# Applications of Locality and Asymmetry to Quantum Fault-Tolerance

by

Michael Newman

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Mathematics)
in The University of Michigan
2018

Doctoral Committee:

Professor Yaoyun Shi, Co-Chair
Professor Martin Strauss, Co-Chair
Adjunct Assistant Professor Carl Miller
Professor Karen Smith
Professor Kim Winick

Michael Newman
mgnewman@umich.edu
ORCID ID: 0000-0002-6640-1072

## ACKNOWLEDGEMENTS

It's always difficult to write acknowledgements. There are so many people that have supported me in so many different ways that it is impossible to give them their due credit. But let's try anyway.

To start with, my adviser Yaoyun Shi first infected me with an enthusiasm for quantum computing. He has taught me so much, both about quantum information, and about being an ambitious researcher. His patience, generosity, and wisdom have helped guide my graduate career, probably more than he realizes. My other adviser Martin Strauss provided me with guidance and advice throughout. He always made himself available when I needed and I am indebted to him for his support.

Audra McMillan shaped my experience in Ann Arbor these past years. When I complained, she listened; when I made puns, she laughed; when I was hurt, she took care of me; when I succeeded, she celebrated with me; and when I failed, she picked me up. It is no exaggeration to say that this thesis would not exist without her.

Cupjin Huang has been a wonderful collaborator and friend these past three years. His generosity in explaining so many concepts to me has been an immeasurable help. I'd also like to thank Carl Miller, who provided both guidance and friendship as a fellow mathematician who became interested in quantum.

I have made so many friends in Ann Arbor that it has become my home. The camaraderie I've had with my friends in the math department has been a mainstay of my years here. I have been so lucky to spend so much time with you all. My friends at volleyball helped me to relax and enjoy my free time. Thank you for putting up with all of the unreasonable spikes I have attempted; each point I've put into the net has been a testament to your patience. I hope that if any of you venture to read this, that you can remember one of the good times we had together. I surely will look back on them fondly.

Finally, I'd like to thank my family. My dad Tom, who has always supported me in everything I've done, gives me confidence. My mom Rosalie, who has always given me both her love and perspective, keeps me grounded. And my brother Jack, whose effervescence and humor give me so much joy. Frankly, they are all a bit strange, and I love them.

# TABLE OF CONTENTS

# LIST OF FIGURES

**Figure**

# LIST OF TABLES

**Table**

# LIST OF APPENDICES

**Appendix**

## ABSTRACT

Quantum computing sounds like something out of a science-fiction novel. If we can exert control over unimaginably small systems, then we can harness their quantum mechanical behavior as a computational resource. This resource allows for astounding computational feats, and a new perspective on information-theory as a whole.

But there's a caveat. The events we have to control are so fast and so small that they can hardly be said to have occurred at all[1]. For a long time after Feynman's proposal [2] and even still, there are some who believe that the barriers to controlling such events are *fundamental*. While we have yet to find anything insurmountable, the road is so pockmarked with challenges both experimental and theoretical that it is often difficult to see the road at all. Only a marriage of both engineering and theory in concert can hope to find the way forward.

Quantum error-correction, and more broadly quantum fault-tolerance, is an unfinished answer to this question. It concerns the scaling of these microscopic systems into macroscopic regimes which we can fully control, straddling practical and theoretical considerations in its design. We will explore and prove several results on the *theory* of quantum fault-tolerance, but which are guided by the ultimate goal of realizing a physical quantum computer.

In this thesis, we demonstrate applications of locality and asymmetry to quantum fault-tolerance. We introduce novel code families which we use to probe the behavior of thresholds in quantum subsystem codes. We also demonstrate codes in this family that are well-suited to efficiently correct asymmetric noise models, and determine their parameters. Next we show that quantum error-correcting encodings are incommensurate with transversal implementations of universal classical-reversible computation. Along the way, we resolve an open question concerning $\epsilon$-information-theoretically secure quantum fully homomorphic encryption, showing that it is impossible. Finally, we augment a framework for transversally mapping between stabilizer subspace codes, and discuss prospects for fault-tolerance.


[1] Or so say *the Watchmen*.

# CHAPTER 1

# Motivation and background

## 1.1 Fault-tolerant quantum computing

Quantum computing is a beautiful theory, admitting a host of possibilities that are either unresolved or unimaginable for classical computers to achieve [2, 3, 4, 5, 6, 7, 8]. However, the beauty of quantum computing lies not only in the power of its computational model, but in its *physical realizability*. Unfortunately, while the laws of quantum mechanics allow us to build a quantum computer in theory, its physical realization remains elusive. Simply put, fighting the effects of decoherence and imperfect controls on such small and fast scales must be an enormous triumph of both theory and engineering.

This thesis is concerned with quantum fault-tolerance, which involves encoding and protecting quantum information against pervasive interactions with the environment and imprecise devices. More abstractly, it is the study of taking the quantum effects of the very small and making them accessible to us.

In this work, we focus mostly on active quantum error-correction. While there are proposals for quantum fault-tolerance on a physical and passive level [9, 10], these are experimentally nascent and are not considered here. We prove results on the storage of quantum information, the reliable processing of quantum information, and their difficulties. Along the way, we resolve an open question in quantum cryptography that we require as a building block. Although these results are not directly related, they are small pieces of the same vast puzzle ultimately aimed at answering the question: *how can we encode and interact with quantum information in a controlled way*?

## 1.2 Overview of results

### 1.2.1 Intermediate 2-D compass codes

In Chapter 4, we study a novel class of codes which we call intermediate 2-D compass codes in the code capacity model. This family of codes includes many famous subfamilies, including rotated surface codes, Bacon-Shor subsystem codes, and Shor subspace codes.

Infamously, Bacon-Shor codes are an example of a family of local codes without *any* asymptotic error threshold, while surface codes boast the highest thresholds currently known. We define families that are intermediate between the two in order to probe threshold behavior. We do this by defining structured families, for which we determine threshold behavior both analytically and with a fixed decoder. Afterwards, we define randomized code families, and use a deep connection between quantum error-correction and statistical mechanics to give evidence that the threshold scales linearly with the ratio of the expected dimension of the stabilizer group and the total number of qubits.

Finally, we show that any code in this class supports an efficient minimum weight matching decoder. We then define toy families within this class that exhibit superior thresholds under asymmetric Pauli noise models using these decoders. We give further evidence of the threshold's linear scaling relation, and conclude with potential applications for such codes.

### 1.2.2 Information-theoretically secure quantum homomorphic encryption

In Chapter 5, we consider the problem of extending classical homomorphic encryption to the quantum setting. In doing so, we review existing proposals for quantum homomorphic encryption. We then focus on the restriction of quantum homomorphic encryption to the information-theoretically secure setting. Here, there are several existing proposals which offer intermediate security guarantees, which we review.

We consider the strongest security guarantee, *perfect security*, and a no-go proof due to [11] which prohibits efficient encoding sizes in any perfect security scheme. We then go on to answer an open question asked in [11] and [12], showing that even an $\epsilon$-relaxation of perfect security must incur significant overhead. This is proven via a reduction to communication lower bounds for quantum random access codes, and was obtained concurrently in [13]. Finally, we discuss prospects for working around this no-go theorem.

This section is notably *not* about quantum fault-tolerance, but rather concerns the capacity for quantum computers to offer cryptographic guarantees that are im-

possible classically. However, we will use this result as a stepping stone to prove results in Chapter 6.

#### 1.2.3 Restrictions on transversal gates

In Chapter 6, we study transversal gate sets for quantum error-detecting and error-correcting codes. We construct a homomorphic encryption scheme that provides information-theoretic security guarantees at inefficient but nontrivial encoding sizes. Using this, we study the question of which transversal gate sets can be implemented for quantum codes. We show that, for a large class of quantum error-correcting codes, implementing a classical-reversible universal transversal gate set is incommensurate with the lower bounds proven in Chapter 4. In particular, this shows that these codes cannot implement the valuable transversal Toffoli gate, answering a question implicit in both [14] and [15].

We then restrict our attention to stabilizer subspace codes, and use the special structure of codes exceptional to our theorem to rule out a transversal Toffoli gate in this setting. We also provide an alternative proof inspired by the Bravyi-Konig hierarchy [16], and note that similar arguments were extended in [17] to show that all transversal gates in stabilizer subspace codes must lie in the Clifford hierarchy. Finally, we discuss several potential workarounds to our no-go theorem.

#### 1.2.4 A framework for transversal code switching

In Chapter 7, we investigate a framework for deforming stabilizer codes recently proposed in [1] known as the stabilizer rewiring algorithm (SRA). This framework allows one to map between different stabilizer codes via a transversal circuit comprised solely of Pauli gates and measurements. As gates along this circuit are applied, the initial code is deformed through a series of intermediate codes before reaching the final code.

We propose a randomized variant to the SRA, the rSRA. We show that there always exists a path of deformations which preserves the code distance throughout the circuit, while using at most linear overhead in the code distance. This answers an open question in [1], although is insufficient for full fault-tolerance. Furthermore, we show that a random path will almost always suffice, and discuss both prospects and barriers for implementing general fault-tolerant code switching circuits.

### 1.3 Dissertation outline

This dissertation is divided into eight chapters. Chapter 2 introduces the basics of quantum information. This is meant to be a short summary of the foundations

required to understand this thesis. For a more complete understanding of quantum information, we recommend [18]. Next, Chapter 3 introduces a somewhat broader view of quantum error-correction relevant to this thesis, but again, we recommend [19] for a complete view of the field. All of the content of Chapters 2 and 3 is background except for subsection 3.5.2, in which we briefly introduce a family of homological codes as an example construction.

Chapter 4 introduces the family of 2-D compass codes and their properties. Chapter 5 is focused on the limitations of information-theoretically secure quantum homomorphic encryption. Chapter 6 discusses limitations on transversal gates. Chapter 7 is concerned with transversal circuits mapping between stabilizer codes. Each of these chapters is prefaced by an introduction summarizing the results in the context of other work.

Finally, in Chapter 8 we summarize these results. We then give a broad overview of potential avenues for future work, as well as rehashing the most salient open questions from each chapter.

### 1.3.1 Works appearing

The work in Chapter 4 is ongoing at the time of this thesis, and will be found in [20]. It has been presented in part during a contributed talk at the 3rd Aspen Winter Conference on Advances in Quantum Algorithms and Computation by a co-author.

The main result in Chapter 5 first appears in [21], but the review is drawn primarily from [22]. The result was shown concurrently in [13].

The work in Chapter 6 is contained almost entirely in [21]. This work was accepted at the 7th International Conference on Quantum Cryptography as a contributed talk by the author. It has been submitted for publication.

Finally, the work in Chapter 7 is drawn from [23]. It was presented as a poster at the 21st Annual Conference on Quantum Information Processing by the author, and has been submitted for publication.

In all of the above works, the author of this thesis has appeared as either the first author or co-first-author, but has benefited tremendously from discussions with his co-authors, colleagues, and friends. Other works to which the author has contributed as a graduate student, but which do not fit into the theme of this thesis, can be found in [24, 25, 26].

# CHAPTER 2

# The basics of quantum information

We begin with a cursory review of quantum computation. The following is not intended to be complete, but is intended to briefly introduce only those fundamental elements of quantum information that will be required in this thesis.

## 2.1 Quantum states

We start from the very beginning. The atomic units of quantum information are called *qubits*. Formally, the state of a qubit can be called pure or mixed. A *single qubit pure state* $|\psi\rangle$ is simply an equivalence class of unit vectors in $\mathbb{C}^2$,

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,/(|\psi\rangle \sim c\,|\psi\rangle : |c| = 1).$$

Here, the $|\cdot\rangle$ notation represents vectors and $\langle\cdot|$ notation represents dual vectors, so that $\langle\psi||\phi\rangle$ (which we abbreviate as $\langle\psi|\phi\rangle$) is a scalar inner product whereas $|\psi\rangle\langle\phi|$ is a rank one operator.

We can extended this definition to $n$-qubit pure states by taking the tensor product of the individual states, so that $|\psi\rangle$ is an *$n$-qubit pure state* if it can be expressed as

$$|\psi\rangle = \sum_{\vec{i}\in\{0,1\}^n} \alpha_{\vec{i}}|\vec{i}\rangle,$$

a unit vector in the state space $(\mathbb{C}^2)^{\otimes n}$. Throughout, we will often suppress the tensor product notation as concatenation, i.e. $|i_1\rangle \otimes |i_2\rangle = |i_1 i_2\rangle$.

A (general) single qubit quantum state can be expressed as a density operator. A *density operator* is a positive semi-definite matrix of unit trace. Then a *single qubit quantum state* is a density operator, $\rho$, acting on the state space of the qubit, $\mathbb{C}^2$. We denote the set of density operators acting on a Hilbert space $\mathcal{H}$ as $D(\mathcal{H})$. Then again, we can extend this definition to an *$n$-qubit quantum state* by defining it as a density operator acting on the combined tensor product of the individual qubit state spaces, and so lying in $D((\mathbb{C}^2)^{\otimes n})$. Note that a pure state is simply a density

operator of rank one via the identification $|\psi\rangle \leftrightarrow |\psi\rangle\langle\psi|$. More generally, we can define *qudit* quantum states by replacing $\mathbb{C}^2$ with $\mathbb{C}^d$ for any integer $d > 2$; all of the definitions follow analogously.

We call quantum states *mixed* if they are not pure, and so a quantum state $\rho$ is mixed if and only if $Tr(\rho^2) \neq 1$. Note also that any density operator $\rho$ can be diagonalized as

$$\rho = \sum_i \alpha_i |\psi_i\rangle\langle\psi_i|$$

where $|\psi_i\rangle \perp |\psi_j\rangle$ for $i \neq j$ and nonnegative $\alpha_i$ that sum to one. Thus, we can think of any density operator as a probabilistic mixture of orthogonal pure states.

Sometimes, when we have a quantum state defined on a bipartite Hilbert space $V \otimes W$, we would like to describe its state on just one of the subsystems. In this case, we define the *partial trace*

$$Tr_V : D(V \otimes W) \to D(W)$$

as the unique linear operator satisfying $Tr_V(\rho \otimes \sigma) = Tr(\rho)\sigma$. This is the right description of a state on one of its subsystems as it preserves the expectations of local observables. We sometimes refer to a state obtained by tracing out a subsystem as a *reduced state*.

Given any mixed state $\rho \in D(\mathcal{H})$, one can construct a pure state $|\psi\rangle$ called *the purification of* $\rho$ inside $\mathcal{H} \otimes \mathcal{H}'$, where $\dim(\mathcal{H}') \leq \dim(\mathcal{H})$. This state satisfies $Tr_{\mathcal{H}'}(|\psi\rangle\langle\psi|) = \rho$, i.e. $\rho$ is the reduced state of $|\psi\rangle$ in $\mathcal{H}$. For any $\rho$, there are many such purifications $|\psi\rangle$, see [18]. Thus, we can ultimately think of general quantum states in terms of pure states in a sufficiently large Hilbert space.

Finally, we call a quantum state $\rho$ *separable* if it can be decomposed as

$$\rho = \sum_i \alpha_i(\sigma_i \otimes \gamma_i),$$

where the $\alpha_i$ are nonnegative and sum to one; otherwise we call it *entangled*. Note that if $\rho$ is pure, then it is separable if and only if can be identified as a simple tensor in its state space, otherwise it is entangled.

### 2.2 Quantum operations

Now that we understand the basic building blocks of quantum information, how are we allowed to manipulate and access them? For a pure quantum state occupying a state space $\mathcal{H}$, evolution is performed unitarily

$$|\psi\rangle \longrightarrow U|\psi\rangle$$

where $UU^\dagger = I$. Just as different quantum states can be combined as a tensor product of individual states, so too can quantum operations. Namely, if $U$ acts on $\mathcal{H}$ and $V$ acts on $\mathcal{H}'$, then the combined action is $U \otimes V$ on $\mathcal{H} \otimes \mathcal{H}'$. Concretely, for finite-dimensional spaces, we can construct the tensor product as the Kronecker product of matrices.

More generally, we would like to define mappings between general quantum states. In this case, a (general) *quantum channel* is defined as a completely-positive trace-preserving linear map $\Phi : D(\mathcal{H}) \to D(\mathcal{H}')$. Trace-preserving and positivity ensures that a quantum channel maps density operators to density operators. Complete positivity further enforces the condition that, for any $n \in \mathbb{N}$, $(I_n \otimes \Phi)$ is also positive as a map $D(\mathcal{R}) \otimes D(\mathcal{H}) \to D(\mathcal{R}) \otimes D(\mathcal{H}')$ where $\dim(\mathcal{R}) = n$. Choi's theorem on completely positive trace-preserving maps [27] tells us that any such $\Phi : \mathbb{C}^{n \times n} \to \mathbb{C}^{n' \times n'}$ may be written as

$$\Psi(A) = \sum_{i=1}^{N} K_i A K_i^\dagger$$

where $N \leq nn'$ and $\sum_i K_i K_i^\dagger = I$. The set of operators $\{K_i\}$ are called the *Kraus operators* of the channel.

Furthermore, just as we could purify quantum states, so too can we purify quantum channels in the following sense. For any quantum channel $\Psi : D(\mathcal{H}) \to D(\mathcal{H})$, Stinespring's dilation theorem [28] states that there exists a reference system $\mathcal{R}$ such that $\dim(\mathcal{R}) \leq 2\dim(\mathcal{H})$ and $\Psi(\rho) = Tr_K(U(\rho \otimes |\vec{0}\rangle\langle\vec{0}|)U^\dagger)$.

There are a few quantum channels that are particularly important, and although they fit into the above description, deserve special attention. A *positive-operator valued measure* (POVM) is a collection of positive semidefinite operators $\{M_i\}$ acting on a Hilbert space and satisfying $\sum_i M_i = I$. We often refer to the $M_i$ as *POVM elements*. In the special case that the $\{M_i\}$ are orthogonal projectors, we say that this set constitutes a *projective measurement*. One can think of a projective measurement $\{M_i\}$ acting on a quantum state $\rho$ as returning a classical output $i$ with probability $Tr(M_i\rho)$, and conditioned on outcome $i$, mapping $\rho \mapsto M_i\rho/Tr(M_i\rho)$. Finally, we call a quantum channel an *isometry* if it preserves the standard inner product.

### 2.3 Gate sets

Although in principle we are allowed to manipulate quantum information via any unitary, there are certain essential unitary gates that are particularly important. We begin with the *Pauli group* $\mathcal{P}$, which is simply the group of unitaries generated by

the matrices

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \qquad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \qquad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

We can generalize the Pauli group to the *n-qubit Pauli group* $\mathcal{P}^n = \{cP_1 \otimes \ldots \otimes P_n : P_i \in \mathcal{P}, c \in \{1, -1, i, -i\}\}$. We can in turn define the *n-qubit Clifford group* $\mathcal{C}^n$ as the normalizer of $\mathcal{P}^n$ inside $U(2^n)$. These groups are generated by the unitaries

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \qquad P = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}, \qquad CX = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

The famous Gottesman-Knill theorem [29] states that any Clifford circuit is classically simulatable. In fact, the Clifford group along with *any* gate not contained within the Clifford group is universal for quantum computing, in the following sense: a finite gate set $G$ is called *universal* if, for any unitary $U$ and any $\epsilon > 0$, there exists a unitary $V = G_{i_1} G_{i_2} \ldots G_{i_\ell}$ such that $\|U - V\|_2 < \epsilon$.

Some common choices for gates that can supplement the Clifford gates and achieve universality are the $CCZ$ gate, defined by the action

$$CCZ : |a, b, c\rangle \mapsto (-1)^{abc} |a, b, c\rangle ,$$

the $CCX$ or *Toffoli* gate defined by the action

$$CCX : |a, b, c\rangle \mapsto |a, b, (c \oplus ab)\rangle ,$$

and the $\sqrt{P} = T$-gate

$$T : |a\rangle \mapsto e^{ai\pi/4} |a\rangle .$$

The Toffoli gate will play a major role in Chapter 6, and so we remark now that the Toffoli gate, with access to ancilla, is universal for classical *reversible* computing [30]. That is to say, any reversible circuit can be decomposed into a product of Toffoli gates. This leads us to a particularly nice interpretation of quantum computing, as it was shown in [31] that {Toffoli, $H$} constitute a universal gate set for quantum computing. Put another way, quantum computing can be seen as classical reversible computing augmented by an $H$ or *Hadamard* gate.

### 2.4 Distances between quantum states

The are a few norms on quantum states that we will refer to in this thesis. Throughout this discussion, when we write $A \leq B$ for $A$ and $B$ Hermitian operators, then $\leq$ refers to the semidefinite ordering so that $0 \leq B - A$, a positive semidefinite matrix.

The first norm we will define on the set of Hermitian operators is known as the *Schatten p-norm*. For a Hermitian matrix $A$ with singular values $a_1, a_2, \ldots a_n$, and for any $p \in [1, \infty)$, we define

$$\|A\|_p = \left( \sum_{i=1}^{n} a_i^p \right)^{1/p}.$$

We can similarly define the trace distance $T$ between any two quantum states $\rho$ and $\sigma$ as

$$T(\rho, \sigma) = \frac{1}{2} \|\rho - \sigma\|_1.$$

If $\rho$ and $\sigma$ can be diagonalized simultaneously, then each can be thought of as a classical probability distribution. In this case, trace distance is simply the statistical distance between these two distributions. Trace distance also satisfies the nice property

$$T(\rho, \sigma) = \max_{M \leq I} \left( M(\rho - \sigma) \right).$$

Put simply, the trace distance between any two quantum states is the maximum probability of distinguishing those two states using an optimal measurement, where one can think of $M$ as a POVM element.

One more notion of distance between two quantum states is the *fidelity* $F$ between those states. The fidelity is defined as

$$F(\rho, \sigma) = Tr \left( \sqrt{\sqrt{\rho} \sigma \sqrt{\rho}} \right).$$

In the case that $\rho = |\psi\rangle$ and $\sigma = |\phi\rangle$ are pure states, this reduces to $F(|\psi\rangle, |\phi\rangle) = |\langle \psi | \phi \rangle|$. Unlike the trace distance, fidelity is not a metric on the set of density operators, but does satisfy

$$1 - F(\rho, \sigma) \leq T(\rho, \sigma) \leq \sqrt{1 - F(\rho, \sigma)^2}.$$

Finally, Uhlmann's theorem [32] gives a nice interpretation of fidelity in terms of purifications. Namely, for any quantum state $\rho$, let $P(\rho)$ denote the set of purifications of $\rho$. Then for any pair of quantum states $\rho, \sigma$, we have

$$F(\rho, \sigma) = \max_{|\psi\rangle \in P(\rho), |\phi\rangle \in P(\sigma)} |\langle \psi | \phi \rangle|.$$

Throughout this thesis, we may occasionally introduce new notation specific to a section. In particular, in Chapter 6, we will sometimes denote reduced states on a bipartite space as $\rho^A := Tr_B(\rho)$, where $\rho \in D(A \otimes B)$. This is the only portion of the thesis where this notation appears. Otherwise, the notation appearing here will be uniform throughout the text.

# CHAPTER 3

# Quantum error-correction and fault-tolerance

In this chapter, we survey some basics of quantum error-correction and quantum fault-tolerance. We will include those ingredients required for this thesis, but recommend [19] for a more comprehensive accounting. For an alternative and perhaps gentler introduction, we also recommend the survey [33].

## 3.1 Quantum noise processes

The general setup is the following: we have an ideal quantum channel $\mathcal{C}$ which we would like to implement. However, because of physical imperfections in our controls and interaction with the environment, the channel that we actually implement is $\mathcal{E} \circ \mathcal{C}$ where we think of $\mathcal{E}$ as a *noise process*. After repeating many such channels, these noise processes will add up and eventually corrupt the underlying information, and this is what we would like to protect against. Sometimes, we even think of $\mathcal{C}$ as the identity channel acting over some period of time; in this case, we are describing a quantum memory where $\mathcal{E}$ represents the degradation of our information over that time.

We make a few simplifying assumptions, which are physically motivated, about the noise channels that we allow. We assume that our noise acts independently on individual physical qubits. Furthermore, we assume that our noise is *incoherent*, in the sense that it can be modeled as a probabilistic ensemble of local Pauli operations. Although general coherent rotations are an important consideration in quantum noise processes [34, 35], they cannot be described within the stabilizer formalism and so are difficult to model. For this reason, we restrict to two simplified noise models.

**Definition 3.1.** We define the *bit-flip-phase-flip channel* $\mathcal{E}_{bp}$ as the concatenation of two channels $\mathcal{E}_b \circ \mathcal{E}_p$, the bit-flip channel

$$\mathcal{E}_b(\rho) = (1 - p_x) I\rho I + p_x X\rho X$$

and the phase-flip channel

$$\mathcal{E}_p(\rho) = (1-p_z)I\rho I + p_z Z\rho Z.$$

Thus, $\mathcal{E}_{bp}$ is described by $(p_x, p_z)$. When $p_x = p_z$, we refer to both as $p$ the physical error rate of the channel.

**Definition 3.2.** We define the *depolarization channel* $\mathcal{E}_{dp}$ as the channel described by

$$\mathcal{E}_{dp}(\rho) = (1-p)I\rho I + \frac{p}{3}(X\rho X + Y\rho Y + Z\rho Z).$$

Thus, the depolarization channel is described by $p$, which we again refer to as the physical error rate of the channel.

In Chapter 4 we will refer to slightly more general asymmetric noise models, but defer that discussion.

### 3.2 Subspace codes and the Knill-Laflamme recovery conditions

In order to protect a few physical qubits of information, we will encode them into a small number of degrees of freedom of many physical qubits. Then, we define an $[[n,k]]$-*quantum code* as a subspace $C$ of dimension $2^k$ inside a Hilbert space $\mathcal{H}$ of dimension $2^n$. This is simply an encoding of $k$ qubits into $n$ qubits.

We would like such a code to be resilient against some number of physically realistic errors. Given a set of errors $\mathcal{E} = \{E_i\}$ that we would like to protect against, we say our quantum code corrects $\mathcal{E}$ if it satisfies the Knill-Laflamme error-correction criterion,

$$P_C E_i E_j^\dagger P_C = \delta_{ij}$$

for all $E_i, E_j \in \mathcal{E}$, where $P_C$ is the projection onto the codespace $C$ [36]. In this case, for every $E \in \mathcal{E}$, there exists a recovery operation $R_E$ independent of the state of the encoded information that corrects $E$. Note that if $E$ and $F$ are both correctable, so are any linear combination of them. We say a quantum code has *distance* $d$ if any operator acting on at most $d$ qubits can be expressed as a linear combination of elements of the form $E_i E_j^\dagger$. In this case, there exists a recovery procedure by which any error localized on at most $\lfloor \frac{d-1}{2} \rfloor$ qubits can be corrected. We denote the parameters of such a code as $[[n,k,d]]$. An expanded discussion on these conditions will be found in Chapter 6.

To define a logical operator acting on a code, we must choose some fiducial logical basis for the code. In this case, we define a *logical operator* acting on our codespace $C$ as any unitary operator $U : \mathcal{H} \to \mathcal{H}$ that restricts to a map $U : C \to C$. We refer to the induced action on the logical basis states of the code as $U_L$.

### 3.3 Subsystem codes

A *subsystem code* is an encoding of logical information into a subsystem $\mathcal{L}$ of a subspace $\mathcal{C}$. Namely, on Hilbert space $\mathcal{H}$, we have decomposition

$$\mathcal{H} = \mathcal{C} \oplus \mathcal{C}^\perp = \mathcal{L} \otimes \mathcal{G} \oplus \mathcal{C}^\perp.$$

We refer to $\mathcal{G}$ as the gauge degrees of freedom. Logical operators are then simply codespace preserving unitaries $\ell : C \to C$ that come in two flavors: dressed and undressed. Simple undressed logical operators take the form $f \otimes I$, while simple dressed logical operators take the form $f \otimes g$ for some gauge operator $g$.

### 3.4 Stabilizer codes

Although the Knill-Laflamme conditions provide a general description of quantum error-correcting codes, the stabilizer formalism makes these objects manageable [37]. A stabilizer group $S$ is an abelian subgroup of $\mathcal{P}^n$ not containing $-I$. Because its operators commute, we can associate to it a nontrivial *stabilizer subspace code* $C$ defined as

$$C := \{|\psi\rangle : g\,|\psi\rangle = |\psi\rangle\, \forall g \in S\}.$$

Stabilizer codes are also called *additive codes*, as they can be identified with certain additive subgroups of $\mathbb{F}_4^n$. We will use this terminology extensively in Chapter 6.

More generally, we can consider *stabilizer subsystem codes* which are specified by their gauge group $\mathcal{G}$. Up to an automorphism of the Pauli group, we can express the gauge group as

$$\mathcal{G} = \langle Z_1, Z_2, \ldots, Z_s, X_{s+1}, Z_{s+1}, \ldots, X_{s+g}, Z_{s+g}\rangle.$$

Here, $\mathcal{Z}(\mathcal{G})$ is the stabilizer group $S$ for the subspace $C$, $\mathcal{N}(G)$ are the undressed logical operators for the code, and $\mathcal{N}(S)$ are the dressed logical operators for the code. For such a code on $n$ physical qubits, encoding $k = n - s - (g/2)$ logical qubits to distance $d$, the parameters are $[[n, k, d, g]]$. Here, the distance $d$ is equal to the weight of the smallest operator in $\mathcal{N}(S)\backslash G$, the set of logical operators that are not gauge symmetries. In the special case that $\mathcal{G}$ is abelian, $\mathcal{G} = S$ and the resulting code is a stabilizer subspace code.

For any subsystem code specified by $\mathcal{G}$, we can produce new codes by measuring gauge degrees of freedom. Postselecting on outcome $+1$, the resulting gauge group $\mathcal{G}'$ consists of all the elements of $G$ that commute with the measured gauge. This process is called *gauge-fixing*. Note that gauge-fixing does not change the number of encoded qubits $k$, and cannot decrease the distance $d$.

### 3.4.1 CSS codes

*CSS codes* are a special class of stabilizer codes whose gauge group can be presented as $\mathcal{G} = \langle \mathcal{G}_X, \mathcal{G}_Z \rangle$ where $\mathcal{G}_X$ consists solely of $X$-type operators and $\mathcal{G}_Z$ consists solely of $Z$-type operators. These codes have a particularly nice structure, since $X$-type error-correction and $Z$-type error-correction can be performed independently. We sometimes refer to $d_x$ as the weight of the minimal nontrivial $X$-type logical operator, and $d_z$ as the weight of the minimal nontrivial $Z$-type logical operator. We will use codes within this class extensively in Chapter 4.

## 3.5 Homological codes

One particular class of subspace CSS codes that appear implicitly in Chapter 4 are homological codes, and so we briefly review them here. Any subspace CSS code is specified by a stabilizer group that can be presented as $S = \langle S_X, S_Z \rangle$. As all of the elements of $S_X$ and $S_Z$ commute with one another, constructing such a code amounts to ensuring that the elements of $S_X$ commute with the elements of $S_Z$.

We will now detail how to construct such a code from any chain complex of vector spaces. Consider any such chain complex $\mathcal{C}$

$$\ldots \xrightarrow{\partial_{i+2}} C_{i+1} \xrightarrow{\partial_{i+1}} C_i \xrightarrow{\partial_i} C_{i-1} \xrightarrow{\partial_{i-1}} \ldots$$

centered about an index $i$, where each $C_j$ is a vector space over $\mathbb{Z}/2\mathbb{Z}$. Fix a basis for $C_i$, and let $n = \dim(C_i)$. Associate to each such basis element a qubit, so that we have $n$ physical qubits. Fix a basis for $Im(\partial_{i+1})$ and to each element of that basis associate an $X$-type stabilizer. Then, fix a basis for $ker(\partial_i)$ and associate to each element of that basis a $Z$-type stabilizer. Let $S = \langle S_X, S_Z \rangle$, the group of stabilizers generated by the aforementioned $X$- and $Z$-type stabilizers. Note that two elements $s_x \in S_X$ and $s_z \in S_Z$ commute if and only if $\langle s_x | s_z \rangle = 0$ as vectors in $C_i$. This holds for all such $s_x, s_z$ as $\partial_i \circ \partial_{i+1} = 0$, and so we have defined a stabilizer group.

Let us determine the parameters of the associated stabilizer code. To any $X$-type operator $g_x$, we can associate an equivalence class $\overline{g_x}$ where $g_x \sim g'_x$ if $g'_x = s_x g_x$ for some $s_x \in S_x$. This induces a quotient space $C_i / Im(\partial_{i+1})$. Furthermore, $g_x$ is a logical operator if and only if it commutes with $S_Z$, and so it must lie in $\ker(\partial_i)$. Thus, there is an isomorphism between the group of $X$-type logical operators, with the group operation inherited from multiplication in $\mathcal{P}^n$, and the additive group given by $\ker(\partial_i)/Im(\partial_{i+1}) = H_i(\mathcal{C}; \mathbb{Z}/2\mathbb{Z})$, the $i$th homology group of $\mathcal{C}$ with coefficients in $\mathbb{Z}/2\mathbb{Z}$. Thus, $k = \text{rank}(H_i(\mathcal{C}))$ and $d_X$ is the weight of the minimal nontrivial cycle representative inside $H_i(\mathcal{C})$.

We can reverse all the maps and in turn consider the cochain complex

$$\ldots \xleftarrow{\partial^*_{i+1}} C^{i+1} \xleftarrow{\partial^*_i} C^i \xleftarrow{\partial^*_{i-1}} C^{i-1} \xleftarrow{\partial^*_{i-2}} \ldots$$

where $\text{rank}(H^i(\mathcal{C}; \mathbb{Z}/2\mathbb{Z})) = \text{rank}(H_i(\mathcal{C}; \mathbb{Z}/2\mathbb{Z}))$ by the universal coefficient theorem, and in fact the coboundary maps satisfy $\partial^*_{i-k} = \partial^T_{i+k}$. Thus $k$ is well-defined and is precisely $\text{rank}(H_i(\mathcal{C}; \mathbb{Z}/2\mathbb{Z}))$ while $d_z$ is the weight of the minimal nontrivial cocycle representative in $H^i(\mathcal{C})$.

In the special case that such a chain complex is obtained from a cellular decomposition of a manifold, the corresponding homological code belongs to a larger class of codes known as *topological codes*. In this case, we can think of placing qubits on the $i$-dimensional cells of the cellular complex. An $X$-type stabilizer is given by those $i$-cells in the boundary of an $(i+1)$-cell, while a $Z$-type stabilizers is given by those $i$-cells incident to an $(i-1)$-cell.

### 3.5.1 Toric codes

Perhaps the most famous topological code is the toric code in 2-dimensions [38]. It can be realized as a square-cellular decomposition of a 2-torus, with qubits placed on edges. Thus, to each plaquette, there is an associated 4-body $X$-stabilizer, and to each vertex, an associated 4-body $Z$-type stabilizer. We will work extensively with this code in Chapter 4. This code has parameters $[[2L^2, 2, L]]$.

One can easily extend this definition to homological codes defined on higher dimensional tori. In this case, we can consider a cubular cellulation of an $n$-dimensional hypercube with periodic boundary conditions. Placing qubits on $k$-cells, the resulting code will have parameters $[[\binom{n}{k}L^n, \binom{n}{k}, d_X = L^k, d_z = L^{n-k}]]$ as $H_k(T^n) = \binom{n}{k}$.

### 3.5.2 Projective codes

One can obtain a similar family of codes by taking an $n$-dimensional hypercube *without* a cellulation and taking the quotient by the antipodal relation $\vec{e} \sim \vec{e} \oplus \vec{1}$. Again, we emphasize that the only vertices in this construction are the corners of the hypercube. The resulting manifold is $\mathbb{R}P^n$, and so because $H_k(\mathbb{R}P^2; \mathbb{Z}/2\mathbb{Z}) = \mathbb{Z}/2\mathbb{Z}$, we will encode one logical qubit no matter which $k$-dimensional cell we choose.

We call these codes *projective codes*. Such codes have parameters $[[\binom{n}{k}2^{n-k-1}, 1, d_X = \binom{n}{k}, d_Z = 2^{n-k-1}]]$. Note that these codes can scale better than their toric code counterparts when the latter is grown by increasing the dimension $n$ analogously.

## 3.6 Quantum fault-tolerance

Quantum fault-tolerance is based off of a simple idea: encode quantum information in such a way so that local errors become correctable. At every time-step, we

assume that each physical qubit may be corrupted by some error with some probability $p$ independently. Our encoding succeeds in preventing an error if, after that time-step, we successfully diagnose the error and restore our information. Of course, this idealized notion of fault-tolerance is far from a realistic setting. Practically speaking, we must worry about things like time spent diagnosing errors, imperfect controls introducing correlated errors, and actually implementing operations on our encoded information. But first things first.

#### 3.6.1 Fault-tolerance threshold theorem

Without encoding, if each of our devices fails with some probability $p$, then we can only hope to realistically compute circuits of depth $1/p$. We cannot hope to increase the accuracy of our components indefinitely, and so something else is needed.

The fault-tolerance threshold theorem [39, 40, 41] states that, with imperfect devices of a fixed but sufficiently low probability of failure, we can increase the encoding size of our information to allow indefinite computation. The original idea is based off of code concatenation. By concatenating two $[[n, 1, d]]$ quantum codes, we can obtain an $[[n^2, 1, d^2]]$ quantum code. This is both good and bad: we can tolerate more errors, but we've also given more opportunities for errors to occur by increasing the size. Given that the error rate is sufficiently low, one can show that increasing the encoding size will suppress the logical error rate. This is the content of the fault-tolerance threshold theorem: for sufficient accuracy, we can implement arbitrarily long quantum computations with polylogarithmic overhead. It was later shown in [38] that a fault-tolerance threshold exists for certain topological codes *without* appealing to code concatenation.

#### 3.6.2 Thresholds and pseudothresholds

More formally, when we talk about a particular code, we typically mean an infinite code family $\{C_L\}$ indexed by a growing size parameter $L$, along with an implicit decoder. We can then define the *logical error rate* $p_{\log}(p, L)$ as the probability that the decoder will unsuccessfully diagnose an error syndrome and introduce an unintended logical operator into the code. The *accuracy threshold* $p_{\text{thr}}$ is then the largest physical error rate satisfying, for all $p < p_{\text{thr}}$,

$$\lim_{L\to\infty} p_{\log}(p, L) = 0.$$

Plainly, it is the physical error rate below which we can make our computation arbitrarily accurate by increasing the size of our encoding.

For a particular code at a fixed size, one can also define the pseudothreshold as the physical error rate $p$ at which $p_{\log}(p) = p$. Colloquially, this is the physical error

rate below which encoding the information reduces the logical error rate.

#### 3.6.3 ExRec formalism

So far, our discussion has centered around the *code capacity model*, wherein errors occur on the physical qubits comprising the code. However, this assumes that our devices are acting perfectly to diagnose errors and correct them. While the code capacity model provides a nice theoretical upper-bound on our accuracy requirements, we must ultimately account for circuit level errors.

The *circuit error model* assumes that at each time step, every circuit element can fail with some probability. We formally act on any inactive physical qubits in a time step by the identity channel, so that every qubit has some probability of failure in every time step. If a circuit element fails with probability $p$, we replace its action by a random Pauli operator on the support of its wires.

The *extended rectangle formalism* (or ExRec formalism) [39] is a technique for analyzing malignant failures of devices in a circuit. For any effective gate $G$, we assume that $G$ is preceded and followed by an error-correction procedure $EC$. We say that the gate $G$ is *$t$-fault tolerant* if any $t$ circuit element failures in the combined circuit $EC \circ G \circ EC$ can be corrected by a *faultless* error-correction procedure. For $t = 1$, we sometimes just refer to the gate $G$ as *fault-tolerant*.

### 3.7 Fault-tolerant gates

Now that we have our definition of fault-tolerant gates, what are some examples? This is tricky: once we have encoded our data, we might have to act on all of the encoded qubits at once in order to implement a logical gate. This is *not* fault-tolerant: if such a gate fails, it could certainly produce an uncorrectable error as it will compromise all of the qubits of the code! Fortunately, there are several standard constructions for fault-tolerant gates and measurements.

#### 3.7.1 Tranversal gates

Transversal gates are one of the core components of fault-tolerant quantum computing proposals, and one of the central themes of this thesis.

**Definition 3.3.** Given several code blocks of a quantum code $C$, partition the physical qubits from the collective blocks into $T_1, T_2, \ldots, T_\ell$ satisfying, for any $T_i$, that there is at most one qubit from each codeblock in $T_i$. Then any logical gate that we can apply as a product of gates $U_1 \otimes \ldots \otimes U_\ell$ such that each $U_i$ acts only on the physical qubits comprising $T_i$ is called *transversal*.

These gates are automatically fault-tolerant: if any single circuit element fails, it will propagate errors to at most one qubit from each codeblock. As error-correction is performed independently on each block, any such gate failure will be correctable by an ideal error-correction circuit.

Gates of this form are extremely valuable to fault-tolerance proposals, but also extremely restrictive. The discussions in Chapter 6 will focus extensively on quantum error-correcting codes and their associated transversal gates. See Figure 3.1 for a pictorial description. We remark that a tremendously computationally expensive technique known as magic state distillation is often required to implement fault-tolerant gates that are not transversal [42, 43].

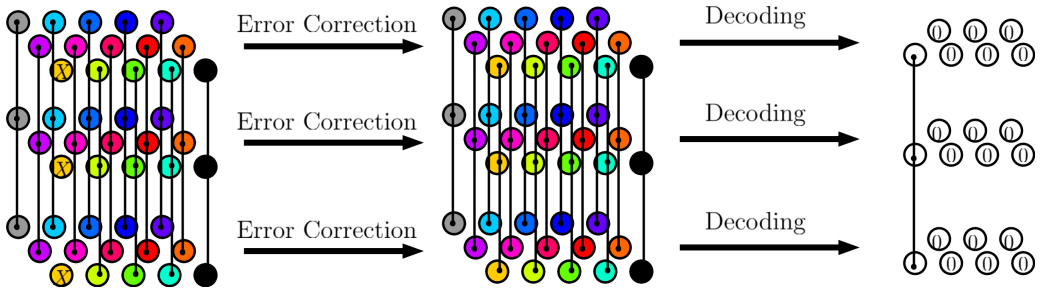



Figure 3.1: An example of transversality. Pictured are three code blocks of the quantum Hamming $[[15, 7, 3]]$ code. The colors correspond to a transversal partition, while the three-dotted lines correspond to a $CCZ$ gate. $CCZ_L$ is realized as the $CCZ^{\otimes 15}$ when the final 6 logical qubits are initialized to $|0\rangle_L$. One of the $CCZ$ gates has failed, producing an $X$-error on each of its supporting orange qubits. Because error-correction is performed independently on each codeblock, the state is recovered and after decoding, we have applied an effective $CZZ$ gate on the first three logical qubits of each block. We say $CCZ_L$ is a *transversal gate* for the $[[15, 1, 3]]$ code, which is obtained by fixing the final 6 logical qubits of the usual $[[15, 7, 3]]$ quantum Hamming code to the $|0\rangle$ state.

### 3.7.2 Fault-tolerant measurement

In order to diagnose errors for error-correction, we must also perform measurements. It is essential to do so in such a way that if measurements fail, they will not propagate errors. There are many popular constructions for doing so, but we include only Shor-style measurement here [44, 4]. The essential idea again is to break the circuit into transversal pieces so that faulty gates do not produce malignant correlated errors. See Figure 3.2 for a pictorial description.

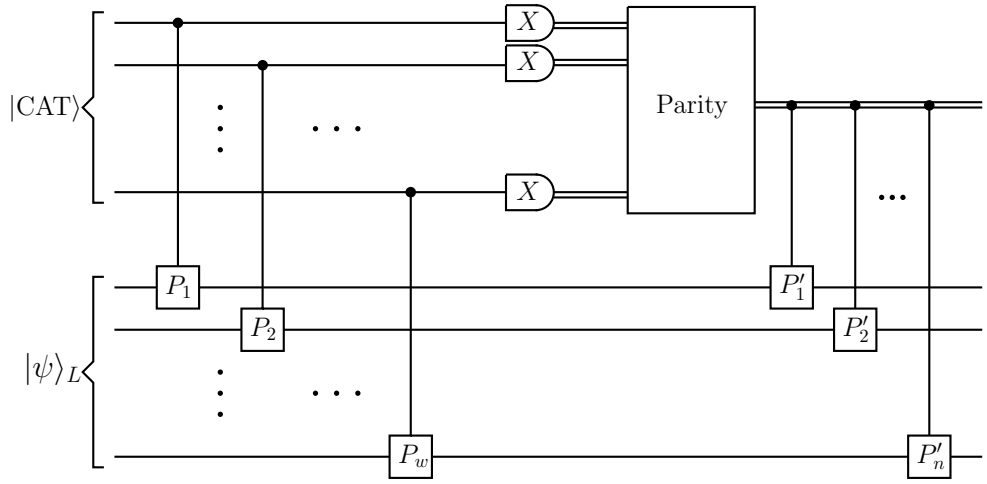



Figure 3.2: A transversal circuit for error-correction using Shor-style measurement. The left-hand side diagnoses the error while the right-hand-size applies a correction conditioned on the outcome. It requires access to verified cat states, which take the form $\frac{1}{\sqrt{2}}(|0\rangle^{\otimes w} + |1\rangle^{\otimes w})$. $P_1 \otimes \ldots \otimes P_w$ is the stabilizer check being measured, while $P'_1 \otimes \ldots \otimes P'_n$ is the correction applied conditioned on the outcome of the check shown and all the other stabilizer checks (not shown). This is also the transversal circuit used in Chapter 7.

## CHAPTER 4

# Intermediate 2-D compass codes

As we have seen, the heart of scalable quantum computing is fault-tolerance. The celebrated quantum threshold theorem [39, 40, 41] ensures us that with sufficiently accurate components, we can perform arbitrarily long quantum computations with polylogarithmic overhead. For physical systems that prefer local interactions, topological codes have emerged as leading candidates for fault-tolerant quantum computation [38, 45, 46, 47, 48]. Among these, the rotated surface code is a particularly enticing candidate, offering depolarization accuracy thresholds in excess of 15% assuming noiseless error-correction with a planar architecture [49].

Another code family which has generated significant interest are the subsystem Bacon-Shor codes [50]. These codes have many desirable properties: their gauge group is 2-local, measurements can be performed with bare ancilla [51], and they support fault-tolerance schemes that obviate magic-state distillation [52]. Unfortunately, while Bacon-Shor codes offer some of the highest concatenated thresholds [51], they fail to have any threshold when grown as a local subsystem family without concatenation [53].

Indeed, asymptotic accuracy thresholds are important for determining the viability of a code. For example, almost all the proposals for 2-D fault-tolerant architectures without magic-state distillation [54, 55, 56] are not expected to exhibit a threshold. With this view, we aim to understand the behavior of code capacity thresholds in toy local systems.

Bacon-Shor codes provide such a system. Rotated surface codes can be realized as a particular gauge-fix of Bacon-Shor codes, and while the former exhibit a threshold, the latter do not. We examine the scaling behavior of thresholds in intermediate gauge-fixed subsystem codes, which we call *intermediate compass codes*. We consider both structured and randomized code families. The former we analyze by estimating the logical error rate with a fixed decoder; the latter we analyze by identifying the phase transition of an associate random-bond Ising model. We observe that,

generically, the threshold scales linearly with the fraction of gauge-fixes.

Finally, we examine the behavior of toy code models in the presence of asymmetric Pauli noise. We observe that these gauge-fixes provide a useful ansatz for tailoring codes to these noise models, providing surface tessellations that match the asymmetry of the noise at some expense to locality.

## 4.1 2-D Bacon-Shor codes

Bacon-Shor codes are defined on an $L \times L$ lattice of qubits, where we assume that $L$ is odd throughout. The gauge group can be presented 2-locally. It is generated by nearest neighbor $XX$ interactions on vertically adjacent qubits, and $ZZ$ interactions on horizontally adjacent qubits. Formally,

$$\mathcal{G} = \langle X_{i,j}X_{i,j+1}, Z_{i,j}Z_{i+1,j} \rangle.$$

The bare logical operators are $X$-type operators acting on a row of qubits, and $Z$-type operators acting on a column of qubits. The stabilizers are then generated by products of adjacent $X$-type row operators and $Z$-type column operators (see Figure 4.1). Bacon-Shor codes then have parameters $[[L^2, 1, L, (L-1)^2]]$.

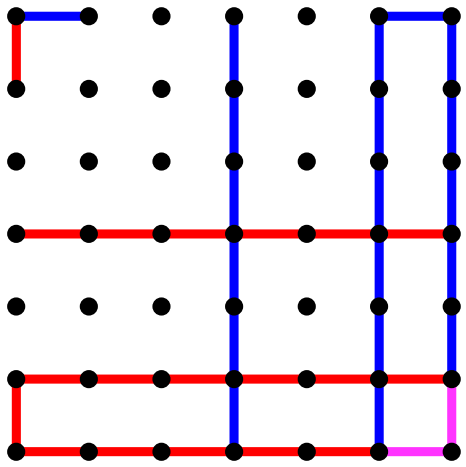


Figure 4.1: The 49-qubit Bacon-Shor code. $X$-type operators are shown as bonds in red; $Z$-type operators are shown as bonds in blue; overlaps are purple. On the top left there are two gauge generators. Spanning the middle of the lattice are undressed logical operators. Spanning the bottom and right of the lattice are stabilizer generators.

## 4.2 Rotated surface codes

Rotated surface codes can be seen as a $\pi/4$-rotation of Kitaev's toric code in the bulk, with modified boundary conditions. They are subspace codes with stabilizer

group generated by an alternating checkerboard lattice of 4-local $X$- and $Z$- type plaquette operators in the bulk. The boundaries are comprised of alternating 2-local edge operators, of $X$-type on the east and west boundaries and of $Z$-type on the north and south boundaries (see Figure 4.2). Thus, logical $X$ operators are strings of $X$-type operators that span the lattice from east to west, and logical $Z$ operators are $Z$-type operators spanning north to south.

Rotated surface codes then have parameters $[[L^2, 1, L, 0]]$. Note that surface codes can be realized as a gauge-fix of Bacon-Shor codes, simply by measuring the corresponding plaquette operators. The boundary operators will then be induced by the initial stabilizers of the Bacon-Shor code.

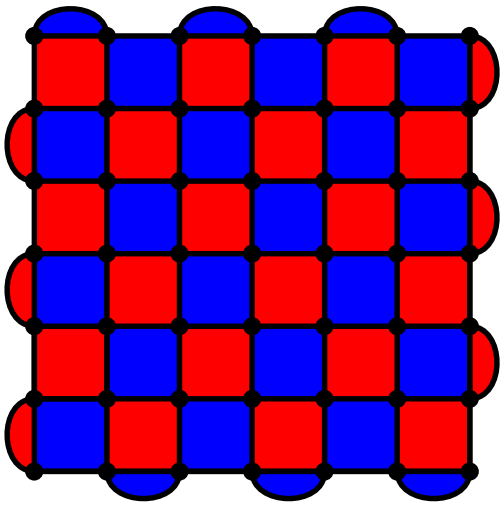


Figure 4.2: The 49-qubit surface code. The red tiling represents $X$-type operators, while the blue tiling represents $Z$-type operators. The bulk stabilizers are 4-local plaquettes, while the boundary operators are 2-local edges.

### 4.3 Intermediate compass codes

We now define the family of intermediate compass codes. Begin with a 2-D Bacon-Shor code defined on an $L \times L$ lattice of qubits with gauge group $\mathcal{G}$.

We designate each of the $(L-1) \times (L-1)$ plaquettes in the lattice as either *$X$-minimal*, *$Z$-minimal*, or *gauge-free*. To each $X$- or $Z$-minimal plaquette, we perform a gauge-fix which ensures that its supporting 2-local $X$- or $Z$-type gauge operators no longer lie in $\mathcal{G}$. For an $X$- or $Z$- minimal plaquette supported on qubits $\{(i,j),(i,j+1),(i+1,j),(i+1,j+1)\}$, this corresponds to measuring the $Z$ or $X$-type gauge operator

$$\prod_{k=0}^{i} Z_{k,j} Z_{k,j+1}; \prod_{k=0}^{j} X_{i,k} X_{i+1,k},$$

respectively. We call any code that can be realized via gauge-fixes of this type an *intermediate compass code*.

Note that fixing a minimal plaquette amounts to cutting the global stabilizer of opposite type into two pieces at that plaquette. Each of the potentially $(L-1)^2$ minimal plaquettes corresponds to fixing one of the $(L-1)^2$ gauge degrees of freedom. Thus, we obtain a subspace code only when there are no gauge-free plaquettes.

For symmetric noise models, we can use the CSS symmetry to argue about general errors in terms of just bit-flip errors. We call an intermediate compass code *CSS-symmetric* if its minimal $X$-plaquettes are mapped to minimal $Z$-plaquettes under a $\pi/4$-rotation of the lattice. Such a code will have identical properties for the correction of $X$- and $Z$-type errors, since they have identical configurations relative to the east-west and north-south boundaries, respectively. For a pictorial description of a CSS-symmetric intermediate compass code, see Figure 4.3.

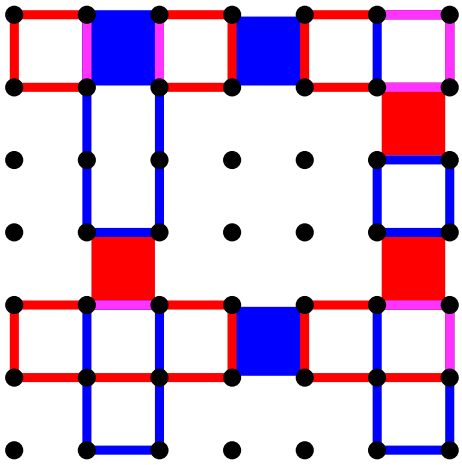


Figure 4.3: A pictorial description of a CSS-symmetric code with three minimal $X$-plaquettes and three minimal $Z$-plaquettes. The three minimal $X$-plaquettes have upper left corner at lattice sites $(4,2)$, $(2,6)$, and $(4,6)$, represented by shaded red blocks. The three minimal $Z$-plaquettes are represented by shaded blue blocks. The edges correspond to the new stabilizers, cut at each minimal plaquette. Red edges are $X$-type, blue edges are $Z$-type, and purple edges a combination.

### 4.3.1 Efficient decoding

Intermediate compass codes have the very nice property that they always support *an efficiently implementable minimum-weight matching decoder*. This is because in any compass code, each qubit is supported on at most two stabilizers of the same type. Thus, excitations form either in pairs or from the boundary, and so matching them will ensure that we return to the code state. Put another way, the corresponding Ising model to any compass code supports at most 2-body interactions. We will detail

the decoding procedure in further depth when analyzing asymmetric noise-tailored codes.

Note that this heuristic does not work in general. For example, color codes can have more exotic excitation structures: their logical string operators can branch, and their excitations need not come in pairs. In this case, the above approach generalizes to hypergraph minimum weight matching, for which there is no known polynomial-time algorithm.

## 4.4 Structured Codes

We first consider two simple intermediate code families, which we call *horizontal* and *vertical* codes. These families emphasize that the structure of the code can significantly affect the behavior of a threshold.

### 4.4.1 Horizontal codes

We define $f(L)$-*horizontal codes* as CSS-symmetric codes for which the first $(L - f(L))$ rows consist entirely of minimal $X$-plaquettes, and the last $f(L)$ rows contain no minimal $X$-plaquettes. Informally, these are codes for which the first $(L - f(L))$ rows behave as the surface code, and the last $f(L)$ rows behave as the Bacon-Shor code (see Figure 4.4). In particular, 0-horizontal codes are surface codes and $L$-horizontal codes are Bacon-Shor codes.

Using techniques similar to [53, 57], one can show analytically that these codes will often fail to have a threshold altogether. Essentially, a dominating portion of the lattice must be equivalent to the surface code in order for this family to exhibit a threshold.

**Proposition 4.1.** *The family of $f(L)$-horizontal codes fail to have a threshold whenever $f(L) = \omega(\log(L))$.*

*Proof.* Without loss of generality, we consider $X$-type errors and assume that every qubit in the first $(L - f(L))$ rows is noiseless; certainly, this only reduces $p_{log}$. The remaining $Z$-type stabilizers whose support intersects noisy qubits are those double column stabilizers extending down from the minimal $X$-plaquettes.

Given any physical error in this model, we can always multiply by an $X$-gauge operator to ensure that all bit flip errors occur on the last row of qubits. We can thus model the code as a repetition code with qubits experiencing an effective noise $p_{\text{eff}}(p, f(L))$. In particular, a qubit experiences an effective bit flip error if the total

number of bit flip errors in its column is odd. Thus,

$$\begin{aligned} p_{\text{eff}} &= \frac{1}{2}\left(((1-p)+p)^{f(L)} - ((1-p)-p)^{f(L)}\right) \\ &= \frac{1}{2}\left(1-(1-2p)^{f(L)}\right). \end{aligned}$$

This reduces the problem to investigating the threshold of a repetition code with an effective physical noise that scales with its length. The repetition code then fails with probability

$$p_{log} = \sum_{k=\frac{L+1}{2}}^{L} \binom{L}{k} p_{\text{eff}}^{k} (1-p_{\text{eff}})^{L-k}.$$

Consider $X_L \sim \mathcal{B}(L, p_{\text{eff}})$, the binomial distribution on $L$ trials with bias $p_{\text{eff}}$. We want to evaluate $\Pr[X_L > L/2]$ in the $L \to \infty$ limit. We can approximate using the normal distribution. For $Y \sim \mathcal{N}(0,1)$,

$$\begin{aligned} X_L &\approx L p_{\text{eff}} + \sqrt{L p_{\text{eff}}(1-p_{\text{eff}})}Y \\ &= L\left(\frac{1-(1-2p)^{f(L)}}{2}\right) + \sqrt{\frac{L(1-(1-2p)^{2f(L)}}{4}}Y \end{aligned}$$

Furthermore, $X_L > \frac{L}{2}$ occurs precisely when $Y > y_L := \frac{\sqrt{L}(1-2p)^{f(L)}}{\sqrt{(1-(1-2p)^{2f(L)})}}$. To evaluate the presence of a threshold, we can choose any $p > 0$. Then for a threshold to exist, we would need that

$$\lim_{L\to\infty} \frac{\sqrt{L}(1-2p)^{f(L)}}{\sqrt{(1-(1-2p)^{2f(L)})}} > 0.$$

Rearranging, we see that this condition fails to be met when

$$(1-2p)^{-f(L)} = \omega(L).$$

This is whenever $f(L) = \omega(\log(L))$, independent of $p$.

The error of our approximation can be bounded by the Berry-Esseen theorem. Namely, defining $x_L \sim \mathcal{B}(1, p_{\text{eff}})$, for some constant $c > 0$, it holds that

$$\|\Pr[X_L > L/2] - \Pr[Y > y_L]\|_1 < \frac{c \cdot \mathbb{E}[(x_L - p_{\text{eff}})^3]}{\text{Var}(x_L - p_{\text{eff}})^{3/2}\sqrt{L}}.$$

As the prefactor on the right is uniformly bounded in $L$, it follows that the sequence $\Pr[Y > y_L]$ converges to $\Pr[X_L > L/2]$ asymptotically as $\frac{1}{\sqrt{L}}$, and so the code family cannot exhibit a threshold whenever $f(L) = \omega(\log(L))$. □

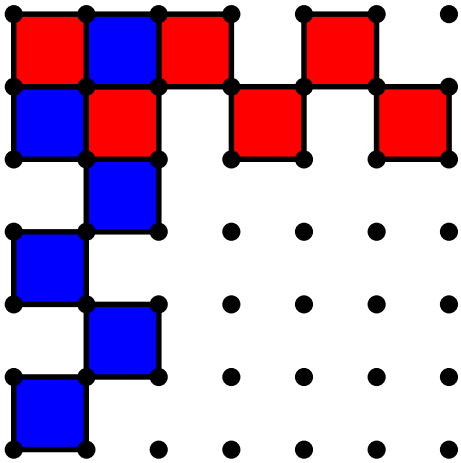


Figure 4.4: An illustration of the $(\frac{5}{7}L)$-horizontal code on 49 qubits. The red and blue plaquettes represent minimal $X$- and $Z$- plaquettes, respectively. One obtains the $(\frac{5}{7}L)$-vertical code on 49 qubits by reflecting the minimal $X$-plaquettes about $y = -x$ and the minimal $Z$-plaquettes about $y = x$.

### 4.4.2 Vertical codes

We define *$f(L)$-vertical codes* similarly as CSS-symmetric codes for which the first $f(L)$ columns consist entirely of minimal $X$-plaquettes, and the last $(L - f(L))$ columns contain no minimal $X$-plaquettes. Intuitively, one would expect these codes to exhibit a threshold for $f(L) = \theta(L)$, as any excitation would have to propagate through a bulk of surface code in order to introduce a logical error.

We simulate the error threshold for such codes using a slightly modified minimum-weight perfect matching decoder on the surface code bulk, efficiently implementable using Edmond's algorithm [58]. In the limit of large lattice sizes, the global stabilizers spanning the boundary of the surface code bulk will flip with probability approaching 1/2. As such, we decode according to the local stabilizers alone, and use the boundary information to project to a code state within the Bacon-Shor type bulk.

To compute the logical error rate, we borrow a technique from [49]. We sample weight $k$ errors uniformly at random to estimate the probability that a weight $k$ error may cause a failure, $f_k$, and use this to estimate the logical error rate as

$$p_L(p) = \sum_{k=\lceil \frac{d-1}{2} \rceil}^{L^2} f_k \binom{L^2}{k} p^k (1-p)^{L^2-k}. \tag{4.1}$$

We overestimate the error-rate slightly for the sake of computational simplicity. We make the approximation in our trials that if $|f_k - 1/2| < 0.005$, then $f_j \approx 1/2$ for all $j > k$. Over several test trials, this never accounted for more than 0.08% absolute error.

We observe that this family has a threshold which is independent of $f(L)$ as long as $f(L) = \theta(L)$ (see Figure 4.5). This is expected as threshold behavior is a *local* property: although at any finite size these globally asymmetric lattices account for higher failure rates, these finite size effects are suppressed in the limit. It would be interesting to examine cases in which $f(L) = o(L)$, but more difficult to simulate as this will further amplify finite-size effects.

## 4.5 Randomized Codes

Motivated by the observation that local structure determines threshold behavior, we consider randomized code constructions to simulate "locally intermediate" behavior. In order to analyze such codes, we cannot hope to define fixed decoders to estimate $p_{\log}$. Instead, we use a randomized decoding strategy and a well-studied connection to statistical mechanics [38, 59] to probe thresholds.

### 4.5.1 Randomized Decoders

Consider a decoder and error-correction procedure for a subsystem code specified by gauge group $\mathcal{G}$. Each Pauli error $E$ is representative of its *error-class* $\overline{E} := \{GE\}_{G\in\mathcal{G}}$. The decoder will map an observed syndrome $s$ to a candidate error-class to correct.

The optimal decoder is the *maximum-likelihood decoder* (ML-decoder) which chooses, for each syndrome $s$, the optimal error-class $\overline{E}$ maximizing $\Pr\big[\overline{E}|s\big]$. Letting $\mathcal{S}$ denote the set of all syndromes and $\overline{E}_s$ denote this optimal choice of $\overline{E}$ conditioned on syndrome measurement $s$, we can then express our success probability as

$$1 - p_{log} = \sum_{s\in\mathcal{S}} \Pr\big[\overline{E}_s\big].$$

Practically speaking, this decoder is often *not* efficiently implementable. Furthermore, it will be difficult to determine this fixed decoder for any randomized code families. To alleviate this difficulty, one can use a suboptimal decoder that connects to statistical mechanics. The decoder is probabilistic, and conditioned on syndrome measurement $s$, chooses to correct error-class $\overline{E}$ with probability $\Pr\big[\overline{E}|s\big]$. We can then express the success probability of this decoder as

$$1 - p_{log} = \sum_{E} \Pr[E] \cdot \Pr\big[\overline{E}|s_E\big].$$

This emphasizes the importance of error-class probabilities, and not the probabilities of individual errors. Note that although this decoder performs worse at any finite size, it will share the same threshold as the ML-decoder.

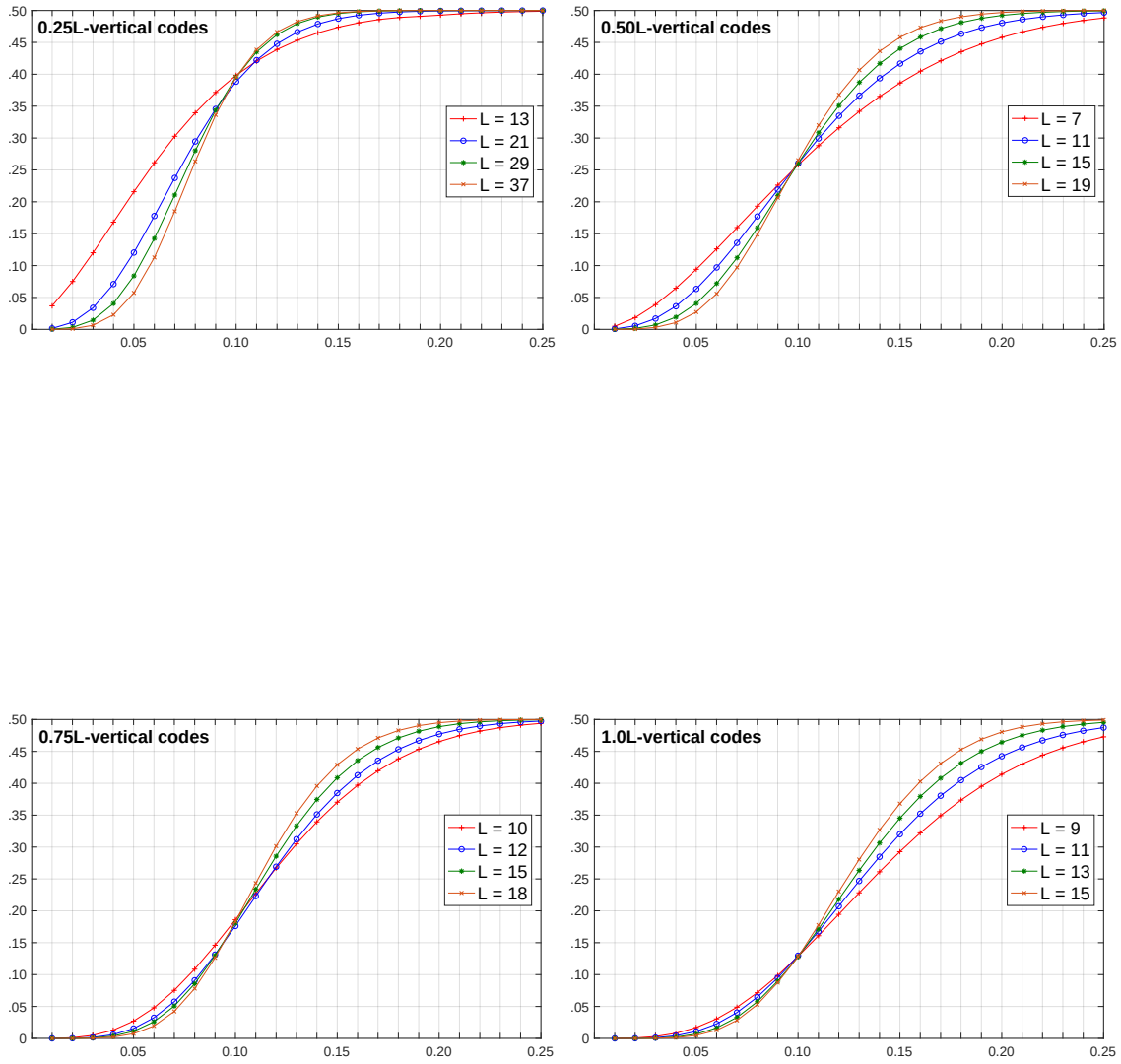



Figure 4.5: Physical vs logical error rates for various $f(L)$-vertical codes. The standard deviation is less than the size of the markers. Our plot agrees with previously evaluated minimum weight matching decoders at a 10.33% threshold for $f(L) = L$, up to statistical error. The coefficient on $L$ amplifies the finite-size effects, but leaves the threshold the same.

### 4.5.2 Mapping to statistical models

We recall a well-known connection relating regular code families to associated statistical models. The presence of a phase transition in the statistical model indicates, though does not imply, an accuracy threshold for the family. We will construct an intermediate code family with an associated Ising model defined on random graphs generated according to a parameter $q$. The parameter $q$ captures the fraction of minimal $X$-plaquettes that appear in the code family, conditioned on there existing a gauge-fix transforming any member of the family into the surface code.

For varying $q$, if the resulting model is ferromagnetic at temperatures *above the Nishimori line* for sufficiently small $p$, then this indicates an accuracy threshold for the code family. While we detail the construction only for $X$-errors arising from the bit flip channel and depending exclusively on $\mathcal{G} := \mathcal{G}_X$, it can be extended to more general codes [59].

Let $\mathcal{G}_0$ be a minimal generating set of $\mathcal{G}_X$. Let the $g_i \in \mathcal{G}_0$ be indexed by $i$, and associate to each generator an Ising spin $s_i = \pm 1$. Index the physical qubits by $j \in \{1, \ldots, L^2\}$ and define

$$g_i(j) := \begin{cases} 1 & \text{if } g_i \text{ is supported on site } j \\ 0 & \text{otherwise.} \end{cases}$$

Then for any vector $\tau \in \{+1, -1\}^{L^2}$, we define the classical spin Hamiltonian

$$H_\tau(s) = -\sum_{j=1}^{L^2} \tau_j \prod_{i=1}^{|\mathcal{G}_0|} s_i^{g_i(j)}.$$

For any Pauli $X$-error, define $(\tau_E)_k$ to be $-1$ if $E$ is supported on site $k$, and $+1$ otherwise. Then for any $g \in \mathcal{G}$, letting $s_g$ denote the corresponding binary string, we have

$$H_{\tau_g}(s) = H_{\tau_1}(s_g s).$$

For inverse temperature $\beta$, let $\mathcal{Z}$ be the partition function

$$\mathcal{Z}(\tau, \beta) = \sum_s e^{-\beta H_\tau(s)}.$$

For physical error-rate $p$, we can define the virtual temperature $\beta_p$ according to the *Nishimori line* [38] so that

$$\beta_p := \frac{\log(1-p) - \log(p)}{2}.$$

Along this line, we can relate the partition function of the statistical model to the probability of certain error-classes occurring. Namely, for individual errors $E$,

$$\Pr(E) = (2\cosh(\beta_p))^{-L^2} e^{-\beta_p H_{\tau_E}(s_1)}.$$

As $\Pr\left[\overline{E}\right] = \sum_G \Pr[EG]$ for representative $E$, it follows from the previous observation on gauge translations that

$$\begin{aligned}\Pr\left(\overline{E}\right) &= \sum_{g\in\mathcal{G}} (2\cosh(\beta_p)))^{-L^2} e^{-\beta_p H_{\tau_E}(s_g s_1)} \\ &= \sum_{s} (2\cosh(\beta_p)))^{-L^2} e^{-\beta_p H_{\tau_E}(s)} \\ &= (2\cosh(\beta_p)))^{-L^2} \mathcal{Z}(\tau_E, \beta_p).\end{aligned}$$

This expresses error-class probabilities in terms of the statistical model. Because our codes encode a single logical qubit, we can simply notation as each syndrome corresponds to two unique error-classes $\overline{E_1}, \overline{E_2}$ related by $\overline{X_L E_1} = \overline{E_2}$ for $X_L \in \mathcal{N}(\mathcal{G})\backslash\mathcal{G}$. Define $\tau$ to be a quenched random variable that takes value $\tau_E$ with probability $p^{|E|}(1-p)^{L^2-|E|}$. Under this randomly-disordered statistical model, we can express our success probability $1 - p_{\log}$ using the randomized decoder as

$$\begin{aligned}&= \sum_E \Pr(E) \Pr\left[\overline{E}|s_E\right] \\ &= \sum_E \Pr(E) \left(\frac{1}{1 + \frac{Z(\tau_{EX_L},\beta_p)}{Z(\tau_E,\beta_p)}}\right) \\ &= \left[(1 + \exp\{-\beta_p \cdot (F(\beta_p, \tau_{EX_L}) - F(\beta_p, \tau_E))\})^{-1}\right]_p\end{aligned}$$

where $[\cdot]_p$ is the average over the random variable $\tau$ distributed according to $p$ and $F$ is the free energy. If our success rate approaches unity, then the free energy cost of introducing a (nontrivial) domain wall $X_L$ diverges with the system size. Since $X_L$ also grows with the size of the lattice, this occurs when the randomly-disordered statistical model experiences a particular phase transition. Thus, finding a transition to the ferromagnetically ordered phase indicates that the free energy cost of introducing a growing domain wall will diverge with $L$, suggesting a threshold [38]. Conversely, temperatures in the disordered phase will *always* be above threshold, and so it is meaningful to study phase transitions of the associated model.

To summarize, if the corresponding statistical model occupies a ferromagnetic phase at temperatures $T$ above the Nishimori line, then it will be ordered at the correct virtual temperature, giving evidence that the corresponding code family will exhibit an accuracy threshold at physical error rate $p$.

### 4.5.3 Randomized code families

We define $f(L)$-*randomized codes* stochastically in the following way. For a lattice of linear dimension $L$, for each $X$-type plaquette in the checkerboard configuration of the rotated surface code, we fix the gauge so that the $X$-plaquette is minimal with probability $2f(L)/(L-1)^2$. The factor $(L-1)^2/2$ is simply the total number of $X$-type plaquettes in the checkerboard configuration, so that $f(L)$ is the expected number of minimal $X$-plaquettes in the resulting lattice.

In particular, we consider $\frac{q}{2}(L-1)^2$-randomized codes for different $0 < q < 1$, which we call $q$-*codes*. Plainly, these are codes for which a $q$-fraction of the $X$-plaquettes are minimal (see Figure 4.6). Thus, 0-codes are Bacon-Shor codes, and 1-codes are surface codes.

We map these codes to corresponding anisotropic Ising models on random graphs defined according to the stochastic gauge-fixing procedure (see Figure 4.7). We analyze the model using virtual inverse temperature $\beta$ and disorder $p$ along the Nishimori line to estimate the phase transition. See also Appendix B for a visualization of the $q$-code model during thermalization in different parameter regimes. We find that the accuracy threshold scales linearly with $q$, suggesting that the threshold depends linearly on the expected connectivity of the lattice in this restricted family.

### 4.5.4 Parameters of the Ising model simulation

To get the data points on this figure, we generate random samples of the random-bond Ising model with the given $q$ and $p$ for various system sizes $L$. The temperature is determined by the Nishimori line from the disorder parameter $p$. For each random trial, we use a cluster algorithm to compute the Binder cumulant [60]. Finally we scan over $p$ at a separation of 0.1 for $\ln(p)$ and look for the crossing point of each of these curves for different $L$, giving the transition point.

The system size we use ranges from $L = 5$ to $L = 61$, and the number of steps for the cluster update ranges from $10^6$ to $5 \times 10^8$. The number of random trials for each $q$, $p$ and $L$ range from 80 to $10^4$. Because of the limited computational power and the relatively small system size, we apply periodic boundary conditions to reduce the finite-size boundary effect. In general, as the transition point $p_{\text{th}}$ increases with $q$, it enhances the frustration in the system and so more steps are needed for convergence, which is verified by the autocorrelation length of the observables. On the other hand, for larger $q$ the slope of the Binder cumulant $U$ with respect to $-\ln(p)$ also increases, and so fewer samples and smaller system sizes are required to achieve the same level of accuracy.

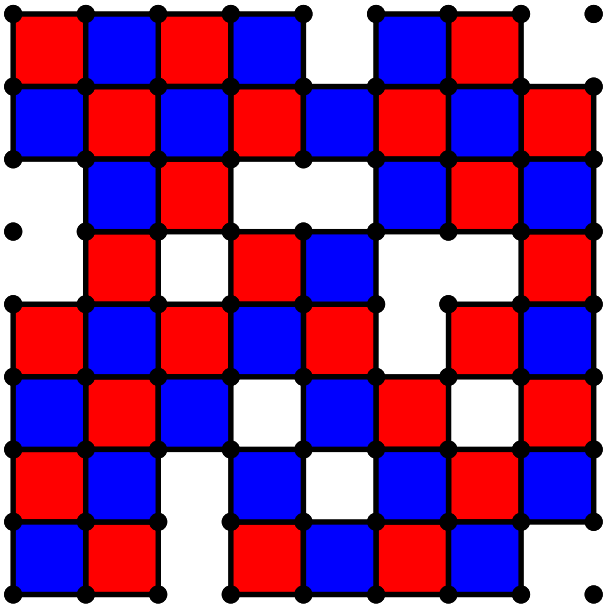


Figure 4.6: The $X$- and $Z$- minimal plaquettes featured as red and blue plaquettes respectively on a $9 \times 9$ lattice. The above was generated as an instance of a $q$-code with $q = \frac{3}{4}$.

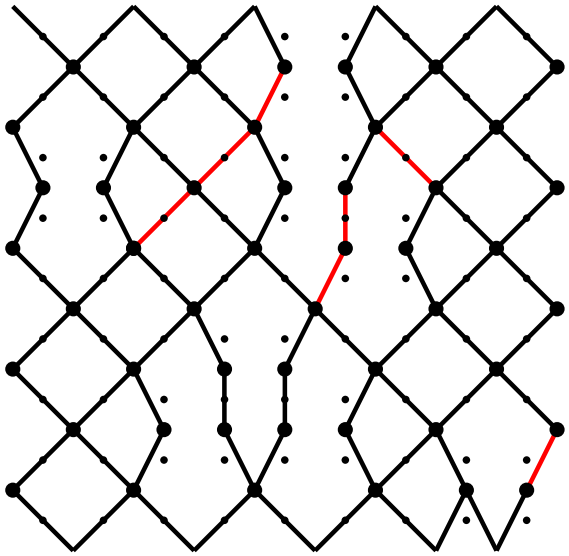


Figure 4.7: The associated random-bond Ising model to the $q$-code described in Figure 4.6. The smaller dots represent qubits, the larger dots represent Ising spins. Black edges represent ferromagnetic spin interactions; red edges represent antiferromagnetic interactions. North and south boundaries experience an external magnetic field. This model was generated with $q = \frac{3}{4}$ and with disorder $p = \frac{1}{10}$.

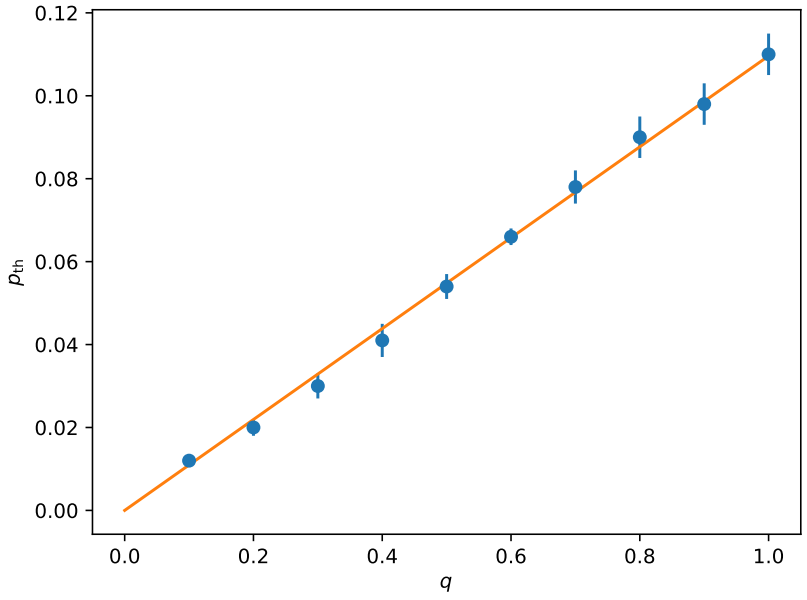



Figure 4.8: Computed threshold $p_{\text{th}}$ from the random-bond Ising model vs. the parameter $q$. The orange line is a linear fit through the origin.

## 4.6 Asymmetric noise tailored codes

We next consider intermediate *subspace* compass codes, and analyze their behavior in the presence of asymmetric Pauli noise. The first of these families amounts to particular surface tessellations, while the second is constructed stochastically. In both cases, the models remain local.

In realistic physical systems, Pauli error is rarely unbiased, with dephasing noise $p_z$ a dominating factor. For this reason, we define *$\eta$-biased Pauli noise with physical error rate $p$* similarly to [61].

For the remainder of this section, we define $p = p_x + p_y + p_z$ where $\eta := p_z/(p_x + p_y)$, and for simplicity, we assume $p_y = p_x$. At $\eta = 1/2$, this is the usual depolarizing channel with physical noise $p$, and as $\eta \to \infty$, this becomes the pure dephasing channel with noise $p$.

Unlike the recent work in [61], we consider modifying the codes directly while maintaining independent $X$- and $Z$- type decoding. A similar mapping to a redundant syndrome set for $Z$-type errors with a correlated $X, Y$ syndrome decoder should push thresholds up even further.

### 4.6.1 Elongated codes

We define a set of intermediate compass code families parametrized by $\ell \in \mathbb{N}^+$. These codes are constructed by fixing the $(i, j)$-th plaquette to be a minimal $X$-plaquette if $i - j \equiv 0 (\text{mod } \ell)$. The remaining plaquettes we fix to be minimal $Z$-plaquettes, resulting in a subspace code. We call the resulting family $\ell$-*elongated codes* (see Figure 4.9).

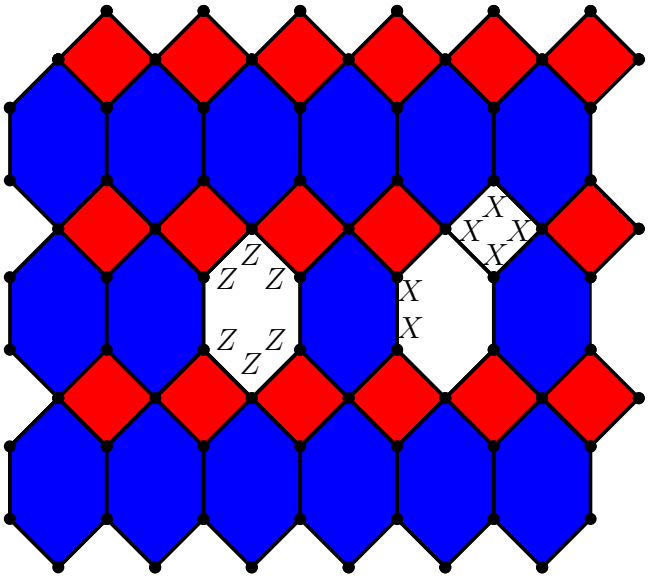



Figure 4.9: The bulk of a 3-elongated code. After deformation, 3-elongated codes form a $[3.6.3.6; 3^2.6^2]$ plane tiling with pairs of triangles identified along their shared edge. Each hexagon is a $Z$-type stabilizer, while each plaquette is an $X$-type stabilizer. There is an additional 2-local $X$-type stabilizer on any edge shared by adjacent hexagons.

Under this definition, we obtain Shor's code [44] for $\ell = 1$ and the surface code for $\ell = 2$. For $\ell > 2$, we obtain an asymmetrization of Kitaev's toric code in the bulk with extended $2\ell$-body plaquette operators. These asymmetric topological codes will naturally behave better in the presence of similarly asymmetric noise, while sacrificing somewhat in locality. The analysis of this behavior using a generalized minimum-weight matching decoder can be found in Figure 4.10. Table 4.1 summarizes the code parameters in the $\eta$-biased Pauli noise model.

### 4.6.2 Randomized bias codes

In this last section, we consider randomized intermediate compass subspace code families. Intuitively, minimal $X$-plaquettes help to correct $X$-errors: each such plaquette represents a cut in the corresponding $Z$-type stabilizer, which allows it to collect more information locally. In the extreme case, performing each cut results in

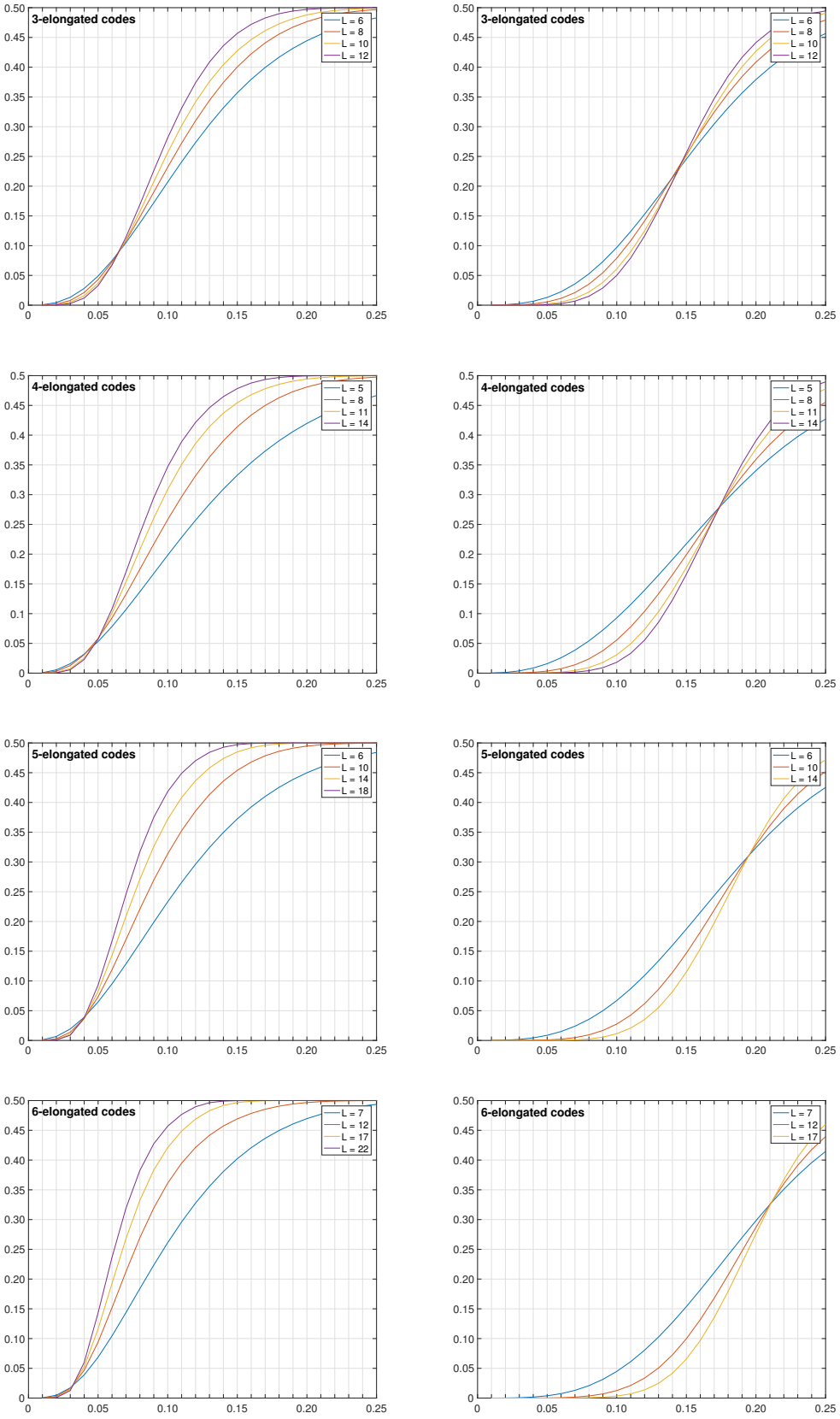



Figure 4.10: Physical vs logical error rates for $\ell = 3, 4, 5, 6$ elongated codes. The plots on the left are for bit-flip errors; the plots on the right are for phase errors.

| $\ell$ | $\eta_{\text{opt}}$ | $p_{\text{thr}}$ | $\eta_*$ | $p_z$ | $p_x$ |
|---|---|---|---|---|---|
| 3 | 1.75 | 17.85% | 1.39 | $14.1\% \pm 0.3\%$ | $6.5\% \pm 0.15\%$ |
| 4 | 3.02 | 20.11% | 2.10 | $17.5\% \pm 0.2\%$ | $5.0\% \pm 0.15\%$ |
| 5 | 4.26 | 21.56% | 2.78 | $19.5\% \pm 0.1\%$ | $4.1\% \pm 0.10\%$ |
| 6 | 5.89 | 22.74% | 3.70 | $21.1\% \pm 0.1\%$ | $3.3\% \pm 0.10\%$ |

Table 4.1: The parameters of $\ell$-elongated codes using the minimum weight matching decoder. Here, $\eta_{\text{opt}}$ is the bias yielding the best threshold $p_{\text{thr}}$ for each code, while $\eta_*$ is the bias above which the code will always outperform the surface code. Finally, $p_z$ and $p_x$ are the dephasing and bit-flip thresholds, respectively.

Shor's code, the quantum analogue of the repetition code.

With this in mind, for $q \in [0, 1]$, we define *$q$-randomized bias codes* as a family of random codes. Each member of the family is obtained by selecting each plaquette independently to be $Z$-minimal with probability $q$. We analyze these codes as well, providing a picture of their behavior; see appendix Figures 1 through 3 for an example.

Using the statistical model to threshold correspondence detailed previously, one could in principle estimate the thresholds of these randomized code families, but the rich connectivity makes this difficult. We again estimate the behavior of these codes using the following minimum weight-matching decoder schematic.

For different code sizes, we sample 100 different codes from the $q$-randomized bias code family. Then for each such code, we build a decoder graph $G$ from its associated Ising model by identifying spins as vertices and interactions as unweighted edges. Here, it is essential that the Ising model associated to any intermediate compass code supports at most 2-body interactions.

For each error weight $k$, we sample 5000 random error configurations and generate the corresponding subgraphs in $G$. We then add and connect boundary vertices to ensure a perfect matching exists [49]. The minimum weight perfect matching is computed using a combination of Djikstra's algorithm and Edmond's blossom algorithm. Finally, we average the probability of success over all samples, and use these to compute Equation (4.1).

See appendix Figures 1 through 6 for a pictorial description of randomized bias codes. We expect that the randomization in these codes may cause a significant gap between the performance of this decoder and the optimal decoder. Figures 5 and 6 emphasize the duality between the $X$-type and $Z$-type Ising models associated to these codes. The sparsity of the $X$-type model corresponds simultaneously to a lower $X$-type phase transition, as well as a more robust $Z$-type error decoder, and vice versa. Intuitively, the gap between the minimum-weight matching decoder per-

formance on one model and the thermodynamic stability of its dual should indicate the expected loss. Again, this is because the minimum weight matching decoder identifies the most probable error given a syndrome, rather than the most probable *error class*.

We observe that the threshold drops slightly for these randomized codes relative to surface codes, and increases slightly compared to the similar family of 4-elongated codes. However, these are relative minor changes in threshold, reinforcing that without problematic structure in our code (such as that in section 4.4.1), the threshold scales roughly linearly with the overall fraction of stabilizer checks for that particular type of error. See Figure 4.11 for a snapshot of their behavior. It is important to note that these codes *do* remain local up to a logarithmic factor for any $q \in (0, 1)$. This is because the maximum weight of any $X$-stabilizer is at most the number of horizontal consecutive $Z$-fixes. In expectation, the longest such chain of $Z$-fixes is $\mathcal{O}(\log(L))$ with a constant depending on the bias.

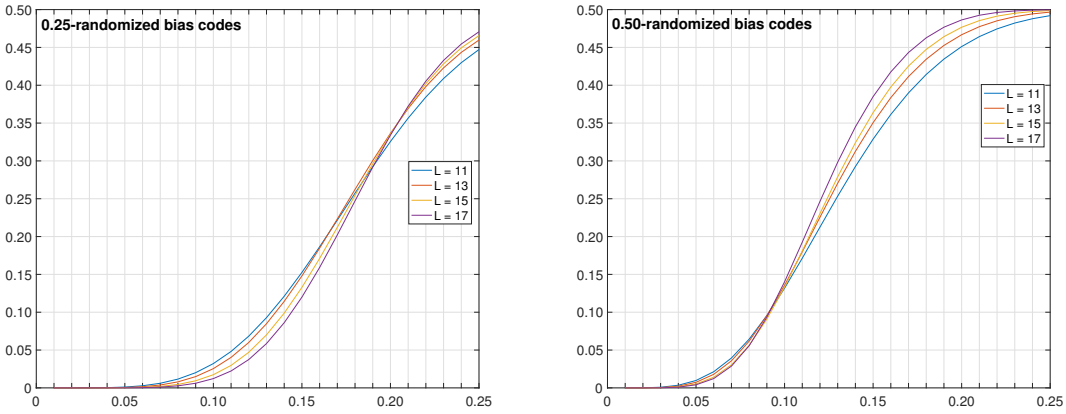



Figure 4.11: Physical vs logical error rates for randomized bias codes, with bias $q = 0.75$ and $q = 0.50$, respectively. The thresholds roughly mirror those of the comparable surface code and 4-elongated code.

## 4.7 Conclusion and future work

In this chapter, we have compared the relative behavior of error thresholds for intermediate compass code families. In particular, we compare threshold results for both structured and randomized intermediate compass code families. We find that the threshold rate scales roughly with the expected *local* connectivity, subject to the constraint that a gauge-fix to the surface code is possible. It would be interesting to relate this to proposals for quantum codes defined on fractals [62] where infinite ramification order (rather than large Hausdorff dimension) seems to be the requisite

property for an ordered model at finite temperature.

We have also identified simple code families within this family that behave better with respect to asymmetric Pauli noise, at some cost to the locality of the codes. As realistic system often exhibit biased noise, schemes that use this bias to increase threshold may prove useful in the future. Of course, there the important consideration of whether this bias propagates down to the circuit level, which we must ultimately consider.

As we have only touched on the design space and properties of such codes, there is much left open about such codes. First and foremost, there is the barrier of transferring from the code capacity error model to the circuit model. Can one hope to design intermediate codes in a way that reduces the effect of correlated errors, similar to the hook errors in surface codes [63] or bare-ancilla error correction in the Bacon-Shor [51] and Bare [[7,1,3]] codes [64]. Ultimately, whatever gains we make in the code capacity model must mitigate losses incurred from reduced locality.

It would be interesting to generalize this family to higher rate codes using techniques from [65]. The difficulty is in defining a model which ensures that the stabilizer cuts of opposite type commute with one another. Investigating a similar approach using the 3-D compass model [50] might prove interesting as well.

Finally, we have only given simple examples of codes within this model. It is reasonable to expect, for example, that $\ell$-elongated codes are not optimal at the intermediate biases we consider. Are there other configurations that perform better? Intuitively, the insertion of minimal $Z$-plaquettes helps to correct $Z$ errors locally around them. These codes may even be tailored to improve error-rates on correlated local noise models, where the bias varies continuously along the lattice. As these noise models may be more physically realistic, we hope to investigate this possibility in future work.

# CHAPTER 5

# Quantum homomorphic encryption and its limitations

Fully homomorphic encryption is one of the great advances of modern cryptography. First discovered by Gentry in 2009 [66], it allows one to delegate the processing of *encrypted* information by a party without access to the secret key. In classical computing, an enormous body of work has gone into developing and optimizing this protocol (see [67] for a summary).

As the development of large scale quantum computers progresses, we must consider the cryptographic consequences of their arrival. While quantum computers can bolster the security of some cryptographic protocols [7], they can also obviate the security of others [68]. Fortunately, the security of existing homomorphic encryption schemes is derived from hard problems on lattices [69, 70], which are expected to be computationally difficult for quantum computers to solve. Nonetheless, it is natural to ask:

*Can quantum computers allow homomorphic encryption schemes which exhibit information-theoretic, rather than computational, security?*

We detail why the answer to this question is many-fold and subtle, but for the strongest security definitions, show that the answer is no [21, 13, 11, 71].

## 5.1 Classical homomorphic encryption

A (classical) homomorphic encryption scheme is typically an asymmetric key encryption scheme with an additional functionality, called evaluation. This functionality allows a third party, in possession of a ciphertext, to meaningfully manipulate the underlying plaintext without possessing the secret key. Formally, a homomorphic encryption scheme HE is a four-tuple of (randomized) algorithms.

**HE.KeyGen**$(1^\kappa, 1^L) = (pk, sk, evk)$. A key generation algorithm that accepts security parameter $\kappa$ and evaluation parameter $L$. It outputs the public key $pk$

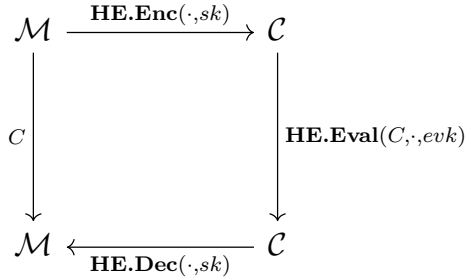
and secret key $sk$. Unlike usual encryption schemes, it also outputs a third key known as the evaluation key $evk$, which depends on $L$. This key will assist with the additional evaluation functionality.

**HE.Enc**$(m, pk) = c$. An encryption algorithm that accepts a public key $pk$ and a single bit plaintext $m$, and then outputs a ciphertext $c$. By slight abuse of notation, we also allow $m$ to be a bit string, and assume the algorithm performs encryption bit-by-bit.

**HE.Dec**$(c, sk) = m$. A decryption algorithm that accepts a single ciphertext $c$ and secret key $sk$ and outputs a single bit plaintext $m$. Again, we allow multi-ciphertext inputs and assume decryption occurs bitwise.

**HE.Eval**$(C, (c_1, \ldots, c_n), evk) = c'$. An evaluation algorithm that accepts a Boolean circuit $C$ with $n$ input wires. It further accepts $n$ ciphertexts $(c_1, \ldots, c_n)$ and an evaluation key $evk$. It outputs a single new ciphertext $c'$.

Note that we can similarly define a symmetric key HE scheme, and it is straightforward to generalize to the evaluation of non-Boolean circuits. Then the encryption scheme HE should satisfy the usual properties of an encryption scheme, but should further satisfy the following homomorphic property. For some circuits $C$ which we call the *homomorphisms* of the scheme, we have the following commutative diagram.




Here, $\mathcal{M}$ is the space of valid plaintexts (i.e. all binary strings), and $\mathcal{C}$ is the space of valid ciphertexts. One can think of the data processing as occurring from top-to-bottom, and the encryption as occurring from left-to-right. Plainly, a party Alice can perform the computation of $C$ herself, or outsource the computation by sending an encrypted message to a third party Bob.

We further call a homomorphic encryption scheme *compact* if the complexity of **HE.Dec** is independent of the function being evaluated. This precludes trivial schemes in which Bob simply sends back a description of the circuit to be evaluated, and the true evaluation function is embedded into the decryption itself. Here and throughout, we will assume that all schemes are compact.

We call a homomorphic encryption scheme *leveled fully homomorphic* if the set of homomorphisms of the scheme is the set of all Boolean circuits up to some size

specified by the evaluation parameter $L$. We call such a scheme *fully homomorphic* if the set of homomorphisms is the set of all Boolean circuits, independent of $L$.

Typically, the ciphertexts in homomorphic encryption schemes experience an accumulation of noise that scales with the evaluated circuit depth. Eventually, this noise will prevent accurate decryption, and this motivates the definition of leveled fully homomorphic schemes. It is important to note that Alice's work may scale with the circuit she is evaluating, but this is realized implicitly as preprocessing in the key generation phase. A bootstrapping procedure introduced in [66] allows for the indefinite refreshing of noisy ciphertexts, but makes the stronger assumption of circular security.

Homomorphic encryption schemes often allow for some small probability of failure. To simplify the discussion, we assume that our schemes are perfectly correct, but note that we can extend all our arguments to the imperfect case with some extra notational baggage.

### 5.2 Quantum homomorphic encryption

In [72], the problem of extending homomorphic encryption to the *quantum* setting was considered. Quantum homomorphic encryption accomplishes a similar task to classical homomorphic encryption, but with some key differences. Formally, we can model QHE as three families of quantum channels acting on four Hilbert spaces: $\mathcal{K}$ the key space, $\mathcal{M}$ the plaintext space, $\mathcal{C}$ the ciphertext space, and $\mathcal{R}$ a reference system with fixed initial state used during evaluation.

The size of $\mathcal{M}$ is chosen to be a pre-specified input size $n$, the size of $\mathcal{K}$ is chosen as a polynomial function of the security parameter $\kappa$, and the size of $\mathcal{C}$ is chosen as a polynomial function of $\kappa$ and the evaluation parameter $L$. We consider the evaluation key as being appended to the encryption in the ciphertext space.

For notational simplicity, we define a symmetric key scheme. Formally, we have a family of schemes parametrized by an input size $n$, security parameter $\kappa$, and evaluation parameter $L$.

**QHE.Enc**: $D(\mathcal{K} \otimes \mathcal{M}) \longrightarrow D(\mathcal{K} \otimes \mathcal{C})$. An encryption isometry that accepts a classical secret key $sk$ and a quantum state $\rho_m$, and then outputs a quantum ciphertext $\rho_c$ which includes the appended pre-specified evaluation key $\rho_{evk}$.

**QHE.Dec**: $D(\mathcal{K} \otimes \mathcal{C}) \longrightarrow D(\mathcal{M})$. A decryption channel that accepts a classical secret key $sk$ and a ciphertext state $\rho_c$, and then outputs a plaintext quantum state $\rho_m$.

**QHE.Eval**$_C$ : $D(\mathcal{C} \otimes \mathcal{R}) \longrightarrow D(\mathcal{C})$. An evaluation channel for a unitary circuit $C$ with $n$ wires that accepts a ciphertext state $\rho_c$ (with evaluation key $\rho_{evk}$), and

a fixed state in reference system $\mathcal{R}$ of arbitrary dimension. It outputs another ciphertext state $\rho_{c'}$.

While the homomorphic property is defined analogously, there are a few subtleties to this reformulation of homomorphic encryption. Encryption and decryption are now performed all at once, rather than bit-by-bit. Also, we assume that any randomness used during encryption is included in the secret key, making that channel isometric. The evaluation key is in general a quantum state, and may be consumed during the evaluation process. Finally, the set of homomorphisms for the scheme are now *unitary* circuits. By the principle of deferred measurement, this does not limit the model's universality in evaluating functions.

In the classical setting, homomorphic evaluation is defined piece-by-piece for the gates comprising a circuit. For *universal* gate sets, such as {AND,OR} or {NAND}, homomorphic evaluation of these constituent gates can be built into (leveled) fully homomorphic encryption schemes. For quantum homomorphic encryption schemes, the set of homomorphisms is augmented by a richer set of constituent gates.

The definitions for compact, leveled fully homomorphic, and fully homomorphic encryption schemes carry over to the quantum setting. However, we will find it useful later to define a homomorphic encryption scheme that is intermediate between classical and quantum schemes. We call this a *reversible fully homomorphic encryption* (RFHE) scheme. This is defined as a quantum homomorphic encryption scheme with the additional stipulation that the inputs are classical bits, and the set of homomorphisms for the scheme are the set of all classical reversible, rather than quantum, circuits.

Certainly, a QFHE scheme yields an RFHE scheme. As we will see, one can place information-theoretic bounds on RFHE schemes, and so in turn, QFHE schemes.

## 5.3 Proposals for quantum homomorphic encryption

In this section, we briefly outline existing proposals for quantum homomorphic encryption with both computational and information-theoretic security guarantees. We will later offer an alternative heuristic for information-theoretically secure homomorphic encryption relating to quantum codes in Chapter 6.

### 5.3.1 Computationally secure proposals

Computationally secure QHE was first considered in [72], along with an appropriate generalization of CPA security. The authors proposed three QHE schemes. The first was a scheme that could homomorphically implement the set of Clifford circuits, and followed directly from the quantum one-time pad. The second was a

*quasicompact* scheme, with a decryption function that scaled quadratically in the number of $T$-gates of the circuit. The final scheme required an evaluation key size scaling superexponentially in the $T$-depth of the circuit, and so was restricted to efficiently evaluating circuits of constant $T$-depth.

More recently, [73] proposed a scheme built off of previous work on instantaneous nonlocal computation [74]. The scheme is centered around an evaluation key consisting of $T$-gadgets, which allows for the homomorphic evaluation of one $T$-gate per $T$-gadget. This provides a leveled quantum fully homomorphic encryption scheme for polynomial-sized circuits, as these are precisely the circuits for which the evaluation key can be generated efficiently. This was further extended to a verifiable scheme in [75]. Very recently, a scheme was proposed for performing leveled QFHE with a purely classical client using entirely different means [76]; this approach remains to be explored.

One common thread throughout all of these proposals is that each is built on a classical FHE scheme, and so inherits its underlying computational security. It is natural to ask if quantum mechanics might allow for an *information-theoretically* secure delegation of computation on encrypted information. To this end, we might be encouraged by the invention of universal blind computation [6], which allows delegated quantum computation that guarantees information-theoretically secure hiding of *both* the plaintext *and* the computation, at the expense of interaction between Alice and Bob.

#### 5.3.2 Information-theoretically secure proposals

There have been several works aimed towards homomorphic encryption with information-theoretic security guarantees. In [77], a homomorphic encryption scheme based on bosonic encodings was proposed. This scheme used a weaker version of information-theoretic security by bounding the information accessible by the adversary. This allowed them to realize a fully unitary group of homomorphisms, although these homomorphisms were not universal as the dimension of the group scaled *polynomially* in the input size.

Later, [12] proposed a homomorphic encryption scheme based off of randomized quantum codes and transversal gates. The homomorphisms for this scheme included all Clifford circuits augmented by a constant number of $T$-gates. The security guarantees for this scheme were stronger, providing exponential suppression on the trace distance between any two ciphertexts.

Very recently, [13] detailed a scheme using a similar methodology of randomized quantum codes and transversal gates in order to implement an enlarged class of IQP circuits homomorphically. Their scheme offers similarly strong information-theoretic

security guarantees.

Finally, [78] proposed a homomorphic encryption scheme with a limited class of operations and modest information-theoretic security claims, but which may be implemented on current optical technologies. We recommend [79] for a more complete summary of securely delegated quantum computing.

## 5.4 Limitations on information-theoretically secure quantum homomorphic encryption

We now elaborate on certain no-go theorems which limit the capacity of homomorphic encryption schemes to exhibit meaningful information-theoretic security. It is well-known that in the classical setting, perfect information-theoretically secure homomorphic encryption is impossible [80]. This follows from communication bounds established for perfectly secure single-server private information retrieval [81], and their relaxations [82].

The first restriction on QHE information-theoretic security was proven in [11]. There, they use a data localization argument via the no-programming theorem [83] to show the following.

**Theorem 5.1** (Yu, Perez-Delgado, Fitzsimons)**.** *Suppose there exists a QHE scheme implementing a set of homomorphism $\mathcal{S}$, with precisely zero mutual information between the plaintext and ciphertext. Then, the size of the evaluated ciphertext must be at least* $\log_2(|\mathcal{S}|)$ *qubits long.*

Using Stirling's approximation, when $S$ is the set of all classical reversible functions, this evaluated ciphertext must be of size at least

$$\log_2((2^n)!) = (n - \log_2(e))2^n + \mathcal{O}(n).$$

Thus, in the case of *perfect* information-theoretic security, any RFHE (and so any QFHE) scheme must be highly inefficient. It is important to note that this data localization technique bounds the efficiency of perfect ITS-QHE for *any* set of homomorphisms.

In spite of this limitation, we have seen several QHE schemes [12, 13] that implement large sets of homomorphisms with strong, but imperfect, information-theoretic security guarantees. As the information localization argument of [11] relies integrally on perfect information-theoretic security, it is natural to ask whether an $\epsilon$-relaxation of the ITS guarantee may allow for much larger sets of homomorphisms?

### 5.4.1 Quantum random access codes

In [21], we show that this $\epsilon$-relaxation does not afford much more delegated computationally power. This was observed concurrently in [13], which used generalizations

of single-server private information retrieval bounds to the quantum setting [71]. At the heart of all three of these arguments is an application of Nayak's bound [84], which places limitations on the compression of classical information into quantum information. We now elaborate on the proof in [21] and discuss some of its subtleties.

**Definition 5.2.** An $(n, m, p)$-*quantum random access code* is a mapping of $n$ classical bits into $m$ qubits, $[\![b \mapsto \rho_b]\!]$, along with a set of POVM's $\{M_i^0, M_i^1\}_{i=1}^n$ satisfying, for all $b \in \{0,1\}^n$ and $i \in [n]$,

$$Tr(M_i^{b_i} \rho_b) \geq p.$$

Informally, this is simply a compression of classical information into quantum information that allows for a local recovery with some probability of success. Nayak's bound then places a fundamental limitation on the recoverability of this compression [84].

**Theorem 5.3** (Nayak)**.** *Any $(n, m, p)$-quantum random access code must satisfy*

$$m \geq n(1 - H(p))$$

*where $H(\cdot)$ is the binary entropy function.*

We now have the required tools to prove the $\epsilon$-ITS QFHE no-go theorem. The essential idea is to extract a quantum random access code from an ITS-RFHE scheme, and then apply Theorem 5.3 to lower bound the communication complexity.

### 5.4.2 Proof of theorem

**Theorem 5.4.** *Suppose we have a QHE scheme that is also an RFHE scheme. Suppose further that, for some $\epsilon < 1$ and for any two ciphertexts $\rho, \rho'$, we have the ITS guarantee that*

$$\|\rho - \rho'\|_1 < \epsilon.$$

*Then the combined size of the (evaluated) ciphertext and secret key must grow exponentially in the input size.*

*Proof.* Consider a QHE scheme which is also an RFHE scheme. Fix a security parameter $\kappa$, evaluation parameter $L$, and input size $n$. For any $x \in \{0,1\}^n$ define the state

$$|\psi_{k,x}\rangle \langle\psi_{k,x}| := Tr_{\mathcal{K}} \left( \mathbf{QHE.Enc}((|k\rangle \langle k| \otimes |x\rangle \langle x|) \right),$$

the encryption of $x$ using secret key $k$, which is pure as the channel is isometric and acts identically on $\mathcal{K}$. Then we can define the state

$$|\psi_x\rangle := \frac{1}{\sqrt{2^{p(\kappa)}}} \sum_{k \in \{0,1\}^{p(\kappa)}} |k\rangle \otimes |\psi_{k,x}\rangle$$

where $p(\kappa)$ is some polynomial in the security parameter. Here, $|\psi_x\rangle$ represents the uniform superposition over all keys and their corresponding encryptions on a fixed input. It follows from our ITS guarantee that, for any $x, x' \in \{0,1\}^n$,

$$\left\| \frac{1}{2^{p(\kappa)}} \sum_{k \in \{0,1\}^{p(\kappa)}} |\psi_{k,x}\rangle \langle\psi_{k,x}| - |\psi_{k,x'}\rangle \langle\psi_{k,x'}| \right\|_1 < \epsilon.$$

Then there must be some unitary $V_{\mathcal{K}}$ acting nontrivially only on $\mathcal{K}$ with the property that

$$\left\| |\psi_x\rangle \langle\psi_x| - V_{\mathcal{K}} |\psi_{x'}\rangle \langle\psi_{x'}| V_{\mathcal{K}}^\dagger \right\|_1 < \epsilon.$$

Fix any base point $x' \in \{0,1\}^n$ and define, for any $x$, the unitary $V_{\mathcal{K}}^x$ that satisfies the above.

Consider the set of Boolean functions $f : \{0,1\}^n \to \{0,1\}$. For any such $f$, let $C_f$ be a classical reversible circuit that computes $f$, with the output of $C_f$ taken as the output of the first wire. Furthermore, identify $f$ with the binary string $(f(y_1), f(y_2), \ldots, f(y_{2^n})) \in \{0,1\}^{2^n}$ where $y_i$ is the $n$-bit binary representation for $i$. Let

$$\rho_f^x := \mathcal{I}_{\mathcal{K}} \otimes \mathbf{QHE.Eval}_{C_f} \left( |\psi_x\rangle \langle\psi_x| \otimes \sigma \right)$$

where $\sigma$ is the fixed state of the reference system, and define $\rho_f := \rho_f^{x'}$. Then we observe that the encoding

$$f \mapsto \rho_f$$

is a quantum random access code for $2^n$ bits.

Our queries to this quantum random access code are indexed by $x \in \{0,1\}^n$. Let $P_j = |j\rangle \langle j|$ on the first qubit. Then for any $x$, the associated POVM's in Definition 5.2 are given by

$$M_x^j = P_j \cdot \left( \mathbf{QHE.Dec}(V_{\mathcal{K}}^x \rho_f V_{\mathcal{K}}^{x\dagger}) \right).$$

Importantly, we are using the homomorphic property that the evaluation occurs only on $\mathcal{C} \otimes \mathcal{R}$, which ensures that it will commute with the key space unitary.

Then, by the assumption that the QHE scheme is a perfectly accurate RFHE scheme, our probability of failure $p_{\text{fail}}$ is bounded above by

$$\begin{aligned}
p_{\text{fail}} &= \max_{x,f} \; Tr\left( M_x^0 \left( \rho_f - \rho_f^x \right) \right) \\
&\leq \max_{x,f} \max_{P \leq I} \; Tr\left( P \left( \rho_f - \rho_f^x \right) \right) \\
&\leq \max_{x,f} \max_{P \leq I} \; Tr\left( P \left( |\psi_x\rangle \langle\psi_x| - V_{\mathcal{K}} |\psi_{x'}\rangle \langle\psi_{x'}| V_{\mathcal{K}}^\dagger \right) \right) \\
&< \epsilon/2.
\end{aligned}$$

Here, the third step follows from the contractivity of trace distance, and the final step follows from the functional definition of trace distance. We then have a $(2^n, |\mathcal{K}| + |\mathcal{C}|, 1 - \frac{\epsilon}{2})$-quantum random access code. Then by Theorem 5.3,

$$|\mathcal{K}| + |\mathcal{C}| \geq 2^n \left(1 - H(\epsilon/2)\right)$$

where $H(\cdot)$ is the binary entropy function. It follows that $|\mathcal{K}| + |\mathcal{C}| = \theta(2^n)$. Noting that the proof holds even when an evaluated ciphertext is larger than a fresh ciphertext, the result follows.

□

As any QFHE scheme is also an RFHE scheme, we obtain the following.

**Corollary 5.5.** *Any QFHE scheme with nontrivial $\epsilon$-information theoretic security must be inefficient.*

### 5.5 Conclusion and no-go workarounds

One thing to note is that, in our definition of QHE, we enforce that the fresh ciphertexts and evaluated ciphertexts are of the same size. This is simply for notational convenience, and so the proof carries through even when the evaluated ciphertext grows in size.

Although Theorem 5.4 represents an $\epsilon$-relaxation of Theorem 5.1, it is restricted in the sense that it only rules out *fully* homomorphic encryption from reversible circuits. While of course this rules out QFHE as well, it does not immediately apply to $\epsilon$-ITS-QHE schemes with *particular* sets of homomorphisms $\mathcal{S}$, as the perfect ITS-QHE limitation does. Intuitively, one would expect to be able to more generally bound the number of distinct *unitaries* in $\mathcal{S}$. However, we are at least superficially constrained by Nayak's bound, which applies specifically to these classical to quantum encodings.

In fact, one may not even be able to consider sets of homomorphisms $\mathcal{S}$ that are properly contained in the set of all Boolean functions realized from reversible circuits. For such a set $S$ of Boolean functions, even if $\log(|S|)$ is super-polynomial, it is conceivable that some symmetry of the set $S$ may allow for a more efficient random access structure.

Another question is whether one can rule out a leveled ITS-QFHE scheme. Suppose we fix a degree $d$ polynomial $p$ and only consider circuit families $\{C_n\}$ with the number of gates in $C_n$ bounded by $p(n)$ for all input sizes $n$. If we directly consider $n$-bit Boolean circuits generated by the universal 2-bit NAND gate, then the total number of unique such circuits is at most $\binom{n}{2}^{p(n)} = n^{\theta(n^d)}$.

For this reason, it seems fundamentally difficult to rule out leveled QFHE for polynomial sized circuits using similar methods, simply because there aren't that

many circuits asymptotically. Nonetheless, it seems implausible that such a scheme would exist as the evaluation key depends only on the length of the circuits being evaluated, and not on the circuits themselves. This is an important possibility to rule out as there are currently *no* non-leveled QFHE proposals, even with computational security.

There are two main workarounds to this no-go theorem. We could limit the number of homomorphisms for the scheme, and we've seen this restriction employed fruitfully to augmented IQP [13] and Clifford circuits [12]. It would be interesting to develop a more general framework for implementing limited circuit classes homomorphically, similar in size to the Clifford group on $n$ qubits which grows as $\theta(2^{n^2})$. The other workaround is to lessen the stringency of the security. We've seen that, when using a weaker security guarantee in terms of accessible information, we can implement a full *continuum* of homomorphisms [77]. Could something similar be made universal?

# CHAPTER 6

# Restrictions on transversal gates

Transversal gates are surprisingly ubiquitous objects, finding applications in quantum cryptography [12], [13], quantum complexity theory [85], and of course quantum fault-tolerance. Although the instability of quantum information is well-documented, we have seen that quantum error-correcting codes [36] allow us a way to preserve quantum data. However, performing computations on these codes carries the risk of propagating errors between different subsystems, unless the code can implement the computation in a way that preserves the subsystem structure. Informally, these types of logical operators that decompose as a product across the subsystems are called *transversal* (see section 3.7.1), and the oft-cited Eastin-Knill theorem [86], [14] limits the ability of quantum codes to prevent this error propagation.

**Theorem 6.1** (Eastin-Knill)**.** *No quantum error-correcting code can implement a quantum universal transversal gate set.*

These transversal gate sets are valuable as most models of fault-tolerant quantum computation implement associated transversal gate sets fault-tolerantly "for free". Incurring comparatively significant overhead, often in the form of magic state distillation [48], [87], gauge fixing [45], [88], or more recently deconstructions of non-transversal gates into fault-tolerant pieces [89], one can fault-tolerantly implement some remaining gate set making the computation space universal. Improving the efficiency of this overhead and designing new fault tolerant architectures to supplement transversal gates is central to quantum fault tolerance.

Implementing fault-tolerant classical reversible computation efficiently would be extremely desirable as many quantum algorithms are primarily classical subroutines with a relatively small number of quantum gates, and there have been several proposals for doing so [90], [15], [91]. For example, factoring a cryptographically large RSA key using Shor's algorithm requires around $3 \times 10^{11}$ Toffoli gates to perform modular exponentiation alone, and is the dominating portion of the circuit [63]. As Toffoli is universal for classical reversible computation, one might ask if there are

any quantum error-correcting codes that can naturally implement Toffoli, and thus classical computations, transversally? We give restrictions on the ability of QECCs to do this.

**Theorem 6.2** (Informal)**.** *Almost no quantum error-correcting code can implement a classical universal transversal gate set. In particular, almost no quantum error-correcting code can implement the Toffoli gate transversally.*

The only exceptions to our theorem are non-additive distance $d$ codes that decompose as $d$-fold product states in their logical computational basis, where each "subcode" itself fails to be erasure-correcting. Essentially, one can think of these as maximally redundant quantum codes: they are the concatenation of a repetition code with some distance 1 inner code, similar to Shor's stabilizer code written as a 3-fold product of $GHZ$ states. We do not expect that any such code can implement Toffoli transversally, but it remains a case our proof technique cannot rule out. In particular, our proof does apply to all binary additive codes. The result is perhaps slightly surprising since there exist QECCs (e.g. triorthogonal codes) that can implement the $CCZ$ gate transversally [90], and in fact transversal Toffoli gates can map between different quantum Reed-Solomon codes by increasing the degree of the underlying polynomial [15] (see section 6.5.1).

## 6.1 Summary of previous results

The five works the most closely resemble our results are [86],[14] and [16], which place restrictions on transversal gate sets for QECCs, and [12] and [13], which use similar ITS-QHE constructions. We very roughly summarize these results and compare them to our own.

In [14], Zeng *et al.* were some of the first to place restrictions on quantum universal transversal gate sets for *additive* quantum codes by elucidating the stabilizer group structure. Further work in [92] classified the set of diagonal gates that can implement one and two qubit logical operations in stabilizer codes. Shortly thereafter, [86] showed that for *any* QECC, the transversal gate set must be finite, and so cannot approximate with arbitrary precision the full unitary group. Intuitively, they make a Lie type argument by showing that infinitesimal transversal operations are themselves linear combinations of local error operators. Since these unitaries must act identically on the codespace, it follows that the group of transversal operations must be finite.

More recently, [16] placed restrictions on the more general class of topologically protected logical gates in topological stabilizer codes, which include transversal gates as an optimal subset. They showed that for a topological stabilizer code defined on a

$d$-dimensional lattice, any such gate must lie in the $d$th level of the Clifford hierarchy. These results were extended in [93] to more general stabilizer subsystem codes, and we will detail how similar arguments can be used to rule out classical reversible transversal computation for the subclass of stabilizer codes.

Our strategy will be to construct an ITS-QHE scheme similar to [12]. Because of the stringent lower bounds placed by Nayak, we actually forgo the noisy encoding circuit and embed QECCs directly into random noise after removing a correctable set of qubits. This has the effect of increasing the overhead by an exponential factor in order to achieve security, but thanks to the roomy lower bound, this factor is still too small to allow an ITS-QFHE scheme.

We can argue directly about the security of this scheme using the nonlocality of the quantum information being encoded in almost any QECC. The idea is conceptually simple: in order to obtain encryptions of the data that are both secure and (sufficiently) short, we must inject randomness into the encodings themselves by withholding qubits from the code. While ordinarily this would negatively affect the correctness of homomorphic evaluation, the error-correcting property allows us to inject this randomness while still maintaining perfect recoverability. Then intuitively, spreading the information across the subsystems limits the complexity of the class of logical operators that don't couple the subsystems, i.e. the *transversal* operators. This differs fundamentally from the approaches in [14] and [86] in that it is a quantitative information-type bound.

It is not without its drawbacks however, as these maximally redundant codes fail to "spread out" the information sufficiently. The prototypical example is Shor's code, which is the concatenation of a bit-flip and phase-flip code. However, we can argue directly using the stabilizer group structure that no such *additive* code can implement Toffoli transversally.

### 6.2 Homomorphic encryption from quantum codes

Without loss of generality, we use a slightly simplified model of *transversal gates* $\mathcal{T}_C$ associated to $C$; they are those logical gates that decompose as a product across the subsystems. That is to say, $U_L \in \mathcal{T}_C$ if $U_L = U_1 \otimes \ldots \otimes U_n$, where $n$ is the length of the code, each $U_i$ acts on a single subsystem, and $U_L$ is a codespace preserving map on the code $C^{\otimes r}$ for $U$ an $r$-qubit gate. This is similar to our previous definition, but for ease of presentation, assumes that any element of a partition contains exactly one qubit from each code block. We further define a logical gate to be *strongly transversal* if it decomposes as $U_L = U^{\otimes n}$. Following the example of [14], we do not allow coordinate permutations in our definition of transversality.

Here and throughout, we will refer to codes with distance at least two as *error-detecting* and refer to stabilizer codes as additive codes. Although we restrict to qubits, identical results should hold for qudits of any dimension by considering generalizations of Nayak's bound. We will apply our argument to any logically encoded qubit within a code block, and so without loss of generality assume $k = 1$.

**Definition 6.3.** We say a quantum code $C = \mathrm{Span}_{\mathbb{C}}(\left|\tilde{0}\right\rangle, \left|\tilde{1}\right\rangle)$ is an *r-fold code* if it can be written as

$$\left|i\right\rangle_L = \bigotimes_{j=1}^{r} \left|\psi_{ij}\right\rangle.$$

where each vector $\left|\psi_{ij}\right\rangle$ does not further decompose as a product state across any bipartition. We additionally assume that $r \leq d$, that $\left|\psi_{0j}\right\rangle$ and $\left|\psi_{1j}\right\rangle$ occupy the same subsystem, and that $\left|\psi_{0j}\right\rangle \perp \left|\psi_{1j}\right\rangle$. It then makes sense to refer to $\mathrm{Span}\{\left|\psi_{0j}\right\rangle, \left|\psi_{1j}\right\rangle\}$ as the *jth subcode*. These assumptions are natural, and we justify them in our discussion.

If the code is additionally an $[[n, 1, d]]$ QECC with $r = d \geq 2$ and each subcode has distance 1, we simply call the resulting code a *maximally redundant code*. Note that any (pure state) code is at least a 1-fold code.

The guiding example is Shor's code, which can be seen as the concatenation of a repetition outer code and a complementary $GHZ$ inner code, neither of which is quantum erasure correcting. In the case that the subcodes are identical, any maximally redundant code is just the concatenation of a repetition code with some distance 1 subcode. Intuitively, these are codes for which you can't erase enough qubits to mix the state while still remaining perfectly correctable: while redundancy can be used in classical error-correcting codes to protect information, quantum error-correcting codes must "spread out" information to protect it. In this sense, these codes are maximally redundant because they "spread out" the information the least. We show that non-additive maximally redundant codes (i.e. maximally redundant codes for which the subcodes are comprised of non-stabilizer subspaces) are the only binary QECCs with the hope of implementing logical Toffoli transversally.

We now consider a strategy for implementing compact QHE using quantum codes. This will be a simple "block" embedding encryption scheme homomorphically implementing *quantum* circuits on *classical* input, and is similar to the construction in [12]. We will use the error-correcting property to withhold a correctable set of qubits from the encoding.

The scheme is detailed in Figures 6.1 and 6.2. Using that notation to summarize, our encryption channel $\mathcal{E}$ is defined, for secret key $S$ and input string $\vec{i}$, as $\mathcal{E}(S, \vec{i}) =$

**Coding QHE Scheme:**

*Arguments:*

$$\begin{array}{rcl} C & = & \text{an } [[n+r,1,d]] \text{ } r\text{-fold QECC with } r < d \\ \vec{i} & \in & \{0,1\}^p \\ m & = & \text{the size of each noise code block} \\ S & \in & [m]^n, \text{ the secret key} \end{array}$$

1. On input $\vec{i} \in \{0,1\}^p$, encode $\vec{i}$ as the pure state $\bigotimes_{\ell=1}^{p} |i_\ell\rangle_L$, for $\{|0\rangle_L, |1\rangle_L\}$ the logical computational basis defining $C$.

2. Let $R$ be a collection of $r$ subsystems, each of $p$-qubits, comprised of one subsystem from each subcode. Then form $\gamma^{\vec{i}} = Tr_R(\bigotimes_\ell |i_\ell\rangle_L)$. Essentially, $\gamma^{\vec{i}}$ is the state of the collection of codewords with each codeword missing one subsystem from each of its subcodes.

3. Initialize $n$ $(p \times m)$ arrays of maximally mixed qubits, and replace the $S_j$-th column of each array with the $j$-th subsystem of $\gamma^{\vec{i}}$. This forms the encrypted state.

4. Publish a constant number of labeled encryptions of 0 and 1, to be used as ancilla in homomorphic evaluation.

Figure 6.1: A description of the encryption procedure for the code based QHE scheme.

$\gamma_S^{\vec{i}}$. We sometimes use the notation $\gamma_S$ instead of $\gamma_S^{\vec{i}}$ or $\gamma$ instead of $\gamma^{\vec{i}}$, omitting $\vec{i}$ when we are unconcerned with the underlying plaintext.

The total size of our encrypted input is $mnp$ qubits. In our preceding notation, the described scheme has parameters $(p, mnp, mnp, \epsilon(m,p), 0)$, implementing the set of gates $\mathcal{T}_C$ homomorphically.

**Lemma 6.4.** *Let $\mathcal{E}$ be the encryption scheme detailed in Figure 6.1. Let $\mathcal{T}_C$ denote the group of transversal operators associated to the underlying quantum code $C$. Then, $\mathcal{E}$ is $\mathcal{T}_C$-homomorphic.*

*Proof.* Let $U_L$ be the logical operator we wish to apply to some codestate $|\psi\rangle_L$. By definition, $U_L \in \mathcal{T}_C$ implies $U_L$ can be decomposed as a product operator $U_1 \otimes \ldots \otimes U_{n+r}$ where $U_i$ is an operator that acts only on the $i$-th subsystem of the code. Then, without knowledge of the secret key $S$, a third party can implement $U_L$ by applying the operator

$$\bigotimes_{i=1}^{n} \bigotimes_{j=1}^{m} U_i$$

where each $U_i$ is an operator local to some subsystem in Server's possession (that is to say, on one of the columns in the corresponding array). Returning the resulting

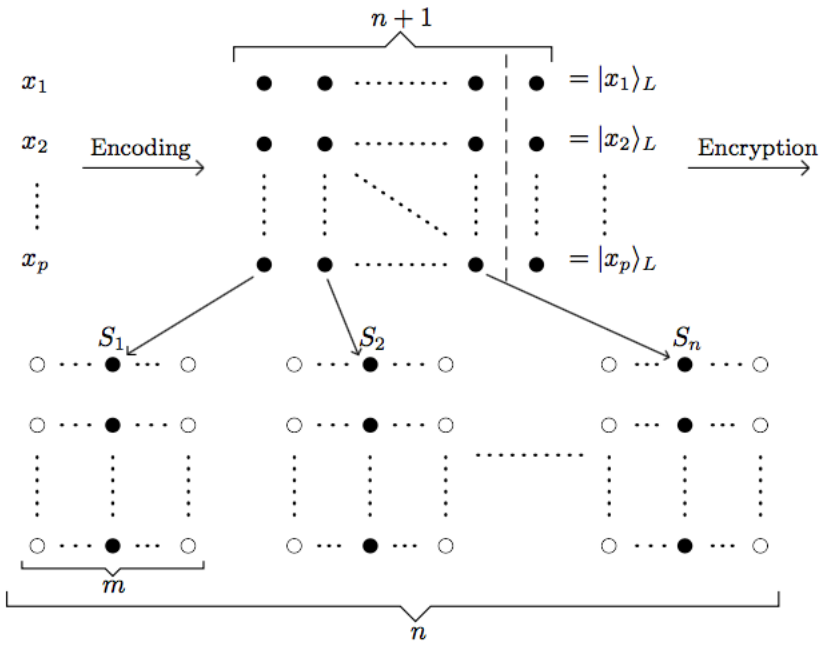



Figure 6.2: A diagram illustrating the code-based QHE scheme for an $(n+1)$-length 1-fold quantum code while withholding a single subsystem. The $(n+1)$-th subsystem remains in the hands of Client. The arrows connecting the subsystems indicate where each subsystem (i.e. column) is being mapped. The filled dots represent code qubits, while the empty dots represent maximally mixed qubits.

data to a party with the secret key, that party can decrypt to obtain a state of the form $V^R U_L |\psi\rangle_L$, where $V^R$ is supported on the $r$ subsystems that Client has withheld. Since $r < d$, viewing $V^R$ as an erasure error on $r$ subsystems, there exists some recovery channel $\mathcal{R}$ such that $\mathcal{R}(V^R U_L |\psi_L\rangle) = U_L |\psi\rangle_L$. Decoding, we obtain $U|\psi\rangle$ as desired.

□

Note that this scheme is a $\mathcal{T}_C$-homomorphic, non-leveled, and compact QHE scheme, since the recovery and decryption channel do not depend on the complexity of $U$.

We now aim to compute the security $\epsilon(m, p)$ of the proposed scheme, namely the tradeoff between the size of the input $p$, the size of the encoding $mnp$, and the ITS guarantee. To avoid confusion, we point out here that the code size $n$ is a constant, as we are not concatenating to achieve security, just amplifying the size of the noise into which we are embedding.

We want to show that while the scheme is inefficient, its parameters still defeat Nayak's bound. To simplify the security proof, we impose the requirement that the outputs are indistinguishable from uniformly random noise. Here, we will see that the nonlocality of the information stored in QECCs is essential in its allowing us

to withhold qubits while still delegating computation to Server. This imposes the requirement of using quantum error-correcting codes, as evidenced by the following observation.

**Lemma 6.5.** *Suppose we replace the preceding scheme with one that does not withhold any of the physical qubits comprising the (pure state) code. Then if $m = o(2^p)$, $\epsilon$ must be bounded away from zero.*

*Proof.* Counting the rank of the encrypted state, note that $\text{rank}(\gamma_S) = 2^{np(m-1)}$. Then,

$$\begin{aligned}\text{rank}(E_S[\gamma_S]) &\leq m^n 2^{np(m-1)} \\ &\leq 2^{n(p(m-1)+\log(m))}.\end{aligned}$$

Thus, the fraction of nonzero eigenvalues must be at most $(2^n)^{\log(m)-p}$. Since $\log(m) = o(p)$, the fraction of nonzero eigenvalues goes to zero, and so $\|E_S[\gamma_S] - I/2^{mnp}\|_1$ must be bounded away from zero as claimed. □

### 6.3 Security proof

Our aim is to give (inefficient, but sufficient) security parameters for the coding QHE scheme. We will then argue that if there were a QECC implementing a sufficiently large transversal gate set (such as the set of all classical reversible gates), then it would violate Nayak's bound with these parameters. We will first need a small lemma on the structure of the partial trace operator.

**Lemma 6.6.** *For Hilbert space decomposition $\mathcal{H} = \mathcal{H}_{\bar{\Delta}_1} \otimes \mathcal{H}_{\Delta} \otimes \mathcal{H}_{\bar{\Delta}_2}$,*

$$Tr\left((\rho^{\bar{\Delta}_1 \Delta} \otimes I^{\bar{\Delta}_2})(I^{\bar{\Delta}_1} \otimes \sigma^{\Delta \bar{\Delta}_2})\right) = Tr\left(Tr_{\bar{\Delta}_1}(\rho) Tr_{\bar{\Delta}_2}(\sigma)\right).$$

*Proof.* Expanding in terms of outer products,

$$\begin{aligned}Tr\left((\rho \otimes I)(I \otimes \sigma)\right) &= Tr\Bigg(\Big(\sum_{i,i'}\sum_{j,j'}\sum_{k} a_{i,i',j,j'} \left|i\right\rangle \left\langle i\right|^{\bar{\Delta}_1} \otimes \left|j\right\rangle \left\langle j'\right|^{\Delta} \otimes \left|k\right\rangle \left\langle k\right|^{\bar{\Delta}_2}\Big) \cdot \\ &\qquad \Big(\sum_{\ell}\sum_{m,m'}\sum_{n,n'} b_{m,m',n,n'} \left|\ell\right\rangle \left\langle \ell\right|^{\bar{\Delta}_1} \otimes \left|m\right\rangle \left\langle m'\right|^{\Delta} \otimes \left|n\right\rangle \left\langle n'\right|^{\bar{\Delta}_2}\Big)\Bigg) \\ &= Tr\left(\sum_{i,i'}\sum_{n,n'}\sum_{j,m'}\left(\sum_{j'} a_{i,i',j,j'} b_{j',m',n,n'}\right) \left|i\right\rangle \left\langle i\right| \otimes \left|j\right\rangle \left\langle m'\right| \otimes \left|n\right\rangle \left\langle n'\right|\right) \\ &= \sum_{i}\sum_{n}\sum_{j,j'} \left(a_{i,i,j,j'} b_{j',j,n,n}\right).\end{aligned}$$

On the other hand, we have

$$\begin{aligned} Tr\left(Tr_{\bar{\Delta}_1}(\rho)Tr_{\bar{\Delta}_2}(\sigma)\right) &= Tr\left(\Big(\sum_i \sum_{j,j'} a_{i,i,j,j'} \left|j\right\rangle \left\langle j'\right| \Big)\Big(\sum_n \sum_{m,m'} b_{m,m',n,n} \left|m\right\rangle \left\langle m'\right| \Big)\right) \\ &= Tr\left(\sum_i \sum_n \sum_{j,j'} \Big(\sum_{j'} a_{i,i,j,j'} b_{j',m',n,n}\Big) \left|j\right\rangle \left\langle m'\right| \right) \\ &= \sum_i \sum_n \sum_{j,j'} \left(a_{i,i,j,j'} b_{j',j,n,n}\right) \end{aligned}$$

as claimed. □

With this we are ready to prove the security tradeoff between $\epsilon, p$, and $m$. We adopt the same notation used in the proposed scheme for convenience, and note that we are demanding the stronger condition that outputs are indistinguishable from random noise.

**Proposition 6.7.** *For the scheme described in Figure 6.1, letting $K = 2^p$ be the dimension of any subsystem and for some $c \in (0,1)$, we have*

$$\| \left(I/K^{mn}\right) - E_S[\gamma_S] \|_1 \leq \epsilon(K,m)$$

*for* $\epsilon(K,m) = \left(\left(\frac{m-1}{m}\right)^n - 1 + K^{-c}\left(\frac{2K}{m}\right)^n\right)^{1/2}$.

*Proof.* By Cauchy-Schwartz,

$$\begin{aligned} \| \left(I/K^{mn}\right) - E_S[\gamma_S] \|_1^2 &\leq K^{mn} \| \left(I/K^{mn}\right) - E_S[\gamma_S] \|_2^2 \\ &\leq K^{mn} Tr(E_S[\gamma_S]^2) - \left(\frac{2}{K^{mn}}\right) Tr(E_S[\gamma_S]) + \left(\frac{1}{K^{2(mn)}}\right) Tr(I) \\ &\leq K^{mn} Tr(E_S[\gamma_S]^2) - 1. \end{aligned}$$

where the third line follows by noting that, as a quantum state, $Tr(E_S[\gamma_S]) = 1$. We write $|S \cap S'|$ to denote the size of the intersection of $S$ and $S'$ considered as sets. We can then decompose, for $p_\ell = \Pr_{S,S'}[|S \cap S'| = \ell]$,

$$\begin{aligned} K^{mn} Tr(E_S[\gamma_S]^2) &= \left(\frac{K^{mn}}{m^{2n}}\right) \sum_{S,S'} Tr(\gamma_S \gamma'_S) \\ (*) \ &= K^{mn} \sum_{\ell=0}^{n} p_\ell Tr\left(E[(\gamma_S \gamma_{S'}) \,\big|\, |S \cap S'| = \ell]\right). \end{aligned}$$

Note that $p_\ell = \frac{\binom{n}{\ell}(m-1)^{(n-\ell)}}{m^n} \leq \binom{n}{l}/m^\ell$ and that $p_0 = (\frac{m-1}{m})^n$. Furthermore, up to a

permutation on the coordinates, we may write for $\dim(I) = K^{mn-2n}$,

$$\begin{aligned} K^{mn}E[(\gamma_S\gamma_{S'}) \,\big|\, |S\cap S'| = 0] &= K^{mn}Tr\left((\gamma/K^n)\otimes(\gamma/K^n)\otimes\left(I/K^{(mn-2n)}\right)^2\right) \\ &= 1 \end{aligned}$$

again by noting that $\gamma$ is a quantum state of trace one and by multiplicativity of trace over tensor products. Next consider the general case $|S\cap S'| = \ell$. Then up to a permutation on the coordinates and for some $\pi \in S_n$, for $\Delta$ the subsystem of the intersection $S\cap S'$,

$$\begin{aligned} K^{mn}Tr(\gamma_S\gamma_{S'}) &= K^{mn}Tr\left((I/K^{n-\ell}\otimes\gamma)(\pi\gamma\pi^\dagger\otimes I/K^{n-\ell})\otimes\left(I/K^{(mn-2n+\ell)}\right)^2\right) \\ &= K^{\ell}Tr\left((I\otimes\gamma)(\pi\gamma\pi^\dagger\otimes I)\right) \\ &= K^{\ell}Tr\left(Tr_{\bar{\Delta}}(\gamma)Tr_{\bar{\Delta}}(\pi\gamma\pi^\dagger)\right) \end{aligned}$$

where the final line follows from Lemma 6.6. Then, because we have withheld a subsystem from each subcode of the underlying QECC, in any row $i$ we have that $Tr_{\bar{\Delta}}(\gamma^i)$ is mixed. It follows that $Tr\left(Tr_{\bar{\Delta}}(\gamma^i)Tr_{\bar{\Delta}}(\pi\gamma^i\pi^\dagger)\right) < 1$. So by separability across each encoded qubit and again by multiplicativity of trace across tensor products,

$$Tr\left(\bigotimes_{j=1}^{p} Tr_{\bar{\Delta}}(\gamma^{i_j})Tr_{\bar{\Delta}}(\pi\gamma^{i_j}\pi^\dagger)\right) = \prod_{j=1}^{p} Tr\left(Tr_{\bar{\Delta}}(\gamma^{i_j})Tr_{\bar{\Delta}}(\pi\gamma^{i_j}\pi^\dagger)\right).$$

It follows that there exists some $c \in (0,1)$ so that

$$K^{mn}Tr(\gamma_S\gamma_{S'}) \leq K^{\ell-c}.$$

Putting this all together, we observe that

$$\begin{aligned} K^{mn}\sum_{\ell=1}^{n} p_\ell Tr\left(E[(\gamma_S\gamma_{S'}) \,\big|\, |S\cap S'| = \ell]\right) &\leq K^{-c}\sum_{\ell=1}^{n}\binom{n}{\ell}\left(\frac{K}{m}\right)^\ell \\ &\leq K^{-c}\left(\left(1+\frac{K}{m}\right)^n - 1\right) \\ &\leq K^{-c}\left(\frac{2K}{m}\right)^n \end{aligned}$$

Including the first term in the sum, we get,

$$(*) \leq \left(\frac{m-1}{m}\right)^n + K^{-c}\left(\frac{2K}{m}\right)^n$$

and so,

$$\epsilon(K,m) = \left(\left(\frac{m-1}{m}\right)^n - 1 + K^{-c}\left(\frac{2K}{m}\right)^n\right)^{1/2}$$

as desired.

□

### 6.4 Almost no classical-universal transversal gate sets

We are left with two competing bounds. On the one hand, it follows from Nayak's bound that, for any $\mathcal{F}$-ITS-QHE encryption scheme with security $\epsilon$ and communication size $s$,

$$s \geq \log(|\mathcal{F}|)(1 - H(\epsilon)).$$

If we choose parameters that do not leak some constant fraction of information about our input, then as $\epsilon \to 0$ we see that for $s$ chosen as some fixed function of the input size, it must be that $s = \Omega(\log(|\mathcal{F}|))$. Using the notation and parameters from the aforementioned coding scheme, this means that $mnp = \Omega(\log(|\mathcal{F}_p|))$ for $\mathcal{F}_p$ the restriction of functions in $\mathcal{F}$ to $p$-bit inputs. Note that we can assume no ancilla overhead since the constant gets absorbed into this asymptotic bound.

Now by construction of the scheme, $\mathcal{F}$ is the transversal gate set for the underlying choice of quantum error-correcting code. Next, we would like to choose $m$ as a function of $K$ so that $\epsilon \to 0$. For this, it suffices to choose $m$ as a function of $K$ so that

$$\lim_{K\to\infty} K^{-c}\left(\frac{2K}{m}\right)^n = 0.$$

Equivalently, we require $m = \omega(K^{1-\left(\frac{c}{n}\right)})$. Then for some $c' < 1$, we can select $m = K^{c'}$ and still have $\epsilon \to 0$. Plugging this back into Nayak's bound, we see that asymptotically

$$K^{c'}\log(K) = \Omega(\log(|\mathcal{F}_p|))$$

for $|\mathcal{F}_p|$ the size of the function class, seen itself as a function returning the number of unique members in the class on $p$-bit inputs. In particular, $\mathcal{F}_p$ cannot be the set of all Boolean functions, for then $\log(|\mathcal{F}_p|) = K$. This shows that no code satisfying the hypotheses of our scheme can implement Toffoli transversally.

We now justify our earlier assumptions on the structure of candidate $r$-fold codes. Suppose an $r$-fold $[[n, 1, d]]$ QECC could implement a logical Toffoli gate transversally. First note that the tensor decomposition between the logical states must align, or else the restriction of logical Toffoli to one element of the product would unitarily map a pure state to a mixed state. Furthermore, we can decompose of the

Knill-Laflamme error-correction criterion in section 3.2 as a diagonal and off-diagonal condition: for all $|E| < d$,

$$\langle 0_L|E|0_L\rangle = \langle 1_L|E|1_L\rangle,$$
$$\langle 0_L|E|1_L\rangle = 0.$$

Since the Paulis form an operator basis, we can always assume that $E$ is an element of the Pauli group. Then, for $r$-fold codes with logical basis states $|i\rangle_L = \bigotimes_{j=1}^{r} |\psi_{ij}\rangle$, this becomes

$$\prod_{k=1}^{r} \langle \psi_{ik}|E_k|\psi_{jk}\rangle = c_E \delta_{ij}$$

where $E = E_1 \otimes \ldots \otimes E_r$. Note then that if $|\psi_{0j}\rangle \not\perp |\psi_{1j}\rangle$, we can trace out the corresponding subsystem and obtain a code with the same correctable error set on the complement of that system. Furthermore, if $r > d$, then we can again trace out any $r - d$ subcode subsystems to obtain a code with the same correctable error set on the complement. Both of these observations follow from noticing that these subcodes must themselves satisfy the diagonal condition,

$$\langle \psi_{0j}|E|\psi_{0j}\rangle = \langle \psi_{1j}|E|\psi_{1j}\rangle.$$

It follows from the security proof that if $r < d$, then the code would satisfy the hypotheses of our scheme and violate the lower bound in Theorem 5.4. Thus, $r = d$. Furthermore, logical transversal Toffoli on the entire code must restrict (up to global phase) to a logical transversal Toffoli gate on the subcodes, each of which is 1-fold by definition. Thus, each subcode must not be error-detecting. To summarize,

**Theorem 6.8.** *If a quantum error-detecting code is not a maximally redundant code, then it does not admit a classical-reversible universal transversal gate set. In particular, no such code can implement the Toffoli gate transversally.*

Note also that for the scheme in Figure 6.1, for any $m = \omega(K^{1-(\frac{c}{n})})$, $\epsilon(K)$ is negligible in $p$. Summarizing the parameters of the coding scheme:

**Proposition 6.9.** *For any $r$-fold $[[n, 1, d]]$ quantum error-detecting code $C$ with $r < d$ and with transversal gate set $\mathcal{T}_C$, the described protocol is a compact quantum $\mathcal{T}_C$-homomorphic encryption scheme with security $\epsilon = negl(p)$ for $p$ the input size and with encoding size $m = 2^{pc'}$ for some $c' < 1$.*

While this is highly inefficient, we pause to give some intuition for why it suits our purposes. On the one hand, we can envision trivial "hiding" schemes that have encoding length $2^p$ in each bit. Nayak's bound allows for higher efficiency, roughly

demanding that encodings implementing the set of all classical functions on $p$ bits homomorphically must have length at least $(2^p/p)$ in each bit. Finally our scheme, with encoding length $2^{pc'}$ for some $c' \in (0,1)$, is just efficient enough to defeat this bound and allow us to argue Theorem 6.8.

Finally, note that by concatenating an $[[n,1,d]]$ $d$-fold code with itself, the code remains $d$-fold while the distance must increase to at least $d^2$. Furthermore, if such a code implements Toffoli strongly transversally, then so does its concatenation with itself. As a result, we observe the following.

**Corollary 6.10.** *No quantum error-detecting code can implement strongly transversal Toffoli.*

#### 6.4.1 Stabilizer code case

Because these maximally redundant codes have a simple design, if we further assume that they are additive, we can use the additional stabilizer structure to argue directly that they cannot implement logical Toffoli transversally. From this observation, we directly obtain the following.

**Corollary 6.11.** *No additive quantum error-detecting can implement transversal Toffoli.*

*Proof.* By Theorem 6.8, it suffices to consider maximally redundant codes. So suppose, for the sake of contradiction, that an $[[n,1,d]]$ additive $d$-fold code could implement Toffoli transversally. Let $[\cdot,\cdot]$ denote the group commutator. We denote by $\bar{\cdot}$ states and operations acting on the subcodes, and $\tilde{\cdot}$ those on the full code. We will assume that each subcode is the same, e.g. $\left|\tilde{i}\right\rangle = |\bar{i}\rangle^{\otimes d}$, so that we can speak directly about the inner and outer codes. The general argument follows similarly.

Since the code is additive, the code distance is the minimal weight logical Pauli operator acting on the code. For any $\bar{Z}_L$, by multiplicativity of the inner product over tensor products,

$$\begin{aligned}\frac{1}{2}\left\langle |\tilde{0}\rangle + |\tilde{1}\rangle \left|\bar{Z}_L\right| |\tilde{0}\rangle - |\tilde{1}\rangle\right\rangle &= \frac{1}{2}\left(\langle\tilde{0}|\bar{Z}_L|\tilde{0}\rangle - \langle\tilde{0}|\bar{Z}_L|\tilde{1}\rangle + \langle\tilde{0}|\bar{Z}_L|\tilde{0}\rangle - \langle\tilde{1}|\bar{Z}_L|\tilde{1}\rangle\right) \\ &= \frac{1}{2}\left(\langle\bar{0}|\bar{0}\rangle^{\frac{n}{d}} + \langle\bar{1}|\bar{1}\rangle^{\frac{n}{d}}\right) \neq 0.\end{aligned}$$

Since the outer code has distance $d$, it follows from the QECC criterion that $\bar{Z}_L$ must have weight at least $d$. Then $\bar{X}_L$ must have weight 1, since the underlying inner code has distance 1 by assumption. Because the outer classical repetition code factors as a tensor product, transversal $\widetilde{\text{Toff}}_L$ on the outer code must restrict (up to a global phase) to transversal $\overline{\text{Toff}}_L$ on the inner code. Since we're now working

with multiqubit gates, let $G_L(i)$ denote the logical gate for $G$ acting on the $i$th code block. We can compute directly,

$$[\overline{\text{Toff}}_L(1,2,3), \bar{X}_L(1)] = \overline{CX}_L(2,3).$$

Furthermore, because $\overline{\text{Toff}}_L$ and $\bar{X}_L$ are transversal, it follows that $\overline{CX}_L$ has a representative that is also transversal and is supported on the subsystems that support $\bar{X}_L$. By a similar argument

$$[\overline{CX}_L(1,2), \bar{Z}_L(1)] = \bar{Z}_L(2)$$

so that $\bar{Z}_L$ must also be contained in the subsystems supporting $\overline{CX}_L$, and in turn $\bar{X}_L$. As we have already observed, the minimal weight of any representative of $\bar{Z}_L$ must be at least $d$, a contradiction as $\bar{X}_L$ has a representative of weight 1.

□

Here we also offer an alternate proof limiting universal transversal reversible computation for the subclass of stabilizer codes. The arguments here are based off of the BK hierarchy [16, 93]. We reproduce the cleaning lemma for completeness. These arguments showcase the rigid structure placed on stabilizer codes because of their transversal Pauli operations.

**Definition 6.12.** The *Clifford hierarchy* $\mathcal{C}$ is a sequence of gate sets $\{\mathcal{C}_k\}_{k\geq 1}$ defined recursively by $\mathcal{C}_k = \{U : U\mathcal{C}_1U^\dagger \subseteq \mathcal{C}_{k-1}\}$, where we define $\mathcal{C}_1$ to be the Pauli group.

Note that $\mathcal{C}_2$ is the Clifford group, and $\mathcal{C}_k$ fails to be a group for $k > 2$. Further note that reversible circuits saturate the Clifford hierarchy (and in fact can lie outside it entirely) by the gate $C^kX$, the $k$-controlled bit-flip gate, which lies in $\mathcal{C}_{k+1}$. Toffoli is simply $C^2X$, and so lies in the third level of the Clifford hierarchy. We next recall the stabilizer cleaning lemma, which can be found in [16].

**Lemma 6.13.** *Let $S$ be a stabilizer code, and let $R$ be any subset of physical qubits of the code such that any logical operator supported on $R$ acts trivially on $S$. Then, for any logical operator $U_L$, there exists a representative of $U_L$ supported on $R^c$.*

We call such subsets $R$ cleanable. Equipped with the cleaning lemma, we can now summarize the following lemma from [93].

**Lemma 6.14.** *Let $S$ be a stabilizer code and let $\{R_0, \ldots, R_k\}$ be a set of cleanable subsets of the physical qubits comprising $S$. Let $U$ be a logical operator supported on $\cup_{i=0}^k R_i$ such that $U$ is transversal with respect to the $R_i$. Then, $U_L \in \mathcal{C}_k$.*

*Proof.* We proceed by induction on $k$. In the base case, we have a logical operator $U$ supported on cleanable subsets $R_0 \cup R_1$. Let $P$ be any logical Pauli operator cleaned off of $R_1$, and let $[\cdot, \cdot]$ denote the group commutator. Since in a stabilizer code the logical Pauli operators are transversal, we have $Supp([U, P]) \subseteq R_0$, which by cleanability implies that $[U_L, P_L] = cI_L$. Since this is true for any $P_L$, it must be that $U_L \in \mathcal{C}_1$.

Similarly, suppose $U$ is supported on $\cup_{i=0}^{k} R_i$. Then, cleaning any logical Pauli $P$ off of $R_k$, we see that $Supp([U, P]) \subseteq \cup_{i=0}^{k-1} R_i$. By our inductive hypothesis, $[U_L, P_L] \subseteq \mathcal{C}_{k-1}$, which implies $U_L P_L U_L^\dagger \in \mathcal{C}_{k-1}$ for any logical Pauli $P_L$. Thus $U_L \in \mathcal{C}_k$, completing the proof. □

This argument generalizes to subsystem codes, and we refer the reader [93] for a more complete description. As a consequence we obtain the following.

**Corollary 6.15.** *No error-detecting stabilizer code can implement a classical reversible universal transversal gate set.*

*Proof.* Partition the code block into single subsystem subsets $\{R_1, \ldots, R_n\}$ where $n$ is the length of the code. Then, since the code is erasure-correcting, any logical operator supported on a single subsystem must act trivially on the codespace, and so these subsets are cleanable. By the lemma, any transversal logical gate must lie in $\mathcal{C}_n$. Since reversible circuits saturate $\mathcal{C}$, they cannot be logically transversally implementable. □

One final remark is that a similar technique was later used in [17] to explicitly demonstrate that any logical operator for a stabilizer code must belong to the Clifford hierarchy. The rough argument follows similarly from the fact that for any error-detecting code, the individual physical qubits themselves must constitute cleanable subsets.

## 6.5 Some sidesteps

In this section, we detail a few existing *stabilizer* code constructions that illustrate potential workarounds for our no-go theorem. While of course none of these construction has a native transversal Toffoli gate, each exemplifies a possible alternative.

### 6.5.1 Quantum Reed-Solomon codes

Let $1 \leq k < n < q$ and let $\mathbb{F}_q^k[x]$ denote the set of polynomials in $\mathbb{F}_q[x]$ of degree at most $k$. Fix an evaluation point $\vec{\gamma} \in \mathbb{F}_q^n : \gamma_i \neq \gamma_j$ for any $i \neq j$. Then we may

define the family of classical error-correcting codes known as Reed-Solomon codes as

$$RS_k^q(\vec{\gamma}) := \{f(\vec{\gamma}) : f \in \mathbb{F}_q^k[x]\}$$

where $f(\vec{\gamma}) := (f(\gamma_i))_i$. In general, Reed-Solomon codes are linear codes generated by the Vandermonde matrix, and so have parameters $[n, k+1, n-k]_q$ [94].

We can generalize such codes to *quantum Reed-Solomon codes* in the following way. These will be codes on qudits of dimension $q$. We define the code states as, for any $b \in \mathbb{F}_q$,

$$|b\rangle_L^k := \frac{1}{\sqrt{q^k}} \sum_{f \in \mathbb{F}_q^k : f(0)=b} |f(\vec{\gamma})\rangle .$$

It can be checked that such codes have parameters $[[n, 1, \min(k+1, n-k)]]_q$ [15]. We can also define the generalized Toffoli gate over $\mathbb{F}_q$ as

$$\mathrm{Toff}_q(|a, b, c\rangle) := |a, b, c + ab \pmod q\rangle .$$

Then, applying $\mathrm{Toff}_q$ strongly transversally to a quantum Reed-Solomon code state results in

$$\begin{aligned}\mathrm{Toff}_q^{\otimes n} |a\rangle_L^k |b\rangle_L^k |c\rangle_L^k &= |a\rangle_L^k |b\rangle_L^k \cdot \left(\frac{1}{q^k}\right) \sum_{j \in \mathbb{F}_2^{2k}[x] : j(0)=c+ab} |j(\vec{\gamma})\rangle \\ &= |a\rangle_L^k |b\rangle_L^k |c + ab \pmod q\rangle_L^{2k}\end{aligned}$$

So we can use a transversal Toffoli gate to map between *different* codes. This is not as desirable since eventually the blowup in $k$ will take us out of the family of quantum Reed-Solomon codes, in particular when $k \geq n$. Thus, this can only be used to practically implement finite-depth classical reversible computations fault-tolerantly. With an additional non-unitary operation called fault-tolerant degree reduction [41], one can reduce the degree of the underlying polynomial. However, because this requires fault-tolerant measurement, it does not fit into our definition of transversality, and as such incurs significant overhead.

### 6.5.2 [[8,3,2]]-color code

Next, we consider the $[[8, 3, 2]]$ color code defined in [95]. This code can be described by the following. The stabilizers are given by $ZZZZIIII$, $ZZIIZZII$, $ZIZIZIZI$, $ZZZZZZZZ$, and $XXXXXXXX$.

The three logical qubits have $X$- and $Z$- logical operators $XXXXIIII$, $ZIIIZIII$; $XXIIXXII$, $ZIZIIIII$; and $XIXIXIXI$, $ZZIIIIII$ respectively.

One can check that this code supports $CCZ_L = T \otimes T^\dagger \otimes T^\dagger \otimes T \otimes T^\dagger \otimes T \otimes T \otimes T^\dagger$. Such a code again does not implement a transversal Toffoli gate, but it demonstrates

how a standalone code can implement multi-qubit gates on its internal degrees of freedom. Although our definition for transversality is standard [14], this possibility for fault-tolerant Toffoli is left open by our construction. Still, this may have limited applicability as it is *a priori* unclear how to grow such a code into an infinite code *family* with similar fault-tolerance properties.

### 6.5.3 [[105,1,9]]-concatenated code

The $[[105, 1, 9]]$ code is an outer concatenation of Steane's $[[7, 1, 3]]$ code with an inner $[[15, 1, 3]]$ Reed-Muller code. It was proposed as a workaround for the Eastin-Knill theorem by supporting a universal transversal gate set over two *different* transversal partitions [96]. See Figures 6.3 and 6.4 for a description. Unfortunately, this quantum universal construction is inefficient, encoding one qubit into 105 with relatively low distance and only 1-fault-tolerance for each gate. Below, $\mathcal{G}$ refers to the transversal logical gates for that particular code.

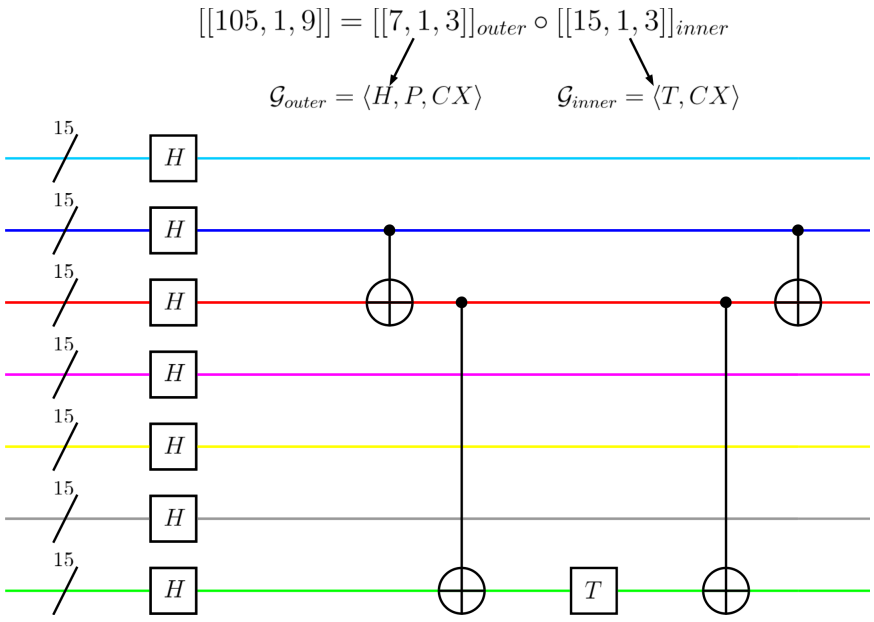



Figure 6.3: A partition of the physical qubits of the $[[105, 1, 9]]$ code on which the Clifford group is transversal. Each physical gate appearing represents a logical gate of the underlying $[[15, 1, 3]]$ code represented by each wire. The wire coloring corresponds to a particular partition. In this partition, the logical Hadamard gate to the left is transversal, but the logical $T$ gate to the right is not. This is because $T_L$ on Steane's code requires $CX$ gates that couple wires from different elements of the partition. However, the circuit for $T_L$ can be made transversal with another partition, see Figure 6.4.

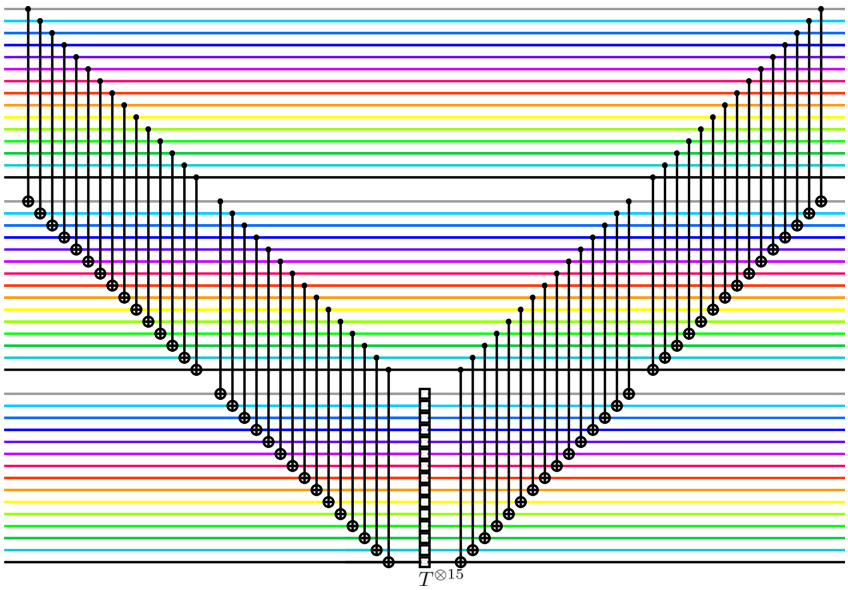



Figure 6.4: Blowing up the second, third, and seventh wires in Figure 6.3 for a total of 45 physical qubits, one can see that changing the partition allows the implementation of a transversal $T_L$. Although each physical gate respects the new partition, the overhead incurred is apparent.

## 6.6 Conclusion and general no-go workarounds

Do there exist non-additive maximally redundant codes that can then implement Toffoli transversally? One can essentially think of these as QECCs formed by concatenating an outer repetition code with a distance 1 inner code that is not a stabilizer subspace. Intuitively, since the inner code is not quantum error-correcting, the code only "spreads out the information in one basis". More precisely, the inner code only satisfies the diagonal QECC criterion. While this is a less restrictive condition, it still must be "complementary" to the outer code, and this allows us to argue impossibility in the additive case. Unfortunately by comparison, the structure of general non-additive codes is less well-understood – in particular, we know of no examples of such a code. We expect that *no QECC can implement Toffoli transversally*, and view this exception as a consequence of the lack of structure on general non-additive codes. We hope to resolve this exception in the future.

The QHE scheme we have detailed is non-leveled and compact, but highly inefficient. An immediate question would be to refine the security proof, which uses too strong a security demand. It would be most interesting to see if a modified approach can achieve *efficient* ITS-QHE for transversal gate sets of general quantum error-correcting codes, where the size of the encoding is some fixed polynomial of the input length. There are certain quantitative properties of "nonlocality" in QECCs (see e.g. [97], [98]) that might be helpful in such an endeavor. Following the out-

line of [12], we could also expect to extend a scheme built on a code with desirable transversal gates to accommodate a constant number of non-transversal gates. Just as one might tailor a QECC for a specific algorithm that makes heavy use of its transversal gate set, one might also tailor an ITS-QHE scheme to homomorphically implement that algorithm. Furthermore, it would be of theoretical interest to find a protocol matching the lower bound from Chapter 4.

Another interesting open question is to consider leveled ITS-QHE schemes: allow the client some preprocessing to scale with the size of the circuit. Can this relaxation allow more efficient or universal schemes for polynomial sized circuits, mirroring the computational security case? A first step might be to try to apply the techniques of instantaneous nonlocal computation [74] that proved invaluable in the computationally secure scheme. Moreover, through gauge-fixing, we have ways of converting between codes that together form a universal transversal gate set. Its not clear how to implement such a strategy, since the noisy embedding and non-interactivity present barriers to measuring syndromes, but these elements taken together might be useful in extending the current scheme.

Finally, one could ask if there is a correspondence between transversal gates for quantum codes and nontrivial ITS homomorphically-implementable gate sets, based on the "richness" of the function classes they can realize. In particular, [86] asked: what is the maximum size of finite group that can be implemented logically and transversally? Indeed, since the Clifford group on $p$-qubits is of size at most $2^{2p^2+3p}$ [99], one could reasonably expect to efficiently implement the Clifford gates homomorphically with information theoretic security, as was done in [12]. We hope that our arguments might extend past classical reversible circuit classes to address this question, although it is unclear how to generalize Nayak's bound to apply to these general finite subgroups of the unitary group.

# CHAPTER 7

## Transversal switching between generic stabilizer codes

As we have seen, methods for implementing fault-tolerant gates outside the framework of transversality are vital. Some of the candidates we have touched on include magic state distillation [48, 42], gauge fixing [90, 45], and more recently pieceable fault-tolerance [89, 52]. These last two candidates can be seen as a special case of the more general approach of code switching [100, 92, 101, 102, 103].

Code switching is a natural idea: given two codes, map information encoded in one code to information encoded in the other. For this mapping to be fault-tolerant, we must often perform several intermediate error-correction steps to ensure that faults do not grow out of hand. Thus, it is essential that during a circuit switching between codes, the extremal error-correcting codes are deformed through a series of intermediate error-correcting codes from one to another. This notion of intermediate error-correction was used in [92] to implement universal transversal computation by switching between the Steane and Reed-Muller codes, whose complementary transversal gate sets are universal when taken together. However, universal fault-tolerant computation is not the only consideration in choosing error-correcting codes, and different codes can be tailored to different tasks. For this reason, it would be nice to have a way of converting between different quantum codes fault-tolerantly.

Simply decoding and re-encoding information is undesirable, since the bare information becomes completely unprotected during this transformation. Past work has succeeded in constructing fault-tolerant circuits for switching between particular quantum error-correcting codes fault-tolerantly, while providing guarantees that these circuits are optimal within some framework [100].

Recently, [1] considered switching between generic stabilizer codes, and proposed the stabilizer rewiring algorithm (SRA) for constructing a transversal circuit mapping between *any* pair of stabilizer codes. The circuit complexity scales quadratically with the code length, and depends on a choice of presentation for the code generators. Different presentations will result in different circuits mapping between different sets

of at most $n$ intermediate codes. This circuit necessarily fails to be fault-tolerant when these intermediate codes have low distance. This leads to the central question: *is there an efficient way of fault-tolerantly converting between generic stabilizer codes?*

Towards this goal, we propose a randomized variant of the SRA, the randomized SRA (rSRA). We show that for any pair of stabilizer codes, with at most linear overhead with respect to the distance of the codes, there always exists a transversal circuit that maps between intermediate codes of high distance. Furthermore, using slightly more overhead, such a path can be found with high probability. In particular, we show the following.

**Theorem 7.1** (Informal)**.** *For any two $[[n, k, d]]$ stabilizer codes $S_1$ and $S_2$, the rSRA scheme gives a transversal circuit mapping from $S_1$ to $S_2$ where each intermediate code has distance at least $d$ with probability $1 - \varepsilon$, using*

$$m = O\left(d \log \frac{n}{d} + \log \frac{1}{\varepsilon}\right)$$

*ancilla qubits.*

This *distance-preserving* property is a necessary, but not sufficient condition to ensure a fault-tolerant mapping. So while the algorithm does *not* necessarily yield a fault-tolerant conversion, it gives a universal upper bound on the number of ancilla qubits required for distance-preserving transversal code transformation. As was noted in [1], the usefulness of this scheme is in its generality. While the upper bound may be of independent conceptual interest, we hope that with modification, the rSRA can be applied as a useful schema for searching for fault-tolerant paths between small codes. We provide small examples of such transversal paths in Section 7.2, including a path between the $[[5, 1, 3]]$ and $[[7, 1, 3]]$ codes that without modification protects against erasure with no overhead.

## 7.1 The randomized stabilizer rewiring algorithm

Let us quickly recall some of the concepts from section 3.4 in the special case of stabilizer *subspace* codes. A stabilizer subspace code $C_S$ has parameters $[[n, k, d]]$. Here, $n$ is the number of physical qubits comprising the code, $k$ is the number of logical qubits of the code $\log(\dim(C_S))$, and $d$ is the distance of the code. More precisely, the normalizer $\mathcal{N}_{\mathcal{P}^n}(S)$ represents the set of logical Pauli operators for $C_S$, and so

$$d := \min_{L \in \mathcal{N}(S) \setminus S} (|L|)$$

where $|\cdot|$ denotes the weight of the Pauli operator. Note that because we are defining a subspace code, the number of stabilizers in the corresponding stabilizer subgroup is $n-k$.

Given any stabilizer group $S$, if we choose a generating set $G_S$ for $S$, we can define a syndrome map

$$Syn_G : \mathcal{P}^n \longrightarrow \{0,1\}^{n-k}$$
$$Syn_G(e)_i = \begin{cases} 0 \text{ if } [e, g_i] = 0 \\ 1 \text{ if } \{e, g_i\} = 0 \end{cases}$$

for $G = (g_1, \dots g_{n-k})$. Then equivalently,

$$d = \min_{L \in \ker(Syn_G) \setminus S} (|L|)$$

and is independent of the choice of $G$.

Another convenient formalism for describing stabilizer groups is as subspaces of symplectic vector spaces, and this is the formulation we will use in this chapter. For any $P \in \mathcal{P}^n/\mathcal{U}(1)$, if

$$P = X^{a_1} Z^{b_1} \otimes X^{a_2} Z^{b_2} \dots \otimes X^{a_n} Z^{b_n}$$

then we can associate to $P$ the vector $\vec{P} := (\vec{a}|\vec{b})^T \in \mathbb{F}_2^{2n}$. Equip $\mathbb{F}_2^{2n}$ with a symplectic bilinear form

$$\langle \vec{v}, \vec{w} \rangle := \vec{v}^T B \vec{w}$$

where $B$ is the $2n \times 2n$ block matrix defined by

$$B = \begin{pmatrix} 0 & I \\ I & 0 \end{pmatrix}.$$

Then Paulis $P, Q$ commute if any only if their associated vectors $\vec{P}, \vec{Q}$ are orthogonal in this vector space. Thus, we can equivalently define a stabilizer group as a self-orthogonal subspace of this vector space. A *generator matrix* $G$ is then a choice of basis for this subspace, so that for $C$ an $[[n,k]]$ code, $G$ will be a rank $(n-k)$ matrix of shape $2n \times (n-k)$. The syndrome map can then be similarly defined as

$$Syn_G(\vec{P}) = G^T B \vec{P}.$$

Further note that for any $A \in GL(\mathbb{F}_2, n-k)$, for any generator matrix $G$ for $S$, $GA^T$ is also a generator matrix for $S$. The syndrome map satisfies

$$Syn_{GA^T}(\vec{P}) = (GA^T)^T B \vec{P} = A G^T B \vec{P} = A \cdot Syn_G(\vec{P}).$$

So any action on the generator matrix induces a corresponding action on the syndrome vectors themselves.

Finally, recall that a circuit $C$ on a class of encoded inputs *t-fault-tolerant* if it is $t$-fault-tolerant in the exRec formalism [39] (see section 3.6.3). Again, given error correction procedure $EC$, $C$ is $t$-fault-tolerant if for any choice of $t$ faulty components in the combined circuit $EC \cdot C \cdot EC$, a faultless version of $EC$ applied to the output of the combined circuit can successfully recover the data. If $t \geq 1$ we may simply call the circuit fault-tolerant.

### 7.1.1 The rSRA schematic

The rSRA modifies the SRA presented in [1], whose central insight is the following. Consider two stabilizer groups $S, S'$ with generating sets $G, G'$ satisfying the following nice property:

$$G = \{g, g_1, \ldots, g_l\}$$
$$G' = \{g', g_1, \ldots, g_l\}$$

where $\{g, g'\} = 0$. We call two such codes for which one can choose such generating sets *adjacent*. Then one can readily check that the Clifford gate $\frac{1}{\sqrt{2}}(1 + g'g)$ maps information encoded in the stabilizer code defined by $G$ to the same information encoded in the stabilizer code defined by $G'$. Letting $|\psi\rangle_G$ denote a logical state in the code associated to $G$, we see that

$$\begin{aligned} g_i \cdot \frac{1}{\sqrt{2}}(1 + g'g)\,|\psi\rangle_G &= \frac{1}{\sqrt{2}}(1 + g'g)\,|\psi\rangle_G\,, \text{ and} \\ g' \cdot \frac{1}{\sqrt{2}}(1 + g'g)\,|\psi\rangle_G &= \frac{1}{\sqrt{2}}(g' + g)\,|\psi\rangle_G \\ &= \frac{1}{\sqrt{2}}(g' + 1)\,|\psi\rangle_G \\ &= \frac{1}{\sqrt{2}}(1 + g'g)\,|\psi\rangle_G\,. \end{aligned}$$

The insight is that this mapping can be done transversally. While the Clifford transformation described need not be transversal, it can be simulated by a transversal Pauli measurement supplemented by a transversal Pauli gate controlled on classical information. This is similar to gauge-fixing, in which one measures a logical operator of the gauge and then applies a corresponding logical gauge operator conditioned on the outcome. To see this, consider the circuit described by:

1. Measure $g'$.

2. Apply $g$ conditioned on measurement outcome $-1$.

Let $P^{\pm}$ denote the projector onto the $+1/-1$ eigenspace of $g'$. Then, if the measurement outcome is $+1$,

$$\frac{1}{\sqrt{2}}(1+g'g)\left|\psi\right\rangle_G = \frac{1}{\sqrt{2}}(1+g')\left|\psi\right\rangle_G = \sqrt{2}P^{+}\left|\psi\right\rangle_G .$$

Furthermore,
If the measurement outcome is $-1$,

$$\begin{aligned}\frac{1}{\sqrt{2}}(1+g'g)\left|\psi\right\rangle_G &= \frac{1}{\sqrt{2}}(g-gg')\left|\psi\right\rangle_G \\ &= \frac{1}{\sqrt{2}}g(1-g')\left|\psi\right\rangle_G \\ &= \sqrt{2}gP^{-}\left|\psi\right\rangle_G .\end{aligned}$$

Thus, we see that we can *transversally* perform the mapping $\left|\psi\right\rangle_G \to \left|\psi\right\rangle_{G'}$.

Now consider the more general case in which we have (non-adjacent) $S, S'$ describing $[[n,k]]$ and $[[n',k]]$ codes respectively. We now describe a general randomized algorithm for outputting a circuit switching between these two codes, similar to [1], and will later show that this circuit is distance-preserving with high probability. The inputs are arbitrary generator matrices $G, G'$ for stabilizer groups $S, S'$, along with a choice of ancilla size $m \in \mathbb{N}$.

### 7.1.2 Preparing the generator matrices

1. Append $\left|0\right\rangle$ ancilla to the smaller code so that the codes are of equal size. We now assume that both codes are $[[n,k]]$ codes.

2. Append $\left|0\right\rangle^{\otimes m}$ to the first code, and $\left|+\right\rangle^{\otimes m}$ to the second. Note that this is equivalent to defining a pair of new stabilizer codes

$$\hat{S} = \langle S \otimes I^{\otimes m}, I^{\otimes n} \otimes Z \otimes I^{\otimes m-1}, \ldots, I^{\otimes n+m-1} \otimes Z \rangle,$$

$$\hat{S}' = \langle S' \otimes I^{\otimes m}, I^{\otimes n} \otimes X \otimes I^{\otimes m-1}, \ldots, I^{\otimes n+m-1} \otimes X \rangle.$$

3. Choose $G_A = G'_A$ to be a basis for the subspace defined by $\hat{S} \cap \hat{S}'$.

4. Choose $G_B$ to extend the basis of $G_A$ to a basis for $\mathcal{N}(\hat{S}') \cap \hat{S}$ and choose $G'_B$ to extend the basis of $G_A$ to a basis for $\mathcal{N}(\hat{S}) \cap \hat{S}'$.

5. Choose $G_C$ to extend the basis $G_A \cup G_B$ to a basis for $\hat{S}$ and $G'_C$ to extend the basis $G'_A \cup G'_B$ to a basis for $\hat{S}'$.

6. Let $H$ be the *commutativity matrix* for $G_C, {G'_C}$ defined by $H := {G'_C}^T B G_C$. By Lemma 7.3, $H$ is invertible with dimension $|G_C| \times |G_C|$, where $|G_C| \geq m$. So we can choose $M, N \in GL(\mathbb{F}_2, |G_C|) : M^T H N = I_{|G_C|}$ and redefine

$$G_C \leftarrow G_C \cdot M$$
$$G'_C \leftarrow G'_C \cdot N.$$

7. Choose uniformly at random $V, V' \in_r \mathbb{F}_2^{|G_C| \times |G_B|}$ and a $U \in_r GL(\mathbb{F}_2, |G_C|)$.

8. Redefine

$$G_C^T \leftarrow U(V G_B^T + G_C^T)$$
$${G'_C}^T \leftarrow (U^{-1})^T (V' {G'_B}^T + {G'_C}^T)$$

   Note that this does not change the commutativity matrix since

$$U(V G_B^T + G_C^T) B (G'_C + G'_B {V'}^T) U^{-1} = I_{|G_C|}.$$

9. Let $G_B = \{g_1, \ldots, g_{|G_B|}\}$ and $G'_B = \{g'_1, \ldots, g_{|G'_B|}\}$. For each $g_i \in G_B$, choose $\overline{g_i}$ satisfying

$$\begin{aligned}
[\overline{g_i}, G_A] &= 0 \\
[\overline{g_i}, G_C] &= 0 \\
[\overline{g_i}, G'_C] &= 0 \\
[\overline{g_i}, \{g_{i+1}, \ldots, g_{|G_B|}\}] &= 0 \\
[\overline{g_i}, \{g'_{i+1}, \ldots, g'_{|G_B|}\}] &= 0 \\
[\overline{g_i}, \{\overline{g_1}, \ldots, \overline{g_{i-1}}\}] &= 0 \\
\{\overline{g_i}, g_i\} &= 0 \\
\{\overline{g_i}, g'_i\} &= 0.
\end{aligned}$$

   To see that such a choice of $\overline{g_i}$ always exists, note that it must satisfy at most $2n$ affine linear equations, all of which are linearly independent, in a space of dimension $2n$.

   Now that we have prepared the generator matrices, we will step-by-step map between adjacent codes transversally.

#### 7.1.3 Applying the transformation

10. For $1 \leq i \leq |G_B|$ indexing the elements of $G_B$, perform the transformation $g_i \mapsto \overline{g_i}$. Note that the resulting stabilizer codes are adjacent, and so the preceding discussion gives a transversal circuit for each mapping.
11. For $1 \leq i \leq |G_C|$ indexing the elements of $G_C$, perform the transformation $g_i \mapsto g_i'$. Again, since the codes are adjacent, the mapping can be done transversally.
12. For $1 \leq i \leq |G_B|$ indexing the elements of $G_B'$, perform the transformation $\overline{g_i} \mapsto g_i'$ starting from $i = |G_B|$ and working backwards towards $i = 1$. Again, we have a transversal circuit for each mapping.
13. Discard the ancilla.

This randomized variant differs from the original SRA in several ways. First, there is the introduction of ancilla, which we will see are vital for preserving the distance. Next, the SRA fixes the generating sets $G, G'$ subject to the same $G_A$ and $G_C$ conditions, but with different $G_B$ conditions. Namely, the SRA fixes the $\overline{g}$ to be the product of the complementary logical operators to those operators in $G_B$ and $G_B'$, which can be seen as nontrivial logical operators on the opposite code. This allows for a certain degree of freedom in choosing the order in which one converts between the two codes, but restricts the $G_C$,$G_C'$ that are available to use. Also in the SRA, only the set of valid permutations among $G_B$ and $G_C$ are considered, which restricts the search for a distance-preserving mapping. In the rSRA, we consider the full set of invertible transformations on $G_C$ for a better chance of success. Finally, the transformation described above is *symmetric* in the sense that switching from $G$ to $G'$ or $G'$ to $G$ after step 9 results in the same set of intermediate codes. We will see that this simplifies the set of errors we must consider.

### 7.2 Distance-preservation for small codes

We have now described a way of constructing a transversal circuit mapping information encoded in $G$ to information encoded in $G'$ through the use of Shor-style measurement (see Figure 3.2).

However, we have no *a priori* guarantee that these intermediate codes, resulting from the sequence of deformations, will themselves be error-correcting. In light of this, we offer several examples of small distance-preserving circuits generated from the rSRA. These illustrate the necessity of the aforementioned modifications, which are centered around choosing a path so that all of the intermediate codes have high distance. In these examples, the extremal codes all have distance 3, and so we call the circuit distance-preserving if the intermediate codes all have distance $\geq 3$.

### 7.2.1 $[[7,1,3]] \longleftrightarrow [[5,1,3]]$

With $m = 0$, one can generate a distance-preserving map from the $[[7,1,3]]$ Steane code to the perfect $[[5,1,3]]$ code using the rSRA with 17 multi-qubit gates. An optimal fault-tolerant (and so distance-preserving) transformation using $CZ$ gates between these two codes was found via brute force search in [100] and involves 14 multi-qubit gates. The circuit output by the rSRA requires no overhead in data qubits compared to the three extra qubits required in [100]. However, because the $[[5,1,3]]$ code is perfect, any conversion without ancilla must only be able to protect against erasure, for reasons detailed in Section 7.4. Note also that there must be conversions with large separation between the circuit provided by the rSRA and the optimal fault-tolerant circuit, in particular when $G$ and $G'$ are locally unitarily equivalent.

| Type | $[[7,1,3]]$ | $[[5,1,3]]$ |
|---|---|---|
| $G_A$ | $-YXXYIZZ$ | $-YXXYIZZ$ |
| $G_C$ | $ZZZZIII$ | $IXZZXII$ |
| | $-YYXXZZI$ | $XZZXIII$ |
| | $-IXZYYZX$ | $XIXZZII$ |
| | $-XIYZYZX$ | $ZXIXZII$ |
| | $-ZYYZIXX$ | $IIIIIZI$ |

Table 7.1: The generator matrices defining a distance-preserving conversion, proceeding from top to bottom. We follow steps 10 - 13 of the algorithm.

### 7.2.2 $(34) \cdot [[7,1,3]] \longleftrightarrow [[9,1,3]]$

With $m = 0$, one can convert from the (34) permutation of the $[[7,1,3]]$ Steane code to Shor's $[[9,1,3]]$ code while preserving the distance. However, for the standard choice of generator matrices, no permutation on the ordering of the deformations will suffice. Thus, we must choose $U \in GL(\mathbb{F}_2, |G_C|)$ rather than restricting $U$ to be a permutation matrix. A choice of generator matrices for which the circuit is distance-preserving is presented below.

### 7.2.3 $[[7,1,3]] \longleftrightarrow (34) \cdot [[7,1,3]]$

For $m = 0$, it was observed in [1] that one cannot use the SRA to convert between the $[[7,1,3]]$ code, and the (34) permutation of the $[[7,1,3]]$ code while preserving the distance. In fact, there does not exist a $U \in GL(\mathbb{F}_2, |G_C|)$ that allows the intermediate codes to be error-correcting. In contrast, with $m = 2$, there does exist such a distance-preserving circuit, emphasizing the need for ancilla. Moreover, brute force search shows that this is the minimal number of ancilla required to

| Type | $(34)\cdot[[7,1,3]]$ | $[[9,1,3]]$ |
|---|---|---|
| $G_A$ | $ZZIIZZIII$ | $ZZIIZZIII$ |
| | $IIIIIIIZZ$ | $IIIIIIIZZ$ |
| $G_C$ | $YIIYYIYII$ | $-YXYZZIXXX$ |
| $G_B$ | $ZZZZIIIZI$ | $ZZIIIIZZI$ |
| $G_C$ | $-ZZYYXXIII$ | $IZZZZIIII$ |
| | $ZIIZZIZII$ | $-YYXXXXIII$ |
| | $-XZZXYIYII$ | $-XYYIIIXXX$ |
| | $-IYZXZXYII$ | $IIIZZIIII$ |

Table 7.2: The conversion proceeds from top to bottom. As the $G_B$ elements commute, we perform an intermediate conversion to the product of the complementary logical operators, which in this case are $XXXXXXXXX$ and $XXXXXXXII$ respectively. This small modification is similar to the SRA [1], which we adopt here for ease of presentation.

produce a distance-preserving circuit within this framework. However, note that qubit permutations are themselves automatically fault-tolerant by simply relabeling the wires, rather than applying a fault-tolerant physical SWAP gate.

| Type | $[[7,1,3]]$ | $(34)\cdot[[7,1,3]]$ |
|---|---|---|
| $G_A$ | $XXIIXXIII$ | $XXIIXXIII$ |
| | $ZZZZIIIII$ | $ZZZZIIIII$ |
| | $ZZIIZZIII$ | $ZZIIZZIII$ |
| | $XXXXIIIII$ | $XXXXIIIII$ |
| $G_C$ | $XIXIXIXII$ | $YIIYYIYXX$ |
| | $IIIIIIIZZ$ | $XIIXXIXIX$ |
| | $ZIZIZIZIZ$ | $IIIIIIIXX$ |
| | $YIYIYIYZZ$ | $XIIXXIXXX$ |

Table 7.3: The conversion proceeds from top to bottom. In particular, we use 2 extra ancilla qubits, for 9 physical qubits in total.

## 7.3 Distance preservation

We now show that, with low overhead and high probability, the described rSRA will yield a distance-preserving circuit. More specifically, we show that the intermediate codes preserve the distance of the extremal codes. We begin with some technical lemmas.

**Lemma 7.2.** *Let* $v, w \in \{0,1\}^n \backslash \{0\}$ *and* $U \in_r GL(\mathbb{F}_2, n)$. *Let* $i_0 = \max\{i : (U\cdot v)_i = 1\}$ *and* $i_1 = \min\{i : ((U^{-1})^T \cdot w)_i = 1\}$. *Then,*

$$\Pr[i_0 < i_1] \leq (n-1)\cdot 2^{-n}.$$

*Proof.* Let $\langle\cdot,\cdot\rangle$ be the dot product over $\mathbb{F}_2$. Note that $\langle v, w\rangle = \langle U\cdot v, (U^{-1})^T \cdot w\rangle$. If $\langle v, w\rangle = 1$, then there must be at least one entry where both $U\cdot v$ and $(U^{-1})^T \cdot w$

are 1 for whichever $U$ we choose, and so $\Pr[i_0 < i_1] = 0$. Therefore we only need to consider the case in which $\langle v, w \rangle = 0$.

Consider the action of $GL(\mathbb{F}_2, n)$ on $A$ defined by $U(v, w) \to (U \cdot v, (U^{-1})^T \cdot w)$, where $A = \{(v, w) | v, w \in \{0,1\}^n \setminus \{0\}, \langle v, w \rangle = 0\}$. We show that the action is transitive by showing that for all such pairs $(v, w)$, there always exists a $U$ sending $(e_1, e_n)$ to $(v, w)$, where $e_1, e_n$ are $(1, 0, \ldots, 0)$ and $(0, 0, 0, \ldots, 1)$, respectively. Given such a $(v, w)$, first extend $v$ to a basis for $w^\perp$, say $(u_1 = v, u_2, \ldots, u_{n-1})$, and then extend it to the whole space by adding in $u_n$. We claim that $U = (u_1, \cdots, u_n)$ is the desired matrix. It is sufficient to show that the last column $w'$ of $(U^{-1})^T$ is exactly $w$. We have $U^T w' = e_n$ given that $U^T (U^{-1})^T = I$, and that $U^T w = e_n$ by construction of $U$. Then, since $U$ is invertible, $w = w'$.

A uniformly random distribution over invertible $U$ then induces a uniformly random distribution over $A$. Then $\Pr[i_0 < i_1]$ can then be bounded by counting the number of such pairs in $A$:

$$\begin{aligned}
\Pr[i_0 < i_1] &= \frac{\sum_{i_0=1}^{n} 2^{i_0 - 1}(2^{n - i_0} - 1)}{(2^n - 1)(2^{n-1} - 1)} \\
&= \frac{(n-2)2^{n-1} + 1}{(2^n - 1)(2^{n-1} - 1)} \\
&\leq (n-1) \cdot 2^{-n}
\end{aligned}$$

when $n \geq 2$. Note that $|A| = 0$ when $n = 1$, so $\Pr[i_0 < i_1] \leq (n-1) \cdot 2^{-n}$ holds for all $n \geq 0$. □

**Lemma 7.3.** *Let $G_A, G_B, G_C$ and $G_A, G'_B, G'_C$ be the matrices defined up to step 5 in the rSRA scheme. The commutativity matrix $H = G_C^T B G'_C$ is invertible, and its dimension is $|G_C|$, with $|G_C| \geq m$.*

*Proof.* For the two codes $\hat{S}$ and $\hat{S}'$, take arbitrary generator matrices $G$,$G'$ and define $H' = G^T B G'$. Note that any two choices of generator matrices for the same code differ by an invertible row transformation, so the rank of $H'$ is invariant under different choices of the generator matrices. In particular, letting $G = (G_A | G_B | G_C), G' = (G'_A | G'_B | G'_C)$, we have

$$H' = \begin{bmatrix} 0 & & \\ & 0 & \\ & & H \end{bmatrix}.$$

Note that $\text{rank}(H) = |G_C|$, or else there would exist a combination of the rows of ${G'_C}^T$ that are orthogonal to all the columns of $G_C$. Since all the vectors in $G'_C$ are already orthogonal to $G_A$ and $G_B$ by definition, this cannot happen as no vector

in $G'_C$ lies in $\mathcal{N}(S)$. The same argument applies to $G'_C$ as well. Therefore $H$ is invertible, with $\text{rank}(H') = \text{rank}(H) = |G_C|$, and is independent of the choice of $G_C$.

To show that $|G_C| > m$, take $\bar{G}_C = (I^{\otimes n} \otimes Z \otimes I^{\otimes m-1}, \ldots, I^{\otimes n+m-1} \otimes Z)$ and $\bar{G}'_C = (I^{\otimes n} \otimes X \otimes I^{\otimes m-1}, \ldots, I^{\otimes n+m-1} \otimes X)$, each of size $m$. By extending them to generator matrices $\bar{G}$ and $\bar{G}'$ for $\hat{S}$ and $\hat{S}'$ respectively, we get a commutativity matrix $\bar{H}$ with an invertible submatrix of size $m \times m$, namely

$$\bar{G}_C^T B \bar{G}'_C = I_m,$$

and so $\text{rank}(\bar{H}) = |G_C| \geq m$. □

**Lemma 7.4.** *For $D(\cdot||\cdot)$ the KL-divergence, let*

$$P(n, m, d) = 4^{n+m} e^{-D(\frac{d}{n+m}\|\frac{3}{4})(n+m)} \cdot (m+1) \cdot 2^{-m}.$$

*Then $P < \varepsilon$ for some $m = O(d \log \frac{n}{d} + \log \frac{1}{\varepsilon})$.*

*Proof.* Let $\alpha = m/n$. Then $P(n, m, d) < \varepsilon$ can be rewritten as

$$\begin{aligned} f(n, m, d) :&= \log \frac{P(n, m, d)}{\varepsilon} \\ &= \log \frac{m+1}{\varepsilon} + n\Bigg((2+\alpha)\log 2 \\ &\quad - (1+\alpha) D\left(\frac{d}{n(1+\alpha)}\|\frac{3}{4}\right)\Bigg) < 0. \end{aligned}$$

We first compute the dominant term, i.e. the $\alpha$ such that

$$(2+\alpha)\ln 2 - (1+\alpha) D\left(\frac{d}{n(1+\alpha)}\|\frac{3}{4}\right) = 0.$$

Doing this we obtain

$$\begin{aligned} (2 - \frac{\alpha}{1+\alpha})\ln 2 &= D\left(\frac{d}{n(1+\alpha)}\|\frac{3}{4}\right) \\ (2 - \frac{\alpha}{1+\alpha})\ln 2 &\geq 2\ln 2 + \frac{d}{n(1+\alpha)}\left(\ln \frac{d}{3n(1+\alpha)} - 1\right) \\ \alpha n &\leq \frac{d}{\ln 2}\left(\ln \frac{3n(1+\alpha)}{d} + 1\right) \\ m &\leq \frac{1}{\ln 2} d(\log \frac{n}{d} + (1 + \ln 3)), \end{aligned}$$

where we have used convexity of $D(p\|q) - p\ln p$ with respect to $p$. Letting $\tilde{\alpha}$ denote the solution to $(2+\alpha)\ln 2 - (1+\alpha) D\left(\frac{d}{n(1+\alpha)}\|\frac{3}{4}\right) = 0$, we have that $\tilde{m} := \tilde{\alpha} n = O(d + d \log \frac{n}{d})$.

We now have that $f(n, \tilde{m}, d) = \log \frac{\tilde{m}+1}{\epsilon}$. Taking the derivative of $f$ with respect to $m$, for all $\alpha > \tilde{\alpha}$ we have

$$\begin{aligned}
&\frac{\partial f(n,m,d)}{\partial m} = \\
&= \frac{1}{m+1} - D(\frac{d}{(n+m)} \| \frac{3}{4}) - (m+n)\frac{\partial D(\frac{d}{n+m} \| \frac{3}{4})}{\partial m} \\
&\leq \frac{1}{m+1} - \frac{2+\tilde{\alpha}}{1+\tilde{\alpha}} \log 2 + \frac{d}{m+n}\left(\log \frac{d}{3(m+n-d)}\right) \\
&\leq \frac{1}{m+1} - \frac{1}{1+\tilde{\alpha}} \log 2 + \frac{d}{n+m}\left(\log \frac{d}{3(n+m-d)}\right) \\
&\leq -\frac{1}{1+\tilde{\alpha}} \ln 2 + 0.1
\end{aligned}$$

for $m \geq 10$. For fixed $n$, $\tilde{\alpha}$ is monotonically increasing as a function of $d$. By the quantum singleton bound, $\frac{d-1}{n} < \frac{1}{2}$, and $\tilde{\alpha} < 3$ even in this case. Therefore $\frac{\partial f(n,m,d)}{\partial m} \leq -0.05$ when $m \geq 10$, so taking

$$m = \tilde{m} + 20 \log \frac{\tilde{m}+1}{\epsilon} + O(1) = O(d \log \frac{n}{d} + \log \frac{1}{\varepsilon})$$

suffices to make $f(n, m, d) < 0$. □

**Theorem 7.5.** *Let $S$,$S'$ be any two stabilizer codes with parameters $[[n_1, k, d_1]]$ and $[[n_2, k, d_2]]$, respectively. Let $d = \min\{d_1, d_2\}$ and $n = \max\{n_1, n_2\}$. Then, the rSRA will output a distance-preserving circuit mapping information encoded in $S$ to information encoded in $S'$ with probability $1 - \epsilon$ using $m = O(d \log \frac{n}{d} + \log \frac{1}{\varepsilon})$ ancilla qubits.*

*Proof.* Consider a particular error $e : |e| < d$. There are four different types of errors to consider.

(1) $\underline{e \in S \cap S'}$: In this case, $e \in Span(G_A)$, and so remains passively corrected throughout the transformation.

(2) $\underline{e \in S \setminus \mathcal{N}(S')}$: In this case, we can decompose $e = g_A + g_B + g_C$ where $g_A \in Span(G_A), g_B \in Span(G_B)$, and $g_C \in Span(G_C)$. Furthermore, $g_C \neq 0$, or else $e$ would be a logical operator of weight $< d$ for $S'$. Thus, $e$ must be detected by $G'_C$, and so it remains detectable after step 11. In particular, before the end of step 11, $e$ must fall out of the intermediate stabilizer group. Suppose this occurs for the first time when transforming between two adjacent codes whose stabilizer groups differ by $g, g'$. Then we can write $e = g + \sum_i a_i g_i$, and as $g'$ commutes with all other $g_i$, it must be that $\{e, g'\} = 0$. Since $g'$ remains in each intermediate code up through

step 11, $e$ must be detectable throughout.

(3) $\underline{e \in S' \setminus \mathcal{N}(S)}$: This error is just an error of type (2) when performing the opposite transformation from $S'$ to $S$. By symmetry of the scheme, the set of intermediate codes during this opposite transformation is the same, and so these errors remain detectable by the preceding argument.

(4) $\underline{e \notin \mathcal{N}(S) \cup \mathcal{N}(S')}$: Let $G_C^{(0)}, G_C'^{(0)}$ be the bases $G_C$ and $G'_C$ we choose after step 6 in the rSRA scheme, and let $G_C^{(1)}, G_C'^{(1)}$ be the bases we choose after step 8. Note that the syndrome map for $G_C^{(1)}$ can then be expressed as

$$Syn_{G_C^{(1)}}(e) = U(V \cdot Syn_{G_B}(e) + Syn_{G_C^{(0)}}(e)).$$

In this case it must be that

$$(Syn_{G_A}(e)|Syn_{G_B}(e)|Syn_{G_C^{(0)}}(e))^T \neq 0,$$
$$(Syn_{G_A}(e)|Syn_{G'_B}(e)|Syn_{G'^{(0)}_C}(e))^T \neq 0.$$

Note that if $Syn_{G_A}(e) \neq 0$, then $e$ is always detectable since each intermediate code includes the check operators from $G_A$. Thus, we only need to consider the case where $Syn_{G_A}(e) = 0$, and so we can assume that $(Syn_{G_B}(e)|Syn_{G_C^{(0)}}(e))^T \neq 0$ and $(Syn_{G'_B}(e)|Syn_{G'^{(0)}_C}(e))^T \neq 0$.

Let $P_e$ denote the probability that the error $e$ is undetectable in some intermediate code over the random choices of $U$, $V$, and $V'$. We divide $P_e$ into three parts. Let $A_e$ denote the event that $Syn_{G_C^{(1)}}(e) = 0$, $B_e$ the event that $Syn_{G'^{(1)}_C}(e) = 0$, and let $C_e$ denote the event that both $Syn_{G_C^{(1)}}(e)$ and $Syn_{G_C'^{(1)}}(e)$ are nonzero, yet $e$ becomes undetectable on some intermediate code during the transformation. Then $P_e \leq \Pr[A_e] + \Pr[B_e] + \Pr[C_e]$. We bound $\Pr[A_e]$, $\Pr[B_e]$, and $\Pr[C_e]$ separately. To bound $\Pr[A_e]$, note that

$$Syn_{G_C^{(1)}}(e) = U(V \cdot Syn_{G_B}(e) + Syn_{G_C^{(0)}}(e)).$$

Since $U \in GL(\mathbb{F}_2, n-k)$, $A_e$ occurs if and only if $V \cdot Syn_{G_B}(e) + Syn_{G_C^{(0)}}(e) = 0$. If $Syn_{G_B}(e) = 0$, it must be the case that $Syn_{G_C^{(0)}}(e) \neq 0$, and so $Syn_{G_C^{(1)}}(e) \neq 0$; otherwise $Syn_{G_B}(e) \neq 0$ and $V \cdot Syn_{G_B}(e) + Syn_{G_C^{(0)}}(e)$ is uniformly random over $\{0,1\}^{|G_C|}$. In either case, we have

$$\Pr[A_e] \leq 2^{-|G_C|}.$$

Repeating the same argument shows that $\Pr[B_e] \leq 2^{-|G_C|}$ as well. To bound $\Pr[C_e]$, define

$$v = V \cdot Syn_{G_B}(e) + Syn_{G_C^{(0)}}(e),$$
$$w = V' \cdot Syn_{G'_B}(e) + Syn_{G'^{(0)}_C}(e).$$

Since $Uv, (U^{-1})^T w \neq 0$, $e$ will be detectable during steps 10 and 12, and so $C_e$ occurs only if $e$ becomes undetectable during step 11. Specifically, it must be that $Syn_{G_A}(e) = 0, Syn_{\overline{G}_B}(e) = 0$, and the last 1 in the vector $Uv$ occurs before the first 1 in the vector $(U^{-1})^T w$. This is because we are sequentially replacing the check operators of $G$ with the check operators of $G'$, and so an error becomes undetectable for some intermediate code only if we produce some zero syndrome during this sequence of substitutions. By Lemma 7.2, for two nonzero vectors $v, w \in \{0,1\}^{|G_C|}$, the probability that the last 1 in $Uv$ comes before the first 1 in $(U^{-1})^T w$ is bounded by $(|G_C| - 1) \cdot 2^{-|G_C|}$.

Summing these three terms, we have $P_e \leq (|G_C| + 1) \cdot 2^{-|G_C|}$. Taking a union bound, the probability $P$ that any of the intermediate codes fail to detect any error of weight less than $d$ is upper bounded by

$$P \leq \sum_{e:|e|<d} P_e \leq |\{e : |e| < d\}| \cdot (|G_C| + 1) \cdot 2^{-|G_C|}.$$

Taking a Chernoff bound, we get that this is in turn upper bounded as

$$P \leq 4^{n+m} \cdot e^{-D(\frac{d-1}{n+m} || \frac{3}{4})(n+m)} \cdot (|G_C| + 1) \cdot 2^{-|G_C|}$$

where $D(\cdot||\cdot)$ is the KL-divergence. By the quantum singleton bound, we can assume $\frac{d-1}{n+m} < \frac{d-1}{n} < \frac{3}{4}$. Furthermore, by Lemma 7.3, $|G_C|$ is given by $\text{rank}(G^T B G')$, which is at least $m$. So the probability of failure can be further upper bounded by

$$P \leq 4^{n+m} \cdot e^{-D(\frac{d-1}{n+m} || \frac{3}{4})(n+m)} \cdot (m + 1) \cdot 2^{-m}.$$

It suffices to choose $m$ such that the above quantity is upper bounded by $\epsilon$ in order to achieve a high probability of success. In particular, the case $\varepsilon = 1$ upper bounds the minimum number of ancilla qubits required for a fault-tolerant transformation. By Lemma 7.4 we observe that taking

$$m = O(d \log \frac{n}{d} + \log \frac{1}{\varepsilon})$$

is sufficient for the rSRA scheme to succeed with probability $1 - \epsilon$.

□

## 7.4 Conclusion and prospects for fault-tolerance

Theorem 7.5 shows that with high probability, the rSRA will produce a transversal circuit with intermediate codes that have distances *at least* the minimum of the distances of the extremal codes. It is important to note that this does *not* necessarily imply fault-tolerance. The reason is because, when measuring $g'$, the randomness in the outcome prevents us from using that syndrome bit during error-correction. More specifically, consider the following two scenarios.

1. We project onto the $(+1)$-eigenspace of $g'$.
2. We project onto the $(-1)$-eigenspace of $g'$ and simultaneously experience an error that anticommutes with only $g'$.

Then we cannot distinguish these two scenarios using only our syndrome bits, and so cannot correct the resulting error. More generally, we can cast the property required for fault-tolerance in terms of subsystem codes. For every conversion between adjacent codes, we consider the subsystem code with a single gauge degree of freedom corresponding to gauge operators $g'$ and $g$. Then the resulting conversion will be $t$-fault-tolerant precisely when the resulting subsystem code has distance $2t + 1$. This is because the redundant syndrome information can diagnose errors without the syndrome bit associated to $g'$, and so ensure that we project onto the correct eigenspace. For this reason, additional techniques will be required to achieve fault-tolerance using the rSRA, such as error-detection on the ancilla. We leave this to future work.

These techniques contrast with recent results from [89], where it was shown that pieceable fault-tolerance offers generic *fault-tolerant* code switching between stabilizer codes subject to certain constraints. However, their techniques require that the codes are nondegenerate and have some set of native fault-tolerant Clifford gates, allowing a fault-tolerant SWAP gate *between* different codes. One could also consider preparing a second code state and using logical teleportation to achieve a fault-tolerant mapping [101].

Practically, on small examples, one finds that often *no* ancilla qubits are required to find a distance-preserving circuit, which is desirable as the resulting circuit may then be fault-tolerant. In general, this can be attributed to a coarse accounting of $|G_C|$ in terms of the number $m$ of ancilla qubits. In most cases, $N(S) \cap N(S')$ will be small, and so the ancilla will be superfluous.

Moreover, the multi-qubit gate complexity of the algorithm is $\sum_{P \in \{\overline{g_i}\} \cup G_B \cup G_C} |P|$, so that choosing a low weight generating set is ideal for reducing the complexity of the code switching circuit. For this reason, LDPC codes might provide more efficient

code switching circuits, although preserving the distance may depend on choosing a high weight set of generators.

This algorithm derives its usefulness from its generality. For specific code switching examples, it may be profitable to modify the circuit using the rSRA as a template, augmented with a larger class of fault-tolerant manipulations such as local Clifford gates, in order to search for a *fault-tolerant* mapping. For large code sizes, the use of high-weight Shor-style measurements is limiting as it requires large verified CAT states. Thus, this technique may be most useful as a step in a concatenated scheme, or simply as a search ansatz.

One subtlety about the rSRA is that, while it outputs a distance-preserving circuit switching between two codes with high probability, this is difficult to check. This follows from the difficulty of computing the minimum distance of a generic error-correcting code, which is an NP-hard problem in general [104]. Indeed, even when restricting to a particular distance, this check remains extremely costly. This poses a barrier to derandomizing the algorithm, which would be one desirable avenue for future improvement.

Another such improvement would be to minimize overhead. One could imagine taking a random local clifford transformation in order to increase the size of $G_C$, rather than introducing ancilla. Such a strategy would be interesting since locally equivalent codes have nearly identical properties. Of course, modifying the algorithm to ensure fault-tolerance is the most important improvement.

If it is true that one can always choose locally equivalent representatives for which the rSRA provides a distance-preserving conversion without ancilla, this would suggest that all error-protected information in stabilizer codes is, in some sense, "transversally equivalent". This contrasts with the diverse set of equivalence classes of locally unitarily equivalent codes, which can be identified as distinct submanifolds of Grassmanians. Indeed, it may be of conceptual interest to interpret these upper-bounds in a broader framework of fault-tolerance, such as the one investigated in [105].

Similarly, the generality of the rSRA provides an aesthetically nice interpretation of error-protected information. It suggests that, with the addition of some minimal overhead, any stabilizer error-protected encoding of information is indeed "transversally equivalent" to any other.

# CHAPTER 8

# Summary and conclusions

In this thesis, we have studied several questions relating to quantum fault-tolerance, and a tangentially related question appearing in quantum cryptography. While all of these results lean on the side of theory, they investigate possibilities that may be available down the road. With quantum computing in its nascent stages, it is unclear which existing or yet to exist proposal may prove a magic bullet for the construction of larger fault-tolerant quantum computers. The intention of this thesis is to inject more knowledge into this burgeoning field, in the hope that we may ultimately utilize the computational power of quantum mechanics. Towards this goal, we have contributed the following.

## 8.1 Intermediate 2-D compass codes

### 8.1.1 Summary

From [20], we constructed a novel class of local subsystem codes which we call intermediate compass codes. We use these codes to study threshold behavior between Bacon-Shor codes, which are optimally 2-local but fail to have a threshold, and rotated surface codes, which are 4-local but boast a high threshold. We give evidence that the threshold scales linearly with the expected ratio of stabilizer generators to qubits.

Noting that these codes have the nice property that their excitations are created in pairs, we use a minimum-weight matching decoder to study their behavior in different subspace code families. We find that these codes can be easily tailored to asymmetric noise models, providing higher thresholds in these settings. However, we do so at the cost of locality, which scales linearly with the bias of the noise. For fixed bias, however, these code families remain local. We also analyze the behavior of randomized subspace code families; for these families, the expected locality scales logarithmically with the lattice size. Such codes display similar threshold behavior to their regularly-defined cousins.

### 8.1.2 Future work

In the future, we would like to test this model on realistic noise models sampled from quantum devices. For example, while dephasing noise is the dominant process on all devices, its bias could vary as a collection of discrete Gaussians centered on problematic qubits. Intermediate compass codes give a natural template for tailoring codes to geometrically correlated noise. We also expect that employing decoder techniques similar to [61, 106] may boost the threshold up even further.

The other natural question is to descend from the code capacity model to the circuit error model to determine whether these gains are maintained. One barrier is the increasing non-locality of the stabilizer checks, which may introduce correlated errors depending on the measurement scheme used. It would be interesting to consider whether these codes can be designed to mitigate such correlated errors, perhaps by using techniques similar to [107, 108]. Finally, while these codes may be promising as quantum memories, we have also not yet detailed schemes for implementing fault-tolerant operations.

## 8.2 Quantum homomorphic encryption

### 8.2.1 Summary

In [22], we study existing proposals for extending classical homomorphic encryption to the quantum setting. This problem is particularly important for quantum information, as the average user is likely to outsource any quantum computation to a third-party quantum device. We study this problem in the information-theoretically secure regime, where there has been evidence that quantum computers may have the potential for accomplishing this task with statistical security [6, 11, 12, 77].

In [21], we show that a quantum device enacting even a classical-reversible universal set of homomorphisms with $\epsilon$-information theoretic security must use encoding sizes that scale exponentially with the input size. This was observed concurrently in [13], and precludes any hope for the strongest notion of information-theoretically secure quantum fully homomorphic encryption.

### 8.2.2 Future work

It would be of theoretical interest to rule out the possibility of information-theoretically secure leveled quantum fully homomorphic encryption. Also, similar to the restrictions found in [11], what we would like to argue is that any doubly-exponential family of unitary homomorphisms cannot be implemented efficiently, rather than just the set of all Boolean functions. Both of these directions seem to require a modification of Nayak's bound.

Finally, although we have ruled out quantum fully homomorphic encryption with the most stringent security guarantees, we have seen that weakened information-theoretic security can sometimes offer much larger sets of homomorphisms. In fact, [77] offers a scheme that implements a *continuum* of homomorphisms with some weaker information-theoretic security guarantees. Although these homomorphisms are not universal for quantum computing (and so the scheme is not fully-homomorphic), can we design a similar scheme that *is* universal?

## 8.3 Transversal gates

### 8.3.1 Summary

In [21], we consider which families of gates may be implemented transversally on quantum subspace error-detecting codes. We show that a particularly valuable gate, the Toffoli gate, cannot hope to be implemented unless the codes take a very special form. We suspect that there are no such codes. In doing so, we show that within our framework, there are no codes that can implement a classical-reversible universal transversal gate set, supplementing the no-go theorem for quantum universal transversal gate sets [109]. Our proof uses an information-theoretic approach, reducing to lower bounds derived in Chapter 5, rather than the previous Lie group approach. We further show that for the special class of error-detecting stabilizer codes, there can be no transversal Toffoli gate by showing that the stabilizer group structure is incommensurate with the special structure required to violate our lower bound.

### 8.3.2 Future work

Finding fault-tolerance schemes for implementing classical-reversible computation are of paramount importance. Many useful quantum algorithms are dominated by classical-reversible circuits. Furthermore, these are the circuits on which we can test the fidelity of our outputs. We details some existing proposals in Chapter 6, but emphasize that we must do better.

Understanding the full picture of transversal gates is also important. Practically speaking, the overwhelming majority of transversal gates are known only for the extremely specialized class of CSS codes. Furthermore, these gates are almost always Clifford gates, iterated roots of $Z$, or iterated controls of $Z$. It is reasonable to ask if these gates constitute *all* such transversal gates? Perhaps there is a connection between the complexity of a circuit class and whether or not it can be realized transversally on a quantum error-detecting code.

## 8.4 Transversal code switching

### 8.4.1 Summary

In [23], we consider the problem of constructing circuits that switch between stabilizer subspace codes, building on the proposal in [1]. We prove theoretical upper-bounds on the requirements for our construction to yield a sequence of codes that preserves the distance throughout, answering an open question in [1]. We note that this is a necessary but insufficient requirement for fault-tolerance, and certain malignant errors can still cause a failure within our circuit. However, there do exist examples within our framework displaying degrees of fault-tolerance, particularly when switching between nondegenerate codes with $|G_C|$ (as defined in Chapter 7) large, as if often the case. We give toy examples of such code mappings on small codes.

### 8.4.2 Future work

The central open question here is whether these constructions can be made fully fault-tolerant. While our proposal may be a useful ansatz for searching for code-switching circuits, we are motivated by the question: *what are the minimal requirements necessary for switching between any two different stabilizer codes fault-tolerantly.* To achieve this goal, entirely new techniques may be needed.

More generally, code switching and gauge-fixing are only one proposal for implementing non-transversal fault-tolerant logical gates. The major obstacle facing such gates is the necessity of magic-state distillation, an enormous expense that accounts for upwards of 99% of the required resources for a computation [48]. While gauge-fixing, code switching, and pieceable fault-tolerance have provided alternative approaches, no such proposal has yielded a competitively high threshold thus far. The major question then becomes, how does one avoid magic-state distillation more generally?

## 8.5 Final remarks

In this thesis, we have considered several problems relating to the fault-tolerance of quantum information using the active error-correction paradigm. There remain alternative proposals for physical fault-tolerance, such as topological quantum computing and thermodynamically stable quantum memories, which we have not touched on. It is unclear in the end what form a fault-tolerant quantum computer will take, but it is the author's personal belief that the concerted efforts of mathematicians, physicists, computer scientists, and chemists will get us there.

# APPENDICES

# APPENDIX A

## Visualizing randomized bias codes

Here, for visualization, we include a pictorial description of two members of the randomized bias code family. The first set of diagrams (Figures 1 through 3) represent a member of the family, along with its corresponding $X$- and $Z$- type stabilizers.

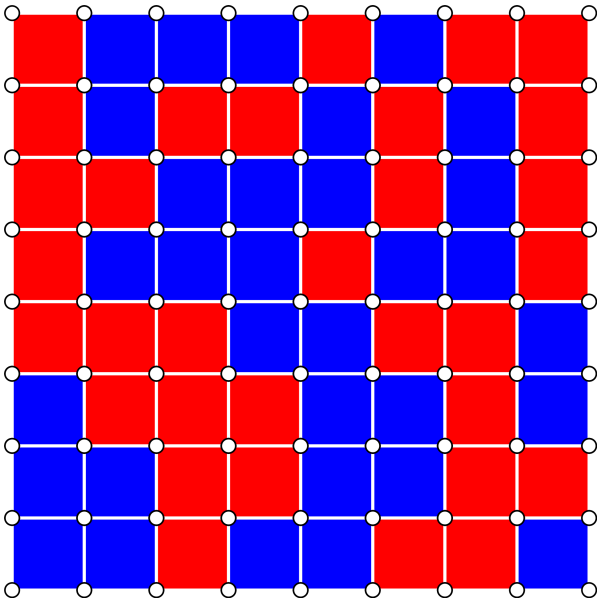


Figure 1: An example of a randomly generated intermediate compass code according to symmetric bias $\eta = 1$ on a $9 \times 9$ lattice. The red plaquettes are minimal $X$-plaquettes and the blue plaquettes are minimal $Z$-plaquettes.

The second set (Figures 4 through 6) depict another member of the randomized family, along with an illustration of its associated $X$- and $Z$- type Ising models.

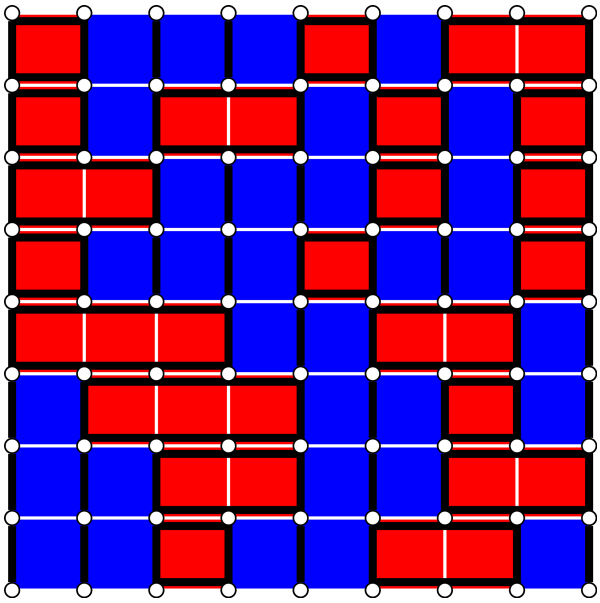


Figure 2: The intermediate compass code from Figure 1 with the $X$-type stabilizers included in black lines.

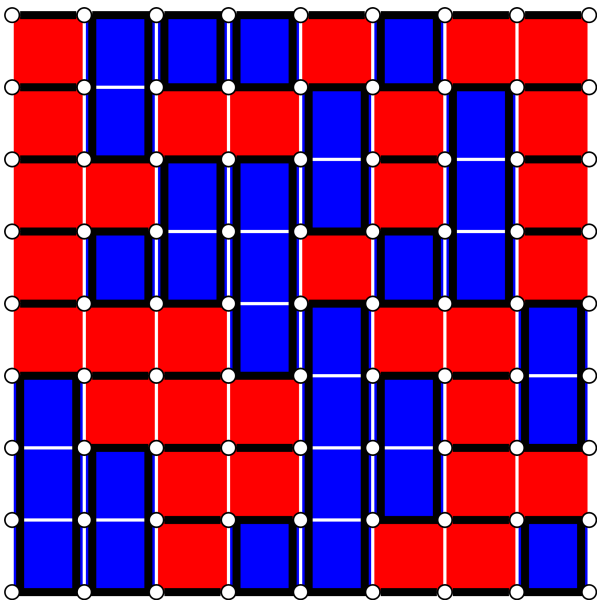


Figure 3: The intermediate compass code from Figure 1 with the $Z$-type stabilizers included in black lines.

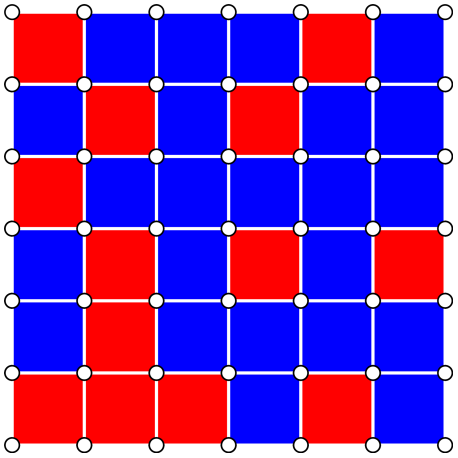


Figure 4: Another example of a randomly generated intermediate compass code according to asymmetric bias $\eta = 2$ on a $7 \times 7$ lattice. The red plaquettes are minimal $X$-plaquettes and the blue plaquettes are minimal $Z$-plaquettes. See Figures 5 and 6 for the associated Ising models.

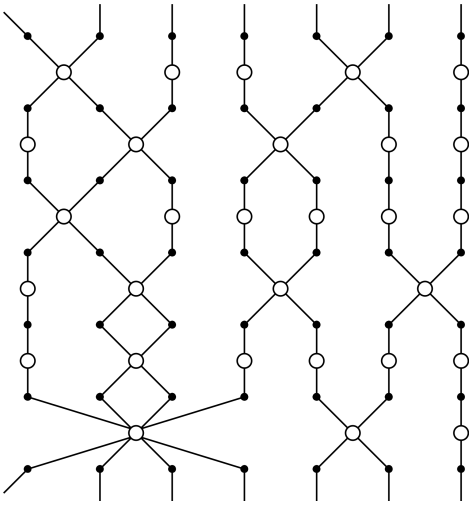


Figure 5: The Ising model associated to $X$-type errors coming from the code defined in Figure 4. Note that the connectivity of the lattice is sparser, corresponding to the relative infrequency of $X$-type errors.

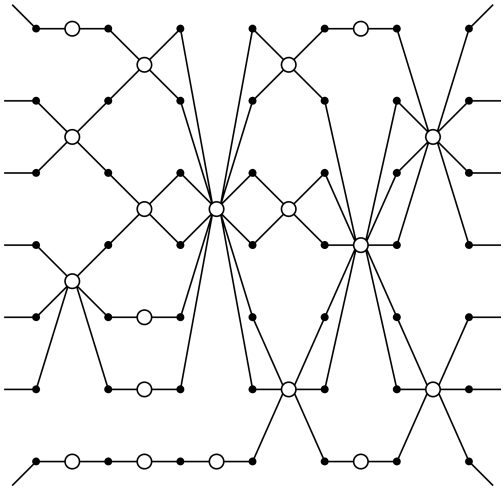


Figure 6: The Ising model associated to $Z$-type errors coming from the code defined in Figure 4. Note that the connectivity of the lattice is denser, corresponding to the relative frequency of $Z$-type errors.

# APPENDIX B

## Thermalization snapshots

In this section, we provide snapshots of the Ising fields during the same time-step in thermalization over different parameter regimes. This helps to visualize the different random-graph Ising models associated to the $q$-codes in section 4.5.3. The snapshots are taken at virtual temperature defined by the Nishimori line

$$T = \frac{2}{\log(1-p) - \log(p)}$$

for different disorders $p$, and for different values of $q$ parametrizing the codes. Each snapshot is taken after 5000 time cycles of a Metropolis algorithm, well before thermalization.

The model used to generate these differs slightly in that it has periodic boundary conditions in order to minimize the effects of the boundary field, and has a constant number of spins in order to fit the plot. In spite of this, it displays similar critical behavior. The lattice is of size $512 \times 512$, and appears in Figure 1.

In the top row of Figure 1, at low temperatures, one can observe the vertical shearing due to exclusively 2-local vertical interactions. At $q = 0$, this corresponds to Bacon-Shor codes, and can be seen as disjoint vertical copies of a 1D Ising model. As we progress down the rows we increase $q$, and see that the vertical shearing disappears and the ordering increases. Eventually, we would see the presence of an ordered phase as domain walls begin to form from clustered spins of the same value at low $p$.

However, as $p$ increases, the model becomes increasingly disordered, nearing threshold behavior around $q = 1.0$, $p = 0.10$. Here, in the random-bond 2D Ising model, the phase transition is estimated to occur around $p \approx 11\%$ [110]. As a final note, the data used here and throughout this thesis will be made publicly available at `https://github.com/mgn2109`.

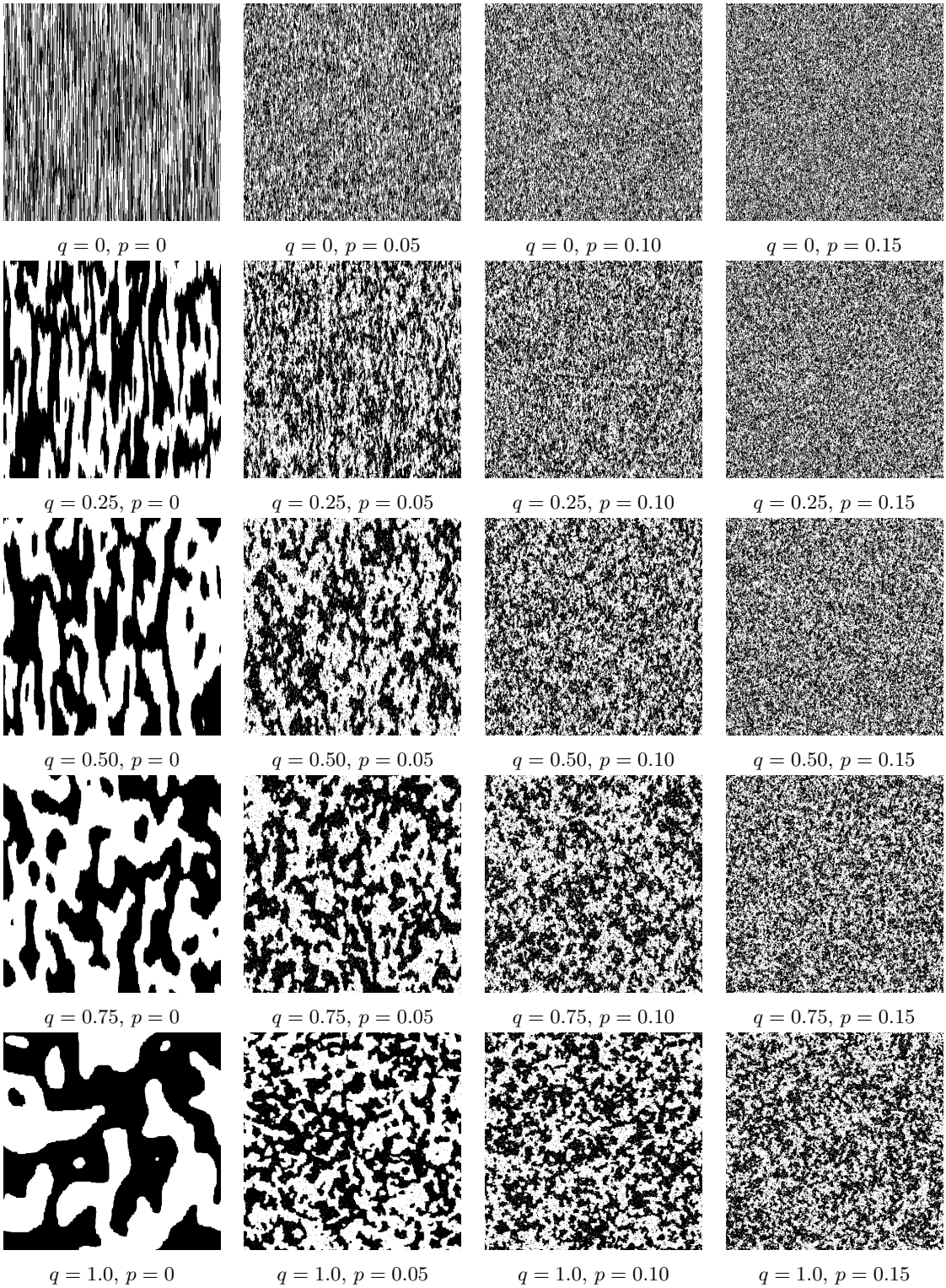



Figure 1: A visualization of the Ising fields associated to different $q$-codes at different disorders $p$ during thermalization.

# BIBLIOGRAPHY

# BIBLIOGRAPHY


[1] Kristina R. Colladay and Erich J. Mueller. Rewiring Stabilizer Codes, July 2017. arXiv:1707.09403.

[2] Richard Feynman. Simulating physics with computers, 1982. Internat. J. Theoret. Phys., 21, pp. 467488.

[3] Lov K. Grover. A fast quantum mechanical algorithm for database search, 1996. Proceedings, 28th Annual ACM Symposium on the Theory of Computing (STOC).

[4] Peter W. Shor. Fault-tolerant quantum computation, 1996. 37th Symposium on Foundations of Computing, IEEE Computer Society Press, pp. 56-65.

[5] Ben W. Reichardt and Falk Unger and Umesh Vazirani. Classical command of quantum systems, April 2013. Nature 496, 456460.

[6] Anne Broadbent and Joseph Fitzsimons and Elham Kashefi. Universal blind quantum computation, 2009. Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2009), pp. 517-526.

[7] C. H. Bennett and G. Brassard. Quantum cryptography: Public key distribution and coin tossing, 1984. International Conference on Computers, Systems and Signal Processing, IEEE.

[8] Carl A Miller and Yaoyun Shi. Robust protocols for securely expanding randomness and distributing keys using untrusted quantum devices, 2016. Journal of the ACM (JACM).

[9] A. Yu. Kitaev. Fault-tolerant quantum computation by anyons, 2003. Annals Phys. 303.

[10] Benjamin J. Brown and Daniel Loss and Jiannis K. Pachos and Chris N. Self and James R. Wootton. Quantum memories at finite temperature, 2016. Rev. Mod. Phys. 88.

[11] Li Yu and Carlos A. Perez-Delgado and Joseph F. Fitzsimons. Limitations on information theoretically secure quantum homomorphic encryption, June 2014. Phys. Rev. A 90, 050303 (2014).

[12] Yingkai Ouyang and Si-Hui Tan and Joseph Fitzsimons. Quantum homomorphic encryption from quantum codes, August 2015. arXiv:1508.00938.

[13] Ching-Yi Lai and Kai-Min Chung. On Statistically-Secure Quantum Homomorphic Encryption, 2017. arXiv:1705.00139.

[14] Bei Zeng and Andrew Cross and Isaac L. Chuang. Transversality versus Universality for Additive Quantum Codes, September 2011. IEEE Transactions on Information Theory, Volume: 57, Issue: 9, 6272 - 6284.

[15] Andrew W. Cross. Fault-tolerant Quantum Computer Architectures Using Hierarchies of Quantum Error-correcting Codes, 2008. PhD Thesis.

[16] Sergey Bravyi and Robert Koenig. Classification of topologically protected gates for local stabilizer codes, 2013. Phys. Rev. Lett. 110, 170503 (2013).

[17] Tomas Jochym-O'Connor and Aleksander Kubica and Theodore J. Yoder. The disjointness of stabilizer codes and limitations on fault-tolerant logical gates, 2017. arXiv:1710.07256.

[18] Michael A. Nielsen and Isaac L. Chuang. Quantum Computation and Quantum Information, 2011. Cambridge University Press New York, NY.

[19] Daniel A. Lidar and Todd A. Brun. Quantum error-correction, 2013. Cambridge University Press.

[20] Michael Newman and Muyuan Li and Daniel Miller and Yukai Wu and Kenneth R. Brown. Intermediate 2-D compass codes, 2018. In preparation.

[21] Michael Newman and Yaoyun Shi. Limitations on transversal computation through quantum homomorphic encryption, April 2017. arXiv:1704.07798.

[22] Michael Newman. Information-theoretically secure quantum homomorphic encryption and its limitations, 2018. Preprint, `http://www-personal.umich.edu/~mgnewman/LimitationsOnHomomorphicEncryption.pdf`.

[23] Cupjin Huang and Michael Newman. Transversal switching between generic stabilizer codes, 2017. arXiv:1709.09282.

[24] Stephen Hancock and Jennifer Hom and Michael Newman. On the knot Floer filtration of the concordance group, 2014. Journal of Knot Theory and Its Ramifications, Volume 22, Issue 14.

[25] Tabes Bridges and Rankeya Datta and Joseph Eddy and Michael Newman and John Yu. Free and Very Free Morphisms into a Fermat Hypersurface, 2013. Involve 6, No. 4, 437445.

[26] Fang Zhang and Cupjin Huang and Michael Newman and Kevin Sung and Yaoyun Shi. Limitations on testing quantum theory, 2017. Preprint, `http://www-personal.umich.edu/~mgnewman/DI_Simulation.pdf`.

[27] M.-D. Choi. Completely Positive Linear Maps on Complex Matrices, 1975. Linear Algebra and its Applications, 10.

[28] W. F. Stinespring. Positive Functions on C*-algebras, 1955. Proceedings of the American Mathematical Society, 6.

[29] Daniel Gottesman. The Heisenberg Representation of Quantum Computers, 1999. Group22: Proceedings of the XXII International Colloquium on Group Theoretical Methods in Physics, eds. S. P. Corney, R. Delbourgo, and P. D. Jarvis, pp. 32-43.

[30] Tomasso Toffoli. Reversible Computing, 1980. ICALP: Automata, Languages and Programming pp 632-644.

[31] Yaoyun Shi. Both Toffoli and Controlled-NOT need little help to do universal quantum computation, 2003. Quantum Information and Computation, 3(1):84-92.

[32] A. Uhlmann. The Transition Probability in the State Space of a *-Algebra, 1976. Rep. Math. Phys. 9.

[33] Simon J. Devitt and Kae Nemoto and William J. Munro. Quantum Error Correction for Beginners, 2013. Rep. Prog. Phys. 76.

[34] Sergey Bravyi and Matthias Englbrecht and Robert Koenig and Nolan Peard. Correcting coherent errors with surface codes, 2017. arXiv:1710.02270.

[35] Daniel Greenbaum and Zachary Dutton. Modeling coherent errors in quantum error correction, 2018. Quantum Sci. Technol. 3 015007.

[36] Emanuel Knill and Raymond Laflamme. A Theory of Quantum Error-Correcting Codes, April 1996. Phys.Rev.Lett.84:2525-2528.

[37] Daniel Gottesman. Stabilizer Codes and Quantum Error Correction, 1997. Caltech Ph.D. Thesis.

[38] E. Dennis and A. Kitaev and A. Landahl and J. Preskill. Topological quantum memory, 2002. J. Math. Phys., 43:4452.

[39] Panos Aliferis and Daniel Gottesman and John Preskill,. Quantum accuracy threshold for concatenated distance-3 codes, 2006. Quantum Inf. Comput. 6, 97165.

[40] E. Knill and R. Laflamme and W.H. Zurek. Accuracy threshold for Quantum Computation, 1996. quant-ph/9610011.

[41] D. Aharonov and M. Ben-Or. Fault-tolerant quantum vomputation with constant error, 1997. Proceedings of 29th Annual ACM Symposium on Theory of Computing, page 46.

[42] Sergey Bravyi and Jeongwan Haah. Magic state distillation with low overhead, 2012. Phys. Rev. A 86, 052329.

[43] Bryan Eastin. Distilling one-qubit magic states into Toffoli states, 2013. Phys. Rev. A 87, 032321.

[44] Peter W. Shor. Scheme for reducing decoherence in quantum computer memory, 1995. Phys. Rev. A 52, R2493(R).

[45] H. Bombin. Gauge Color Codes: Optimal Transversal Gates and Gauge Fixing in Topological Stabilizer Codes, August 2015. New J. Phys. 17 (2015) 083002.

[46] Y. Tomita and K.M. Svore. Low-distance Surface Codes under Realistic Quantum Noise, 2014. Phys. Rev. A, 90, 062320.

[47] Theodore J. Yoder and Isaac H. Kim. The surface code with a twist, 2017. Quantum 1, 2.

[48] Austin G. Fowler and Simon J. Devitt and Cody Jones. Surface code implementation of block code state distillation, January 2013. Scientific Reports 3, 1939.

[49] D.S. Wang and A.G. Fowler and A.M. Stephens and L.C.L. Hollenberg. Threshold error rates for the toric and surface codes, 2010. Quant. Inf. Comput. 10:456.

[50] D. Bacon. Operator quantum error-correcting subsystems for self-correcting quantum memories, 2006. Phys. Rev. A 73, 012340.

[51] P. Aliferis and A. Cross. Subsystem fault-tolerance with the Bacon-Shor code, 2007. Physical Review Letters 98, 220502.

[52] Theodore J. Yoder. Universal fault-tolerant quantum computation with Bacon-Shor codes, May 2017. arXiv:1705.01686.

[53] Fernando Pastawski and Alastair Kay and Norbert Schuch and Ignacio Cirac. Limitations of Passive Protection of Quantum Information, 2010. Quantum Inf. Comput. 10, 580.

[54] Cody Jones and Peter Brooks and Jim Harrington. Gauge color codes in two dimensions, 2016. Phys. Rev. A 93, 052332.

[55] Sergey Bravyi and Andrew Cross. Doubled Color Codes, 2015. arXiv:1509.03239.

[56] Tomas Jochym-O'Connor and Stephen D. Bartlett. Stacked codes: universal fault-tolerant quantum computation in a two-dimensional layout, 2016. Phys. Rev. A 93, 022323.

[57] J. Napp and J. Preskill. Optimal Bacon-Shor Codes, 2013. Quant. Inf. Comput. 13, 490-510.

[58] Martin Suchara and Andrew W. Cross and Jay M. Gambetta. Leakage Suppression in the Toric Code, 2015. Quant. Inf. Comp. Vol. 15, No. 11/12, pp. 997-1016.

[59] Hector Bombin. Topological Subsystem Codes, 2010. Phys. Rev. A 81, 032301.

[60] K. Binder. Finite size scaling analysis of Ising model block distribution functions, 1981. Physik B - Condensed Matter 43: 119.

[61] David K. Tuckett and Stephen D. Bartlett and Steven T. Flammia. Ultrahigh Error Threshold for Surface Codes with Biased Noise, 2018. Phys. Rev. Lett. 120, 050505.

[62] Courtney G. Brell. A proposal for self-correcting stabilizer quantum memories in 3 dimensions (or slightly less), 2016. New J. Phys. 18, 013050.

[63] Austin G. Fowler and Matteo Mariantoni and John M. Martinis and Andrew N. Cleland. Surface codes: Towards practical large-scale quantum computation, August 2012. Phys. Rev. A 86, 032324 (2012).

[64] Muyuan Li and Mauricio Gutirrez and Stanley E. David and Alonzo Hernandez and Kenneth R. Brown. Fault Tolerance with Bare Ancillae for a [[7,1,3]] Code, 2017. Phys. Rev. A 96, 032341.

[65] Sergey Bravyi. Subsystem codes with spatially local generators, 2011. Phys. Rev. A 83, 012320.

[66] Craig Gentry. A fully homomorphic encryption scheme, 2009. Ph.D. Thesis, Stanford University.

[67] Chris Peikert. A Decade of Lattice Cryptography, March 2016. Foundations and Trends in Theoretical Computer Science 10(4):283-424.

[68] Peter W. Shor. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer, 1997. SIAM J.Sci.Statist.Comput. 26, 1484.

[69] Zvika Brakerski and Adeline Langlois and Chris Peikert and Oded Regev and Damien Stehl. Classical Hardness of Learning with Errors, 2013. arXiv:1306.0281 [cs.CC].

[70] Craig Gentry and Amit Sahai and Brent Waters. Homomorphic Encryption from Learning with Errors: Conceptually-Simpler, Asymptotically-Faster, Attribute-Based, 2013. Volume 8042 of the series Lecture Notes in Computer Science pp. 75-92.

[71] Amin Baumeler and Anne Broadbent. Quantum Private Information Retrieval has linear communication complexity, April 2013. Journal of Cryptology. Volume 28, Issue 1, pp 161-175 (2015).

[72] Anne Broadbent and Stacey Jeffery. Quantum homomorphic encryption for circuits of low T-gate complexity, 2015. In Proceedings of Advances in Cryptology – CRYPTO 2015, pp 609-629.

[73] Yfke Dulek and Christian Schaffner and Florian Speelman. Quantum homomorphic encryption for polynomial-sized circuits, August 2016. CRYPTO 2016: Advances in Cryptology - CRYPTO 2016, pp 3-32.

[74] Florian Speelman. Instantaneous non-local computation of low T-depth quantum circuits, November 2015. arXiv:1511.02839.

[75] Gorjan Alagic and Yfke Dulek and Christian Schaffner and Florian Speelman. Quantum Fully Homomorphic Encryption With Verification, 2017. Proceedings of ASIACRYPT.

[76] Urmila Mahadev. Classical Homomorphic Encryption for Quantum Circuits, 2017. arXiv:1708.02130.

[77] Si-Hui Tan and Joshua A. Kettlewell and Yingkai Ouyang and Lin Chen and Joseph F. Fitzsimons. A quantum approach to homomorphic encryption, 2016. Sci. Rep. 6, 33467.

[78] Si-Hui Tan and Yingkai Ouyang and Peter P. Rohde. Practical quantum somewhat-homomorphic encryption with coherent states, 2017. arXiv:1710.03968.

[79] Joseph F. Fitzsimons. Private quantum computation: an introduction to blind quantum computing and related protocols, 2017. NPJ Quantum Information, Volume 3, Article number: 23.

[80] Max Fillinger. Lattice based cryptography and fully homomorphic encryption, 2012. `http://homepages.cwi.nl/~schaffne/courses/reports/MaxFillinger_FHE_2012.pdf`.

[81] Eyal Kushilevitz and Rafail Ostrovsky. Replication is not needed: Single database, computationally-private information retrieval, 1997. FOCS, pg. 364.

[82] Amit Chakrabarti and Anna Shubina. Nearly Private Information Retrieval, 2007. MFCS, pgs. 383-393.

[83] M. A. Nielsen and I. L. Chuang. Programmable quantum gate arrays, 1997. Phys. Rev. Lett, 79, 321-4.

[84] Ashwin Nayak. Optimal lower bounds for quantum automata and random access codes, April 1999. FOCS 1999.

[85] Anne Broadbent and Zhengfeng Ji and Fang Song and John Watrous. Zero-knowledge proof systems for QMA, April 2016. Proceedings of the 2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS 2016) pp.31-40.

[86] Bryan Eastin and Emanuel Knill. Restrictions on Transversal Encoded Quantum Gate Sets, July 2009. Phys. Rev. Lett. 102, 110502.

[87] Cody Jones. Composite Toffoli gate with two-round error detection, March 2013. Phys. Rev. A 87, 052334.

[88] Hendrik Poulsen Nautrup and Nicolai Friis and Hans J. Briegel. Topological Code Switching in Two Dimensions, September 2016. arXiv:1609.08062.

[89] Theodore J. Yoder and Ryuji Takagi and Isaac L. Chuang. Universal fault-tolerant gates on concatenated stabilizer codes, March 2016. Phys. Rev. X 6, 031039 (2016).

[90] Adam Paetznick and Ben W. Reichardt. Universal fault-tolerant quantum computation with only transversal gates and error correction, April 2013. Phys. Rev. Lett. 111, 090505 (2013).

[91] Cody Jones. Novel constructions for the fault-tolerant Toffoli gate, 2013. Phys. Rev. A 87, 022328.

[92] Jonas T. Anderson and Guillaume Duclos-Cianci and David Poulin. Fault-tolerant conversion between the Steane and Reed-Muller quantum codes, 2014. Phys. Rev. Lett. 113, 080501.

[93] Fernando Pastawski and Beni Yoshida. Fault-tolerant logical gates in quantum error-correcting codes, 2015. Phys. Rev. A 91, 012305.

[94] Irving Reed and Gustave Solomon. Polynomial Codes over Certain Finite Fields, 1960. Journal of the Society for Industrial and Applied Mathematics (SIAM), 8 (2): 300304.

[95] Earl T. Campbell. The smallest interesting colour code, 2016. `https://earltcampbell.com/2016/09/26/the-smallest-interesting-colour-code/`.

[96] T. Jochym-O'Connor and R. Laflamme. Using concatenated quantum codes for universal fault-tolerant quantum gates, 2014. Physical Review Letters 112 (1), 010505.

[97] Ludovic Arnaud and Nicolas J. Cerf. Exploring pure quantum states with maximally mixed reductions, January 2013. Phys. Rev. A 87, 012319 (2013).

[98] A. J. Scott. Multipartite entanglement, quantum-error-correcting codes, and entangling power of quantum evolutions, May 2004. Phys. Rev. A 69, 052330.

[99] Maris Ozols. Notes on the Clifford group, July 2008. `http://home.lu.lv/~sd20008/papers/essays/Clifford%20group%20[paper].pdf`.

[100] Charles D Hill and Austin G Fowler and David S Wang and Lloyd CL Hollenberg. Fault-tolerant quantum error correction code conversion, 2013. Quantum Inf. Comput. 13, 439451.

[101] Todd A. Brun and Yi-Cong Zheng and Kung-Chuan Hsu and Joshua Job and Ching-Yi Lai. Teleportation-based Fault-tolerant Quantum Computation in Multi-qubit Large Block Codes, April 2015. arXiv:1504.03913.

[102] H. Bombin and M.A. Martin-Delgado. Quantum Measurements and Gates by Code Deformation, 2009. J. Phys. A: Math. Theor. 42.

[103] Hendrik Poulsen Nautrup and Nicolai Friis and Hans J. Briegel. Fault-tolerant interface between quantum memories and quantum processors, November 2017. Nature Communications 8, Article number: 1321.

[104] Ilya Dumer and Daniele Micciancio and Madhu Sudan. Hardness of approximating the minimum distance of a linear code, January 2003. IEEE Transactions on Information Theory, 49(1):22-37.

[105] Daniel Gottesman and Lucy Liuxuan Zhang. Fibre bundle framework for unitary quantum fault tolerance, 2013. arXiv:1309.7062.

[106] Sergey Bravyi and Martin Suchara and Alexander Vargo. Efficient Algorithms for Maximum Likelihood Decoding in the Surface Code, 2014. Phys. Rev. A 90, 032326.

[107] Rui Chao and Ben W. Reichardt. Quantum error correction with only two extra qubits, 2017. arXiv:1705.02329.

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

[109] Bryan Eastin and Emanuel Knill. Restrictions on Transversal Encoded Quantum Gate Sets, July 2009. Phys. Rev. Lett. 102, 110502.

[110] A. Honecker and M. Picco and P. Pujol. Nishimori point in the 2D +/- J random-bond Ising model, 2001. Phys. Rev. Lett. 87:047201.