The XYZ^2 hexagonal stabilizer code

Basudha Srivastava[1], Anton Frisk Kockum[2], and Mats Granath[1]

[1]Department of Physics, University of Gothenburg, SE-41296 Gothenburg, Sweden

[2]Department of Microtechnology and Nanoscience, Chalmers University of Technology, SE-41296 Gothenburg, Sweden

We consider a topological stabilizer code on a honeycomb grid, the "XYZ^2" code. The code is inspired by the Kitaev honeycomb model and is a simple realization of a "matching code" discussed by Wootton [1], with a specific implementation of the boundary. It utilizes weight-six ($XYZXYZ$) plaquette stabilizers and weight-two (XX) link stabilizers on a planar hexagonal grid composed of $2d^2$ qubits for code distance d, with weight-three stabilizers at the boundary, stabilizing one logical qubit. We study the properties of the code using maximum-likelihood decoding, assuming perfect stabilizer measurements. For pure X, Y, or Z noise, we can solve for the logical failure rate analytically, giving a threshold of 50 %. In contrast to the rotated surface code and the XZZX code, which have code distance d^2 only for pure Y noise, here the code distance is $2d^2$ for both pure Z and pure Y noise. Thresholds for noise with finite Z bias are similar to the XZZX code, but with markedly lower sub-threshold logical failure rates. The code possesses distinctive syndrome properties with unidirectional pairs of plaquette defects along the three directions of the triangular lattice for isolated errors, which may be useful for efficient matching-based or other approximate decoding.

1 Introduction

In quantum computing, quantum bits, or qubits, are the basic unit of information. Whereas a classical computer acts on a single bit string, i.e., a list of 0's and 1's, a quantum computer can ad-

Basudha Srivastava: basudha.srivastava@physics.gu.se

Mats Granath: mats.granath@physics.gu.se

dress states that are superpositions of bit strings, or even non-classical entangled bit strings. While this enhanced representability gives quantum algorithms the potential for significant speed-up compared to classical algorithms [2, 3, 4, 5, 6, 7, 8, 9, 10], it also makes quantum computing much more difficult to implement in practice. Whereas unavoidable errors due to noise in a classical computer are discrete bit flips, errors in a quantum computer are continuous, corresponding to arbitrary rotations on a unit (Bloch) sphere.

One way to cope with errors is to implement error correction [11, 12, 13, 14, 15]. In quantum computing, this can be done using stabilizer codes. These codes are quantum algorithms that turn a collection of noisy qubits into a single logical qubit which is less error-prone [2]. The stabilizers of the code correspond to a set of commuting measurements on groups of qubits that project continuous errors to discrete errors on the code, which can be represented by Pauli X, Y, or Z operations on individual qubits. Topological codes, such as Kitaev's surface code [16, 17, 18, 19, 20], are particularly attractive for practical purposes since they use stabilizers consisting of geometrically local qubit operations on a two-dimensional grid. At the same time, logical errors are exponentially suppressed with increasing linear dimension of the code, provided the error rate is below the error threshold of the code [21, 22, 17]. Several stabilizer codes, including topological codes, have been experimentally implemented recently [23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35].

In this paper, we study a topological stabilizer code defined on a finite honeycomb grid. As shown in Fig. 1, the code has weight-six stabilizers on the hexagonal plaquettes of the form $XYZXYZ$, and weight-two XX stabilizers on "vertical" links. The code is the simplest example in the set of "matching codes" introduced by Wootton [1]. For concreteness, we will refer to
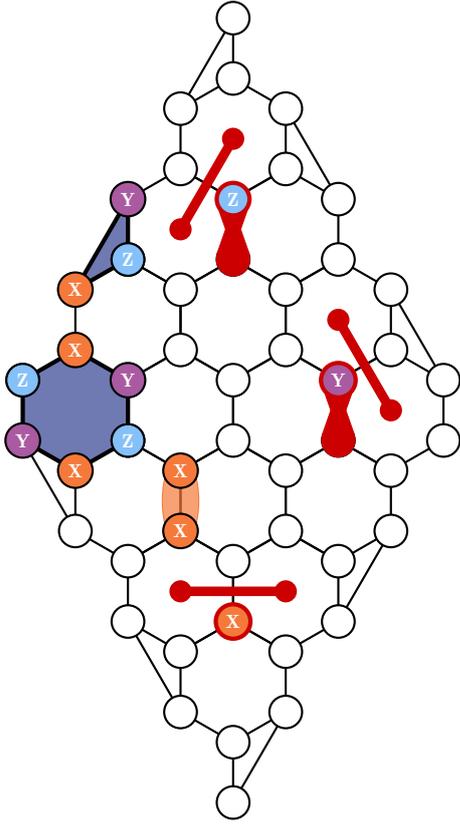
Figure 1: The XYZ^2 code for distance $d = 5$. The stabilizers of the code are weight-six $XYZXYZ$ on each hexagonal plaquette, weight-two XX on each vertical link, and weight-three XYZ on the boundary. Also shown are the unidirectional syndromes (red) for isolated X, Y, and Z errors.

it as the XYZ^2 code, based on the structure of the plaquette and link stabilizers. We introduce boundaries with half-plaquette stabilizers, which yields a $[[2d^2, 1, d]]$ code, i.e., it has code distance d and one logical qubit for $2d^2$ data qubits.

We study the basic properties of this code under the assumption of perfect stabilizer measurements and maximum-likelihood decoding. In particular, we compare the XYZ^2 code to the recently discussed XZZX code [36]. The latter is a modification of the surface code [37, 38], where the original pure X or pure Z weight-four stabilizers are replaced by mixed $XZZX$ stabilizers on each plaquette of the square lattice. Similarly, the XYZ^2 code can be derived through a suitable set of single-qubit transformations on the rotated surface code, followed by a doubling of each qubit stabilized by weight-two operators.

For depolarizing noise, up to the accuracy of our maximum-likelihood decoder, we find that the logical failure rate of these three codes depend

in a similar way on the number of data qubits and the code distance. However, analogously to the XZZX code, but in contrast to the rotated surface code, the XYZ^2 code has very favourable properties for biased noise, with a threshold that increases with bias, reaching 50 % for pure noise. In fact, for phase-biased noise, the XYZ^2 code has lower sub-threshold logical failure rates, for the same number of data qubits, even though the two codes have the same threshold. The reason for the suppressed failure rate is that for pure Z noise, the XYZ^2 code has code distance $2d^2$ while the XZZX code has code distance d.

Another interesting property of the XYZ^2 code is its distinct syndrome signatures (see Fig. 1), which follow from the mixed plaquette stabilizers and the triangular structure. For isolated X, Y, and Z errors, there are pairs of plaquette defects created, with unique orientations. We anticipate that this property may be used for approximate but efficient minimum-weight matching or clustering-type decoders.

Although these results suggest that there are potential benefits of the XYZ^2 code compared to surface-type codes on a square grid, a potential disadvantage is that it requires high qubit connectivity to measure the weight-six stabilizers, which could make it more susceptible to measurement noise. On the other hand, this downside may be compensated by the fact that it also has weight-two stabilizers that may be possible to measure without ancilla qubits [39, 40, 41, 42, 43]. For similar number of data qubits, the XYZ^2 code has approximately half as many high-weight (plaquette) stabilizers as the XZZX code.

This article is structured as follows. In Sec. 2, we first provide a more detailed account of the works and ideas leading up to the XYZ^2 code. Then we define the XYZ^2 code and some of its variations, and show how it can be constructed by a transformation of the rotated surface code. After explaining the methods used for error simulations and decoding in Sec. 3, we present results in Sec. 4, comparing the logical failure rates and thresholds of the XYZ^2 and XZZX codes for different types of biased noise. We conclude and give an outlook for future work in Sec. 5.

2 The XYZ^2 code

2.1 Background

The XYZ^2 code can be traced back to the Kitaev honeycomb model, which is an archetypical model for spin-liquid physics and topological order [44, 45]. The model has spin-half degrees of freedom on every site of a honeycomb lattice with Ising-type interactions on links, but with a spin quantization axis that depends on the orientation of the link. The product of link operators around each hexagonal plaquette of the lattice commutes with the Hamiltonian, thus presenting a set of conserved quantities that can serve as stabilizers for an error-correcting code.

The most direct way to formulate an error-correcting code based on the Kitaev honeycomb model is to use the links as gauge operators of a subsystem code [46, 47]. The centralizer of the gauge group, i.e., all group elements that commute with all elements in the group, make up the stabilizers of the code. Measuring the link operators thus preserves the code space, corresponding to the stabilized subspace of the Hilbert space. As the stabilizers are part of the gauge group, these measurement outcomes can be constructed from link measurements. However, in order for the subsystem code to have any logical qubits, there has to be a set of logical operators that commute with the gauge group but that are not part of the gauge group. The link measurements must not change the logical state, but the measurements must also not measure the logical operators, since this would project the logical code word. Unfortunately, the subsystem code based on the link gauge operators can be shown to not have any logical qubits, so it is not directly useful as a quantum memory [47, 48].

Nevertheless, as shown recently by Hastings and Haah [49, 50], it is possible to dynamically generate logical qubits by a careful implementation of the cycle of measurements of link operators. This type of code has been found in simulations to be competitive with the surface code, especially for systems where the two-qubit link measurements do not require an additional ancilla qubit, thus reducing the detrimental effects of the large number of measurements [43].

Another alternative, presented by Wootton [1, 51, 52], is to formulate a stabilizer code on the hexagonal lattice by, in addition to the plaque-

ttes, including as stabilizers strings of link operators between pairs of vertices, where each vertex is uniquely matched to a single other vertex. The simplest version of such "matching codes" is to match adjacent pairs of vertices along one direction, i.e., using one of the three link operators as a stabilizer. This is the XYZ^2 code, where our standard choice is to use the XX links as stabilizers (but YY or ZZ links can be used instead, as discussed below).

As shown in Refs. [1, 51], introducing defects, by measuring a different link operator, logical qubits can be introduced and elegantly braided to perform logical operations. However, as discussed in the introduction, we will in this article only consider the properties of the code as a quantum memory, introducing boundaries and boundary stabilizers to construct a set of $[[2d^2, 1, d]]$ codes. Furthermore, we only consider the ideal properties of these codes, assuming perfect stabilizer measurements. Even though this is not a realistic assumption for a real implementation of the code, using maximum-likelihood decoding and ideal measurements provides a baseline for the performance of the code. This baseline can be compared to that of other codes, independent of different schemes for modelling and decoding circuit-level noise.

2.2 Construction of the code

As shown in Fig. 1, the XYZ^2 code is constructed by placing $2d^2$ qubits on a hexagonal grid with a diamond-shaped boundary such that the setup contains $(d-1)^2$ plaquettes ($XYZXYZ$ stabilizers) and d^2 "vertical" links (XX stabilizers). As we will see, the integer d corresponds to the code distance, i.e., the minimum number of single-qubit operations required to perform a logical operation on a code word. We also introduce the notation d_σ, with $\sigma = X$, Y or Z, to denote the distance of the code for a chain of only X, Y, or Z operators. The qubits at the boundary are stabilized by an additional set of $2d - 2$ weight-three half-plaquette XYZ stabilizers, which are chosen such that no isolated single-qubit errors can go undetected. All together, these stabilizers make up the $2d^2 - 1$ constraints needed to encode a single logical qubit in $2d^2$ physical qubits. We note that the code can be truncated by removing qubits at the top and bottom of the grid, resulting in similar properties, but in this article, we

only consider the full code as presented here. We also only consider odd $d \geq 3$, as the properties for even d (as for the surface code) are different.

The logical operators of a stabilizer code are the set of operators that commute with the stabilizers but are not part of the stabilizer group. These operators can thus operate on the code space without the corresponding changes being detectable by the stabilizer measurements. Any set of three anticommuting operators X_L, Y_L, and Z_L, defined up to deformations with stabilizers, provide the logical Pauli group. In the XYZ^2 code, the shortest chain of single-qubit operators that commutes with all stabilizers is the horizontal chain of d Xs through the center of the grid [see Fig. 2(a)]. We designate this logical operator X_L and infer from its existence that the code distance is d. As this chain of Xs lies on d qubits which are part of XX vertical link stabilizers, the logical operator can be deformed by applying these d stabilizers, giving a total of 2^d possible distance-d pure X_L operators. In addition, there is a large number of longer X_L operators of mixed type, generated by acting with other stabilizers.

We define the logical Z operator Z_L as the vertical chain of ZY/YZ link operators on the $2d$ qubits through the center of the grid, as shown in Fig. 2(b). This chain can be deformed using adjacent stabilizers to form a mixed chain on the boundary of the code consisting of X, Y, and Z errors. We note, however, that there is no representation of Z_L which only has a single type of Pauli operator. Thus, for highly biased noise, this operator is unlikely to be generated through a random set of single-qubit errors.

The logical Y operator Y_L can be expressed as a product of any representation of Z_L and X_L. However, it turns out that we can deform this operator using plaquette and link stabilizers to create a pure-Y logical operator that acts on all $2d^2$ qubits [Fig. 2(c)]. To see this, we note that the vertical links are made up of XX stabilizers, which means that to construct a commuting operator made of only Ys, we need either a YY or an II (I = Identity) operator on each vertical link. However, due to the plaquette and half-plaquette stabilizers consisting of Xs, Ys, and Zs in a cyclic configuration, one needs a YY on each link (or no link, but then it is trivial) in order for the logical operator to commute with each stabilizer. This argument works in a similar manner

for a logical operator consisting of only Z errors. In fact, these two logical operators are equivalent and can be transformed into each other by acting with all link stabilizers. In conclusion, a logical operator consisting of only Z or Y errors would need a chain of length $2d^2$ qubits. Hence the distance of the XYZ^2 code under pure Z or Y noise is $d_Z = d_Y = 2d^2$.

While we have chosen the vertical links to be XX in this work, and hereafter assume this in all results shown unless stated otherwise, the code can be transformed into one using ZZ or YY link stabilizers by acting on the qubits with appropriate rotations. The logical operators will then transform accordingly as well. We show the different variations of the XYZ^2 code for distance 3 in Fig. 3.

Similar to the XZZX code, the mixed nature and the uniformity of the plaquette stabilizers in the XYZ^2 code give rise to the desirable property that X or Z errors on single qubits result in syndromes in single, distinct directions. For the XZZX code, which lies on a square grid, these directions are the two diagonal directions. However, in the XYZ^2 code, this property is extended to cover Y errors as well: X, Y, and Z errors give rise to syndromes along their unique individual directions, as shown in Fig. 1. For the XZZX code, this property was exploited by Ataides et al [36] to build an efficient matching-based decoder. It would be interesting to explore whether the tri-directional nature of the syndromes for the XYZ^2 code can be similarly exploited to implement an efficient decoder. However, in contrast to the XZZX code and the surface code, where syndrome bits (except at the boundary) always appear in pairs, this is not the case for the link stabilizers of the XYZ^2 code, which means they are not directly amenable to minimum-weight matching. This is also in contrast to the honeycomb code, for which only the static stabilizer space, corresponding to plaquette stabilizers, is decoded [49, 43].

2.3 Transformation of the rotated surface code to the XYZ^2 code

As shown in Refs. [44, 53, 54] the low-energy sector of the honeycomb lattice model in the limit of large ZZ coupling, replacing each such link by a single effective spin, is equivalent to the Wen model [55, 56, 57] with $YZZY$ interactions on
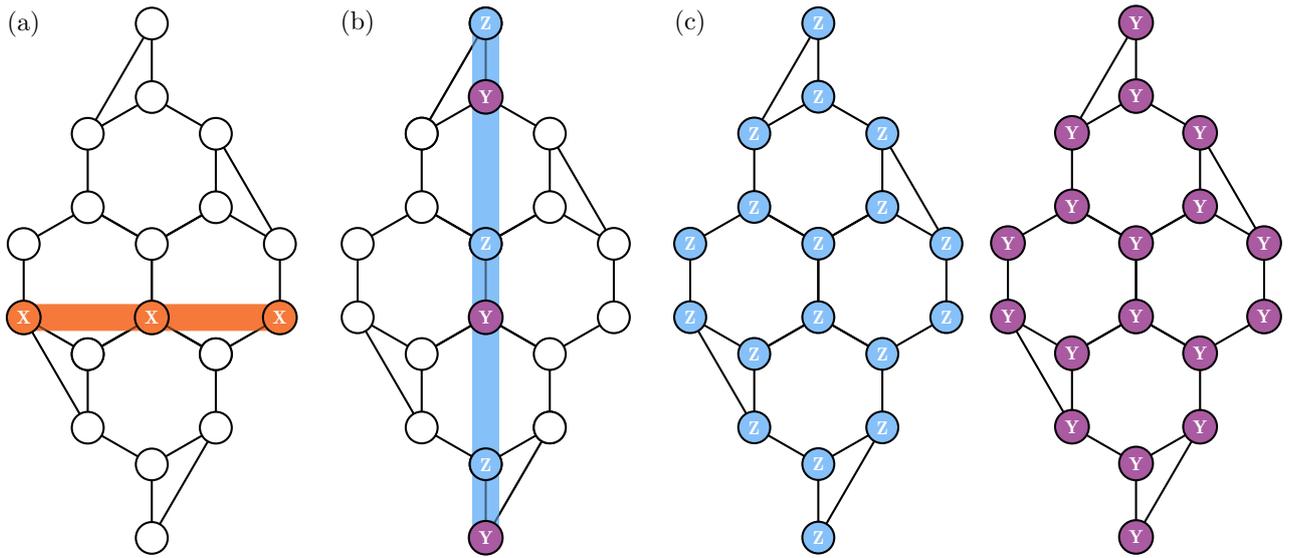
Figure 2: Logical operators on a distance-3 XYZ^2 code. (a) A pure logical X operator X_L formed by Xs horizontally through the centre of the grid. This weight-d operator is the shortest logical operator on the code and hence determines the code distance: $d_X = d$. (b) A logical Z operator Z_L that goes vertically through the center of the grid. As this operator consists of both Zs and Ys, it is not a pure logical operator. (c) Left: A chain of Z operators applied on every qubit in the code constitutes a pure logical Y operator Y_L. As this is the only pure operator consisting of Zs, it determines the code distance for pure Z noise to be $d_Z = 2d^2$. Right: we can transform this operator by applying all XX link stabilizers to obtain an operator consisting of Ys on all qubits. Thus we have $d_Z = d_Y = 2d^2$.
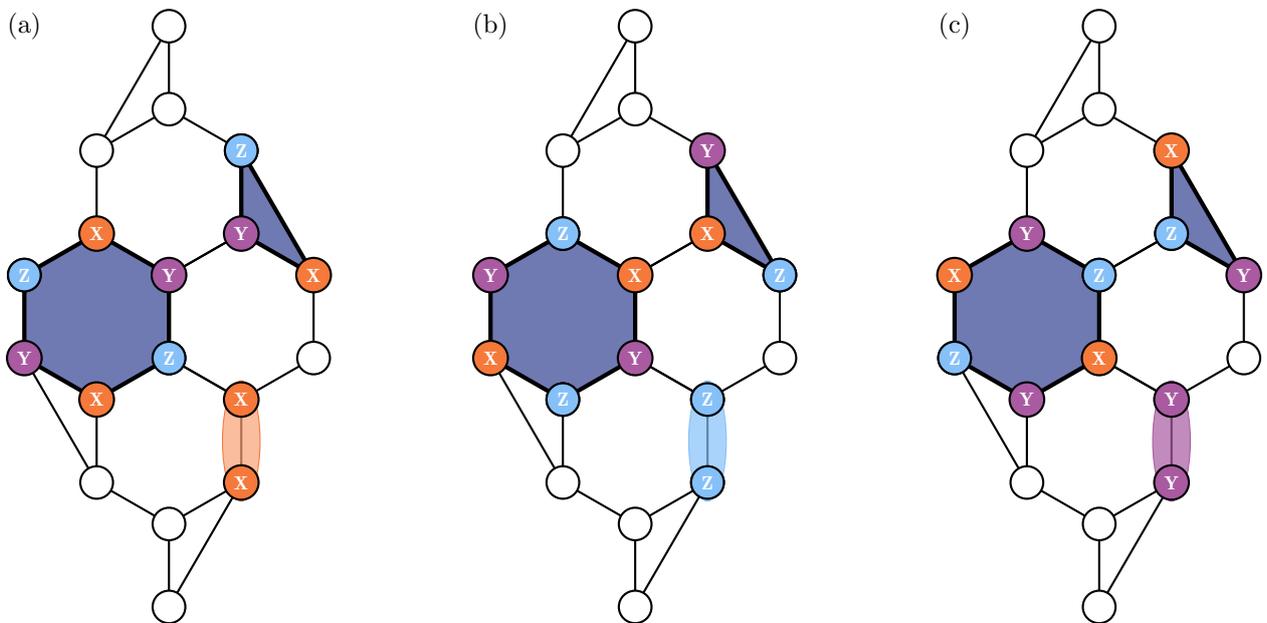
Figure 3: Variations of the XYZ^2 code for distance 3 with (a) XX, (b) ZZ, and (c) YY link stabilizers.

a square lattice. This model can then be transformed to the toric code by applying individual rotations on the spins. Here we perform similar transformations within the stabilizer formalism in reverse order, to show how the rotated surface code, the XZZX code, and the XYZ^2 code can be transformed into each other.

The rotated surface code is defined on a square lattice with alternating $XXXX$ and $ZZZZ$ plaquette stabilizers in the bulk and half-plaquette XX and ZZ stabilizers on the boundaries of the code. The shortest logical operators on this code are "horizontal" chains of Xs that go across each column and "vertical" Z chains that go across each row. In addition to these logical operators, there exist a pure-X and a pure-Z logical operator on each diagonal of the square grid. The only pure-Y logical operator on the rotated surface code is the operator which acts on each of the d^2 qubits with a Y. This effectively makes the distance of the code d under pure X or Z noise and d^2 under pure Y noise.

The XZZX code can be obtained from the rotated surface code by applying a Hadamard rotation on every other qubit. This operation modifies both the $XXXX$ and $ZZZZ$ plaquette stabilizers of the rotated surface code to identical $XZZX$ stabilizers on each plaquette of the square lattice. As a consequence of the rotation, the shortest logical operators of the rotated surface code are transformed from pure X and pure Z chains to mixed chains, containing both X and Z, going across horizontal and vertical dimensions of the grid. The two diagonal logical operators are exceptions, and still remain chains of pure X or pure Z errors. These last logical operators thus keep the code distance under pure X or Z noise d. The distance under pure Y noise is d^2, as for the rotated surface code.

To transform to the XYZ^2 code, we start by doubling the number of qubits, with each pair stabilized by XX. The code space stabilized by this operator is spanned by $|++\rangle$ and $|--\rangle$, where $|\pm\rangle = (|0\rangle \pm |1\rangle)/\sqrt{2}$. We make the choice of mapping the original single-qubit state $|0\rangle$ to $|++\rangle$ and $|1\rangle$ to $|--\rangle$. With this choice, the action of the Pauli operators on the single qubits is,
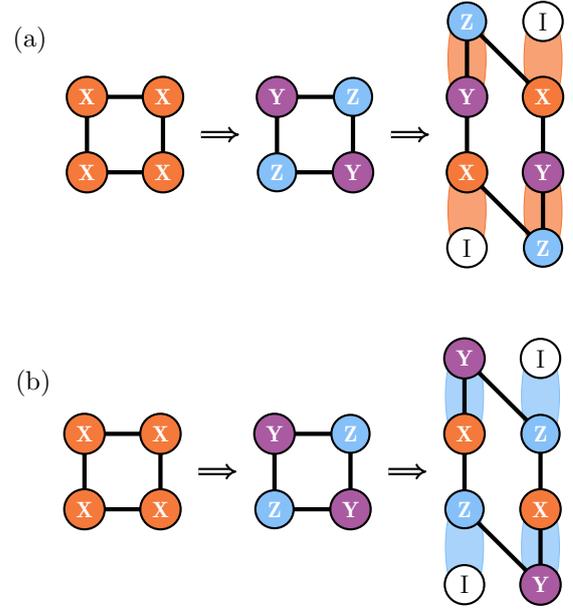
Figure 4: Transforming $XXXX$ (and $ZZZZ$) plaquette stabilizers of the rotated surface code to variations of the XYZ^2 code. (a) Transforming to the YZZY code by rotations and subsequently to the XYZ^2 code with vertical XX link stabilizers by replacing each data qubit in the $YZZY$ plaquette with two qubits stabilized by XX. (b) Transforming to the YZZY code by rotations and subsequently to the XYZ^2 code with vertical ZZ link stabilizers by replacing each data qubit in the $YZZY$ plaquette with two qubits stabilized by ZZ.

in the two-qubit basis, mapped to

$$X \to ZZ \text{ or } YY, \tag{1}$$
$$Y \to YZ \text{ or } ZY, \tag{2}$$
$$Z \to XI \text{ or } IX. \tag{3}$$

Now, starting from a slightly modified version of the XZZX code, with $YZZY$ plaquettes, this mapping results in the plaquette becoming a weight-six stabilizer ($XYZXYZ$), as shown in Fig. 4(a). Scaling this transformation up to the entire code gives us the XYZ^2 code [see Fig. 3(a)]. The shortest horizontal and vertical logical operators on the YZZY code now become mixed chains containing X, Y, and Z, extending from one side to another. The diagonal weight-d pure-Z logical operator becomes a pure X chain of length d, with 2^d degeneracy [Fig. 2(a)], whereas the orthogonal diagonal operator transforms to a YZ logical operator of length $2d$ [Fig. 2(b)]. The pure-X logical operator of the YZZY code acting on all qubits is mapped to pure Z or Y using Eq. (1) [Fig. 2(c)-(d)]. This ensures that

while the distance of the XYZ^2 code under pure X noise remains d, under pure Y or Z noise it becomes $2d^2$.

To obtain the variation of the XYZ^2 code shown in Fig. 3(b), where the vertical link stabilizers are ZZ instead of XX, a similar transformation can be applied, as shown in Fig. 4(b). In this case, plaquettes from the rotated surface code are transformed into $YZZY$ plaquettes and then further into $XYZXYZ$ hexagonal plaquette stabilizers and ZZ vertical link stabilizers by replacing each data qubit in the $YZZY$ plaquette by a pair of qubits stabilized by ZZ, such that $|0\rangle$ is represented by $|00\rangle$ and $|1\rangle$ is represented by $|11\rangle$.

3   Methods

In order to characterize quantum error-correcting codes, we need to mimic, to an extent, the actual noisy conditions of an experiment. To do so, we simulate the code under various different noise models and observe its performance. In this work, we assume perfect stabilizer measurements and ancilla readouts, and focus only on single-qubit errors that may occur on the data qubits.

The error model we use is the standard asymmetric depolarizing channel,

$$\rho \to (1-p)\rho + p_x X\rho X + p_y Y\rho Y + p_z Z\rho Z , \quad (4)$$

where $p = p_x + p_y + p_z$ is the total error probability per qubit per syndrome measurement cycle and ρ is the density matrix describing the state of the code. This error channel introduces an X error on a qubit with probability p_x, a Y error with probability p_y, and a Z error with probability p_z. If the probabilities of X, Y, and Z errors occurring is equal, we call the error channel a symmetric depolarizing channel, and the noise simply depolarizing noise.

For phase-biased noise, we define

$$\eta = \frac{p_z}{p_x + p_y} \qquad (p_x = p_y) \qquad (5)$$

following Refs. [58, 38, 36]. The limit $\eta \to \infty$ corresponds to pure Z noise and $\eta = 0.5$ to depolarizing noise. We can correspondingly define bit-flip-biased (X-biased) or Y-biased noise. We note that X-biased noise for our standard version of the XYZ^2 code with XX link stabilizers corresponds to Z-biased noise for the version with ZZ link stabilizers.

In practice, for simulating the code, we do not need to consider the full density matrix, but only generate sets of single-qubit errors, so-called error chains, by sampling from the error distribution. From a single such chain, one chain in each of four equivalence classes of chains is generated by acting with each of the logical operators and the identity operator. These four chains are then deformed by acting with random stabilizers to preserve only the syndrome of the initial chain. A maximum-likelihood decoder [17, 59, 60, 61, 62] (MLD) then calculates which of the four equivalence classes of chains is most likely to correspond to the syndrome. The event is counted as a successful error correction if the initial chain is in this most likely class, and a failed error correction otherwise. By sampling a large number of random initial chains, the logical failure rate of the code for a given error rate and noise bias is approximated.

Any practical MLD is approximate, since a full count of all errors consistent with a random syndrome grows exponentially with the number of stabilizers and thus is infeasible to evaluate exactly for anything but very low error rates or very small codes. In this work, we use the decoder recently presented in Ref. [62], the "effective weight and degeneracy" (EWD) decoder, which is based on Monte Carlo sampling of error configurations using the Metropolis algorithm. This decoder is essentially a more efficient, simplified version of the Markov-chain Monte Carlo decoder formulated in Refs. [61, 60], which we have adapted to the XYZ^2 code and biased noise. As discussed in Ref. [62], the EWD requires the tuning of a hyperparameter, corresponding to the error rate p_{sample} (which acts as an effective temperature) of the Metropolis sampling, to minimize the logical failure rate for a given physical error rate p. There is thus some systematic uncertainty in the results, in the sense that the results provide upper bounds on the failure rate. For the XZZX code, we use the matrix-product-state (MPS) decoder as implemented in Ref. [63]. As shown in [62], the two methods give almost identical results for the XZZX code for moderate code distances.

4   Results

Here we present results that compare the logical failure rates and the corresponding thresh-

olds of the XYZ^2 and XZZX codes under the assumption of perfect stabilizer measurements and maximum-likelihood decoding, for biased noise. When comparing the results for these codes, it is important to keep in mind that while the rotated surface code and the XZZX code utilize d^2 data qubits and $d^2 - 1$ ancilla qubits placed in a square grid for a code of distance d, the XYZ^2 code uses twice the amount of data qubits, $2d^2$, and the same amount of ancilla qubits, $d^2 - 1$, to achieve the same distance (although this requires a triangular lattice of qubits, where an ancilla qubits sits at the centre of each hexagonal plaquette in Fig. 1 and connects to all its nearest neighbours). Nevertheless, we will see that the XYZ^2 code matches the threshold values of the XZZX code, while having lower sub-threshold failure rates for phase-biased noise for similar numbers of data qubits.

We begin by deriving exact expressions for logical failure rates for pure X, Y, and Z noise, and compare those to the corresponding expressions for the XZZX code. This provides a good foundation for understanding the later observations using a finite bias. We also present results for depolarizing noise.

4.1 Analytical expressions for pure X, Y, and Z noise

We consider the version of the XYZ^2 code with XX link stabilizers, $N = 2d^2$ qubits, and the code distance d odd. Pure Z noise and pure Y noise are then equivalent, as discussed in Sec. 2.2. There is only one pure-Z operator that commutes with all the stabilizers: the weight-N operator over all qubits [see Fig. 2(c)]. Thus, for a syndrome corresponding to any pure-Z error chain, there is only one additional, complementary, chain that has the same syndrome. Maximum-likelihood decoding implies that for any syndrome, the most likely of the two will provide the correction chain. In other words, whenever an error chain with more than $N/2$ errors occurs, the error correction will fail. The marginal case, $N/2$ errors, fails with $50\,\%$ probability.

The logical failure rate, considering all possible error chains for pure Z or Y noise with error rate

$p < 0.5$, is thus

$$P_{f,Z}(p) = \sum_{n=N/2}^{N} \binom{N}{n} p^n (1-p)^{N-n}$$
$$- \frac{1}{2} \binom{N}{N/2} p^{N/2} (1-p)^{N/2}. \qquad (6)$$

The total probability of the chains for which the error correction will succeed is given by the complementary sum, which can be rewritten as $P_s(p) = P_f(1-p)$. Thus, given that $P_f(p) + P_s(p) = 1$, we find, independently of the size of the code, that $P_f(p = 0.5) = 0.5$. This is the maximal failure rate for pure noise. Since it is independent of code distance, $50\,\%$ is also the threshold error rate. For low error rates,

$$P_{f,Z} \sim p^{N/2} = p^{d^2}, \qquad (7)$$

reflecting the code distance $2d^2$ for pure Z or Y noise. Both the surface code and the XZZX code have these same properties for pure Y noise (for d odd), for which there is only a single logical operator, which consists of a Y on all $N = d^2$ qubits. The expression for the failure rate is the same as in Eq. (6), except that the last term is missing and the sum runs from $\lceil N/2 \rceil$ to N.

For pure X noise, the story is somewhat more complicated. As discussed in Sec. 2.2, there are 2^d weight-d pure-X logical operators along the central stabilizer row of the code [see Fig. 2(a)], with the degeneracy corresponding to acting with the XX link operators on this central row. To find out which chains will lead to logical failure, we first note that we can group chains by the parity of errors on each link: the probability of no error or two errors (even parity) on a link is p_e, and the probability of a single error (odd parity) is p_o, with

$$p_e = (1-p)^2 + p^2,$$
$$p_o = 2p(1-p). \qquad (8)$$

Acting with a logical operator changes the parity on each central link. Thus all of the chains which have the same parity of errors on each link are in the same equivalence class.

Given the syndrome, maximum-likelihood decoding corresponds to picking an arbitrary correction chain in the most likely equivalence class. For $p < 0.5$, $p_o < p_e$, which means that for a given syndrome, the set of chains that are in the

equivalence class with more than $d/2$ odd-parity errors on the central row will fail. Whatever the error configuration is outside the central row is irrelevant, since these will be equivalent for the two equivalence classes. The failure rate for pure X noise is thus given by

$$P_{f,X}(p) = \sum_{n=\lceil d/2 \rceil}^{d} \binom{d}{n} p_o^n p_e^{d-n}. \tag{9}$$

with $p_{o/e}$ given by Eq. (8). Just as for pure Z and Y noise above, $P_f(0.5) = 0.5$, meaning that $p = 0.5$ is the threshold also for pure X noise. For low error rates,

$$P_{f,X} \sim p^{\lceil d/2 \rceil}, \tag{10}$$

as expected for code distance d.

For the surface code, there is no simple analytical expression for either pure X or Z noise known. This lack of analytics is due to the fact that there is a large number of pure logical operators corresponding to the pure-X and -Z stabilizers. For the XZZX code, on the other hand, there is only a single pure logical operator of each type, of length d, such that the failure rate is given by Eq. (9), with $p_o = p$ and $p_e = 1 - p$, for both pure X noise and pure Z noise.

In Fig. 5, we plot the pure-noise logical failure rates for a few low-d code distances to visualize the differences between the XYZ^2 and XZZX codes. As shown above, both these codes have a 50% error threshold for pure noise, where the logical failure rate $P_f(p = 0.5) = 0.5$ is the maximal error rate given that there are only two relevant equivalence classes. For pure bit-flip noise, even though the code distance is $d_X = d$ for both codes, the XYZ^2 code is clearly inferior to the XZZX code, which is a consequence of the 2^d-fold degeneracy of the logical X_L operator for the former. For pure phase-flip noise, the roles are reversed: here the $d_Z = 2d^2$ code distance of the XYZ^2 code yields logical fail rates that are significantly lower than those of the $d_Z = d$ XZZX code. Finally, for pure Y noise, the two codes are equivalent (both to each other and to the rotated surface code), in the sense that the code distance is given by the total number of data qubits. Thus, e.g., the $d = 5$ XYZ^2 code and the $d = 7$ XZZX code overlap within the resolution of the plot. However, we note that, as discussed at the beginning of this section, for a given number of
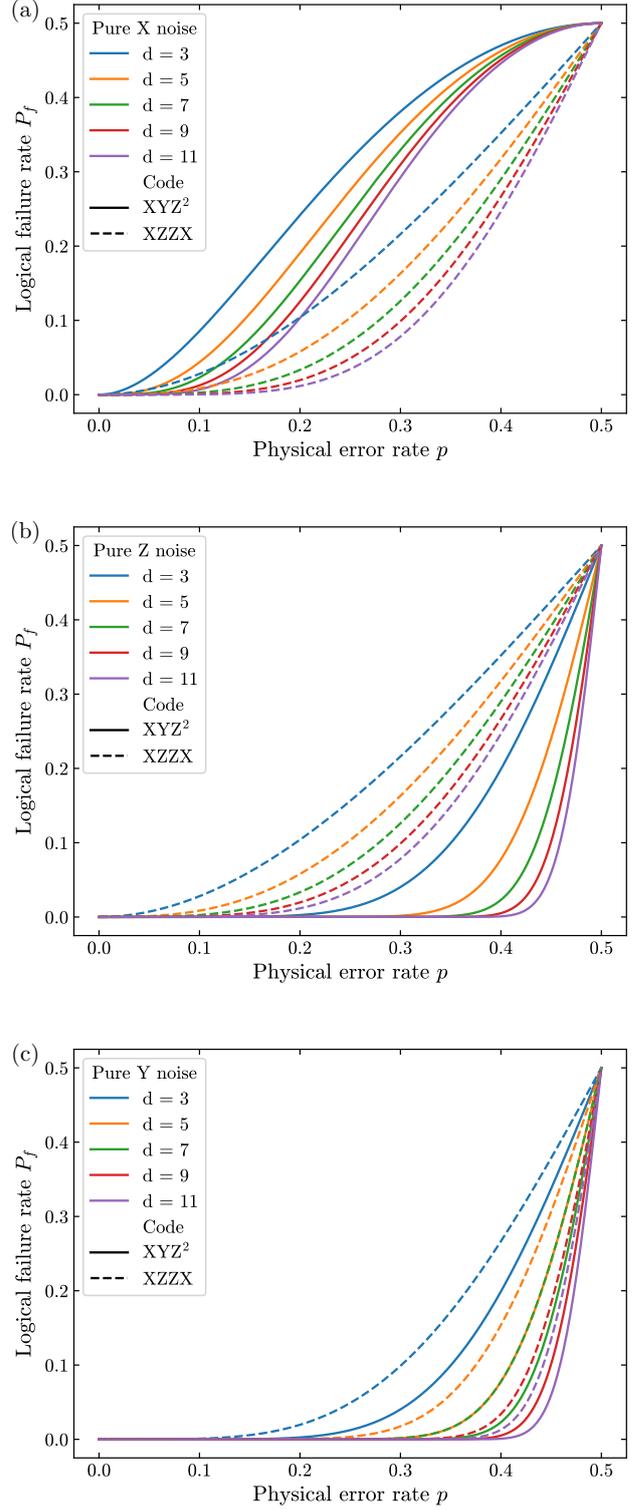
Figure 5: Comparison of the logical failure rate P_f for the XYZ^2 code (solid curves) and the XZZX code (dashed curves) as a function of the physical error rate p for different code distances d for (a) pure X, (b) pure Z, and (c) pure Y noise, plotted using analytical expressions, given in Eqs. (6) and (9) for the XYZ^2 code.
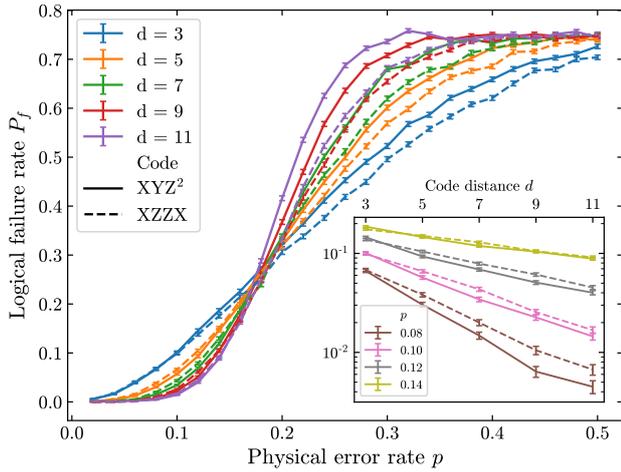
Figure 6: Logical failure rates as a function of physical error rates for different code distances d of the XYZ^2 (solid curves) and XZZX (dashed curves) codes for $\eta = 0.5$ (depolarizing noise). Each data point is evaluated using 10,000 syndromes for each code, code distance, and physical error rate, using the EWD decoder for the XYZ^2 code and the MPS decoder for the XZZX code. The inset shows the logical failure rate as a function of code distance for a few error rates below the threshold $p \approx 0.18$. The solid and dashed curves are a guide to the eye, connecting subsequent data points. Error bars indicate the statistical error (one standard deviation) based on the number of sampled syndromes and the mean logical failure rate. We attribute larger-than-error-bar variations of the EWD decoder for large physical error rates to difficulties with optimizing the decoder's sampling error rate [62].

data qubits, the XYZ^2 code requires fewer ancilla qubits for stabilizer measurements than the XZZX code.

4.2 Depolarizing noise

Before considering how the lessons from studying the model for pure noise translate to finite noise bias, we here compare the performance of the XYZ^2 and XZZX codes for depolarizing noise. Logical failure rates for depolarizing noise on the two codes are shown as a function of the physical error rate in Fig. 6 for code distances between $d = 3$ and $d = 11$. The results for the XYZ^2 code are obtained using the EWD decoder (see Sec. 3) with sampling error rate p_{sample} increasing with p and in the range 0.05 to 0.6, while the results for the XZZX code are obtained using the MPS decoder.

Up to the accuracy of our decoders, we observe the same threshold, around 18 %, for the

two codes. The logical failure rate is found to be a function of the number of data qubits above the threshold (for example, the lines for $d = 5$ for the XYZ^2 code is here close to the line for $d = 7$ for the XZZX, since this corresponds to almost the same number of data qubits) while for low error rates (inset of Fig. 6) the logical failure rate instead scales with the code distance.

4.3 Biased noise, $\eta = 10$

We now explore finite-bias noise with $\eta = 10$, which corresponds to one dominant error channel, ten times more likely than the other two error channels combined. The logical failure rates for the XYZ^2 and XZZX codes under the three possible such noise biases are shown in Fig. 7. Results for the XYZ^2 code use the EWD decoder (with $p_{\text{sample}} \in [0.14, 0.37]$ for X-biased noise and $p_{\text{sample}} \in [0.26, 0.41]$ for Z- and Y-biased noise).

For Z-biased noise, the case shown in Fig. 7(a), the threshold is about 28 % for both the XYZ^2 code and the XZZX code. However, we note that the logical failure rates below this threshold are highly suppressed for the XYZ^2 code compared to the XZZX code in this case. Comparing the results for an equivalent number of qubits instead of by code distance [inset in Fig. 7(a)], we observe still lower logical failure rates of the XYZ^2 code below the threshold.

In Fig. 7(b), which shows the results for bit-flip-biased noise, the roles of the two codes are reversed: they exhibit similar thresholds, but the logical failure rates are significantly higher for the XYZ^2 code, consistent with a naive extrapolation from the pure-noise limit.

For Y-biased noise [Fig. 7(c)], the results for the XYZ^2 code are the same as for the case of Z-biased noise in Fig. 7(a), while the logical failure rates for the XZZX code are suppressed below threshold compared to that case. As shown in the inset of Fig. 7(c), the failure rate scales with the number of data qubits, again consistent with an extrapolation from pure noise. Thus, in summary, the results for $\eta = 10$ show that the basic distinctions and similarities between the codes, based on the effective code distance for pure noise in Sec. 4.1, survive also for moderate finite bias.

Figure 7: Logical failure rates as a function of physical error rates comparing equivalent code distances of the XYZ^2 code (solid curves) and the XZZX code (dashed curves) for $\eta = 10$ for (a) Z-biased noise, (b) X-biased noise, and (c) Y-biased noise. The insets display the logical failure rates as a function of the number of data qubits for a few different sub-threshold error rates. For the XYZ^2 code, the same data is used in panels (a) and (c), since the code operates identically for Z- and Y-biased noise, and correspondingly for the XZZX code in panels (a) and (b). The number of syndromes used and the meaning of the error bars are the same as in Fig. 6.

5 Conclusion

We have studied a $[[2d^2, 1, d]]$ stabilizer code, the "XYZ^2" code, defined on a honeycomb lattice with specific boundary conditions. The code can be derived as a concatenation of the surface code and a two-qubit phase-error-detection code, with suitable single-qubit rotations, and is the simplest realization of a "matching code" proposed by Wootton [1]. The XYZ^2 code consists of weight-six $XYZXYZ$ stabilizers on hexagonal plaquettes, XX link stabilizers on the vertical links, and XYZ half-plaquette stabilizers on the boundary. Because of the hexagonal structure and the corresponding mixed-type stabilizers, the code has a remarkable syndrome signature, where isolated X, Y, and Z errors have pairs of defects with three different respective orientations.

We studied the logical fidelity of the XYZ^2 code assuming perfect stabilizer measurements and maximum-likelihood decoding, and compared the results to those of the rotated surface code and the XZZX code. We found that the XYZ^2 code has high thresholds for biased-noise error models that far surpass those of the rotated surface code and matches the thresholds demonstrated by the XZZX code. In contrast to those two codes, the XYZ^2 code also has a quadratic code distance, $2d^2$, for both pure Z and pure Y noise. This distinction survives also for phase-biased noise with finite bias, where the XYZ^2 code has significantly lower sub-threshold logical failure rates for the same number of data qubits. We also noted that the XYZ^2 code requires fewer ancilla qubits per data qubits than the XZZX and rotated surface codes to carry out the stabilizer measurements, although this assumes higher connectivity: a triangular lattice of qubits instead of a square grid.

In order for these apparent advantages of the XYZ^2 code compared to the XZZX code to be of practical value, it will be necessary to device an efficient approximate decoder, potentially exploiting the tri-directional syndrome properties of the XYZ^2 code, while at the same time dealing with the fact that link stabilizers do not give rise to paired syndrome bits. A possible way to address the latter, which we are currently exploring, is to treat the XYZ^2 code as a concatenated code of two-qubit blocks [64]. A minimum-weight matching scheme could incorporate edges weighted by conditional probabilities based on the link syndrome [65]. Something that also

needs to be explored is the detrimental effects of the weight-six stabilizer on circuit-level noise [52], while taking into account that this may be partially offset by the fact that approximately half of the stabilizers are instead weight-two. In short, a complete analysis of the XYZ^2 code should be undertaken that relaxes the assumptions of perfect stabilizer measurements used in the present work. Furthermore, since the XYZ^2 code is especially apt at addressing biased noise, it would also be interesting to explore how the code would perform compared to the XZZX code as a concatenated code with Kerr-cat continuous-variable qubits [66].

Acknowledgments

References

[1] James R Wootton. "A family of stabilizer codes for $D(\mathbb{Z}_2)$ anyons and Majorana modes". Journal of Physics A: Mathematical and Theoretical **48**, 215302 (2015).

[2] Michael A. Nielsen and Isaac L. Chuang. "Quantum Computation and Quantum Information". Cambridge University Press. Cambridge (2010).

[3] I.M. Georgescu, S. Ashhab, and Franco Nori. "Quantum simulation". Reviews of Modern Physics **86**, 153–185 (2014).

[4] Ashley Montanaro. "Quantum algorithms: an overview". npj Quantum Information **2**, 15023 (2016).

[5] G Wendin. "Quantum information processing with superconducting circuits: a review". Reports on Progress in Physics **80**, 106001 (2017).

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

[7] Sam McArdle, Suguru Endo, Alán Aspuru-Guzik, Simon C. Benjamin, and Xiao Yuan. "Quantum computational chemistry". Reviews of Modern Physics **92**, 015003 (2020).

[8] Bela Bauer, Sergey Bravyi, Mario Motta, and Garnet Kin-Lic Chan. "Quantum Algorithms for Quantum Chemistry and Quantum Materials Science". Chemical Reviews **120**, 12685–12717 (2020).

[9] Román Orús, Samuel Mugel, and Enrique Lizaso. "Quantum computing for finance: Overview and prospects". Reviews in Physics **4**, 100028 (2019).

[10] M. Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C. Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R. McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, and Patrick J. Coles. "Variational quantum algorithms". Nature Reviews Physics **3**, 625–644 (2021).

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

[12] A. M. Steane. "Error Correcting Codes in Quantum Theory". Physical Review Letters **77**, 793–797 (1996).

[13] Daniel Gottesman. "Stabilizer Codes and Quantum Error Correction" (1997). arXiv:quant-ph/9705052.

[14] Barbara M. Terhal. "Quantum error correction for quantum memories". Reviews of Modern Physics **87**, 307–346 (2015).

[15] Steven M. Girvin. "Introduction to quantum error correction and fault tolerance" (2021). arXiv:2111.08894.

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

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

[18] A.Yu. Kitaev. "Fault-tolerant quantum computation by anyons". Annals of Physics **303**, 2–30 (2003).

[19] Robert Raussendorf and Jim Harrington. "Fault-Tolerant Quantum Computation

with High Threshold in Two Dimensions". Physical Review Letters **98**, 190504 (2007).

[20] Austin G. Fowler, Matteo Mariantoni, John M. Martinis, and Andrew N. Cleland. "Surface codes: Towards practical large-scale quantum computation". Physical Review A **86**, 032324 (2012).

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

[22] Emanuel Knill, Raymond Laflamme, and Wojciech H. Zurek. "Resilient Quantum Computation". Science **279**, 342–345 (1998).

[23] J. Kelly, R. Barends, A. G. Fowler, A. Megrant, E. Jeffrey, T. C. White, D. Sank, J. Y. Mutus, B. Campbell, Yu Chen, Z. Chen, B. Chiaro, A. Dunsworth, I.-C. Hoi, C. Neill, P. J. J. O'Malley, C. Quintana, P. Roushan, A. Vainsencher, J. Wenner, A. N. Cleland, and John M. Martinis. "State preservation by repetitive error detection in a superconducting quantum circuit". Nature **519**, 66–69 (2015).

[24] Maika Takita, Andrew W. Cross, A. D. Córcoles, Jerry M. Chow, and Jay M. Gambetta. "Experimental Demonstration of Fault-Tolerant State Preparation with Superconducting Qubits". Physical Review Letters **119**, 180501 (2017).

[25] Christian Kraglund Andersen, Ants Remm, Stefania Lazar, Sebastian Krinner, Nathan Lacroix, Graham J. Norris, Mihai Gabureac, Christopher Eichler, and Andreas Wallraff. "Repeated quantum error detection in a surface code". Nature Physics **16**, 875–880 (2020).

[26] K. J. Satzinger et al. "Realizing topologically ordered states on a quantum processor". Science **374**, 1237 (2021).

[27] Laird Egan, Dripto M. Debroy, Crystal Noel, Andrew Risinger, Daiwei Zhu, Debopriyo Biswas, Michael Newman, Muyuan Li, Kenneth R. Brown, Marko Cetina, and Christopher Monroe. "Fault-tolerant control of an error-corrected qubit". Nature **598**, 281–286 (2021).

[28] Zijun Chen et al. "Exponential suppression of bit or phase errors with cyclic error correction". Nature **595**, 383–387 (2021).

[29] Alexander Erhard, Hendrik Poulsen Nautrup, Michael Meth, Lukas Postler, Roman Stricker, Martin Stadler, Vlad Negnevitsky, Martin Ringbauer, Philipp Schindler, Hans J. Briegel, Rainer Blatt, Nicolai Friis, and Thomas Monz. "Entangling logical qubits with lattice surgery". Nature **589**, 220–224 (2021).

[30] Ming Gong, Xiao Yuan, Shiyu Wang, Yulin Wu, Youwei Zhao, Chen Zha, Shaowei Li, Zhen Zhang, Qi Zhao, Yunchao Liu, Futian Liang, Jin Lin, Yu Xu, Hui Deng, Hao Rong, He Lu, Simon C Benjamin, Cheng-Zhi Peng, Xiongfeng Ma, Yu-Ao Chen, Xiaobo Zhu, and Jian-Wei Pan. "Experimental exploration of five-qubit quantum error correcting code with superconducting qubits". National Science ReviewPage nwab011 (2021).

[31] C. Ryan-Anderson, J. G. Bohnet, K. Lee, D. Gresh, A. Hankin, J. P. Gaebler, D. Francois, A. Chernoguzov, D. Lucchetti, N. C. Brown, T. M. Gatterman, S. K. Halit, K. Gilmore, J. A. Gerber, B. Neyenhuis, D. Hayes, and R. P. Stutz. "Realization of real-time fault-tolerant quantum error correction". Phys. Rev. X **11**, 041058 (2021).

[32] J. F. Marques, B. M. Varbanov, M. S. Moreira, H. Ali, N. Muthusubramanian, C. Zachariadis, F. Battistel, M. Beekman, N. Haider, W. Vlothuizen, A. Bruno, B. M. Terhal, and L. DiCarlo. "Logical-qubit operations in an error-detecting surface code". Nature Physics **18**, 80–86 (2021).

[33] Lukas Postler, Sascha Heußen, Ivan Pogorelov, Manuel Rispler, Thomas Feldker, Michael Meth, Christian D. Marciniak, Roman Stricker, Martin Ringbauer, Rainer Blatt, Philipp Schindler, Markus Müller, and Thomas Monz. "Demonstration of fault-tolerant universal quantum gate operations" (2021). arXiv:2111.12654.

[34] Sebastian Krinner, Nathan Lacroix, Ants Remm, Agustin Di Paolo, Elie Genois, Catherine Leroux, Christoph Hellings, Stefania Lazar, Francois Swiadek, Johannes Herrmann, Graham J. Norris, Christian Kraglund Andersen, Markus Müller, Alexandre Blais, Christopher Eichler, and Andreas Wallraff. "Realizing Repeated Quantum Error Correction in a Distance-Three Surface Code" (2021). arXiv:2112.03708.

[35] Dolev Bluvstein, Harry Levine, Giulia Semeghini, Tout T. Wang, Sepehr Ebadi, Marcin Kalinowski, Alexander Keesling, Nishad Maskara, Hannes Pichler, Markus Greiner, Vladan Vuletić, and Mikhail D. Lukin. "A quantum processor based on coherent transport of entangled atom arrays". Nature **604**, 451–456 (2022).

[36] J. Pablo Bonilla Ataides, David K. Tuckett, Stephen D. Bartlett, Steven T. Flammia, and Benjamin J. Brown. "The XZZX surface code". Nature Communications **12**, 2172 (2021).

[37] H. Bombin and M. A. Martin-Delgado. "Optimal resources for topological two-dimensional stabilizer codes: Comparative study". Physical Review A **76**, 012305 (2007).

[38] David K. Tuckett, Andrew S. Darmawan, Christopher T. Chubb, Sergey Bravyi, Stephen D. Bartlett, and Steven T. Flammia. "Tailoring Surface Codes for Highly Biased Noise". Physical Review X **9**, 041031 (2019).

[39] K. Lalumière, J. M. Gambetta, and A. Blais. "Tunable joint measurements in the dispersive regime of cavity QED". Physical Review A **81**, 040301 (2010).

[40] A. F. Kockum, L. Tornberg, and G. Johansson. "Undoing measurement-induced dephasing in circuit QED". Physical Review A **85**, 052318 (2012).

[41] D. Ristè, M. Dukalski, C. A. Watson, G. de Lange, M. J. Tiggelman, Y. M. Blanter, K. W. Lehnert, R. N. Schouten, and L. DiCarlo. "Deterministic entanglement of superconducting qubits by parity measurement and feedback". Nature **502**, 350 (2013).

[42] William P. Livingston, Machiel S. Blok, Emmanuel Flurin, Justin Dressel, Andrew N. Jordan, and Irfan Siddiqi. "Experimental demonstration of continuous quantum error correction" (2021). arXiv:2107.11398.

[43] Craig Gidney, Michael Newman, Austin Fowler, and Michael Broughton. "A fault-tolerant honeycomb memory". Quantum **5**, 605 (2021).

[44] Alexei Kitaev. "Anyons in an exactly solved model and beyond". Annals of Physics **321**, 2–111 (2006).

[45] Maria Hermanns, Itamar Kimchi, and Johannes Knolle. "Physics of the Kitaev model: Fractionalization, dynamic correlations, and material connections". Annual Review of Condensed Matter Physics **9**, 17–33 (2018).

[46] David Poulin. "Stabilizer Formalism for Operator Quantum Error Correction". Physical Review Letters **95**, 230504 (2005).

[47] Martin Suchara, Sergey Bravyi, and Barbara Terhal. "Constructions and noise threshold of topological subsystem codes". Journal of Physics A: Mathematical and Theoretical **44**, 155301 (2011).

[48] Yi-Chan Lee, Courtney G Brell, and Steven T Flammia. "Topological quantum error correction in the Kitaev honeycomb model". Journal of Statistical Mechanics: Theory and Experiment **2017**, 083106 (2017).

[49] Matthew B. Hastings and Jeongwan Haah. "Dynamically Generated Logical Qubits". Quantum **5**, 564 (2021).

[50] Jeongwan Haah and Matthew B. Hastings. "Boundaries for the Honeycomb Code" (2021). arXiv:2110.09545.

[51] James R Wootton. "Demonstrating non-Abelian braiding of surface code defects in a five qubit experiment". Quantum Science and Technology **2**, 015006 (2017).

[52] James R. Wootton. "Hexagonal matching codes with 2-body measurements" (2021). arXiv:2109.13308.

[53] Alexei Kitaev and Chris Laumann. "Topological phases and quantum computation" (2009). arXiv:0904.2771.

[54] Julien Vidal, Kai Phillip Schmidt, and Sébastien Dusuel. "Perturbative approach to an exactly solved problem: Kitaev honeycomb model". Physical Review B **78**, 245121 (2008).

[55] Xiao-Gang Wen. "Quantum Orders in an Exact Soluble Model". Physical Review Letters **90**, 016803 (2003).

[56] Alastair Kay. "Capabilities of a Perturbed Toric Code as a Quantum Memory". Physical Review Letters **107**, 270502 (2011).

[57] Benjamin J Brown, Wonmin Son, Christina V Kraus, Rosario Fazio, and Vlatko Vedral. "Generating topological order from a two-dimensional cluster state using a duality mapping". New Journal of Physics **13**, 065010 (2011).

[58] David K. Tuckett, Stephen D. Bartlett, and Steven T. Flammia. "Ultrahigh Error Threshold for Surface Codes with Biased Noise". Physical Review Letters **120**, 050505 (2018).

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

[60] James R. Wootton and Daniel Loss. "High Threshold Error Correction for the Surface Code". Physical Review Letters **109**, 160503 (2012).

[61] Adrian Hutter, James R. Wootton, and Daniel Loss. "Efficient Markov chain Monte Carlo algorithm for the surface code". Physical Review A **89**, 022326 (2014).

[62] Karl Hammar, Alexei Orekhov, Patrik Wallin Hybelius, Anna Katariina Wisakanto, Basudha Srivastava, Anton Frisk Kockum, and Mats Granath. "Error-rate-agnostic decoding of topological stabilizer codes" (2021). arXiv:2112.01977.

[63] David Kingsley Tuckett. "Tailoring surface codes: Improvements in quantum error correction with biased noise". PhD thesis. University of Sydney. (2020).

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

[65] Ben Criger and Imran Ashraf. "Multipath summation for decoding 2D topological codes". Quantum **2**, 102 (2018).

[66] Andrew S. Darmawan, Benjamin J. Brown, Arne L. Grimsmo, David K. Tuckett, and Shruti Puri. "Practical Quantum Error Correction with the XZZX Code and Kerr-Cat Qubits". PRX Quantum **2**, 030345 (2021).

[67] EWD-QEC. code: QEC-project-2020/EWD-QEC.

Accepted in ⟨ ⟩uantum 2022-04-18, click title to verify. Published under CC-BY 4.0.

15