Performance Analysis of Quantum CSS Error-Correcting Codes via MacWilliams Identities

Diego Forlivesi, Lorenzo Valentini, and Marco Chiani

Department of Electrical, Electronic, and Information Engineering "Guglielmo Marconi" and CNIT/WiLab, University of Bologna, 40136 Bologna, Italy.

arXiv:2305.01301v4 [quant-ph] 11 Dec 2025

We analyze the performance of quantum stabilizer codes, one of the most important classes for practical implementations, on both symmetric and asymmetric quantum channels. To this aim, we first derive the weight enumerator (WE) for the undetectable errors based on the quantum MacWilliams identities. The WE is then used to evaluate tight upper bounds on the error rate of CSS quantum codes with minimum weight decoding. For surface codes we also derive a simple closed form expression of the bounds over the depolarizing channel. We introduce a novel approach that combines the knowledge of WE with a logical operator analysis, allowing the derivation of the exact asymptotic error rate for short codes. For example, on a depolarizing channel with physical error rate $\rho \to 0$, the logical error rate ρ_{L} is asymptotically $\rho_{\mathrm{L}} \approx 16\rho^2$ for the $[[9,1,3]]$ Shor code, $\rho_{\mathrm{L}} \approx 16.3\rho^2$ for the $[[7,1,3]]$ Steane code, $\rho_{\mathrm{L}} \approx 18.7\rho^2$ for the $[[13,1,3]]$ surface code, and $\rho_{\mathrm{L}} \approx 149.3\rho^3$ for the $[[41,1,5]]$ surface code. For larger codes our bound provides $\rho_{\mathrm{L}} \approx 1215\rho^4$ and $\rho_{\mathrm{L}} \approx 663\rho^5$ for the $[[85,1,7]]$ and the $[[181,1,10]]$ surface codes, respectively. Finally, we extend our analysis to include realistic, noisy syndrome extraction circuits by modeling error propagation throughout gadgets. This enables estimation of logical error rates under faulty measurements. The performance analysis serves as a design tool for developing fault-tolerant quantum systems by guiding the selection of quantum codes based on their error correction capability. Additionally, it offers a novel perspective on quantum degeneracy, showing it represents the fraction of non-correctable error patterns shared by multiple logical operators.

Diego Forlivesi: diego.forlivesi2@unibo.it
Lorenzo Valentini: lorenzo.valentini13@unibo.it
Marco Chiani: marco.chiani@unibo.it

1 Introduction

The exploitation of the unique features of quantum mechanics has opened new perspectives on how we can sense, process, and communicate information [1–6]. From an engineering point of view, there are many challenges to solve, calling for both theoretical and experimental research studies. The aim is to progress towards the already known possible applications of quantum information technologies, as well as those currently still unforeseen, that will arise when practical implementations become available. One of the main challenges is how to deal with the noise caused by unwanted interaction of the quantum information with the environment [7–13]. Quantum error correcting codes, where a redundant representation of quantum states protects from certain types of errors, are therefore of paramount importance for quantum computation, quantum memories, and quantum communication systems [14–25].

Quantum error correction is made difficult by the laws of quantum mechanics which imply that qubits cannot be copied or measured without perturbing state superposition. Moreover, there is a continuum of errors that could occur on a qubit. However, it has been shown that to correct an arbitrary qubit error, assuming it remains within the computational space (i.e., neglecting leakage), it is sufficient to consider error correction over the discrete set of Pauli operators [10, 12, 15]. Indeed, assume that a generic continuous phase rotation has occurred on a qubit. Applying the standard syndrome-extraction circuit, we first interact with ancilla qubits to determine the error

syndrome associated with these errors. Next, we measure the ancilla qubits. This measurement collapses the superposition, yielding one of two results: either no error has occurred, or a Pauli error has been identified [9,15]. Thus, we can consider in general a Pauli channel, commonly used in literature for analyzing stabilzer-based quantum error correcting code (QECC) [10,12,15,26]. This channel introduces qubit Pauli errors $\boldsymbol{X}$, $\boldsymbol{Y}$, and $\boldsymbol{Z}$ with probabilities p_{X}, p_{Y}, and p_{Z}, respectively, and leaving the qubit intact with probability $1-\rho$, where $\rho = p_{\mathrm{X}} + p_{\mathrm{Y}} + p_{\mathrm{Z}}$. A special case of this model is the so-called depolarizing (or symmetric) channel for which $p_{\mathrm{X}} = p_{\mathrm{Y}} = p_{\mathrm{Z}} = \rho/3$. Quantum error-correcting codes for this channel are thus designed to protect against equiprobable Pauli errors [7, 8, 12, 14, 27]. However, depending on the technology adopted for the system implementation, the different types of Pauli error can have different probabilities of occurrence, leading to polarizing (or asymmetric) quantum channels [28–31]. Let us now consider channels that arise in the study of decoherence in concrete physical systems, specifically the combined *amplitude damping and dephasing channels*. Key parameters for the noise process underlying this channel are the relaxation time T_1 and dephasing time T_2. Using a technique called twirling, often employed in quantum information theory, it is possible to transform this channel into the corresponding Pauli channel by conjugating the channel with Pauli matrices and averaging the results [29,32–34]. The significance of the twirling method lies in the fact that a correctable code for the twirled channel will also be a correctable code for the original channel [35–37]. For instance, given a combined amplitude damping and dephasing channel, the associated Pauli-twirled channel has $p_{\mathrm{X}} = p_{\mathrm{Z}} = (1 - e^{-\frac{t}{T_1}})/4$ and $p_{\mathrm{Z}} = (1 - 2p_{\mathrm{X}} - e^{-\frac{t}{T_2}})/2$. In this context, the proposed analysis can be easily extended to more realistic channel models.

In this paper we provide an analytical evaluation of the performance of generic stabilizer codes, including surface codes [9, 12, 26, 38–43]. These codes can be interpreted as structured realizations of quantum low-density parity-check (QLDPC) codes, characterized by low-weight stabilizer generators and a planar architectural layout. Stabilizer codes are indeed the most important class of QECCs for practical implementations. Despite their importance, the theoretical performance of these codes has been investigated in the literature only partially, and mainly in terms of accuracy threshold over symmetric channels for fault-tolerant quantum computing [9, 12, 26, 38, 44–46]. We propose a framework for the performance investigation of stabilizer codes by means of the quantum MacWilliams identities. In [47, 48] these identities have been employed to derive theoretical bounds for quantum error detection. Here, we exploit the undetectable errors weight enumerator to provide upper bounds on the logical error rate for Calderbank Shor Steane (CSS) codes [49, 50]. Moreover, we develop a logical operator analysis leading to exact expressions for the logical error rates, assuming complete decoders (decoders that always attempt to correct the error). Specifically, we analyze minimum weight (MW) decoding, which finds the lowest weight error consistent with the syndrome [51–53]. This choice is driven by practical considerations related to the complexity of the decoder [54, 55]. In particular, for surface codes, an instance of QLDPC codes, the decoder is typically realized with the help of the blossom algorithm for finding a minimum weight perfect matching (MWPM) in a graph [56–59]. Although maximum likelihood (ML) decoders exist, such as the matrix product state (MPS) decoder [45], their higher latency introduces decoherence effects that outweigh the benefits gained from improved decoding performance. Since MW decoding is suboptimal with respect to ML decoding, our upper bounds remain applicable; however, their tightness may be limited for codes with high degeneracy. For non-QLDPC codes, where degeneracy could be difficult to assess, the tightness of the bounds cannot be guaranteed. Currently, long non-QLDPC codes are of little practical use owing to their implementation complexity, which further justifies the focus of our analysis.

The performance analysis presented in this work is conducted for both symmetric and asymmetric models of quantum channel errors. In practical quantum systems, however, syndrome extraction is a critical yet error-prone component of quantum error correction. Measurements are inherently noisy and typically require repetition to ensure reliability. Additionally, faults during extraction can propagate, causing high-weight correlated errors [15]. To address these

challenges, we introduce a framework that models the full syndrome extraction process, incorporating gate-specific noise and measurement imperfections. This enables the estimation of logical error rates under realistic circuit-level noise assumptions. Our analysis provides a practical valuable tool for the development of fault-tolerant quantum systems, helping guide the selection of specific quantum codes based on their error correction capability.

Furthermore, we provide advancements in the understanding of quantum degeneracy. Quantum degeneracy typically refers to the non-uniqueness of the error-correcting syndrome, meaning that different errors can produce the same syndrome. Since degeneracy is related to physical errors and has no direct counterpart in classical information theory, it is often challenging to study, especially in the construction of quantum codes. Here, we offer a completely new perspective by showing that the degeneracy of a quantum code represents the fraction of error patterns with weight greater than $t = \lfloor (d-1)/2 \rfloor$ that are shared by more than one logical operator. This characteristic depends on the structure of the code and can therefore serve as a useful guideline in the design of new quantum codes.

The key contributions of the paper can be summarized as follows:

- we derive the weight enumerator for the undetectable errors $L(z)$ of arbitrary stabilizer codes via MacWilliams identities;
- we derive theoretical upper bounds for the error correction capability of CSS stabilizer codes;
- we derive closed form expressions for the $L(z)$ coefficients which significantly impact the performance of surface codes for any code distance;
- we derive the exact performance of stabilizer codes under MW decoding, including surface codes under MWPM decoding, over symmetric and asymmetric channels;
- we extend the analysis to include realistic, noisy syndrome extraction circuits, providing a method for estimating logical error rates under circuit-level noise models;
- we introduce a novel perspective on quantum degeneracy, analyzing its influence on the error correction capability of a quantum code.

As relevant examples, we give the expressions for the performance of the Shor code, the Steane code, and of surface codes of arbitrary size.

This paper is organized as follows. Section 2 introduces preliminary concepts and models together with some background material. Sections 2.3 and 3 provide the analytical investigation of QECC with bounded distance, and with complete decoding. In Section 4 we derive the weight enumerator (WE) for the undetectable errors from MacWilliams identities and apply it to the evaluation of the logical error rate of arbitrary stabilizer codes. In Section 5, we analyze the impact of noisy syndrome measurement extraction circuits. Numerical results are discussed in Section 6.

2 Preliminaries and Background

2.1 Quantum Stabilizer Codes

A qubit is an element of the two-dimensional Hilbert space $\mathcal{H}^2$, with basis $|0\rangle$ and $|1\rangle$ [12]. The Pauli operators $\boldsymbol{I}, \boldsymbol{X}, \boldsymbol{Z}$, and $\boldsymbol{Y}$, are defined by $\boldsymbol{I}|a\rangle = |a\rangle$, $\boldsymbol{X}|a\rangle = |a \oplus 1\rangle$, $\boldsymbol{Z}|a\rangle = (-1)^a|a\rangle$, and $\boldsymbol{Y}|a\rangle = i(-1)^a|a \oplus 1\rangle$ for $a \in \{0,1\}$. These operators either commute (e.g. $\boldsymbol{IX} = \boldsymbol{XI}$) or anticommute (e.g. $\boldsymbol{XZ} = -\boldsymbol{ZX}$) with each other. Also, apart from an overall factor $\pm 1, \pm i$, the composition of two Pauli produces another Pauli (e.g. $\boldsymbol{XY} = i\boldsymbol{Z}$). Thus, all the Pauli operators, together with multiplicative factors $\pm 1, \pm i$ constitute a group, indicated as $\mathcal{G}_1$. Similarly, all Pauli operators on n qubits together with multiplicative factors $\pm 1, \pm i$ form the $\mathcal{G}_n$ Pauli group [12,15]. We indicate with $[[n,k,d]]$ a QECC with minimum distance d, that encodes k information qubits $|\varphi\rangle$ (called logical qubits) into a codeword of n qubits $|\psi\rangle$ (called data or physical qubits), allowing the decoder to correct all patterns up to $t = \lfloor (d-1)/2 \rfloor$ errors (and some patterns of more errors). To simplify our analysis, we consistently adopt the assumption that d is an odd number. The codewords will be assumed equiprobable in the following. Using the stabilizer formalism, we start by choosing $n-k$ independent and commuting operators $\boldsymbol{G}_i \in \mathcal{G}_n$, called stabilizer generators (or simply generators). The subgroup of

$\mathcal{G}_n$ generated by all combinations of the $\boldsymbol{G}_i \in \mathcal{G}_n$ is a stabilizer, indicated as $\mathcal{S}$. The code $\mathcal{C}$ is the set of quantum states $|\psi\rangle$ stabilized by $\mathcal{S}$, i.e., satisfying $\boldsymbol{S}|\psi\rangle = |\psi\rangle \; \forall \boldsymbol{S} \in \mathcal{S}$, or, equivalently, $\boldsymbol{G}_i|\psi\rangle = |\psi\rangle, i = 1, 2, \ldots, n-k$. For a subgroup $\mathcal{H}$ of a group $\mathcal{G}$ we indicate with $\mathcal{N}(\mathcal{H})$ the normalizer, and with $\mathcal{C}(\mathcal{H})$ the centralizer. The centralizer and the normalizer of the stabilizer $\mathcal{S}$ are coincident, $\mathcal{N}(\mathcal{S}) = C(\mathcal{S})$.

Assume a codeword $|\psi\rangle \in \mathcal{C}$ is affected by a channel error. Measuring the received state according to the generators $\boldsymbol{G}_i$ with the aid of ancilla qubits, the error collapses on a discrete set of possibilities represented by the Pauli operators $\boldsymbol{E} \in \mathcal{G}_n$ [15]. We call this $\boldsymbol{E}$ a Pauli error. The weight of an error $\boldsymbol{E} \in \mathcal{G}_n$ is the number of single qubits Pauli operators which are not equal to the identity. For example, the error $\boldsymbol{E} = \boldsymbol{X}_2\boldsymbol{Y}_3$ has weight two, with $\boldsymbol{X}$ occurred on the second qubit and $\boldsymbol{Y}$ occurred on the third qubit (we implicitly mean that the others qubits see the Pauli identity operator). The measurement procedure over the ancilla qubits results in a quantum error syndrome $\boldsymbol{s}(\boldsymbol{E}) = (s_1, s_2, \ldots, s_{n-k})$, with each $s_i = 0$ or 1 depending on $\boldsymbol{E}$ commuting or anticommuting with $\boldsymbol{G}_i$, respectively [15]. In the following, we will refer to ancillas measuring $s_i = 1$ as *defects*. Note that an error $\boldsymbol{E} \in \mathcal{S}$ has no effect on a codeword since in this case $\boldsymbol{E}|\psi\rangle = |\psi\rangle$. A minimum weight decoder will infer the most probable error $\hat{\boldsymbol{E}} \in \mathcal{G}_n$ compatible with the measured syndrome. Within the category of stabilizer codes, among the earliest introduced quantum codes were the $[[9, 1, 3]]$ Shor code [7], with generators

$$\begin{aligned}
&\boldsymbol{G}_1 = \mathbf{Z}_1\mathbf{Z}_2 && \boldsymbol{G}_2 = \mathbf{Z}_2\mathbf{Z}_3\\
&\boldsymbol{G}_3 = \mathbf{Z}_4\mathbf{Z}_5 && \boldsymbol{G}_4 = \mathbf{Z}_5\mathbf{Z}_6\\
&\boldsymbol{G}_5 = \mathbf{Z}_7\mathbf{Z}_8 && \boldsymbol{G}_6 = \mathbf{Z}_8\mathbf{Z}_9\\
&\boldsymbol{G}_7 = \mathbf{X}_1\mathbf{X}_2\mathbf{X}_3\mathbf{X}_4\mathbf{X}_5\mathbf{X}_6\\
&\boldsymbol{G}_8 = \mathbf{X}_4\mathbf{X}_5\mathbf{X}_6\mathbf{X}_7\mathbf{X}_8\mathbf{X}_9,
\end{aligned}$$

the $[[7, 1, 3]]$ Steane code [60], with generators

$$\begin{aligned}
&\boldsymbol{G}_1 = \mathbf{X}_1\mathbf{X}_3\mathbf{X}_5\mathbf{X}_7 && \boldsymbol{G}_2 = \mathbf{X}_2\mathbf{X}_3\mathbf{X}_6\mathbf{X}_7\\
&\boldsymbol{G}_3 = \mathbf{X}_4\mathbf{X}_5\mathbf{X}_6\mathbf{X}_7 && \boldsymbol{G}_4 = \mathbf{Z}_1\mathbf{Z}_3\mathbf{Z}_5\mathbf{Z}_7\\
&\boldsymbol{G}_5 = \mathbf{Z}_2\mathbf{Z}_3\mathbf{Z}_6\mathbf{Z}_7 && \boldsymbol{G}_6 = \mathbf{Z}_4\mathbf{Z}_5\mathbf{Z}_6\mathbf{Z}_7,
\end{aligned}$$

and the $[[5, 1, 3]]$ perfect code [61] with generators

$$\begin{aligned}
&\boldsymbol{G}_1 = \mathbf{X}_1\mathbf{Z}_2\mathbf{Z}_3\mathbf{X}_4 && \boldsymbol{G}_2 = \mathbf{X}_2\mathbf{Z}_3\mathbf{Z}_4\mathbf{X}_5\\
&\boldsymbol{G}_3 = \mathbf{X}_1\mathbf{X}_3\mathbf{Z}_4\mathbf{Z}_5 && \boldsymbol{G}_4 = \mathbf{Z}_1\mathbf{X}_2\mathbf{X}_4\mathbf{Z}_5.
\end{aligned}$$

A possible channel model is one characterized by errors occurring independently and with the same statistic on the individual qubits of each codeword. In this model, the error on each physical qubit can be $\boldsymbol{X}$, $\boldsymbol{Z}$ or $\boldsymbol{Y}$ with probabilities p_{X}, p_{Z}, and p_{Y}, respectively. The probability of a generic error on a physical qubit is $\rho = p_{\mathrm{X}} + p_{\mathrm{Z}} + p_{\mathrm{Y}}$. Two important models are the *depolarizing channel* where $p_{\mathrm{X}} = p_{\mathrm{Z}} = p_{\mathrm{Y}} = \rho/3$, and the *phase flip channel* where $\rho = p_{\mathrm{Z}}$, $p_{\mathrm{X}} = p_{\mathrm{Y}} = 0$. We will also consider more general asymmetric channels with the constraint $p_{\mathrm{X}} = p_{\mathrm{Y}}$, therefore completely characterized by the bias parameter $A = 2p_{\mathrm{Z}}/(\rho - p_{\mathrm{Z}})$. Note that for $A = 1$ we have the depolarizing channel, and for $A \to \infty$ we have the phase flip channel.

In the following, we will also adopt the notation $[[n, k, d_{\mathrm{X}}/d_{\mathrm{Z}}]]$ for asymmetric codes able to correct all patterns up to $t_{\mathrm{X}} = \lfloor (d_{\mathrm{X}} - 1)/2 \rfloor$ Pauli $\boldsymbol{X}$ errors and $t_{\mathrm{Z}} = \lfloor (d_{\mathrm{Z}} - 1)/2 \rfloor$ Pauli $\boldsymbol{Z}$ errors.

2.2 Quantum Topological Codes

One of the most important families of stabilizer QECC is that of *topological* codes. The general design principle behind these codes is that they are built by patching together repeated elements. Using this kind of approach, they can be easily scaled in size in order to increase the distance of the code, still guaranteeing commutativity of the generators. With regard to the actual implementation, these codes have a great intrinsic advantage. In fact, they require only nearest-neighbor interactions [18]. The most important codes within this category are the *surface* codes, in which all the check operators are local and the qubits are arranged on planar sheets. Specifically, the following will focus on unrotated surface codes. Each stabilizer is associated either with one of the sites or one of the cells that are called "plaquettes" [62]. The stabilizer's generators in the interior are four-qubits plaquette or site operators, while the ones at the boundaries are three-qubits operators. Along a plaquette or "rough" edge, each generator is a three-qubits operator $\boldsymbol{Z}^{\otimes 3}$, while, along a site edge or "smooth" edge, each generator is a three-qubits operator $\boldsymbol{X}^{\otimes 3}$. The entire lattice is able to encode one logical qubit ($k = 1$). It can be shown that in a code with distance d, the lattice has $d^2 + (d-1)^2$ physical qubits [57]. For example, two equivalent graphical representations of the resulting lattice for the $[[13, 1, 3]]$ surface code are shown in Fig. 1.

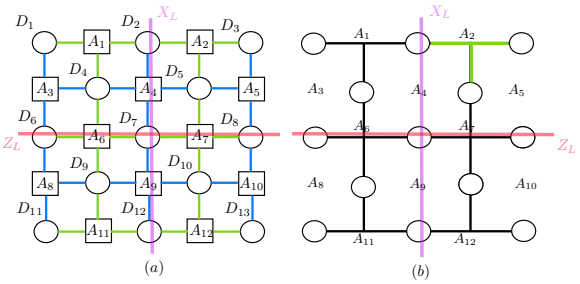

Figure 1: $[[13, 1, 3]]$ Surface code. (a) Actual lattice with data qubits D (circles) and ancillae A (squares). (b) Simplified representation with $\boldsymbol{X}$ generators corresponding to sites and $\boldsymbol{Z}$ generators to plaquettes. A smooth edge (A_2) and a rough edge (A_{10}) are depicted in green and blue, respectively. Examples of logical operators are drawn on the lattice.

For this surface code, the generators are

$$\begin{aligned}
&\boldsymbol{G}_1 = \mathbf{X}_1\mathbf{X}_2\mathbf{X}_4 \quad && \boldsymbol{G}_2 = \mathbf{X}_2\mathbf{X}_3\mathbf{X}_5\\
&\boldsymbol{G}_3 = \mathbf{Z}_1\mathbf{Z}_4\mathbf{Z}_6 && \boldsymbol{G}_4 = \mathbf{Z}_2\mathbf{Z}_4\mathbf{Z}_5\mathbf{Z}_7\\
&\boldsymbol{G}_5 = \mathbf{Z}_3\mathbf{Z}_5\mathbf{Z}_8 &&\\
&\boldsymbol{G}_6 = \mathbf{X}_4\mathbf{X}_6\mathbf{X}_7\mathbf{X}_9 && \boldsymbol{G}_7 = \mathbf{X}_5\mathbf{X}_7\mathbf{X}_8\mathbf{X}_{10}\\
&\boldsymbol{G}_8 = \mathbf{Z}_6\mathbf{Z}_9\mathbf{Z}_{11} && \boldsymbol{G}_9 = \mathbf{Z}_7\mathbf{Z}_9\mathbf{Z}_{10}\mathbf{Z}_{12}\\
&\boldsymbol{G}_{10} = \mathbf{Z}_8\mathbf{Z}_{10}\mathbf{Z}_{13} &&\\
&\boldsymbol{G}_{11} = \mathbf{X}_9\mathbf{X}_{11}\mathbf{X}_{12} && \boldsymbol{G}_{12} = \mathbf{X}_{10}\mathbf{X}_{12}\mathbf{X}_{13}\,.
\end{aligned}$$

The logical logical $\boldsymbol{Z}_L$ operator can be chosen as a tensor product of $\boldsymbol{Z}$'s acting on a chain of qubits running from a rough edge to the one at the opposite side of the lattice. Similarly, the $\boldsymbol{X}_L$ logical operator will be the tensor product of $\boldsymbol{X}$'s acting on a chain running from a smooth link to the correspondent one at the other side of the dual lattice.

Another important feature is that a simple complete decoder exists for surface codes: the MWPM [58]. This decoder connects pairs of defects in the shortest way. Moreover, any set of $\boldsymbol{Z}$ ($\boldsymbol{X}$) operators which form a closed loop on the edges of the lattice is contained in the stabilizer. Hence, if the correction operator applied by the decoder together with the channel error closes a loop, the error is correctly recovered. As a consequence, surface codes can also easily correct a large number of errors with a weight larger than $t = \lfloor (d-1)/2 \rfloor$.

In general, a decoding error occurs every time that the concatenation of actual channel error and correction operator realizes a logical operator, so the whole chain operator crosses the lattice from boundary to boundary, realizing a logical operator. For instance, consider a two qubits error $\boldsymbol{Z}_2\boldsymbol{Z}_3$ for the code in Fig. 1. In this scenario, the ancilla qubit A_1 is the only one that changes its state, hence the decoder assumes (wrongly) an error $\boldsymbol{Z}$ on data qubit D_1. It can be easily noticed that the whole operator applied to the lattice represents the logical $\boldsymbol{Z}_L$.

Rectangular surface codes, with differing horizontal and vertical dimensions, can be effectively employed in asymmetric channels. For instance, in hardware implementations, $\boldsymbol{Z}$ errors tend to occur more frequently [28,31,63]. In such scenarios, the $[[23, 1, 3/5]]$ surface code is a viable option, where the horizontal direction is two qubits longer than the vertical one. This configuration results in the logical $\boldsymbol{Z}_L$ operator being a chain of 5 qubits, in contrast to the $[[13, 1, 3]]$ code, where it spans only 3 qubits. Despite requiring a lattice of 23 qubits, this code has a distance of $d_Z = 5$, enabling it to correct weight two $\boldsymbol{Z}$ errors.

2.3 Quantum Codes with Bounded Distance Decoding

The codeword error probability, ρ_{L}, also called logical error probability, is defined as the probability that the decoder does not correct the errors introduced by the quantum channel.

Let us first assume an $[[n, k, d]]$ QECC together with a decoder which corrects up to $t = \lfloor (d-1)/2 \rfloor$ generic errors (i.e., $\boldsymbol{X}$, $\boldsymbol{Z}$, or $\boldsymbol{Y}$) per codeword, and no others. For this bounded distance (BD) decoder, the logical error probability is simply

$$\rho_{\mathrm{L}} = 1 - \sum_{j=0}^{t} \binom{n}{j} \rho^j \, (1-\rho)^{n-j} \tag{1}$$

that, for $\rho \ll 1$, can be approximated as

$$\rho_{\mathrm{L}} \simeq \binom{n}{t+1} \rho^{t+1}\,. \tag{2}$$

We can see that the slope in the log-log plot of the logical error probability ρ_{L} vs. the physical error probability ρ is $t+1$.

The error probability analysis has been recently generalized to asymmetric QECCs assuming a decoder able to correct up to e_{g} generic errors plus up to e_{Z} Pauli $\boldsymbol{Z}$ errors per codeword, and no others. In this case, weighting each pattern with the corresponding probability of occurrence, the bounded distance decoding error rate

(1) over an asymmetric quantum channel with arbitrary p_{X}, p_{Y}, and p_{Z} becomes [31]

$$\rho_{\mathrm{L}} = 1 - \sum_{j=0}^{e_{\mathrm{g}}+e_{\mathrm{Z}}} \binom{n}{j} (1-\rho)^{n-j} \times \sum_{i=(j-e_{\mathrm{g}})^+}^{j} \binom{j}{i} p_{\mathrm{Z}}^{i} \, (\rho - p_{\mathrm{Z}})^{j-i} \tag{3}$$

where $(x)^+ = \max\{x, 0\}$. For a channel with asymmetry parameter $A = 2p_{\mathrm{Z}}/(\rho - p_{\mathrm{Z}})$, noting that $(j-e_{\mathrm{g}})^+ = 0$ if $j \leq e_{\mathrm{g}}$ and using the binomial theorem, the expression in (3) can be rewritten

$$\rho_{\mathrm{L}} = 1 - \sum_{j=0}^{e_{\mathrm{g}}+e_{\mathrm{Z}}} \alpha_j \binom{n}{j} \rho^j (1-\rho)^{n-j} \tag{4}$$

where

$$\alpha_j = \begin{cases} 1 & \text{if } j \leq e_{\mathrm{g}} \\ \left(\frac{2}{A+2}\right)^j \sum_{i=j-e_{\mathrm{g}}}^{j} \binom{j}{i} \left(\frac{A}{2}\right)^i & \text{otherwise.} \end{cases} \tag{5}$$

Note that, when the code is symmetric $e_{\mathrm{Z}} = 0$, and (4) reduces to (1).

Similarly to the approximation done in (2) derived from (1) when $\rho \ll 1$, considering the most significant terms in (4), the asymptotic slope analysis can be extended to asymmetric codes with $e_{\mathrm{Z}} \geqslant 1$ as

$$\rho_{\mathrm{L}} \simeq \begin{cases} \binom{n}{e_{\mathrm{g}}+1} \left(\frac{2\rho}{A+2}\right)^{e_{\mathrm{g}}+1} & 1 \leq A < \infty \\ \binom{n}{e_{\mathrm{g}}+e_{\mathrm{Z}}+1} \rho^{e_{\mathrm{g}}+e_{\mathrm{Z}}+1} & A \to \infty \,. \end{cases} \tag{6}$$

We observe that in this case, the asymptotic slope is $e_{\mathrm{g}} + 1$ for all finite A. On the other hand, considering a phase flip channel ($A \to \infty$) the slope becomes $e_{\mathrm{g}} + e_{\mathrm{Z}} + 1$.

3 Quantum Codes with Minimum Weight Decoding

Let us now assume to have a decoder that always outputs a codeword (complete decoder). This could allow correcting also some (but not all) error patterns which are not correctable with bounded distance decoding. Specifically, we analyse the MW decoder, which can be implemented as the MWPM decoder for surface codes.

Definition 1. We indicate with β_j the fraction of errors of weight j that can be corrected by a complete decoder.

Note that β_j depends in general on the code structure, on the decoder, and on the channel asymmetry parameter A.

Definition 2 (Error class)**.** We state that two error patterns are in the same class if they have the same Pauli weight with respect to each Pauli operator, i.e., they contain the same number of $\boldsymbol{X}$, $\boldsymbol{Y}$, and $\boldsymbol{Z}$ errors, respectively.

In general, the logical error probability of an error-correcting code of length n is

$$\rho_L = \sum_{\boldsymbol{E} \in \mathcal{C}_n} \mathbb{P}\{\text{error}|\boldsymbol{E}\} \, \mathbb{P}\{\boldsymbol{E}\} \tag{7}$$

where $\mathcal{C}_n$ is the set of all possible error classes over n qubits, $\mathbb{P}\{\text{error}|\boldsymbol{E}\}$ is the probability to have an error given the particular error class $\boldsymbol{E}$, and $\mathbb{P}\{\boldsymbol{E}\}$ is the occurrence probability of $\boldsymbol{E}$. Since the probability $\mathbb{P}\{\text{error}|\boldsymbol{E}\}$ depends only on how many $\boldsymbol{X}$, $\boldsymbol{Y}$, and $\boldsymbol{Z}$ the error $\boldsymbol{E}$ contains, we define $f_j(i,\ell)$ as the fraction of errors of weight j with i Pauli $\boldsymbol{Z}$ and ℓ Pauli $\boldsymbol{X}$ errors that are not corrected (thereby leading to a logical error). Then, we can write

$$\rho_L = \sum_{j=1}^{n} \binom{n}{j} (1-\rho)^{n-j} \rho^j (1-\beta_j) \tag{8}$$

where

$$1 - \beta_j = \frac{1}{\rho^j} \sum_{i=0}^{j} \binom{j}{i} p_{\mathrm{Z}}^{i} \sum_{\ell=0}^{j-i} \binom{j-i}{\ell} p_{\mathrm{X}}^{\ell} \, p_{\mathrm{Y}}^{j-i-\ell} f_j(i,\ell) \,. \tag{9}$$

Considering a symmetric error correcting code with bounded distance decoding which corrects up to e_{g} errors, we have that $f_j(i,\ell) = 0$ for $j \leq e_{\mathrm{g}}$ and $f_j(i,\ell) = 1$ otherwise. In the case of an asymmetric code able to correct e_{g} generic errors plus e_{Z} Pauli $\boldsymbol{Z}$ errors (see Section 2.3), $f_j(i,\ell) = 0$ for $j \leq e_{\mathrm{g}}$, $f_j(i,\ell) = 0$ for $e_{\mathrm{g}} < j \leq e_{\mathrm{g}} + e_{\mathrm{z}}$ and $i \geqslant e_{\mathrm{z}}$, and $f_j(i,\ell) = 1$ otherwise. For channels with $p_{\mathrm{X}} = p_{\mathrm{Y}}$ (e.g., depolarizing and asymmetric) we have that

$$1 - \beta_j = \frac{1}{(A+2)^j} \sum_{i=0}^{j} A^i \sum_{\ell=0}^{j-i} \binom{j}{i} \binom{j-i}{\ell} f_j(i,\ell) \tag{10}$$

and $\beta_j = \beta_j(A)$. The dependence on A will be indicated only when necessary. In the following we will assume $p_{\mathrm{X}} = p_{\mathrm{Y}}$.

For a symmetric code, starting from (8) we obtain the upper bound

$$\rho_{\mathrm{L}} \leqslant (1-\beta_{t+1}) \binom{n}{t+1} \rho^{t+1} (1-\rho)^{n-t-1} + \sum_{j=t+2}^{n} \binom{n}{j} (1-\rho)^{n-j} \rho^{j} . \tag{11}$$

Also, we can approximate the logical error rate for $\rho \ll 1$ as

$$\rho_{\mathrm{L}} \approx (1-\beta_{t+1}) \binom{n}{t+1} \rho^{t+1} . \tag{12}$$

Note that equation (12) differs from (2) in that the latter assumes that all errors with weight greater than t are not correctable. The asymptotic slope in log-log domain remains $t+1$, but with an offset depending on $(1-\beta_{t+1})$, compared to (2).

In a similar way, we can find the asymptotic error correction capability of an asymmetric code. In particular, for $\rho \ll 1$ the performance of the code becomes

$$\rho_{\mathrm{L}} \approx (1-\beta_{e_{\mathrm{g}}+e_{\mathrm{Z}}+1}) \binom{n}{e_{\mathrm{g}}+e_{\mathrm{Z}}+1} \rho^{e_{\mathrm{g}}+e_{\mathrm{Z}}+1} + (1-\beta_{e_{\mathrm{g}}+1}) \binom{n}{e_{\mathrm{g}}+1} \rho^{e_{\mathrm{g}}+1} . \tag{13}$$

As indicated by (8) and its approximations, to calculate the performance of quantum codes we need to determine the fraction of correctable errors β_j.

4 Undetectable errors weight enumerator from Quantum MacWilliams identities

4.1 Undetectable Error WE

Definition 3 (Logical operators)**.** The logical operators of a $[[n,k,d]]$ QECC are the elements of the set $\mathcal{N}(\mathcal{S}) \setminus S$, namely the operators that commute with the stabilizer but are not contained in it.

Definition 4 (Undetectable errors)**.** The undetectable errors operators are those coincident with the logical operators. They transform a codeword into another codeword and are therefore undetectable.

Thus, the set of logical operators coincides with the set of undetectable errors. In the following, we will use the two terms interchangeably.

The *undetectable errors weight enumerator* for a $[[n,k,d]]$ quantum code is

$$L(z) = \sum_{w=0}^{n} L_w z^w \tag{14}$$

where L_w is the number of undetectable errors (logical operators) of weight w.

Then, we will show that $L(z)$ can be written as

$$L(z) = \frac{1}{2^k} B(z) - \frac{1}{4^k} A(z) \tag{15}$$

where $A(z) = \sum_{w=0}^{n} A_w z^w$, $B(z) = \sum_{w=0}^{n} B_w z^w$,

$$A_w = 4^k \sum_{\boldsymbol{E}_w} |\boldsymbol{E}_w \cap \mathcal{S}| \tag{16}$$

$$B_w = \frac{1}{2^n} \sum_{\ell=0}^{n} \sum_{s=0}^{w} \binom{\ell}{s} \binom{n-\ell}{w-s} (-1)^s 3^{w-s} A_\ell \tag{17}$$

and where the sum in (16) is over all operators $\boldsymbol{E}_w \in \mathcal{G}_n$ of weight w.

We prove now that (15), together with (16) and (17), gives indeed the undetectable WE. To this aim, let us consider the operators $\boldsymbol{E}_w \in \mathcal{G}_n$ with weight w, i.e., containing exactly w Pauli operators different from the identity. For any two hermitian operators $\boldsymbol{O}_1$ and $\boldsymbol{O}_2$ we can introduce two WEs A_w and B_w [64,65]

$$A_w(\boldsymbol{O}_1, \boldsymbol{O}_2) = \sum_{\boldsymbol{E}_w} \mathrm{tr}\,(\boldsymbol{E}_w \boldsymbol{O}_1)\, \mathrm{tr}\,(\boldsymbol{E}_w \boldsymbol{O}_2) \tag{18}$$

$$B_w(\boldsymbol{O}_1, \boldsymbol{O}_2) = \sum_{\boldsymbol{E}_w} \mathrm{tr}\,(\boldsymbol{E}_w \boldsymbol{O}_1 \boldsymbol{E}_w \boldsymbol{O}_2) \tag{19}$$

where the sum is over all the $\boldsymbol{E}_w$, and $w = 0, \ldots, n$. We will often drop the operators $\boldsymbol{O}_1, \boldsymbol{O}_2$ when the dependence is clear in the context. In the case in which $\boldsymbol{O}_1 = \boldsymbol{O}_2 = \Pi_{\mathcal{C}}$, the projector onto a $[[n,k,d]]$ binary stabilizer code, $A(z)$ and $B(z)$ carry some important properties of the code.

Indeed, let $\boldsymbol{G}_i \in \mathcal{S}$, with $i = 1, ..., n-k$, generators of the stabilizer group of a code, then [66]

$$\Pi_{\mathcal{C}} = \frac{1}{2^{n-k}} \prod_{i=1}^{n-k} (\boldsymbol{I} + \boldsymbol{G}_i) = \frac{1}{2^{n-k}} \sum_{\boldsymbol{S} \in \mathcal{S}} \boldsymbol{S} \,. \quad (20)$$

It can be shown [66] that the WE $A(z)$ defined in (18) is proportional to the stabilizer WE

$$\frac{1}{4^k} A(z) = \sum_{w=0}^{n} \sum_{\boldsymbol{E}_w} |\boldsymbol{E}_w \cap \mathcal{S}| \, z^w. \quad (21)$$

Moreover, $B(z)$ is proportional to the normalizer WE [66]

$$\frac{1}{2^k} B(z) = \sum_{w=0}^{n} \sum_{\boldsymbol{E}_w} |\boldsymbol{E}_w \cap \mathcal{N}(\mathcal{S})| \, z^w. \quad (22)$$

In order to find the relation between A_w and B_w we write the associated WEs in the form

$$A(v, z) = \sum_{w=0}^{n} A_w v^{n-w} z^w$$
$$B(v, z) = \sum_{w=0}^{n} B_w v^{n-w} z^w \,. \quad (23)$$

These two polynomials are related through the quantum MacWilliam identities [65–67]

$$B(v, z) = A\left(\frac{v+3z}{2}, \frac{v-z}{2}\right). \quad (24)$$

Using (24) in (23) we get (17). Finally, the number of undetectable errors of weight w is given by the number of operators of weight w which commute with $\mathcal{S}$, given in (22), minus the number of stabilizers of weight w, given by (21), which leads to (15).

The evaluation of the undetectable errors weight enumerator (15) requires therefore only computing the A_w in (16), which can be carried out by direct inspection of $\mathcal{S}$ for small code sizes, or more in general by using the tools for the computation of the weight distribution of classical codes as discussed in Section 4.4.

For example, let's consider the [[3,1]] repetition code, able to correct one bit flip error. In this case, we have $\boldsymbol{G}_1 = \mathbf{Z}_1\mathbf{Z}_2$, $\boldsymbol{G}_2 = \mathbf{Z}_2\mathbf{Z}_3$. Considering $\boldsymbol{O}_1 = \boldsymbol{O}_2 = \Pi_{\mathcal{C}}$, we have from (20) that the stabilizer is $\mathcal{S} = \{\mathbf{I}_1\mathbf{I}_2\mathbf{I}_3, \mathbf{Z}_1\mathbf{Z}_2, \mathbf{Z}_2\mathbf{Z}_3, \mathbf{Z}_1\mathbf{Z}_3\}$. If we combine (20) and (18), we obtain $\frac{1}{4}A(z) = 1 + 3z^2$. Then we use (17) to compute $\frac{1}{2}B(z) = 1+3z+3z^2+9z^3$. Finally, the undetectable errors WE is: $L(z) = 3z + 9z^3$.

More in general, a trivial way to obtain $A(z)$ for the $[[n, k, d]]$ code that we want to study is to compute all linear combinations among the set of generators. Alternatively, it is possible to consider the connection between stabilizer codes and codes over the Galois field $GF(4)$ by identifying the operators $\boldsymbol{I}$, $\boldsymbol{X}$, $\boldsymbol{Z}$ and $\boldsymbol{Y}$ with the four elements of the field [15, 68]. Hence, the evaluation of $A(z)$ can be seen as the computation of the weight distribution of classical codes over $GF(4)$. Although this problem may be classified as NP-hard [69], a variety of advanced and optimized algorithms have been developed. These algorithms surpass traditional brute force methods in efficiently calculating key metrics such as the weight distribution and the minimum weight. Some of them are the Brouwer–Zimmermann algorithm and its various modifications for cyclic codes, quasi-cyclic codes, and divisible codes [70–76]. Such algorithms are implemented in software tools related to coding theory, such as MAGMA [77]. For all codes analyzed in this paper, the required time for the computation on a laptop of the weight distribution was less then one second. Recently, new techniques have been developed for computing the quantum weight enumerator polynomial $A(z)$, which, for some degenerate stabilizer codes, provide up to an exponential speed up compared to the previous methods [78].

4.2 Bounds on β_{t+1} via Undetectable Errors Weight Enumerator

The performance of an $[[n, k, d]]$ QECC is mainly determined, according to (11), (12), and (13), by the value of β_{t+1}. We show here that, even without analyzing in details the logical operators of a code, it is possible to exploit the undetectable error WE $L(z)$ in order to find upper bounds on the performance for a depolarizing channel. Specifically, considering the general case of a complete MW decoder, we will derive several lower bounds on the value of β_{t+1}, indicated as $\hat{\beta}_{t+1} \leqslant \beta_{t+1}$, for some families of codes. Unless otherwise stated we will assume d odd.

A first bound, valid in general for stabilizer codes, is obtained assuming that each logical operator of weight w can be caused by all the 3^{t+1}

different Pauli errors of weight $t+1$, as

$$\hat{\beta}_{t+1} = \left(1 - \frac{3^{t+1} \sum_{w=2t+1}^{2t+2} L_w \binom{w}{t+1}}{\binom{n}{t+1} 3^{t+1}}\right)^+ . \quad (25)$$

In this equation, L_w is the number of logical operators of weight w, 3^{t+1} is the number of permutations of the different Pauli operators of weight $t+1$ that cause a logical operator, and $\binom{w}{t+1}$ refers to the number of patterns of errors (made of fixed Pauli operators) with weight $t+1$ that can realize the corresponding logical operator. For instance, if we consider $t+1=2$, we have $3^2=9$ different Pauli errors: $\boldsymbol{ZZ}$, $\boldsymbol{XZ}$, $\boldsymbol{ZX}$, $\boldsymbol{XX}$, $\boldsymbol{ZY}$, $\boldsymbol{YZ}$, $\boldsymbol{YY}$, $\boldsymbol{YX}$, $\boldsymbol{XY}$. Moreover, a logical operator of weight $w=3$ can be produced by $\binom{3}{2}=3$ patterns of each of the previous Pauli errors of weight $t+1$. Note that, the summation in (25) starts from $2t+1$ since $L_w=0$ for $w \leqslant 2t+1$, and is limited to $2t+2$ because, when an error of weight $t+1$ is introduced by the channel, a MW decoder will never choose a codeword which differs in more than $t+1$ positions from the original one. The previous bound can be made more tight if we deal with CSS codes, where the $\boldsymbol{X}$ and $\boldsymbol{Z}$ corrections are performed independently. In this case the logical operators of weight d are formed by one Pauli type only. Consequently, a logical operator may be caused by only two types of Pauli errors (instead of the previously considered three). For instance, let us consider the logical operator $\boldsymbol{ZZZ}$: this can be caused only by $\boldsymbol{ZZ}$, $\boldsymbol{ZY}$, $\boldsymbol{YZ}$ and $\boldsymbol{YY}$ Pauli errors. Therefore, for CSS, the number of permutations of Pauli errors that cause the generic logical operator is 2^{t+1}. Moreover, since, for a MW decoder, the error recovery operator has always a weight lower or equal to the weight of the channel error pattern, only half of the error patterns of weight $t+1$ cause a logical operator of weight $w=2t+2$. Hence, the fraction of corrected errors can be bounded by

$$\hat{\beta}_{t+1}^{\mathrm{CSS}} = \left(1 - \frac{2^{t+1} \sum_{w=2t+1}^{2t+2} L_w \binom{w}{t+1} / \left\lfloor \frac{w}{t+1} \right\rfloor}{\binom{n}{t+1} 3^{t+1}}\right)^+ . \quad (26)$$

We remark that this expression applies also to surface codes with MWPM decoder. Furthermore, it is possible to obtain a tighter bound if the generators are composed by $\boldsymbol{X}$ or $\boldsymbol{Z}$ Pauli measurements on the same qubits. In particular, this condition holds for the category of CSS Dual-Containing codes [79]. These codes have a third of the logical operators of minimum weight composed by only $\boldsymbol{Y}$ Pauli operators. Hence, we know that these logical operators can be caused only by Pauli errors composed by $\boldsymbol{Y}$ operators. In this situation, (26) can be rewritten as

$$\hat{\beta}_{t+1}^{\mathrm{CSS-DC}} = \left(1 - \frac{\frac{1}{3} \sum_{w=2t+1}^{2t+2} L_w \left(2\left(2^{t+1}-1\right)+1\right) \binom{w}{t+1} / \left\lfloor \frac{w}{t+1} \right\rfloor}{\binom{n}{t+1} 3^{t+1}}\right)^+ . \quad (27)$$

A more detailed discussion on the derivation of this expression can be found in Appendix.

The values provided by (25), (26), and (27), can be used in (11), (12), and (13) to compute upper bounds on the error rate. These new bounds are easy to compute, as they just need the WE polynomial $L(z)$ derived from the MacWilliams identities.

Remark on degeneracy: If we want to have a more precise estimation of the error performance we should get tighter bounds on β_{t+1}, and this is only possible by a closer analysis of the code. In particular, we should take into account the code degeneracy, which requires a more detailed description of the code logical operators, as discussed in Section 4.2. To explain the role of degeneracy, assume we have a code with logical operators that share the same Pauli operators on $t+1$ common qubits. In the event of an error composed exclusively of these Pauli operators, a deterministic decoder will only trigger one of these logical operators. Therefore, by having knowledge of the structure of the logical operators within a stabilizer code, we could improve the previous bounds. For example, the estimation $\hat{\beta}_{t+1}^{\mathrm{CSS}}$ of (26) can be extended to account for degeneracy as

$$\hat{\beta}_{t+1}^{\mathrm{CSS}} = \left(1 - \frac{2^{t+1} \sum_{w=2t+1}^{2t+2} L_w \binom{w}{t+1} \gamma_w / \left\lfloor \frac{w}{t+1} \right\rfloor}{\binom{n}{t+1} 3^{t+1}}\right)^+ \quad (28)$$

where $\gamma_w \in \left[1/\binom{w}{t+1}, 1\right]$ is the average fraction of potential Pauli error patterns of weight $t+1$ that are not shared between two or more logical operators of weight w. Note that this parameter is contingent on both the weight and the distinct Pauli operators that constitute the logical operators. For instance, if the code is asymmetrically

degenerate with respect to the different Pauli operators, different logical operators of the same weight could share a different number of Pauli operators. We remark that the bounds (25) to (27) are already in closed form and do not necessitate the evaluation of γ_w. Their computational complexity solely arises from evaluating the weight enumerator of the specific quantum code.

Remark on even code distance: The expressions derived in (25) to (27) can be easily modified to the case where d is even by keeping only the weight $w = 2t + 2$ in the summation, and avoiding the division by $\left\lfloor \frac{w}{t+1} \right\rfloor$. For example, for codes with even distance $d = 2t + 2$, (26) becomes

$$\hat{\beta}_{t+1}^{\mathrm{CSS}} = \left(1 - \frac{2^{t+1}\, L_{2t+2} \binom{2t+2}{t+1}}{\binom{n}{t+1} 3^{t+1}}\right)^{+}. \tag{29}$$

Let us provide now examples for some important quantum codes. As regards the $[[7, 1, 3]]$ Steane code [60], using (15) we compute the WE as $L(z) = 21z^3 + 126z^5 + 45z^7$. Hence, by applying (27) it is possible to compute $\hat{\beta}_2$ as

$$\begin{aligned} \hat{\beta}_2^{\mathrm{CSS-DC}} &= \left(1 - \frac{\frac{2}{3}\, 21 \cdot 3 \cdot 3 + \frac{1}{3}\, 21 \cdot 3}{\binom{7}{2} 3^2}\right)^{+} \\ &= \frac{2}{9} \simeq 0.22 \end{aligned} \tag{30}$$

i.e., at least 22% of the errors of weight $j = 2$ are corrected by a minimum weight decoder. For this particular code this estimate is exact, i.e., $\hat{\beta}_2 = \beta_2$, as will be shown in section 4.4. This is because the logical operators of weight $w = 3$ share only $t = 1$ Pauli operators on common qubits. Hence, the code degeneracy does not affect the computation of $\hat{\beta}_2$.

In case of the $[[9, 1, 3]]$ Shor code [7], from (15) the WE results $L(z) = 39z^3 + 208z^5 + 332z^7 + 189z^9$. Moreover, using (26) we get $\hat{\beta}_2 = 0$. This result, which gives us no more information with respect to the bounded distance performance, is due to the strong degeneracy of the code. In fact, even if the number of logical operators of weight $w = 3$ is quite large for a 9 qubits code, a lot of them share $t + 1 = 2$ Pauli operators, affecting the actual β_2 (i.e. $\gamma_3 \ll 1$). An accurate estimation taking into account the effect of degeneracy will be provided in Section 4.2 by a counting argument on the logical operators.

Taking into consideration the $[[13, 1, 3]]$ surface code, from (15) we first find $L(z) = 6z^3 + 24z^4 + 75z^5 + 240z^6 + 648z^7 + 1440z^8 + 2538z^9 + 3216z^{10} + 2634z^{11} + 1224z^{12} + 243z^{13}$. Then, considering a MWPM decoder, from (26) we have

$$\begin{aligned} \hat{\beta}_2^{\mathrm{CSS}} &= \left(1 - \frac{2^2 \left(6 \binom{3}{2} + 24 \binom{4}{2}/2\right)}{\binom{13}{2} 3^2}\right)^{+} \\ &= \frac{57}{117} \simeq 0.49 \end{aligned} \tag{31}$$

so that at least 49% of the errors of weight $j = 2$ is corrected.

The method works also for asymmetric CSS codes. For instance, taking the $[[23, 1, 3/5]]$ surface code, we have $L(z) = 5z^3 + 20z^4 + 51z^5 + 172z^6 + \ldots$. Considering that errors of weight $j = 2$ can be caused only by $\boldsymbol{X}$ operators, while errors of weight $j = 3$ can be caused both by $\boldsymbol{X}$ and $\boldsymbol{Z}$ operators, from (26) we obtain

$$\begin{aligned} \hat{\beta}_2^{\mathrm{CSS}} &= \left(1 - \frac{2^2 (5 \binom{3}{2} + 20 \binom{4}{2}/2)}{\binom{23}{2} 3^2}\right)^{+} \\ &= \frac{659}{759} \simeq 0.87 \end{aligned} \tag{32}$$

and

$$\begin{aligned} \hat{\beta}_3^{\mathrm{CSS}} &= \left(1 - \frac{2^3 (51 \binom{5}{3} + 172 \binom{6}{3}/2)}{\binom{23}{3} 3^3}\right)^{+} \\ &= \frac{17840}{47817} \simeq 0.63\,. \end{aligned} \tag{33}$$

The exact values of β_{t+1} for these codes are provided below. In literature, some lower bounds on the logical error rate of surface codes have been proposed for $\rho \ll 1$, not for the depolarizing but for the phase flip or the bit flip channels [80, 81]. These bounds are quite far from the true performance, as they consider only channel errors leading to logical operators of minimum weight $w = 2t + 1$, i.e. straight horizontal or vertical chains (see Section 6).

4.3 Closed form expression for the WE of Surface Codes

As evident from (26), when dealing with surface codes with MWPM, a channel error with weight $j = t+1$ has the potential to induce logical operators of weight $w = 2t + 1$ and $w = 2t + 2$, when d is odd. On the other hand, when d is even β_{t+1} requires only the knowledge of the number of logical operators of weight $w = 2t+2$. Consequently,

in order to assess the performance bounds of surface codes over a depolarizing channel, it suffices to determine the WE coefficients associated with these two degrees. In the following, we present an expression for determining these values without the necessity of computing the MacWilliams identities. Specifically, given a surface code with minimum distance d, there are exactly $2d$ logical operators of weight $w = d$. These operators correspond to straight horizontal $\boldsymbol{Z}$ and vertical $\boldsymbol{X}$ chains crossing the lattice, as depicted in Fig. 1. In fact, all other paths from a boundary to an opposite one have at least $d+1$ Pauli operators. We conclude that $L_d = 2d$. Furthermore, logical operators of weight $w = d+1$ can be classified into two distinct groups. The first category, composed by $4(d-1)^2$ operators, representing the chains starting from a boundary and reaching the opposite side of the lattice with only one turn. Some examples of these $\boldsymbol{Z}_L$ and $\boldsymbol{X}_L$ operators for the $[[13, 1, 3]]$ surface code are depicted in Fig. 2 (a) and (a'). The last group of logical operators is a tricky one. In particular, these operators are obtained from the $\boldsymbol{Z}_L$ ($\boldsymbol{X}_L$) operators of weight $w = d$ by applying one site (plaquette) generator. Let us consider firstly a logical operator with $w = d$ that traverse qubits which are measured by four-qubits generators. If we apply a four-qubits generator to these logical operators we end up with another logical operator of weight $w = d + 2$. For this reason, we exclude these from the counting. On the other hand, considering logical operators running along a boundary of the lattice, we can apply a three-qubits generator to end up with another logical operator of weight $w = d + 1$, corresponding to the overall count of three-qubits generators, i.e., $4(d-1)$. An example of these operators is shown in Fig. 2 (e) and (f). Counting all the combinations, we conclude that $L_{d+1} = 4d(d-1)$.

In summary, a $[[d^2+(d-1)^2, 1, d]]$ surface code has WE

$$L_d = 2d \qquad L_{d+1} = 4d(d-1) \tag{34}$$

which can be used in (26) and (29). For example, for the $[[41, 1, 5]]$ surface code $L_5 = 10$ and $L_6 = 80$, resulting in $\hat{\beta}_3 = 0.975$.

As a check, (34) is in accordance with what we can obtain by applying (15) and (17) to the numerically computed $A(z)$ for $d = 10$ in [78]. Therefore, due to (34) we do not need to compute $A(z)$ for evaluating the performance of surface codes (see also Section 6).

4.4 QECC Performance via Logical Operators Analysis

Now we show that it is possible to obtain the exact β_j parameters with no need of simulations. This requires an analysis of the structure of the code logical operators, which we do explicitly for some small-size codes. The same approach may become too complicated for large codes, where the use of the closed-form bounds (26), and (27) is preferable. Since we are interested in the value of β_j, we have to find the fraction of errors of weight j that can cause the decoder to miscorrect and induce a logical error. In order to compute it, we will consider not only $L(z)$ but also the structure of the stabilizer.

In the case of a CSS stabilizer code, taking into account that the total number of different error patterns of weight j is $\binom{n}{j}$, we can express β_j as

$$\beta_j = 1 - \frac{\sum_{i=0}^{j} A^i \sum_{\ell=0}^{j-i} \sum_w L_w(i,\ell)\, \mu_j^{(w)}(i,\ell)\, \gamma_w(i,\ell)}{(A+2)^j \binom{n}{j}} \tag{35}$$

where: $L_w(i,\ell)$ stands for the number of logical operators of weight w that can be caused by the combined action of channel errors of weight j composed by i $\boldsymbol{Z}$ and ℓ $\boldsymbol{X}$ Pauli operators and $w-j$ correction operators applied by a complete decoder; $\mu_j^{(w)}(i,\ell)$ refers to the number of different patterns of errors with weight j that can induce the corresponding logical operator of weight w; and $\gamma_w(i,\ell) \in \left[1/\binom{w}{j}, 1\right]$ is the average fraction of potential Pauli error patterns of weight j that are not shared between two or more logical operators. Note that (35) and (10) are equivalent, since

$$\sum_w L_w(i,\ell)\, \mu_j^{(w)}(i,\ell)\, \gamma_w(i,\ell) = \binom{n}{j}\binom{j}{i}\binom{j-i}{\ell} f_j(i,\ell). \tag{36}$$

In particular, on both sides of (36) we find the total number of logical errors induced by the weight j error class identified by i operators of type $\boldsymbol{Z}$ and ℓ operators of type $\boldsymbol{X}$.

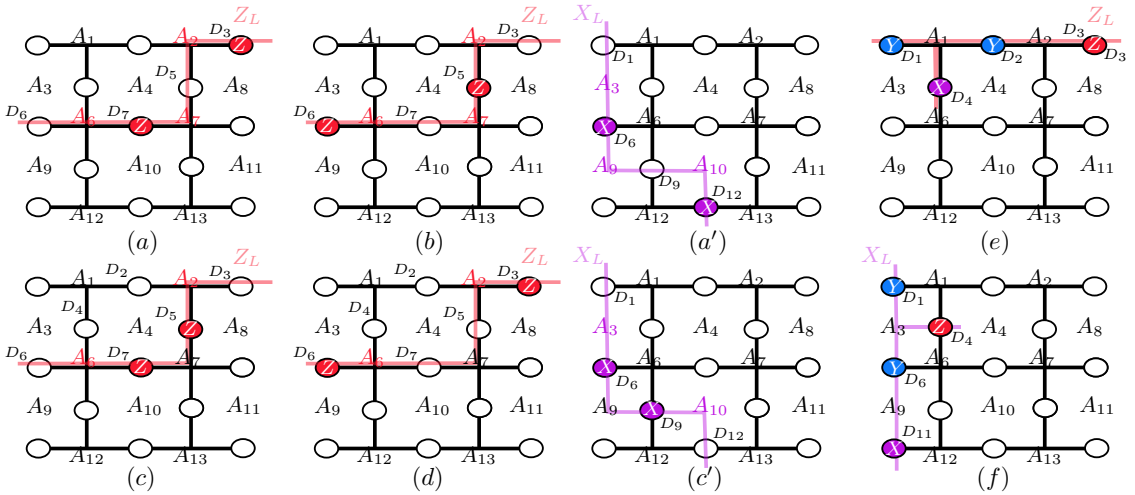

Figure 2: Example of errors leading to a logical operator of $w = 4$ for the $[[13, 1, 3]]$ surface code. $\boldsymbol{Z}$, $\boldsymbol{X}$, and $\boldsymbol{Y}$ errors on qubits are depicted in red, purple, and blue, respectively. (a) $\boldsymbol{Z}_L$ occurs if $\boldsymbol{Z}$ correction operators are applied on data qubits D_5 and D_6. The errors are corrected if the MWPM decoder applies $\boldsymbol{Z}$ on D_3 and D_7. (b) $\boldsymbol{Z}_L$ occurs if $\boldsymbol{Z}$ correction operators are applied on D_3 and D_7. The errors are corrected if the MWPM decoder applies $\boldsymbol{Z}$ on D_5 and D_6. (c) $\boldsymbol{Z}_L$ occurs if $\boldsymbol{Z}$ correction operators are applied on D_3 and D_6. The MWPM decoder could apply also $\boldsymbol{Z}$ operators on D_5 and D_7, correcting the error, or on D_2 and D_4, correcting the error. (d) $\boldsymbol{Z}_L$ occurs if $\boldsymbol{Z}$ correction operators are applied on D_5 and D_7. The MWPM decoder could apply also $\boldsymbol{Z}$ operators on D_3 and D_6 or on D_2 and D_4, causing a different logical operator. (a', c') Analogous examples for $\boldsymbol{X}_L$ logical operator. (e, f) Logical operators of $w = 4$ caused by $\boldsymbol{YY}$ errors.

In the following, we provide some examples over a depolarizing channel ($A = 1$), in order to clarify the reasoning behind the evaluation of β_j.

First, let us consider the $[[5, 1, 3]]$ perfect code [61]. Note that this is not a CSS code. Using (15) we obtain $L(z) = 30z^3 + 18z^5$, so we know that the number of logical operators of weight 3 is 30. Then, we have

$$\beta_2 = 1 - \frac{30 \cdot 3}{\binom{5}{2}3^2} = 0 \tag{37}$$

where we have considered that the total number of pairs of Pauli errors is $\binom{5}{2}3^2$, and that each logical operator of weight 3 can be caused exactly by three different combinations of errors of weight $j = 2$. This result was expected since the code is perfect.

A more interesting case is that of the $[[7, 1, 3]]$ Steane code [60]. Starting from $L(z) = 21z^3 + 126z^5 + 45z^7$ and assuming a MW decoder, we calculate

$$\begin{aligned}\beta_2 = 1 - \frac{1}{\binom{7}{2}3^2}\Big[&7\mu_2^{(3)}(2,0) + 7\mu_2^{(3)}(1,0) \\ &+ 7\mu_2^{(3)}(0,2) + 7\mu_2^{(3)}(0,1) + 7\mu_2^{(3)}(0,0)\Big] \\ = 1 - \frac{7\,(3+6) + 7\,(3+6) + 7 \cdot 3}{\binom{7}{2}3^2} &= \frac{2}{9} \simeq 0.22.\end{aligned} \tag{38}$$

To derive this value we observed that, unlike the perfect code, each logical operator of $w = 3$ is composed of only one kind of Pauli operator. Specifically, we have seven $\boldsymbol{XXX}$, seven $\boldsymbol{ZZZ}$, and seven $\boldsymbol{YYY}$ logical operators. In addition, $\gamma_3 = 1$ since there are no logical operators that share Pauli operators on the same qubits. Note that these logical operators act on the same qubits, leading to a simple expression. Moreover, we have to consider that each one of these $\boldsymbol{X}$ ($\boldsymbol{Z}$) logical operators can be generated by $\boldsymbol{ZZ}$ ($\boldsymbol{XX}$) errors, and also by $\boldsymbol{XY}$ ($\boldsymbol{ZY}$), since single errors can always be corrected, while $\boldsymbol{Y}$ logical operators are caused only by two $\boldsymbol{Y}$ errors. From (12) and (38) we then obtain for the $[[7, 1, 3]]$ code over the depolarizing channel

$$p_L \approx 16.3\rho^2\,, \quad \rho \ll 1\,. \tag{39}$$

As a third example let us analyze the $[[9,1,3]]$ Shor code [7]. From (15) the WE results $L(z) = 39z^3 + 208z^5 + 332z^7 + 189z^9$. Assuming a MW decoder, 39 undetectable errors of weight three are caused by channel errors of weight two. A closer look reveals that the 39 operators include three $\boldsymbol{XXX}$, 27 $\boldsymbol{ZZZ}$, and 9 $\boldsymbol{YYX}$ logical operators. The code is degenerate with respect to $\boldsymbol{Z}$ Pauli errors, so it is necessary to treat differently the logical operators of each type. In particular, since each channel error of the kind $\boldsymbol{ZZ}$ can cause three different $\boldsymbol{ZZZ}$ logical operators, $\gamma_3(2,0) = \gamma_3(1,0) = \frac{1}{3}$. For instance, the channel error $\boldsymbol{Z}_1\boldsymbol{Y}_4$ could cause either $\boldsymbol{Z}_1\boldsymbol{Z}_4\boldsymbol{Z}_7$, $\boldsymbol{Z}_1\boldsymbol{Z}_4\boldsymbol{Z}_8$, or $\boldsymbol{Z}_1\boldsymbol{Z}_4\boldsymbol{Z}_9$. (The error $\boldsymbol{X}_4$ is corrected by the decoder).

Thus, considering that $\gamma_3(0,0) = \gamma_3(0,1) = \gamma_3(0,2) = 1$, we obtain

$$\begin{aligned}\beta_2 &= 1 - \frac{1}{\binom{9}{2}3^2}\Big[27\,[\mu_2^{(3)}(2,0) + \mu_2^{(3)}(1,0)]\,\gamma_3(2,0) \\ &\quad + 3\,[\mu_2^{(3)}(0,2) + \mu_2^{(3)}(0,1)] + (27+9)\,\mu_2^{(3)}(0,0)\Big] \\ &= 1 - \frac{27\,(3+6)\,\frac{1}{3} + 3\cdot 3 + 3\cdot 6 + 27 + 9}{\binom{9}{2}3^2} \\ &= \frac{5}{9} \simeq 0.56 \end{aligned} \tag{40}$$

From (12) and (40) we obtain for the $[[9,1,3]]$ code

$$\rho_L \approx 16\rho^2\,, \quad \rho \ll 1\,. \tag{41}$$

Finally, we investigate the β_j values for surface codes, which belong to the class of CSS codes, assuming MWPM decoding. Considering the $[[13,1,3]]$ surface code, we have $L(z) = 6z^3 + 24z^4 + 75z^5 + ...$ As before, we need to analyze how channel errors of weight $j = 2$ cause logical operators. Let us first look at the six logical operators of weight three. For the surface codes we know that the logical operators cross the lattice from side to side. Thus, there are three $\boldsymbol{XXX}$ (crossing horizontally) and three $\boldsymbol{ZZZ}$ (crossing vertically) logical operators. As a result, $\mu_2^{(3)}(0,2) = \mu_2^{(3)}(2,0) = 3$, given that, for instance, a $\boldsymbol{ZZZ}$ logical operator can arise from $\binom{3}{2}$ error patterns of the $\boldsymbol{ZZ}$ type. Furthermore, $\mu_2^{(3)}(0,1) = \mu_2^{(3)}(1,0) = 6$, as a $\boldsymbol{ZZZ}$ logical operator is induced by $2\binom{3}{2}$ error patterns of the $\boldsymbol{ZY}$ kind. In this code we have also logical operators of weight $w = 4$, as illustrated in Fig. 2.

Table 1: Coefficients for performance evaluation, $[[13,1,3]]$ surface code.

	$i=2$ $\ell=0$	$i=1$ $\ell=0$	$i=0$ $\ell=0$	$i=0$ $\ell=1$	$i=0$ $\ell=2$
L_3, γ_3	3, 1	3, 1	6, 1	3, 1	3, 1
$\mu_2^{(3)}$	3	6	3	6	3
L_4, γ_4	8, 3/4	8, 3/4	16, 3/4	8, 3/4	8, 3/4
$\mu_2^{(4)}$	2	4	2	4	2

Specifically, among the $L_4 = 24$ logical operators, 16 are composed by $\boldsymbol{XXXX}$ and $\boldsymbol{ZZZZ}$, and the remaining eight are of type $\boldsymbol{YYXZ}$, as illustrated in Fig. 2 $(e), (f)$. For these eight cases, we are left, after MWPM decoding, with a logical operator with three $\boldsymbol{Z}$ or three $\boldsymbol{X}$, which have been already counted when discussing the weight $w = 3$. As regards the other 16 logical operators, we have $\mu_2^{(4)}(2,0) = \mu_2^{(4)}(0,2) = \mu_2^{(4)}(0,0) = 2$. In particular, for each logical operator of weight $w = 4$ there are $\binom{w}{j} = \binom{4}{2} = 6$ different patterns of errors of weight $j = 2$ that can cause it. However, one of them is always corrected (due to the degeneracy of the code), while another one cause a logical operator of weight $w = 3$ that we have already taken into consideration. About the remaining four, they cause, in pairs, the same syndrome. We consider a deterministic decoder, such as the MWPM, that associate one error pattern to each syndrome. Thus, only two patterns will not be corrected. Also, note that, among the four patterns of errors of weight $j = 2$ that can cause a logical operator of weight $w = 4$, one is in common with another logical operator. For example, the error pattern $\boldsymbol{Z}_3\boldsymbol{Z}_6$ depicted in Fig. 2(d) may also cause the logical operator $\boldsymbol{Z}_2\boldsymbol{Z}_3\boldsymbol{Z}_4\boldsymbol{Z}_6$. Hence, among the four faulty error patterns corresponding to the pair of logical operators, one pattern is repeated twice. Therefore we have $\gamma_4(i,\ell) = 3/4$. In Tab. 1, we report, for the $[[13,1,3]]$ surface code, the values of $L(z)$, $\gamma_w(i,\ell)$, and $\mu_j^{(w)}(i,\ell)$ that are needed in order to compute β_2. If we put these parameters into (35), we obtain

$$\beta_2 = \frac{267}{351} \simeq 0.76. \tag{42}$$

From (12) and (42) we obtain for the $[[13,1,3]]$ surface code over the depolarizing channel

$$\rho_L \approx 18.7\rho^2\,, \quad \rho \ll 1\,. \tag{43}$$

Using (35) it is also possible to obtain the value of β_j for asymmetric channels. For instance, in the

case of the [[13,1,3]] surface code over a phase flip channel, we have only three $\boldsymbol{Z}$ logical operators with $w = 3$ and eight $\boldsymbol{Z}$ logical operators with $w = 4$, giving

$$\beta_2 = 1 - \frac{3\mu_2^{(3)}(0,2) + 8\mu_2^{(4)}(0,2)\gamma_4(0,2)}{\binom{13}{2}} = 1 - \frac{3 \cdot 3 + 8 \cdot 2\frac{3}{4}}{\binom{13}{2}} = \frac{19}{26} \simeq 0.73\,. \tag{44}$$

4.5 β_j by exhaustive search

A different approach to compute the exact β_j coefficients for a given code and a given decoder is by exhaustive search. In Tab. 2 we report for some surface codes the percentage of non-correctable errors for each error class $f_j(i,\ell)$, which we have evaluated by exhaustive search with a MWPM decoder. In doing so we exploited the Lemon C++ library [82], which provides an efficient implementation of graphs and networks algorithms. For example, in the case of the $[[23,1,3/5]]$ surface code, it results $f_2(0,2) = 0.16$. Once we have $f_j(i,\ell)$, we can compute the value of $1-\beta_j$ for arbitrary A (for example in Tab. 2 we report the cases $A = 1$, $A = 10$, $A = 100$, and $A \to \infty$), by weighting the percentages of non-correctable errors as indicated by (10). From Tab. 2, we observe that, as anticipated, surface codes can correct a large number of errors above the guaranteed error correction capability. Note that, as the code's length increases, performing an exhaustive search to derive the β_j values may pose significant computational demands.

5 Noisy Syndrome Measurement

In realistic quantum systems, syndrome measurements are inherently noisy and often require repeated execution over time to yield reliable outcomes. In this section, we shift our attention to syndrome extraction circuits and introduce a framework that accounts for the presence of noise during measurement. We recall that a gadget for a specific function is defined as a circuit to perform that function on the encoded state [15]. Then, we replace each measurement in the original circuit with a fault-tolerant gadget that replicates its intended action on logical qubits, ensuring it behaves as the ideal measurement would in the unencoded circuit. By systematically propagating error probabilities through the circuit, we derive upper bounds on the resulting physical error rate. When combined with the previously established theoretical bounds, this approach enables us to estimate the logical error rate including the effects of the noisy gadget, offering a practical tool for analyzing performance in real-world quantum error correction scenarios. Specifically, in realistic syndrome extraction gadgets, two main challenges arise. First, syndrome measurements are inherently noisy, making a single-shot measurement unreliable [15, 83]. This issue is typically addressed by repeating the measurement until the same syndrome is obtained $t+1$ times consecutively [83]. In many practical scenarios, for distance-d codes, it is common to perform d repeated measurements [12, 57, 84]. The second challenge stems from error propagation during the extraction process. In particular, errors on the syndrome qubit can spread to multiple data qubits through entangling gates, leading to high-weight errors known as hook errors. To mitigate these effects, techniques such as the use of cat states or flag qubits have been developed, which can signal the presence of correlated faults [83, 85, 86]. An alternative is the Steane error correction gadget, which, by construction, implements a fault-tolerant gate through transversal operations and destructive measurement of a logical ancilla [87].

In the following, we consider realistic syndrome extraction gadgets that include faulty initialization, single-qubit and two-qubit gates, and measurements. We assume that after each of these operations, a depolarizing channel acts on the qubits involved, with error probabilities specific to each type of gate. In addition to these, we also include an initial depolarizing channel acting on each data qubit before the syndrome extraction gadget is applied. Moreover, in practical quantum experiments, error correction is executed over multiple consecutive correction cycles [39, 88]. Between cycles, data qubits may accumulate errors due to decoherence or as a result of gate operations performed during encoded computation. Importantly, errors arising from an incorrect syndrome bit in a given cycle, as well as those introduced by the final two-qubit gate during syndrome extraction (which are not de-

Table 2: Surface codes: fraction of non-correctable error patterns per error class using MWPM.

Code	XX	XZ	XY	ZZ	ZY	YY				
$[[13,1,3]]$	0.27	0	0.27	0.27	0.27	0.51				
$[[23,1,3/5]]$	0.16	0	0.16	0	0	0.15				
Code	XXX	XXZ	XXY	XZZ	XZY	XYY	ZZZ	ZZY	ZYY	YYY
$[[13,1,3]]$	0.52	0.27	0.52	0.27	0.45	0.67	0.53	0.53	0.68	0.78
$[[23,1,3/5]]$	0.39	0.15	0.39	0	0.15	0.39	0.08	0.08	0.22	0.45
$[[41,1,5]]$	0.023	0	0.023	0	0	0.023	0.024	0.024	0.024	0.046

	$1-\beta_2(A)$				$1-\beta_3(A)$			
Code	$A=1$	$A=10$	$A=100$	$A\to\infty$	$A=1$	$A=10$	$A=100$	$A\to\infty$
$[[13,1,3]]$	0.24	0.233	0.265	0.27	0.52	0.48	0.523	0.53
$[[23,1,3/5]]$	0.07	0.0044	$6\cdot10^{-5}$	0	0.203	0.0736	0.0778	0.08
$[[41,1,5]]$	0	0	0	0	0.014	0.019	0.023	0.024

tected within the same cycle), can be dealt with in subsequent cycles.

5.1 Cat States

In this approach, the stabilizer generator $\boldsymbol{G}_i$ of weight γ_i is measured using a cat state (or GHZ state) composed of γ_i ancilla qubits. This configuration ensures that each controlled gate in the extraction circuit interacts with a dedicated syndrome qubit, thereby limiting the spread of errors [83]. The entire syndrome extraction procedure is assumed to be repeated for $r > t$ shots [15, 83]. The error probability of a specific syndrome bit can be bounded by considering that $t+1$ consecutive syndrome extractions providing the same syndrome, i.e., no errors have occurred [83]. This is

$$P_i \leq 1-(1-p_i)^{t+1} \tag{45}$$

where p_i is the error probability in a single shot syndrome bit measurement. This can be upper bounded as

$$p_i \leqslant 1-(1-\rho_{\text{init}}^{\text{cat}})\Big[(1-\rho_{1Q}) \times (1-\rho_{2Q})(1-\rho_{\text{meas}})\Big]^{\gamma_i} \tag{46}$$

with $\rho_{\text{init}}^{\text{cat}}$, ρ_{1Q}, ρ_{2Q}, and ρ_{meas} denote the depolarizing error probabilities associated with the cat state initialization, single-qubit gates, two-qubit gates, and qubit measurement, respectively. Moreover, note that a MW decoder can apply corrections to at most t qubits. As a result, a single syndrome bit error can, in the worst case, be equivalent to a Pauli error affecting t qubits. For a given stabilizer generator, the probability that a physical qubit is affected due to a faulty syndrome bit is t/n, where n is the total number of qubits. Therefore, the overall qubit error probability resulting from faulty syndrome measurements can be upper bounded as

$$P_{\text{syn}} \leqslant \frac{t}{n}\left[1-\prod_{i=1}^{n-k}(1-P_i)\right]. \tag{47}$$

To compute the depolarizing error probability after syndrome extraction, we also account for an initial depolarizing channel on each data qubit, as well as a depolarizing channel associated with each two-qubit gate that interacts with a data qubit. Since we aim to obtain an upper bound for the depolarizing error probability, which is assumed to be the same for every data qubit, we are interested in the maximum number of two-qubit gates applied to any single qubit. Thus, we define $\delta_{\max} = \max_j \delta_j$, where δ_j represents the number of two-qubit gates applied to qubit j in a single round of syndrome extraction. Hence, the equivalent depolarizing error probability on each qubit at the end of the measurement based on cat state can be upper bounded as

$$\rho_{\text{eq}}^{\text{cat}} \leqslant 1-(1-\rho)(1-P_{\text{syn}}) \times \Big[(1-\rho_{2Q})(1-\rho_{\text{init}}^{\text{cat}})\Big]^{r\,\delta_{\max}} \tag{48}$$

where ρ is the depolarizing error probability on each qubit before the gadget, and P_{syn} is the faulty syndrome error probability due to the previous error correction cycle. Finally, an upper bound on the logical error rate after the gadget can be obtained by equation (11) with ρ replaced with $\rho_{\text{eq}}^{\text{cat}}$ from equation (48), in conjunction with the results from Section 4.

5.2 Flag Error Detection

In this scenario, a single ancillary qubit is assigned to each syndrome bit. However, to mitigate hook errors, additional flag qubits can be introduced [85, 86]. These flag qubits are designed to propagate any hook errors, and their subsequent measurements allow for the detection or the correction of such errors. Upon detection, the computation can be post-selected and restarted, as an example, to maintain the integrity of the syndrome extraction process. Error detection and correction flag circuits used for measuring syndrome bits in codes of distance three and five are presented in [86, 89]. Here, the probability of syndrome error per single measurement can be upper bounded as

$$p_i \leqslant 1-(1-\rho_{\text{init}})(1-\rho_{1Q})^2 \times (1-\rho_{2Q})^{\gamma_i+2f_i}(1-\rho_{\text{meas}}) \tag{49}$$

where f_i denotes the number of flag qubits required for the $i-th$ generator, and ρ_{init} represents the depolarizing error probability associated with qubit initialization. Note that each flag qubit is associated with two two-qubit gates involving the syndrome qubit. Also, we can compute P_{syn} using (49) in (45) and (47). In this case, the equivalent depolarizing error probability per qubit at the end of the gadget $\rho_{\text{eq}}^{\text{flag}}$ is

$$\rho_{\text{eq}}^{\text{flag}} \leqslant 1-(1-\rho)(1-P_{\text{syn}})(1-\rho_{2Q})^{r\,\delta_{\max}} \tag{50}$$

where ρ is the depolarizing error probability on each qubit before the gadget, and P_{syn} is the faulty syndrome error probability due to the previous error correction cycle.

Moreover, we must account for the possibility that a flag gadget might fail to detect a hook error, for instance, due to an additional error occurring during its measurement. Since we are computing an upper bound, we assume that such a failure results in a logical error. Specifically, the error probability in a flag gadget can be upper bounded as

$$p_{\text{flag}} \leqslant 1-(1-\rho_{\text{init}})(1-\rho_{1Q})^2 \times (1-\rho_{2Q})^2(1-\rho_{\text{meas}}) \tag{51}$$

Given a generator i, with $1 \leqslant i \leqslant n-k$, and a specific two-qubit gate c, with $1 \leqslant c \leqslant \gamma_i$, used for measuring that generator, we define $F_{i,c}$ as the number of flag qubits that, in the absence of errors, are capable of detecting a hook error resulting from that particular two-qubit gate. This corresponds to all the flag qubits that have been entangled with the syndrome qubit before the two-qubit gate c, but have not yet been disentangled at the time c is applied. Then, the probability that an hook error is undetected is upper bounded by

$$P_{\text{un,hook}} \leqslant \rho_{2Q} \sum_{i=1}^{n-k} \sum_{c=2}^{\gamma_i-2} p_{\text{flag}}^{F_{i,c}}. \tag{52}$$

Note that the inner summation excludes the first and last two two-qubit gates of the generator, as these cannot give rise to hook errors (up to a stabilizer generator). Finally, an upper bound on the logical error rate can be computed as

$$\rho_L' \leqslant \rho_L + P_{\text{un,hook}} - \rho_L P_{\text{un,hook}} \tag{53}$$

where ρ_L is given by equation (11) with ρ replaced with $\rho_{\text{eq}}^{\text{flag}}$ from equation (50), in conjunction with the results from Section 4.

5.3 Steane Error Correction

This gadget, specifically designed for CSS codes, involves preparing logical $|0\rangle$ and $|+\rangle$ ancilla states, followed by the transversal application of CNOT gates to propagate X and Z errors from the data block to the ancilla [87]. The ancilla states are then destructively measured to extract the syndrome. Due to the transversality of the CNOT operations, no hook errors can occur, and the circuit does not require repeated measurements. Here, the ancilla error probability per qubit can be upper bounded as

$$P_a \leqslant 1-(1-\rho_a)(1-\rho_{1Q})(1-\rho_{2Q})(1-\rho_{\text{meas}}) \tag{54}$$

where ρ_a is the depolarizing error probability on each ancilla qubit before the gadget. Note that, employing this error correction gadget, a single qubit error in the ancilla state can only produce a single qubit error in the encoded state. The depolarizing error probability on each qubit after the gadget is upper bounded as

$$\rho_{\text{eq}}^{\text{Ste}} \leqslant 1-(1-\rho)(1-\rho_{2Q})^2(1-P_a)^2 \tag{55}$$

where ρ is the depolarizing error probability on each qubit before the gadget. Finally, an upper

bound on the logical error rate after the gadget can be obtained by equation (11) with ρ replaced with $\rho_{\text{eq}}^{\text{Ste}}$ from equation (55), in conjunction with the results from Section 4.

6 Numerical Results

In this section, we evaluate the performance of some codes, in terms of logical error rate p_L vs. physical error rate ρ, based on the analysis described previously. In particular, the initial four analyses concentrate on assessing the asymptotic logical error rate of certain quantum codes introduced in preceding sections, namely when $\rho \ll 1$, across symmetric and asymmetric channels. Moreover, the concluding analysis addresses the scenario of elevated physical error rates.

1) Comparison between the $\hat{\beta}_{t+1}$ from the WE and the exact β_{t+1}. In Tab. 3 we compare the estimated $\hat{\beta}_{t+1}$ from Section 4.2 (equations (26), (27), (29)), and the exact β_{t+1} from Sections 4.4 and 4.5 over the depolarizing channel. We remark that the estimates $\hat{\beta}_{t+1}$ are derived solely by employing the logical weight enumerator of the particular quantum code (i.e., from the MacWilliams identities, or (34) for surface codes), whereas the exact values are obtained through a counting argument applied to the logical operators. Specifically, both the $[[5,1,3]]$ perfect code and the $[[7,1,3]]$ Steane code exhibit asymptotic non-degeneracy, leading to a congruence between the estimated values and their exact counterparts. Instead, the $[[9,1,3]]$ Shor code displays strong degeneracy, so that the estimate is useless ($\hat{\beta}_2 = 0$). In this case, we can resort to the logical operator analysis detailed in Section 4.4 which gives the exact β_2. Regarding surface codes, we observe that the estimated $\hat{\beta}_{t+1}$ closely approximate the exact values, with the disparity diminishing as the code's distance is increased.

2) Comparison between analysis and simulation for the $[[9,1,3]]$ Shor code. To verify the correctness of the proposed analytical approach we start by studying the $[[9,1,3]]$ Shor code. As observed, for this code the estimation (26) is not useful, as it gives $\hat{\beta}_2 = 0$, which coincides with the bounded distance decoding due to the strong degeneracy of the Shor code. Therefore, we use the logical operator analysis of Section 4.4. Fig. 3 shows, for the [[9,1,3]] code, a comparison between the upper bound (11), the asymptotic approximation (12) (which becomes (41)), and the logical error rate obtained via Monte Carlo simulations, adopting a MW decoder. It can be seen that the results are in perfect agreement for $\rho < 0.1$, while there is a small gap for larger ρ. This gap arises because (11) implicitly assumes $\beta_j = 0$ for $j \geqslant 3$, while the Shor code is able to correct also a little percentage of errors of weight $j \geqslant 3$, as for instance $\boldsymbol{X}$ errors in three different qubit triplets $(\boldsymbol{X}_1\boldsymbol{X}_4\boldsymbol{X}_7)$. Moreover, in the plot, we report the error probability with the BD decoder, computed using (1), which has the same trend as the MW decoder. The gap between the two curves is due to the fraction of weight two errors which are corrected by the MW decoder.

Table 3: Comparison between the bounds from Section 4.2, and the exact values from Sections 4.4 and 4.5, depolarizing channel.

	$1-\hat{\beta}_{t+1}$	$1-\beta_{t+1}$
$[[5,1,3]]$	1	1
$[[7,1,3]]$	0.88	0.88
$[[9,1,3]]$	1	0.44
$[[13,1,3]]$	0.51	0.24
$[[23,1,3/5]]$ $e_{\text{g}}=1$	0.13	0.07
$[[23,1,3/5]]$ $e_{\text{g}}+e_{\text{z}}=2$	0.37	0.20
$[[41,1,5]]$	$2.5\cdot10^{-2}$	$1.4\cdot10^{-2}$
$[[85,1,7]]$	$6.00\cdot10^{-4}$	-
$[[145,1,9]]$	$1.02\cdot10^{-5}$	-
$[[181,1,10]]$	$4.33\cdot10^{-7}$	-

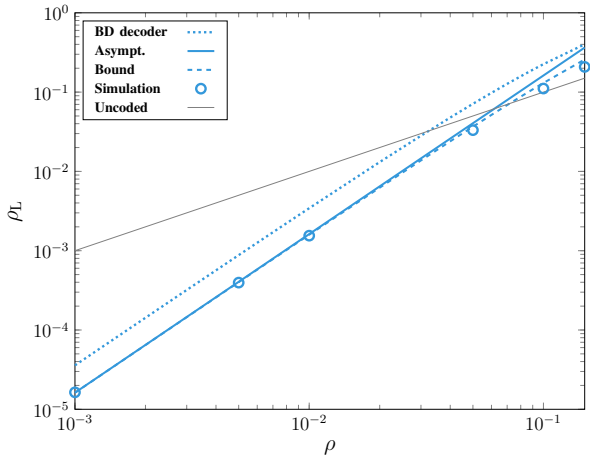

Figure 3: Logical error rate, $[[9,1,3]]$ Shor code over a depolarizing channel. Comparison between theoretical analysis (curves) and simulation (symbols). The curves refer to: the BD decoding performance (1); the MW decoding upper bound (11) and its asymptotic approximation (12) with the exact β_2 from Tab. 3.

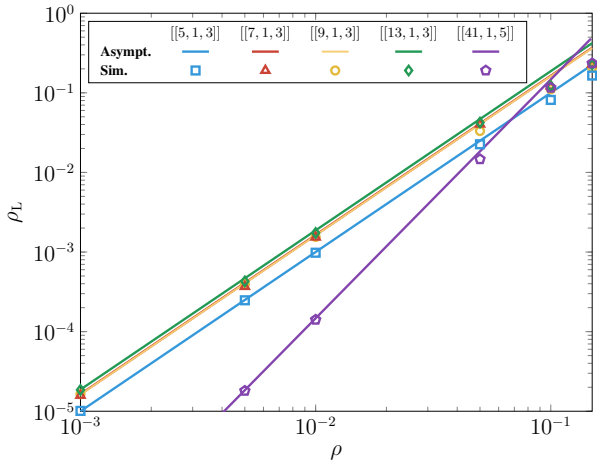

Figure 4: Logical error rate vs. physical error rate, $[[5,1,3]]$ code, $[[7,1,3]]$ Steane code, $[[9,1,3]]$ Shor code, the $[[13,1,3]]$ surface code, and the $[[41,1,5]]$ surface code, over a depolarizing channel. Comparison between theoretical analysis (curves) and simulation (symbols). The solid curves refer to the asymptotic approximation (12) with the exact β_{t+1} from Tab. 3.

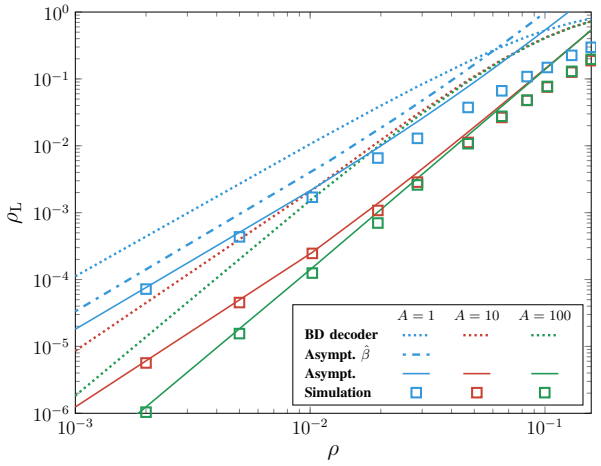

Figure 5: Logical error rate, $[[23,1,3/5]]$ surface code over symmetric and asymmetric channels. Comparison between: the BD decoding performance (4); the asymptotic approximation (13) with the estimated $\hat{\beta}_2$ and $\hat{\beta}_3$ from Tab. 3, and with their exact values from Tab. 2; the simulations with a MWPM decoder.

3) Asymptotic performance analysis over depolarizing channel. In Fig. 4, we present a comparison between the asymptotic approximation of the logical error rate and the results obtained from Monte Carlo simulations using a MW decoder. The analysis includes the $[[5,1,3]]$ perfect code, the $[[7,1,3]]$ Steane code, the $[[9,1,3]]$ Shor code, the $[[13,1,3]]$ surface code, and the $[[41,1,5]]$ surface code. In doing so, we use (12) and the values of β_2 obtained in Section 4.4. In particular, we can see that the $[[5,1,3]]$ perfect code has the best error correction capability among the codes with distance $d = 3$, despite it is not able to correct any error of weight $j \geqslant 2$. This is because it is shorter than the others, so less prone to channel errors. For the same reason, Steane and Shor codes show almost the same performance even if the former has a much smaller value of β_2. Finally, surface codes pay the price of all the implementation benefits that their lattice provides. For instance, even if the $[[13,1,3]]$ surface code is able to correct many errors of weight $j = 2$, it uses a large number of physical qubits, resulting in worst performance than the previous codes.

4) Asymptotic performance analysis over asymmetric channels. In Fig. 5, we study the performance over channels with asymmetry parameter $A = 1$, $A = 10$, and $A = 100$, for the $[[23,1,3/5]]$ surface code. Specifically, we have determined the bounded distance performance using (4). It is interesting to note that, for all kinds of bias of the channel, the simulated logical error rate has the same behavior of the bound error probability, but with a gap between each couple of curves. This gap is due to the capability of surface codes to correct many errors of weight $w \geqslant t+1$. However, since not all the errors of weight $w = t+1$ can be corrected, we have $\beta_{t+1} > 0$, and this makes the asymptotic slope to be $t+1$, no matter how small is β_{t+1}. Moreover, we computed the asymptotic approximations of the logical error rate for the $[[23,1,3/5]]$ surface code, over the same symmetric and asymmetric channels using (13). The values of β_j used in this figure are shown in Tab. 2. We can observe that, for $p \ll 1$, these curves are tight to the respective Monte Carlo simulations. Finally, in the case of a depolarizing channel, we provide the approximation obtained with the estimated $\hat{\beta}_j$ from MacWilliams identities, computed by (32) and (33). Notably, this closely approximates the actual performance of the code.

5) High error-rate performance analysis. We want here to show that, by using (8) with several β_j besides β_{t+1}, it is possible to estimate the code performance not only for small p (for this the β_{t+1} would suffices), but also for larger values of p. In Fig. 6 and Fig. 7 we show the performance as given by equation (8) over depolarizing and phase flip channels, taking into account many values of β_j. To this aim, we compare the analytical formulas with simulations for the

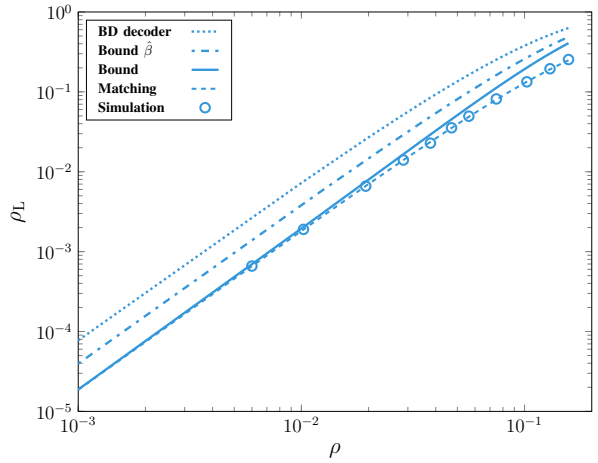

Figure 6: Logical error rates, comparison between theoretical analysis and simulation, MWPM decoder: $[[13, 1, 3]]$ surface code over a depolarizing channel. The curves refer to: the BD decoding performance (1); the upper bound (11) with the estimated β_2 and its exact value from Tab. 3; the matching (8) using $(\beta_2, \beta_3, \beta_4, \beta_5, \beta_6) = (0.76, 0.48, 0.48, 0.46, 0.5)$.

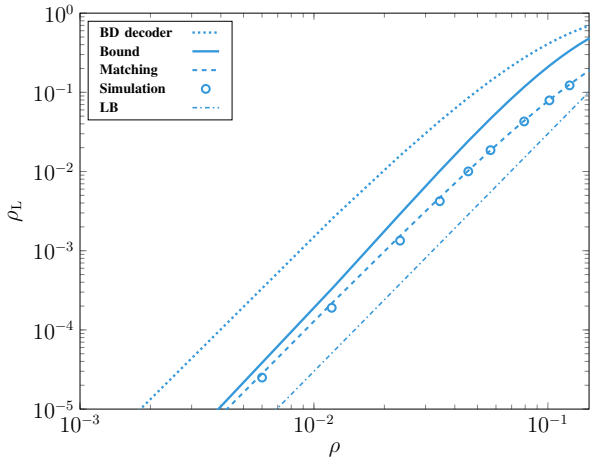

Figure 7: Logical error rates, comparison between theoretical analysis and simulation, MWPM decoder: $[[23, 1, 3/5]]$ surface code over a phase flip channel. The curves refer to: the BD decoding performance (1); the upper bound (11) with the exact value from Tab. 2; the matching (8) using $(\beta_3, \beta_4, \beta_5, \beta_6, \beta_7) = (0.92, 0.76, 0.59, 0.52, 0.49)$. The dash-dotted curve represents the asymptotic lower bound from [80].

$[[13, 1, 3]]$ surface code over a depolarizing channel, and for the $[[23, 1, 3/5]]$ surface code over a phase flip channel. In Fig. 7 we also report the asymptotic lower bound, valid only over phase flip channel, from [80].

Specifically, for the $[[13, 1, 3]]$ surface code over depolarizing channel, we plot (8) using $(\beta_2, \beta_3, \beta_4, \beta_5, \beta_6) = (0.76, 0.48, 0.48, 0.46, 0.5)$ where (β_2, β_3) are computed by exhaustive search according to (10), while $(\beta_4, \beta_5, \beta_6)$ are approximated using the values computed for $A \to \infty$. The β_j with $j > 6$ are set to zero. Similarly, for the $[[23, 1, 3/5]]$ surface code over a phase flip channel, we used $(\beta_3, \beta_4, \beta_5, \beta_6, \beta_7) = (0.92, 0.76, 0.59, 0.52, 0.49)$, and $\beta_j = 0$ for $j > 7$. In both instances, we have seen that with five coefficients the resulting curves exhibited already a close correspondence to the actual logical error rate across all values of physical error probability, while computing the others β_j would be computationally expensive without appreciable improvements in the approximation. In the figures, we also plot their upper bound (11), with $\beta_2 = 0.76$ and $\beta_3 = 0.92$ for the $[[13, 1, 3]]$ and the $[[23, 1, 3/5]]$ codes, respectively. Furthermore, in the case of the $[[13, 1, 3]]$ surface code, we furnish both the computed bounded distance performance using (1) and the upper bound on the logical error rate, given by (11) with (31). It's noteworthy that the straightforward upper bound obtained solely through the WE exhibits a close alignment with the exact performance. As expected, our approximations are tight for $\rho \ll 1$, allowing to estimate logical error rates not achievable by Monte Carlo simulations.

6) Noisy syndrome extraction performance analysis. In Fig. 8 and Fig. 9, for the $[[13, 1, 3]]$ surface code, we compare the simulated logical error rates under noisy syndrome extraction with the upper bounds described in Section 5. In Fig. 8 we set the error rates as $\rho_{2Q} = \rho_{1Q} = \rho_{\text{init}} = \rho_{\text{init}}^{\text{cat}} = \rho_{\text{meas}} = \rho/100$, while in Fig. 9 we set $\rho_{2Q} = \rho_{1Q} = \rho_{\text{init}} = \rho_{\text{init}}^{\text{cat}} = \rho_{\text{meas}} = \rho/10$. Furthermore, we set the Steane ancillary state preparation error $\rho_a = \rho/10$, or $\rho_a = \rho/2$, to ensure that the ancillary resource state employed for syndrome extraction is more reliable than the data state it is used to correct. To achieve fault-tolerant syndrome extraction, one strategy is to repeatedly measure all stabilizer generators until the same syndrome is obtained $t+1$ times consecutively. Considering that up to t errors may occur during this process, in the worst-case scenario, $(t+1)^2$ measurement rounds are required [9, 83]. Hence, we choose $r = (t+1)^2 = 4$, while $\delta_{\max} = 4$, since some qubits participate in two $\boldsymbol{X}$ and two $\boldsymbol{Z}$ generators. For weight-four generators, two flags ancillas are needed for fault detection. Setting $\beta_2 = 0.76$, we compute upper bounds on the log-

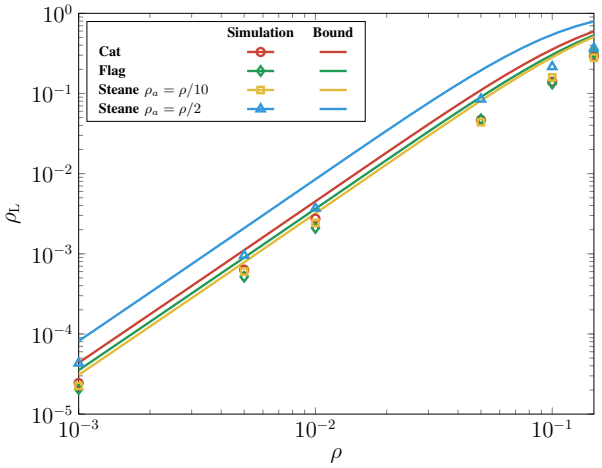

Figure 8: Logical error rates considering noisy syndrome extraction: comparison between theoretical analysis and simulation using the MWPM decoder for the $[[13, 1, 3]]$ surface code over a depolarizing channel, assuming uniform noise parameters $\rho_{2Q} = \rho_{1Q} = \rho_{\text{init}} = \rho_{\text{init}}^{\text{cat}} = \rho_{\text{meas}} = \rho/100$.

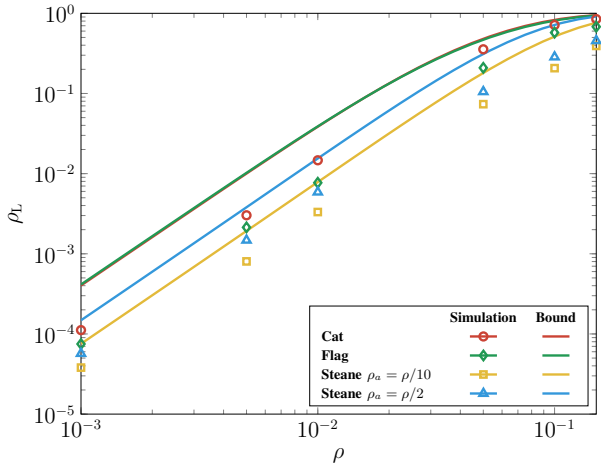

Figure 9: Logical error rates considering noisy syndrome extraction: comparison between theoretical analysis and simulation using the MWPM decoder for the $[[13, 1, 3]]$ surface code over a depolarizing channel, assuming uniform noise parameters $\rho_{2Q} = \rho_{1Q} = \rho_{\text{init}} = \rho_{\text{init}}^{\text{cat}} = \rho_{\text{meas}} = \rho/10$.

ical error rate by applying equations (48), (50), and (55) in (11), for the various gadgets discussed in Section 5. From our analysis, we observe that the proposed upper bounds are tight with respect to the numerical simulations. For circuit error rates equal to $\rho/100$ and $\rho_a = \rho/10$, the three techniques exhibit comparable performance. In particular, the effectiveness of the Steane syndrome extraction gadget is highly dependent on the fidelity of the resource state used in the procedure. Specifically, when $\rho_a = \rho/10$, it achieves the lowest logical error rates among the considered gadgets; however, for $\rho_a = \rho/2$, its performance deteriorates significantly. In contrast, the performance of the cat and flag syndrome extraction gadgets depends strongly on the circuit error rates: when these are set to $\rho/10$, the resulting logical error rates are substantially higher.

In Fig. 10, we present a comparison of simulated logical error rates for the $[[13, 1, 3]]$ surface code under noisy syndrome extraction with realistic noise parameters, alongside the upper bounds introduced in Section 5. Based on recent progress in trapped-ion technology, which exhibits some of the lowest physical error rates, we set the relevant error parameters to $\rho_{2Q} = \rho_{\text{init}} = \rho_{\text{init}}^{\text{cat}} = \rho_{\text{meas}} = 1 \cdot 10^{-4}$, and $\rho_{1Q} = 2 \cdot 10^{-5}$ [90–92]. Additionally, we take $\rho_a = 10\rho_{2Q}$ to account for the gates required in the preparation of the ancillary resource state used in Steane syndrome extraction [93]. We observe that quantum error correction remains effective only as long as gate error rates are significantly lower than the errors it is designed to protect against.

7 Conclusions

The aim of this work was to propose theoretical bounds on the error correction capability of CSS stabilizer quantum codes, providing guidance for the development of future fault-tolerant quantum systems once the physical qubit error rate and required reliability are established. These bounds were obtained starting from the weight enumerator for undetectable errors, which was derived starting from the quantum MacWilliams identities. Furthermore, we proposed a method to achieve even more stringent bounds by analyzing the structure of the logical operators for a specific stabilizer code. As an example of application of the method, we examined the performance of some stabilizer codes with MW decoding, as well as surface codes with MWPM decoding, comparing the theoretical results with simulations on both symmetric and asymmetric quantum channels. Notably, we have made advances in understanding quantum degeneracy. Indeed, we have shown that the asymptotic degeneracy of a quantum code is strongly related to the fraction of errors of weight $t + 1$ that are shared between more than one logical operator of weight $2t + 1$ and $2t + 2$. For instance, we can observe

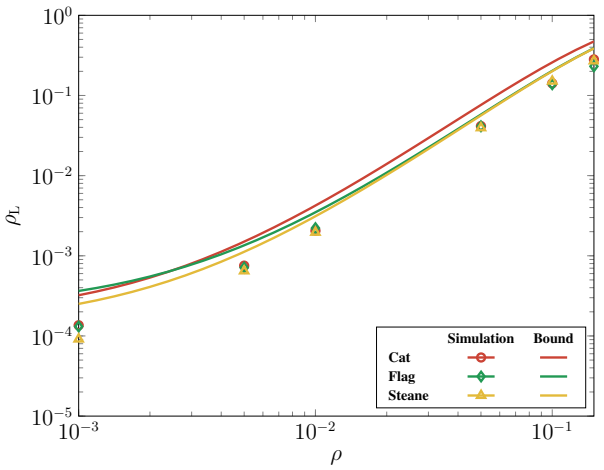

Figure 10: Logical error rates considering noisy syndrome extraction: comparison between theoretical analysis and simulation using the MWPM decoder for the $[[13, 1, 3]]$ surface code over a depolarizing channel, assuming noise parameters $\rho_{2Q} = \rho_{\text{init}} = \rho_{\text{init}}^{\text{cat}} = \rho_{\text{meas}} = 10^{-4}$, $\rho_{1Q} = 2 \cdot 10^{-5}$, and $\rho_a = 10\rho_{2Q}$.

that the Steane code, while being degenerate, can be regarded asymptotically non degenerate. This is because there are no logical operators of weight $2t + 1$ overlapping with Pauli patterns of weight $t+1$. Conversely, the Shor code is strongly asymptotically degenerate, as a number of $\boldsymbol{ZZZ}$ logical operators share a pair of Pauli $\boldsymbol{Z}$. This is closely tied to the performance of these codes. Indeed, the Shor code, being able to correct a larger number of errors beyond the code's distance, exhibits superior error correction capability. This fact provides a useful criterion in the design of quantum codes where logical operators of minimum weight share error patterns greater than the guaranteed error correction capability of the code. Finally, by extending the analysis to include realistic noisy syndrome extraction circuits, we provide a method for bounding logical error rates under circuit-level noise models. This approach to the theoretical analysis of noisy circuits paves the way for an in-depth evaluation of fault-tolerant quantum error correction performance.

Acknowledgment

This work was partially supported by the European Union under the Italian National Recovery and Resilience Plan (NRRP) of NextGenerationEU, HPC National Centre for HPC, Big Data and Quantum Computing (CN00000013).

Appendix

Here we provide the detailed derivation of (27). Specifically, for a general CSS code, a logical operator made of $\boldsymbol{Y}$ Pauli operators can be caused only by channel errors of weight $t + 1$ composed exclusively by $\boldsymbol{Y}$ operators. This is a consequence to the fact that half of the generators consists of $\boldsymbol{X}$ Pauli operators while the other half of $\boldsymbol{Z}$ operators. For instance, considering a CSS code of distance $d = 3$, we have that one channel error of weight $t+1 = 2$ that is not exclusively composed by $\boldsymbol{Y}$ Pauli operators, such as $\boldsymbol{ZY}$, will never cause a $\boldsymbol{YYY}$ logical operator. In fact, the $\boldsymbol{X}$ error is always corrected by a code of distance $d = 3$, while the remaining $\boldsymbol{ZZ}$ error will cause a logical operator composed by $\boldsymbol{ZZZ}$. Furthermore, assuming no knowledge about the fraction of $\boldsymbol{Y}$ logical operators, we conclude that a generic logical operator of weight $w = 2t+1$ or $w = 2t+2$ can be caused by 2^{t+1} different Pauli errors of weight $t + 1$. Moreover, a CSS dual-containing quantum code is characterized by a set of generators made of all $\boldsymbol{X}$ or $\boldsymbol{Z}$ Pauli operators applied to the same qubits. As a result, logical operators of minimum weight are made of only one kind of Pauli, i.e., there is an equal number of logical operator composed only of $\boldsymbol{X}$, of $\boldsymbol{Z}$, or of $\boldsymbol{Y}$. Therefore, for the CSS dual-containing, two third of logical operators made either of $\boldsymbol{X}$ or $\boldsymbol{Z}$ Pauli matrices are caused by $2^{t+1} - 1$ different Pauli errors, where the -1 is to exclude the error composed by $t + 1$ $\boldsymbol{Y}$ operators. Note that this kind of error is the only one that can cause a logical operator composed of only $\boldsymbol{Y}$ Pauli operators.

References

[1] H Jeff Kimble. "The quantum internet". Nature **453**, 1023 (2008).

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

[3] Thomas Ndousse-Fetter, Nicholas Peters, Warren Grice, Prem Kumar, Tom Chapuran, Saikat Guha, Scott Hamilton, Inder Monga, Ray Newell, Andrei Nomerotski, Don Towsley, and Ben Yoo. "Quantum networks for open science" (2019). arXiv:1910.11658.

[4] Emily Grumbling and Mark Horowitz, editors. "Quantum computing: Progress and prospects". The National Academies Press. Washington, DC (2019).

[5] Stephanie Wehner, David Elkouss, and Ronald Hanson. "Quantum internet: A vision for the road ahead". Science**362** (2018).

[6] Angela Sara Cacciapuoti, Marcello Caleffi, Rodney Van Meter, and Lajos Hanzo. "When entanglement meets classical communications: Quantum teleportation for the quantum Internet". IEEE Trans. Commun. **68**, 3808–3833 (2020).

[7] Peter W. Shor. "Scheme for reducing decoherence in quantum computer memory". Phys. Rev. A **52**, R2493–R2496 (1995).

[8] Raymond Laflamme, Cesar Miquel, Juan Pablo Paz, and Wojciech Hubert Zurek. "Perfect quantum error correcting code". Phys. Rev. Lett. **77**, 198 (1996).

[9] D. Gottesman. "Class of quantum error-correcting codes saturating the quantum Hamming bound". Phys. Rev. A**54:1862** (1996).

[10] Emanuel Knill and Raymond Laflamme. "Theory of quantum error-correcting codes". Phys. Rev. A **55**, 900–911 (1997).

[11] Andrew S. Fletcher, Peter W. Shor, and Moe Z. Win. "Structured near-optimal channel-adapted quantum error correction". Phys. Rev. A **77**, 012320 (2008).

[12] Michael A. Nielsen and Isaac L. Chuang. "Quantum computation and quantum information: 10th anniversary edition". Cambridge University Press. (2010).

[13] Junaid ur Rehman and Hyundong Shin. "Entanglement-free parameter estimation of generalized Pauli channels". Quantum **5**, 490 (2021).

[14] A. M. Steane. "Error correcting codes in quantum theory". Phys. Rev. Lett. **77**, 793–797 (1996).

[15] Daniel Gottesman. "An introduction to quantum error correction and fault-tolerant quantum computation" (2009). arXiv:0904.2557.

[16] Barbara M. Terhal. "Quantum error correction for quantum memories". Rev. Mod. Phys. **87**, 307–346 (2015).

[17] Sreraman Muralidharan, Linshu Li, Jungsang Kim, Norbert Lütkenhaus, Mikhail D Lukin, and Liang Jiang. "Optimal architectures for long distance quantum communication". Scientific reports **6**, 20463 (2016).

[18] Joschka Roffe. "Quantum error correction: an introductory guide". Contemporary Physics **60**, 226–245 (2019).

[19] Z. Babar, D. Chandra, H. V. Nguyen, P. Botsinis, D. Alanis, S. X. Ng, and L. Hanzo. "Duality of quantum and classical error correction codes: Design principles and examples". Commun. Surveys Tuts. **21**, 970–1010 (2019).

[20] Marco Chiani, Andrea Conti, and Moe Z. Win. "Piggybacking on quantum streams". Physical Review A**102** (2020).

[21] Fabio Zoratti, Giacomo De Palma, Bobak Kiani, Quynh T Nguyen, Milad Marvian, Seth Lloyd, and Vittorio Giovannetti. "Improving the speed of variational quantum algorithms for quantum error correction". Physical Review A **108**, 022611 (2023).

[22] Dolev Bluvstein, Simon J Evered, Alexandra A Geim, Sophie H Li, Hengyun Zhou, Tom Manovitz, Sepehr Ebadi, Madelyn Cain, Marcin Kalinowski, Dominik Hangleiter, et al. "Logical quantum processor based on reconfigurable atom arrays". NaturePages 1–3 (2023).

[23] Earl T Campbell, Barbara M Terhal, and Christophe Vuillot. "Roads towards fault-tolerant universal quantum computation". Nature **549**, 172–179 (2017).

[24] Theodore J Yoder and Isaac H Kim. "The surface code with a twist". Quantum **1**, 2 (2017).

[25] Hans J Briegel, David E Browne, Wolfgang Dür, Robert Raussendorf, and Maarten Van den Nest. "Measurement-based quantum computation". Nature Physics **5**, 19–26 (2009).

[26] Austin G. Fowler, Matteo Mariantoni, John M. Martinis, and Andrew N. Cleland.

"Surface codes: Towards practical large-scale quantum computation". Physical Review A**86** (2012).

[27] Dimiter Ostrev, Davide Orsucci, Francisco Lázaro, and Balazs Matuz. "Classical product code constructions for quantum Calderbank-Shor-Steane codes". Quantum **8**, 1420 (2024).

[28] Lev Ioffe and Marc Mézard. "Asymmetric quantum error-correcting codes". Physical Review A **75**, 032345 (2007).

[29] Pradeep Kiran Sarvepalli, Andreas Klappenecker, and Martin Rötteler. "Asymmetric quantum codes: constructions, bounds and performance". Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences **465**, 1645–1672 (2009).

[30] David Layden, Mo Chen, Paola Cappellaro, et al. "Efficient quantum error correction of dephasing induced by a common fluctuator". Phys. Rev. Lett. **124**, 020504 (2020).

[31] Marco Chiani and Lorenzo Valentini. "Short codes for quantum channels with one prevalent Pauli error type". IEEE J. Sel. Areas Inf. T. **1**, 480–486 (2020).

[32] D.P. DiVincenzo, D.W. Leung, and B.M. Terhal. "Quantum data hiding". IEEE Transactions on Information Theory **48**, 580–598 (2002).

[33] Joseph Emerson, Robert Alicki, and Karol Życzkowski. "Scalable noise estimation with random unitary operators". Journal of Optics B: Quantum and Semiclassical Optics **7**, S347 (2005).

[34] Christoph Dankert, Richard Cleve, Joseph Emerson, and Etera Livine. "Exact and approximate unitary 2-designs and their application to fidelity estimation". Physical Review A**80** (2009).

[35] Antonio deMarti iOlius, Patricio Fuentes, Román Orús, Pedro M. Crespo, and Josu Etxezarreta Martinez. "Decoding algorithms for surface codes". Quantum **8**, 1498 (2024).

[36] M. Silva, E. Magesan, D. W. Kribs, and J. Emerson. "Scalable protocol for identification of correctable codes". Phys. Rev. A **78**, 012347 (2008).

[37] Josu Etxezarreta Martinez, Patricio Fuentes, Pedro M. Crespo, and J. Garcia-Frias. "Approximating decoherence processes for the design and simulation of quantum error correction codes on classical computers". IEEE Access **8**, 172623–172643 (2020).

[38] S. B. Bravyi and A. Yu. Kitaev. "Quantum codes on a lattice with boundary" (1998). arXiv:quant-ph/9811052.

[39] Rajeev Acharya, Igor Aleiner, Richard Allen, Trond I Andersen, Markus Ansmann, Frank Arute, Kunal Arya, Abraham Asfaw, Juan Atalaya, Ryan Babbush, et al. "Suppressing quantum errors by scaling a surface code logical qubit". Nature **614**, 676–681 (2023).

[40] Sebastian Krinner, Nathan Lacroix, Ants Remm, Agustin Di Paolo, Elie Genois, Catherine Leroux, Christoph Hellings, Stefania Lazar, Francois Swiadek, Johannes Herrmann, et al. "Realizing repeated quantum error correction in a distance-three surface code". Nature **605**, 669–674 (2022).

[41] Youwei Zhao, Yangsen Ye, He-Liang Huang, Yiming Zhang, Dachao Wu, Huijie Guan, et al. "Realization of an error-correcting surface code with superconducting qubits". Phys. Rev. Lett. **129**, 030501 (2022).

[42] Craig Gidney. "Stim: a fast stabilizer circuit simulator". Quantum **5**, 497 (2021).

[43] Scott Aaronson and Daniel Gottesman. "Improved simulation of stabilizer circuits". Phys. Rev. A **70**, 052328 (2004).

[44] John Preskill. "Reliable quantum computers". Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences **454**, 385–410 (1998).

[45] Sergey Bravyi, Martin Suchara, and Alexander Vargo. "Efficient algorithms for maximum likelihood decoding in the surface code". Physical Review A **90**, 032326 (2014).

[46] J. Pablo Bonilla Ataides, David K. Tuckett, Stephen D. Bartlett, Steven T. Flammia, and Benjamin J. Brown. "The XZZX surface code". Nature Communications**12** (2021).

[47] Alexei E Ashikhmin, Alexander M Barg, Emanuel Knill, and Simon N Litsyn. "Quantum error detection. I. statement of the prob-

lem". IEEE Trans. Inf. Theory **46**, 778–788 (2000).

[48] Alexei E Ashikhmin, Alexander M Barg, Emanuel Knill, and Simon N Litsyn. "Quantum error detection. II. bounds". IEEE Trans. Inf. Theory **46**, 789–800 (2000).

[49] A Robert Calderbank and Peter W Shor. "Good quantum error-correcting codes exist". Physical Review A **54**, 1098 (1996).

[50] Andrew Steane. "Multiple-particle interference and quantum error correction". Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences **452**, 2551–2577 (1996).

[51] David Poulin. "Optimal and efficient decoding of concatenated quantum block codes". Physical Review A **74**, 052333 (2006).

[52] Min-Hsiu Hsieh and François Le Gall. "Np-hardness of decoding quantum error-correction codes". Physical Review A **83**, 052331 (2011).

[53] Pavithran Iyer and David Poulin. "Hardness of decoding quantum stabilizer codes". IEEE Trans. Inf. Theory **61**, 5209–5223 (2015).

[54] Oscar Higgott and Craig Gidney. "Sparse Blossom: correcting a million errors per core second with minimum-weight matching". Quantum **9**, 1600 (2025).

[55] Tristan Müller, Thomas Alexander, Michael E. Beverland, Markus Bühler, Blake R. Johnson, Thilo Maurer, and Drew Vandeth. "Improved belief propagation is sufficient for real-time decoding of quantum memory" (2025). arXiv:2506.01779.

[56] Jack Edmonds. "Paths, trees, and flowers". Canadian Journal of mathematics **17**, 449–467 (1965).

[57] Eric Dennis, Alexei Kitaev, Andrew Landahl, and John Preskill. "Topological quantum memory". Journal of Mathematical Physics **43**, 4452–4505 (2002).

[58] Oscar Higgott. "Pymatching: A python package for decoding quantum codes with minimum-weight perfect matching" (2021).

[59] Luka Skoric, Dan E. Browne, Kenton M. Barnes, Neil I. Gillespie, and Earl T. Campbell. "Parallel window decoding enables scalable fault tolerant quantum computation". Nature Communications **14**, 7040 (2023).

[60] Andrew M Steane. "Error correcting codes in quantum theory". Phys. Rev. Lett. **77**, 793 (1996).

[61] Raymond Laflamme, Cesar Miquel, Juan Pablo Paz, and Wojciech Hubert Zurek. "Perfect quantum error correcting code". Phys. Rev. Lett. **77**, 198 (1996).

[62] Clare Horsman, Austin G Fowler, Simon Devitt, and Rodney Van Meter. "Surface code quantum computing by lattice surgery". New Journal of Physics **14**, 123011 (2012).

[63] Utkarsh Azad, Aleksandra Lipiń ska, Shilpa Mahato, Rijul Sachdeva, Debasmita Bhoumik, and Ritajit Majumdar. "Surface code design for asymmetric error channels". IET Quantum Communication **3**, 174–183 (2022).

[64] Peter Shor and Raymond Laflamme. "Quantum analog of the MacWilliams identities for classical coding theory". Phys. Rev. Lett. **78**, 1600 (1997).

[65] Eric M Rains. "Quantum weight enumerators". IEEE Trans. Inf. Theory **44**, 1388–1394 (1998).

[66] ChunJun Cao and Brad Lackey. "Quantum weight enumerators and tensor networks". IEEE Trans. Inf. TheoryPages 1–1 (2023).

[67] F.J. MacWilliams and N.J.A. Sloane. "The theory of error-correcting codes". Volume 16. North Holland Mathematical Libray. Amsterdam, the Netherlands (1977).

[68] A. R. Calderbank, E. M Rains, P. W. Shor, and N. J. A. Sloane. "Quantum error correction via codes over GF(4)". IEEE Trans. Inf. Theory **44**, 1369–1387 (1998).

[69] Elwyn Berlekamp, Robert McEliece, and Henk Van Tilborg. "On the inherent intractability of certain coding problems (corresp.)". IEEE Trans. Inf. Theory **24**, 384–386 (1978).

[70] T Aaron Gulliver, Vijay K Bhargava, and Jack M Stein. "Q-ary gray codes and weight distributions". Applied mathematics and computation **103**, 97–109 (1999).

[71] Iliya Bouyukliev, Stefka Bouyuklieva, Tatsuya Maruta, and Paskal Piperkov. "Characteristic vector and weight distribution of a linear code". Cryptography and Communications **13**, 263–282 (2021).

[72] K.H. Zimmermann. "Integral hecke modules, integral generalized reed-muller codes, and linear codes". Berichte des Forschungsschwerpunktes Informations- und Kommunikationstechnik. Techn. Univ. Hamburg-Harburg. (1996). url: http://hdl.handle.net/11420/9651.

[73] Anton Betten, Michael Braun, Harald Fripertinger, Adalbert Kerber, Axel Kohnert, and Alfred Wassermann. "Error-correcting linear codes: Classification by isometry and applications". Volume 18. Springer Science & Business Media. (2006).

[74] Anne Canteaut and Florent Chabaud. "A new algorithm for finding minimum-weight words in a linear code: Application to McEliece's cryptosystem and to narrow-sense bch codes of length 511". IEEE Trans. Inf. Theory **44**, 367–378 (1998).

[75] Markus Grassl. "Searching for linear codes with large minimum distance". In Discovering Mathematics with Magma: Reducing the Abstract to the Concrete. Pages 287–313. Springer (2006).

[76] Petr Lisonek and Layla Trummer. "Algorithms for the minimum weight of linear codes.". Adv. Math. Commun. **10**, 195–207 (2016).

[77] Wieb Bosma, John Cannon, and Catherine Playoust. "The Magma algebra system I: The user language". Journal of Symbolic Computation **24**, 235–265 (1997).

[78] ChunJun Cao, Michael J. Gullans, Brad Lackey, and Zitao Wang. "Quantum lego expansion pack: Enumerators from tensor networks". PRX Quantum **5**, 030313 (2024).

[79] D.J.C. MacKay, G. Mitchison, and P.L. McFadden. "Sparse-graph codes for quantum error correction". IEEE Trans. Inf. Theory **50**, 2315–2330 (2004).

[80] Jonghyun Lee, Jooyoun Park, and Jun Heo. "Rectangular surface code under biased noise". Quantum Information Processing **20**, 1–16 (2021).

[81] Fern HE Watson and Sean D Barrett. "Logical error rate scaling of the toric code". New Journal of Physics **16**, 093045 (2014).

[82] Balázs Dezső, Alpár Jüttner, and Péter Kovács. "Lemon–an open source C++ graph template library". Electronic notes in theoretical computer science **264**, 23–45 (2011).

[83] P.W. Shor. "Fault-tolerant quantum computation". In Proceedings of 37th Conference on Foundations of Computer Science. Pages 56–65. (1996).

[84] Yifan Hong. "Single-shot preparation of hypergraph product codes via dimension jump". Quantum **9**, 1879 (2025).

[85] Rui Chao and Ben W. Reichardt. "Quantum error correction with only two extra qubits". Physical Review Letters**121** (2018).

[86] Christopher Chamberland and Michael E. Beverland. "Flag fault-tolerant error correction with arbitrary distance codes". Quantum **2**, 53 (2018).

[87] A. M. Steane. "Active stabilization, quantum computation, and quantum state synthesis". Physical Review Letters **78**, 2252–2255 (1997).

[88] Google Quantum AI et al. "Quantum error correction below the surface code threshold". Nature **638**, 920 (2024).

[89] Prithviraj Prabhu and Ben W. Reichardt. "Fault-tolerant syndrome extraction and cat state preparation with fewer qubits". Quantum **7**, 1154 (2023).

[90] K. R. Brown, A. C. Wilson, Y. Colombe, C. Ospelkaus, A. M. Meier, E. Knill, D. Leibfried, and D. J. Wineland. "Single qubit gate error below 10^{-4} in a trapped ion". Physical Review A **84**, 030303 (2011).

[91] Stephen D. Erickson, Jenny J. Wu, Pan-Yu Hou, Daniel C. Cole, Shawn Geller, Alex Kwiatkowski, Scott Glancy, Emanuel Knill, Daniel H. Slichter, Andrew C. Wilson, and Dietrich Leibfried. "High-fidelity indirect readout of trapped-ion hyperfine qubits". Physical Review Letters**128** (2022).

[92] A. C. Hughes, R. Srinivas, C. M. Löschnauer, H. M. Knaack, R. Matt, C. J. Ballance, M. Malinowski, T. P. Harty, and R. T. Sutherland. "Trapped-ion two-qubit gates

with >99.99% fidelity without ground-state cooling" (2025). arXiv:2510.17286.

[93] Diego Forlivesi and David Amaro. "Flag at origin: a modular fault-tolerant preparation for css codes" (2025). arXiv:2508.14200.