On maximum-likelihood decoding with circuit-level errors

Leonid P. Pryadko
Department of Physics & Astronomy, University of California, Riverside, California 92521, USA
(Dated: July 15, 2020)

Error probability distribution associated with a given Clifford measurement circuit is described exactly in terms of the circuit error-equivalence group, or the circuit subsystem code previously introduced by Bacon, Flammia, Harrow, and Shi. This gives a prescription for maximum-likelihood decoding with a given measurement circuit. Marginal distributions for subsets of circuit errors are also analyzed; these generate a family of related asymmetric LDPC codes of varying degeneracy. More generally, such a family is associated with any quantum code. Implications for decoding highly-degenerate quantum codes are discussed.

I. INTRODUCTION

Quantum computation offers exponential algorithmic speed-up for some classically hard problems. This promise is conditional in a fundamental way upon the use of quantum error correction (QEC). However, despite an enormous progress achieved both in theory and experiment during the quarter century since the invention of QEC[1], universal repetitive QEC protecting against both phase and amplitude errors has not yet been demonstrated in any of the qubit systems constructed to date. This illustrates the enormous difficulty of building quantum computers (QCs) with sufficiently small error rates.

Given also the engineering difficulties with scaling up the number of qubits[2], it is important that on the algorithmic side one tries to achieve every optimization possible. Among other things, we would like to maximize the probability of successful syndrome-based decoding with QEC. Given the available hardware, this requires choosing the optimal code, the optimal measurement circuit, and the optimal decoder. In particular, a decoder should be designed for the specific syndrome measurement circuit, as it must be aware of the associated correlations between the errors[3–5]—at sufficiently high error probabilities such correlations are present even in fault-tolerant (FT) circuits designed to prevent a single fault from affecting multiple data qubits.

In contrast, the standard approach is to use a decoder optimized for the underlying code, regardless of the actual circuit used for syndrome measurement. In some cases, e.g., in the case of surface codes with minimum-weight perfect matching (MWPM) decoder, some leading-order correlations can be included in the edge weights of the graph used for decoding[4, 6–8]. However, such a scheme is limited to codes and measurement circuits where MWPM can be done efficiently, e.g., surface codes with single-ancilla measurement circuits[9], and certain other classes or topological codes[10, 11]. Further, there is necessarily a decoding accuracy loss when measurement circuit is not simply repeated, e.g., with code deformation or lattice surgery[12].

A feasible approach to building a decoder optimized for the specific measurement circuit is to train a neural network (NN) using extensive simulation data[13–20]. However, as is commonly the case with NNs, there is always a question whether training has been sufficient to achieve optimal decoding performance.

A parameterization of the probability distribution of correlated quantum errors in terms of a spin model has been recently considered by Chubb and Flammia[5]. In particular, they describe how such a formalism can be used for maximum-likelihood (ML) decoding in the presence of circuit-level noise. However, Chubb and Flammia focused on larger circuits composed of measurement blocks. Errors are assumed uncorrelated between the blocks, while a model of error correlations at the output of each block has to be constructed offline, e.g., using error propagation for a Clifford measurement circuit. In particular, Chubb and Flammia stopped short of analyzing circuit-level errors, and only considered numerically a toy model of correlated errors.

The goal of this work is to give an explicit numerically efficient algorithm for analyzing error correlations resulting from a given qubit-based Clifford measurement circuit, and for constructing decoders optimized for such a circuit. Main result is that such correlations can be accounted for by using phenomenological error model (no error correlations) with the circuit-associated subsystem code constructed by Bacon, Flammia, Harrow, and Shi[21, 22]. Thus, any generic decoder capable of dealing with uncorrelated data and syndrome measurement errors in highly-degenerate sparse-generator subsystem codes can be rendered to account for circuit-level error correlations. Error correlations needed for implementing the scheme of Chubb and Flammia are recovered by calculating the marginals of the constructed distribution, which can be done in practice using Ising model star-polygon transformations[23]. The construction naturally includes additional correlations between errors in different fault locations on the circuit.

An immediate application is for designing decoders approaching true ML decoding for Clifford circuits, to be used in quantum memory. In particular, more accurate decoding and optimization, with the ability to account for error rates' variations on individual qubits, could help improve QEC to the level sufficient to pass the break-even point and show coherent lifetimes longer than that of an unprotected qubit in present-day or near-future devices. Related techniques could also be more

widely applicable for design and analysis of FT protocols and control schemes. Examples include error correction with FT gadgets like flag error correction[24–26], schemes for protected evolution, e.g., using code deformations where conventional approaches may result in a reduced distance[12], and optimized single-shot error-correction protocols[27–29].

In addition, analysis of marginal error distributions and associated families of asymmetric codes of varying degeneracy could become an important tool in the theory of QECCs. In particular, it could help constructing better decoders for highly-degenerate quantum LDPC codes. Also, thorough analytical understanding of error correlations could be useful for fundamental analysis of thresholds, e.g., in order to extend and/or improve the bounds in Refs. 30 and 31.

Paper outline: After this Introduction, an overview of relevant notations and results in quantum codes and multi-variable Bernoulli distributions is given in Sec. II. Pauli error channels with correlated errors, including circuit-level correlations, are discussed in Sec. III. Corresponding marginal distributions are constructed in Sec. IV, followed by a discussion of possible implications for exact and approximate ML decoding with circuit-level errors in Sec. V. Sec. VI gives the concluding remarks.

II. BACKGROUND

A. Quantum codes

Generally, an n-qubit quantum code $\mathcal{Q}$ is a subspace of the n-qubit Hilbert space $\mathbb{H}_2^{\otimes n}$. A quantum $[[n,k,d]]$ stabilizer code is a 2^k-dimensional subspace specified as a common $+1$ eigenspace of all operators in an Abelian *stabilizer* group $\mathcal{S} \subset \mathcal{P}_n$, $-\mathbb{1} \notin \mathcal{S}$, where the n-qubit Pauli group $\mathcal{P}_n$ is generated by tensor products of single-qubit Pauli operators. The stabilizer is typically specified in terms of its generators, $\mathcal{S} = \langle S_1, \ldots, S_{n-k} \rangle$. The weight of a Pauli operator is the number of qubits that it affects. The distance d of a quantum code is the minimum weight of a Pauli operator $E \in \mathcal{P}_n$ which commutes with all operators from the stabilizer $\mathcal{S}$, but is not a part of the stabilizer, $E \notin \mathcal{S}$. Such operators act non-trivially in the code and are called logical operators.

Subsystem codes[32, 33] are a generalization of stabilizer codes where only some of the logical qubits are used. More precisely, given a stabilizer group $\mathcal{S}$, the stabilized subspace $\mathcal{Q}_\mathcal{S}$ is further decomposed into a tensor product of two subspaces, $\mathcal{Q}_S = \mathcal{Q}_L \otimes \mathcal{Q}_G$, where $\mathcal{Q}_L$ is a 2^k-dimensional "logical" subspace used to store the quantum information, while we do not care about the state of the "gauge" qubits in the subspace $\mathcal{Q}_G$ after the recovery. Logical operators of the original code which act non-trivially only in $\mathcal{Q}_G$, together with the operators from the stabilizer group $\mathcal{S}$, generate the non-Abelian gauge group $\mathcal{G}$ which fully characterizes the subsystem code. In particular, the center $Z(\mathcal{G})$ is formed by the elements of the original stabilizer $\mathcal{S}$, up to a phase i^m, $m \in \{0,1,2,3\}$.

A Pauli error E that anticommutes with an element of the stabilizer, $ES = -SE$, $S \in \mathcal{S}$, is called detectable. Such an error results in a non-zero syndrome $\mathbf{s} = \{s_1, \ldots, s_r\}$ whose bits $s_i \in \{0,1\}$ are obtained by measuring the eigenvalues $(-1)^{s_i}$ of the chosen set of stabilizer generators S_i, $i = \{1, \ldots, r\}$, $r \geq n-k$. Unlike in the case of classical codes, there may be many equivalent (*mutually degenerate*) errors resulting in the same syndrome. For a subsystem code, errors E' degenerate with E, denoted $E' \simeq E$, have the form $E' = EG$, where $G \in \mathcal{G}$ is an element of the gauge group; such errors can not and need not be distinguished. Non-trivial logical operators of the subsystem code are logical operators of the original stabilizer code which act non-trivially in $\mathcal{Q}_L$. A *bare* logical operator U acts trivially in $\mathcal{Q}_G$ and commutes with any element of $\mathcal{G}$. Such restrictions do not apply to *dressed* logical operators which are only required to commute with elements of the stabilizer $\mathcal{S}$. In any case, multiplication by a logical operator U gives an error with the same syndrome but from a different equivalence class, $EU \not\simeq E$.

Analysis of error correction is conveniently done using quaternary, or an equivalent binary, representation of the Pauli group[34, 35]. A Pauli operator $U \equiv i^m X^{\mathbf{u}} Z^{\mathbf{v}}$, where $\mathbf{u}, \mathbf{v} \in \{0,1\}^{\otimes n}$ and $X^{\mathbf{u}} = X_1^{u_1} X_2^{u_2} \ldots X_n^{u_n}$, $Z^{\mathbf{v}} = Z_1^{v_1} Z_2^{v_2} \ldots Z_n^{v_n}$, is mapped, up to a phase, to a length-$2n$ binary vector $\mathbf{e} = (\mathbf{u}|\mathbf{v})$. Two Pauli operators U_1, U_2 commute iff the symplectic inner product

$$\mathbf{e}_1 \star \mathbf{e}_2^T \equiv \mathbf{e}_1 \Sigma\, \mathbf{e}_2^T, \quad \Sigma \equiv \begin{pmatrix} & I_n \\ I_n & \end{pmatrix}, \tag{1}$$

of the corresponding binary vectors is zero, $\mathbf{e}_1 \star \mathbf{e}_2^T = 0$. Here I_n is the $n \times n$ identity matrix. Thus, if $H = (H_Z|H_X)$ is an $m \times 2n$ binary matrix whose rows represent stabilizer generators, and $\widetilde{H} \equiv H\,\Sigma = (H_X|H_Z)$, then the syndrome $\mathbf{s}$ of an error $U_\mathbf{e}$ with binary representation $\mathbf{e} = (\mathbf{u}|\mathbf{v})$ is given by $\mathbf{s}^T = H\mathbf{e}^T$. If we similarly denote $G = (G_X|G_Z)$ an $r \times 2n$ matrix formed by the gauge group generators (for a stabilizer code, $G = \widetilde{H}$), the errors with binary representation $\mathbf{e}$ and $\mathbf{e}' = \mathbf{e} + \boldsymbol{\alpha} G$ are equivalent, $\mathbf{e}' \simeq \mathbf{e}$, for any length-$r$ binary string $\boldsymbol{\alpha} \in \mathbb{F}_2^{\otimes m}$. Generally, $GH^T = 0$, and

$$\operatorname{rank} H = n - k - \kappa, \quad \operatorname{rank} G = n - k + \kappa, \tag{2}$$

where κ is the number of gauge qubits. Matrices G and H can be viewed as generator matrices of an auxiliary length-$2n$ CSS code which encodes $2k$ qubits. These correspond to a basis set of $2k$ independent logical operators of the original subsystem code. It will be convenient to introduce a logical generating matrix L such that $LH^T = 0$, $\operatorname{rank} L = 2k$. Rows of L map to basis logical operators; a non-zero linear combination of the rows of L must be linearly independent from the rows of G.

Gauge or stabilizer generators can be measured with a Clifford circuit which consists of ancillary qubit initial-

ization and measurement in the preferred (e.g., Z) basis, and a set of unitary Clifford gates, e.g., single-qubit Hadamard H and phase P gates and two-qubit CNOT. Generally, a Clifford unitary U maps Pauli operators to Pauli operators, $E' = U^\dagger E U$, $E, E' \in \mathcal{P}_n$. Ignoring the overall phase (for complete description, see Refs. 36 and 37), this corresponds to a linear map of the corresponding binary vectors, $(\mathbf{e}')^T = C\mathbf{e}^T$, where $C \equiv C_U$ is a symplectic matrix with the property $C^T \Sigma C = \Sigma$.

B. Bernoulli distribution

Multi-variate Bernoulli distribution describes a joint probability distribution of m single-bit variables $x_i \in \mathbb{F}_2$, e.g., components of a binary vector $\mathbf{x} \in \mathbb{F}_2^m$. Most generally, such a distribution can be specified in terms of 2^m probabilities $p_{\mathbf{x}} \geq 0$, with normalization $\sum_{\mathbf{x}\in\mathbb{F}_2^m} p_{\mathbf{x}} = 1$. A convenient representation of such a distribution as an exponent of a polynomial of m binary variables with real coefficients was given by Dai et al. in Ref. 38. Namely, for a single variable $x \in \{0, 1\}$, we can write $\mathbb{P}(x) = p_0^{1-x} p_1^x$ as a product of two terms, where $p_0 + p_1 = 1$ are the outcome probabilities. For m variables we have, similarly, the product of 2^m terms,

$$\mathbb{P}(\mathbf{x}) = p_{00\ldots0}^{(1-x_1)(1-x_2)\ldots(1-x_m)} \times p_{00\ldots01}^{(1-x_1)(1-x_2)\ldots(1-x_{m-1})x_m} \ldots p_{11\ldots1}^{x_1 x_2 \ldots x_m}, \quad (3)$$

where the probabilities are assumed positive, $p_{\mathbf{x}} > 0$. For any given $\mathbf{x} \in \mathbb{F}_2^m$ only one exponent is non-zero so that the result is $p_{\mathbf{x}}$. Taking the logarithm and expanding, one obtains the corresponding "energy" $\mathscr{E} \equiv -\ln \mathbb{P}(\mathbf{x})$ as a polynomial of m binary variables. The corresponding coefficients can be viewed as binary cumulants[38]; presence of high-degree terms indicates a complex probability distribution with highly non-trivial correlations.

For applications it is more convenient to work with spin variables $s_i = (-1)^{x_i} \equiv 1 - 2x_i \in \pm 1$, and rewrite the energy function using the general Ising representation first introduced by Wegner[39],

$$\mathscr{E} \equiv \mathscr{E}(\{s_j\}) = -\sum_b K_b R_b + \text{const}, \quad R_b = \prod_i s_i^{\theta_{ib}}, \quad (4)$$

parameterized by the binary spin-bond incidence matrix θ with m rows, and the bond coefficients K_b. While most general m-variate Bernoulli distribution requires $2^m - 1$ bonds, in the absence of high-order correlations significantly fewer terms may be needed. Of particular interest are distributions with sparse matrices θ, e.g., with bounded row and column weights, which also limits the number of columns. In the simplest case of independent identically-distributed (i.i.d.) bits with equal set probabilities $p_1 = p$, $p_0 = 1 - p$, we can take $\theta = I_m$, the identity matrix, and all coefficients equal, $K = \ln[(1-p)/p]/2$. The logarithm $\ln[(1-p)/p]$ is commonly called a log-likelihood ratio (LLR); the coefficient K here and K_b in Eq. (4) are thus called half-LLR coefficients.

III. ERROR CORRELATIONS IN A CLIFFORD MEASUREMENT CIRCUIT

A. Pauli error channel with correlations.

Consider the most general Pauli error channel

$$\rho \to \sum\nolimits_{\mathbf{e}\in\mathbb{F}_2^{2n}} \mathbb{P}(\mathbf{e}) E_{\mathbf{e}} \rho E_{\mathbf{e}}^\dagger, \quad (5)$$

where $\mathbb{P}(\mathbf{e})$ is the probability of an error $E_{\mathbf{e}}$ with binary representation $\mathbf{e}$, with the irrelevant phase disregarded. Technically, $\mathbb{P}(\mathbf{e})$ describes a $2n$-variate Bernoulli distribution. The probability $\mathbb{P}(\mathbf{e})$ can be parameterized in terms of a $2n \times m$ binary coupling matrix θ with $m < 2^{2n}$ columns, and a set of coefficients K_b, $b \in \{1, \ldots, m\}$,

$$\mathbb{P}(\mathbf{e}; \theta, \{K_b\}) = Z^{-1} \exp\left(\sum_b K_b (-1)^{[\mathbf{e}\,\theta]_b}\right), \quad (6)$$

where $[\mathbf{e}\,\theta]_b$ in the exponent is the corresponding component of the row-vector $\mathbf{e}\,\theta$. The normalization constant $Z \equiv Z(\theta, \{K_b\})$ in Eq. (6) is the partition function of the Ising model in Wegner's form, cf. Eq. (4),

$$Z(\theta, \{K_b\}) = \sum_{\{s_i \in \pm 1\}} \prod_b e^{K_b R_b}. \quad (7)$$

In the simplest case of independent X and Z errors, $\theta = I_{2n}$, the identity matrix, while $e^{2K_b} = (1 - p_X)/p_X$ for $b \leq n$, and the corresponding expression with $p_X \to p_Z$ for $n < b \leq 2n$. In the case of the depolarizing channel with error probability p, we have, instead,

$$\theta = \left(\begin{array}{c|c|c} I_n & 0 & I_n \\ \hline 0 & I_n & I_n \end{array}\right), \quad e^{4K} = 3(1-p)/p, \quad (8)$$

where the additional column block is to account for correlations between X and Z errors. Additional correlations between the errors can be introduced by adding columns to matrix θ and the corresponding coefficients K_b.

Given a probability distribution in the form (6), it is easy to construct an expression for probability of an error equivalent to $\mathbf{e}$ in a subsystem code with gauge generator matrix G, extending the approach of Refs. 6, 40, and 41, and reproducing some of the results from Ref. 5. A substitution $\mathbf{e} \to \mathbf{e} + \boldsymbol{\alpha} G$ and a summation over $\boldsymbol{\alpha}$ gives

$$\mathbb{P}(\mathbf{e}' \in \mathbb{F}_2^{2n} | \mathbf{e}' \simeq \mathbf{e}) = 2^{\operatorname{rank} G - r} \frac{Z\left(G\theta, \{K_b (-1)^{[\mathbf{e}\,\theta]_b}\}\right)}{Z\left(\theta, \{K_b\}\right)}. \quad (9)$$

Here G is an $r \times 2n$ binary matrix, cf. Eq. (2), and the prefactor accounts for a possible redundancy in the summation. Notice that the partition function in the numerator has the same number of bonds m as that in the denominator, but with the signs of the coefficients K_b corresponding to non-zero elements of the binary vector $\boldsymbol{\epsilon} \equiv \mathbf{e}\,\theta$ flipped. When both the gauge generator matrix G and the error correlation matrix θ are sparse, the incidence matrix $\Theta \equiv G\theta$ in the numerator of Eq. (9) must also be sparse.

B. Clifford measurement circuit and associated input/output codes.

Let us now consider error correlations resulting from a Clifford measurement circuit. Specifically, following Refs. 21 and 22, consider an n-qubit circuit, $n = n_0 + n_a$, with n_0 data qubits and n_a ancillary qubits. First, ancillary qubits are initialized to $|0\rangle$, second, a collection of Clifford gates forms a unitary U, and finally the ancillary qubits are measured in the Z basis, see Fig. 1. In the absence of errors and in the event of all measurements returning $+1$ (zero syndrome), the corresponding post-selected evolution is described by the matrix

$$V = (I^{\otimes n_0} \otimes \langle 0|^{\otimes n_a})\, U\, (I^{\otimes n_0} \otimes |0\rangle^{\otimes n_a}), \qquad (10)$$

where I is a single-qubit identity operator. The circuit is assumed to be a *good error-detecting circuit* (good EDC), namely, $V^\dagger V$ be proportional to the projector onto a subspace $\mathcal{Q}_0 \subseteq \mathbb{H}_2^{n_0}$,

$$V^\dagger V = c\Pi_0, \quad c > 0 \qquad (11)$$

see Def. 4 in Ref. 22. Here $\mathcal{Q}_0$, called the *input code*, is an $[[n_0, k, d_0]]$ stabilizer code encoding k qubits with distance d_0.

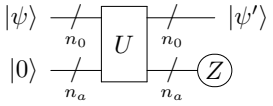

FIG. 1. Generic circuit with n_0 data and n_a ancillary qubits initialized to $|0\rangle$ and measured in Z basis. In practice, ancillary qubit can be measured during evolution, and subsequently reused after initialization. However, there is no mechanism for adapting the gates to the measurement results.

A good EDC also defines an *output code* $\mathcal{Q}_0' \subseteq \mathcal{H}_2^{\otimes n_0}$ which encodes the same number of qubits k. Indeed, since $V^\dagger V = c\Pi_0$, matrix V has only one non-zero singular value, $\sqrt{c}$; this immediately gives $VV^\dagger = c\Pi_0'$, with the projector onto a space $\mathcal{Q}_0' \subseteq \mathcal{H}_2^{\otimes n_0}$ of the same dimension 2^k, the output code. Moreover, for any input state $|\psi\rangle \in \mathcal{Q}_0$, the output $|\psi\rangle' \equiv V|\psi\rangle \in \mathcal{Q}_0'$ is in the output code, and the corresponding transformation is a (scaled) Clifford unitary.

Even though the map between $\mathcal{Q}_0$ and $\mathcal{Q}_0'$ is unitary, the distance d_0' of the output code does not necessarily equals d_0. In particular, adding a unitary decoding circuit on output data qubits may be used to render $d_0' = 1$.

C. Errors in a Clifford circuit.

Using standard circuit identities, any circuit error $\mathcal{E}$ can be propagated forward to the output of the circuit, thus giving an equivalent data error $E_0'(\mathcal{E}) \in \mathcal{P}_{n_0}$ and the (gauge) syndrome $\boldsymbol{\sigma}'(\mathcal{E}) \in \mathbb{F}_2^{n_a}$ corresponding to the measurement results. Clearly, there is a big redundancy even if phases are ignored, as many circuit errors can result in the same or equivalent $E_0'(\mathcal{E})$ and $\boldsymbol{\sigma}'(\mathcal{E})$. The goal is to find the conditional probability distribution for the equivalence class of the output error $E_0'(\mathcal{E})$ given the measured value of $\boldsymbol{\sigma}'(\mathcal{E})$.

In a given Clifford circuit, consider N possible *error locations*, portions of horizontal wires starting and ending on a gate or an input/output end of the wire. For example, the circuit in Fig. 2 has $N = 15$ error locations. A Pauli error may occur on any of these locations. A *circuit error* $\mathcal{E}$ is a set of N single-qubit Pauli operators without the phase, $P_i \in \{I, X, Y, Z\}$, $i \in \{1, \ldots, N\}$. When two circuit errors are applied sequentially, the result is a circuit error whose elements are pointwise products of Pauli operators with the phases dropped. The algebra defined by such a product is isomorphic to the N-qubit Pauli group without the phase. This is an Abelian group which also admits a representation in terms of length-$2N$ binary vectors $\mathbf{e} \equiv (\mathbf{u}|\mathbf{v}) \in \mathbb{F}_2^{2N}$. Multiplication of two circuit errors amounts to addition of the corresponding binary vectors $\mathbf{e}$.

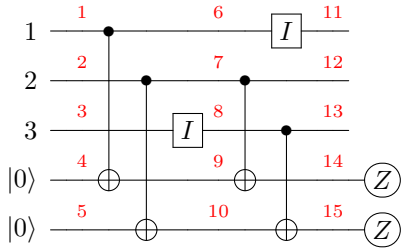

FIG. 2. Circuit measuring generators Z_1Z_2 and Z_2Z_3 of a three-qubit toy code. Digits indicate distinct circuit locations. Identity operators I are inserted so that the number of circuit locations along each qubit wire be odd.

With these definitions, error propagation through a Clifford circuit can be described as products of the original circuit error $\mathcal{E}$ with trivial circuit errors which have no effect on the outcome. The collection of all such errors forms *error equivalence group* (EEG) of the circuit. The generators of this group involved in error propagation are trivial errors, each formed as a union of a single-qubit Pauli P_i, $i \in \{1, \ldots, N\}$ (but not in the output for data qubits, or right before the measurements for ancillary qubits) with the result of error propagation of P_i across the subsequent gate. For example, for the circuit in Fig. 2, some of the EEG generators, with identity operators dropped, are $\{Z_1, Z_6\}$, $\{X_4, X_9\}$, $\{X_1, X_6, X_9\}$, and $\{Z_4, Z_6, Z_9\}$ (propagation across the leftmost CNOT), and $\{X_3, X_8\}$, $\{Z_3, Z_8\}$ (propagation across the leftmost identity gate labeled I).

In addition, for a circuit which includes qubits' initialization and measurement, the list of EEG generators includes errors Z_j on ancillary qubits right after initialization to $|0\rangle$ and right before the Z-basis measurement. For the circuit in Fig. 2, these are $\{Z_4\}$, $\{Z_5\}$ (ancillary qubits right after initialization), and $\{Z_{14}\}$, $\{Z_{15}\}$ (an-

cillary qubits just before measurement). It is important that stabilizer group of the input code, after its elements are promoted as circuit errors, forms a subgroup of thus constructed full circuit EEG[22]. Same applies to the stabilizer group of the output code.

In the binary representation, a single-qubit Hadamard gate connecting circuit positions i and i' corresponds to a pair of generators with non-zero elements $(u_i, v_i|u_{i'}, v_{i'}) = (10|01)$ and $(01|10)$ (X_i propagates into $Z_{i'}$ and v.v.), a phase gate similarly corresponds to $(10|10)$ and $(01|11)$ (Z_i propagates into $Z_{i'}$ and X_i into $Y_{i'}$), and a CNOT gate $(10\,00|10\,00)$ (input Z_i on the control and an output $Z_{i'}$ on the same wire), $(01\,00|01\,01)$ (input X_i on the control and $X_{i'}$, $X_{j'}$ on the outputs), $(00\,01|00\,01)$ (target X pass-through), and, finally, $(00\,10|10\,10)$. The generators for the single-qubit trivial errors are even simpler, since a Z_j maps to the weight-one vector $(u_j, v_j) = (10)$.

For a given circuit error $\mathcal{E}$ with binary form $\mathbf{e} \in \mathbb{F}_2^{2N}$, any equivalent error can be obtained by adding a linear combination of thus constructed circuit EEG generators $\mathbf{g}_j$. It is convenient to combine the corresponding generators into a generator matrix G with $2N$ columns. As discussed in the Appendix A, for a good EDC with the rank of the input-code stabilizer group $r_0 = n_0 - k$ and the constant $c = 1/2^\kappa$, see Eq. (10) and below, the generator matrix has the rank

$$\operatorname{rank} G = 2N - 2n_0 - f, \quad f \equiv n_a - \kappa - r_0 \geq 0. \quad (12)$$

Generally, a non-zero value $f > 0$ indicates a measurement redundancy.

With the circuit EEG generator matrix in place, it is easy to construct a formal expression for the probability of an error in a given equivalence class. Namely, if the circuit error probability distribution is given by an analog of Eq. (6), the probability of an error equivalent to a given $\mathbf{e} \in \mathbb{F}_2^{2N}$ is proportional to the Ising partition function,

$$\mathbb{P}(\mathbf{e}' \in \mathbb{F}_2^{2N} | \mathbf{e}' \simeq \mathbf{e}) = \text{Const}\, Z(G\theta, \{K_b(-1)^{[\mathbf{e}\theta]_b}\}), \quad (13)$$

cf. Eq. (9). It is also easy to find the conditional probability distribution of output errors with a given syndrome. Namely, given the binary form of an output data error $\mathbf{e}'_0 \in \mathbb{F}_2^{\otimes 2n_0}$ and a syndrome vector $\boldsymbol{\sigma}' \in \mathbb{F}_2^{\otimes n_a}$, we need to form the corresponding vector $\mathbf{e} \equiv \mathbf{e}(\mathbf{e}'_0, \boldsymbol{\sigma}') \in \mathbb{F}_0^{2N}$, filling only the components corresponding to the output data qubits and ancillary X_j just before the measurements which give non-zero syndrome bits in $\boldsymbol{\sigma}'$.

For ML decoding, we compare thus computed probabilities for all inequivalent errors consistent with the measured syndrome $\boldsymbol{\sigma}'$. In particular, these include errors that differ by a logical operator of the input (or, equivalently, the output) code, since non-trivial logical operators are outside the circuit EEG. In other words, length-$2N$ binary vectors corresponding to logical operators are linearly independent from the rows of the circuit EEG generator matrix G. It is convenient to define a binary logical generator matrix L of dimension $2k \times 2N$, whose rows correspond to mutually inequivalent logical operators of the input code.

For the ease of decoding, it is also convenient to introduce the *parity-check* matrix H, also with $2N$ columns, whose rows are orthogonal to the rows of both matrices G and L, and whose rank satisfies

$$\operatorname{rank} H = 2N - \operatorname{rank} G - \operatorname{rank} L = n_a - \kappa + r_0. \quad (14)$$

Clearly, H is dual to a matrix combing the rows of G and L. As in the case of a subsystem code, see Eq. (2), these matrices can be seen as forming a half of a CSS code, with stabilizer generator matrices $G_X = G$, $G_Z = H$, and X-logical operator generator matrix $L_X = L$.

The orthogonality requirement for rows of H can also be interpreted in terms of the N-qubit Pauli group associated with the circuit. Namely, the Pauli operators corresponding to the rows of the symplectic dual matrix $\widetilde{H} = H\Sigma$, see Eq. (1) and below, must commute with generators of the circuit EEG. This guarantees that each of these operators be a *spackle*, i.e., a circuit error where the single-qubit Pauli operators in any time layer can be obtained by error propagation from those in the previos time layer, see Ref. 22. Respectively, row weights of H scale linearly with the circuit depth.

D. Circuit subsystem code

The discussion in Ref. 22 focused on the special case of good EDCs where each qubit line is *required* to have an odd number of locations. In this special case, the circuit EEG can be seen as the gauge group of a quantum subsystem code of length N which encodes the same number of qubits k as the input/output codes, and has a distance not exceeding the corresponding distances, $d \leq \min(d_0, d'_0)$. Respectively, rows of the matrix $\widetilde{H}$ which correspond to generators of the stabilizer group of the subsystem code are necessarily given by linear combinations of the rows of G. In addition, circuit errors corresponding to the rows of the matrix L can be seen as dressed logical operators, while the bare logical operators which commute with the elements of the circuit EEG can be constructed as spackles.

In practice, any circuit can be easily modified to satisfy this additional requirement by inserting a null (identity) gate into each qubit line with an even number of locations, see Fig. 2 for an example. While it is not strictly necessary to work with circuits that satisfy this requirement, it is convenient, as the additional structure of the subsystem code can be used to verify the validity of the constructed matrices.

However, once the matrices G, H, and L are constructed, there is no need to refer to the subsystem code. In fact, the generator matrix row-reduction transformation described in the following Section IV preserves the orthogonality and the relation Eq. (14) between the ranks inherent in the CSS code map, but not the structure of the circuit subsystem code.

E. Circuit code distance.

How good can a measurement protocol be? What are the bounds on the distance d of the subsystem code associated with the circuit?

Generally, if d_0 and d_0' are the distances of the input and the output codes, the distance of the corresponding circuit code satisfies $d \le \min(d_0, d_0')$. This follows from the fact that a logical operator of the input code, e.g., is naturally mapped to a (dressed) logical operator of the circuit code. An important result in Ref. [22] is that one can always design a fault-tolerant circuit so that the distance d of the corresponding subsystem code be as good as that of the input code, $d = d_0$.

Unfortunately, circuit-code distance d does not have a direct relation to the probability distribution of the output errors; even single-qubit output errors may remain undetected. This is a well known "feature" of quantum error-correcting codes operating in a fault-tolerant regime, even for codes with single-shot properties[27–29]. Indeed, regardless of the circuit structure, errors on the data qubits in the locations just before the circuit output will not be detected.

In comparison, with a formally defined circuit code, such an error can be propagated back to the input layer and (when it is detectable, e.g., if its weight is smaller than the distance d_0' of the output code) it would necessarily anticommute with one or more combination(s) $Z_{\mathbf{g}}$ of the ancillary qubits. The original error would thus be detectable in the circuit code. Causality does not permit such an operation with actual circuit evolution. Formally, this functionality is removed due to assumed ancillary qubit initialization to $|0\rangle$.

Of course, even if a small-weight error goes undetected, it may get corrected after one or more additional measurement rounds. In practice, when an error-correcting code is analyzed in a fault-tolerant setting, the standard numerical procedure is to add a layer of perfect stabilizer measurements (no measurement errors). This guarantees that all small-weight errors at the end of the simulation be detected, and thus recovers the distance $d > 1$ of the circuit code, without the need to violate causality.

IV. MARGINAL DISTRIBUTIONS FOR CORRELATED ERRORS

The circuit EEG fully describes correlations between the circuit errors. However, it also contains a lot of excessive information: for the purpose of error correction, we are only interested in the distribution of the output errors and the syndrome, which are all supported at the rightmost locations of the circuit. In addition, the large size and sparsity of the circuit generator matrix G makes decoding difficult, except with the simplest circuits.

Present goal is to reduce the number of independent variables, by constructing the marginal distribution for a given subset of the variables. This amounts to a summation over the variables one is not interested in, e.g.,

$$\mathbb{P}(e_{s+1}, \ldots, e_m) = \sum_{e_1} \sum_{e_2} \cdots \sum_{e_s} \mathbb{P}(e_1, e_2, \ldots, e_m). \quad (15)$$

In the case of binary variables $e_i \in \{0, 1\}$, both the original and the resulting marginal distributions are multivariate Bernoulli distributions, and each can be described in terms of the Ising energy function (4).

A. Row-reduction transformation

1. Generator matrix and the coupling coefficients

Given an n-variate Bernoulli distribution described by the coupling matrix Θ, e.g., $\Theta = G\theta$ in Eq. (9), and a set of half-LLR coefficients K_b, $1 \le b \le m$, what are the corresponding parameters of the marginal distribution (15)? In the equivalent Ising-model representation (4), the goal is to describe the couplings between the remaining spins after a partial summation. Such a *star-polygon* transformation for a general Ising model has been constructed in Ref. 23. The transformation includes two special cases long known in the Ising model literature: the Onsager's star-triangle transformation[42] and the (inverse) decoration transformation[43, 44].

It is convenient to derive the result directly, focusing on the marginal distribution after a summation over just one spin variable $s_i \in \pm 1$ corresponding to ith row of Θ. Without limiting generality, assume that the chosen row has w non-zero elements in positions 1, 2, ..., w, decompose the corresponding bond variables $R_b = s_i T_b$, $T_b \in \pm 1$, $1 \le b \le w$, and perform the summation explicitly (with the additional one-half factor for convenience),

$$B_{\boldsymbol{\tau}} \equiv \frac{1}{2} \sum_{s_i = \pm 1} \exp\Big(s_i \sum_{b=1}^{w} K_b T_b\Big) = \cosh\Big(\sum_{b=1}^{w} K_b T_b\Big), \quad (16)$$

where $\boldsymbol{\tau} \in \mathbb{F}_2^w$ is a composite index with elements τ_b such that $T_b = (-1)^{\tau_b}$. To exponentiate this expression, rewrite $B_{\boldsymbol{\tau}}$ by analogy with Eq. (3),

$$B_{\boldsymbol{\tau}} = B_{00\ldots0}^{\frac{1+\tau_1}{2}\frac{1+\tau_2}{2}\cdots\frac{1+\tau_w}{2}} B_{00\ldots01}^{\frac{1+\tau_1}{2}\frac{1+\tau_2}{2}\cdots\frac{1+\tau_{w-1}}{2}\frac{1-\tau_w}{2}} \times \ldots B_{11\ldots1}^{\frac{1-\tau_1}{2}\frac{1-\tau_2}{2}\cdots\frac{1-\tau_w}{2}}, \quad (17)$$

where the coefficients $B_{\ldots}$ in the base of the exponents are the hyperbolic cosines (16) of the sum of coefficients $\pm K_b$ with the signs fixed, and matching exactly the signs in the exponents. As in Eq. (3), after a substitution of any given $\boldsymbol{\tau} \in \mathbb{F}_2^w$, only one term with the correct index $\boldsymbol{\tau}$ remains in the product. The modified bonds and the corresponding coefficients K_b' are obtained by expanding the polynomial in the exponent of Eq. (17). Because of the symmetry of the hyperbolic cosine, only even-weight products of the original bonds result from this expansion. Thus, in general, for an original row of weight w, the corresponding w columns are combined to

produce $w' = 2^{w-1}-1$ even-weight column combinations, a change of $\Delta w = 2^{w-1} - w - 1$ columns.

Specifically, for a row of weight $w = 1$, the transformation amounts to simply dropping the row and the corresponding column of Θ. The values of K_b remain the same, except for the one value that is dropped.

For a row of weight $w = 2$, only the sum of the corresponding columns is retained in Θ, with the coefficient

$$K'_{1,2} = \frac{1}{2}\ln\frac{\cosh(K_1+K_2)}{\cosh(K_1-K_2)} \equiv \frac{1}{2}\ln\frac{B_{00}}{B_{01}},$$

cf. Eq. (16). Equivalently, $\tanh K'_{1,2} = \tanh K_1 \tanh K_2$.

For a row of weigth $w = 3$, the three columns of the original matrix Θ are replaced by their pairwise sums, with the coefficient

$$K'_{1,2} = \frac{1}{4}\ln\frac{B_{000}B_{001}}{B_{010}B_{011}}$$

for the combination of the first two columns. The remaining coefficients $K'_{2,3}$ and $K'_{3,1}$ can be obtained with cyclic permutations of the indices.

For a row of weight $w = 4$, the four columns of the original matrix Θ are replaced with six pairwise sums and the seventh column combining all four original columns, with the coefficients, e.g.,

$$K'_{1,2} = \frac{1}{8}\ln\frac{B_{0000}B_{0001}B_{0010}B_{0011}}{B_{0100}B_{0101}B_{0110}B_{0111}},$$
$$K'_{1,2,3,4} = \frac{1}{8}\ln\frac{B_{0000}B_{0011}B_{0101}B_{0110}}{B_{0001}B_{0010}B_{0100}B_{0111}}.$$

In general, the coefficient K'_J in front of the product of T_b with indices b in an (even) subset $J \subseteq \{1, 2, \ldots, w\}$ is given by the sum of logarithms of the hyperbolic cosines $B_{\boldsymbol{\tau}}$ with $\tau_1 = 0$ (this accounts for symmetry of hyperbolic cosine), with the coefficients $\pm 1/2^{w-1}$, where the sign is determined by the parity of the weight of the subset $\boldsymbol{\tau}[J]$ restricted to J. It is easy to check that the numbers of positive and negative coefficients always match. Respectively, the coefficients for high-weight products are typically small in magnitude.

2. *Transformation for a parity check matrix*

The row-reduction transformation can also be written in terms of the *parity-check* matrix H, also with m columns, and *dual* to Θ, such that

$$\mathrm{H}\Theta^T = 0 \quad \text{and} \quad \operatorname{rank}\mathrm{H} = m - \operatorname{rank}\Theta. \tag{18}$$

To this end, consider the row-reduction of Θ as a combination of the following elementary column steps:

(i) The 1st column of Θ is added to columns 2, 3, …, w of Θ; as a result the i th row of Θ has a non-zero element only in the first position.

(ii) The 1st column of thus modified Θ is dropped, which leaves the i th row zero—it may be dropped as well;

(iii) If $w > 2$, $2^{w-1} - w$ combinations of two, three, …, $w-1$ columns with indices $b \le w-1$ are added to the matrix Θ. These can be sorted by weight so that each added column be a combination of exactly two existing columns in the modified Θ.

The corresponding steps for H are:

(i′) Columns $b \in \{2, 3, \ldots, w\}$ of H are added to its 1st column, which becomes identically zero as a result.

(ii′) Drop all-zero 1st column from thus modified H.

(iii′) For each column, e.g., b', added to Θ as a linear combination of two existing columns b_1 and b_2, H acquires a new row with the support on $\{b_1, b_2, b'\}$ to express this linear relation.

It is easy to check that row orthogonality, $\mathrm{H}\Theta^T = 0$, is preserved. Also, the rank of Θ is reduced by one, while the increase of the rank of H matches the number of columns added in step (iii′), so that the exact duality (18) is preserved.

B. Marginal distribution for error equivalence classes

This analysis is easily carried over to the problem of syndrome-based ML decoding for an $[[n, k]]$ subsystem code under a Pauli channel characterized by a $2n \times m$ matrix θ and a set of m half-LLR coefficients $\{K_b\}$, see Sec. III A. Given a gauge generator matrix G, the probability of an error equivalent to $\mathbf{e}$ is given by Eq. (9). Generally, for ML decoding we need to choose the largest of the 2^{2k} partition functions

$$Z(G\theta, \{K_b(-1)^{[\mathbf{e}'\theta]_b}\}, \quad \mathbf{e}' = \mathbf{e} + \boldsymbol{\alpha}L, \quad \boldsymbol{\alpha} \in \mathbb{F}_2^{2k}. \tag{19}$$

Typically, this needs to be done for a large number of error vectors $\mathbf{e}$. Can the calculation be simplified en masse by doing partial summation over the spins corresponding to *all* rows of $G\theta$ as described in the previous section?

The structure of the logical operators can be accounted for by extending the rows of the generator matrix which now has two row blocks,

$$\Theta = \begin{pmatrix} G\theta \\ L\theta \end{pmatrix}, \quad \tilde{K}_b = K_b(-1)^{[\mathbf{e}\theta]_b}, \tag{20}$$

and the half-LLR coefficients $\tilde{K}_b$, $b \in \{1, 2, \ldots, m\}$. A matching parity check matrix H is dual to Θ, see Eq. (18).

1. *Independent X and Z errors*

Let us first consider the simpler case of independent X and Z errors, where matrix $\theta = I_{2n}$. In this case $\mathrm{H} = H = \widetilde{H}\Sigma$, where $\widetilde{H}$ is a generator matrix of the code's stabilizer group, see Eq. (2). Marginal distribution being independent of the choice of the generator matrix, use

row transformations and column permutations to render

$$\Theta = \left(\frac{G}{L}\right) = \left(\begin{array}{c|c|c} I_r & A & B \\ \hline & I_{2k} & C \end{array}\right), \tag{21}$$

$$H = \left(B^T + C^T A^T \,\middle|\, C^T \,\middle|\, I_\ell \right), \tag{22}$$

where matrices B and C have $\ell \equiv 2n - r - 2k$ columns, and $r = \mathrm{rank}(G)$. The matrices Θ and H are mutually dual as can be immediately verified.

Row-reduction operations applied to each row in the upper block of Θ correspond to:

(i″) Column operations to set both blocks A and B to zero, and conjugate column operations on H to set its left-most column block to zero.

(ii″) Drop the upper row-block of the obtained Θ, as well as the left-most column blocks of the resulting Θ and H.

(iii″) Add an extra column block $M_1 + CM_2$ to the resulting Θ, where columns of M_1 and M_2 specify the linear combinations of the columns in its two remaining blocks, and a matching row-block to H.

As a result, the transformed matrices acquire the form

$$\Theta' = \left(I_{2k} \,\middle|\, C \,\middle\|\, M_1 + CM_2 \right), \tag{23}$$

$$H' = \left(\begin{array}{c|c|c\|c} C^T & I_\ell & 0 \\ \hline M_1^T & M_2^T & I \end{array}\right), \tag{24}$$

where double vertical lines are used to separate the newly added columns.

When the described transformation is applied to any of the original partition functions (19), the result is just an exponential $\exp\left(\sum_b K_b^{(\mathrm{fin})}\right)$ of the sum of the final half-LLR coefficients. Can identical columns of the final generator matrix (23) be similarly combined to simplify the structure of the final marginal distribution for error equivalence classes? The answer is yes, as long as we account for the effect of the error $\mathbf{e}$ on the values of the coefficients $K_b^{(\mathrm{fin})}$.

In fact, it is easy to check that the row-reduction transformations in Sec. IV A are such that the additional signs in Eq. (19) only affect the signs of the coefficients $K_b^{(\mathrm{fin})}$. Moreover, these signs correspond to the bits of the transformed error vector, cf. Eq. (23),

$$\mathbf{e}^{(\mathrm{fin})} = \left(\boldsymbol{\varepsilon}_1 \,\middle|\, \boldsymbol{\varepsilon}_2 \,\middle\|\, \boldsymbol{\varepsilon}_1 M_1 + \boldsymbol{\varepsilon}_2 M_2 \right), \tag{25}$$

where the vector $\boldsymbol{\varepsilon}_1$ selects the equivalence class, and $\boldsymbol{\varepsilon}_2 = \mathbf{s} + \boldsymbol{\varepsilon}_1 C$, with $\mathbf{s} \equiv \mathbf{e}H^T$ the original syndrome. Clearly, the right-most blocks in Eqs. (23) and (25) are obtained from $\boldsymbol{\varepsilon} \equiv (\boldsymbol{\varepsilon}_1|\boldsymbol{\varepsilon}_2)$ with $2k + \ell = 2n - r$ components as a right product with the combined matrix $M = \binom{M_1}{M_2}$. All $\ell_0 \equiv 2n - r$ components of $\boldsymbol{\varepsilon}$ being independent, there are

$$m'_{\max} = 2^{\ell_0} - 1 \tag{26}$$

non-trivial combinations. Combining identical columns in the transformation from $\boldsymbol{\varepsilon}$ to $\mathbf{e}^{(\mathrm{fin})}$, we can ensure that the final matrices contain no more than $m'_{\max}$ columns.

With Eq. (2), $r = n-k+\kappa$, so that $\ell_0 = n+k-\kappa$, where κ is the number of gauge qubits in the subsystem code. For a stabilizer code, $\kappa = 0$, thus $\ell_0 = n+k$. Clearly, the latter is just the number of logical generators, $2k$, plus the number of independent syndrome bits, $n-k$.

2. *A more general Pauli channel*

Instead of considering most general situation, consider an important case of a Pauli channel where any single-qubit error has a non-zero probability. Then, the incidence matrix can be chosen to have an identity-matrix block, $\theta = (I_{2n}|T)$, where T is an $(m-2n) \times 2n$ binary matrix. As a result, the matrix Θ and the half-LLR coefficients in Eq. (20) both acquire additional blocks of linearly-dependent components, while the parity-check matrix dual to Θ can be chosen in the form

$$\mathrm{H} = \left(\begin{array}{c|c} H & 0 \\ \hline T^T & I_{m-2n} \end{array}\right). \tag{27}$$

Since the relevant error in Eq. (20) is $\mathbf{e}\theta = (\mathbf{e}|\mathbf{e}T)$, the lower row-block of H gives a zero syndrome, just like the lower row-block in Eq. (24). Basically, after degeneracy is taken into account, the number of independent components of $\mathbf{e}\theta$ is $2n - r$, the same as for $\mathbf{e}$; final matrices Θ' and H' can always be constructed to have $m' \le m'_{\max}$ components given by Eq. (26). Of course, the actual resulting matrices, as well as the final half-LLR coefficients do depend on the assumed error model.

How much freedom is there to choose the matrices Θ' and H'? For the purpose of ML decoding, we need to go over the entire linear space $\mathcal{C}_{\Theta'}$ generated by the rows of Θ'; the choice of basis is irrelevant. The same is true regarding the parity check matrix H'. These are the same symmetries as for a generator and a parity-check matrices of a binary code.

In essence, the original quantum subsystem code with gauge G and logical L generator matrices has been transformed into a *classical* binary code, with the transformation dependent in a non-trivial fashion on the error probability distribution. This binary code has length $m' \le m'_{\max}$, and it encodes $2k$ bits.

Further, any non-trivial linear combination of rows of $L\theta$ with rows of $G\theta$ has weight lower-bounded by the distance of the quantum code, which gives the lower bound $d' \ge d$ on the distance of this binary code. In general, given the structure of the row-reduction transformation, the distance may be quite a bit larger, possibly scaling linearly with m'. Notice, however, that with highly non-uniform error probability distribution, more relevant parameter is not the distance, but the corresponding quantity weighted with half-LLR coefficients $K_b^{(\mathrm{fin})}$, related to the probability of the most-likely logical error. By

construction, this quantity is exactly the same as for the original quantum code.

Let us consider an important case of a sparse original parity check matrix H, e.g., with row and column weights bounded. This requires a low-density parity-check (LDPC) code with a sparse stabilizer generator matrix $\widetilde{H} = H\Sigma$, and an error channel with the generator matrix $\theta = (I_{2n}|T)$ also sparse. When acting on the parity check matrix, each row-reduction transformation drops a column of H, and may also add one or more rows of weight 3, see Sec. IV A 2. Thus, the parity-check matrix of the classical binary code describing the marginal probability distribution of error equivalence classes must also be sparse, with row weights not exceeding $w' \le \max(w, 3)$, where w is the maximum row weight of H in Eq. (27).

C. Marginal distribution for output errors in a good measurement circuit

This discussion also applies to the code associated with the error equivalence group of a good EDC. In this case the matrices G and H have $2N$ columns each, where N is the number of circuit locations, and their ranks are given by Eqs. (12) and (14). Just as any circuit error can be pushed all the way to the right, row-reduction can also be done starting with the bits at the beginning of the circuit and pushing toward its output. This way, a circuit error equivalence class can be characterized by

$$\ell_1 = 2N - \operatorname{rank} G = 2n_0 + f = (n_0 + k) + (n_a - \kappa) \quad (28)$$

bits, where $n_0 + k$ is the number of linearly-independent error equivalence classes in an $[[n_0, k]]$ stabilizer code and $n_a - \kappa$ is the number of syndrome bits measured in the circuit. Alternatively, as in a subsystem code with κ gauge qubits, $\ell_0 = n_0 + k - \kappa$ is the exponent in Eq. (26), and n_a is the number of additional error positions right before the measurements. As in the previous section, this gives an upper bound on the maximum number M' of columns in the matrices Θ' and H′ that may be necessary,

$$M' \le M'_{\max} = 2^{\ell_1} - 1. \quad (29)$$

After row-reduction for all generators of the circuit EEG, we get a classical $[M', 2k, d']$ binary code, where k is the number of encoded qubits, and $d' \ge d$.

Is this an LDPC code? This question has not been answered in the previous section: the parity check matrix H of the circuit code is not necessarily sparse as its row weights scale linearly with circuit depth. To analyze the sparsity of the output-error parity-check matrix H′, write it in a block form similar to Eq. (27),

$$\mathrm{H}' = \left(\begin{array}{c|c} \mathrm{H}'_0 & 0 \\ \hline H''_0 & H''_1 \end{array}\right), \quad (30)$$

where the upper row-block contains only the original columns of the circuit EEG parity check matrix H remaining after row-reduction steps (i′) and (ii′), while any row with exactly three non-zero entries added in a step (iii′) goes to the lower row-block. Further, if we assume circuit EEG H in the form (27), with the lower row-block of bounded weight (e.g., $w = 3$ for depolarizing errors), any potentially large-weight row in H'_0 must correspond to (i) a stabilizer generator of the output code, or (ii) a generator of the group $\mathcal{H}'$, see Ref. 22. All of these have bounded weights for any bounded-weight stabilizer LDPC code with a measurement circuit where stabilizer generators are measured independently and with a bounded number of ancillary qubits. On the other hand, these conditions do not generally hold for any family of concatenated codes based on a given code, or for subsystem codes where a stabilizer generator may have an unbounded weight or cannot be expressed as a product of gauge generators with a bounded number of terms. Nevertheless, even in these cases, given a potentially very large number of columns (29), it is reasonable to expect the final H′ to be a sparse matrix, with only a small fraction of non-zero elements.

V. DECODING STRATEGIES

A. Decoding based on circuit EEG

The present approach is to analyze error propagation in a Clifford measurement circuit in terms of its circuit EEG characterized by a logical generator matrix L and either a generator matrix G or a parity check matrix H. With a minor constraint on the circuit, these correspond to the circuit subsystem code constructed in Refs. 21 and 22. More generally, these matrices form (a half of) a CSS code, with generator matrices $G_X = G$ and $G_Z = H$, while rows of L define X-type logical operators. Simply put, any decoder capable of dealing with a CSS code with uncorrelated X-type errors can now be used to account for error correlations in a given measurement circuit.

Additional correlations can also be accounted for. Assuming a Pauli error channel (with or without correlations between different circuit locations) characterized by a coupling matrix θ and a set of half-LLR coefficients K_b (one per column), probability of a circuit error equivalent to a given one is given by the Ising partition function (13), with the spin-bond coupling matrix $G\theta$.

As already discussed, for ML decoding we need to find the largest of the Ising partition functions (19). Such a calculation can be expensive. Indeed, given a code encoding k qubits, we need to compute and choose the largest of all 2^{2k} partition functions corresponding to all non-trivial sectors; this can only be done in reasonable time for a code with k small. Previously, a computationally efficient ML decoder using this approach and 2D tensor network contraction for computing the partition functions has been constructed for surface codes[45] in the channel model (perfect syndrome measurement).

Feasible approaches for evaluating the partition functions (19) include tensor network contraction (see, e.g.,

in Ref. 46, for a 3D network contraction with complexity scaling as $\propto n\chi^9$, where χ is the bond dimension) and Monte-Carlo (MC) methods constructed specifically for efficient calculations of free energy differences, e.g., the non-equilibrium dynamics method [47] or the classical Bennett acceptance ratio[48]. In application to surface codes in FT regime, MC calculations are in essence simulations of bond-disordered 3D Ising model; such calculations can be done using GPU[49], FPGA[50], or TPU[51] hardware acceleration.

Notice also that the circuit code is extremely degenerate: with Hadamard, Phase, and CNOT gates the rows of the generator matrix G have (quaternary) weights not exceeding three. On the other hand, the row weights of the parity-check matrix H scale linearly with the circuit depth. By these reasons, iterative decoders like belief propagation (BP) are expected to fare even worse than with the usual (not so degenerate) quantum LDPC codes[52–54].

B. Generator-based decoding via marginal distributions

The calculation of the Ising partition functions needed for ML decoding can be simplified via partial summation over a subset of the spins. Technically, this corresponds to using a marginal distribution for the subset of variables needed, as discussed in Sec. IV.

Denote $\mathcal{V}$ the set of rows of the original generator matrix G [also, rows in the upper block of Θ in Eq. (20)]. Then, row-reduction in an increasing sequence of subsets

$$\mathcal{I}_0 \equiv \emptyset \subset \mathcal{I}_1 \subset \mathcal{I}_2 \subset \ldots \subset \mathcal{I}_s \equiv \mathcal{V} \tag{31}$$

defines a sequence of mutually-dual pairs of matrices $\{\Theta^{(j)}, \mathrm{H}^{(j)}\}$ with m_j columns, where

$$\Theta^{(j)} = \begin{pmatrix} G^{(j)} \\ L^{(j)} \end{pmatrix}, \quad j \in \{0, \ldots, s\}. \tag{32}$$

The ranks of logical generator matrices remain the same, $\operatorname{rank} L^{(j)} = 2k$, while the sequence $r_j \equiv \operatorname{rank} G^{(j)}$ is decreasing with increasing j, ending at $r_s = 0$. In essence, this defines a sequence of asymmetric (half) CSS codes $[[m_j, 2k]]$, with generator matrices $G_X^{(j)} = G^{(j)}$ and $G_Z^{(j)} = \mathrm{H}^{(j)}$, where rows of $L^{(j)}$ define X-type logical operators. The sequence ends with a CSS code with an empty X-generator matrix, i.e., a classical binary code with the parity check matrix $\mathrm{H}^{(s)}$.

For each of the codes in the sequence (32), there is also a set of half-LLR coefficients $\{K_b^{(j)}, 1 \le b \le m_j\}$. Given an error vector $\mathbf{e} \in \mathbb{F}_2^{m_j}$ matching the syndrome, ML decoding can be done by choosing the largest of the 2^{2k} Ising partition functions [cf. Eq. (19)]

$$Z\left(G^{(j)}, \{K_b^{(j)}(-1)^{e'_b}\}\right), \quad \mathbf{e}' = \mathbf{e} + \boldsymbol{\alpha} L^{(j)}, \quad \boldsymbol{\alpha} \in \mathbb{F}_2^{2k}. \tag{33}$$

By construction, the result should be the same for every $j \le s$. However, the complexities for computing the partition functions may differ dramatically.

One possibility for exact ML decoding is thus to choose a subset $\mathcal{I} \subset \mathcal{V}$ to optimize the partition function calculation. In particular, one option is to choose $\mathcal{I}_1$ such that all rows of weights one, two, and three are eliminated from the corresponding matrix $G^{(1)}$. This minimizes the number of columns in the exact generator matrix of the marginal-distribution. Even though all rows in the original circuit EEG generator matrix G have row weights not exceeding three, row-reduction on rows of weight three tends to create higher-weight rows. Thus, in general, $r_1 > 0$: the resulting partition function is not expected to be trivial.

To quote some numbers, Tables I and II give the dimensions of generator matrices and row-reduced generator matrices for two code families: the toy codes $[[n_0, 1, d_X = n_0/d_Z = 1]]$ with stabilizer generators $Z_i Z_{i+1 \bmod n_0}$, $0 \le i < n_0$, and the rotated toric codes (Ex. 11 in Ref. 55) $[[n_0, 1, 2t+1]]$ with $n_0 = t^2 + (t+1)^2$ and stabilizer generators $Z_i X_{i+t \bmod n_0} X_{i+t+1 \bmod n_0} Z_{i+2t+1 \bmod n_0}$, where $0 \le i < n_0$ and $t = 1, 2, \ldots$, with up to three measurements cycles. The simplest examples of the measurement circuits used for these two code families are shown in Figs. 3 and 4, respectively. In particular, while the original generator matrix for the circuit EEG of the $[[13, 1, 5]]$ code with measurements repeated $n_{\mathrm{cyc}} = 3$ times has dimensions 1066×1092, row-reduction of rows of weight up to three reduces the dimension to 182×247.

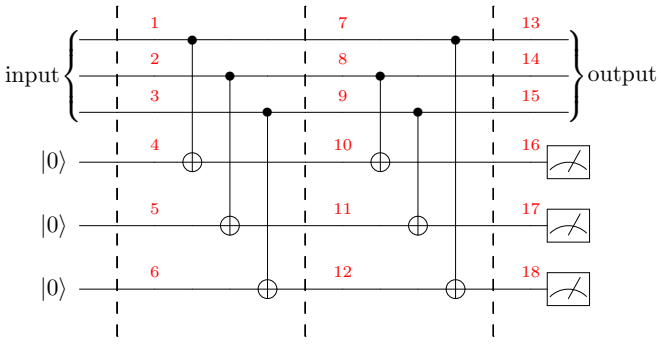

FIG. 3. Single-cycle circuit measuring the overcomplete set $\{Z_1Z_2, Z_2Z_3, Z_3Z_1\}$ of stabilizer generators for the toy code with $n_0 = 3$ data and $n_a = 3$ ancillary qubits.

Second possibility for exact ML decoding is to perform a summation over *all* spins, as described in Sec. IV C. This gives a probability distribution directly for the error equivalence classes; the logarithm of probability of an error equivalent to $\mathbf{e}$ is given, up to an additive constant, by a weighted sum of the half-LLR coefficients $-2\sum_b e_b K_b^{(\mathrm{fin})}$. Unfortunately, such an exact expression is expected to have an exponentially long list of coefficients, see Eqs. (28) and (29) for an upper bound. The corresponding column numbers are large even for the relatively simple circuits in Tables I and II. In fact, the

circuit parameters				generator matrix dimensions					remaining columns m_ϵ		
n_0	n_a	$n_{\rm cyc}$	d_0	orig	$w=2$	$w=3$	final	$w_{\rm fin}$	$\epsilon=10^{-2}$	$\epsilon=10^{-1}$	ℓ_1
3	3	1	3	30×36	3×10	0×10	0×10	0	10	10	7
5	5	1	5	50×60	5×16	0×16	0×16	0	16	16	11
7	7	1	7	70×84	7×22	0×22	0×22	0	22	22	15
3	6	2	3	72×78	9×19	3×19	2×43	21	38	21	10
5	10	2	5	120×130	15×31	5×31	3×79	21	69	34	16
7	14	2	7	168×182	21×43	7×43	4×115	21	99	48	22
3	9	3	3	102×108	15×28	6×27	4×75	35	69	31	13
5	15	3	5	170×180	25×46	10×45	7×116	20	104	50	21
7	21	3	7	238×252	35×64	14×63	10×157	20	140	69	29

TABLE I. Parameters of the original and row-reduced generator matrices for repetition code circuits as in Fig. 3 but with $n_{\rm cyc}$ measurement cycles, n_0 data and $n_a = n_{\rm cyc}n_0$ ancillary qubits. Also shown are dimensions of row-reduced generator matrices with rows of weights $w=2$ and $w=3$ (and smaller) eliminated; $w_{\rm fin}$ is the minimum row-weight of the final generator matrix with the smallest number of rows remaining. Remaining columns m_ϵ is the number of columns after columns with $|K_b|<\epsilon$ are dropped from the final generator matrix, assuming $p=0.05$ corresponding to a half-LLR value $K\approx 1.472$. The last column gives the value of ℓ_1 in the upper bound (29) on the number of columns.

circuit parameters				generator matrix dimensions						remaining columns m_ϵ			
n_0	n_a	$n_{\rm cyc}$	d_0	orig	$w=2$	$w=3$	$w=4$	final	$w_{\rm fin}$	$\epsilon=10^{-3}$	$\epsilon=10^{-2}$	$\epsilon=10^{-1}$	ℓ_1
5	5	1	3	130×140	25×35	20×35	16×43	13×75	10	65	47	23	11
5	10	2	3	280×290	55×65	45×65	37×81	32×139	10	112	82	46	16
5	15	3	3	410×420	85×95	70×95	58×119	51×203	10	159	118	70	21
13	13	1	5	338×364	65×91	52×91	40×115	32×163	9	163	120	61	27
13	26	2	5	728×754	143×169	117×169	93×217	83×291	9	273	213	118	40
13	39	3	5	1066×1092	221×247	182×247	146×319	134×419	9	384	305	182	53
25	25	1	7	650×700	125×175	100×175	76×223	58×331	9	331	234	116	51
25	50	2	7	1400×1450	275×325	225×325	177×421	157×555	9	528	413	223	76

TABLE II. Same as in Tab. I but for rotated toric codes $[[t^2+(t+1)^2, 1, 2t+1]]$ with $t=1,2,3$, represented as single-generator cyclic codes, see Example 11 in Ref. 55.

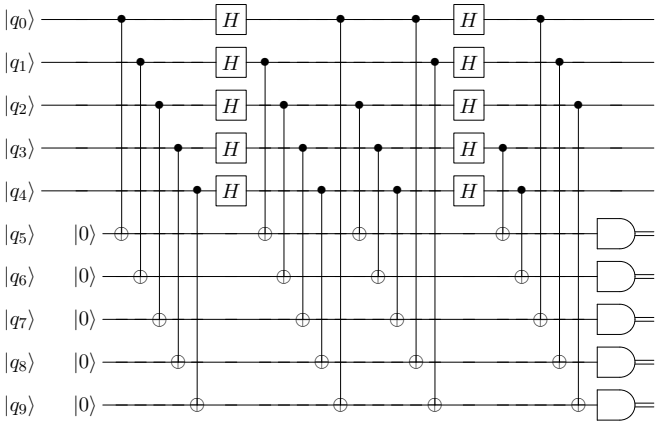

FIG. 4. Cingle-cycle measurement circuit for the five-qubit code with $n_a=5$ ancillary qubits.

Mathematica program (which was not written for efficiency) failed to complete full row-reduction except for the simplest repetition codes with $n_{\rm cyc}=1$ round of measurements.

In practice, the list of coefficients $K_b^{(\rm fin)}$ often contains a large number of entries with small magnitudes. This suggests a range of approximations, where only sufficiently large coefficients are preserved, e.g., $|K_b^{(\rm fin)}|>\epsilon$, for a given $\epsilon>0$. For incompletely reduced matrices in Tables I and II, with $\epsilon=0.1$, the number of columns is reduced, roughly, by a factor of two. On general grounds, the reduction factor is expected to be much larger for fully-reduced generator matrices.

Alternatively, only a fixed number χ of the largest in magnitude coefficients may be preserved. This latter approach is similar in spirit to approximate tensor network contraction using singular value decomposition and a fixed maximum bond dimension. Notice that if only the coefficients corresponding to columns of the matrix H_0' in Eq. (30) are preserved, we get an approximation similar to the conventional phenomenological noise model.

The "history code" decoding algorithm suggested by Chubb and Flammia[5], can be seen as a special case of generator-based decoding. Here the measurement circuit is assumed to have a block structure, and summation is

done over all circuit locations with the exception of those at the output of each block. With the circuits in Tables I and II, a block corresponds to a single measurement cycle. The full probability distribution is then recovered using Bayes rules, assuming no correlations between measurements in different blocks. Evidently, even in this case, the complexity of the probability distribution accounting for full error correlations could be prohibitive for exact ML decoding.

C. Parity-based decoding via marginal distributions

Summations over the spins in the subsets $\mathcal{I}_j$ with increasing j from a sequence like (31) give marginal distributions which account exactly for increasing numbers of alternative spin configurations. Respectively, the degeneracy of the corresponding row-reduced half CSS codes decreases with increasing j, down to a classical code with no degeneracy at the end of the sequence, $j = s$. A general expectation is that the accuracy of minimum-energy (ME) decoding would be increasing with increasing j. ME decoding becomes strictly equivalent to ML decoding for the end-of-the-sequence classical code.

Formally, given a parity-check matrix H and a set of LLR weights K_b, ME decoding aims to find an error vector $\mathbf{e}$ which gives the correst syndrome $\mathbf{e}H^T = \mathbf{s}$ and maximizes the error likelihood $\sum_b (-1)^{e_b} K_b$ or, equivalently, minimizes the error energy $\mathscr{E} = \sum_b e_b K_b$. Compared to generator-based decoding, an obvious advantage is that there is no need to go over all 2^{2k} logical operators. Unfortunately, for generic codes, even the relatively simple problem of ME decoding has an exponential complexity[56]. Given that the intermediate codes in the sequence (31) tend to be long, this makes it unpractical to use generic ME decoding algorithms with exponential complexity, e.g., the information subset[57, 58] or the classical Viterbi[59] algorithm.

Notice, however, that the sparsity of parity-check matrices $\mathrm{H}^{(j)}$ increase with j. The final matrix $\mathrm{H}^{(s)}$ is expected to be sparse whenever the output code is an LDPC code. Ideally, this classical code could be decoded with a linear complexity using a variant of belief propagation (BP) algorithm[60–63]. Assuming a sufficiently small fraction of failed convergence cases, the result would be equivalent to ML decoding of the correlated errors.

Unfortunately, this does not look so promising given the fact that this final code is expected to have an exponential length, see Eq. (29). Additionally, as confirmed with limited numerical simulations[64], having a large number of small in magnitude LLR coefficients tends to reduce the convergence rate. Using approximate decoding schemes with reduced number of LLR coefficients as discussed in Sec. V B is expected to help with both issues. Notice, however, that for such a reduction certain columns in the generator matrix Θ are merely dropped (puncturing). The corresponding transformation for the parity-check matrix H is *shortening*[65], which may reduce the sparsity of the resulting matrix and, in turn, negatively affect the convergence of BP decoding.

VI. DISCUSSION

Improving the accuracy of syndrome-based decoding in the presence of circuit-level error correlations would both increase the threshold to scalable quantum computation and improve the finite-size performance of quantum error correction. Present results make two steps in this direction. First, the observation that ML decoding under these conditions amounts to decoding the code[21, 22] associated with the circuit EEG, in the absence of correlations. Second, the structure of this latter code can be significantly simplified using row-reduction transformations, while leaving the probability of ML decoding unchanged. A variety of approximate decoding schemes naturally follow, which interpolate between the exact ML decoding and the decoding within a relatively simple phenomenological error model, with an additional handle on the degeneracy of the actual code to be used for decoding. Designing practical decoding algorithms, e.g., in application to surface codes with well-developed near-optimal syndrome measurement circuits, would require a substantial additional effort. However, this does not look like an unsolvable problem.

Decoding quantum LDPC codes is a major problem in the theory of QECCs, especially in a fault-tolerant regime with syndrome measurement errors present. While a substantial progress has been made in recent years[66–70], this problem remains open in application to generic highly-degenerate codes. Transformations discussed in Sec. IV change the degeneracy of a quantum code, and can even map from a quantum to a classical code. This opens new avenues to explore in decoding, in particular, new ways to apply existing iterative decoding algorithms to highly degenerate codes.

Circuit optimization: Traditional approach to quantum error correction is to start with a code, come up with an FT measurement circuit, compile it to a set of gates available on a specific quantum computer, and then finally design a decoder. Instead, one could start with the list of permitted two-qubit gates on a particular device and enumerate all good error-detecting circuits, increasing circuit depth and the number of gates. Given the circuit, it is easy to find the parameters of the input/output codes, as well as construct the associated circuit code. While at the end we would still need to evaluate and compare the performance of thus constructed codes, such a procedure could offer a shortcut to circuit optimization for specific hardware.

Acknowledgment: This work was supported in part by the NSF Division of Physics via grant No. 1820939.

Appendix A: Ranks of the matrices

1. Rank of the circuit-EEG generator matrix

Consider a good error-detecting circuit in Fig. 1 with the constant $c = 1/2^\kappa$ in Eq. (11) and the input (or output) code encoding $k = n_0 - r_0$ qubits, where r_0 is the rank of the input-code stabilizer group. Here κ is an integer[22]. Also assume that f measurements are redundant, so that the total number of ancillary qubits is $n_a = \kappa + r_0 + f$. Generators of the circuit EEG can be used to propagate any circuit error all the way to the output layer on the right, which requires $2N - 2(n_0 + n_a)$ independent generators, where N is the number of circuit locations. In addition, there are n_a ancillary Z_j generators on the input and n_a on the output layers. However, not all of them are independent. Indeed, when the n_a ancillary Z_j operators are propagated to the right, we get r_0 independent operators, each containing a product of ancillary Z_j and an element of the stabilizer group, κ independent operators containing ancillary X_j, and f additional combinations that are redundant except for a product of ancillary Z_j. Overall, this gives $\operatorname{rank} G = 2N - 2(n_0 + n_a) + n_a + \kappa + r_0$, which is the same as in Eq. (12).

2. Rank of the circuit stabilizer group

The circuit stabilizer group must commute with any EEG generator and also with any logical operator of the output code (say). Necessarily, its elements must be spackles[22]. Any spackle is uniquely determined by its support on the input layer, thus, there are total of $2(n_0 + n_a)$ independent spackles. We also have to ensure commutativity with ancillary Z_j generators on the left (drop n_a spackles) and on the right (drop κ spackles). Additional r_0 spackles have to be removed to ensure commutativity with the elements of the output code stabilizer generators, and $2k$ spackles to ensure commutativity with the output code logical operators. Overall, this leaves

$$\begin{aligned}\operatorname{rank} H &= 2(n_0 + n_a) - n_a - \kappa - r_0 - 2(n_0 - r_0) \\ &= n_a - \kappa + r_0,\end{aligned}$$

which is exactly the result in Eq. (14).

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

[2] C. G. Almudever, L. Lao, X. Fu, N. Khammassi, I. Ashraf, D. Iorga, S. Varsamopoulos, C. Eichler, A. Wallraff, L. Geck, A. Kruth, J. Knoch, H. Bluhm, and K. Bertels, "The engineering challenges in quantum computing," in *Design, Automation Test in Europe Conference Exhibition (DATE), 2017* (2017) pp. 836–845.

[3] P. Aliferis, D. Gottesman, and J. Preskill, "Quantum accuracy threshold for concatenated distance-3 codes," Quantum Inf. Comput. **6**, 97–165 (2006), quant-ph/0504218.

[4] David S. Wang, Austin G. Fowler, and Lloyd C. L. Hollenberg, "Surface code quantum computing with error rates over 1%," Phys. Rev. A **83**, 020302 (2011).

[5] Christopher T. Chubb and Steven T. Flammia, "Statistical mechanical models for quantum codes with correlated noise," (2018), unpublished, 1809.10704.

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

[7] Austin G. Fowler, Adam C. Whiteside, and Lloyd C. L. Hollenberg, "Towards practical classical processing for the surface code," Phys. Rev. Lett. **108**, 180501 (2012).

[8] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, "Surface codes: Towards practical large-scale quantum computation," Phys. Rev. A **86**, 032324 (2012).

[9] Austin G. Fowler, Adam C. Whiteside, Angus L. McInnes, and Alimohammad Rabbani, "Topological code autotune," Phys. Rev. X **2**, 041003 (2012).

[10] Christopher Chamberland, Guanyu Zhu, Theodore J. Yoder, Jared B. Hertzberg, and Andrew W. Cross, "Topological and subsystem codes on low-degree graphs with flag qubits," Phys. Rev. X **10**, 011022 (2020).

[11] Christopher Chamberland, Aleksander Kubica, Theodore J Yoder, and Guanyu Zhu, "Triangular color codes on trivalent graphs with flag qubits," New Journal of Physics **22**, 023019 (2020).

[12] Christophe Vuillot, Lingling Lao, Ben Criger, Carmen García Almudéver, Koen Bertels, and Barbara M. Terhal, "Code deformation and lattice surgery are gauge fixing," New Journal of Physics **21**, 033028 (2019).

[13] Giacomo Torlai and Roger G. Melko, "Neural decoder for topological codes," Phys. Rev. Lett. **119**, 030501 (2017).

[14] S. Krastanov and L. Jiang, "Deep neural network probabilistic decoder for stabilizer codes," Scientific Reports **7**, 11003 (2017), 1705.09334.

[15] N. P. Breuckmann and X. Ni, "Scalable neural network decoders for higher dimensional quantum codes," Quantum **2**, 68 (2018), 1710.09489.

[16] Zhih-Ahn Jia, Yuan-Hang Zhang, Yu-Chun Wu, Liang Kong, Guang-Can Guo, and Guo-Ping Guo, "Efficient machine-learning representations of a surface code with boundaries, defects, domain walls, and twists," Phys. Rev. A **99**, 012307 (2019).

[17] Paul Baireuther, Thomas E. O'Brien, Brian Tarasinski, and Carlo W. J. Beenakker, "Machine-learning-assisted correction of correlated qubit errors in a topological code," Quantum **2**, 48 (2018).

[18] Christopher Chamberland and Pooya Ronagh, "Deep neural decoders for near term fault-tolerant experiments," Quantum Science and Technology **3**, 044002 (2018).

[19] P. Baireuther, M. D. Caio, B. Criger, C. W. J. Beenakker, and T. E. O'Brien, "Neural network decoder for topological color codes with circuit level noise," New Journal of

Physics **21**, 013003 (2019).
[20] Nishad Maskara, Aleksander Kubica, and Tomas Jochym-O'Connor, "Advantages of versatile neural-network decoding for topological codes," Phys. Rev. A **99**, 052351 (2019).
[21] D. Bacon, S. T. Flammia, A. W. Harrow, and J. Shi, "Sparse quantum codes from quantum circuits," in *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing*, STOC '15 (ACM, New York, NY, USA, 2015) pp. 327–334, 1411.3334.
[22] D. Bacon, S. T. Flammia, A. W. Harrow, and J. Shi, "Sparse quantum codes from quantum circuits," IEEE Transactions on Information Theory **63**, 2464–2479 (2017).
[23] Jozef Strečka, "Generalized algebraic transformations and exactly solvable classical-quantum models," Physics Letters A **374**, 3718 – 3722 (2010).
[24] Christopher Chamberland and Michael E. Beverland, "Flag fault-tolerant error correction with arbitrary distance codes," Quantum **2**, 53 (2018), 1708.02246.
[25] C. Chamberland and A. W. Cross, "Fault-tolerant magic state preparation with flag qubits," Quantum **3**, 143 (2019), 1811.00566.
[26] Rui Chao and Ben W. Reichardt, "Quantum error correction with only two extra qubits," Phys. Rev. Lett. **121**, 050502 (2018).
[27] Héctor Bombín, "Single-shot fault-tolerant quantum error correction," Phys. Rev. X **5**, 031043 (2015).
[28] Benjamin J. Brown, Naomi H. Nickerson, and Dan E. Browne, "Fault-tolerant error correction with the gauge color code," Nature Communications **7**, 12302 (2016).
[29] Earl T. Campbell, "A theory of single-shot error correction for adversarial noise," Quantum Science and Technology **4**, 025006 (2019), 1805.09271.
[30] I. Dumer, A. A. Kovalev, and L. P. Pryadko, "Thresholds for correcting errors, erasures, and faulty syndrome measurements in degenerate quantum codes," Phys. Rev. Lett. **115**, 050502 (2015), 1412.6172.
[31] A. A. Kovalev, S. Prabhakar, I. Dumer, and L. P. Pryadko, "Numerical and analytical bounds on threshold error rates for hypergraph-product codes," Phys. Rev. A **97**, 062320 (2018), 1804.01950.
[32] David Poulin, "Stabilizer formalism for operator quantum error correction," Phys. Rev. Lett. **95**, 230504 (2005).
[33] Dave Bacon, "Operator quantum error-correcting subsystems for self-correcting quantum memories," Phys. Rev. A **73**, 012340 (2006).
[34] Daniel Gottesman, *Stabilizer Codes and Quantum Error Correction*, Ph.D. thesis, Caltech (1997).
[35] A. R. Calderbank, E. M. Rains, P. M. Shor, and N. J. A. Sloane, "Quantum error correction via codes over GF(4)," IEEE Trans. Info. Theory **44**, 1369–1387 (1998).
[36] Jeroen Dehaene and Bart De Moor, "Clifford group, stabilizer states, and linear and quadratic operations over GF(2)," Phys. Rev. A **68**, 042318 (2003).
[37] Scott Aaronson and Daniel Gottesman, "Improved simulation of stabilizer circuits," Phys. Rev. A **70**, 052328 (2004).
[38] Bin Dai, Shilin Ding, and Grace Wahba, "Multivariate Bernoulli distribution," Bernoulli **19**, 1465–1483 (2013).
[39] F. Wegner, "Duality in generalized Ising models and phase transitions without local order parameters," J. Math. Phys. **2259**, 12 (1971).
[40] A. J. Landahl, J. T. Anderson, and P. R. Rice, "Fault-tolerant quantum computing with color codes," (2011), presented at QIP 2012, December 12 to December 16, arXiv:1108.5738.
[41] A. A. Kovalev and L. P. Pryadko, "Spin glass reflection of the decoding transition for quantum error-correcting codes," Quantum Inf. & Comp. **15**, 0825 (2015), arXiv:1311.7688.
[42] Lars Onsager, "Crystal statistics. I. a two-dimensional model with an order-disorder transition," Phys. Rev. **65**, 117–149 (1944).
[43] Shigeo Naya, "On the spontaneous magnetizations of honeycomb and Kagomé Ising lattices," Progress of Theoretical Physics **11**, 53–62 (1954).
[44] Michael E. Fisher, "Transformations of Ising models," Phys. Rev. **113**, 969–981 (1959).
[45] Sergey Bravyi, Martin Suchara, and Alexander Vargo, "Efficient algorithms for maximum likelihood decoding in the surface code," Phys. Rev. A **90**, 032326 (2014).
[46] Markus Hauru, Clement Delcamp, and Sebastian Mizera, "Renormalization of tensor networks using graph-independent local truncations," Phys. Rev. B **97**, 045111 (2018).
[47] M. de Koning, Wei Cai, A. Antonelli, and S. Yip, "Efficient free-energy calculations by the simulation of nonequilibrium processes," Computing in Science Engineering **2**, 88–96 (2000).
[48] Charles H. Bennett, "Efficient estimation of free energy differences from Monte Carlo data," Journal of Computational Physics **22**, 245268 (1976).
[49] Tobias Preis, Peter Virnau, Wolfgang Paul, and Johannes J. Schneider, "{GPU} accelerated monte carlo simulation of the 2d and 3d ising model," Journal of Computational Physics **228**, 4468 – 4477 (2009).
[50] A. Gilman, A. Leist, and K. A. Hawick, "3D lattice Monte Carlo simulations on FPGAs," in *Proceedings of the International Conference on Computer Design (CDES)* (The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2013).
[51] Kun Yang, Yi-Fan Chen, Georgios Roumpos, Chris Colby, and John Anderson, "High performance Monte Carlo simulation of Ising model on TPU clusters," (2019), unpublished, 1903.11714.
[52] D. Poulin and Y. Chung, "On the iterative decoding of sparse quantum codes," Quant. Info. and Comp. **8**, 987 (2008), arXiv:0801.1241.
[53] Ye-Hua Liu and David Poulin, "Neural belief-propagation decoders for quantum error-correcting codes," Phys. Rev. Lett. **122**, 200501 (2019), 1811.07835.
[54] Alex Rigby, J. C. Olivier, and Peter Jarvis, "Modified belief propagation decoders for quantum low-density parity-check codes," Phys. Rev. A **100**, 012330 (2019), 1903.07404.
[55] A. A. Kovalev, I. Dumer, and L. P. Pryadko, "Design of additive quantum codes via the code-word-stabilized framework," Phys. Rev. A **84**, 062319 (2011).
[56] Pavithran Iyer and David Poulin, "Hardness of decoding quantum stabilizer codes," IEEE Transactions on Information Theory **61**, 5209–5223 (2015), arXiv:1310.3235.
[57] E. A. Kruk, "Decoding complexity bound for linear block codes," Probl. Peredachi Inf. **25**, 103–107 (1989), (In Russian).

[58] J. T. Coffey and R. M. Goodman, "The complexity of information set decoding," IEEE Trans. Info. Theory **36**, 1031 –1037 (1990).

[59] Andrew J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," IEEE Transactions on Information Theory **13**, 260–269 (1967).

[60] R. G. Gallager, *Low-Density Parity-Check Codes* (M.I.T. Press, Cambridge, Mass., 1963).

[61] M. P. C. Fossorier, "Iterative reliability-based decoding of low-density parity check codes," IEEE Journal on Selected Areas in Communications **19**, 908–917 (2001).

[62] Thomas J. Richardson and Rüdiger L. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," Information Theory, IEEE Transactions on **47**, 599–618 (2001).

[63] David Declerq, Marc Fossorier, and Ezio Biglieri, eds., *Channel Coding. Theory, Algorithms, and Applications* (Academic Press Library in Mobile and Wireless Communications, San Francisco, 2014).

[64] Weilei Zeng and Leonid P. Pryadko, "Iterative decoding of row-reduced quantum LDPC codes," (2020), unpublished.

[65] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes* (North-Holland, Amsterdam, 1981).

[66] Omar Fawzi, Antoine Grospellier, and Anthony Leverrier, "Efficient decoding of random errors for quantum expander codes," (2017), unpublished, 1711.08351.

[67] Omar Fawzi, Antoine Grospellier, and Anthony Leverrier, "Constant overhead quantum fault-tolerance with quantum expander codes," in *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018* (2018) pp. 743–754.

[68] A. Grospellier and A. Krishna, "Numerical study of hypergraph product codes," (2018), unpublished, 1810.03681.

[69] Pavel Panteleev and Gleb Kalachev, "Degenerate quantum LDPC codes with good finite length performance," (2019), unpublished, 1904.02703.

[70] Antoine Grospellier, Lucien Grouès, Anirudh Krishna, and Anthony Leverrier, "Combining hard and soft decoders for hypergraph product codes," (2020), unpublished, arXiv:2004.11199.