for error correction purposes. □

Therefore, by utilizing the expanded syndrome space and optimizing the CNOT order, we obtain an FT $\begin{bmatrix}2\\2\end{bmatrix}$ SE scheme of depth 7 for the $[\![5, 1, 3]\!]$ code.

C. DECISION TREE FOR DISTANCE-5 CODES

The decision tree for distance-5 codes is depicted in Fig. 12, which extends the decision tree for distance-3 codes. Our goal is to correct any w-fault events for $w \leq 2$, using fewer than five rounds of flagged or raw SE. Therefore, we need to accurately identify no-fault, one-fault, and two-fault events. The decision tree has four routes, leading to

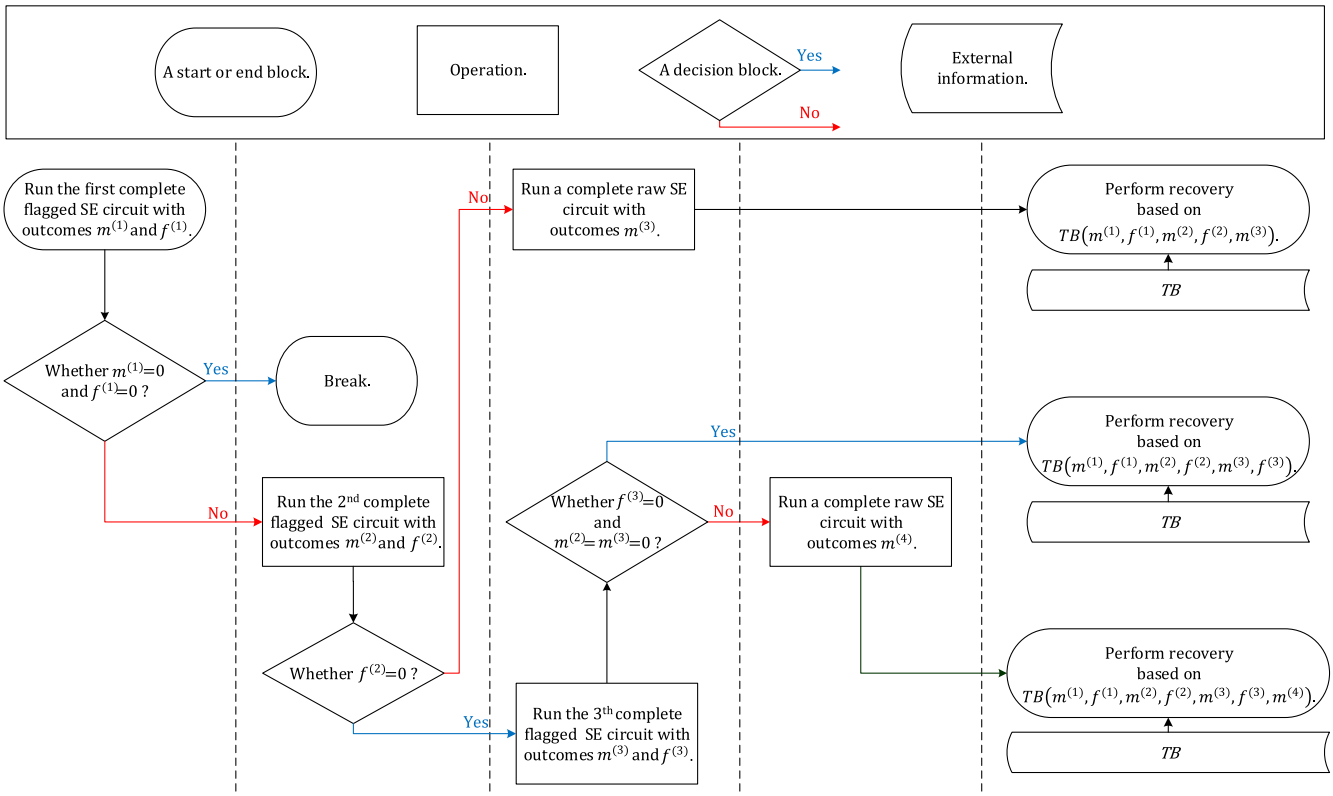

FIGURE 12. Decision tree for distance-5 codes. The building blocks are defined similarly to those in Fig. 11.

four possible end blocks. We will explain each route as follows. Assume that a lookup table, denoted TB, has been precomputed.

If all the measurement outcomes of the first complete flagged SE circuit are zero, we are confident that no faults have occurred. However, if a fault occurs in the data qubit during the first round and a corresponding error simultaneously affects the measurement qubit, the result manifests as an error in the data block that remains undetected through measurement. As errors occurring on data qubits do not propagate, the residue error maintains its weight-1. According to Definition 1, this scenario still maintains fault-tolerance properties. These scenarios correspond to the first end block in the decision tree.

When a single fault occurs in the first complete flagged SE circuit, nonzero measurement outcomes $m^{(1)}$ and $f^{(1)}$ are obtained from the measurement and flag qubits, respectively. We then run the second complete flagged SE circuit, which yields measurement outcomes $m^{(2)}$ and $f^{(2)}$. If $f^{(2)}$ is nonzero, it indicates a second fault that has triggered certain flag qubits, confirming a two-fault event. Consequently, a complete raw SE is applied, yielding outcomes $m^{(3)}$. Recovery is then performed using the precomputed lookup table TB with the entry $TB(m^{(1)}, f^{(1)}, m^{(2)}, f^{(2)}, m^{(3)})$. This scenario corresponds to the second end block in the decision tree. However, if the raw SE circuit also experiences a fault, it constitutes a three-fault event, which exceeds the error correction capability of this code.

On the other hand, if $f^{(2)} = 0$, we may have either a two-fault event or a one-fault event. In either case, we need to run the third complete flagged SE circuit, resulting in measurement outcomes $m^{(3)}$ and $f^{(3)}$. If $f^{(3)} = 0$ and $m^{(2)} = m^{(3)}$, we can confidently identify this as a one-fault event and perform error recovery accordingly. Otherwise, if the conditions do not match, it confirms a two-fault event, necessitating another round of complete raw SE to obtain outcome $m^{(4)}$ for error recovery. These scenarios correspond to the last two end blocks in the decision tree.

Note that all measurement outcomes will be used to expand the syndrome space. Overall, we can construct a lookup table TB indexed by $m^{(1)}$, $f^{(1)}$, $m^{(2)}$, $f^{(2)}$, $m^{(3)}$, $f^{(3)}$, and $m^{(4)}$ by exhaustively enumerating two-fault events in the aforementioned procedure and ensuring that they have distinct generalized error syndromes up to degeneracy.

D. DECISION TREE FOR DISTANCE-d CODES

Through analysis of consecutive measurement strings for equivalence, we can determine whether errors have occurred, while a raised flag qubit definitively indicates the occurrence of at least one fault. By quantifying the number of errors that have occurred, we can identify which round of measurement provides reliable results for recovery operations. Based on these principles, we have designed a decision tree structure for distance-d codes.

IEEE Transactions on Quantum Engineering

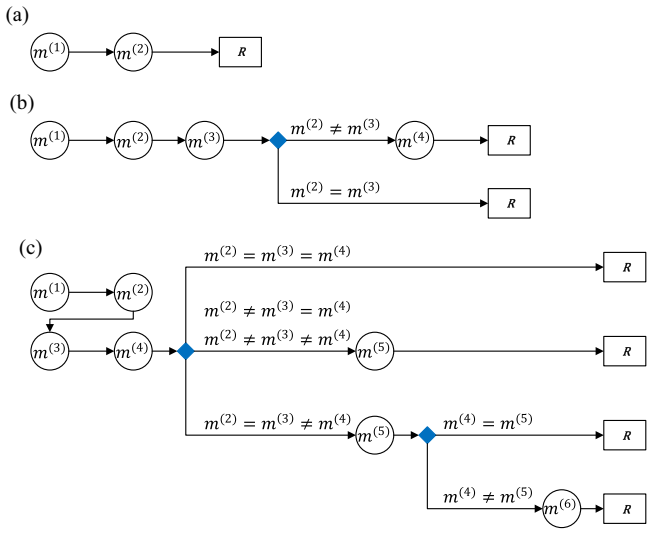

FIGURE 13. Simplified decision trees for various distance codes. (a) Simplified decision tree for distance-3 code derived from Fig. 11, excluding flag-raised scenarios. (b) Simplified decision tree for distance-5 code derived from Fig. 12, excluding flag-raised scenarios. (c) Simplified decision tree for distance-7 code where flag-raised cases are omitted. In these decision trees, blue diamonds represent decision nodes, with decision conditions shown on the connecting lines. Circles containing $m^{(i)}$ represent measurement outcomes at the ith round, while boxes marked with R indicate recovery operations performed using a lookup table, marking the termination of the procedure.

For clarity, we present a simplified tree structure in Fig. 13, omitting flag-raised scenarios for simplicity. Note that regardless of which round a flag qubit is raised in, this definitively indicates the occurrence of at least one fault. This identification does not require the condition of consecutive raised flags. Consequently, we have elected to omit discussion of flag-raised scenarios in our analysis. In addition, when $m^{(1)} = 0$, the procedure terminates immediately; this branch is excluded from the diagram for conciseness.

Let us examine case (c) in Fig. 13. The first decision block is reached when the initial round incurs between one and three faults. The decision conditions are outlined as follows.

1) When $m^{(2)} = m^{(3)} = m^{(4)}$, this indicates either:
   a) three faults occurred in the first round, or
   b) up to two faults occurred in the second round, with no faults in the third and fourth rounds. At this point, all three faults have occurred, and the generalized error syndrome is sufficient for recovery.
2) When $m^{(2)} \neq m^{(3)} = m^{(4)}$, this indicates:
   a) up to one fault in the second round, up to one fault in the third round, and no faults in the fourth round. All three faults have occurred, providing sufficient generalized error syndrome for recovery.
3) When $m^{(2)} \neq m^{(3)} \neq m^{(4)}$, this indicates:
   a) no errors in the second round, up to one fault in the third round, and up to one fault in the fourth round. All three faults have occurred, providing sufficient generalized error syndrome for recovery.
4) When $m^{(2)} = m^{(3)} \neq m^{(4)}$, this indicates:
   a) no errors in the second and third rounds, up to one fault in the fourth round, and one fault remains unrealized at this point, leading to the second decision block:
      i) If $m^{(4)} = m^{(5)}$: no faults in the fifth round, enabling recovery based on the generalized error syndrome.
      ii) If $m^{(4)} \neq m^{(5)}$: up to one fault in the fifth round; proceed to measure the sixth round and perform recovery using the generalized error syndrome.

By employing this approach to construct decision trees for distance-d codes, the measurement process can be completed within $d-1$ rounds. From the analysis of decision trees for distance-3, distance-5, and distance-7 codes, we can generalize the pattern for an arbitrary distance-d code as follows.

1) *General properties:*
   a) The number of faults that need to be considered is $\lfloor \frac{d-1}{2} \rfloor$.
   b) The maximum number of measurement rounds required is $d-1$.
   c) The first measurement $m^{(1)}$ being 0 leads to immediate termination.
2) *Decision tree structure:*
   For a distance-d code, where d is odd:
   a) the decision blocks compare consecutive measurement outcomes $m^{(i)}$;
   b) the maximum number of rounds $(d-1)$ is reached.
3) *Decision conditions:*
   For round i, the decision block evaluates the following.
   a) When consecutive measurements show equality $(m^{(i)} = m^{(i+1)} = \cdots = m^{(i+\lfloor \frac{d-1}{2} \rfloor)})$, the recovery operation can be performed.
   b) When measurements differ, the process continues to the next round until the accumulated faults reach $\lfloor \frac{d-1}{2} \rfloor$, at which point recovery can be executed.
4) *Termination criteria:*
   The process terminates when either:
   a) the generalized error syndrome is sufficient for recovery;
   b) $d-1$ rounds of measurements have been completed.

E. ADAPTIVE SE

Based on previous work [26], [27], [29], [30], we propose a straightforward adaptive SE protocol to enhance FT

TABLE 1. Partial Adaptive Table for the $\begin{bmatrix}2\\2\end{bmatrix}$ SE Scheme for the ⟦5, 1, 3⟧ Code of Fig. 10

Fault	Data X error	Data Z error	$m_{1534}f_{12}$	m'_{1534}	Adaptive index
a_{XZ}	10010	00100	111010	0001	2002
a_{YI}	10010	00100	111010	0001	2002
B_{ZY}	10010	00100	111010	0001	2002
b_{XZ}	10000	00011	111010	1000	2002
B_{YY}	10110	00100	111010	1001	2002

procedures using a lookup-table-based decoder. This protocol reduces unnecessary stabilizer measurements in the final stage of complete raw SE.

The protocol is as follows.

1) At each parent node of a leaf of a decision tree, errors yielding identical measurement outcomes in previous SE rounds are assigned the same group number.
2) For each group of errors, identify stabilizers in the raw SE stage that produce identical outcomes for all errors in the group. These stabilizers can then be excluded from the final stage of raw SE.

Consider the Pauli error a_{XZ}, a_{YI}, B_{ZY}, b_{XZ}, and B_{YY} in the $\begin{bmatrix}2\\2\end{bmatrix}$ SE scheme for the [[5,1,3]] code, as shown in Fig. 10. Their syndrome measurement outcomes are provided in Table 1 . This group of errors has an identical error syndrome 111010, with a_{XZ}, a_{YI}, and B_{ZY} being degenerate. Thus, there are three distinct errors after the first round of SE, and they can be further distinguished by measuring g_1 and g_4. In addition, we observe that m'_2 and m'_3 have identical values for these errors, making it unnecessary to measure g_2 and g_3.

Consequently, errors with identical measurement outcomes $m_{1534}f_{12}$ are assigned to the same group, and we define a four-digit adaptive index $a_1a_5a_3a_4$ to represent the adaptive measurements for m'_{1534} needed in the final stage of raw SE. If $a_j = 2$, it indicates that m'_j must be remeasured to distinguish between errors within the same group. Conversely, $a_j = 0$ or $a_j = 1$ implies that $m'_j = 0$ or $m'_j = 1$ for all errors in the group, allowing those corresponding measurements to be omitted. Using this method, we can further reduce the effective circuit area required for SE.

Partial adaptive tables are provided in Appendix D.

This approach is also applicable to high-distance codes, reducing the overhead in the final stage of raw SE at each parent node of the leaf nodes in a decision tree. Further details are provided in Appendix E.

VI. SIMULATION RESULTS

In this section, we simulate various proposed SE schemes and compare these results with the literature. We also include the ⟦19, 1, 5⟧ code, which is defined by the stabilizers

$$g_1 = Z_1Z_2Z_3Z_4 \qquad g_2 = Z_1Z_3Z_5Z_7$$
$$g_3 = Z_{12}Z_{13}Z_{14}Z_{15} \qquad g_4 = Z_1Z_2Z_5Z_6Z_8Z_9$$
$$g_5 = Z_6Z_9Z_{16}Z_{19} \qquad g_6 = Z_{16}Z_{17}Z_{18}Z_{19}$$
$$g_7 = Z_{10}Z_{11}Z_{12}Z_{15} \qquad g_8 = Z_8Z_9Z_{10}Z_{11}Z_{16}Z_{17}$$
$$g_9 = Z_5Z_7Z_8Z_{11}Z_{12}Z_{13} \qquad g_{10} = X_1X_2X_3X_4$$
$$g_{11} = X_1X_3X_5X_7 \qquad g_{12} = X_{12}X_{13}X_{14}X_{15}$$
$$g_{13} = X_1X_2X_5X_6X_8X_9 \qquad g_{14} = X_6X_9X_{16}X_{19}$$
$$g_{15} = X_{16}X_{17}X_{18}X_{19} \qquad g_{16} = X_{10}X_{11}X_{12}X_{15}$$
$$g_{17} = X_8X_9X_{10}X_{11}X_{16}X_{17} \qquad g_{18} = X_5X_7X_8X_{11}X_{16}X_{17}$$

and the logical operators are

$$\bar{X} = X^{\otimes 19} \qquad \bar{Z} = Z^{\otimes 19}.$$

Various SE schemes for the ⟦19, 1, 5⟧ code are illustrated in Fig. 1, with detailed explanations provided in Appendix F.

We consider the circuit-level noise model, as outlined in Section II-B. Experiments across various quantum platforms reveal differing error rates. In trapped ions, idle qubits have an error rate of approximately $p_{\text{idle}} \approx 7.5 \times 10^{-5}$, with Mølmer–Sørensen gate errors at $p_{\text{MS}} \approx 10^{-3}$ [37]. On superconducting platforms, idle qubit error rates range from $p_{\text{idle}} = 10^{-4}$ to 4×10^{-2}, while CZ gate errors range from $p_{\text{CZ}} = 10^{-3}$ to 10^{-2} [44]. Neutral atom systems show idle qubit error rates from $p_{\text{idle}} = 10^{-3}$ to 4×10^{-2} and two-qubit controlled gate errors around $p_2 \approx 5 \times 10^{-3}$ [55]. We conduct simulations for γ values of 0.01 and 1, where γ represents the ratio between idle qubit error rates and gate error rates. A γ value of 0 indicates that no errors occur on idle qubits, while $\gamma = 1$ indicates that idle qubits have the same error rate p as gates. In our simulations, $\gamma = 0.01$ represents the case where idle qubit error rates are significantly reduced to 1% of the gate error rates, approximating a scenario where idle qubit errors have minimal impact on the circuit's performance [37], [44], [55]. On the other hand, we simulate for β values of 1 and 10. Conventionally, the measurement error rate is assumed to be roughly the same as the gate error rate. However, recent experimental results indicate that measurement error rates can be up to ten times higher than gate error rates in worst case scenarios [44].

We summarize the number for gate for each type, number of qubits, circuit depth, and circuit area for one round of flagged SE for each scheme in Table 2. For the ⟦17, 1, 5⟧ and ⟦19, 1, 5⟧ CSS codes, only the SE circuits of the Z-type stabilizers are accounted for. For simplicity, we make the following assumptions.

1) The circuit depth is measured as the longest CNOT or CZ depth.
2) Ancilla qubits can be reused, so the number of physical qubits for a scheme is the maximum number of qubits required at any instance.
3) The number of idle qubits is calculated as the circuit depth times the number of qubits minus twice the number of two-qubit gates.

IEEE Transactions on Quantum Engineering

TABLE 2. Resources Required for a Round of Flagged SE for Various Schemes

Code	$[\![5,1,3]\!]$			$[\![17,1,5]\!]$			$[\![19,1,5]\!]$		
Scheme	$\mathbf{1}^T$	$\begin{bmatrix}2\\2\end{bmatrix}$	$\mathbf{1}$	$\mathbf{1}^T$	$\begin{bmatrix}2\\2\\2\\1\\1\end{bmatrix}$	$\mathbf{1}$	$\mathbf{1}^T$	$\begin{bmatrix}2\\2\\2\\1\\1\\1\end{bmatrix}$	$\mathbf{1}$
ancillary state preparation	8	6	8	18	15	18	21	18	21
measurement in X	4	4	4	10	7	10	12	9	12
measurement in Z	4	2	4	8	8	8	9	9	9
Two-qubit gate	24	24	24	56	56	56	66	66	66
Idle qubit	120	29	30	1064	336	378	1320	238	268
Circuit depth	24	7	6	56	14	14	66	10	10
Physical qubits[a]	5+2	5+6	5+8	17+4	17+15	17+18	19+3	19+18	19+21
Circuit area [b] ($\beta=1,\gamma=1$)	284.8	135.2	140.8	1917.6	746.8	820	2365.2	628	682
Circuit area ($\beta=1,\gamma=0.01$)	56.704	51.248	55.264	143.52	125.872	132.544	169.776	146.464	152.944
Circuit area ($\beta=10,\gamma=1$)	356.8	189.2	212.8	2079.6	881.8	982	2554.2	790	871
Circuit area ($\beta=10,\gamma=0.01$)	128.704	105.248	127.264	305.52	260.872	294.544	358.776	308.464	341.944

[a] (a_1)+(a_2): (a_1) represents the number of data qubits, and (a_2) represents the total number of measurement qubits and flag qubits.
[b] Due to space limitations, the term "Circuit area" hereafter refers to the effective circuit area.

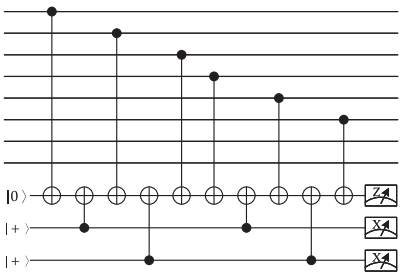

FIGURE 14. Example of the weight-6 flagged SE.

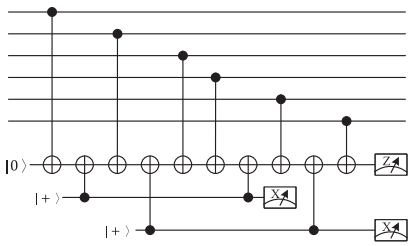

FIGURE 15. Example of the weight-6 flagged SE.

In practice, the circuit area can be reduced. For example, the circuit in Fig. 14 can be improved to Fig. 15 with a smaller circuit area. We omit this kind of optimization.

In the serial $\mathbf{1}^T$ schemes, the ancillary qubits will be measured and reset, but we omit these steps here.

In the following simulated performance curves of physical gate error rate p versus logical error rate p_L, we collect at least 800 logical errors for each physical error rate using Monte Carlo simulations.

Recall that the error threshold considered in Definition 4 compares the logical error rate of an encoded codeword with the physical error rate of the dominant CNOT/CZ gates. Therefore, a simulated pseudothreshold for a scheme is determined by the intersection point between its performance curve and the reference line, which is defined by $p = p_L$. As shown in Table 2, the three flag-sharing schemes have the smallest effective circuit areas in all four scenarios considered for the three codes, respectively. Consequently, we expect that the flag-sharing schemes will outperform the others in terms of error thresholds.

In this section, the label "fully parallel with serial" indicates that the "fully parallel" component refers to the flag scheme, while "with serial" denotes that the raw SE circuit for the final measurement round is arranged in a serial configuration. Other types include "with parallel," where the raw SE circuit for the final measurement round is arranged in a parallel configuration. The label "with serial adaptive" indicates a serial arrangement of the raw SE circuit in the final round, incorporating the adaptive SE scheme from Section V-E. Similarly, "with parallel adaptive" denotes a parallel arrangement of the raw SE circuit in the final round, also incorporating the adaptive syndrome extraction scheme from Section V-E.

A. RESULTS FOR THE $[\![5,1,3]\!]$ CODE

Using the decision tree (see Fig. 11) and lookup table (see Table 7), we conduct simulations of the $[\![5, 1, 3]\!]$ code under various combinations of SE schemes.

Fig. 16 shows the simulation results for $\beta = 1$ and $\gamma = 0.01$. The adaptive versions of both serial and parallel SE schemes outperform their nonadaptive counterparts. At low idle error rates, the advantage of the parallel scheme is not as evident, but it may introduce intricate error propagation. Therefore, the fully parallel scheme with serial adaptive raw SE delivers better performance.

The performance curves in this regime are linear in a log-scale plot with slope close to 2, and thus, the performance of various schemes can be compared by their pseudothresholds. Consequently, we omit the performance curves for the other cases and summarize the simulated pseudothresholds in Table 3.

Both the $\begin{bmatrix}2\\2\end{bmatrix}$ and $\mathbf{1}$ schemes have better pseudothreshold at $\gamma = 1$ than the traditional serial flag scheme. The effective circuit areas for the $\begin{bmatrix}2\\2\end{bmatrix}$ flag-sharing scheme and the

TABLE 3. Pseudothresholds for the Flag-Sharing $\begin{bmatrix}2\\2\end{bmatrix}$ and the Fully Parallel 1 SE Schemes for the $[\![5, 1, 3]\!]$ Code

Scheme	$\gamma = 0.01$	$\gamma = 1$	$\gamma = 0.01$	$\gamma = 1$
	$\beta = 1$		$\beta = 10$	
$[\![5, 1, 3]\!]$ serial [1 1 1 1] scheme [25]	–	7.09×10^{-5}	–	–
$[\![5, 1, 3]\!]$ fully parallel $[1\ 1\ 1\ 1]^T$ with serial	9.48×10^{-4}	2.86×10^{-4}	2.34×10^{-4}	1.10×10^{-4}
$[\![5, 1, 3]\!]$ fully parallel $[1\ 1\ 1\ 1]^T$ with serial adaptive	1.03×10^{-3}	3.09×10^{-4}	2.28×10^{-4}	1.14×10^{-4}
$[\![5, 1, 3]\!]$ fully parallel $[1\ 1\ 1\ 1]^T$ with parallel	9.27×10^{-4}	5.09×10^{-4}	2.32×10^{-4}	1.66×10^{-4}
$[\![5, 1, 3]\!]$ fully parallel $[1\ 1\ 1\ 1]^T$ with parallel adaptive	1.01×10^{-3}	3.30×10^{-4}	2.14×10^{-4}	1.16×10^{-4}
$[\![5, 1, 3]\!]$ flag-sharing $[2\ 2]^T$ with serial	9.71×10^{-4}	2.75×10^{-4}	3.46×10^{-4}	1.36×10^{-4}
$[\![5, 1, 3]\!]$ flag-sharing $[2\ 2]^T$ with serial adaptive	9.75×10^{-4}	2.76×10^{-4}	3.10×10^{-4}	1.26×10^{-4}
$[\![5, 1, 3]\!]$ flag-sharing $[2\ 2]^T$ with parallel	9.78×10^{-4}	4.70×10^{-4}	3.27×10^{-4}	2.01×10^{-4}
$[\![5, 1, 3]\!]$ flag-sharing $[2\ 2]^T$ with parallel adaptive	9.79×10^{-4}	4.64×10^{-4}	2.96×10^{-4}	1.84×10^{-4}

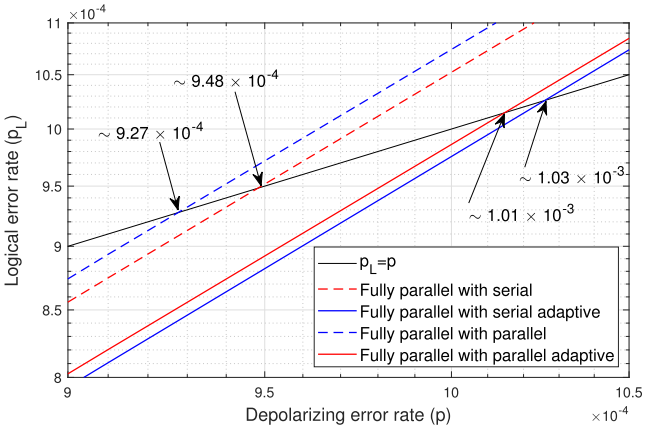

FIGURE 16. Simulations of the 1 SE schemes for the $[\![5, 1, 3]\!]$ code for $\gamma = 0.01$ at $\beta = 1$.

fully parallel **1** scheme are similar, resulting in comparable overall performances. However, the flag-sharing scheme utilizes fewer ancillary qubits.

At $\beta = 1$ and $\gamma = 1$, although the number of idle qubits in the $\begin{bmatrix}2\\2\end{bmatrix}$ scheme is only slightly lower than in the **1** scheme, the CNOT structure in the shared-flag scheme is significantly more intricate, leading to a slightly poorer performance.

Figs. 17 and 18 show the variation of the pseudothreshold for the $[\![5, 1, 3]\!]$ code as a function of $\gamma \in [0, 1]$ under different schemes, with $\beta = 1$ and $\beta = 10$, respectively.

Note that a CNOT gate introduces, on average, 1.6 qubit errors on two qubits. Therefore, when γ is high, two idle qubit errors could be more harmful than a CNOT gate error. In the parallel scheme, an adaptive protocol replaces some CNOT gates with idle locations, which become problematic at high idle error rates. As a result, for $\gamma \gtrsim 0.33$, nonadaptive parallel schemes outperform their adaptive counterparts.

When considering the case $\beta = 10$, where the measurement errors are the worst error sources, the $\begin{bmatrix}2\\2\end{bmatrix}$ scheme outperforms the **1** scheme for both $\gamma = 0.01$ and $\gamma = 1$. This reflects the fact that the $\begin{bmatrix}2\\2\end{bmatrix}$ scheme uses 25% less measurements than the **1** scheme. Moreover, adaptive schemes perform worse than their nonadaptive counterparts at both low and high values of γ. This is because adaptive schemes are highly dependent on the measurement outcomes, making them less robust when the measurement error rate is high.

TABLE 4. Optimal Scheme Selection for the $[\![5, 1, 3]\!]$ Code for Various Scenarios Across Different Values of γ and β

$\beta = 1$	
$\gamma \gtrsim 0.33$	the **1** scheme with parallel raw SE
$0.08 \lesssim \gamma \lesssim 0.33$	the $\begin{bmatrix}2\\2\end{bmatrix}$ scheme with parallel adaptive raw SE
$\gamma \lesssim 0.08$	the **1** scheme with serial adaptive raw SE
$\beta = 10$	
$\gamma \gtrsim 0.09$	the $\begin{bmatrix}2\\2\end{bmatrix}$ scheme with parallel raw SE
$\gamma \lesssim 0.09$	the $\begin{bmatrix}2\\2\end{bmatrix}$ scheme with serial raw SE

TABLE 5. Pseudothresholds for Various Distance-3 Code

Scheme	$\gamma = 0.01$	$\gamma = 1$
	$\beta = 1$	
$[\![7, 1, 3]\!]$ parallel [32]	1.25×10^{-3}	1.75×10^{-4}
$[\![9, 1, 3]\!]$ parallel [32]	8.81×10^{-3}	8.77×10^{-4}
Bacon-Shor-13 [56]	8.09×10^{-3}	6.41×10^{-4}
Scheme	$\gamma = 0.01$	$\gamma = 1$
	$\beta = 10$	
$[\![7, 1, 3]\!]$ parallel	9.82×10^{-4}	1.6×10^{-4}
$[\![9, 1, 3]\!]$ parallel	5.07×10^{-3}	8.24×10^{-4}
Bacon-Shor-13	2.71×10^{-3}	4.24×10^{-4}

The optimal scheme selections for different values of γ and β are as summarized in Table 4.

For reference, we have also simulated other distance-3 codes in Table 5. We can see that at $\gamma = 1$, the $[\![5, 1, 3]\!]$ code emerges as a competitive candidate for near-term experiments, as it has a comparable pseudothreshold to the $[\![7, 1, 3]\!]$ and $[\![9, 1, 3]\!]$ codes, while using fewer qubits.

B. RESULTS FOR DISTANCE-5 CODES

Given the limited improvements offered by adaptive schemes for the $[\![5, 1, 3]\!]$ code, we focus only on the parallel and serial raw SE configurations for the final round, excluding adaptive schemes, in the following simulations.

We utilize the decision tree for distance-5 codes in Fig. 12 to simulate the flag-sharing $\begin{bmatrix}2&2&2&1&1\end{bmatrix}^T$ and fully parallel **1** SE schemes for the $[\![17, 1, 5]\!]$ code and the flag-sharing $\begin{bmatrix}2&2&2&1&1&1\end{bmatrix}^T$ and fully parallel **1** SE schemes for the $[\![19, 1, 5]\!]$ code. Fig. 19 presents the simulation results for

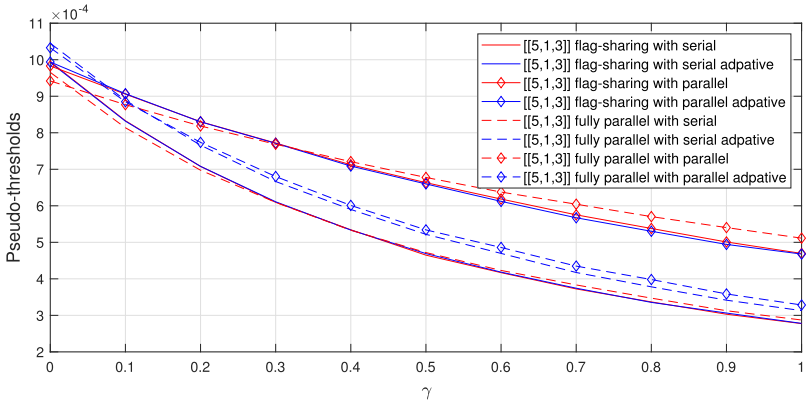

FIGURE 17. Pseudothreshold of the $\begin{bmatrix} 2 \\ 2 \end{bmatrix}$ and 1 SE schemes for the $[\![5, 1, 3]\!]$ code for different γ at $\beta = 1$.

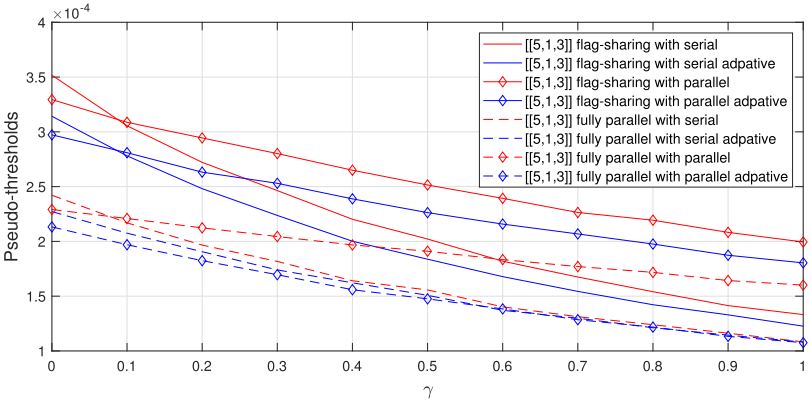

FIGURE 18. Pseudothreshold of the $\begin{bmatrix} 2 \\ 2 \end{bmatrix}$ and 1 SE schemes for the $[\![5, 1, 3]\!]$ code for different γ at $\beta = 10$.

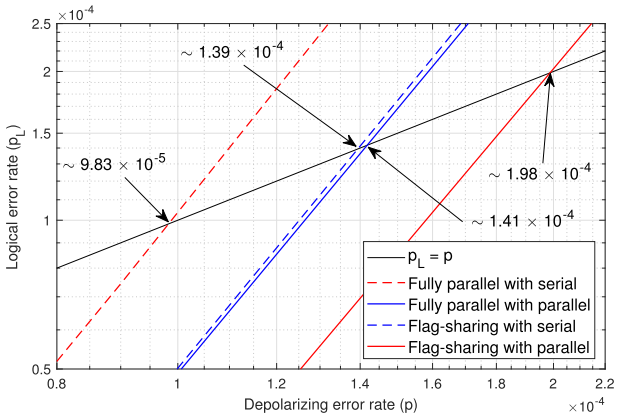

FIGURE 19. Simulations of the 1 and $[2 \quad 2 \quad 2 \quad 1 \quad 1 \quad 1]^T$ SE schemes for the $[\![19, 1, 5]\!]$ code at $\gamma = 1$ and $\beta = 1$.

the $[\![19, 1, 5]\!]$ code at $\beta = 1$ and $\gamma = 1$. In this regime, the performance of various schemes can be effectively compared using their pseudothresholds, which are summarized in Table 6. In general, parallel configurations in the raw SE circuits exhibit improved performance over serial counterparts.

For the $[\![17, 1, 5]\!]$ codes at $\gamma = 1$ and $\beta = 1$, both our flag-sharing and fully parallel schemes exhibit pseudothresholds almost an order of magnitude better than the serial scheme and the [4 4] scheme in [33], as shown in Table 2. This is expected given the effective circuit areas in Table 2, where both our flag-sharing and fully parallel schemes use less than half the area of the serial scheme in a single round of SE. Since three or four rounds of SE are performed in general, the effects are further amplified, resulting in much better pseudothresholds for our schemes.

Similarly, both our flag-sharing and fully parallel schemes for the $[\![19, 1, 5]\!]$ codes demonstrate pseudothresholds three to eight times better than the serial scheme and the [3 3 3] scheme in [33]. These results highlight the effectiveness of our method, using flag sharing and parallelism with generalized error syndromes to reduce the effective circuit areas.

At $\beta = 1$, the flag-sharing $\begin{bmatrix} 2 & 2 & 2 & 1 & 1 \end{bmatrix}^T$ scheme performs slightly better than the fully parallel $\mathbf{1}$ scheme for the $[\![17, 1, 5]\!]$ code since its effective circuit area is smaller. At $\beta = 10$, the gap between the two schemes is amplified since the flag-sharing scheme uses $1/6$ less measurements.

In summary, our flag-sharing schemes outperform the fully parallel schemes for the $[\![17, 1, 5]\!]$ and $[\![19, 1, 5]\!]$ codes across the parameter ranges considered.

The decoding performance of these nonadaptive schemes is strongly correlated with their effective circuit areas. As shown in Fig. 20, the calculated effective circuit areas under

TABLE 6. Pseudothresholds for Various Schemes of the ⟦17, 1, 5⟧ and ⟦19, 1, 5⟧ Codes

Scheme	$\gamma = 0.01$	$\gamma = 1$	$\gamma = 0.01$	$\gamma = 1$
	$\beta = 1$		$\beta = 10$	
$[\![17, 1, 5]\!]$ serial scheme [33]	–	5.497×10^{-6}	–	–
$[\![17, 1, 5]\!]$ [4 4] scheme [33]	–	7.067×10^{-6}	–	–
$[\![17, 1, 5]\!]$ fully parallel with serial	4.55×10^{-4}	7.34×10^{-5}	2.68×10^{-4}	6.40×10^{-5}
$[\![17, 1, 5]\!]$ fully parallel with parallel	4.60×10^{-4}	8.76×10^{-5}	2.68×10^{-4}	7.95×10^{-5}
$[\![17, 1, 5]\!]$ $[2\,2\,2\,1\,1]^T$ with serial	4.64×10^{-4}	7.47×10^{-5}	2.93×10^{-4}	6.78×10^{-5}
$[\![17, 1, 5]\!]$ $[2\,2\,2\,1\,1]^T$ with parallel	4.59×10^{-4}	9.61×10^{-5}	3.01×10^{-4}	8.31×10^{-5}
$[\![19, 1, 5]\!]$ serial scheme [25]	–	1.14×10^{-5}	–	–
$[\![19, 1, 5]\!]$ serial scheme [33]	–	1.183×10^{-5}	–	–
$[\![19, 1, 5]\!]$ [3 3 3] scheme [33]	–	1.992×10^{-5}	–	–
$[\![19, 1, 5]\!]$ fully parallel with serial	4.82×10^{-4}	9.83×10^{-5}	3.55×10^{-4}	8.74×10^{-5}
$[\![19, 1, 5]\!]$ fully parallel with parallel	4.91×10^{-4}	1.41×10^{-4}	3.53×10^{-4}	1.22×10^{-4}
$[\![19, 1, 5]\!]$ $[2\,2\,2\,1\,1\,1]^T$ with serial	6.01×10^{-4}	1.39×10^{-4}	4.41×10^{-4}	1.16×10^{-4}
$[\![19, 1, 5]\!]$ $[2\,2\,2\,1\,1\,1]^T$ with parallel	6.06×10^{-4}	1.98×10^{-4}	4.37×10^{-4}	1.70×10^{-4}

Note that entries marked with a "–" indicate either cases where different reference lines were used for calculations or where data were unavailable.

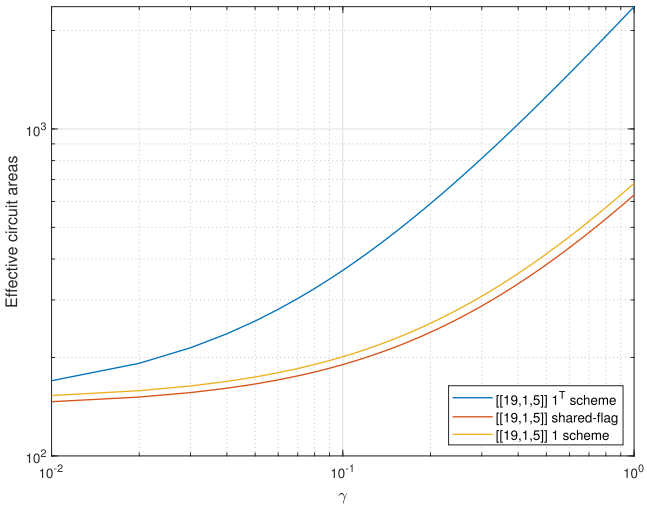

FIGURE 20. Effective circuit areas for the three ⟦19, 1, 5⟧ schemes with $\beta = 1$.

various parameters indicate that the flag-sharing schemes consistently result in smaller effective circuit areas compared to the fully parallel schemes. Based on this observation, we anticipate that the flag-sharing schemes will outperform the fully parallel schemes across all regimes.

VII. CONCLUSION

In this article, we proposed both sequential flag-sharing and fully parallel SE schemes for general CSS stabilizer codes, which outperform traditional serial flag schemes [25] and other structural designs proposed in [33].

Various flag-sharing schemes can be realized through different forms of tiling, providing a flexible method in the design of FT SE circuits. In addition, we introduced the notion of effective circuit area, serving as a rough but quick tool for evaluating circuit quality.

By constructing a decision tree for decoding [25], [37], we obtained a generalized error syndrome table with an enlarged syndrome space . However, most of these generalized syndrome spaces are not used, leaving room for optimization. For instance, we did not construct a table of 2^{10} corrections for the ⟦5, 1, 3⟧ code, suggesting potential optimization to handle some two-fault events. Similar opportunities for optimization exist for the ⟦17, 1, 5⟧ and ⟦19, 1, 5⟧ codes. In addition, our decision trees for distance-3 and distance-5 codes have shorter depths compared to those presented in [25].

We did not find a [2 2 2 2] SE scheme for the ⟦17, 1, 5⟧ code, which, if it exists, may imply an improved threshold over the flag-sharing $[2\,2\,2\,1\,1]^T$ scheme. Similar discussion applies to the ⟦19, 1, 5⟧ code as well.

We also presented versatile SE schemes for the ⟦5, 1, 3⟧ code. By using a different set of stabilizer generators and generalized error syndromes, we created an SE circuit featuring the smallest circuit area reported in the literature. Moreover, we utilized the cyclic structure of the stabilizer generators to construct versatile SE schemes for the ⟦5, 1, 3⟧ code, which may be generalized to other non-CSS codes.

We proposed an adaptive scheme for the final stage of raw SE to reduce the number of stabilizer measurements. However, we observe that adaptive SE performs best from our chosen variants only at low idle and measurement error rates. In parallel schemes, higher idle error rates can be detrimental, as idle errors may outweigh the benefits of reducing stabilizer measurements. In addition, the adaptive protocol's reliance on measurement outcomes makes it less robust when the measurement error rate is high.

Our estimated thresholds are approximately 10^{-5} and 10^{-4}, which are still one to two orders of magnitude lower than the gate error rates reported in recent experiments involving trapped ions, superconducting platforms, and neutral atom systems [37], [44], [55].

Using generalized error syndromes for lookup tables, we constructed decision trees simplified from those in [25]. The decisions we presented in Figs. 11 and 12 can be recursively extended for codes with higher distances. By using all the measurement outcomes as the index for the lookup table, we presented the decision tree in a compressed manner. It is worth exploring the rapid and automated construction of syndrome tables or developing alternative types of decoders.

The source files of our simulations can be found in [57].

IEEE Transactions on Quantum Engineering

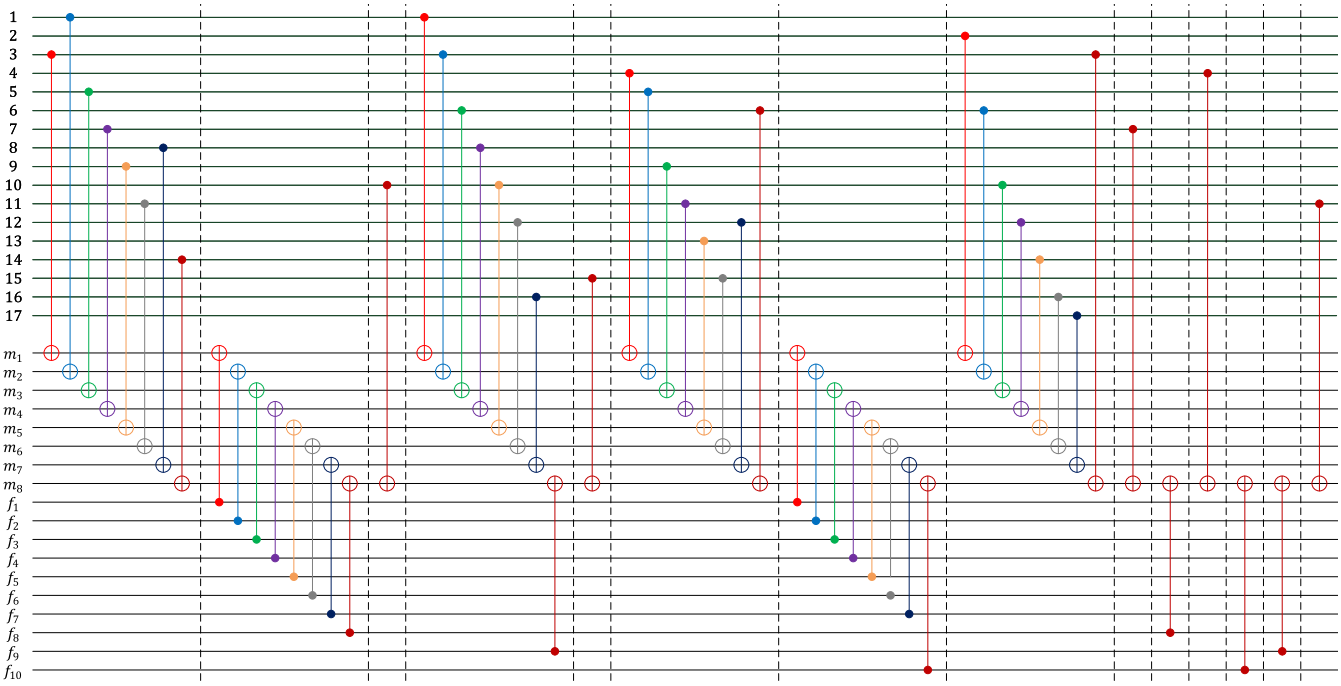

FIGURE 21. Fully parallel 1 SE circuit for the Z-type stabilizers of the $[\![17, 1, 5]\!]$ code. Each stabilizer measurement is represented by a distinct color. Each stabilizer measurement has an independent set of ancilla and flag qubits. Note that, for space-saving purpose, we have omitted state preparations and measurements. This circuit has the same depth as the longest stage of the [4 4] SE scheme in [33], but all stabilizers are measured.

APPENDIX A
EFFECTIVE CIRCUIT AREA DERIVATION FOR THE $\mathbf{1}^T$ SCHEME

For an n-qubit code with s stabilizer generators of mean weight w, we assume that each stabilizer measurement requires $\frac{1}{2}w$ flag qubits. The depth of each SE circuit stage is $w + 2(\frac{1}{2}w) = 2w$, which includes both stabilizer and flag qubit operations. A single-stage SE circuit consists of w CNOT gates for stabilizer operation and $2(\frac{1}{2}w) = w$ CNOT gates for flag qubit operations, summing to $2w$ CNOT gates. When implemented across all s stabilizers in the $\mathbf{1}^T$ scheme, the total number of CNOT gates is $2ws$. The protocol requires $s(1 + \frac{1}{2}w) = (s + \frac{1}{2}ws)$ state preparations, where s ancilla qubits are used for stabilizer measurements and $s(\frac{1}{2}w)$ for flag qubits, with an equal number of measurements.

The circuit area can be calculated using Definition 4 as follows:

$$1.6(2ws)\left(n + 1 + \frac{1}{2}w\right) + \left(s + \frac{1}{2}sw\right) + \beta\left(s + \frac{1}{2}sw\right) \tag{6}$$

where 1.6 represents the CNOT coefficient and β denotes the measurement modification factor. The number of idle qubits is obtained by subtracting the active components from (6), specifically the CNOT operations multiplied by 1.6, state preparations, and measurements multiplied by β

$$\frac{16}{5}sw\left(n + \frac{w}{2} + 1\right) - \frac{16}{5}sw. \tag{7}$$

The effective circuit area is then calculated by incorporating the idle qubit penalty factor γ, through subtracting the idle qubit count and adding back the idle qubits multiplied by γ, yielding

$$s + \frac{37}{10}sw + \beta\left(s + \frac{1}{2}sw\right) - \gamma\left(\frac{16}{5}sw - \frac{16}{5}sw\left(n + \frac{1}{2}w + 1\right)\right). \tag{8}$$

APPENDIX B
EFFECTIVE CIRCUIT AREA DERIVATION FOR THE 1 SCHEME

For an n-qubit code with r stabilizer generators of high weight $2w$, each stabilizer measurement requires $\frac{1}{2}w$ flag qubits. Assume that the depth of **1** SE circuit stage is $3w$. The protocol requires $r + \frac{1}{2}rw$ state preparations, where r ancilla qubits are used for stabilizer measurements and $\frac{1}{2}rw$ for flag qubits, with an equal number of measurements.

Similar to Appendix A, the circuit area can be calculated using Definition 4 as follows:

$$1.6(3w)\left(n + r + \frac{1}{2}rw\right) + \left(r + \frac{1}{2}rw\right) + \beta\left(r + \frac{1}{2}rw\right). \tag{9}$$

The number of idle qubits is obtained by subtracting the active components from (9), specifically the CNOT operations multiplied by 1.6, state preparations, and measurements multiplied by β

$$\frac{24}{5}w\left(n + r + \frac{rw}{2}\right) - \frac{24}{5}sw. \tag{10}$$

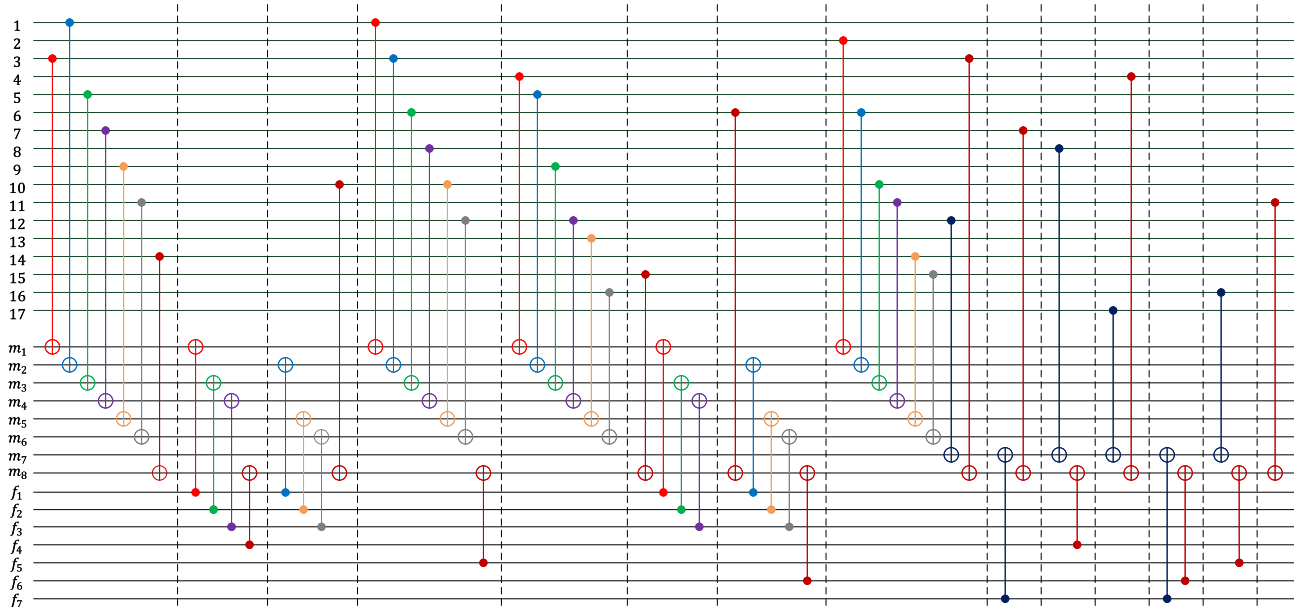

FIGURE 22. Parallel $[2 \quad 2 \quad 2 \quad 1 \quad 1]^T$ flag-sharing SE circuit for the Z-type stabilizers of the $[\![17, 1, 5]\!]$ code. Note that g_1 and g_2 share a flag qubit f_1, g_3 and g_5 share a flag qubit f_2, and g_4 and g_6 share a flag qubit f_3. The stabilizer g_7 uses the flag qubit f_7, while the stabilizer g_8 uses the flag qubits f_4, f_5, and f_6. This circuit has the same depth as the longest stage of the $[4 \quad 4]$ SE scheme in [33], but all stabilizers are measured.

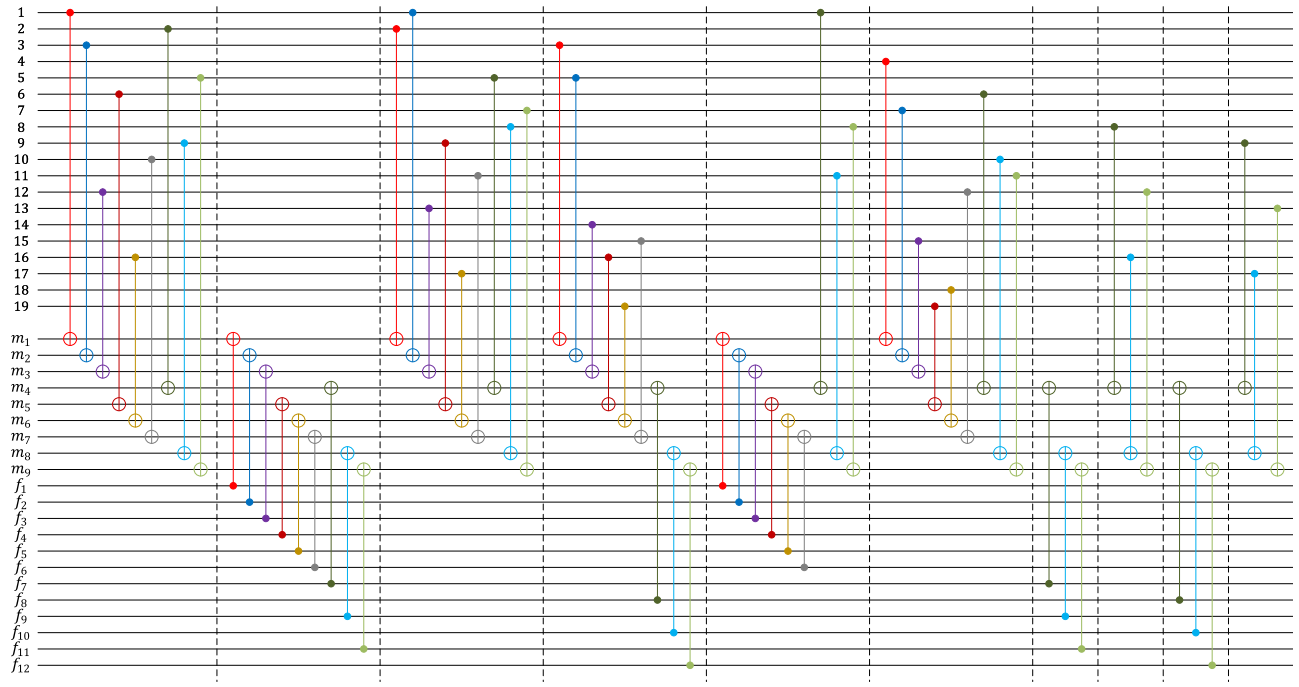

FIGURE 23. **1** SE scheme for the Z-type stabilizers of the $[\![19, 1, 5]\!]$ code. This circuit has a shorter depth than any stage of the $[3 \quad 3 \quad 3]$ SE scheme in [33], while still measuring all stabilizers.

The effective circuit area is then calculated by incorporating the idle qubit penalty factor γ, through subtracting the idle qubit count and adding back the idle qubits multiplied by γ, yielding

$$\left(r + \frac{24}{5}w + \frac{1}{2}rw\right) + \beta\left(r + \frac{1}{2}rw\right) - \gamma\left(\frac{24}{5}w - \left(\frac{24}{5}w\left(n + r + \frac{1}{2}rw\right)\right)\right). \quad (11)$$

Hence, (11) derives the effective circuit area for the **1** SE schemes with r stabilizers.

In this work, we utilize CSS codes, which enables to set $r = \frac{s}{2}$ for $X(Z)$-type stabilizers in the **1** SE schemes, and

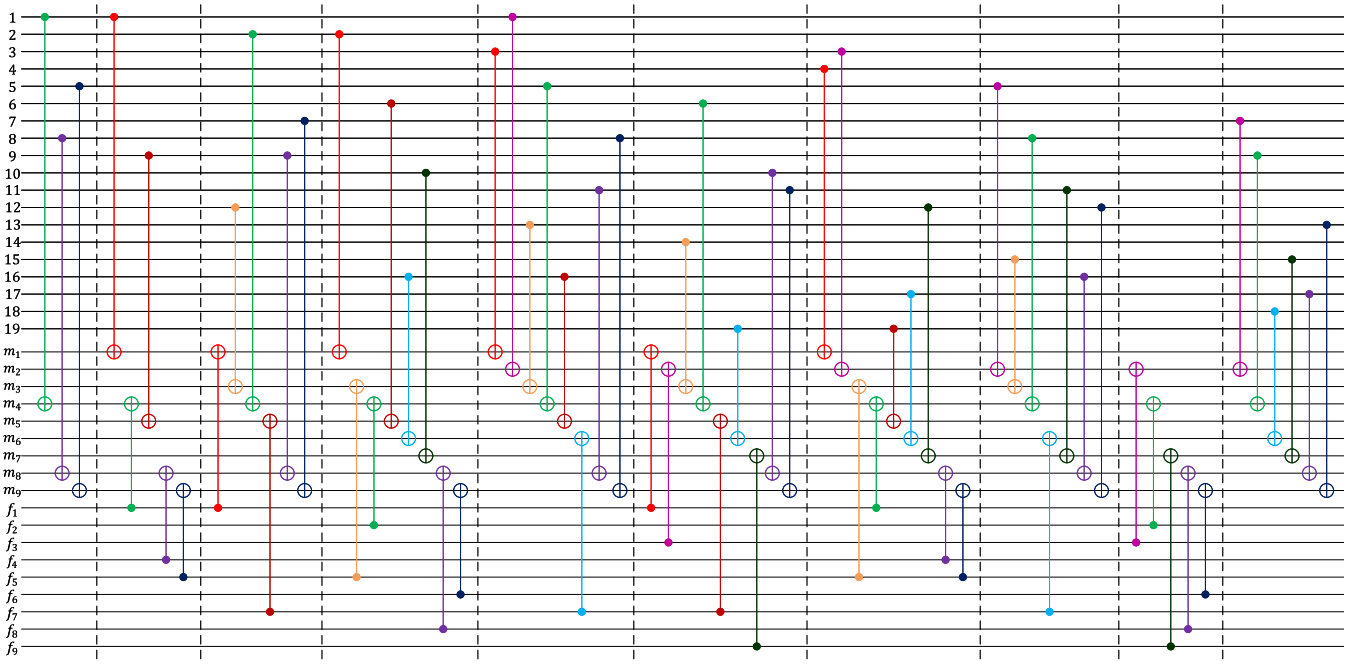

FIGURE 24. Parallel $[2\ \ 2\ \ 2\ \ 1\ \ 1\ \ 1]^T$ SE scheme for the Z-type stabilizers of the $[\![19, 1, 5]\!]$ code. This circuit has a shorter depth than any stage of the [3 3 3] SE scheme in [33], while still measuring all stabilizers.

we have to calculate the effective circuit area for two **1** SE schemes. As a result, the circuit depth, state preparation, and measurement operations are effectively doubled. These are represented as $2 \times (3w)$, $2 \times (\frac{s}{2} + \frac{1}{2} \times \frac{s}{2}w)$, and $2 \times \beta(\frac{s}{2} + \frac{1}{2} \times \frac{s}{2}w)$, respectively. Similarly, the circuit area can be calculated as

$$1.6 \times 2 \times (3w)\left(n + \frac{s}{2} + \frac{1}{2} \times \frac{s}{2}w\right) + 2 \times \left(\frac{s}{2} + \frac{1}{2} \times \frac{s}{2}w\right) + 2\beta\left(\frac{s}{2} + \frac{1}{2} \times \frac{s}{2}w\right). \quad (12)$$

The number of idle qubits is expressed as

$$\frac{48}{5}\left(n + \frac{s}{2} + \frac{sw}{4}\right)w - \frac{48}{5}w. \quad (13)$$

Consequently, the effective circuit area is given by

$$\left(s + \frac{48}{5}w + \frac{1}{2}sw\right) + \beta\left(s + \frac{1}{2}sw\right) - \gamma\left(\frac{48}{5}w - \frac{48}{5}w\left(n + \frac{1}{2}s + \frac{1}{4}sw\right)\right). \quad (14)$$

APPENDIX C
SE SCHEMES FOR THE $[\![17, 1, 5]\!]$ CODE

We provide the detailed circuits for the **1** SE scheme and the $[2\ 2\ 2\ 1\ 1]^T$ flag-sharing SE scheme for the $[\![17, 1, 5]\!]$ code in Figs. 21 and 22, respectively.

APPENDIX D
PARTIAL LOOKUP TABLE FOR THE $[\![5, 1, 3]\!]$ CODE

We provide a partial lookup table in Table 7 for the $[2; 2]$ scheme for the $[\![5, 1, 3]\!]$ code.

APPENDIX E
ADAPTIVE SE PROTOCOL FOR HIGH-DISTANCE CODES

Consider a decision tree designed for high-distance quantum error correction codes. A path in the decision tree is defined as a sequence of nodes that begins at the root node and progresses layer by layer through parent and child nodes, ultimately terminating at a leaf node. A valid path is characterized by the property that only the final leaf node executes the complete raw SE, encapsulating the entire decision-making process from the initial state to the final outcome.

The protocol is as follows.

1) At each path of a decision tree, residual data errors that result in identical generalized error syndromes, which are defined as the measurement outcomes from all nodes in the path except the complete raw SE outcome at the final leaf node, are assigned the same group number.
2) For each group of errors, identify stabilizers in the raw SE stage that produce identical outcomes for all errors in the group. These stabilizers can then be excluded from the final stage of raw SE.

TABLE 7. Lookup Table for the $\begin{bmatrix}2\\2\end{bmatrix}$ SE Scheme for the ⟦5, 1, 3⟧ Code of Fig. 10

Fault	Data X error	Data Z error	$m_{1534}f_{12}$	m'_{1534}	group	Adaptive index
A_{IX}	00101	00000	000101	1011	5	2222
A_{XX}	01101	00000	000101	0111	5	2222
A_{ZX}	00101	01000	000001	1010	2	2222
A_{YX}	01101	01000	000001	0110	2	2222
B_{IX}	10010	00000	001010	0111	8	2222
B_{XX}	10110	00000	001010	1111	8	2222
B_{ZX}	10010	00100	011010	0001	17	2002
B_{YX}	10110	00100	011010	1001	17	2002
C_{IX}	01000	00001	000001	1000	2	2222
C_{XX}	11000	00001	000001	1101	2	2222
C_{ZX}	01000	10001	101001	0010	22	0212
C_{YX}	11000	10001	101001	0111	22	0212
D_{IX}	00001	00000	000101	0011	5	2222
D_{XX}	00101	00000	000101	1011	5	2222
D_{ZX}	00001	00100	000101	0101	5	2222
D_{YX}	00101	00100	000101	1101	5	2222
E_{IX}	10000	00000	000010	0101	3	2202
E_{XX}	10010	00000	001010	0111	8	2222
E_{ZX}	10000	00010	000010	1100	3	2202
E_{YX}	10010	00010	001010	1110	8	2222
F_{IX}	10000	00010	000010	1100	3	2202
F_{XX}	10001	00010	000110	1111	6	1111
F_{ZX}	10000	00011	010010	1000	12	1000
F_{YX}	10001	00011	010110	1011	15	1011
G_{IX}	00000	00010	000010	1001	3	2202
G_{XX}	10000	00010	000010	1100	3	2202
G_{ZX}	00000	10010	100010	0011	20	2222
G_{YX}	10000	10010	100010	0110	20	2222
H_{IX}	00000	00001	000001	0100	2	2222
H_{XX}	01000	00001	000001	1000	2	2222
H_{ZX}	00000	01001	000001	0101	2	2222
H_{YX}	01000	01001	000001	1001	2	2222
a_{XI}	10010	00100	011010	0001	17	2002
a_{XX}	10010	00100	011000	0001	16	2002
a_{XZ}	10010	00100	111010	0001	30	2002
a_{XY}	10010	00100	111000	0001	28	2022
b_{XI}	10000	00011	010010	1000	12	1000
b_{XX}	10000	00011	010000	1000	10	2022
b_{XZ}	10000	00011	111010	1000	30	2002
b_{XY}	10000	00011	111000	1000	28	2022
c_{XI}	10000	00000	000000	0101	1	2222
c_{XX}	10000	00000	000010	0101	3	2202
c_{XZ}	10000	00000	001000	0101	7	2201
c_{XY}	10000	00000	001010	0101	8	2222
d_{XI}	00000	00010	000000	1001	1	2222
d_{XX}	00000	00010	000010	1001	3	2202
d_{XZ}	00000	00010	000000	1001	1	2222
d_{XY}	00000	00010	000010	1001	3	2202
e_{XI}	00101	01000	000001	1010	2	2222
e_{XX}	00101	01000	000000	1010	1	2222
e_{XZ}	00101	01000	010001	1010	11	2212
e_{XY}	00101	01000	010000	1010	10	2022
f_{XI}	01000	10001	101001	0010	22	0212
f_{XX}	01000	10001	101000	0010	21	0222
f_{XZ}	01000	10001	111101	0010	32	0010
f_{XY}	01000	10001	111100	0010	31	0010
g_{XI}	00001	00000	000100	0011	4	0222
g_{XX}	00001	00000	000101	0011	5	2222
g_{XZ}	00001	00000	000000	0011	1	2222
g_{XY}	00001	00000	000001	0011	2	2222
h_{XI}	00000	00001	000000	0100	1	2222
h_{XX}	00000	00001	000001	0100	2	2222
h_{XZ}	00000	00001	000000	0100	1	2222
h_{XY}	00000	00001	000001	0100	2	2222
A_{IY}	00101	00000	010101	1011	14	2221
A_{XY}	01101	00000	010101	0111	14	2221
A_{ZY}	00101	01000	010001	1010	11	2212
A_{YY}	01101	01000	010001	0110	11	2212
B_{IY}	10010	00000	101010	0111	23	2222
B_{XY}	10110	00000	101010	1111	23	2222
B_{ZY}	10010	00100	111010	0001	30	2002
B_{YY}	10110	00100	111010	1001	30	2002
C_{IY}	01000	00001	000101	1000	5	2222
C_{XY}	11000	00001	000101	1101	5	2222
C_{ZY}	01000	10001	101101	0010	25	0212
C_{YY}	11000	10001	101101	0111	25	0212
D_{IY}	00001	00000	010101	0011	14	2221
D_{XY}	00101	00000	010101	1011	14	2221
D_{ZY}	00001	00100	010101	0101	14	2221
D_{YY}	00101	00100	010101	1101	14	2221
E_{IY}	10000	00000	100010	0101	20	2222
E_{XY}	10010	00000	101010	0111	23	2222
E_{ZY}	10000	00010	100010	1100	20	2222
E_{YY}	10010	00010	101010	1110	23	2222
F_{IY}	10000	00010	001010	1100	8	2222
F_{XY}	10001	00010	001110	1111	9	1111
F_{ZY}	10000	00011	011010	1000	17	2002
F_{YY}	10001	00011	011110	1011	18	1011
G_{IY}	00000	00010	001010	1001	8	2222
G_{XY}	10000	00010	001010	1100	8	2222
G_{ZY}	00000	10010	101010	0011	23	2222
G_{YY}	10000	10010	101010	0110	23	2222
H_{IY}	00000	00001	000101	0100	5	2222
H_{XY}	01000	00001	000101	1000	5	2222
H_{ZY}	00000	01001	000101	0101	5	2222
H_{YY}	01000	01001	000101	1001	5	2222
a_{YI}	10010	00100	111010	0001	30	2002
a_{YX}	10010	00100	111000	0001	28	2022
a_{YZ}	10010	00100	011010	0001	17	2002
a_{YY}	10010	00100	011000	0001	16	2002
b_{YI}	10000	00011	011010	1000	17	2002
b_{YX}	10000	00011	011000	1000	16	2002
b_{YZ}	10000	00011	110010	1000	27	1000
b_{YY}	10000	00011	110000	1000	26	1000
c_{YI}	10000	00000	100000	0101	19	0101
c_{YX}	10000	00000	100010	0101	20	2222
c_{YZ}	10000	00000	101000	0101	21	0222
c_{YY}	10000	00000	101010	0101	23	2222
d_{YI}	00000	00010	001000	1001	7	2201
d_{YX}	00000	00010	001010	1001	8	2222
d_{YZ}	00000	00010	001000	1001	7	2201
d_{YY}	00000	00010	001010	1001	8	2222
e_{YI}	00101	01000	010001	1010	11	2212
e_{YX}	00101	01000	010000	1010	10	2022
e_{YZ}	00101	01000	000001	1010	2	2222
e_{YY}	00101	01000	000000	1010	1	2222
f_{YI}	01000	10001	101101	0010	25	0212
f_{YX}	01000	10001	101100	0010	24	0010
f_{YZ}	01000	10001	111001	0010	29	0010
f_{YY}	01000	10001	111000	0010	28	2022
g_{YI}	00001	00000	010100	0011	13	0011
g_{YX}	00001	00000	010101	0011	14	2221
g_{YZ}	00001	00000	010000	0011	10	2022
g_{YY}	00001	00000	010001	0011	11	2212
h_{YI}	00000	00001	000100	0100	4	0222
h_{YX}	00000	00001	000101	0100	5	2222
h_{YZ}	00000	00001	000100	0100	4	0222
h_{YY}	00000	00001	000101	0100	5	2222

The Fault entry letters (A to H and a to h) correspond to the markers in Figure 10, with subscripts indicating the errors applied to qubits in their top-to-bottom circuit order. The "Data X error" and "Data Z error" entries represent the final Pauli X and Z errors, respectively, that manifest in the data block as a result of the specified faults. The entry $m_{1534}f_{12}$ denotes the measurement outcomes of the first round of the $\begin{bmatrix}2\\2\end{bmatrix}$ SE,while the entry m'_{1534} denotes the measurement outcomes of the subsequent round of a raw SE. The entries Data X error and Data Z error that prescribe the necessary error corrections: they indicate which Pauli X and Z errors should be corrected on the corresponding qubits in the data block. The errors with identical measurement outcomes $m_{1534}f_{12}$ are grouped under the same group number in the column "group." The four-digit adaptive index $a_1a_2a_3a_4$ represents the adaptive measurements for m'_{1534} required in the final stage of raw SE. If $a_j = 2$, it indicates that m'_j must be remeasured to distinguish errors within the same group. Conversely, $a_j = 0$ or $a_j = 1$ implies that $m'_j = 0$ or $m'_j = 1$ for all errors in this group, allowing the corresponding measurements to be omitted.

APPENDIX F
SE SCHEMES FOR THE ⟦19, 1, 5⟧ CODE

We provide the detailed circuits for the **1** SE scheme and the $\begin{bmatrix}2&2&2&1&1&1\end{bmatrix}^T$ flag-sharing SE scheme for the ⟦19, 1, 5⟧ code in Figs. 23 and 24, respectively.

REFERENCES

[1] J. Preskill, "Quantum computing in the NISQ era and beyond," *Quantum*, vol. 2, Aug. 2018, Art. no. 79, doi: 10.22331/q-2018-08-06-79.

[2] T. Albash and D. A. Lidar, "Adiabatic quantum computation," *Rev. Modern Phys.*, vol. 90, Jan. 2018, Art. no. 015002, doi: 10.1103/RevModPhys.90.015002.

[3] E. T. Campbell, B. M. Terhal, and C. Vuillot, "Roads towards fault-tolerant universal quantum computation," *Nature*, vol. 549, pp. 172–179, 2017, doi: 10.1038/nature23460.

[4] M. Takita, A. W. Cross, A. D. Córcoles, J. M. Chow, and J. M. Gambetta, "Experimental demonstration of fault-tolerant state preparation with superconducting qubits," *Phys. Rev. Lett.*, vol. 119, Oct. 2017, Art. no. 180501, doi: 10.1103/PhysRevLett.119.180501.

[5] D. Aharonov and M. Ben-Or, "Fault-tolerant quantum computation with constant error," in *Proc. 29th Annu. ACM Symp. Theory Comput.*, 1997, pp. 176–188. [Online]. Available: https://dl.acm.org/doi/pdf/10.1145/258533.258579

[6] P. W. Shor, "Fault-tolerant quantum computation," in *Proc. 37th Annu. Symp. Theory Comput. Sci.*, Los Alamitos, CA, USA, 1996, pp. 56–65, doi: 10.1109/SFCS.1996.548464.

[7] D. P. DiVincenzo and P. W. Shor, "Fault-tolerant error correction with efficient quantum codes," *Phys. Rev. Lett.*, vol. 77, no. 15, pp. 3260–3263, 1996, doi: 10.1103/PhysRevLett.77.3260.

[8] E. Knill, R. Laflamme, and W. Zurek, "Threshold accuracy for quantum computation," 1996, *arXiv:quant-ph/9610011*, doi: 10.48550/arXiv.quant-ph/9610011.

[9] D. Gottesman, "Stabilizer codes and quantum error correction," Ph.D. dissertation, California Inst. Technol., Pasadena, CA, USA, 1997.

[10] J. Preskill, "Reliable quantum computers," *Proc. Roy. Soc. London A*, 1998, vol. 454, pp. 385–410, doi: 10.1098/rspa.1998.0167.

[11] A. M. Steane, "Active stabilization, quantum computation, and quantum state synthesis," *Phys. Rev. Lett.*, vol. 78, pp. 2252–2255, Mar. 1997, doi: 10.1103/PhysRevLett.78.2252.

[12] A. M. Steane, "Efficient fault-tolerant quantum computing," *Nature*, vol. 399, pp. 124–126, 1999, doi: 10.1038/20127.

[13] M. A. Nielsen and C. M. Dawson, "Fault-tolerant quantum computation with cluster states," *Phys. Rev. A*, vol. 71, Apr. 2005, Art. no. 042323, doi: 10.1103/PhysRevA.71.042323.

[14] P. W. Shor, "Scheme for reducing decoherence in quantum computer memory," *Phys. Rev. A*, vol. 52, no. 4, pp. 2493–2496, 1995, doi: 10.1103/PhysRevA.52.R2493.

[15] P. Aliferis, D. Gottesman, and J. Preskill, "Quantum accuracy threshold for concatenated distance-3 codes," *Quantum Inf. Comput.*, vol. 6, pp. 97–165, Mar. 2006, doi: 10.5555/2011665.2011666.

[16] P. Aliferis, D. Gottesman, and J. Preskill, "Accuracy threshold for post-selected quantum computation," *Quantum Inf. Comput.*, vol. 8, no. 3, pp. 181–244, 2008, doi: 10.5555/2011763.2011764.

[17] D. Gottesman, "Fault-tolerant quantum computation with constant overhead," *Quantum Inf. Comput.*, vol. 14, pp. 1338–1372, Nov. 2014, doi: 10.5555/2685179.2685184.

[18] E. Knill, "Quantum computing with realistically noisy devices," *Nature*, vol. 434, pp. 39–44, 2005, doi: 10.1038/nature03350.

[19] C.-Y. Lai, Y.-C. Zheng, and T. A. Brun, "Fault-tolerant preparation of stabilizer states for quantum Calderbank-Shor-Steane codes by classical error-correcting codes," *Phys. Rev. A*, vol. 95, Mar. 2017, Art. no. 032339, doi: 10.1103/PhysRevA.95.032339 .

[20] Y.-C. Zheng, C.-Y. Lai, T. A. Brun, and L.-C. Kwek, "Constant depth fault-tolerant Clifford circuits for multi-qubit large block codes," *Quantum Sci. Tech.*, vol. 5, 2020, Art. no. 045007, doi: 1088/2058-9565/aba34d.

[21] A. Ashikhmin, C.-Y. Lai, and T. A. Brun, "Quantum data-syndrome codes," *IEEE J. Sel. Area. Comm.*, vol. 38, no. 3, pp. 449–462, Mar. 2020, doi: 10.1109/JSAC.2020.2968997.

[22] K.-Y. Kuo, I.-C. Chern, and C.-Y. Lai, "Decoding of quantum data-syndrome codes via belief propagation," in *Proc. IEEE Int. Symp. Inf. Theory*, 2021, pp. 1552–1557, doi: 10.1109/ISIT45174.2021.9518018.

[23] R. Chao and B. W. Reichardt, "Quantum error correction with only two extra qubits," *Phys. Rev. Lett.*, vol. 121, Aug. 2018, Art. no. 050502, doi: 10.1103/PhysRevLett.121.050502.

[24] R. Chao and B. W. Reichardt, "Fault-tolerant quantum computation with few qubits," *npj Quantum Inf.*, vol. 4, Sep. 2018, Art. no. 42, doi: 10.1038/s41534-018-0085-z.

[25] C. Chamberland and M. E. Beverland, "Flag fault-tolerant error correction with arbitrary distance codes," *Quantum*, vol. 2, Feb. 2018, Art. no. 53, doi: 10.22331/q-2018-02-08-53.

[26] N. Delfosse and B. W. Reichardt, "Short Shor-style syndrome sequences," 2020, *arXiv:2008.05051*, doi: 10.48550/arXiv.2008.05051.

[27] T. Tansuwannont, B. Pato, and K. R. Brown, "Adaptive syndrome measurements for Shor-style error correction," *Quantum*, vol. 7, Aug. 2023, Art. no. 1075, doi: 10.22331/q-2023-08-08-1075.

[28] B. Pato, T. Tansuwannont, S. Huang, and K. R. Brown, "Optimization tools for distance-preserving flag fault-tolerant error correction," *PRX Quantum*, vol. 5, May 2024, Art. no. 020336, doi: 10.1103/PRXQuantum.5.020336.

[29] D. Bhatnagar, M. Steinberg, D. Elkouss, C. G. Almudever, and S. Feld, "Low-depth flag-style syndrome extraction for small quantum error-correction codes," in *Proc. IEEE Int. Conf. Quantum Comput. Eng.*, Los Alamitos, CA, USA, Sep. 2023, pp. 63–69, doi: 10.1109/QCE57702.2023.00016.

[30] C. Du, Z. Ma, Y. Liu, H. Wang, and Q. Duan, "Low-depth flagged syndrome extraction for Calderbank-Shor-Steane codes of distance 3," *IEEE Trans. Inf. Theory*, vol. 70, no. 9, pp. 6326–6349, Sep. 2024, doi: 10.1109/TIT.2024.3425853.

[31] B. W. Reichardt, "Fault-tolerant quantum error correction for Steane's seven-qubit color code with few or no extra qubits," *Quantum Sci. Tech.*, vol. 6, Dec. 2020, Art. no. 015007, doi: 10.1088/2058-9565/abc6f4.

[32] P.-H. Liou and C.-Y. Lai, "Parallel syndrome extraction with shared flag qubits for Calderbank-Shor-Steane codes of distance three," *Phys. Rev. A*, vol. 107, Feb. 2023, Art. no. 022614, doi: 10.1103/PhysRevA.107.022614.

[33] C. Du, Z. Ma, Y. Liu, Q. Duan, and Y. Fei, "Parallel flagged fault-tolerant error correction for stabilizer codes of distance 5," *Adv. Quantum Technol.*, vol. 7, no. 5, 2024, Art. no. 2300291, doi: 10.1002/qute.202300291.

[34] H. Bombin and M. A. Martin-Delgado, "Topological quantum distillation," *Phys. Rev. Lett.*, vol. 97, Oct. 2006, Art. no. 180501, doi: 10.1103/PhysRevLett.97.180501.

[35] H. Bombin and M. A. Martin-Delgado, "Optimal resources for topological two-dimensional stabilizer codes: Comparative study," *Phys. Rev. A*, vol. 76, Jul. 2007, Art. no. 012305, doi: 10.1103/PhysRevA.76.012305.

[36] R. Laflamme, C. Miquel, J. P. Paz, and W. H. Zurek, "Perfect quantum error correcting codes," *Phys. Rev. Lett.*, vol. 77, no. 1, pp. 198–201, 1996, doi: 10.1103/PhysRevLett.77.198.

[37] S. Heußen et al., "Strategies for a practical advantage of fault-tolerant circuit design in noisy trapped-ion quantum computers," *Phys. Rev. A*, vol. 107, Apr. 2023, Art. no. 042422, doi: 10.1103/PhysRevA.107.042422.

[38] L. Egan et al., "Fault-tolerant control of an error-corrected qubit," *Nature*, vol. 598, pp. 281–286, Oct. 2021, doi: 10.1038/s41586-021-03928-y.

[39] C. Ryan-Anderson et al., "Implementing fault-tolerant entangling gates on the five-qubit code and the color code," 2022, *arXiv:2208.01863*, doi: 10.48550/arXiv.2208.01863.

[40] M. H. Abobeih et al., "Fault-tolerant operation of a logical qubit in a diamond quantum processor," *Nature*, vol. 606, pp. 884–889, Jun. 2022, doi: 10.1038/s41586-022-04819-6.

[41] S. J. Evered et al., "High-fidelity parallel entangling gates on a neutral-atom quantum computer," *Nature*, vol. 622, pp. 268–272, Oct. 2023, doi: 10.1038/s41586-023-06481-y.

[42] D. Bluvstein et al., "Logical quantum processor based on reconfigurable atom arrays," *Nature*, vol. 626, pp. 58–65, Feb. 2024, doi: 10.1038/s41586-023-06927-3.

[43] R. Acharya et al., "Suppressing quantum errors by scaling a surface code logical qubit," *Nature*, vol. 614, pp. 676–681, Feb. 2023, doi: 10.1038/s41586-022-05434-1.

[44] R. Acharya et al., "Quantum error correction below the surface code threshold," *Nature*, vol. 368, pp. 920–926, Dec. 2024, doi: 10.1038/s41586-024-08449-y.

[45] C. Ryan-Anderson et al., "Realization of real-time fault-tolerant quantum error correction," *Phys. Rev. X*, vol. 11, Dec. 2021, Art. no. 041058, doi: 10.1103/PhysRevX.11.041058.

[46] J. Hilder et al., "Fault-tolerant parity readout on a shuttling-based trapped-ion quantum computer," *Phys. Rev. X*, vol. 12, Feb. 2022, Art. no. 011032, doi: 10.1103/PhysRevX.12.011032.

[47] A. R. Calderbank and P. W. Shor, "Good quantum error-correcting codes exist," *Phys. Rev. A*, vol. 54, no. 2, pp. 1098–1105, 1996, doi: 10.1103/PhysRevA.54.1098.

[48] A. M. Steane, "Error correcting codes in quantum theory," *Phys. Rev. Lett.*, vol. 77, no. 5, pp. 793–797, 1996, doi: 10.1103/PhysRevLett.77.793.

[49] D. Gottesman, "An introduction to quantum error correction and fault-tolerant quantum computation," in *Quantum Information Science and Its Contributions to Mathematics* (Proceedings of Symposia in Applied Mathematics Series), vol. 68. Providence, RI, USA: Amer. Math. Soc., 2010, pp. 13–58. [Online]. Available: https://mathscinet.ams.org/mathscinet/relay-station?mr=2762145

[50] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, "Topological quantum memory," *J. Math. Phys.*, vol. 43, no. 9, pp. 4452–4505, 2002, doi: 10.1063/1.1499754.

[51] C. Chamberland, A. Kubica, T. J. Yoder, and G. Zhu, "Triangular color codes on trivalent graphs with flag qubits," *New J. Phys.*, vol. 22, Feb. 2020, Art. no. 023019, doi: 10.1088/1367-2630/ab68fd.

[52] K.-Y. Kuo and C.-Y. Lai, "Fault-tolerant belief propagation for practical quantum memory," 2024, *arXiv:2409.18689*, doi: 10.48550/arXiv.2409.18689.

[53] R. Chao and B. W. Reichardt, "Flag fault-tolerant error correction for any stabilizer code," *PRX Quantum*, vol. 1, Sep. 2020, Art. no. 010302, doi: 10.1103/PRXQuantum.1.010302.

[54] P. Prabhu and B. W. Reichardt, "Fault-tolerant syndrome extraction and cat state preparation with fewer qubits," *Quantum*, vol. 7, Oct. 2023, Art. no. 1154, doi: 10.22331/q-2023-10-24-1154.

[55] S. Veroni, M. Müller, and G. Giudice, "Optimized measurement-free and fault-tolerant quantum error correction for neutral atoms," *Phys. Rev. Res.*, vol. 6, Dec. 2024, Art. no. 043253, doi: 10.1103/PhysRevResearch.6.043253.

[56] M. Li, D. Miller, and K. R. Brown, "Direct measurement of Bacon-Shor code stabilizers," *Phys. Rev. A*, vol. 98, Nov. 2018, Art. no. 050301, doi: 10.1103/PhysRevA.98.050301.

[57] P.-H. Liou and C.-Y. Lai. Accessed: Jan. 18, 2025. [Online]. Available: https://github.com/slorqwq/Reduce_CA