Mitigating Coherent Noise

by

Qingzhong Liang

Department of Mathematics
Duke University

Date: _____

Approved:

_____
Robert Calderbank, Advisor

_____
Kenneth Brown

_____
Iman Marvian

_____
Jianfeng Lu

Dissertation submitted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy
in the Department of Mathematics
in the Graduate School of
Duke University

2023

ABSTRACT

Mitigating Coherent Noise

by

Qingzhong Liang

Department of Mathematics
Duke University

Date: _____

Approved:

_____
Robert Calderbank, Advisor

_____
Kenneth Brown

_____
Iman Marvian

_____
Jianfeng Lu

An abstract of a dissertation submitted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy
in the Department of Mathematics
in the Graduate School of
Duke University

2023

Abstract

Stochastic errors in quantum systems occur randomly but coherent errors may be more damaging since they can accumulate in a particular direction. We develop a framework for designing decoherence free subspaces (DFS), that are unperturbed by coherent noise. We consider a particular form of coherent Z-errors and construct stabilizer codes that form DFS for such noise ("Z-DFS"). More precisely, we develop conditions for transversal $\exp(\imath\theta\sigma_Z)$ to preserve a stabilizer code subspace for all θ. If the code is error-detecting, then this implies a trivial action on the logical qubits. These conditions require the existence of a large number of weight-2 Z-stabilizers, and together, these weight-2 Z-stabilizers generate a direct product of single-parity-check codes.

By adjusting the size of these components, we are able to construct a constant rate family of CSS Z-DFS codes. Invariance under transversal $\exp(\frac{\imath\pi}{2^l}\sigma_Z)$ translates to a trigonometric equation satisfied by $\tan\frac{2\pi}{2^l}$, and for every non-zero X-component of a stabilizer, there is a trigonometric equation that must be satisfied. The Z-stabilizers supported on this non-zero X-component form a classical binary code C, and the trigonometric constraint connects signs of Z-stabilizers to divisibility of weights in $C^\perp$. This construction may be of independent interest to classical coding theorists

who have long been interested in codes C with the property that all weights are divisible by some integer d. If we require that transversal $\exp(\frac{\imath\pi}{2^l}\sigma_Z)$ preserves the code space only up to some finite level l in the Clifford hierarchy, then we can construct higher level gates necessary for universal quantum computation. The aforesaid code C contains a self-dual code and the classical Gleason's theorem constrains its weight enumerator.

The trigonometric conditions corresponding to higher values of l lead to generalizations of Gleason's theorem that may be of independent interest to classical coding theorists. The $[[16, 4, 2]]$ Reed-Muller code and the family of $[[4L^2, 1, 2L]]$ Shor codes are included in our general framework.

Acknowledgements

First of all, I would like to express my deepest gratitude to my advisor Professor Robert Calderbank for the continuous support for my PhD research. He always encouraged me to explore new topics, and use mathematical tools to solve practical problems. Under his guide, I built my understanding and tool-set in the field of quantum error correction from the ground as well as developed my writing and presenting skills.

I would like to extend my sincere thanks to my thesis committee: Professor Kenneth Brown, Professor Iman Marvian, and Professor Jianfeng Lu. I enjoyed discussing problems with them during the weekly QEC seminar. I apprciate all the insightful suggestions provided by them.

I am also thankful to Professor Richard Durrett and Professor Chadmark Schoen for their advice during my early PhD journey. They served as my oral qualifying committee and helped me to adapt the PhD study.

To my wife and my parents, I could not have undertaken this journey without the love and the support from you. Thank you for always being there and making me strong.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Classical Error Correcting Codes

In 1948, Shannon published his landmark paper *A Mathematical Theory of Communication* [Sha48], in which he introduced the idea of encoding various types of information into binary numbers (bits), transmitting them through a noisy channel, and decoding them at the other end of the channel. His idea changed how people view information. Different types of information, such as images and voice, can now be represented as strings of binary digits. The process of digitization is the key to data processing, storage, and transmission of information. For example, when a person takes a photo with a smart phone, the image is stored as numbers, which contain the information about the red, green, and blue scales of each pixel of the photo. If the photo is sent to friends, the cellphone sends the binary numbers instead of the picture itself through a digital communication channel. After other phones receive the string of digits, they use the string to reproduce the same picture. This process sounds natural today, but it was based on the idea in Shannon's paper, which is the cornerstone of information theory.

In reality, the channel used for transmitting information is noisy. That is, for each bit that goes through the channel, there is a probability p that the bit does not carry the original number when it arrives at the receiver's end. Motivated by reducing the damage caused by noise, scientists studied and designed error correcting codes to carry the message through the channel. For example, if a noisy channel has a

probability $p = 0.01$ that flips a bit (i.e changed from 0 to 1, or from 1 to 0), then the receiver gets on average one wrong bit of information for every 100 bits transmitted through the channel, which is considered high since a simple text message may contain thousands of bits. This error model is an example of the binary symmetric channel (BSC).

One strategy that can reduce the error rate is to use the simple code $\mathcal{C} = \{000, 111\}$. If we would like to send a bit x, where $x \in \{0, 1\}$, we first encode it to $xxx \in \mathcal{C}$. In other words, we send 3 bits of repeated information. By using the maximum likelihood decoder, the receiver would decode the message to 0 if one of 000, 001, 010, and 100 was obtained, and to 1 otherwise. As a result, the decoder makes a wrong guess only when two or more bits are flipped and the error rate is reduced from $p = 0.01$ to $3p^2(1-p)+p^3 \approx 0.0003$. We say the code $\mathcal{C}$ has distance 3, which is the distance between the only two codewords 000 and 111. In this example, the cost of reducing the error rate from 0.01 to 0.0003 is to use 3 bits to encode 1 bit of information, and we say the rate of the code $\mathcal{C}$ is $\frac{1}{3}$. This code is called a repetition code because we repeat the message bit.

In general, if we use a classical code $\mathcal{C} \subset \{0, 1\}^n$ with n bits to encode information of k bits, the distance is defined as the minimum distance d between any two distinct codewords in C. We call such a code an $[n, k, d]$ classical code. Each codeword in C can be viewed as a point in the n-dimensional binary space $\{0, 1\}^n$. To visualize the codespace, we consider a n-dimensional ball with a fixed radius R centered at each codeword in C. However, the $|\mathcal{C}|$ balls should satisfy the constraint that no two balls have non-trivial intersection. Under this constraint, we would like the balls to cover as much space as possible (thereby maximizing R). By the definition of code distance d, we have $R_{\max} = \lfloor \frac{d-1}{2} \rfloor$. Now, the decoding rule works as follows: if the codeword received is covered by the ball centered at some $c \in \mathcal{C}$, then the receiver will decode

it to c. As a result, if the noise in the channel causes less than $R_{\max}$ bit-flips, we can recover the correct codework as it does not escape from the ball centered at the correct codeword. In other words, the number of errors it can correct is $R = \lfloor \frac{d-1}{2} \rfloor$. Moreover, if there are more than R but less then d mistakes, we can still detect that the received codeword is different from the original one, but we cannot guarantee that the decoding rule can recover the correct information, since the received codeword may not be in the ball centered at the original codeword.

Shannon showed that there is a upper bound on the code rate for error correcting codes [Sha48]. Many families of error correcting codes with well designed structures were subsequently constructed. The binary Hamming code $[2^m, 2^m - m - 1, 3]$ [Ham50] and the binary Golay code $[23, 12, 7]$ are examples of perfect codes [Gol49]. Every point in the binary space is covered by some ball centered at a codeword of a perfect code, so no space is wasted. Later in 1950 - 1960, many highly symmetric codes, such as Reed-Muller codes [Mul54], were discovered. The Reed-Muller codes provide a family of linear block codes with highly symmetric algebraic structure. An efficient decoding algorithm of Reed-Muller code was then proposed [Ree54] and the code were incorporated in billions of consumer devices. In the 1990s, as computers became more powerful, more and more attention shifted to decoding algorithms, which led to the discovery of polar codes and LDPC (low density parity check codes), and they now dominate coding practice. We will provide more detailed examples about these classical codes in Chapter 2.

In addition to the parameters n, k, and d, the weight distribution also contains important structural information, such as divisibility, of a code. Given a binary linear code $\mathcal{C}$, its dual code is defined as $\mathcal{C}^{\perp} = \left\{ c | c^T c' = 0 \pmod 2 \text{ for all } c' \in \mathcal{C} \right\}$ (i.e the set of codewords that are perpendicular to the original code). The dual code plays an important role in constructing quantum error correcting codes by using classical

codes. It is interesting to note that the weight distribution of the dual code $\mathcal{C}^{\perp}$ of $\mathcal{C}$ is completely determined by the weight distribution of $\mathcal{C}$. This connection was described by the famous MacWilliams identities [Mac63], which open the door to many algebraic methods in classical and quantum information theory. In Chapter 2, we will introduce the MacWilliams identities and dual codes in detail. In later Chapters, we will apply the MacWilliams identities to prove divisibility properties in Quantum error correcting codes.

1.2 Quantum Error Correcting Codes

Just like the bit, which is a fundamental unit of classical information theory, quantum information theory is built on the unit called qubit (quantum bit). While a qubit is a physical object, its definition and properties are easier to be described in mathematics. Like a bit which can be in either state 0 or state 1, two possible states for a qubit are $|0\rangle$ and $|1\rangle$, which are the computational bases denoted in Dirac notation. However, the key difference that distinguishes a qubit from a bit is that a qubit can be in states other than the two base states. More precisely, a qubit can be in any linear combination of the two computational bases:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \tag{1.1}$$

where $\alpha, \beta \in \mathbb{C}$ and $|\alpha|^2 + |\beta|^2 = 1$. In physics, it is called a superposition of $|0\rangle$ and $|1\rangle$. One can measure a qubit in the computational base and the measuring result is either $|0\rangle$, with probability $|\alpha|^2$, or $|1\rangle$ with probability $|\beta|^2$. However, once a qubit is measured, it collapses to the measurement result and no longer contains the probabilistic information.

Similar to the classical error correcting codes, quantum error correction is essential to developing scalable and fault-tolerant quantum computers by constructing error resilient quantum codes. However, there are constraints in the quantum computing. As a result, many methods, such as the repetition codes introduced in the previous section, cannot be directly applied to construct quantum error correcting codes. The three main constraints and challenges are the no-cloning theorem, continuous errors, and collapsing after measurement.

The no-cloning theorem states that we cannot make a copy of an unknown quantum state. The proof is straight forward [NC11]. Assume some unitary matrix U can copy any unknown quantum state $|\psi\rangle$. That is, given the initial state $|\psi\rangle \otimes |s\rangle$, where $|s\rangle$ is some standard pure state, we have $U(|\psi\rangle \otimes |s\rangle) = |\psi\rangle \otimes |\psi\rangle$ for any $|\psi\rangle$. Therefore, for any two states $|\psi\rangle$ and $|\varphi\rangle$, we have $U(|\psi\rangle \otimes |s\rangle) = |\psi\rangle \otimes |\psi\rangle$ and $U(|\varphi\rangle \otimes |s\rangle) = |\varphi\rangle \otimes |\varphi\rangle$. Note that a unitary matrix preserves inner products. By taking the inner product of both sides, we have $\langle\psi|\varphi\rangle = \langle\psi|\varphi\rangle^2$. Note that however, the equation $x = x^2$ has only 0 and 1 as solutions in $\mathbb{R}$. The no-cloning theorem means that we cannot simply create redundant quantum messages and therefore quantum error correcting codes have to be more complicated than repeating quantum bits.

In addition, as a quantum state is a linear combination of the base states, the quantum errors are continuous. The continuity requires infinite precision to determine which error occurred, and methods like parity checking in classical coding cannot be simply reproduced.

Moreover, in classical coding theory, the received classical bits can be observed and then recovered. However, measuring a qubit can destroy the quantum state and therefore make it impossible to recover it to the original state. Therefore, it requires us to develop techniques to detect the syndrome and make the correction without knowing the exact state.

Fortunately, many quantum error correcting codes that overcome these challenges have been developed. Two basic errors in quantum computing are the bit flip error and the phase flip error, which are described by the Pauli matrices

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \text{ and } Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \tag{1.2}$$

respectively. More details about the errors and Pauli matrices will be introduced in Chapter 2. The bit flip quantum code is designed to adjust the bit flip error $X|0\rangle = |1\rangle$ and $X|1\rangle = |0\rangle$. Suppose we have a noisy channel with probability p that a state $|\psi\rangle$ will be affected by the bit flip error and become $X|\psi\rangle$. The bit flip code encodes any state $|\psi\rangle = a|0\rangle + b|1\rangle$ to $|\psi_L\rangle = a|0_L\rangle + b|1_L\rangle = a|000\rangle + b|111\rangle$.

The bit flip code can detect and correct up to one error. Once the receiver has obtained a state $|\phi\rangle$, the following projection operators will be used to perform measurement:

$$\begin{aligned} P_0 &\equiv |000\rangle\langle 000| + |111\rangle\langle 111| \\ P_1 &\equiv |100\rangle\langle 100| + |011\rangle\langle 011| \\ P_2 &\equiv |010\rangle\langle 010| + |101\rangle\langle 101| \\ P_3 &\equiv |001\rangle\langle 001| + |110\rangle\langle 110|. \end{aligned}$$

With the assumption that $|\phi_L\rangle$ has no more than two bit flip errors, we measure symmetry of code state and infer the error. We then have the following syndrome

diagnosis:

$$\langle\phi|P_0|\phi\rangle = 1 \Leftrightarrow \text{ no error}$$
$$\langle\phi|P_1|\phi\rangle = 1 \Leftrightarrow \text{ bit flip on qubit one}$$
$$\langle\phi|P_2|\phi\rangle = 1 \Leftrightarrow \text{ bit flip on qubit two}$$
$$\langle\phi|P_3|\phi\rangle = 1 \Leftrightarrow \text{ bit flip on qubit three.}$$

It is important to note that measuring the syndrome does not leak any information about the state $|\phi\rangle$ and therefore the state $|\phi\rangle$ is protected from being collapsed. The correction is made according to the syndrome we obtained. For example, if we have the syndrome $\langle\phi|P_1|\phi\rangle = 1$, then we will flip the first qubit by applying $X \otimes I \otimes I$ to the received state $|\phi\rangle$. The error rate is reduced from p to $(1-p)^3 + 3p(1-p)^2$, which is the probability that at least two qubits are flipped.

By a similar idea, we can also construct the phase flip code to correct up to one phase flip error Z. The $[[9, 1, 3]]$ Shor code combines the bit flip code and the phase flip code to correct up to one non-trivial Pauli error X, Z, or $Y = \imath XZ$ [CS96].

1.3 Noise and Error Model

In quantum systems, noise can broadly be classified into two types – stochastic and coherent errors. Stochastic errors occur randomly and do not accumulate over time along a particular direction. Coherent errors may be viewed as rotations about a particular axis, and can be more damaging, since they can accumulate coherently over time [IP20]. As quantum computers move out of the lab and become generally programmable, the research community is paying more attention to coherent errors, and especially to the decay in coherence of the effective induced logical chan-

nel [BWG+18, HDF19]. It is natural to consider coherent noise acting *transversally*, where the effect of the noise is to implement a separate unitary on each qubit. Consider, for example, an n-qubit physical system with a uniform background magnetic field acting on the system according to the Hamiltonian $H = Z_1 + Z_2 + \ldots + Z_n$, where Z_i denotes the Pauli Z operator on the i^{th} qubit. Then the effective error is a (unitary) Z-rotation on each qubit by some (small) angle θ, i.e., $\exp(\imath\theta H) = \exp(\imath\theta Z)^{\otimes n}$, where $\imath = \sqrt{-1}$.

While it is possible to address coherent noise through active error correction, it can be more economical to passively mitigate such noise through decoherence free subspaces (DFSs) [KBLW01]. In such schemes, one designs a computational subspace of the full n-qubit Hilbert space which is unperturbed by the noise. In the language of stabilizer codes, we require the noise to preserve the code space, and to act trivially (as the logical identity operator) on the protected information. Inspired by the aforementioned Hamiltonian, which is physically motivated by technologies such as trapped-ion systems, we develop conditions for *all* transversal Z-rotations to preserve the code space of a stabilizer code, i.e., $\exp(i\theta H)\rho\exp(i\theta H)^\dagger = \rho$ for all code states ρ in the stabilizer code. When all angles preserve the code space, the logical action must be trivial for any error-detecting stabilizer code (see Section 4.2). Trigonometric identities for a given transversal Z-rotation in the Clifford hierarchy to preserve the code space of a stabilizer code [RCNP20] were presented in [GC99, CGK17, RCP19]. In our work, by exploiting the celebrated MacWilliams Identities in classical coding theory [Mac63], we develop necessary and sufficient conditions that contain structural information and serve as instructions to construct quantum codes that are oblivious to coherent noise.

1.4 Contributions of this Dissertation

The introduction of magic state distillation by Bravyi and Kitaev [BK05] led to the construction of a sequence of quantum codes, where the code space is preserved by a transversal Z-rotation of the underlying physical space [BK05, Rei05, ACB12, CAB12, BH12, LC13, CH17, HH18, Haa18, KT19, VB19]. The approach in each paper is to examine the action of a transversal Z-rotation on the basis states of a CSS code [CS96, Ste96]. This approach results in *sufficient conditions* for a transversal Z-rotation to realize a logical operation on the code space.

In contrast, we derive *necessary and sufficient conditions* by examining the action of the transversal Z-rotation on the stabilizer group that determines the code. Thus we study the code space by studying the symmetries of the code space. We start from Rengaswamy et al. [RCNP20] which derived trigonometrical conditions for a stabilizer code to be preserved by a transversal $\pi/2^l$ rotation. Note that the condition $l \geq 2$ corresponds to a non-Clifford physical operator.

Our first main contribution (Theorem 3) is a structure theorem that depends on technical arguments which might be of independent interest to classical coding theorists. The structure theorem forces a product structure on a stabilizer code that is oblivious to coherent noise. To state the conditions, we need to introduce some notation in Chapter 2.

A Hermitian Pauli matrix $\pm E(\boldsymbol{a}, \boldsymbol{b})$ is determined by binary vectors $\boldsymbol{a}$ and $\boldsymbol{b}$. The X-component of $\pm E(\boldsymbol{a}, \boldsymbol{b})$ is $\boldsymbol{a}$ and the Z-component is $\boldsymbol{b}$. A stabilizer group $\mathcal{S}$ is generated by r independent commuting Hermitian Pauli matrices, subject to the requirement that if $E(\boldsymbol{a}, \boldsymbol{b}) \in \mathcal{S}$, then $-E(\boldsymbol{a}, \boldsymbol{b}) \notin \mathcal{S}$. The fixed space $\mathcal{V}(\mathcal{S})$ of $\mathcal{S}$ is an $[[n, n-r]]$ stabilizer code. Recall that the Hamming weight $w_H(\boldsymbol{v})$ of a binary vector $\boldsymbol{v}$ is the number of non-zero entries, and that the support $\mathrm{supp}(\boldsymbol{v})$ is the index set of

the non-zero entries. Let $\mathbf{0}$ ($\mathbf{1}$) be the binary vector with every entry 0 (1). Given $\epsilon E(\boldsymbol{a}, \boldsymbol{b}) \in \mathcal{S}$ for some $\epsilon \in \{\pm 1\}$ and $\boldsymbol{a} \neq \mathbf{0}$, define

$$\mathcal{B}(\boldsymbol{a}) \coloneqq \{\boldsymbol{z} \in \mathbb{F}_2^{w_H(a)} : \mathrm{supp}(\boldsymbol{z}) \subseteq \mathrm{supp}(\boldsymbol{a}), \epsilon_{\boldsymbol{z}} E(\mathbf{0}, \boldsymbol{z}) \in \mathcal{S}\} \tag{1.3}$$

$$\text{and } \mathcal{O}(\boldsymbol{a}) \coloneqq \mathbb{F}_2^{w_H(\boldsymbol{a})} \setminus \mathcal{B}(\boldsymbol{a}), \tag{1.4}$$

Remark 1. To simplify notation, we shall sometimes view $\boldsymbol{z}$ as a subset of $\mathrm{supp}(\boldsymbol{a})$, sometimes as a subset of the n qubits, and sometimes as a binary vector either of length $w_H(\boldsymbol{a})$ or of length n (where entries outside $\mathrm{supp}(\boldsymbol{a})$ are set equal to zero). The meaning will be clear from the context.

Remark 2. Here, $\epsilon_{\boldsymbol{v}} \in \{\pm 1\}$ is the sign of $E(0, \boldsymbol{v})$ in the stabilizer group $\mathcal{S}$. Note that the sign $\epsilon_{\boldsymbol{v}}$ of the pure Z-stabilizer $\epsilon_{\boldsymbol{v}} E(\mathbf{0}, \boldsymbol{v})$ takes the form $\epsilon_{\boldsymbol{v}} = (-1)^{\boldsymbol{y}\boldsymbol{v}^T}$ for $\boldsymbol{y} \in \mathbb{F}_2^n$. Also note that vectors from the same coset of $\mathcal{C}_1$ (the group of logical X operators) determine the same signs (since $\mathcal{C}_1$ is perpendicular to $\mathcal{C}_1^{\perp}$, the group of Z stabilizers). It is useful to think of $y \in \mathbb{F}_2^n$ as a fixed vector when we extend signs to Pauli matrices outside the stabilizer group.

A stabilizer code is oblivious to coherent noise if and only if transversal $\pi/2^l$ Z-rotation preserves the code space $\mathcal{V}(\mathcal{S})$ for all $l \geq 2$ (see Section 4.2). We define the support

$$\Gamma = \bigcup_{\epsilon E(\boldsymbol{a}, \boldsymbol{b}) \in \mathcal{S}} \mathrm{supp}(\boldsymbol{a}) \tag{1.5}$$

and a graph with vertex set Γ, where two vertices are joined by an edge if there exists a weight 2 Z-stabilizer in $\mathcal{S}$ involving these two qubits. Let $\Gamma_1, \ldots, \Gamma_t$ be the connected components of this graph and let $|\Gamma_k| = N_k$. The weight 2 Z-stabilizers

supported on Γ_k take the form

$$(-1)^{\boldsymbol{y_k}\boldsymbol{v}^T} E(\mathbf{0}, \boldsymbol{v}) \text{ where } \boldsymbol{y_k} = \boldsymbol{y}\big|_{\Gamma_k}. \tag{1.6}$$

Here $\boldsymbol{y}\big|_{\Gamma_k}$ represents the restriction of $\boldsymbol{y}$ to Γ_k. (In $\boldsymbol{y_k}\boldsymbol{v}^T$, we add zeros to $\boldsymbol{y_k}$ appropriately.) Our main result is

Theorem 3. *Transversal $\pi/2^l$ Z-rotation preserves the stabilizer code for all $l \geq 2$ if and only if for every $\epsilon E(\boldsymbol{a}, \boldsymbol{b}) \in \mathcal{S}$ with $\boldsymbol{a} \neq \mathbf{0}$,*

(1) $\mathrm{supp}(\boldsymbol{a})$ *is the disjoint union of components* $\Gamma_k \subseteq \mathrm{supp}(\boldsymbol{a})$

(2) N_k *is even and* $w_H(\boldsymbol{y_k}) = N_k/2$ *for all* k *such that* $\Gamma_k \subseteq \mathrm{supp}(\boldsymbol{a})$.

Note that for every $\epsilon E(\boldsymbol{a}, \boldsymbol{b}) \in \mathcal{S}$ we have $\boldsymbol{a}\big|_{\Gamma_k} = \mathbf{0}$ or $\mathbf{1}$ for $k = 1, \ldots, t$. Hence Theorem 3 forces a product structure on a stabilizer code that is oblivious to coherent noise. It also provides constraints on the signs of weight 2 Z-stabilizers.

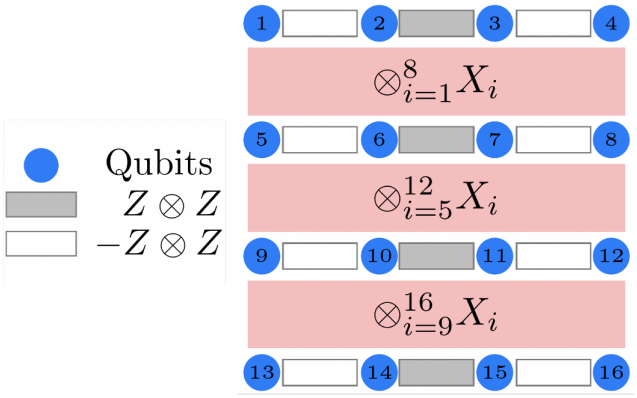


Figure 1.1: The $[[16, 1, 4]]$ Shor code constructed by concatenating the $[[4, 1]]$ bit-flip code and the $[[4, 1]]$ phase-flip code.

Example 1. A set of generators of the $[[16, 1, 4]]$ Shor code is shown in Fig. 1.1, and it follows from Theorem 3 that this code is oblivious to coherent noise. In

Fig. 1.1, the filled circles represent physical qubits, the white (resp. gray filled) squares represent weight-2 Z-stabilizers with negative (resp. positive) sign, and the three large filled rectangles represent weight-8 X-stabilizers. The graph on Γ has four connected components, and the component Γ_k is simply the k-th row of the 4×4 array. Condition (1) is satisfied since every X stabilizer is the sum of an even number of rows. Condition (2) is satisfied since the choice $\boldsymbol{y_k} = [0, 1, 1, 0]$ for $k = 1, 2, 3, 4$ properly accounts for the signs of Z-stabilizers. Observe that $[[16, 1, 4]]$ is also a constant excitation code (defined in Section 2.7). The quotient space $\mathcal{C}_1/\mathcal{C}_2 = \{0, \boldsymbol{w} = (1000) \otimes (1111)\}$, where $\mathcal{C}_2$ defines the X-stabilizers and $\mathcal{C}_1$ defines the logical X operators. Under the general encoding map, the codewords are

$$|\overline{0}\rangle = \frac{1}{2\sqrt{2}} \sum_{\boldsymbol{x} \in \mathcal{C}_2} |\boldsymbol{x} \oplus \boldsymbol{y}\rangle \text{ and } |\overline{1}\rangle = \frac{1}{2\sqrt{2}} \sum_{\boldsymbol{x} \in \mathcal{C}_2} |\boldsymbol{w} \oplus \boldsymbol{x} \oplus \boldsymbol{y}\rangle. \tag{1.7}$$

The restriction of $\boldsymbol{w}$ and $\boldsymbol{x} \in \mathcal{C}_2$ to the k-th row is either $\mathbf{0}$ and $\mathbf{1}$. Since $w_H(\boldsymbol{y_k}) = 2 = \frac{4}{2}$, we have $w_H(\boldsymbol{x} \oplus \boldsymbol{y}) = w_H(\boldsymbol{w} \oplus \boldsymbol{x} \oplus \boldsymbol{y}) = 8$ for all $\boldsymbol{x} \in \mathcal{C}_2$.

We show that a CSS code is oblivious to coherent noise if and only if it is a constant excitation code (Corollary 15). Sufficiency is straightforward since a transversal Z-rotation acts as a global phase. Ouyang [Ouy20, Ouy21] observed that one can construct constant excitation codes by concatenating a stabilizer code with the dual rail code [KLM01]. His original paper was independent of and contemporaneous with our original paper [HLRC21a]. After we shared our results he realized that he could connect his dual rail construction to stabilizer code [Ouy].

Beyond developing conditions, we also construct a linear rate CSS code family with growing distance that possesses this property, thereby acting as a decoherence free subspace for this noise, which is the second main contribution. The conditions we derive lead to a systematic construction of new quantum error correcting codes,

which are oblivious to coherent noise and have increasing distance. Given any even M, and any stabilizer code on t qubits, we construct a product code on Mt qubits that is oblivious to coherent noise. The Mt qubits are partitioned into t blocks of M qubits, with each block supporting a DFS. The product code inherits the distance properties of the initial stabilizer code. In the construction, a product structure with DFS components provides resilience to coherent noise. The cost of forming DFS components for QECC is just scaling the total qubits by an even number. Thus, the minimal cost of becoming oblivious to coherent noise is scaling the number of qubits by 2. The result is remarkable since if we have a familiy of QECC with finite rate and growing distance, then the output QECC family with resilience to coherent noise keeps these good properties.

The necessary and sufficient conditions for a stabilizer code to be oblivious to coherent noise require the product code structure, resulting in a code rate less than 1/2. To relax the restrictions, we can consider stabilizer codes that are preserved by all the transversal Z-rotations through angle $\pi/2^l$ up to some finite integer l, inducing the logical identities. This leads to the third main contribution. More precisely, by relaxing the condition in 3 from all $l \geq 2$ to all $l \leq l_{\max} < \infty$ for some $l_{\max}$, we allow transversal Z-rotations to induce non-identity logical operations on a stabilizer code. Let Z_j denote the classical binary codes formed by the Z-stabilizers supported on a given X-stabilizer. We showed that the weight enumerator of such a code Z_j must satisfy a sequence of constraints, which relate to Gleason's Theorem [Gle71] and field extensions.

Chapter 2

Preliminaries and Notation

2.1 Classical Codes

Let $\mathbb{F}_2 = \{0, 1\}$ denote the binary field. A n-bit binary classical code $\mathcal{C}$ is a subset of the n-dimensional binary space $\mathbb{F}_2^n$. If the subset is a linear subspace, then $\mathcal{C}$ is called a binary linear code. The dimension of C, denoted by k, defines the number of bits it can encode. The Hamming weight of a codeword $\boldsymbol{c}$ is simply the number of nonzero entries. In general, we denote the Hamming weight of a binary vector $\boldsymbol{v}$ by $w_H(\boldsymbol{v})$. The distance of $\mathcal{C}$ is defined as $d = \min\{w_H(\boldsymbol{c}) | \boldsymbol{c} \in \mathcal{C} \setminus \{\mathbf{0}\}\}$, the minimum Hamming weight among nonzero codewords. The Hamming distance between two codewords $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{C}$ is the minimal number of changes needed to transform $\boldsymbol{x}$ to $\boldsymbol{y}$, which can be represented by the Hamming weight $w_H(\boldsymbol{x} \oplus \boldsymbol{y})$. Here, $\oplus$ represents the binary addition. One more concept needs to be introduced before we can consider an example and that is the dual code. The dual code $\mathcal{C}^\perp$ of a code $\mathcal{C}$ is defined as $\mathcal{C}^\perp := \{\boldsymbol{c} | \boldsymbol{c}^T \boldsymbol{c'} = 0 \pmod 2 \text{ for all } \boldsymbol{c'} \in \mathcal{C}\}$.

Example 2 (Reed-Muller Codes, [MS77])**.** Consider a binary linear code $\mathcal{C}$ generated by 11111111, 00001111, 00110011, and 01010101. It is a 4-dimensional subspace of $\{0, 1\}^8$. The minimum Hamming weight in $\mathcal{C}$ is 4. So $\mathcal{C}$ is a $[8, 4, 4]$ code. Note that if we let $\mathbf{1} = 11111111$, $\boldsymbol{x_1} = 00001111$, $\boldsymbol{x_2} = 00110011$, and $\boldsymbol{x_3} = 01010101$, then the codewords in $\mathcal{C}$ can be represented as $f(\boldsymbol{x_1}, \boldsymbol{x_2}, \boldsymbol{x_3})$, where f is a Boolean function which is a polynomial of degree at most 1. This is an example of the more general Reed-Muller structure defined as

Definition 4. *The r-th order binary Reed-Muller code, denoted as $RM(r, m)$ of length $n = 2^m$, for $0 \leq r \leq m$, is the set of all vectors $f(\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots, \boldsymbol{x_m})$, where f is a Boolean function which is a polynomial of degree at most r.*

Following Definition 4, the code $\mathcal{C}$ is actually $RM(1, 3)$. The distance of a r-th order Reed-Muller code of length m ($RM(r, m)$) is 2^{m-r}. It is also important to note that the dual code of a Reed-Muller code is still a Reed-Muller code. In particular, $RM(r, m)^{\perp} = RM(m - r - 1, m)$ for $0 \leq r \leq m - 1$.

2.2 The MacWilliams Identities

Recall that the Hamming weight of a binary codeword $\boldsymbol{c}$ is denoted by $w_H(\boldsymbol{c})$. The weight distribution of code is a frequently used term in coding theory. It is an aggregate item but contains information such as divisibility of a code. The weight distribution of a binary linear code $\mathcal{C} \subset \mathbb{F}_2^m$ is summarized by the weight enumerator, which is the polynomial defined as

$$P_{\mathcal{C}}(x, y) = \sum_{\boldsymbol{v} \in \mathcal{C}} x^{m - w_H(\boldsymbol{v})} y^{w_H(\boldsymbol{v})}. \tag{2.1}$$

There may exist different codes $\mathcal{C}$ and $\mathcal{C}'$ with the same weight distribution. However, it turns out that if $\mathcal{C}$ and $\mathcal{C}'$ have identical weight distributions, so do $\mathcal{C}^{\perp}$ and $\mathcal{C}'^{\perp}$. In other words, the weight distribution of the dual code $\mathcal{C}^{\perp}$ is completely determined by the weight distribution of $\mathcal{C}$. This relation is precisely described by the MacWilliams Identities [Mac63], which relates the weight enumerator of a code $\mathcal{C}$ to that of the dual code $\mathcal{C}^{\perp}$. It is given by

$$P_{\mathcal{C}}(x, y) = \frac{1}{|\mathcal{C}^{\perp}|} P_{\mathcal{C}^{\perp}}(x + y, x - y). \tag{2.2}$$

Example 3. Let $\mathcal{R}_{2n+1}$ be the repetition code of length $2n+1$. So $\mathcal{R}_{2n+1} = \{\mathbf{0}_{2n+1}, \mathbf{1}_{2n+1}\}$, where $\mathbf{0}_{2n+1}$ represents the vector of all zeros and $\mathbf{1}_{2n+1}$ represents the vector of all ones. The weight enumerator of $\mathcal{R}_{2n+1}$ is

$$P_{\mathcal{R}_{2n+1}}(x,y) = x^{2n+1} + y^{2n+1}. \tag{2.3}$$

The dual code of $\mathcal{R}_{2n+1}$ is the subspace of all even-weight vectors (i.e. $\mathcal{R}_{2n+1}^{\perp} = \{\boldsymbol{c} \in \mathbb{F}_2^{2n+1} | w_H(\boldsymbol{c}) \text{ is even } \}$, which is also referred as the single parity check code of length $2n+1$. By directly counting, we have the weight distribution follows $|\{\boldsymbol{c} \in \mathcal{R}_{2n+1}^{\perp} | w_H(\boldsymbol{c}) = 2j\}| = \binom{2n+1}{2j}$ for $0 \le j \le n$. We can verify that

$$\begin{aligned} P_{\mathcal{R}_{2n+1}^{\perp}}(x,y) &= \sum_{j=0}^{n} \binom{2n+1}{2j} x^{n-2j} y^{2j} && (2.4)\\ &= \frac{1}{2}\left((x+y)^{2n+1} + (x-y)^{2n+1}\right) && (2.5)\\ &= \frac{1}{|\mathcal{R}_{2n+1}|} P_{\mathcal{R}_{2n+1}}(x+y, x-y). && (2.6) \end{aligned}$$

In this thesis, we frequently make the substitution $x = \cos\frac{2\pi}{2^l}$ and $y = -\imath \sin\frac{2\pi}{2^l}$, and we define

$$P[\mathcal{C}] \coloneqq P_{\mathcal{C}}\left(\cos\frac{2\pi}{2^l}, -\imath\sin\frac{2\pi}{2^l}\right) = \sum_{\boldsymbol{v}\in\mathcal{C}} \left(\cos\frac{2\pi}{2^l}\right)^{m-w_H(\boldsymbol{v})} \left(-\imath\sin\frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})}. \tag{2.7}$$

2.3 The Pauli Group

Let $N = 2^n$. Any 2×2 Hermitian matrix can be uniquely expressed as a real linear combination of the four single qubit Pauli matrices/operators

$$I_2 \coloneqq \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \; X \coloneqq \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \; Z \coloneqq \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \; Y = \imath XZ, \tag{2.8}$$

where $\imath = \sqrt{-1}$ is the imaginary unit. The operators satisfy $X^2 = Y^2 = Z^2 = I_2$, $XY = -YX$, $XZ = -ZX$, and $YZ = -ZY$.

Let $A \otimes B$ denote the Kronecker product (tensor product) of two matrices A and B. Given vectors $\boldsymbol{a} = [a_1, a_2, \ldots, a_n]$ and $\boldsymbol{b} = [b_1, b_2, \ldots, b_n]$ with $a_i, b_j = 0$ or 1, we define the operators

$$D(\boldsymbol{a}, \boldsymbol{b}) \coloneqq X^{a_1} Z^{b_1} \otimes X^{a_2} Z^{b_2} \otimes \cdots \otimes X^{a_n} Z^{b_n}, \tag{2.9}$$

$$E(\boldsymbol{a}, \boldsymbol{b}) \coloneqq \imath^{\boldsymbol{a}\boldsymbol{b}^T \pmod 4} D(\boldsymbol{a}, \boldsymbol{b}). \tag{2.10}$$

We often abuse notation and write $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{F}_2^n$, though entries of vectors are sometimes interpreted in $\mathbb{Z}_4 = \{0, 1, 2, 3\}$. Note that $D(\boldsymbol{a}, \boldsymbol{b})$ can have order $1, 2$ or 4 (order means the smallest positive integer h such that $D(\boldsymbol{a}, \boldsymbol{b})^h = I_N$), but $E(\boldsymbol{a}, \boldsymbol{b})^2 = \imath^{2\boldsymbol{a}\boldsymbol{b}^T} D(\boldsymbol{a}, \boldsymbol{b})^2 = \imath^{2ab^T}(\imath^{2\boldsymbol{a}\boldsymbol{b}^T} I_N) = I_N$. The n-qubit *Pauli group* is defined as

$$\mathcal{P}_n \coloneqq \{\imath^\kappa D(\boldsymbol{a}, \boldsymbol{b}) : \boldsymbol{a}, \boldsymbol{b} \in \mathbb{F}_2^n, \kappa = 0, 1, 2, 3\}. \tag{2.11}$$

The n-qubit Pauli matrices form an orthonormal basis for the vector space of $N \times N$ complex matrices $\mathbb{C}^{N \times N}$ under the normalized Hilbert-Schmidt inner product $\langle A, B \rangle \coloneqq \mathrm{Tr}(A^\dagger B)/N$.

We will use the *Dirac notation*, $|\cdot\rangle$ to represent the basis states of a single qubit

in $\mathbb{C}^2$. For any $\boldsymbol{v} = [v_1, v_2, \cdots, v_n] \in \mathbb{F}_2^n$, we define $|\boldsymbol{v}\rangle = |v_1\rangle \otimes |v_2\rangle \otimes \cdots \otimes |v_n\rangle$, the standard basis vector in $\mathbb{C}^N$ with 1 in the position indexed by $\boldsymbol{v}$ and 0 elsewhere. We write the Hermitian transpose of $|\boldsymbol{v}\rangle$ as $\langle \boldsymbol{v}| = |\boldsymbol{v}\rangle^\dagger$. We may write an arbitrary n-qubit quantum state as $|\psi\rangle = \sum_{\boldsymbol{v}\in\mathbb{F}_2^n} \alpha_{\boldsymbol{v}} |\boldsymbol{v}\rangle \in \mathbb{C}^N$, where $\alpha_{\boldsymbol{v}} \in \mathbb{C}$ and $\sum_{\boldsymbol{v}\in\mathbb{F}_2^n} |\alpha_{\boldsymbol{v}}|^2 = 1$. The Pauli matrices act on a single qubit as

$$X|0\rangle = |1\rangle, X|1\rangle = |0\rangle, Z|0\rangle = |0\rangle, \text{ and } Z|1\rangle = -|1\rangle. \tag{2.12}$$

The symplectic inner product is $\langle [\boldsymbol{a}, \boldsymbol{b}], [\boldsymbol{c}, \boldsymbol{d}] \rangle_S = \boldsymbol{a}\boldsymbol{d}^T + \boldsymbol{b}\boldsymbol{c}^T \pmod 2$. Since $XZ = -ZX$, we have

$$E(\boldsymbol{a}, \boldsymbol{b})E(\boldsymbol{c}, \boldsymbol{d}) = (-1)^{\langle [\boldsymbol{a},\boldsymbol{b}], [\boldsymbol{c},\boldsymbol{d}] \rangle_S} E(\boldsymbol{c}, \boldsymbol{d})E(\boldsymbol{a}, \boldsymbol{b}). \tag{2.13}$$

2.4 The Clifford Hierarchy

The *Clifford hierarchy* of unitary operators was introduced in [GC99]. The first level of the hierarchy is defined to be the Pauli group $\mathcal{C}^{(1)} = \mathcal{P}_n$. For $l \geq 2$, the levels l are defined recursively as

$$\mathcal{C}^{(l)} := \{U \in \mathbb{U}_N : UE(\boldsymbol{a}, \boldsymbol{b})U^\dagger \in \mathcal{C}^{(l-1)}, \text{ for all } E(\boldsymbol{a}, \boldsymbol{b}) \in \mathcal{P}_n\}, \tag{2.14}$$

where $\mathbb{U}_N$ is the group of $N \times N$ unitary matrices. The second level is the Clifford Group [Got98a], $\mathcal{C}^{(2)}$, which can be generated using the unitaries *Hadamard*, *Phase*, and either of *Controlled-NOT* (CX) or *Controlled-Z* (CZ) defined respectively as

$$H := \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, P := \begin{bmatrix} 1 & 0 \\ 0 & \imath \end{bmatrix}, \tag{2.15}$$

$$\mathrm{C}X_{a\to b} \coloneqq |0\rangle\langle 0|_a \otimes (I_2)_b + |1\rangle\langle 1|_a \otimes X_b,\ \mathrm{C}Z_{a\to b} \coloneqq |0\rangle\langle 0|_a \otimes (I_2)_b + |1\rangle\langle 1|_a \otimes Z_b. \tag{2.16}$$

It is well-known that Clifford unitaries in combination with *any* unitary from a higher level can be used to approximate any unitary operator arbitrarily well [BMP+99]. Hence, they form a universal set of gates for quantum computation. A widely used choice for the non-Clifford unitary is the T gate defined by

$$T := \begin{bmatrix} 1 & 0 \\ 0 & e^{\frac{i\pi}{4}} \end{bmatrix} = \sqrt{P} = Z^{\frac{1}{4}} \equiv \begin{bmatrix} e^{-\frac{i\pi}{8}} & 0 \\ 0 & e^{\frac{i\pi}{8}} \end{bmatrix} = e^{-\frac{i\pi}{8}Z}. \tag{2.17}$$

2.5 Stabilizer Codes

We define a stabilizer group $\mathcal{S}$ to be a commutative subgroup of the Pauli group $\mathcal{P}_n$, where every group element is Hermitian and no group element is $-I_N$. We say $\mathcal{S}$ has dimension r if it can be generated by r independent elements as $\mathcal{S} = \langle \nu_i E(\boldsymbol{c_i}, \boldsymbol{d_i}) : i = 1, 2, \ldots, r\rangle$, where $\nu_i \in \{\pm 1\}$ and $\boldsymbol{c_i}, \boldsymbol{d_i} \in \mathbb{F}_2^n$. Since $\mathcal{S}$ is commutative, we must have $\langle [\boldsymbol{c_i}, \boldsymbol{d_i}], [\boldsymbol{c_j}, \boldsymbol{d_j}]\rangle_S = \boldsymbol{c_i}\boldsymbol{d_j}^T + \boldsymbol{d_i}\boldsymbol{c_j}^T = 0 \pmod 2$.

Given a stabilizer group $\mathcal{S}$, the corresponding *stabilizer code* is the fixed subspace $\mathcal{V}(\mathcal{S}) := \{|\psi\rangle \in \mathbb{C}^N : g|\psi\rangle = |\psi\rangle \text{ for all } g \in \mathcal{S}\}$. We refer to the subspace $\mathcal{V}(\mathcal{S})$ as an $[[n, k, d]]$ stabilizer code because it encodes $k := n - r$ logical qubits into n *physical* qubits. The minimum distance d is defined to be the minimum weight of any operator in $\mathcal{N}_{\mathcal{P}_n}(\mathcal{S}) \setminus \mathcal{S}$. Here, the weight of a Pauli operator is the number of qubits on which it acts non-trivially (i.e., as X, Y or Z), and $\mathcal{N}_{\mathcal{P}_n}(\mathcal{S})$ denotes the normalizer of $\mathcal{S}$ in

$\mathcal{P}_n$ defined by

$$\begin{aligned}\mathcal{N}_{\mathcal{P}_n}(\mathcal{S}) :=& \{i^\kappa E(\boldsymbol{a},\boldsymbol{b}) \in \mathcal{P}_n : E(\boldsymbol{a},\boldsymbol{b})\, E(\boldsymbol{c},\boldsymbol{d})\, E(\boldsymbol{a},\boldsymbol{b}) = \\ & E(\boldsymbol{c}',\boldsymbol{d}') \in \mathcal{S} \text{ for all } \nu E(\boldsymbol{c},\boldsymbol{d}) \in \mathcal{S}, \kappa \in \mathbb{Z}_4\} \\ =& \{i^\kappa E(\boldsymbol{a},\boldsymbol{b}) \in \mathcal{P}_n : E(\boldsymbol{a},\boldsymbol{b})\, E(\boldsymbol{c},\boldsymbol{d})\, E(\boldsymbol{a},\boldsymbol{b}) = \\ & E(\boldsymbol{c},\boldsymbol{d}) \text{ for all } \nu E(\boldsymbol{c},\boldsymbol{d}) \in \mathcal{S}, \kappa \in \mathbb{Z}_4\}. \end{aligned} \tag{2.18}$$

Note that the second equality defines the centralizer of $\mathcal{S}$ in $\mathcal{P}_n$, and it follows from the first since Pauli matrices commute or anti-commute and $-I_N \notin \mathcal{S}$.

For any Hermitian Pauli matrix $E(\boldsymbol{c},\boldsymbol{d})$ and $\nu \in \{\pm 1\}$, the projector $\frac{I_N+\nu E(\boldsymbol{c},\boldsymbol{d})}{2}$ projects on to the ν-eigenspace of $E(\boldsymbol{c},\boldsymbol{d})$. Thus, the projector on to the codespace $\mathcal{V}(\mathcal{S})$ of the stabilizer code defined by $\mathcal{S} = \langle \nu_i E(\boldsymbol{c_i},\boldsymbol{d_i}) : i = 1,2,\ldots,r\rangle$ is

$$\Pi_{\mathcal{S}} = \prod_{i=1}^{r} \frac{(I_N + \nu_i E(\boldsymbol{c_i},\boldsymbol{d_i}))}{2} = \frac{1}{2^r}\sum_{j=1}^{2^r} \epsilon_j E(\boldsymbol{a_j},\boldsymbol{b_j}), \tag{2.19}$$

where $\epsilon_j \in \{\pm 1\}$ is a character of the group $\mathcal{S}$, and is determined by the signs of the generators that produce $E(\boldsymbol{a_j},\boldsymbol{b_j})$: $\epsilon_j E(\boldsymbol{a_j},\boldsymbol{b_j}) = \prod_{t\in J\subset\{1,2,\ldots,r\}} \nu_t E(\boldsymbol{c_t},\boldsymbol{d_t})$ for a unique J.

2.6 CSS Codes

A *CSS (Calderbank-Shor-Steane) code* is a type of stabilizer code with generators that can be separated into strictly X-type and Z-type operators [CS96, Ste96]. Consider two classical binary codes $\mathcal{C}_1, \mathcal{C}_2$ such that $\mathcal{C}_2 \subset \mathcal{C}_1$, and let $\mathcal{C}_1^\perp$, $\mathcal{C}_2^\perp$ denote the dual codes. Note that $\mathcal{C}_1^\perp \subset \mathcal{C}_2^\perp$. Suppose that $\mathcal{C}_2 = \langle \boldsymbol{c_1}, \boldsymbol{c_2}, \ldots, \boldsymbol{c_{k_2}}\rangle$ is an $[n, k_2]$ code and $\mathcal{C}_1^\perp = \langle \boldsymbol{d_1}, \boldsymbol{d_2} \ldots, \boldsymbol{d_{n-k_1}}\rangle$ is an $[n, n-k_1]$ code. Then, the corresponding CSS code

has the stabilizer group

$$\mathcal{S} = \langle \nu_{(\boldsymbol{c_i},\boldsymbol{0})} E\left(\boldsymbol{c_i},\boldsymbol{0}\right), \nu_{(\boldsymbol{0},\boldsymbol{d_j})} E\left(\boldsymbol{0},\boldsymbol{d_j}\right) : i = 1,\ldots,k_2 \text{ and } j = 1,\ldots,n-k_1 \rangle \tag{2.20}$$

$$= \{\epsilon_{(\boldsymbol{a},\boldsymbol{0})}\epsilon_{(\boldsymbol{0},\boldsymbol{b})} E\left(\boldsymbol{a},\boldsymbol{0}\right) E\left(\boldsymbol{0},\boldsymbol{b}\right) : \boldsymbol{a} \in \mathcal{C}_2, \boldsymbol{b} \in \mathcal{C}_1^{\perp}\}, \tag{2.21}$$

where $\nu_{(\boldsymbol{c_i},\boldsymbol{0})}, \nu_{(\boldsymbol{d_j},\boldsymbol{0})}, \epsilon_{(\boldsymbol{a},\boldsymbol{0})}, \epsilon_{(\boldsymbol{0},\boldsymbol{b})} \in \{\pm 1\}$. The CSS code projector can be written as the product:

$$\Pi_{\mathcal{S}} = \Pi_{\mathcal{S}_X} \Pi_{\mathcal{S}_Z}, \tag{2.22}$$

where

$$\Pi_{\mathcal{S}_X} =: \prod_{i=1}^{k_2} \frac{(I_N + \nu_{(\boldsymbol{c_i},\boldsymbol{0})} E(\boldsymbol{c_i},\boldsymbol{0}))}{2} = \frac{\sum_{\boldsymbol{a}\in\mathcal{C}_2} \epsilon_{(\boldsymbol{a},\boldsymbol{0})} E(\boldsymbol{a},\boldsymbol{0})}{|\mathcal{C}_2|}, \tag{2.23}$$

and

$$\Pi_{\mathcal{S}_Z} =: \prod_{j=1}^{n-k_1} \frac{(I_N + \nu_{(\boldsymbol{0},\boldsymbol{d_j})} E(\boldsymbol{0},\boldsymbol{d_j}))}{2} = \frac{\sum_{\boldsymbol{b}\in\mathcal{C}_1^{\perp}} \epsilon_{(\boldsymbol{0},\boldsymbol{b})} E(\boldsymbol{0},\boldsymbol{b})}{|\mathcal{C}_1^{\perp}|}. \tag{2.24}$$

Each projector defines a resolution of the identity. Since our work focuses on Z errors, we state the one for Z type below and the other two types can be defined in similar ways. The Z type errors commute with $\prod_{S_z}$ so we only consider $\prod_{S_x}$. For $\boldsymbol{\mu} \in \mathbb{F}_2^n/\mathcal{C}_2^{\perp}$, we define

$$\mathcal{S}_X(\boldsymbol{\mu}) := \left\{(-1)^{\boldsymbol{a}\boldsymbol{\mu}^T} \epsilon_{(\boldsymbol{a},0)} E(\boldsymbol{a},\boldsymbol{0}) : \boldsymbol{a} \in \mathcal{C}_2\right\} \text{ and } \Pi_{\mathcal{S}_X(\boldsymbol{\mu})} = \frac{1}{|\mathcal{C}_2|} \sum_{\boldsymbol{a}\in\mathcal{C}_2} (-1)^{\boldsymbol{a}\boldsymbol{\mu}^T} \epsilon_{(\boldsymbol{a},\boldsymbol{0})} E(\boldsymbol{a},\boldsymbol{0}). \tag{2.25}$$

Then, we have

$$\Pi_{\mathcal{S}_X(\boldsymbol{\mu})} \Pi_{\mathcal{S}_X(\boldsymbol{\mu}')} = \begin{cases} \Pi_{\mathcal{S}_X(\boldsymbol{\mu})} & \text{if } \boldsymbol{\mu} = \boldsymbol{\mu}', \\ 0 & \text{if } \boldsymbol{\mu} \neq \boldsymbol{\mu}', \end{cases} \quad \text{and} \quad \sum_{\boldsymbol{\mu}\in\mathbb{F}_2^n/\mathcal{C}_2^{\perp}} \Pi_{S_X(\boldsymbol{\mu})} = I_{2^n}. \tag{2.26}$$

The detectable Z errors map the original projector $\prod_{\mathcal{S}_X}$ to $\prod_{\mathcal{S}_{X(\boldsymbol{\mu})}}$ for some $\boldsymbol{\mu} \neq \boldsymbol{0}$,

whereas the undetectable Z errors fix $\prod_{\mathcal{S}_X}$.

If $\mathcal{C}_1$ and $\mathcal{C}_2^\perp$ can correct up to t errors, then S defines an $[[n,k,d]]$ CSS code, $k = k_1 - k_2$, with $d \geq 2t+1$, which we will represent as $\mathrm{CSS}(X, \mathcal{C}_2; Z, \mathcal{C}_1^\perp)$. If G_2 and $G_1^\perp$ are the generator matrices for $\mathcal{C}_2$ and $\mathcal{C}_1^\perp$ respectively, then the $(n-k_1+k_2)\times(2n)$ matrix

$$G_{\mathcal{S}} = \left[\begin{array}{c|c} G_2 & \\ \hline & G_1^\perp \end{array}\right] \tag{2.27}$$

generates $\mathcal{S}$. The codespace defined by the stabilizer group $\mathcal{S}$ is $\mathcal{V}(\mathcal{S}) := \{|\psi\rangle \in \mathbb{C}^N : g|\psi\rangle = |\psi\rangle \text{ for all } g \in \mathcal{S}\}$.

2.7 Encoding Map for CSS Codes

Given an $[[n,k,d]]$ $\mathrm{CSS}(X, \mathcal{C}_2; Z, \mathcal{C}_1^\perp)$ code with all positive signs, let $G_{\mathcal{C}_1/\mathcal{C}_2} \in \mathbb{F}_2^{k\times n}$ be a matrix that generates all coset representatives for $\mathcal{C}_2$ in $\mathcal{C}_1$ (note that the choice of coset representatives is not unique). The canonical encoding map $f : \mathbb{F}_2^k \to \mathcal{V}(\mathcal{S})$ is given by $|\overline{\boldsymbol{v}}\rangle := f(|\boldsymbol{v}\rangle_L) := \frac{1}{\sqrt{|\mathcal{C}_2|}} \sum_{\boldsymbol{x}\in\mathcal{C}_2} |\boldsymbol{v}G_{\mathcal{C}_1/\mathcal{C}_2} \oplus \boldsymbol{x}\rangle$. Changing the signs of stabilizers changes the fixed subspace. Hence we need to modify the encoding map to account for nontrivial signs. Define subspaces $\mathcal{B}$ and $\mathcal{D}$ as below.

$$\begin{array}{cc} \mathcal{C}_1^\perp & \mathcal{B}^\perp \\ | & | \\ \mathcal{B} = \{\boldsymbol{z} \in \mathcal{C}_1^\perp | \epsilon_{\boldsymbol{z}} = 1\} & \mathcal{C}_1 \end{array} \qquad \begin{array}{cc} \mathcal{C}_2 & \mathcal{D}^\perp \\ | & | \\ \mathcal{D} = \{\boldsymbol{x} \in \mathcal{C}_2 | \epsilon_{\boldsymbol{x}} = 1\} & \mathcal{C}_2^\perp \end{array}$$

We capture sign information through character vectors $\boldsymbol{y},\ \boldsymbol{u} \in \mathbb{F}_2^n$ (note that the choice of $\boldsymbol{y}, \boldsymbol{u}$ is unique only up to elements in $\mathcal{C}_1, \mathcal{C}_2^\perp$ respectively) satisfying

$$\mathcal{B} = \mathcal{C}_1^\perp \cap \boldsymbol{y}^\perp, \text{or equivalently, } \mathcal{B}^\perp = \langle \mathcal{C}_1, \boldsymbol{y} \rangle, \tag{2.28}$$

and

$$\mathcal{D} = \mathcal{C}_2 \cap \boldsymbol{u}^{\perp}, \text{or equivalently, } D^{\perp} = \langle \mathcal{C}_2^{\perp}, \boldsymbol{u} \rangle. \tag{2.29}$$

Then, for $\epsilon_{(\boldsymbol{a},\boldsymbol{0})}\epsilon_{(\boldsymbol{0},\boldsymbol{b})} E(\boldsymbol{a},\boldsymbol{0}) E(\boldsymbol{0},\boldsymbol{b}) \in S$, we have $\epsilon_{(\boldsymbol{a},\boldsymbol{0})} = (-1)^{\boldsymbol{a}\boldsymbol{u}^T}$ and $\epsilon_{(\boldsymbol{0},\boldsymbol{b})} = (-1)^{\boldsymbol{b}\boldsymbol{y}^T}$.

The canonical bijective map $f : \mathbb{F}_2^k \to \mathcal{V}(\mathcal{S})$ becomes [HLC22b]

$$|\overline{\boldsymbol{v}}\rangle = f(|\boldsymbol{v}\rangle_L) \coloneqq \frac{1}{\sqrt{|\mathcal{C}_2|}} \sum_{\boldsymbol{x}\in\mathcal{C}_2} (-1)^{\boldsymbol{x}\boldsymbol{u}^T} |\boldsymbol{v}G_{\mathcal{C}_1/\mathcal{C}_2} \oplus \boldsymbol{x} \oplus \boldsymbol{y}\rangle. \tag{2.30}$$

To verify that the image of the encoding map f is in $\mathcal{V}(\mathcal{S})$, we show that for $\epsilon_{(\boldsymbol{a},\boldsymbol{0})}\epsilon_{(\boldsymbol{0},\boldsymbol{b})} E(\boldsymbol{a},\boldsymbol{0}) E(\boldsymbol{0},\boldsymbol{b}) \in \mathcal{S}$ (that is $\boldsymbol{a} \in \mathcal{C}_2$, $\epsilon_{(\boldsymbol{a},\boldsymbol{0})} = (-1)^{\boldsymbol{a}\boldsymbol{u}^T}$, $\boldsymbol{b} \in \mathcal{C}_1^{\perp}$, and $\epsilon_{(\boldsymbol{0},\boldsymbol{b})} = (-1)^{\boldsymbol{b}\boldsymbol{y}^T}$),

$$\begin{aligned}
&\epsilon_{(\boldsymbol{a},\boldsymbol{0})}\epsilon_{(\boldsymbol{0},\boldsymbol{b})} E(\boldsymbol{a},\boldsymbol{0}) E(\boldsymbol{0},\boldsymbol{b}) |\overline{\boldsymbol{v}}\rangle \\
=&\epsilon_{(\boldsymbol{a},\boldsymbol{0})}\epsilon_{(\boldsymbol{0},\boldsymbol{b})} E(\boldsymbol{a},\boldsymbol{0}) E(\boldsymbol{0},\boldsymbol{b}) \frac{1}{\sqrt{|\mathcal{C}_2|}} \sum_{\boldsymbol{x}\in\mathcal{C}_2} (-1)^{\boldsymbol{x}\boldsymbol{u}^T} |\boldsymbol{v}G_{\mathcal{C}_1/\mathcal{C}_2} \oplus \boldsymbol{x} \oplus \boldsymbol{y}\rangle \\
=&\frac{1}{\sqrt{|\mathcal{C}_2|}} \sum_{\boldsymbol{x}\in\mathcal{C}_2} \epsilon_{(\boldsymbol{a},\boldsymbol{0})} (-1)^{\boldsymbol{x}\boldsymbol{u}^T} \epsilon_{(\boldsymbol{0},\boldsymbol{b})} (-1)^{\boldsymbol{b}(\boldsymbol{v}G_{\mathcal{C}_1/\mathcal{C}_2} \oplus \boldsymbol{x} \oplus \boldsymbol{y})^T} |\boldsymbol{v}G_{\mathcal{C}_1/\mathcal{C}_2} \oplus \boldsymbol{a} \oplus \boldsymbol{x} \oplus \boldsymbol{y}\rangle \\
=&\frac{1}{\sqrt{|\mathcal{C}_2|}} \sum_{\boldsymbol{x}\in\mathcal{C}_2} (-1)^{(\boldsymbol{a}\oplus\boldsymbol{x})\boldsymbol{u}^T} |\boldsymbol{v}G_{\mathcal{C}_1/\mathcal{C}_2} \oplus \boldsymbol{a} \oplus \boldsymbol{x} \oplus \boldsymbol{y}\rangle \\
=&|\overline{\boldsymbol{v}}\rangle.
\end{aligned}$$

The CSS code is said to be a *constant excitation code* [ZR97] if, for each fixed $\boldsymbol{v} \in \mathbb{F}_2^k$, the weight $w_H(\boldsymbol{v}G_{\mathcal{C}_1/\mathcal{C}_2} \oplus \boldsymbol{x} \oplus \boldsymbol{y})$ is constant for all $\boldsymbol{x} \in \mathcal{C}_2$. Recall that a common kind of coherent noise is modeled by $U = \exp(\imath\theta Z)^{\otimes n}$ for arbitrary θ. When U acts on a $|0\rangle\&|1\rangle$ computational basis state in a constant excitation code, each term in (2.30) generates the same phase term $\exp(\imath\theta w_H(\boldsymbol{v}G_{\mathcal{C}_1/\mathcal{C}_2} \oplus \boldsymbol{x} \oplus \boldsymbol{y}))$, leading to a global phase, which leaves the state invariant. Hence, a constant excitation code

is oblivious to coherent noise.

2.8 Trigonometric Conditions

The conditions for a code to be preserved by transversal Z gate at a given level of the Clifford Hierarchy (derived in [RCNP20]) are expressed as two trigonometric constraints on weights of pure Z-stabilizers in $\mathcal{S}$.

Theorem 5 (Rengaswamy et al. [RCNP20])**.** *Transversal $\pi/2^l$ Z-rotation ($l \geq 2$) preserves $\mathcal{V}(\mathcal{S})$ if and only if for $\epsilon E(\boldsymbol{a}, \boldsymbol{b}) \in \mathcal{S}$ with $\boldsymbol{a} \neq \mathbf{0}$,*

$$\sum_{\boldsymbol{v} \in \mathcal{B}(\boldsymbol{a})} \epsilon_{\boldsymbol{v}} \left(\imath \tan \frac{2\pi}{2^l} \right)^{w_H(\boldsymbol{v})} = \left(\sec \frac{2\pi}{2^l} \right)^{w_H(\boldsymbol{a})}, \tag{2.31}$$

$$\sum_{\boldsymbol{v} \in \mathcal{B}(\boldsymbol{a})} \epsilon_{\boldsymbol{v}} \left(\imath \tan \frac{2\pi}{2^l} \right)^{w_H(\boldsymbol{v} \oplus \boldsymbol{\omega})} = 0 \quad \textit{for all } \boldsymbol{\omega} \in \mathcal{O}(\boldsymbol{a}). \tag{2.32}$$

Here, $\epsilon_{\boldsymbol{v}} \in \{\pm 1\}$ is the sign of $E(0, \boldsymbol{v})$ in the stabilizer group $\mathcal{S}$, and $\oplus$ denotes the binary (modulo 2) sum of vectors.

These identities provide a mathematical framework that enables us to check whether a quantum code is preserved by transversal Z-rotations. However, it is hard for us to gain structural information of the code from the expression. In the later Chapters, as two main contributions of this thesis, we will build sufficient and necessary conditions that explicitly show the structure of the quantum codes that are oblivious to coherent noise, which results in a systematical construction of coherent-error-free codes starting from any stabilizer code.

Chapter 3

Divisibility of Weights in Binary Codes

3.1 Application of the MacWilliams Identities

The defining property of a divisible linear code [War01] is that codeword weights share a common divisor larger than one. Codes obtained by repeating each coordinate in a shorter code the same number of times are automatically divisible, and they are essentially the only ones for divisors prime to the field size. Examples that are more interesting occur when the divisor is a power of the characteristic. For example, the theorem of Ax [Ax64] governing the existence of zeros of polynomials in several variables characterizes divisibility of weights in Reed-Muller codes [Ax64, McE72, MS77, Bor13].

Divisible codes (in particular Reed-Muller codes) appear in protocols designed for magic state distillation [BK05, ACB12, CAB12, BH12] which achieves universal quantum computation through transversal implementation of Clifford gates and ancillary magic states. Divisibility tests [LC13, VB19] are introduced to ensure that a quantum error correcting code is preserved by a transversal $\pi/2^l$ Z-rotation. We argue in the reverse direction, showing that divisibility of weights is forced by the requirement that the quantum error correcting code is fixed by a transversal gate. We will make repeated use of the following trigonometric identity that is equivalent to code divisibility and may be of independent interest to classical coding theorists.

Lemma 6. *Let $\mathcal{C}$ be a binary linear code with block length m, where all weights are*

even. Let $l \geq 2$. Then,

$$\sum_{\boldsymbol{v}\in\mathcal{C}} \left(\imath \tan\frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})} = \left(\sec\frac{2\pi}{2^l}\right)^m \tag{3.1}$$

if and only if $(m - 2w_H(\boldsymbol{w}))$ is divisible by 2^l for all $\boldsymbol{w} \in \mathcal{C}^\perp$.

Proof. We rewrite (3.1) as

$$P[\mathcal{C}] = \sum_{\boldsymbol{v}\in\mathcal{C}} \left(\cos\frac{2\pi}{2^l}\right)^{m-w_H(\boldsymbol{v})} \left(\imath \sin\frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})} = 1. \tag{3.2}$$

Let $t_+ \coloneqq \cos\frac{2\pi}{2^l} + \imath \sin\frac{2\pi}{2^l}$ and $t_- \coloneqq \cos\frac{2\pi}{2^l} - \imath \sin\frac{2\pi}{2^l}$. After applying the MacWilliams identities, (3.2) becomes

$$\frac{1}{|\mathcal{C}^\perp|} P_{\mathcal{C}^\perp}(t_+, t_-) = 1. \tag{3.3}$$

Since $(\cos\theta + \imath \sin\theta)(\cos\theta - \imath \sin\theta) = 1$ for all θ, we may rewrite (3.3) as

$$\frac{1}{|\mathcal{C}^\perp|} \sum_{\boldsymbol{w}\in\mathcal{C}^\perp} t_+^{m-w_H(\boldsymbol{w})} t_-^{w_H(\boldsymbol{w})} = 1, \tag{3.4}$$

which may be further simplified to

$$\frac{1}{|\mathcal{C}^\perp|} \sum_{\boldsymbol{w}\in\mathcal{C}^\perp} t_+^{m-2w_H(\boldsymbol{w})} = 1. \tag{3.5}$$

Since $\mathbf{1} \in \mathcal{C}^\perp$, the complement of a codeword in $\mathcal{C}^\perp$ is again a codeword in $\mathcal{C}^\perp$, so we may rewrite (3.5) as

$$\frac{1}{|\mathcal{C}^\perp|} \left[\sum_{\boldsymbol{w}\in\mathcal{C}^\perp} t_+^{m-2w_H(\boldsymbol{w})} + \sum_{\boldsymbol{w}\in\mathcal{C}^\perp} t_+^{-(m-2w_H(\boldsymbol{w}))}\right] = 2. \tag{3.6}$$

Since $(\cos\theta + \imath \sin\theta)^n = e^{\imath n\theta}$, for all θ, equation (3.6) reduces to,

$$\frac{1}{|\mathcal{C}^{\perp}|} \sum_{\boldsymbol{w} \in \mathcal{C}^{\perp}} \cos\left(\frac{2\left(m - 2w_H\left(\boldsymbol{w}\right)\right)\pi}{2^l}\right) = 1. \tag{3.7}$$

We observe that equation (3.7) is satisfied if and only if each term contributes 1 to the sum, and this is equivalent to 2^l dividing $m - 2w_H(\boldsymbol{w})$ for all codewords $\boldsymbol{w}$ in $\mathcal{C}^{\perp}$. □

Setting $\mathcal{C} = \mathcal{B}(\boldsymbol{a})$ in the above lemma provides insights into the conditions of Theorem 5.

Chapter 4

Coherent Noise and Z-Stabilizers

4.1 Conditions on Z-Stabilizers

Given two binary vectors $\boldsymbol{x}, \boldsymbol{y}$, we write $\boldsymbol{x} \preceq \boldsymbol{y}$ to mean that the *support* of $\boldsymbol{x}$ is contained in the support of $\boldsymbol{y}$. We define $\boldsymbol{y}|_{\mathrm{supp}(\boldsymbol{x})} \in \mathbb{F}_2^{w_H(\boldsymbol{x})}$ to be the restriction of $\boldsymbol{y}$ to $\mathrm{supp}(\boldsymbol{x})$. Consider the $[[n, n-r]]$ stabilizer code $\mathcal{V}(\mathcal{S})$ determined by the stabilizer group $S = \langle \nu_i E(\boldsymbol{c_i}, \boldsymbol{d_i}) : \nu_i \in \{\pm 1\}, i = 1, \cdots, r \rangle$. Recall that given a stabilizer $\epsilon E(\boldsymbol{a}, \boldsymbol{b})$ with $\boldsymbol{a} \neq \boldsymbol{0}$, we define

$$\mathcal{B}(\boldsymbol{a}) = \{\boldsymbol{z}\big|_{\mathrm{supp}(\boldsymbol{a})} \in \mathbb{F}_2^{w_H(\boldsymbol{a})} : \epsilon_{\boldsymbol{z}} E(\boldsymbol{0}, \boldsymbol{z}) \in \mathcal{S} \text{ and } \boldsymbol{z} \preceq \boldsymbol{a}\} \tag{4.1}$$

and

$$\mathcal{O}(\boldsymbol{a}) = \mathbb{F}_2^{w_H(\boldsymbol{a})} \setminus \mathcal{B}(\boldsymbol{a}) = \{\boldsymbol{\omega} \in \mathbb{F}_2^{w_H(\boldsymbol{a})} : \boldsymbol{\omega} \notin \mathcal{B}(\boldsymbol{a})\}. \tag{4.2}$$

Since $\mathcal{S}$ is commutative, $\boldsymbol{a}\big|_{\mathrm{supp}(\boldsymbol{a})} = \boldsymbol{1} \in \mathcal{B}(\boldsymbol{a})^{\perp}$, and it follows that all weights in $\mathcal{B}(\boldsymbol{a})$ are even.

Example 4. Consider the $[[16, 1, 4]]$ Shor code shown in Figure 1.1. Setting $E(\boldsymbol{a}, \boldsymbol{0}) = \otimes_{i=1}^{8} X_i$, where X_i means Pauli X on the i-th qubit, we have

$$\mathcal{B}(\boldsymbol{a}) = \mathbb{F}_2^2 \otimes \langle [1,1,0,0], [0,1,1,0], [0,0,1,1] \rangle.$$

We now consider Theorem 5 in the special case $l = 2$ (Transversal T). Let

$$s = \sum_{\boldsymbol{v} \in \mathcal{B}(\boldsymbol{a})} \epsilon_{\boldsymbol{v}} \imath^{w_H(\boldsymbol{v})}. \tag{4.3}$$

Since $\tan \frac{\pi}{4} = 1$ and $\sec \frac{\pi}{4} = \sqrt{2}$, we may rewrite (2.31) as

$$s^2 = 2^{w_H(\boldsymbol{a})} = \sum_{\boldsymbol{v}, \boldsymbol{w} \in \mathcal{B}(\boldsymbol{a})} \epsilon_{\boldsymbol{v}} \epsilon_{\boldsymbol{w}} \imath^{w_H(\boldsymbol{v}) + w_H(\boldsymbol{w})} \tag{4.4}$$

$$= \sum_{\boldsymbol{v}, \boldsymbol{w} \in \mathcal{B}(\boldsymbol{a})} \epsilon_{\boldsymbol{v} \oplus \boldsymbol{w}} \imath^{w_H(\boldsymbol{v} \oplus \boldsymbol{w}) + 2\boldsymbol{v}\boldsymbol{w}^T}. \tag{4.5}$$

Changing variables to $\boldsymbol{z} = \boldsymbol{v} \oplus \boldsymbol{w}$ and $\boldsymbol{v}$, we obtain

$$2^{w_H(\boldsymbol{a})} = \sum_{\boldsymbol{z}, \boldsymbol{v} \in \mathcal{B}(\boldsymbol{a})} \epsilon_{\boldsymbol{z}} \imath^{w_H(\boldsymbol{z})} (-1)^{(\boldsymbol{z} \oplus \boldsymbol{v})\boldsymbol{v}^T} \tag{4.6}$$

$$= \sum_{\boldsymbol{z} \in \mathcal{B}(\boldsymbol{a})} \epsilon_{\boldsymbol{z}} \imath^{w_H(\boldsymbol{z})} \sum_{\boldsymbol{v} \in \mathcal{B}(\boldsymbol{a})} (-1)^{\boldsymbol{z}\boldsymbol{v}^T} \tag{4.7}$$

$$= |\mathcal{B}(\boldsymbol{a})| \sum_{\boldsymbol{z} \in \mathcal{B}(\boldsymbol{a}) \cap \mathcal{B}(\boldsymbol{a})^\perp} \epsilon_{\boldsymbol{z}} \imath^{w_H(\boldsymbol{z})}, \tag{4.8}$$

where the second step follows from $\boldsymbol{v}\boldsymbol{v}^T$ is even. Since $2^{w_H(\boldsymbol{a})} = |\mathcal{B}(\boldsymbol{a})| \cdot |\mathcal{B}(\boldsymbol{a})^\perp|$ and $|\mathcal{B}(\boldsymbol{a}) \cap \mathcal{B}(\boldsymbol{a})^\perp| \leq |\mathcal{B}(\boldsymbol{a})^\perp|$, $\mathcal{B}(\boldsymbol{a})^\perp$ is contained in $\mathcal{B}(\boldsymbol{a})$ and so $\mathbf{1} \in \mathcal{B}(\boldsymbol{a})$. Since $\mathcal{B}(\boldsymbol{a})^\perp \subseteq \mathcal{B}(\boldsymbol{a})$, it now follows that $\mathcal{B}(\boldsymbol{a})$ contains a self-dual code. Since

$$|\mathcal{B}(\boldsymbol{a})^\perp| = \sum_{\boldsymbol{z} \in \mathcal{B}(\boldsymbol{a})^\perp} \epsilon_{\boldsymbol{z}} \imath^{w_H(\boldsymbol{z})}, \tag{4.9}$$

we must have $\epsilon_{\boldsymbol{z}} = \imath^{w_H(\boldsymbol{z})}$ for all $\boldsymbol{z} \in \mathcal{B}(\boldsymbol{a})^\perp$.

Remark 7. The above derivation provides the three necessary conditions given in [RCNP20, Theorem 2] that are necessary for a stabilizer code to be preserved by the

transversal T gate.

1. For each $\epsilon E(\boldsymbol{a}, \boldsymbol{b}) \in \mathcal{S}$ with $\boldsymbol{a} \neq \boldsymbol{0}$, the Hamming weight $w_H(\boldsymbol{a})$ is even.

2. For each $\epsilon E(\boldsymbol{a}, \boldsymbol{b}) \in \mathcal{S}$ with $\boldsymbol{a} \neq \boldsymbol{0}$, the binary code $\mathcal{B}(\boldsymbol{a})$ contains an $\left[n = w_H(\boldsymbol{a}), k = \frac{w_H(\boldsymbol{a})}{2}\right]$ self-dual code.

3. For each $\boldsymbol{z} \in \mathcal{B}(\boldsymbol{a})^{\perp}$, the sign of the corresponding stabilizer $E(\boldsymbol{0}, \boldsymbol{z}) \in \mathcal{S}$ is given by $\imath^{w_H(\boldsymbol{z})}$.

Example 5. Consider the $[[16, 4, 2]]$ code that is a member of the $[[2^m, \binom{m}{1}, 2]]$ quantum Reed-Muller (QRM) family constructed in [RCNP20]. It is the CSS$(X, \mathcal{C}_2; Z, \mathcal{C}_1^{\perp})$ code, where $\mathcal{C}_2 = \langle \mathbf{1} \rangle = \text{RM}(0,4) \subset \mathcal{C}_1 = \text{RM}(1,4)$ and $\mathcal{C}_1^{\perp} = \text{RM}(2,4) \subset \mathcal{C}_2^{\perp} = \text{RM}(3,4)$ (see [MS77] for more details of classical Reed-Muller codes). The signs of all stabilizers are positive. We know from [RCNP20, Theorem 19] that the code space is fixed by transversal $\sqrt{T}$ ($\frac{\pi}{2^4}$ Z-rotation), and direct calculation shows that the corresponding logical operator is CCCZ up to some local Pauli corrections. We first verify invariance under transversal T by checking the sufficient conditions given in Remark 7.

The $[[16, 4, 2]]$ code has a single non-zero X-stabilizer $\boldsymbol{a} = \mathbf{1}$, with even weight, and a single subcode $\mathcal{B}(\boldsymbol{a}) = \mathcal{C}_1^{\perp} = \text{RM}(2,4)$. This subcode contains a self-dual code, denoted $\text{RM}(1.5, 4)$, which is generated by $\mathbf{1}$, all the degree one monomials, and half of the degree two monomials, i.e., x_1x_2, x_1x_3, x_1x_4. Since the weights in $\text{RM}(1.5, 4)$ are 0, 4, 8, 12, and 16, we have $\imath^{w_H(\boldsymbol{v})} = 1$ for all $\boldsymbol{v} \in \text{RM}(1.5, 4)$. This matches the signs specified in the definition of the code above. Hence, the $[[16, 4, 2]]$ code satisfies the sufficient conditions for invariance under transversal T. We note that the logical operator induced by transversal T is the identity (obtained by applying CCCZ twice).

Finally, we verify invariance under transversal $\sqrt{T}$ by checking the first of the trigonometric conditions given in Theorem 5. The weight distribution of $\mathrm{RM}(2,4)$ is given by

$$P(x) = 1 + 140x^4 + 448x^6 + 870x^8 + 448x^{10} + 140x^{12} + x^{16}. \tag{4.10}$$

Let $\alpha_4 = \tan\frac{2\pi}{2^4} = \tan\frac{\pi}{8}$. Since $(\sec\theta)^2 = 1 + (\tan\theta)^2$ and $\epsilon_{\boldsymbol{v}} = 1$, for all $\boldsymbol{v} \in \mathcal{B}(\boldsymbol{a})$, we have

$$\begin{aligned}
&\sum_{\boldsymbol{v}\in\mathrm{RM}(2,4)} \epsilon_{\boldsymbol{v}} \left(\imath\alpha_4\right)^{w_H(\boldsymbol{v})} - \left(1+\alpha_4^2\right)^8 \\
&= \left(\imath\alpha_4\right)^0 + 140\left(\imath\alpha_4\right)^4 + 448\left(\imath\alpha_4\right)^6 + 870\left(\imath\alpha_4\right)^8 \\
&\quad + 448\left(\imath\alpha_4\right)^{10} + 140\left(\imath\alpha_4\right)^{12} + \left(\imath\alpha_4\right)^{16} - \left(1+\alpha_4^2\right)^8 \\
&= -8\alpha_4^2(1-\alpha_4)^2(1+\alpha_4)^2(\alpha_4^2+2\alpha_4-1)^2(\alpha_4^2-2\alpha_4-1)^2.
\end{aligned} \tag{4.11}$$

The first trigonometric condition is satisfied since $\alpha_4 = \sqrt{2}-1$ is a root of $x^2+2x-1 = 0$. We verified the second condition directly using MATLAB for each nonzero coset representative in $\mathbb{F}_2^{16}/\mathcal{B}(\boldsymbol{a})$ and it is also implicit in [RCNP20, Theorem 19].

Remark 7 motivates the following extension to Lemma 6.

Corollary 8. *Let $\mathcal{C}$ be a binary linear code with block length m where all codewords have even weight. Suppose that*

$$\sum_{\boldsymbol{v}\in\mathcal{C}} \epsilon_{\boldsymbol{v}} \left(\imath\tan\frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})} = \left(\sec\frac{2\pi}{2^l}\right)^m, \tag{4.12}$$

where $\epsilon : \mathcal{C} \to \{\pm 1\}$ is a character of the additive group $\mathcal{C}$.

1. *If $\epsilon_{\boldsymbol{v}} = 1$ for all $\boldsymbol{v} \in \mathcal{C}$, then 2^l divides $(m - 2w_H(\boldsymbol{w}))$ for all $\boldsymbol{w} \in \mathcal{C}^\perp$.*

2. *If* $\epsilon_{\boldsymbol{v}} \neq 1$ *for all* $\boldsymbol{v} \in \mathcal{C}$*, and if* $\mathcal{B} = \{\boldsymbol{v} \in \mathcal{C} : \epsilon_{\boldsymbol{v}} = 1\}$*, then* 2^l *divides* $(m - 2w_H(\boldsymbol{w}))$ *for all* $\boldsymbol{w} \in \mathcal{B}^\perp \setminus \mathcal{C}^\perp$.

Proof. Part (1) follows from Lemma 6.

To prove part (2), rewrite (4.12) as

$$\begin{aligned} P[\mathcal{B}] - P[\mathcal{C} \setminus \mathcal{B}] =& \sum_{\boldsymbol{v} \in \mathcal{B}} \left(\cos \frac{2\pi}{2^l}\right)^{m - w_H(\boldsymbol{v})} \left(\imath \sin \frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})} \\ & - \sum_{\boldsymbol{v} \in \mathcal{C} \setminus \mathcal{B}} \left(\cos \frac{2\pi}{2^l}\right)^{m - w_H(\boldsymbol{v})} \left(\imath \sin \frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})} \\ =& 1 \end{aligned} \tag{4.13}$$

Recall the notations we used in the proof of Lemma 6 that $t_+ = \cos \frac{2\pi}{2^l} + \imath \sin \frac{2\pi}{2^l}$ and $t_- = \cos \frac{2\pi}{2^l} - \imath \sin \frac{2\pi}{2^l}$. Since $\mathbf{1} \in \mathcal{C}^\perp \subset \mathcal{B}^\perp$, we may apply the MacWilliams Identities to obtain

$$P[\mathcal{B}] + P[\mathcal{C} \setminus \mathcal{B}] = \sum_{\boldsymbol{v} \in \mathcal{C}} \left(\cos \frac{2\pi}{2^l}\right)^{m - w_H(\boldsymbol{v})} \left(\imath \sin \frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})} \tag{4.14}$$

$$= \frac{1}{|\mathcal{C}^\perp|} P_{\mathcal{C}^\perp}(t_+, t_-) \tag{4.15}$$

$$= \frac{1}{|\mathcal{C}^\perp|} \sum_{\boldsymbol{w} \in \mathcal{C}^\perp} \cos\left(\frac{2\left(m - 2w_H(\boldsymbol{w})\right)\pi}{2^l}\right). \tag{4.16}$$

Note that $\mathcal{B} \subset \mathcal{C}$ is a subspace of index 2. Since $|\mathcal{B}^\perp| = 2|\mathcal{C}^\perp|$, we may apply the MacWilliams Identities to $P_{\mathcal{B}}\left(\cos \frac{2\pi}{2^l}, i \sin \frac{2\pi}{2^l}\right)$ and obtain

$$\begin{aligned} P[\mathcal{B}] &= \frac{1}{|\mathcal{B}^\perp|} P_{\mathcal{B}^\perp}(t_+, t_-) \\ &= \frac{1}{2|\mathcal{C}^\perp|} \sum_{\boldsymbol{w} \in \mathcal{B}^\perp} \cos\left(\frac{2\left(m - 2w_H(\boldsymbol{w})\right)\pi}{2^l}\right). \end{aligned} \tag{4.17}$$

Combining equations (4.16) and (4.17) gives

$$\begin{aligned} 1 &= P[\mathcal{B}] - P[\mathcal{C} \setminus \mathcal{B}] = 2P[\mathcal{B}] - (P[\mathcal{B}] + P[\mathcal{C} \setminus \mathcal{B}]) \\ &= \frac{1}{|\mathcal{C}^\perp|} \sum_{\boldsymbol{w} \in \mathcal{B}^\perp \setminus \mathcal{C}^\perp} \cos\left(\frac{2\left(m - 2w_H\left(\boldsymbol{w}\right)\right)\pi}{2^l}\right). \end{aligned} \tag{4.18}$$

We complete the proof by observing that each term in (4.18) must contribute 1 to the sum. □

Remark 9. If $m \neq 0 \pmod{2^l}$, then since $\mathbf{0} \in \mathcal{C}^\perp$, it must be case 2 of Corollary 8 that applies. This is always the case when $2^l > m$. We must have $w_H(\boldsymbol{v}) = m/2$ for all $\boldsymbol{v} \in \mathcal{B}^\perp \setminus \mathcal{C}^\perp$, and we remark that if we expand the MacWilliams Identities using Krawtchouk polynomials [MS77], then we can show that there exist at least $m/2$ codewords in $\mathcal{C}$ with Hamming weight 2.

By setting $\mathcal{C} = \mathcal{B}(\boldsymbol{a})$ in Theorem 5, we see that the scenario $2^l > w_H(\boldsymbol{a})$ applies whenever we require that Theorem 5 holds for all $l \geq 2$. Thus, the observation using Krawtchouk polynomials implies the existence of a large set of weight 2 Z-stabilizers in the code. This motivates the study of stabilizers groups with such structure, which we embark upon next (Section 5), noting that existence is proved in Theorem 3.

4.2 Logical Identity induced by infinite transversal Z-rotations

The goal of constructing quantum codes that are oblivious to coherent noise motivates us to study the conditions for transversal Z rotations to preserve a code space. In this section, we provide a proof of the following result:

Theorem 10. *A stabilizer code is oblivious to coherent noise if and only if transversal $\pi/2^l Z$-rotation preserves the code space $\mathcal{V}(S)$ for all $l \geq 2$.*

Assume S defines an error-detecting code $[[n, n-r, d]]$, i.e., $d \geq 2$, which is invariant under all the transversal $\frac{\pi}{2^l}$ Z-rotations. Set $\theta_l = \frac{\pi}{2^l}$. Then, we can write the Taylor expansion

$$\bigotimes_{i=1}^{n} e^{\imath\theta_l Z_i} = \bigotimes_{i=1}^{n} \sum_{k=0}^{\infty} \frac{(\imath\theta_l Z_i)^k}{k!} = \bigotimes_{i=1}^{n} (I_2 + \imath\theta_l Z_i + \mathcal{O}(\theta_l^2) I_2) \tag{4.19}$$

$$\begin{aligned} &= I_{2^n} + \imath\theta_l (Z_1 \otimes I_2 \otimes \cdots I_2 + I_2 \otimes Z_2 \otimes I_2 \otimes \cdots \otimes I_2 \\ &\qquad + \cdots + I_2 \otimes I_2 \otimes \cdots \otimes Z_n) + \mathcal{O}(\theta_l^2) I_{2^n}. \end{aligned} \tag{4.20}$$

We can choose l large enough (say $l \geq L$ for some positive integer L) in order to ignore the last term,

$$\begin{aligned} &\bigotimes_{i=1}^{n} e^{\imath\theta_l Z_i} \\ &\approx I_{2^n} + \imath\theta_l (Z_1 \otimes I_2 \otimes \cdots I_2 + I_2 \otimes Z_2 \otimes I_2 \otimes \cdots \otimes I_2 \\ &\qquad + \cdots + I_2 \otimes I_2 \otimes \cdots \otimes Z_n). \end{aligned} \tag{4.21}$$

On one hand, since the code can detect any single-qubit error, it can detect any linear combination of them (Theorem 10.2 in [NC11]). Therefore, $\bigotimes_{i=1}^{n} e^{\imath\theta_l Z_i}$ is detectable (i.e., it maps all the codewords outside the codespace or acts trivially on the codespace). On the other hand, $\bigotimes_{i=1}^{n} e^{\imath\theta_l Z_i}$ preserves the code space by assumption. Therefore, $\bigotimes_{i=1}^{n} e^{\imath\theta_l Z_i}$ acts trivally on the codespace, which implies that the logical operator induced by $\bigotimes_{i=1}^{n} e^{\imath\theta_l Z_i}$ is the identity for all $l \geq L$. Note that if the logical operator induced by $\bigotimes_{i=1}^{n} e^{\imath\theta_l Z_i}$ is the identity for larger l, the logical operator induced by $\bigotimes_{i=1}^{n} e^{\imath\theta_l Z_i}$ is also the identity for smaller l via repeated applications.

Therefore, the logical operator induced by $\bigotimes_{i=1}^{n} e^{\imath\theta_l Z_i}$ is the identity for all l.

Chapter 5

Conditions for Quantum Codes to be Oblivious to Coherent Noise

5.1 Weight Two Z-Stabilizers

We begin this section by examining the structure of a stabilizer group $\mathcal{S}$ that contains weight 2 Z-stabilizers. Later in this section we show (in the proof of necessity in Theorem 3) that if a stabilizer code $\mathcal{V}(\mathcal{S})$ is preserved by the transversal $\pi/2^l$ Z-rotation for all $l \geq 2$, then $\mathcal{S}$ contains a large number of weight 2 Z-stabilizers.

Let $\boldsymbol{e_i},\ i = 1, 2, \ldots, n$ be the standard basis of $\mathbb{F}_2^n$. Recall the graph with vertex set

$$\Gamma = \bigcup_{\epsilon E(\boldsymbol{a},\boldsymbol{b}) \in \mathcal{S}} \mathrm{supp}(\boldsymbol{a}), \tag{5.1}$$

where vertices i and j are joined if $\epsilon E(\mathbf{0}, \boldsymbol{e_i} \oplus \boldsymbol{e_j}) \in S$ for some $\epsilon \in \{\pm 1\}$. Recall that we denote the connected components of the graph by $\Gamma_1, \cdots, \Gamma_t$, and set $N_k = |\Gamma_k|$ for $k = 1, 2, \cdots, t$.

Lemma 11. *Each component Γ_k, $k = 1, 2, \cdots, t$ is a complete graph.*

Proof. If a path $r_0, r_1, \cdots, r_j$ connects vertices r_0 and r_j, then r_0 is joined to r_j since

$$\pm E\left(\mathbf{0}, \boldsymbol{e_{r_0}} \oplus \boldsymbol{e_{r_j}}\right) = \prod_{i=0}^{j-1} \left[\pm E\left(\mathbf{0}, \boldsymbol{e_{r_i}} \oplus \boldsymbol{e_{r_{i+1}}}\right)\right].$$

□

This implies that the Z-stabilizers corresponding to Γ_k are given by all length N_k vectors of even weight, i.e., the $[N_k, N_k - 1, 2]$ single parity check code. Henceforth,

we denote the $[m, m-1, 2]$ single parity check code of any length m by $\mathcal{W}$. Theorem 5 forces us to consider all Z-stabilizers $\mathcal{B}(\boldsymbol{a})$ supported on the X-component $\boldsymbol{a}$ of some stabilizer $\epsilon E(\boldsymbol{a}, \boldsymbol{b})$. The next observation shows that $\boldsymbol{a}$ either has full support or no support on a given Γ_k. Together with the above result, this means that each Γ_k either contributes $(N_k - 1)$ dimensions worth of Z-stabilizers or nothing at all to $\mathcal{B}(\boldsymbol{a})$. This suggests that we split the sum that appears in Theorem 5 in terms of smaller sums over the Γ_k's lying within the support of $\boldsymbol{a}$. Indeed, we are building up towards such an argument in Theorem 3.

Given $\boldsymbol{v} \in \mathbb{F}_2^n$, let $\boldsymbol{v_k} = \boldsymbol{v}\big|_{\Gamma_k} \in \mathbb{F}_2^{N_k}$ be the restriction of $\boldsymbol{v}$ to Γ_k for $k = 1, \cdots, t$.

Lemma 12. *If $\pm E(\boldsymbol{a}, \boldsymbol{b})$ is a stabilizer in S, then $\boldsymbol{a_k} = \mathbf{0}$ or $\mathbf{1}$.*

Proof. If $\boldsymbol{z_k}$ is an even weight vector supported on Γ_k, then $\pm E(\mathbf{0}, \tilde{\boldsymbol{z_k}})$ is a Z-stabilizer in $\mathcal{S}$. Since $\mathcal{S}$ is commutative, $\boldsymbol{a_k}$ is orthogonal to every even weight vector $\boldsymbol{z_k}$, and so $\boldsymbol{a_k} = \mathbf{0}$ or $\mathbf{1}$. □

5.2 Product Structure

The Z-stabilizers supported on Γ_k take the form $(-1)^{\boldsymbol{y_k v}^T} E(\mathbf{0}, \boldsymbol{v})$, where $\boldsymbol{v}$ is a vector of even weight supported on Γ_k. Here $\boldsymbol{y}_k$ is a fixed binary vector supported on Γ_k. We now investigate trigonometric identities satisfied by the weights in these component codes $\mathcal{W}$ representing Z-stabilizers from Γ_k.

Lemma 13. *Let $\mathcal{W}$ be the $[m, m-1]$ code consisting of all vectors with even weight, and let $\epsilon_{\boldsymbol{v}} = (-1)^{\boldsymbol{vy}^T}$ be a character on $\mathcal{W}$. Then*

$$\sum_{\boldsymbol{v} \in \mathcal{W}} \epsilon_{\boldsymbol{v}} \left(\imath \tan \frac{2\pi}{2^l} \right)^{w_H(\boldsymbol{v})} = \cos\gamma \cdot \left(\sec \frac{2\pi}{2^l} \right)^m, \tag{5.2}$$

where $\gamma = \frac{2\pi(M-2w_H(\boldsymbol{y}))}{2^l}$ *and* $l \geq 3$.

Proof. If ϵ is the trivial character, then $\boldsymbol{y} = \mathbf{0}$, and we have

$$\frac{\sum_{\boldsymbol{v}\in\mathcal{W}} \left(\imath \tan\frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})}}{\left(\sec\frac{2\pi}{2^l}\right)^m} = P\left[\mathcal{W}\right]. \tag{5.3}$$

Note that $\mathcal{W}^\perp = \{\mathbf{0}, \mathbf{1}\}$. We apply the MacWilliams Identities to obtain

$$\begin{aligned} P\left[\mathcal{W}\right] &= \frac{1}{|\mathcal{W}^\perp|} P_{\mathcal{W}^\perp}\left(\cos\frac{2\pi}{2^l} + \imath\sin\frac{2\pi}{2^l}, \cos\frac{2\pi}{2^l} - \imath\sin\frac{2\pi}{2^l}\right) \\ &= \frac{1}{|\mathcal{W}^\perp|} P_{\mathcal{W}^\perp}\left(e^{\imath\frac{2\pi}{2^l}}, e^{-\imath\frac{2\pi}{2^l}}\right) \\ &= \frac{1}{2}\left[\left(e^{\imath\frac{2\pi}{2^l}}\right)^m \left(e^{-\imath\frac{2\pi}{2^l}}\right)^0 + \left(e^{\imath\frac{2\pi}{2^l}}\right)^0 \left(e^{-\imath\frac{2\pi}{2^l}}\right)^m\right] \\ &= \cos\frac{2\pi m}{2^l}, \end{aligned} \tag{5.4}$$

which means

$$\sum_{\boldsymbol{v}\in\mathcal{W}} \left(\imath\tan\frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})} = \cos\frac{2\pi M}{2^l}\left(\sec\frac{2\pi}{2^l}\right)^m. \tag{5.5}$$

If ϵ is a non-trivial character, then there exists $\boldsymbol{y} \in \mathbb{F}_2^m$ with $\boldsymbol{y} \neq \mathbf{0}$ or $\mathbf{1}$ such that

$$\mathcal{B} = \{\boldsymbol{v} \in W : \epsilon_{\boldsymbol{v}} = 1\} = \langle \mathbf{1}, \boldsymbol{y}\rangle^\perp, \tag{5.6}$$

and

$$\mathcal{B}^\perp = \langle \mathbf{1}, \boldsymbol{y}\rangle = \{\mathbf{0}, \mathbf{1}, \boldsymbol{y}, \mathbf{1}\oplus\boldsymbol{y}\}. \tag{5.7}$$

Note that $|\mathcal{B}| = \frac{|\mathcal{W}|}{2}$ and $|\mathcal{B}^{\perp}| = 2|\mathcal{W}^{\perp}|$. We rewrite

$$\sum_{\boldsymbol{v}\in\mathcal{W}} \epsilon_{\boldsymbol{v}} \left(\imath \tan\frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})} = \sum_{\boldsymbol{v}\in\mathcal{B}} \left(\imath \tan\frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})} - \sum_{\boldsymbol{v}\in\mathcal{W}\setminus\mathcal{B}} \left(\imath \tan\frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})} \tag{5.8}$$

$$= 2\sum_{\boldsymbol{v}\in\mathcal{B}} \left(\imath \tan\frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})} - \sum_{\boldsymbol{v}\in\mathcal{W}} \left(\imath \tan\frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})}, \tag{5.9}$$

so that

$$\frac{\sum_{\boldsymbol{v}\in\mathcal{W}} \epsilon_{\boldsymbol{v}} \left(\imath \tan\frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})}}{\left(\sec\frac{2\pi}{2^l}\right)^{m}} = 2P\left[\mathcal{B}\right] - P\left[\mathcal{W}\right]. \tag{5.10}$$

We apply the MacWilliams Identities to obtain

$$P\left[\mathcal{B}\right] = \frac{1}{|\mathcal{B}^{\perp}|} P_{\mathcal{B}^{\perp}}\left(e^{\imath\frac{2\pi}{2^l}}, e^{-\imath\frac{2\pi}{2^l}}\right) = \frac{1}{2}\left[\cos\frac{2\pi m}{2^l} + \cos\frac{2\pi(m - 2w_H(\boldsymbol{y}))}{2^l}\right]. \tag{5.11}$$

We combine with (5.5) to obtain

$$2P\left[\mathcal{B}\right] - P\left[\mathcal{W}\right] = \cos\frac{2\pi\left(m - 2w_H\left(\boldsymbol{y}\right)\right)}{2^l} \tag{5.12}$$

as required. □

When $\mathcal{B}(\boldsymbol{a}) = \mathcal{W}$, the second trigonometric identity in Theorem 5 becomes a sum over all odd weight vectors $(\mathbb{F}_2^m \setminus \mathcal{W})$. The character ϵ is given by $\epsilon_{\boldsymbol{v}} = (-1)^{\boldsymbol{v}\boldsymbol{y}^T}$ for some $\boldsymbol{y} \in \mathbb{F}_2^m$ and we extend the domain of ϵ from $\mathcal{W}$ to $\mathbb{F}_2^m$. If ϵ is trivial, then

$$\frac{\sum_{\boldsymbol{v}\in\mathbb{F}_2^m\setminus\mathcal{W}} \epsilon_{\boldsymbol{v}} \left(\imath \tan\frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})}}{\left(\sec\frac{2\pi}{2^l}\right)^{m}} = P\left[\mathbb{F}_2^m \setminus \mathcal{W}\right] = P\left[\mathbb{F}_2^m\right] - P\left[\mathcal{W}\right]. \tag{5.13}$$

Note that $(\mathbb{F}_2^m)^\perp = \langle \mathbf{0} \rangle$. We apply the MacWilliams Identities to obtain

$$P\left[\mathbb{F}_2^m\right] = P_{\langle \mathbf{0} \rangle}\left(e^{\imath \frac{2\pi}{2^l}}, e^{-\imath \frac{2\pi}{2^l}}\right) \tag{5.14}$$

$$= \left(e^{\imath \frac{2\pi}{2^l}}\right)^{m-0} \left(e^{\imath \frac{2\pi}{2^l}}\right)^{0} \tag{5.15}$$

$$= \cos \frac{2\pi m}{2^l} + \imath \sin \frac{2\pi m}{2^l}. \tag{5.16}$$

It now follows from equation (5.5) that

$$P\left[\mathbb{F}_2^m\right] - P\left[\mathcal{W}\right] = \imath \sin \frac{2\pi m}{2^l} = \imath \sin \frac{2\pi \left(m - 2w_H\left(\mathbf{0}\right)\right)}{2^l}. \tag{5.17}$$

If ϵ is non-trivial, let $\mathcal{B}' = \{x \in \mathbb{F}_2^m | \epsilon_x = 1\}$. If $\mathcal{B}' = \mathcal{W}$, then

$$\frac{\sum_{\boldsymbol{v} \in \mathbb{F}_2^m \setminus \mathcal{W}} \epsilon_v \left(\imath \tan \frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})}}{\left(\sec \frac{2\pi}{2^l}\right)^m} = -\imath \sin \frac{2\pi m}{2^l} = \imath \sin \frac{2\pi(m - 2w_H(\mathbf{1}))}{2^l}. \tag{5.18}$$

Note that since $\langle \boldsymbol{y} \rangle \subseteq \langle \mathbf{1}, \boldsymbol{y} \rangle = \mathcal{B}^\perp$, we have $B \subseteq \boldsymbol{y}^\perp$. It remains to consider the case where ϵ is non-trivial and $\mathcal{B}' \neq \mathcal{W}$. Here $\mathcal{B}' = \boldsymbol{y}^\perp$ where $\boldsymbol{y} \neq \mathbf{1}$.

Lemma 14. *Let $\mathcal{W}$ be the $[m, m-1]$ code consisting of all vectors with even weight. Let $\epsilon_{\boldsymbol{v}} = (-1)^{\boldsymbol{v}\boldsymbol{y}^T}$, let $\mathcal{B} = \{\boldsymbol{v} \in \mathcal{W} | \epsilon_{\boldsymbol{v}} = 1\} = \langle \mathbf{1}, \boldsymbol{y} \rangle^\perp$, and let $\mathcal{B}' = \{\boldsymbol{x} \in \mathbb{F}_2^m | \epsilon_{\boldsymbol{x}} = 1\}$. Then*

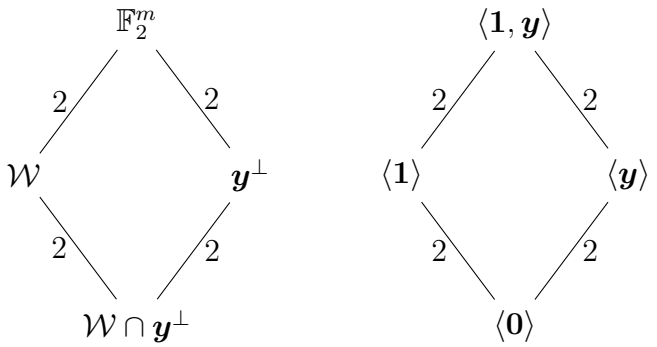
$$\sum_{\boldsymbol{v} \in \mathbb{F}_2^m \setminus \mathcal{W}} \epsilon_{\boldsymbol{v}} \left(\imath \tan \frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})} = \imath \sin \gamma \cdot \left(\sec \frac{2\pi}{2^l}\right)^m, \tag{5.19}$$

where $\gamma = \frac{2\pi(m - 2w_H(\boldsymbol{y}))}{2^l}$.

Proof. We may assume that $\boldsymbol{y} \neq \mathbf{0}, \mathbf{1}$, and that the subspaces $\mathcal{W}, \boldsymbol{y}^\perp$ and their duals $\langle \mathbf{1} \rangle$, $\langle \boldsymbol{y} \rangle$ intersect as shown below. The edge label is the index of the smaller subspace in the group larger subspace.

Table 5.1: Sign patterns: the entries of each row specify how the set corresponding to the subsets A can be written as a union of subsets in different columns.

A \ T	$(\mathbb{F}_2^m \setminus \mathcal{W}) \cap (\mathbb{F}_2^m \setminus \boldsymbol{y}^\perp)$	$(\mathbb{F}_2^m \setminus \mathcal{W}) \cap \boldsymbol{y}^\perp$	$\mathcal{W} \cap (\mathbb{F}_2^m \setminus \boldsymbol{y}^\perp)$
$\mathbb{F}_2^m \setminus \mathcal{W}$	$+$	$+$	0
$\mathbb{F}_2^m \setminus \boldsymbol{y}^\perp$	$+$	0	$+$
$W \setminus (\mathcal{W} \cap \boldsymbol{y}^\perp)$	0	0	$+$



We have

$$\frac{\sum_{\boldsymbol{v} \in \mathbb{F}_2^m \setminus \mathcal{W}} \epsilon_{\boldsymbol{v}} \left(\imath \tan \frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})}}{\left(\sec \frac{2\pi}{2^l}\right)^m}$$
$$= P\left[(\mathbb{F}_2^m \setminus \mathcal{W}) \cap \boldsymbol{y}^\perp\right] - P\left[(\mathbb{F}_2^m \setminus \mathcal{W}) \cap \left(\mathbb{F}_2^m \setminus \boldsymbol{y}^\perp\right)\right]. \tag{5.20}$$

Table 5.1 specifies how subsets T appearing (5.20) can be expressed as disjoint unions of subsets A that appear in the MacWilliams Identities.

It follows from Table 5.1 that we may rewrite the right hand side of (5.20) as

$$\frac{\sum_{\boldsymbol{v} \in \mathbb{F}_2^m \setminus \mathcal{W}} \epsilon_{\boldsymbol{v}} \left(\imath \tan \frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})}}{\left(\sec \frac{2\pi}{2^l}\right)^m}$$
$$= P\left[\mathbb{F}_2^m \setminus \mathcal{W}\right] - 2P\left[\mathbb{F}_2^m \setminus \boldsymbol{y}^\perp\right] + 2P\left[\mathcal{W} \setminus (\mathcal{W} \cap \boldsymbol{y}^\perp)\right]. \tag{5.21}$$

It follows from (5.17) that

$$P\left[\mathbb{F}_2^m \setminus \mathcal{W}\right] = \imath \sin \frac{2\pi m}{2^l}. \tag{5.22}$$

We rewrite (5.16) as

$$P\left[\mathbb{F}_2^m \setminus \boldsymbol{y}^\perp\right] = e^{\imath \frac{2\pi m}{2^l}} - P[y^\perp]. \tag{5.23}$$

Recall that we define $t_+ = \cos\frac{2\pi}{2^l} + \imath \sin\frac{2\pi}{2^l}$ and $t_- = \cos\frac{2\pi}{2^l} - \imath \sin\frac{2\pi}{2^l}$. We apply the MacWilliams Identities to obtain

$$\begin{aligned} P\left[y^\perp\right] &= \frac{1}{|\langle \boldsymbol{y} \rangle|} P_{|\langle \boldsymbol{y} \rangle|}\left(t_+, t_-\right) \\ &= \frac{1}{2}\left(e^{\imath \frac{2\pi m}{2^l}} + e^{\imath \frac{2\pi (m - 2w_H(\boldsymbol{y}))}{2^l}}\right), \end{aligned} \tag{5.24}$$

so that

$$P\left[\mathbb{F}_2^m \setminus \boldsymbol{y}^\perp\right] = \frac{1}{2}\left(e^{\imath \frac{2\pi m}{2^l}} - e^{\imath \frac{2\pi (m - 2w_H(\boldsymbol{y}))}{2^l}}\right). \tag{5.25}$$

It follows from (5.5) that

$$P\left[\mathcal{W} \setminus (\mathcal{W} \cap \boldsymbol{y}^\perp)\right] = \cos \frac{2\pi m}{2^l} - P[\mathcal{W} \cap \boldsymbol{y}^\perp]. \tag{5.26}$$

We apply the MacWilliams Identities to obtain

$$\begin{aligned} &P\left[\mathcal{W} \cap \boldsymbol{y}^\perp\right] \\ &= \frac{1}{|\langle \mathbf{1}, \boldsymbol{y} \rangle|} P_{|\langle \mathbf{1}, \boldsymbol{y} \rangle|}\left(t_+, t_-\right) \\ &= \frac{1}{4}\left[e^{\imath \frac{2\pi m}{2^l}} + e^{-\imath \frac{2\pi m}{2^l}} + e^{\imath \frac{2\pi (m - 2w_H(\boldsymbol{y}))}{2^l}} + e^{\imath \frac{2\pi (2w_H(\boldsymbol{y}) - m)}{2^l}}\right] \end{aligned} \tag{5.27}$$

so that

$$P\left[\mathcal{W} \setminus (\mathcal{W} \cap \boldsymbol{y}^{\perp})\right] = \frac{1}{2}\left[\cos\frac{2\pi m}{2^l} - \cos\frac{2\pi(m - 2w_H(\boldsymbol{y}))}{2^l}\right]. \tag{5.28}$$

We now use (5.22), (5.25), (5.28) to rewrite the right hand side of (5.21) as

$$\imath \sin\frac{2\pi m}{2^l} - e^{\imath\frac{2\pi m}{2^l}} + e^{\imath\frac{2\pi(m-2w_H(\boldsymbol{y}))}{2^l}} + \cos\frac{2\pi m}{2^l} - \cos\frac{2\pi(m - 2w_H(\boldsymbol{y}))}{2^l}, \tag{5.29}$$

which reduces to (5.19). □

5.3 Proof of Conditions

We now consider a stabilizer code $\mathcal{V}(\mathcal{S})$ that is preserved by $\pi/2^l$ Z-rotation for all $l \geq 2$. The sign $\epsilon_{\boldsymbol{v}}$ of the Z-stabilizer $\epsilon_{\boldsymbol{v}} E(\boldsymbol{0}, \boldsymbol{v})$ is given by $\epsilon_{\boldsymbol{v}} = (-1)^{\boldsymbol{y}\boldsymbol{v}^T}$, and we let $\boldsymbol{y_k} = \boldsymbol{y}\big|_{\Gamma_k}$ be the restriction of the binary vector $\boldsymbol{y}$ to Γ_k. Given $\epsilon E(\boldsymbol{a}, \boldsymbol{b}) \in \mathcal{S}$ with $\boldsymbol{a} \neq \boldsymbol{0}$, we now investigate the trigonometric conditions satisfied by Z-stabilizers supported on $\mathrm{supp}(\boldsymbol{a})$. We first show that $\mathrm{supp}(\boldsymbol{a})$ is the disjoint union of components $\Gamma_k \subseteq \mathrm{supp}(\boldsymbol{a})$. We then glue together the trigonometric conditions satisfied by the Z-stabilizers supported on these components Γ_k.

Theorem 3. *Transversal $\pi/2^l$ Z-rotation preserves the stabilizer code for all $l \geq 2$ if and only if for every $\epsilon E(\boldsymbol{a}, \boldsymbol{b}) \in \mathcal{S}$ with $\boldsymbol{a} \neq \boldsymbol{0}$,*

(1) $\mathrm{supp}(\boldsymbol{a})$ *is the disjoint union of components* $\Gamma_k \subseteq \mathrm{supp}(\boldsymbol{a})$

(2) N_k *is even and* $w_H(\boldsymbol{y_k}) = N_k/2$ *for all* k *such that* $\Gamma_k \subseteq \mathrm{supp}(\boldsymbol{a})$.

Proof of Necessity. First, we need to show that the hypothesis implies the presence of many weight 2 Z-stabilizers, and hence that the discussion of Γ_k is material. Though we remarked on their presence in Remark 9, we will see in this proof that such a

structure is revealed by the trigonometric conditions in Theorem 5 itself. For now, we begin by assuming their presence and introducing related quantities.

We divide the weight 2 Z-stabilizers in Γ_k into two classes of sizes P_k and Q_k where $P_k = |\{\boldsymbol{v} \in \mathbb{F}_2^{|\Gamma_k|} : w_H(\boldsymbol{v}) = 2 \text{ and } \epsilon_{\boldsymbol{v}} = 1\}|$ and $Q_k = |\{\boldsymbol{v} \in \mathbb{F}_2^{|\Gamma_k|} : w_H(\boldsymbol{v}) = 2 \text{ and } \epsilon_{\boldsymbol{v}} = -1\}|$. Setting $w_H(\boldsymbol{y_k}) = s$, we have

$$Q_k - P_k = \binom{s}{1}\binom{N_k - s}{1} - \left(\binom{s}{2} + \binom{N_k - s}{2}\right) \tag{5.30}$$

$$= -2\left(s - \frac{N_k}{2}\right)^2 + \frac{N_k}{2}. \tag{5.31}$$

Thus, $Q_k - P_k \leq \frac{N_k}{2}$, and equality holds if and only if $w_H(\boldsymbol{y_k}) = \frac{N_k}{2}$. Theorem 5 implies all $w_H(\boldsymbol{a})$ are even and

$$\sum_{\boldsymbol{v} \in \mathcal{B}(\boldsymbol{a})} \epsilon_{\boldsymbol{v}} \left(\imath \tan\theta\right)^{w_H(\boldsymbol{v})} = \left(\sec\theta\right)^{w_H(\boldsymbol{a})} = \left(1 + (\tan\theta)^2\right)^{\frac{w_H(\boldsymbol{a})}{2}} \tag{5.32}$$

for all $\theta = \frac{\pi}{2^l}$ with $l \geq 2$. Let $\mathcal{B}_{2j}(\boldsymbol{a}) = \{\boldsymbol{z} \in \mathcal{B}(\boldsymbol{a}) | w_H(\boldsymbol{z}) = 2j\}$. We have

$$\sum_{j=0}^{\frac{w_H(\boldsymbol{a})}{2}} \sum_{\boldsymbol{v} \in \mathcal{B}_{2j}(\boldsymbol{a})} \epsilon_{\boldsymbol{v}} (-1)^j \left(\tan\theta\right)^{2j} = \left(1 + (\tan\theta)^2\right)^{\frac{w_H(\boldsymbol{a})}{2}}. \tag{5.33}$$

for all $\theta = \frac{\pi}{2^l}$ with $l \geq 2$. Since a finite degree polynomial (in $(\tan\theta)^2$) cannot have infinitely many roots $\left(\tan\frac{\pi}{2^l}\right)^2$, it must be identically zero and we may equate the coefficients of $(\tan\theta)^2$ to obtain

$$\frac{w_H(\boldsymbol{a})}{2} = \sum_{\boldsymbol{v} \in \mathcal{B}_2(\boldsymbol{a})} \epsilon_{\boldsymbol{v}} \cdot (-1) = \sum_{k : \Gamma_k \subseteq \mathrm{supp}(\boldsymbol{a})} (Q_k - P_k). \tag{5.34}$$

Note that this observation has established the presence of weight 2 vectors in $\mathcal{B}(\boldsymbol{a})$,

as we intended. It follows from (5.31) that

$$\frac{w_H(\boldsymbol{a})}{2} \leq \sum_{k:\Gamma_k \subseteq \mathrm{supp}(\boldsymbol{a})} \frac{N_k}{2} \leq \frac{w_H(\boldsymbol{a})}{2}. \tag{5.35}$$

Therefore equality holds in (5.35) and $Q_k - P_k = \frac{N_k}{2}$ for all k such that $\Gamma_k \subseteq \mathrm{supp}(\boldsymbol{a})$, which completes the proof of necessity.

Proof of Sufficiency. Let $\mathcal{W}_k^0$ be the $[N_k, N_k - 1]$ single-parity-check code and let $\mathcal{W}_k^1 = \mathbb{F}_2^{N_k} \setminus \mathcal{W}_k^0$. Let $\mathcal{W}(\boldsymbol{r}) = \bigoplus_{k:\Gamma_k \subseteq \mathrm{supp}(\boldsymbol{a})} \mathcal{W}_k^{r_k}$, where $\boldsymbol{r} \in \mathbb{F}_2^{|\{k:\Gamma_k \subseteq \mathrm{supp}(\boldsymbol{a})\}|}$ and r_k is the entry of $\boldsymbol{r}$ corresponding to Γ_k. Then, for all $\boldsymbol{r}$,

$$\sum_{\boldsymbol{v} \in \mathcal{W}(\boldsymbol{r})} \epsilon_{\boldsymbol{v}} \left(\imath \tan \frac{2\pi}{2^l} \right)^{w_H(\boldsymbol{v})} = \prod_{\substack{k \\ \Gamma_k \subseteq \mathrm{supp}(\boldsymbol{a})}} f_k(r_k), \tag{5.36}$$

where

$$f_k(\delta) = \sum_{\boldsymbol{\eta} \in \mathcal{W}_k^\delta} (-1)^{\boldsymbol{y_k}\boldsymbol{\eta}^T} \left(\imath \tan \frac{2\pi}{2^l} \right)^{w_H(\boldsymbol{\eta})}, \text{ for } \delta \in \{0, 1\}. \tag{5.37}$$

Here, $\boldsymbol{y_k} = \boldsymbol{y}\big|_{\Gamma_k}$ be the restriction of the character vector $\boldsymbol{y}$ to Γ_k. Let $\gamma = \frac{2\pi(N_k - 2w_H(\boldsymbol{y_k}))}{2^l}$. We apply (5.5) and (5.19) to simplify (5.37) as

$$\begin{aligned} f_k(\delta) &= \begin{cases} \cos\gamma \cdot \left(\sec \frac{2\pi}{2^l}\right)^{N_k} & \text{if } \delta = 0, \\ \imath \sin\gamma \cdot \left(\sec \frac{2\pi}{2^l}\right)^{N_k} & \text{if } \delta = 1, \end{cases} \\ &= \begin{cases} \left(\sec \frac{2\pi}{2^l}\right)^{N_k} & \text{if } \delta = 0, \\ 0 & \text{if } \delta = 1. \end{cases} \end{aligned} \tag{5.38}$$

Therefore, the summation (5.36) is nonzero if only if $\boldsymbol{r} = \boldsymbol{0}$ (i.e. summing over $\mathcal{W}(\boldsymbol{0})$).

To show the first trigonometric identity in Theorem 5, we note that $\mathcal{B}(\boldsymbol{a}) \supset \mathcal{W}(\boldsymbol{0})$.

Then, for all $l \geq 3$

$$\sum_{\boldsymbol{v}\in\mathcal{B}(\boldsymbol{a})} \epsilon_v \left(\imath \tan\frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})} = \sum_{\boldsymbol{v}\in\mathcal{W}} \epsilon_{\boldsymbol{v}} \left(\imath \tan\frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v})} = \prod_{\substack{k \\ \Gamma_k\subseteq \mathrm{supp}(\boldsymbol{a})}} \left(\sec\frac{2\pi}{2^l}\right)^{N_k}$$
$$= \left(\sec\frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{a})}. \qquad (5.39)$$

To verify the second condition, let $\boldsymbol{\omega} \in \mathcal{O}(\boldsymbol{a}) = \mathbb{F}_2^{w_H(\boldsymbol{a})} \setminus \mathcal{B}(\boldsymbol{a})$ and we change variables to $\boldsymbol{\beta} = \boldsymbol{v} \oplus \boldsymbol{\omega}$ and $\boldsymbol{\omega}$ on the right hand side (note that we have extended the $\epsilon_{\boldsymbol{v}}$ to all binary vectors). Since $\mathcal{W}(\mathbf{0})$ is not contained in any nontrival coset of $\mathcal{B}(\boldsymbol{a})$, we have

$$\sum_{\boldsymbol{v}\in\mathcal{B}(\boldsymbol{a})} \epsilon_{\boldsymbol{v}} \left(\imath \tan\frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{v}\oplus\boldsymbol{\omega})} = \epsilon_{\boldsymbol{\omega}} \sum_{\boldsymbol{\beta}\in\boldsymbol{\omega}\oplus\mathcal{B}(\boldsymbol{a})} \epsilon_{\boldsymbol{\beta}} \left(\imath \tan\frac{2\pi}{2^l}\right)^{w_H(\boldsymbol{\beta})}$$
$$= 0, \qquad (5.40)$$

for all $l \geq 3$ and $\boldsymbol{\omega} \neq \mathbf{0}$. □

5.4 Constant Excitation Code

We now use the two conditions in Theorem 3 to show that if a CSS code is oblivious to coherent noise, then it is a constant excitation code.

Corollary 15. *A CSS code is oblivious to coherent noise if and only if it is a constant excitation code.*

If the CSS code is error-detecting ($d > 1$) then the weights in different cosets of the X-stabilizers are identical.

Proof. Consider an $[[n, k, d]]$ CSS$(X, \mathcal{C}_2; Z, \mathcal{C}_1^\perp)$ code with a fixed character vector $\boldsymbol{y}$ for Z-stabilizers. If $\boldsymbol{w}$ is a coset representative for $\mathcal{C}_2$ in $\mathcal{C}_1$, then $\boldsymbol{w} \perp \mathcal{C}_1^\perp$ so

$\boldsymbol{w}\big|_{\Gamma_k} = \mathbf{0}$ or $\mathbf{1}$. If $\boldsymbol{x} \in \mathcal{C}_2$, then by Lemma 12, we have $\boldsymbol{x}\big|_{\Gamma_k} = \mathbf{0}$ or $\mathbf{1}$ for all k. Theorem 3 implies $w_H(\boldsymbol{y_k}) = \frac{|\Gamma_k|}{2}$ for all k, where $\boldsymbol{y_k} = \boldsymbol{y}\big|_{\Gamma_k}$. Since $(\boldsymbol{w} \oplus \boldsymbol{x}) = \mathbf{0}$ or $\mathbf{1}$ on any Γ_k, adding $\boldsymbol{y_k}$ to the sum either leaves $\boldsymbol{y_k}$ unchanged or just flips all entries of $\boldsymbol{y_k}$. In both cases, the Hamming weight of the sum $(\boldsymbol{w} \oplus \boldsymbol{x} \oplus \boldsymbol{y})$ is exactly $\frac{|\Gamma_k|}{2}$ on any Γ_k. If $\Gamma = \bigcup_{k=1}^{t} \Gamma_k$, then

$$w_H(\boldsymbol{w} \oplus \boldsymbol{x} \oplus \boldsymbol{y}\big|_{\Gamma}) = \frac{\sum_{k=1}^{t} |\Gamma_k|}{2}. \tag{5.41}$$

If $V = \{1, 2, \ldots, n\} \backslash \Gamma$, then the first condition in Theorem 3 implies that $w_H(\boldsymbol{x}\big|_V) = \mathbf{0}$, so that for fixed $\boldsymbol{w}$

$$w_H(\boldsymbol{w} \oplus \boldsymbol{x} \oplus \boldsymbol{y}) = w_H(\boldsymbol{w} \oplus \boldsymbol{x} \oplus \boldsymbol{y}\big|_{\Gamma}) + w_H(\boldsymbol{w} \oplus \boldsymbol{x} \oplus \boldsymbol{y}\big|_{V}) \tag{5.42}$$

is constant for all $\boldsymbol{x} \in \mathcal{C}_2$, and the CSS code is a constant excitation code. The sufficiency follows from the observation that a transversal θ Z-rotation acts as a global phase on a constant excitation code. If the CSS code is error detecting, then for all $i \in V$ there exists $\epsilon_i \in \{\pm 1\}$ such that $\epsilon_i E(\mathbf{0}, \boldsymbol{e_i})$ is a Z-stabilizer. Hence $\boldsymbol{w}\big|_{\boldsymbol{v}} = \mathbf{0}$ for all coset representatives $\boldsymbol{w} = \boldsymbol{v} G_{\mathcal{C}_1/\mathcal{C}_2}$ of $\mathcal{C}_2$ in $\mathcal{C}_1$. It now follows from (5.42) that $w_H(\boldsymbol{w} \oplus \boldsymbol{x} \oplus \boldsymbol{y}) = \frac{|\Gamma|}{2} + w_H(\boldsymbol{y}\big|_{\boldsymbol{v}})$ is constant. □

Chapter 6

Construction of Quantum Codes Oblivious to Coherent Noise

6.1 CSS Code Construction

Let $\mathcal{A}_2 \subset \mathcal{A}_1$ be two classical codes with length t, and let R_2, R_1 respectively be the rates of $\mathcal{A}_2, \mathcal{A}_1$. We may construct a $[[t, (R_1 - R_2)t, d = \min\{d_{\min}(\mathcal{A}_1), d_{\min}(\mathcal{A}_2^{\perp})\}]]$ CSS code by choosing X-stabilizers from $\mathcal{A}_2$ and Z-stabilizers from $\mathcal{A}_1^{\perp}$. Let $M \geq 2$ be even, and let $\mathcal{W}$ be the $[M, M-1]$ single parity check code consisting of all vectors with even weight of length M. Consider the CSS$(X, \mathcal{C}_2; Z, \mathcal{C}_1^{\perp})$ code where

$$\mathcal{C}_2 = \mathcal{A}_2 \otimes \mathbf{1}_M, \tag{6.1}$$

$$\mathcal{C}_1^{\perp} = \left\{ (\boldsymbol{b} \otimes \boldsymbol{e_1}) \oplus \boldsymbol{w} : \boldsymbol{b} \in \mathcal{A}_1^{\perp} \text{ and } \boldsymbol{w} \in \bigoplus_{k=1}^{t} \mathcal{W} \right\}, \tag{6.2}$$

and $\mathbf{1}_M$ is the all-ones vector of length M. Note that the code $\mathcal{C}_1^{\perp}$ includes the direct sum of t single-parity-check codes $\mathcal{W}$. We determine signs of elements in $\mathcal{C}_1^{\perp}$ (Z stabilizers) by choosing a character vector $\boldsymbol{y} \in \mathbb{F}_2^{tM}$, and we satisfy condition (2) of Theorem 3 by choosing $w_H(\boldsymbol{y_k}) = M/2$, where $\boldsymbol{y_k} = \boldsymbol{y}\big|_{\Gamma_k}$. The sign $\epsilon_{\boldsymbol{z}}$ of the Z-stabilizer $\epsilon_{\boldsymbol{z}} E(\mathbf{0}, \boldsymbol{z})$ is given by $\epsilon_{\boldsymbol{z}} = (-1)^{\boldsymbol{y_k z}^T}$. The number of logical qubits is

$$\begin{aligned} &tM - \dim(\mathcal{C}_1^{\perp}) - \dim(\mathcal{C}_2) \\ &= tM - t(M-1) - (1-R_1)t - R_2 t = (R_1 - R_2)t. \end{aligned} \tag{6.3}$$

If z is a vector of minimum weight that is orthogonal to all X-stabilizers, then either $\boldsymbol{z}$ is a Z-stabilizer or $\boldsymbol{z}$ is a vector from $\mathcal{A}_2^{\perp}$ interspersed with zeros. Hence the minimum distance d of the CSS code is at least $\min(d_{\min}(\mathcal{A}_1)M, d_{\min}(\mathcal{A}_2^{\perp}))$. Thus, we have constructed a CSS code family with parameters $[[tM, (R_2 - R_1)t, \geq \min(d_{\min}(\mathcal{A}_1)M, d_{\min}(\mathcal{A}_2^{\perp}))]]$, that is oblivious to coherent noise.

For fixed M, if we choose a family CSS codes with finite rate, then the new CSS family also have finite rate but with possible higher distances. If we allow both M and t to grow without bound, then the new CSS family may achieve increased distance but will have vanishing rate.

Example 6. *We may choose $\mathcal{A}_1 = \mathbb{F}_2^{2L}$, $\mathcal{A}_2$, and $M = 2L$ to be the $[2L, 2L-1]$ single-parity-check code to obtain the family of $[[4L^2, 1, 2L]]$ Shor codes.*

The dual-rail inner code [KLM01] is the CSS code determined by the specific stabilizer group $\mathcal{S} = \langle -Z_1 Z_2 \rangle$. Ouyang [Ouy20] observed that it was possible to construct a constant excitation code by concatenating an outer stabilizer code with an inner dual-rail code. This is simply because concatenation maps $|0\rangle$ to $|01\rangle$ and $|1\rangle$ to $|10\rangle$. In this case the number of physical qubits doubles. When $M = 2$, the construction described above coincides with the dual-rail construction. However, our approach has shown that *any* CSS code can be made oblivious to coherent noise, without requiring a special stabilizer group as in the original dual-rail construction. In fact, our approach can be extended to any stabilizer code as shown in the following section.

6.2 Generalizing to Stabilizer Codes

Consider an $[[n, k, d]]$ stabilizer code with generator matrix

$$G_{\mathcal{S}} = \left[\begin{array}{c|c} A & B \\ \hline & C \end{array}\right] \begin{array}{c} r-l \\ l \end{array} \quad . \tag{6.4}$$

Here, $r = n - k$, and the matrix C is the generator matrix of the space $\{\boldsymbol{z} \in \mathbb{F}_2^n | \epsilon_{\boldsymbol{z}} E(0, \boldsymbol{z}) \in S\}$ (thus the matrix A has full row rank). The stabilizer code derived from our construction has generator matrix

$$G_{\mathcal{S}'} = \left[\begin{array}{c|c} A \otimes \mathbf{1}_M & B \otimes \boldsymbol{e}_1 \\ \hline & C \otimes \boldsymbol{e}_1 \\ \hline & I_n \otimes W \end{array}\right] \begin{array}{c} r-l \\ l \\ n(M-1) \end{array} \quad , \tag{6.5}$$

where the $(M-1) \times M$ matrix W generates the single-parity-check code. We choose signs of the $n(M-1)$ stabilizers generated by $I_n \otimes W$ so that the new stabilizer code is oblivious to coherent noise.

Theorem 16. *The minimum distance d' of the stabilizer code generated by $G_{\mathcal{S}'}$ satisfies $d \leq d' \leq Md$.*

Proof. Suppose that $(\boldsymbol{x}, \boldsymbol{y})$ is not in the row space of $G_{\mathcal{S}'}$ and $G_{\mathcal{S}'}(\boldsymbol{y}, \boldsymbol{x})^T = 0$. Note

that $M \mid w_H(\boldsymbol{x})$. We may write

$$\boldsymbol{x} = \boldsymbol{f} \otimes \mathbf{1}_M \text{ where } \boldsymbol{f} \in \mathbb{F}_2^n, \tag{6.6}$$

and $\boldsymbol{y} = (\mathbf{1}_M \otimes (\boldsymbol{w_1}, \ldots, \boldsymbol{w_n})) \oplus (\boldsymbol{g} \otimes \boldsymbol{e_1})$ where $\boldsymbol{w_i} \in \mathcal{W}$ and $\boldsymbol{g} \in \mathbb{F}_2^n$. Then

$$G_{\mathcal{S}'}(\boldsymbol{y}, \boldsymbol{x})^T = \left[\begin{array}{c|c} A & B \\ \hline & C \end{array}\right] (\boldsymbol{g}, \boldsymbol{f})^T = 0. \tag{6.7}$$

The weight of $(\boldsymbol{x}, \boldsymbol{y})$ is at least the weight of $(\boldsymbol{f}, \boldsymbol{g})$ which is at least d, and so $d' \geq d$. Furthermore, there exists a weight d vector $(\boldsymbol{u}, \boldsymbol{v})$ not in the row space of $G_{\mathcal{S}}$ and $G_{\mathcal{S}}(\boldsymbol{v}, \boldsymbol{u})^T = 0$. Then, we have $(\boldsymbol{u} \otimes \mathbf{1}_M, \boldsymbol{v} \otimes \boldsymbol{e_1})$ is not in the row space of $G_{\mathcal{S}'}$ and $G_{\mathcal{S}'}(\boldsymbol{v} \otimes \boldsymbol{e_1}, \boldsymbol{u} \otimes \mathbf{1}_M)^T = 0$. Hence,

$$d' \leq w_H(\boldsymbol{u} \otimes \mathbf{1}_M, \boldsymbol{v} \otimes \boldsymbol{e_1}) \leq M \cdot w_H(\boldsymbol{u}, \boldsymbol{v}) = Md.$$

□

The next example also demonstrates that the dual-rail construction may sometimes increase minimum distance, and this may be a reason to investigate $M > 2$ in the above construction, where the distance d' satisfies $d \leq d' \leq Md$ (Theorem 16).

Example 7. Consider the $[[5, 1, 3]]$ stabilizer code with generator matrix $G_{\mathcal{S}} = [A|B]$ where

$$A = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix} \text{ and } B = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{6.8}$$

The code is not a CSS code. The stabilizer code derived from our construction has generator matrix

$$G_{\mathcal{S}'} = \left[\begin{array}{c|c} A \otimes [1,1] & B \otimes [1,0] \\ \hline & I_5 \otimes [1,1] \end{array}\right] \begin{array}{c} \text{signs} \\ + \\ - \end{array} . \tag{6.9}$$

Consider $(\boldsymbol{y}, \boldsymbol{x})$ such that $(\boldsymbol{x}, \boldsymbol{y})$ is not in the row space of $G_{\mathcal{S}'}$ and $G_{\mathcal{S}'}(\boldsymbol{y}, \boldsymbol{x})^T = 0$. We observe that $2 \mid w_H(x)$. If $\boldsymbol{x} = \mathbf{0}$, then $\boldsymbol{y} = \boldsymbol{w} \otimes [1,1] \oplus \mathbf{1}_5 \otimes [1,0]$ for some $\boldsymbol{w} \in \mathbb{F}_2^5$, then after possibly applying the cyclic symmetry, we may assume $\boldsymbol{x} = \boldsymbol{e_1} \oplus \boldsymbol{e_2}$ and $(A \otimes [1,1])\boldsymbol{y}^T = [0,0,0,1]^T$. We observe that neither $[0,0,0,1]$ nor $[1,0,1,0] \oplus [0,0,0,1] = [1,0,1,1]$ is a column of A. It follows that the distance $d' \geq 4$. In fact, we see $d' = 4$ by taking

$$(\boldsymbol{x}', \boldsymbol{y}') = [1,1,0,0,0,0,0,0,0,0|0,0,1,0,0,0,0,0,1,0]. \tag{6.10}$$

Hence, the stabilizer code derived from the above construction has parameters $[[10,1,4]]$.

By choosing y to be either $[0,1]$ or $[1,0]$ for each of the five connected components with size $M = 2$, we ensure $\mathcal{V}(\mathcal{S}')$ to satisfy Theorem 3, and thus it is oblivious to coherent noise.

We now consider the cases that when some qubits are not involved in any X-stabilizer.

Example 8. Consider the $[[5,1,2]]$ CSS code with the character vector $\boldsymbol{y} = [1,0,1,0,1]$

defined by the following generator matrix

$$G_S = \left[\begin{array}{ccccc|ccccc} 1 & 1 & 1 & 1 & 0 & & & & & \\ \hline & & & & & 1 & 1 & 0 & 0 & 0 \\ & & & & & 0 & 0 & 1 & 1 & 0 \\ & & & & & 0 & 0 & 0 & 0 & 1 \end{array}\right]. \tag{6.11}$$

Here, we have two connected components $\Gamma_1 = \{1, 2\}$ and $\Gamma_2 = \{3, 4\}$. Since $\text{supp}([1, 1, 1, 1, 0]) = \Gamma_1 \cup \Gamma_2$, and $w_H(\boldsymbol{y_k}) = \frac{|\Gamma_k|}{2} = 1$ for $k = 1, 2$, the two conditions in Theorem 3 are satisfied. Hence, the $[[5, 1, 2]]$ CSS code is oblivious to coherent noise, and we use (2.30) to compute computational states to verify it is a constant excitation code:

$$|\bar{0}\rangle = \frac{1}{\sqrt{2}}(|01011\rangle + |10101\rangle), \tag{6.12}$$

$$|\bar{1}\rangle = \frac{1}{\sqrt{2}}(|10011\rangle + |10101\rangle). \tag{6.13}$$

Here, the constant excitation is $3 \neq \frac{5}{2}$ (half of the number of physical qubits). After the concatenation, we may introduce extra physical qubits by adding zeros to the current X-stabilizers and including all weight 1 Z-stabilizers on the extra qubits. This construction reduces rate, but provides a large class of codes that may be useful in implementing logical gates.

Given any $[[n, k, d]]$ stabilizer code, the theoretical construction in (6.5) and the observation in Example 8 provide a $[[Mn + s, k, d']]$ stabilizer code that is oblivious to coherent noise, where $d \leq d' \leq Md$, $M \geq 2$ is even, and $s \geq 0$.

Chapter 7

Coherent Noise in the Form of Generator Coefficients

7.1 Review of Generator Coefficients

In the previous Chapters, we focused on transversal Z-rotations, which are a special form of diagonal gates. In this Chapter, more general conditions for a quantum codes to be preserved by diagonal gates are derived. We first review the *Generator Coefficient Framework* which describes the evolution of stabilizer code states under a physical diagonal gate $U_Z = \sum_{\boldsymbol{u}\in\mathbb{F}_2^n} d_{\boldsymbol{u}}|\boldsymbol{u}\rangle\langle\boldsymbol{u}|$ (See [HLC22b] for more details).

Note that $|\boldsymbol{u}\rangle\langle\boldsymbol{u}| = \frac{1}{2^n}\sum_{\boldsymbol{v}\in\mathbb{F}_2^n}(-1)^{\boldsymbol{u}\boldsymbol{v}^T}E(\boldsymbol{0},\boldsymbol{v})$. Alternatively, we may expand U_Z in the Pauli basis

$$U_Z = \sum_{\boldsymbol{v}\in\mathbb{F}_2^n} f(\boldsymbol{v})E(\boldsymbol{0},\boldsymbol{v}), \tag{7.1}$$

where

$$f(\boldsymbol{v}) = \frac{1}{2^n}\sum_{\boldsymbol{u}\in\mathbb{F}_2^n}(-1)^{\boldsymbol{u}\boldsymbol{v}^T}d_{\boldsymbol{u}}. \tag{7.2}$$

The Hadamard gate H_{2^n} connects the coefficients in the standard basis with those in the Pauli basis as follows

$$[f(\boldsymbol{v})]_{\boldsymbol{v}\in\mathbb{F}_2^n} = [d_{\boldsymbol{u}}]_{\boldsymbol{u}\in\mathbb{F}_2^n}\, H_{2^n}, \tag{7.3}$$

where $H = \frac{1}{\sqrt{2}}\left(|0\rangle\langle 0| + |0\rangle\langle 1| + |1\rangle\langle 0| - |1\rangle\langle 1|\right)$ and $H_{2^n} = H \otimes H_{2^{n-1}} = H^{\otimes n}$ is the

Hadamard gate.

We consider the average logical channel induced by U_Z on an $[\![n,k,d]\!]$ CSS$(X,\mathcal{C}_2;Z,\mathcal{C}_1^\perp,\boldsymbol{y})$ code that results from the four steps : (1) preparing any code state ρ_1; (2) applying a diagonal physical gate U_Z to obtain ρ_2; (3) using X-stabilizers to measure ρ_2 (we only consider Z-errors as the same reasons in [BK05, BH12]), to obtain the syndrome $\boldsymbol{\mu}$ with probability $p_{\boldsymbol{\mu}}$, and the post-measurement state ρ_3; (4) applying a Pauli correction to ρ_3, to obtain ρ_4. The correction might induce some undetectable Z-logical $\epsilon_{(\mathbf{0},\boldsymbol{\gamma}_{\boldsymbol{\mu}})}E(\mathbf{0},\boldsymbol{\gamma}_{\boldsymbol{\mu}})$ with $\boldsymbol{\gamma}_{\mathbf{0}}=\mathbf{0}$. Let $B_{\boldsymbol{\mu}}$ be the effective physical operator corresponding to the syndrome $\boldsymbol{\mu}$. Then the evolution of code states can be described as

$$\rho_4 = \sum_{\boldsymbol{\mu}\in\mathbb{F}_2^n/\mathcal{C}_2^\perp} B_{\boldsymbol{\mu}}\rho_1 B_{\boldsymbol{\mu}}^\dagger. \tag{7.4}$$

The generator coefficients $A_{\boldsymbol{\mu},\boldsymbol{\gamma}}$ are obtained by expanding the logical operator $B_{\boldsymbol{\mu}}$ in terms of Z-logical Pauli operators $\epsilon_{(\mathbf{0},\boldsymbol{\gamma})}E(\mathbf{0},\boldsymbol{\gamma})$,

$$B_{\boldsymbol{\mu}} = \epsilon_{(\mathbf{0},\boldsymbol{\gamma}_{\boldsymbol{\mu}})}E(\mathbf{0},\boldsymbol{\gamma}_{\boldsymbol{\mu}}) \sum_{\boldsymbol{\gamma}\in\mathcal{C}_2^\perp/\mathcal{C}_1^\perp} A_{\boldsymbol{\mu},\boldsymbol{\gamma}}\ \epsilon_{(\mathbf{0},\boldsymbol{\gamma})}E(\mathbf{0},\boldsymbol{\gamma}), \tag{7.5}$$

where $\epsilon_{(\mathbf{0},\boldsymbol{\gamma}_{\boldsymbol{\mu}})}E(\mathbf{0},\boldsymbol{\gamma}_{\boldsymbol{\mu}})$ models the Z-logical Pauli correction introduced by a decoder. For each pair of an X-syndrome $\boldsymbol{\mu}\in\mathbb{F}_2^n/\mathcal{C}_2^\perp$ and a Z-logical $\boldsymbol{\gamma}\in\mathcal{C}_2^\perp/\mathcal{C}_1^\perp$, the generator coefficient $A_{\boldsymbol{\mu},\boldsymbol{\gamma}}$ corresponding to U_Z is

$$A_{\boldsymbol{\mu},\boldsymbol{\gamma}} \coloneqq \sum_{\boldsymbol{z}\in\mathcal{C}_1^\perp+\boldsymbol{\mu}+\boldsymbol{\gamma}} \epsilon_{(\mathbf{0},\boldsymbol{z})}f(\boldsymbol{z}), \tag{7.6}$$

where $\epsilon_{(\mathbf{0},\boldsymbol{z})}=(-1)^{\boldsymbol{z}\boldsymbol{y}^T}$ is the sign of the Z-stabilizer $E(\mathbf{0},\boldsymbol{z})$. The chosen Z-logicals and X-syndromes are not unique, but different choices only differ by a global phase.

Generator coefficients use the CSS code to organize the Pauli coefficients of U_Z into groups and to balance them by tuning the signs of Z-stabilizers. We use (7.2) to simplify (7.6) as

$$\begin{aligned} A_{\boldsymbol{\mu},\boldsymbol{\gamma}} &= \frac{1}{2^n} \sum_{\boldsymbol{u}\in\mathbb{F}_2^n} \sum_{\boldsymbol{z}\in\mathcal{C}_1^\perp+\boldsymbol{\mu}+\boldsymbol{\gamma}} (-1)^{\boldsymbol{z}\boldsymbol{y}^T} (-1)^{\boldsymbol{z}\boldsymbol{u}^T} d_{\boldsymbol{u}} \\ &= \frac{1}{|\mathcal{C}_1|} \sum_{\boldsymbol{u}\in\mathcal{C}_1} (-1)^{(\boldsymbol{\mu}\oplus\boldsymbol{\gamma})\boldsymbol{u}^T} d_{\boldsymbol{u}\oplus\boldsymbol{y}}, \end{aligned} \tag{7.7}$$

where $|\mathcal{C}_1| = 2^{k_1}$ is the size of $\mathcal{C}_1$. We organize the generator coefficients in a matrix $M_{(\mathbb{F}_2^n/\mathcal{C}_2^\perp,\mathcal{C}_2^\perp/\mathcal{C}_1^\perp)}$ with rows indexed by X-syndromes and columns by Z-logicals,

$$M_{(\mathbb{F}_2^n/\mathcal{C}_2^\perp,\mathcal{C}_2^\perp/\mathcal{C}_1^\perp)} = \begin{bmatrix} [A_{\boldsymbol{\mu}=\mathbf{0},\boldsymbol{\gamma}}]_{\boldsymbol{\gamma}\in\mathcal{C}_2^\perp/\mathcal{C}_1^\perp} \\ [A_{\boldsymbol{\mu}=\boldsymbol{\mu}_1,\boldsymbol{\gamma}}]_{\boldsymbol{\gamma}\in\mathcal{C}_2^\perp/\mathcal{C}_1^\perp} \\ \vdots \\ [A_{\boldsymbol{\mu}=\boldsymbol{\mu}_{\mathbf{2}^{\boldsymbol{k_2}}-\mathbf{1}},\boldsymbol{\gamma}}]_{\boldsymbol{\gamma}\in\mathcal{C}_2^\perp/\mathcal{C}_1^\perp} \end{bmatrix}_{\boldsymbol{\mu}\in\mathbb{F}_2^n/\mathcal{C}_2^\perp}. \tag{7.8}$$

For fixed $\boldsymbol{\mu} \in \mathbb{F}_2^n/\mathcal{C}_2^\perp$,

$$[A_{\boldsymbol{\mu},\boldsymbol{\gamma}}]_{\boldsymbol{\gamma}\in\mathcal{C}_2^\perp/\mathcal{C}_1^\perp} = \frac{1}{|C_1|} [d_{\boldsymbol{u}\oplus\boldsymbol{y}}]_{\boldsymbol{u}\in\mathcal{C}_1} H^{\boldsymbol{\mu}}_{(\mathcal{C}_1,\mathcal{C}_2^\perp/\mathcal{C}_1^\perp)}, \tag{7.9}$$

where $H^{\boldsymbol{\mu}}_{(\mathcal{C}_1,\mathcal{C}_2^\perp/\mathcal{C}_1^\perp)} = [(-1)^{(\boldsymbol{\mu}\oplus\boldsymbol{\gamma})\boldsymbol{u}^T}]_{\boldsymbol{u}\in\mathcal{C}_1,\boldsymbol{\gamma}\in\mathcal{C}_2^\perp/\mathcal{C}_1^\perp}$.

Theorem 17 (Theorem 7 in [HLC22b])**.** *The physical gate $U_Z = \sum_{\boldsymbol{u}\in\mathbb{F}_2^n} d_{\boldsymbol{u}} |\boldsymbol{u}\rangle\langle\boldsymbol{u}|$ preserves a CSS$(X, \mathcal{C}_2; Z, \mathcal{C}_1^\perp, \boldsymbol{y})$ codespace if and only if*

$$\sum_{\boldsymbol{\gamma}\in\mathcal{C}_2^\perp/\mathcal{C}_1^\perp} |A_{\mathbf{0},\boldsymbol{\gamma}}|^2 = \sum_{\boldsymbol{\gamma}\in\mathcal{C}_2^\perp/\mathcal{C}_1^\perp} \overline{A_{\mathbf{0},\boldsymbol{\gamma}}} A_{\mathbf{0},\boldsymbol{\gamma}} = 1. \tag{7.10}$$

Here, $|\cdot|$ denotes the complex norm.

Proof. Invariance of the codespace is equivalent to requiring the effective physical operator corresponding to the trivial syndrome $B_{\boldsymbol{\mu}=\mathbf{0}}$ to be unitary. □

Note that (7.10) is also equivalent to $[A_{\boldsymbol{\mu}\neq\mathbf{0},\boldsymbol{\gamma}}]_{\boldsymbol{\gamma}\in\mathcal{C}_2^\perp/\mathcal{C}_1^\perp} = \mathbf{0}$ [HLC22b, Theorem 6]. The induced logical operator is

$$\begin{aligned} U_Z^L &= \sum_{\boldsymbol{\alpha}\in\mathbb{F}_2^k} A_{\mathbf{0},g(\boldsymbol{\alpha})} E(\mathbf{0},\boldsymbol{\alpha}) \\ &= \frac{1}{|\mathcal{C}_1|} \sum_{\boldsymbol{\alpha}\in\mathbb{F}_2^k} \sum_{\boldsymbol{u}\in\mathcal{C}_1} (-1)^{g(\boldsymbol{\alpha})\boldsymbol{u}^T} d_{\boldsymbol{u}\oplus\boldsymbol{y}} E(\mathbf{0},\boldsymbol{\alpha}), \end{aligned} \tag{7.11}$$

where $g : \mathbb{F}_2^k \to \mathcal{C}_2^\perp/\mathcal{C}_1^\perp$ is a bijective map defined by $g(\boldsymbol{\alpha}) = \boldsymbol{\alpha} G_{\mathcal{C}_2^\perp/\mathcal{C}_1^\perp}$. Here, $G_{\mathcal{C}_2^\perp/\mathcal{C}_1^\perp}$ is one choice of the generator matrix of Z-logicals (coset representatives of $\mathcal{C}_2^\perp/\mathcal{C}_1^\perp$).

Example 9. *The $[\![15,1,3]\!]$ punctured quantum Reed-Muller code [BK05] is a CSS(X, $\mathcal{C}_2$; Z, $\mathcal{C}_1^\perp$, $\boldsymbol{y}=\mathbf{0}$) code, where $\mathcal{C}_2$ is generated by the degree one monomials, x_1, x_2, x_3, x_4, and $\mathcal{C}_1^\perp = \langle x_1, x_2, x_3, x_4, x_1x_2, x_1x_3, x_1x_4, x_2x_3, x_2x_4, x_3x_4\rangle$, with the first coordinate removed in both $\mathcal{C}_2$ and $\mathcal{C}_1^\perp$. It's also a triorthogonal code [BH12] for which a physical transversal T gate, $U_Z = \sum_{\boldsymbol{u}\in\mathbb{F}_2^n} \left(e^{\imath\pi/4}\right)^{w_H(\boldsymbol{u})} |\boldsymbol{u}\rangle\langle\boldsymbol{u}|$, induces a logical transversal T gate up to some Clifford gates. Here, $w_H(\boldsymbol{u}) = \boldsymbol{u}\boldsymbol{u}^T$ denotes the Hamming weight of the binary vector $\boldsymbol{u}$. Note that $\mathcal{C}_1$ is the classical punctured $RM(1,4)$ code with weight distribution given in Table 7.1 below.*

Table 7.1: The weight distribution of $\mathcal{C}_1$ for the $[\![15,1,3]\!]$ code

weight	0	7	8	15
multiplicity	1	15	15	1

Then, $d_{\boldsymbol{u}} = 1$ for $\boldsymbol{u}\in\mathcal{C}_1$ satisfying $w_H(\boldsymbol{u}) = 0$ or 8, and $d_{\boldsymbol{u}} = e^{-\imath\pi/4}$ for $\boldsymbol{u}\in\mathcal{C}_1$ satisfying $w_H(\boldsymbol{u}) = 7$ or 15. Since the Z-logical $\boldsymbol{\gamma} = \mathbf{1}$, the all-one vector, it only changes the signs of $d_{\boldsymbol{u}}$ with odd weight. It follows from (7.11) *that the induced logical*

operator is

$$U_Z^L = \frac{16}{32}(1 + e^{-\imath\pi/4})E(0,0) + \frac{16}{32}(1 - e^{-\imath\pi/4})E(0,1) = T^\dagger. \tag{7.12}$$

Remark 18. *It follows from* (7.11) *that the induced logical operator is completely specified by* $|\mathcal{C}_1|$ *diagonal entries in the physical gate* U_Z*. If we choose a CSS code and target a particular logical gate, then the constraints on the corresponding physical gates only apply to the diagonal elements corresponding to the coset* $\mathcal{C}_1 + \boldsymbol{y}$*.*

7.2 Generalizing the Error Model

Given a CSS code, the generator coefficient framework not only represents when a physical diagonal gate preserves the codespace, but it also characterizes all the possible physical gates that realize a target diagonal logical gate. We start from the simplest case, when the logical operator is the identity.

Lemma 19. *The physical gate* $U_Z = \sum_{\boldsymbol{u}\in\mathbb{F}_2^n} d_{\boldsymbol{u}}|\boldsymbol{u}\rangle\langle\boldsymbol{u}|$ *acts as the logical identity on the CSS*$(X, \mathcal{C}_2; Z, \mathcal{C}_1^\perp, \boldsymbol{y})$ *codespace if and only if* $d_{\boldsymbol{u}\oplus\boldsymbol{y}}$ *are the same for all* $\boldsymbol{u} \in \mathcal{C}_1$*.*

Proof. It follows from (7.11) that $U_Z^L = I_{2^k}$ if and only if

$$|A_{\boldsymbol{\mu}=\boldsymbol{0},\boldsymbol{\gamma}=\boldsymbol{0}}| = \left|\frac{1}{|\mathcal{C}_1|}\sum_{\boldsymbol{u}\in\mathcal{C}_1} d_{\boldsymbol{u}\oplus\boldsymbol{y}}\right| = 1, \tag{7.13}$$

which is equivalent to requiring that 2^{k_1} diagonal entries of the physical gate U_Z indexed by the set $\mathcal{C}_1 + \boldsymbol{y}$ are identical. □

The mapping from a physical gate that preserves a given CSS code to the induced logical operator is a group homomorphism. The kernel of this homomorphism is the group of phsyical gates that induce the logical identity.

Remark 20. *Given a CSS code, Lemma 19 characterizes all the diagonal physical gates that induce the identity on the codespace. This enables code design within a decoherence-free subspace (DFS) for a particular noise system. For homogeneous coherent noise (same angle on each physical qubit), we consider*

$$U_Z = \begin{bmatrix} 1 & 0 \\ 0 & e^{\imath\theta} \end{bmatrix}^{\otimes n} \equiv \sum_{\boldsymbol{u}\in\mathbb{F}_2^n} \left(e^{\imath\theta}\right)^{w_H(\boldsymbol{u})} |\boldsymbol{u}\rangle\langle\boldsymbol{u}|, \tag{7.14}$$

with $\theta \in (0, 2\pi)$. We design CSS codes that are oblivious to all such gates by making sure all the Hamming weights in the coset $\mathcal{C}_1 + \boldsymbol{y}$ are the same (a new perspective on the results in [HLC22a, Ouy21]). For coherent noise with inhomogeneous angles, this perspective enables code design to mitigate these correlated errors. For example, we consider $U_Z = \begin{bmatrix} 1 & 0 \\ 0 & e^{\imath\theta_1} \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 0 & e^{\imath\theta_2} \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 0 & e^{\imath\theta} \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 0 & e^{\imath\theta} \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 0 & e^{\imath\theta_1'} \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 0 & e^{\imath\theta_2'} \end{bmatrix}$, with $\theta \in (0, 2\pi)$ and $\theta_1 + \theta_2 = \theta_1' + \theta_2' = \theta$. By selecting the diagonal elements of U_Z with the same value, we design a $[\![6, 1, 2]\!]$ CSS code within a DFS for the inhomogeneous noise system, where

$$G_{\mathcal{C}_2} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \subset G_{\mathcal{C}_1} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix} \text{ and } \boldsymbol{y} = [1, 1, 1, 0, 0, 0]. \tag{7.15}$$

Chapter 8

Constraints Associated with Climbing the Clifford Hierarchy

When transversal Z-rotations are used to induce non-identity logical operations on a stabilizer code, the classical binary codes Z_j formed by the Z-stabilizers supported on a given X-stabilizer must satisfy Theorem 5 for all $l \leq l_{\max} < \infty$. In this section, we show that as l increases, the weight enumerator of such a code Z_j must satisfy a sequence of constraints.

When the code Z_j is self-dual, the first of the constraints connects to Gleason's Theorem [Gle71] that the weight enumerator is a sum of products of certain given polynomials. We note that there are many connections between self dual codes, lattices, quadratic forms, and quantum error correcting codes (see [NRS06] for more information).

We derive constraints that apply to a polynomial $R_j(x)$ determined by the character ϵ_v and the weight enumerator of Z_j; when Z_j is self-dual, the polynomial $R_j(x)$ depends only on the weight enumerator. Invariance under transversal $\frac{\pi}{2^l}$ Z-rotations implies that $R_j(x)$ is divisible by the minimal polynomial of $\tan \frac{2\pi}{2^l}$ for $l = 3, ..., l_{\max}$.

Since $l_{\max} \geq 3$, it follows from the second condition of Remark 7 that every code Z_j contains a $[w_H(a_j), \frac{w_H(a_j)}{2}]$ self-dual code A_j. Here we assume $Z_j = A_j$, then add a Z-stabilizer $E(0, z)$ to the stabilizer group S, and verify that the identities (2.31) and (2.32) still hold. If $z \not\preceq a_j$, then Z_j is unchanged. If $z \preceq a_j$, then $Z_j' = \langle Z_j, z \rangle$ and we have

$$\sum_{v\in Z_j'}\epsilon_v\left(\imath\tan\frac{2\pi}{2^l}\right)^{w_H(v)}=\sum_{v\in Z_j}\epsilon_v\left(\imath\tan\frac{2\pi}{2^l}\right)^{w_H(v)}+\sum_{v\in Z_j}\epsilon_v\epsilon_z\left(\imath\tan\frac{2\pi}{2^l}\right)^{w_H(v\oplus z)} \tag{8.1}$$

$$=\sec\left(\frac{2\pi}{2^l}\right)^{w_H(a_j)}+\epsilon_z\sum_{v\in Z_j}\epsilon_v\left(\imath\tan\frac{2\pi}{2^l}\right)^{w_H(v\oplus z)} \tag{8.2}$$

$$=\sec\left(\frac{2\pi}{2^l}\right)^{w_H(a_j)}. \tag{8.3}$$

Note that if $\omega\in O_j'$, then $z\oplus\omega\in O_j$, and we have

$$\sum_{v\in Z_j'}\epsilon_v\left(\imath\tan\frac{2\pi}{2^l}\right)^{w_H(v\oplus w)}=\sum_{v\in Z_j}\epsilon_v\left(\imath\tan\frac{2\pi}{2^l}\right)^{w_H(v\oplus w)}+\sum_{v\in Z_j}\epsilon_v\epsilon_z\left(\imath\tan\frac{2\pi}{2^l}\right)^{w_H(v\oplus z\oplus w)} \tag{8.4}$$

$$=0+0=0. \tag{8.5}$$

Once conditions (2.31) and (2.32) are satisfied by a subcode of Z_j (for example A_j), they remain satisfied as Z-stabilizers are added to the stabilizer group. Conversely, it is natural to ask whether the conditions of Theorem 5 for $\theta=\frac{\pi}{4}$ (preserved by T gate) imply there exists a self-dual code satisfying (2.31) and (2.32). We have the following result:

Theorem 21. *For $\epsilon E(\boldsymbol{a},\boldsymbol{b})\in\mathcal{S}$ with $\boldsymbol{a}\neq\boldsymbol{0}$, if the condition (2.31) is satisfied for $\theta=\frac{\pi}{4}$ (preserved by T gate), then we have $\dim\mathcal{B}(\boldsymbol{a})\geq\frac{w_H(\boldsymbol{a})}{2}$. Furthermore, if $\dim\mathcal{B}(\boldsymbol{a})=\frac{w_H(\boldsymbol{a})}{2}$, then $\mathcal{B}(\boldsymbol{a})$ is a self-dual code.*

Proof. By assumption, we have

$$\sum_{\boldsymbol{v}\in\mathcal{B}(\boldsymbol{a})} \epsilon_{\boldsymbol{v}} \left(\imath \tan\frac{\pi}{4}\right)^{w_H(\boldsymbol{v})} = \left(\sec\frac{\pi}{4}\right)^{w_H(\boldsymbol{a})} \tag{8.6}$$

$$\sum_{\boldsymbol{v}\in\mathcal{B}(\boldsymbol{a})} \epsilon_{\boldsymbol{v}}(-1)^{\frac{w_H(\boldsymbol{v})}{2}} = 2^{\frac{w_H(\boldsymbol{a})}{2}} \tag{8.7}$$

Note that each term of the left hand side contributes either 1 or -1. So there are least $2^{w_H(\boldsymbol{a})}$ terms, which implies that $\dim \mathcal{B}(\boldsymbol{a}) \geq \frac{w_H(\boldsymbol{a})}{2}$.

Now, we assume $\dim \mathcal{B}(\boldsymbol{a}) = \frac{w_H(\boldsymbol{a})}{2}$. There are exactly $2^{w_H(\boldsymbol{a})}$ terms on the left hand side, and all of them are 1. Hence,

$$1 = \epsilon_{\boldsymbol{v}}(-1)^{\frac{w_H(\boldsymbol{v})}{2}} = (-1)^{\boldsymbol{v}\boldsymbol{y}^T + \frac{w_H(\boldsymbol{v})}{2}}, \tag{8.8}$$

where y is the characteristic vector. Therefore, we have

$$\boldsymbol{v}\boldsymbol{y}^T + \frac{w_H(\boldsymbol{v})}{2} \text{ is even for all } \boldsymbol{v} \in \mathcal{B}(\boldsymbol{a}). \tag{8.9}$$

For any $\boldsymbol{u}, \boldsymbol{v} \in \mathcal{B}(\boldsymbol{a})$, we have

$$\boldsymbol{u}\boldsymbol{v}^T = w_H(\boldsymbol{u} * \boldsymbol{v}) = \frac{w_H(\boldsymbol{u}) + w_H(\boldsymbol{v}) - w_H(\boldsymbol{u} \oplus \boldsymbol{v})}{2} \tag{8.10}$$

$$= \boldsymbol{u}\boldsymbol{y}^T + \frac{w_H(\boldsymbol{u})}{2} + \boldsymbol{v}\boldsymbol{y}^T + \frac{w_H(\boldsymbol{v})}{2} - \left((\boldsymbol{u} \oplus \boldsymbol{v})\boldsymbol{y}^T + \frac{w_H(\boldsymbol{u} \oplus \boldsymbol{v})}{2}\right) \tag{8.11}$$

$$= \text{even} + \text{even} - \text{even} = \text{even}. \tag{8.12}$$

Therefore, $\mathcal{B}(\boldsymbol{a})$ is a self-dual code. □

We now make the connection to Gleason's Theorem.

Theorem 22 (Gleason [Gle71])**.** *Let C a binary self-dual code with all Hamming*

weights divisible by c, and let $P_C(x, y)$ be the weight enumerator of C.

1. *If $c = 2$, then $P_C(x, y)$ is a sum of products of the polynomials $f(x, y) = x^2 + y^2$ and $g(x, y) = x^2y^2(x^2 - y^2)^2$.*

2. *If $c = 4$, then $P_C(x, y)$ is a sum of products of the polynomials $f(x, y) = x^8 + 14x^4y^4 + y^8$ and $g(x, y) = x^4y^4(x^4 - y^4)^4$.*

8.1 Applications of Algebraic Number Theory

Given a stabilizer code fixed by a transversal $\frac{\pi}{2^l}$ Z-rotation, we set $m_j = w_H(a_j)$ and rewrite (2.31) of Theorem 5 as

$$\sum_{v \in Z_j} \epsilon_v \left(\imath \tan \frac{2\pi}{2^l} \right)^{w_H(v)} = \left(\sec \frac{2\pi}{2^l} \right)^{m_j}. \tag{8.13}$$

Since $\sec\theta = \sqrt{1 + (\tan\theta)^2}$, we can rewrite the right hand side as

$$\left(\sec \frac{2\pi}{2^l} \right)^{m_j} = \left(1 + \left(\tan \frac{2\pi}{2^l} \right)^2 \right)^{\frac{m_j}{2}} = \sum_{t=0}^{\frac{m_j}{2}} \binom{\frac{m_j}{2}}{t} \left(\tan \frac{2\pi}{2^l} \right)^{2t}. \tag{8.14}$$

Let $Z_j(2t)$ be the set of vectors in Z_j with Hamming weight $2t$. It follows from (8.13) that the polynomial

$$R_j(x) \coloneqq \sum_{t=0}^{\frac{m_j}{2}} \left[\sum_{v \in Z_j(2t)} \epsilon_v (-1)^t - \binom{\frac{m_j}{2}}{t} \right] x^{2t} \tag{8.15}$$

vanishes at $\alpha_l = \tan \frac{2\pi}{2^l}$. When a stabilizer code $V(S)$ is preserved by all transversal Z-rotations, we must have $R_j(x) = 0$ for all Z_j. When $V(S)$ is preserved by the transversal $\frac{\pi}{2^l}$ Z-rotation for $l \leq l_{\max} < \infty$, then since the polynomial $R_j(x)$

only involves even powers of x, it is divisible by the minimal polynomials of $\tan\frac{2\pi}{2^l}$ and $-\tan\frac{2\pi}{2^l}$ for $l \leq l_{\max}$. We derive these minimal polynomials in Theorem 26 below, starting with two technical lemmas. Note that both minimal polynomials are irreducible in $\mathbb{Q}[x]$.

Lemma 23. *Let* $f(x) = \frac{2x}{1-x^2}$. *Then*

$$f^k(x) = \frac{\sum_{i=0}^{2^{k-1}-1}(-1)^i\binom{2^k}{2i+1}x^{2i+1}}{\sum_{j=0}^{2^{k-1}}(-1)^j\binom{2^k}{2j}x^{2j}}, \tag{8.16}$$

where $f^k(x) = \underbrace{f(f(\cdots f(x)))}_{k}$.

Proof. We use induction. When $k = 1$, we have

$$f^1(x) = \frac{2x}{1-x^2} = \frac{\binom{2}{1}x}{\binom{2}{0} - \binom{2}{2}x^2}. \tag{8.17}$$

When $k = 2$, we have

$$f^2(x) = \frac{2\frac{2x}{1-x^2}}{1-\left(\frac{2x}{1-x^2}\right)^2} = \frac{4x-4x^3}{1-6x^2+x^4} = \frac{\binom{4}{1}x - \binom{4}{3}x^3}{\binom{4}{0} - \binom{4}{2}x^2 + \binom{4}{4}x^4}. \tag{8.18}$$

Assume the Equation 8.16 holds for some $k \geq 2$. By induction, we have

$$f^{k+1}(x) = f\left(f^k(x)\right) = \frac{2f^k(x)}{1-(f^k(x))^2} = \frac{\frac{2\sum_{i=0}^{2^{k-1}-1}(-1)^i\binom{2^k}{2i+1}x^{2i+1}}{\sum_{j=0}^{2^{k-1}}(-1)^j\binom{2^k}{2j}x^{2j}}}{1-\left(\frac{\sum_{i=0}^{2^{k-1}-1}(-1)^i\binom{2^k}{2i+1}x^{2i+1}}{\sum_{j=0}^{2^{k-1}}(-1)^j\binom{2^k}{2j}x^{2j}}\right)^2} \tag{8.19}$$

$$\Rightarrow f^{k+1}(x) = \frac{2\left(\sum_{i=0}^{2^{k-1}-1}(-1)^i\binom{2^k}{2i+1}x^{2i+1}\right)\left(\sum_{j=0}^{2^{k-1}}(-1)^j\binom{2^k}{2j}x^{2j}\right)}{\left(\sum_{j=0}^{2^{k-1}}(-1)^j\binom{2^k}{2j}x^{2j}\right)^2 - \left(\sum_{i=0}^{2^{k-1}-1}(-1)^i\binom{2^k}{2i+1}x^{2i+1}\right)^2} \tag{8.20}$$

$$= \frac{2\sum_{r=0}^{2^k-1}\sum_{i+j=r}(-1)^r\binom{2^k}{2i+1}\binom{2^k}{2j}x^{2r+1}}{\left(\sum_{i=0}^{2^k}(-1)^{\lceil\frac{i}{2}\rceil}\binom{2^k}{i}x^i\right)\left(\sum_{j=0}^{2^k}(-1)^{\lfloor\frac{i}{2}\rfloor}\binom{2^k}{j}x^j\right)}. \tag{8.21}$$

We first look at the numerator of $f^{k+1}(x)$

$$\text{Numerator} = 2\sum_{r=0}^{2^k-1}\sum_{i+j=r}(-1)^r\binom{2^k}{2i+1}\binom{2^k}{2j}x^{2r+1} \tag{8.22}$$

$$= \sum_{r=0}^{2^k-1}\left[2\sum_{i+j=r}\binom{2^k}{2i+1}\binom{2^k}{2j}\right](-1)^r x^{2r+1} \tag{8.23}$$

$$= \sum_{r=0}^{2^k-1}\left[\sum_{i+j=r}\binom{2^k}{2i+1}\binom{2^k}{2j} + \sum_{i+j=r}\binom{2^k}{2j}\binom{2^k}{2i+1}\right](-1)^r x^{2r+1} \tag{8.24}$$

$$= \sum_{r=0}^{2^k-1}\left[\sum_{s=0}^{r}\binom{2^k}{s}\binom{2^k}{2r+1-s}\right](-1)^r x^{2r+1} \tag{8.25}$$

$$= \sum_{r=0}^{2^k-1}(-1)^r\binom{2^{k+1}}{2r+1}x^{2r+1}. \tag{8.26}$$

Then, we simplify the denominator of $f^{k+1}(x)$

$$\text{Denominator} = \left(\sum_{i=0}^{2^k}(-1)^{\lceil\frac{i}{2}\rceil}\binom{2^k}{i}x^i\right)\left(\sum_{j=0}^{2^k}(-1)^{\lfloor\frac{i}{2}\rfloor}\binom{2^k}{j}x^j\right) \tag{8.27}$$

$$= \sum_{r=0}^{2^{k+1}}\left[\sum_{i+j=r}(-1)^{\lceil\frac{i}{2}\rceil+\lfloor\frac{j}{2}\rfloor}\binom{2^k}{i}\binom{2^k}{j}\right]x^r. \tag{8.28}$$

If $r = 2p$ for some $0 \leq p \leq 2^k$, we have

$$\left[\sum_{i+j=2p} (-1)^{\lceil \frac{i}{2} \rceil + \lfloor \frac{j}{2} \rfloor} \binom{2^k}{i}\binom{2^k}{j}\right] x^{2p} = \sum_{i=0}^{2p} (-1)^{\lceil \frac{i}{2} \rceil + \lfloor \frac{2p-i}{2} \rfloor} \binom{2^k}{i}\binom{2^k}{2p-i} x^{2p} \tag{8.29}$$

$$= \sum_{i=0}^{2p} (-1)^{\lceil \frac{i}{2} \rceil + p - \lceil \frac{i}{2} \rceil} \binom{2^k}{i}\binom{2^k}{2p-i} x^{2p} \tag{8.30}$$

$$= \sum_{i=0}^{2p} (-1)^{p} \binom{2^k}{i}\binom{2^k}{2p-i} x^{2p} \tag{8.31}$$

$$= \left[\sum_{i=0}^{2p} \binom{2^k}{i}\binom{2^k}{2p-i}\right] (-1)^p \, x^{2p} \tag{8.32}$$

$$= (-1)^p \binom{2^{k+1}}{2p} x^{2p}. \tag{8.33}$$

If $r = 2p+1$ for some $0 \leq p \leq 2^k - 1$, we have

$$\left[\sum_{i+j=2p+1} (-1)^{\lceil \frac{i}{2} \rceil + \lfloor \frac{j}{2} \rfloor} \binom{2^k}{i}\binom{2^k}{j}\right] x^{2p+1} \tag{8.34}$$

$$= \sum_{i=0}^{2p+1} (-1)^{\lceil \frac{i}{2} \rceil + \lfloor \frac{2p+1-i}{2} \rfloor} \binom{2^k}{i}\binom{2^k}{2p+1-i} x^{2p+1} \tag{8.35}$$

$$= \sum_{i=0}^{p} \left[(-1)^{\lceil \frac{i}{2} \rceil + \lfloor \frac{2p+1-i}{2} \rfloor} \binom{2^k}{i}\binom{2^k}{2p+1-i} x^{2p+1}\right] \tag{8.36}$$

$$+ \sum_{i=0}^{p} \left[(-1)^{\lceil \frac{2p+1-i}{2} \rceil + \lfloor \frac{i}{2} \rfloor} \binom{2^k}{2p+1-i}\binom{2^k}{i} x^{2p+1}\right] \tag{8.37}$$

$$= \sum_{i=0}^{p} \left[(-1)^{\lceil \frac{i}{2} \rceil + p + \lfloor -\frac{i-1}{2} \rfloor} \binom{2^k}{i}\binom{2^k}{2p+1-i} x^{2p+1}\right] \tag{8.38}$$

$$+ \sum_{i=0}^{p} \left[(-1)^{p + \lceil -\frac{i-1}{2} \rceil + \lfloor \frac{i}{2} \rfloor} \binom{2^k}{2p+1-i}\binom{2^k}{i} x^{2p+1}\right] \tag{8.39}$$

$$= \sum_{i=0}^{p} \left[(-1)^{\lceil \frac{i}{2} \rceil + p - \lceil \frac{i-1}{2} \rceil} \binom{2^k}{i}\binom{2^k}{2p+1-i} x^{2p+1}\right] \tag{8.40}$$

$$+ \sum_{i=0}^{p} \left[(-1)^{p - \lfloor \frac{i-1}{2} \rfloor + \lfloor \frac{i}{2} \rfloor} \binom{2^k}{2p+1-i}\binom{2^k}{i} x^{2p+1}\right]. \tag{8.41}$$

Since exactly one of $\frac{i-1}{2}$ and $\frac{i}{2}$ is integer, we observe that

$$\left(\left\lceil\frac{i}{2}\right\rceil+p-\left\lceil\frac{i-1}{2}\right\rceil\right)+\left(p-\left\lfloor\frac{i-1}{2}\right\rfloor+\left\lfloor\frac{i}{2}\right\rfloor\right) \tag{8.42}$$

$$=2p+\left(\left\lceil\frac{i}{2}\right\rceil+\left\lfloor\frac{i}{2}\right\rfloor\right)-\left(\left\lceil\frac{i-1}{2}\right\rceil+\left\lfloor\frac{i-1}{2}\right\rfloor\right) \tag{8.43}$$

is odd. Hence,

$$(-1)^{\lceil\frac{i}{2}\rceil+p-\lceil\frac{i-1}{2}\rceil}\binom{2^k}{i}\binom{2^k}{2p+1-i}x^{2p+1}+(-1)^{p-\lfloor\frac{i-1}{2}\rfloor+\lfloor\frac{i}{2}\rfloor}\binom{2^k}{2p+1-i}\binom{2^k}{i}x^{2p+1} \tag{8.44}$$

$$=0 \tag{8.45}$$

for all $1\leq i\leq p$, which means that

$$\left[\sum_{i+j=2p+1}(-1)^{\lceil\frac{i}{2}\rceil+\lfloor\frac{j}{2}\rfloor}\binom{2^k}{i}\binom{2^k}{j}\right]x^{2p+1}=0. \tag{8.46}$$

Hence,

$$\text{Denominator}=\sum_{p=0}^{2^k}(-1)^p\binom{2^{k+1}}{2p}x^{2p}. \tag{8.47}$$

By equations (8.26) and (8.47), we have

$$f^{k+1}(x)=\frac{\sum_{i=0}^{2^k-1}(-1)^i\binom{2^{k+1}}{2i+1}x^{2i+1}}{\sum_{j=0}^{2^k}(-1)^j\binom{2^{k+1}}{2j}x^{2j}}. \tag{8.48}$$

□

Lemma 24. $[\mathbb{Q}(\tan\frac{2\pi}{2^l}):\mathbb{Q}]=2^{l-3}$ *for* $l\geq 3$.

Proof. We use induction. When $l=3$, we have $\left[\mathbb{Q}(\tan\frac{\pi}{4})\colon\mathbb{Q}\right]=1=2^{3-3}$.

Now, we assume that $\left[\mathbb{Q}(\tan\frac{2\pi}{2^l})\colon \mathbb{Q}\right] = 2^{l-3}$ and consider

$$\left[\mathbb{Q}\left(\tan\frac{2\pi}{2^{l+1}}\right) : \mathbb{Q}\right] = \left[\mathbb{Q}\left(\tan\frac{2\pi}{2^{l+1}}\right) : \mathbb{Q}\left(\tan\frac{2\pi}{2^{l}}\right)\right] \cdot \left[\mathbb{Q}\left(\tan\frac{2\pi}{2^{l}}\right) : \mathbb{Q}\right] \tag{8.49}$$

$$= \left[\mathbb{Q}\left(\tan\frac{2\pi}{2^{l+1}}\right) : \mathbb{Q}\left(\tan\frac{2\pi}{2^{l}}\right)\right] \cdot 2^{l-3}. \tag{8.50}$$

The double angle formula gives us

$$\tan\frac{2\pi}{2^l} = \frac{2\tan\frac{2\pi}{2^{l+1}}}{1-\left(\tan\frac{2\pi}{2^{l+1}}\right)^2} \Rightarrow \left(\tan\frac{2\pi}{2^{l+1}}\right)^2 + \frac{2}{\tan\frac{2\pi}{2^l}}\tan\frac{2\pi}{2^{l+1}} - 1 = 0. \tag{8.51}$$

By the quadratic formula, we have

$$\tan\frac{2\pi}{2^{l+1}} = \frac{-\frac{2}{\tan\frac{2\pi}{2^l}} + \sqrt{\frac{4}{\left(\tan\frac{2\pi}{2^l}\right)^2}+4}}{2} = \frac{-1+\sqrt{1+\left(\tan\frac{2\pi}{2^l}\right)^2}}{\tan\frac{2\pi}{2^l}} = \frac{-1+\sec\frac{2\pi}{2^l}}{\tan\frac{2\pi}{2^l}} \tag{8.52}$$

We want to show that $\tan\frac{2\pi}{2^{l+1}} \notin \mathbb{Q}(\tan\frac{2\pi}{2^l})$ by contradiction. Assume $\tan\frac{2\pi}{2^{l+1}} \in \mathbb{Q}(\tan\frac{2\pi}{2^l})$. Then

$$\sec\frac{2\pi}{2^l} = \tan\frac{2\pi}{2^{l+1}}\cdot\tan\frac{2\pi}{2^l} + 1 \in \mathbb{Q}\left(\tan\frac{2\pi}{2^l}\right) \Rightarrow \cos\frac{2\pi}{2^l} \in \mathbb{Q}\left(\tan\frac{2\pi}{2^l}\right), \tag{8.53}$$

which implies that

$$\left[\mathbb{Q}\left(\cos\frac{2\pi}{2^l}\right) : \mathbb{Q}\right] \leq \left[\mathbb{Q}\left(\tan\frac{2\pi}{2^l}\right) : \mathbb{Q}\right] = 2^{l-3}. \tag{8.54}$$

However, by Lemma 25 (showed below), we have the $\left[\mathbb{Q}(\cos\frac{2\pi}{2^l}) : \mathbb{Q}\right] = 2^{l-2} > 2^{l-3}$, which is a contradiction. Thus, $\left[\mathbb{Q}(\tan\frac{2\pi}{2^{l+1}}) : \mathbb{Q}\right] = 2\cdot 2^{l-3} = 2^{(l+1)-3}$. □

Lemma 25. $[\mathbb{Q}(\cos\frac{2\pi}{2^l}) : \mathbb{Q}] = 2^{l-2}$ *for* $l \geq 2$.

Proof. For $l \geq 2$, set

$$\xi_l = e^{\imath \frac{2\pi}{2^l}} = \cos\frac{2\pi}{2^l} + i\sin\frac{2\pi}{2^l}, \tag{8.55}$$

and note that $[\mathbb{Q}(\xi_l) : \mathbb{Q}] = 2^{l-1}$. Then,

$$\frac{\xi_l + \xi_l^{-1}}{2} = \cos\frac{2\pi}{2^l} \in \mathbb{Q}(\xi_l). \tag{8.56}$$

Hence, $\mathbb{Q} \subset \mathbb{Q}(\cos\frac{2\pi}{2^l}) \subset \mathbb{Q}(\xi_l)$ and ξ_l is a root of

$$x^2 - 2\cos\frac{2\pi}{2^l}x + 1 = 0 \in \mathbb{Q}\left(\cos\frac{2\pi}{2^l}\right)[x]. \tag{8.57}$$

Now, we have

$$2^{l-1} = [\mathbb{Q}(\xi_l) : \mathbb{Q})] = \left[\mathbb{Q}(\xi_l) : \mathbb{Q}\left(\cos\frac{2\pi}{2^l}\right)\right] \cdot \left[\mathbb{Q}\left(\cos\frac{2\pi}{2^l}\right) : \mathbb{Q}\right]. \tag{8.58}$$

Note that $i \in \mathbb{Q}(\xi_l)$ and $i \notin \mathbb{Q}(\cos\frac{2\pi}{2^l})$, $[\mathbb{Q}(\xi_l) : \mathbb{Q}(\cos\frac{2\pi}{2^l})] > 1$. Then, the equation (8.57) is the minimal polynomial in $\mathbb{Q}(\cos\frac{2\pi}{2^l})$ of ξ_l, we have $[\mathbb{Q}(\xi_l) : \mathbb{Q}(\cos\frac{2\pi}{2^l})] = 2$. Thus,

$$\left[\mathbb{Q}\left(\cos\frac{2\pi}{2^l}\right) : \mathbb{Q}\right] = \frac{[\mathbb{Q}(\xi_l) : \mathbb{Q})]}{\left[\mathbb{Q}(\xi_l) : \mathbb{Q}\left(\cos\frac{2\pi}{2^l}\right)\right]} = \frac{2^{l-1}}{2} = 2^{l-2}, \tag{8.59}$$

which completes the proof. □

8.2 Minimal Polynomial and Gleason's Theorem

Theorem 26. *Let $\alpha_l = \tan\frac{2\pi}{2^l}$ for some $l \geq 3$. The minimal polynomial of α_l over $\mathbb{Q}$ is*

$$p_l(x) = \sum_{t=0}^{2^{l-3}} (-1)^{\lceil \frac{t}{2} \rceil} \binom{2^{l-3}}{t} x^t \in \mathbb{Q}[x]. \tag{8.60}$$

Proof. Consider the double angle formula $\tan 2\alpha = \frac{2\tan\alpha}{1-\tan^2\alpha}$. Let $f(x) = \frac{2x}{1-x^2}$. Then

we have $f^{l-3}(\alpha_l) = \tan(2^{l-3}\alpha_l) = \tan(\frac{2\pi}{2^3}) = 1$. After applying Lemma 23 we have

$$1 = f^{l-3}(\alpha_l) = f^k(x) = \frac{\sum_{i=0}^{2^{k-1}-1}(-1)^i\binom{2^k}{2i+1}(\alpha_l)^{2i+1}}{\sum_{j=0}^{2^{k-1}}(-1)^j\binom{2^k}{2j}(\alpha_l)^{2j}}. \tag{8.61}$$

After rearranging terms we have

$$0 = \sum_{j=0}^{2^{k-1}}(-1)^j\binom{2^k}{2j}(\alpha_l)^{2j} - \sum_{i=0}^{2^{k-1}-1}(-1)^i\binom{2^k}{2i+1}(\alpha_l)^{2i+1} \tag{8.62}$$

$$= \sum_{t=0}^{2^{l-3}}(-1)^{\lceil\frac{t}{2}\rceil}\binom{2^{l-3}}{t}(\alpha_l)^t = p_l(\alpha_l). \tag{8.63}$$

Therefore, α_l is a root of p_l. Moreover, by Lemma 24, we have $\deg p_l = 2^{l-3} = [\mathbb{Q}(\alpha_l) : \mathbb{Q}]$. Hence, p_l is the minimal polynomial of α_l over $\mathbb{Q}$ for $l \geq 3$. □

Remark 27. If $p_l(x)$ is the minimal polynomial of α_l, then $p_l(-x)$ is the minimal polynomial of $-\alpha_l$ since $[\mathbb{Q}(\alpha_l) : \mathbb{Q}] = [\mathbb{Q}(-\alpha_l) : \mathbb{Q}] = \deg p_l(x)$. Theorem 26 shows that $p_l(x)$ has a root of $\alpha_l = \tan\frac{2\pi}{2^l}$. We can use the same iterative method of field extensions to show that $p_l(x)$ has roots $S_l = \{\tan\frac{k\cdot 2\pi}{2^l} : k = 1 \pmod 4 \text{ and } 1 \leq k \leq 2^{l-1} - 3\}$. Similarly, we can check that $p_l(-x)$ has roots $S_l' = \{\tan\frac{k\cdot 2\pi}{2^l} : k = 3 \pmod 4 \text{ and } 3 \leq k \leq 2^{l-1} - 1\}$.

We now show that the polynomial $R_j(x)$ is divisible by the square of the minimal polynomials of α_3 and $-\alpha_3$. The first step is to show that the coefficients of $R_j(x)$ are symmetric.

Lemma 28. *For each Z_j, the coefficients of $R_j(x)$ are symmetric, that is*

$$\sum_{v\in Z_j(2t)}\epsilon_v\,(-1)^t - \binom{\frac{m_j}{2}}{t} = \sum_{w\in Z_j(m_j-2t)}\epsilon_w\,(-1)^{\frac{m_j}{2}-t} - \binom{\frac{m_j}{2}}{\frac{m_j}{2}-t}. \tag{8.64}$$

Proof. Let $v \in Z_j(2t)$ and we can write $v = w \oplus \underline{1}_{m_j}$, for some $w \in Z_j\,(m_j - 2t)$.

After making the substitution for v in terms of w, we have

$$\sum_{v\in Z_j(2t)} \epsilon_v \left(-1\right)^t - \binom{\frac{m_j}{2}}{t} = \sum_{w\in Z_j(m_j-2t)} \epsilon_{w\oplus \underline{1}_{m_j}} \left(-1\right)^t - \binom{\frac{m_j}{2}}{t} \tag{8.65}$$

$$= \sum_{w\in Z_j(m_j-2t)} \epsilon_w \epsilon_{\underline{1}_{m_j}} \left(-1\right)^t - \binom{\frac{m_j}{2}}{t}, \tag{8.66}$$

where the last step follows by the facts that the ϵ is multiplicative. Note that $\underline{1}_{m_j} \in Z_j^{\perp}$ since all vectors in Z_j have even Hamming weight. By the third necessary condition in Remark 7, we have $\epsilon_{\underline{1}_{m_j}} = (-1)^{\frac{m_j}{2}}$. Thus, $\epsilon_{\underline{1}_{m_j}}(-1)^t = (-1)^{\frac{m_j}{2}+t} = (-1)^{\frac{m_j}{2}-t}$ and it follows from the symmetry of binomial coefficients that

$$\sum_{w\in Z_j(m_j-2t)} \epsilon_w \epsilon_{\underline{1}_{m_j}} \left(-1\right)^t - \binom{\frac{m_j}{2}}{t} = \sum_{w\in Z_j(m_j-2t)} \epsilon_w \left(-1\right)^{\frac{m_j}{2}-t} - \binom{\frac{m_j}{2}}{\frac{m_j}{2}-t}. \tag{8.67}$$

Combining (8.65) and (8.67), we obtain (8.64) as required. □

Lemma 29. *If $\alpha_3 = \tan\frac{\pi}{4} = 1$ is a root of $R_j(x)$. Then α_3 has multiplicity of at least 2. The same holds for $-\alpha_3$.*

Proof. Let $D = \deg R_j(x)$. Lemma 28 implies $R_j\left(\frac{1}{x}\right) x^D = R_j(x)$, and taking derivatives of both sides we obtain

$$-R_j'\left(\frac{1}{x}\right)\cdot\frac{1}{x^2}\cdot x^D + R_j\left(\frac{1}{x}\right)\cdot D\cdot x^{D-1} = R_j'(x). \tag{8.68}$$

By assumption, we have $R_j(1) = 0$. After substituting $x = 1$, we have $-R_j'(1) = R_j'(1)$, which implies that $R_j'(1) = 0$. Similarly, we can show $R_j'(-1) = 0$. Thus, if α_3 and $-\alpha_3$ are roots of $R_j(x)$, then they have multiplicity at least 2. □

Remark 30. Note that x^2 always divides $R_j(x)$ since all powers of x in (8.15) are even. Given a stabilizer code $V(S)$ preserved by transversal $\frac{\pi}{2^l}$ Z-rotation for $l \leq$

$l_{\max} < \infty$, it follows from Theorems 5 and 26, and from Lemma 29 that $R_j(x)$ is divisible by $x^2(x-1)^2(x+1)^2 \prod_{l=4}^{l_{\max}} p_l(x)p_l(-x)$. Note that $(x-1)^2(x+1)^2 = (p_3(x)p_3(-x))^2$.

Corollary 31 (Connecting to Gleason's Theorem)**.** *Let S define a stabilizer code $V(S)$ that is preserved by (finitely many) transversal applications of $\exp(\frac{\imath\pi}{2^l}\sigma_Z)$, with $l \leq l_{\max} < \infty$. If there exists a stabilizer $\epsilon_j E(a_j, b_j)$ with $a_j \neq 0$ such that $Z_j = \{\tilde{z}\big|_{\mathrm{supp}(a_j)} : \epsilon_{\tilde{z}} E(0, \tilde{z}) \in S \text{ and } \tilde{z} \preceq a_j\}$ is self-dual, then the weight enumerator of Z_j is*

$$P_{Z_j}(x, y) = (x^2 + y^2)^{\frac{m_j}{2}} + x^2y^2(x^2 - y^2)^2 h(x, y), \tag{8.69}$$

where $h(x, y) \in \mathbb{Q}[x, y]$.

Proof. Based on Remark 30, we know that the corresponding $R(x)$ is divisible by the factor $x^2(x-1)^2(x+1)^2$, i.e.,

$$R_j(x) = \sum_{t=0}^{\frac{m_j}{2}} \left[\sum_{v \in Z_j(2t)} \epsilon_v (-1)^t - \binom{\frac{m_j}{2}}{t} \right] x^{2t} = x^2(x-1)^2(x+1)^2 h(x) \tag{8.70}$$

for some $h(x) \in \mathbb{Q}[x]$. Note that Z_j is self-dual, i.e., $Z_j = Z_j^{\perp}$. It follows from the third condition in Remark 7 that $\epsilon_v = \imath^{w_H(v)} = (-1)^t$ for all $v \in Z_j$. Thus, we can rewrite (8.70) as

$$R_j(x) = \sum_{t=0}^{\frac{m_j}{2}} \left[|Z_j(2t)| - \binom{\frac{m_j}{2}}{t} \right] x^{2t} = x^2(x-1)^2(x+1)^2 h(x). \tag{8.71}$$

Let $D = \deg R_j(x)$. Then, we have $\frac{m_j}{2} + 2 \leq D \leq m_j - 2$ and $\deg h(x) = D - 6$.

Then,

$$R_j(x) = \sum_{t=\frac{m-D}{2}}^{\frac{D}{2}} \left[|Z_j(2t)| - \binom{\frac{m_j}{2}}{t} \right] x^{2t} = x^2(x-1)^2(x+1)^2 h(x). \tag{8.72}$$

Note that $x^{m_j-D} | R_j(x) = x^2(x-1)^2(x+1)^2 h(x)$ but $x^{m_j-D} \nmid R_j(x)$, which implies that x^{m_j-D-2} is the factor of $h(x)$ with the highest degree in x. Assume $h(x) = x^{m_j-d-2} l(x)$, where $\deg l(x) = d - 6 - (m_j - d - 2) = 2d - m_j - 4$ and $x \nmid l(x)$. Replacing x by $\frac{y}{x}$ and multiplying both side by x^{m_j} in (8.71), we have

$$\sum_{t=\frac{m-D}{2}}^{\frac{D}{2}} \left[|Z_j(2t)| - \binom{\frac{m_j}{2}}{t} \right] x^{m_j-2t} y^{2t} = x^{m_j-8} x^2 y^2 (y-x)^2 (y+x)^2 \left(\frac{y}{x}\right)^{m_j-d-2} l\left(\frac{y}{x}\right), \tag{8.73}$$

which implies that

$$\sum_{t=0}^{\frac{m_j}{2}} \left[|Z_j(2t)| - \binom{\frac{m_j}{2}}{t} \right] x^{m_j-2t} y^{2t} = x^2 y^2 (y-x)^2 (y+x)^2 x^{m_j-d-2} y^{m_j-d-2} x^{2d-m_j-4} l\left(\frac{y}{x}\right). \tag{8.74}$$

Note that $P_{Z_j}(x,y) = \sum_{t=0}^{\frac{m_j}{2}} |Z_j(2t)| \cdot x^{m_j-2t} y^{2t}$, we have

$$P_{Z_j}(x,y) = (x^2+y^2)^{\frac{m_j}{2}} + x^2 y^2 (x^2-y^2)^2 h(x,y), \tag{8.75}$$

where $h(x,y) = x^{m_j-d-2} y^{m_j-d-2} x^{2d-m_j-4} l(\frac{y}{x})$. Note that $\deg l(x) = 2d - m_j - 4$ and $x \nmid l(x)$, we have $h(x,y) \in \mathbb{Q}[x,y]$.

□

Remark 32. Since Z_j is self-dual, it follows from Theorem 22 that $P_{Z_j}(x,y)$ is a sum of products of Gleason's polynomials $f(x,y)$ and $g(x,y)$ according to divisibility of weights. As divisible by 4 is a special case of divisible by 2, we choose the general

case that $f(x,y) = x^2 + y^2$ and $g(x,y) = x^2y^2(x^2 - y^2)^2$. Then, we rewrite (8.69) as

$$P_{Z_j}(x,y) - (f(x,y))^{\frac{m_j}{2}} = g(x,y)h(x,y), \tag{8.76}$$

which implies that $g(x,y)h(x,y)$ is a sum of products of $f(x,y)$ and $g(x,y)$, i.e. $g(x,y)h(x,y) = \sum_{i=1}^{T} c_i \left(f(x,y)\right)^{\sigma_i} \left(g(x,y)\right)^{\xi_i}$, with $c_i \neq 0$. Note that $S = \{(x,y) \subset \mathbb{R}^2 : x = 0\}$ is a set of roots for $g(x,y)$ but not for $f(x,y)$. Thus, $g(x,y)$ cannot divide a nonzero polynomial that is purely in terms of $f(x,y)$, which implies that $\xi_i > 0$ for all i. Thus, $h(x,y)$ is a sum of products of $f(x,y)$ and $g(x,y)$, which implies that $h(x,y) = h(y,x)$. Equivalently, $h(x)$ is a sum of products of $(1+x^2)$ and $x^2(x-1)^2(x+1)^2$.

Remark 33. By Remark 30, we know that if $l_{\max} \geq 4$, we can determine more factors of $R_j(x)$. By following the same procedures, we can obtain a generalized version of (8.69) as

$$P_{Z_j}(x,y) = (x^2 + y^2)^{\frac{m_j}{2}} + x^2y^2(x^2 - y^2)^2 h'(x,y) \prod_{l=4}^{l_{\max}} p_l(x,y)p_l(-x,y), \tag{8.77}$$

for some $h'(x,y) \in \mathbb{Q}[x,y]$, where $p_l(x,y) = x^{2^{l-3}} p_l(\frac{y}{x})$.

Through the computation of (8.71) for each Z_j, Examples 10 and 11 illustrate how Corollary 31 and the property in Remark 32 work for self-dual Z_j's of different stabilizer codes invariant under transversal T. The term $h(x)$ in (8.71) provides the freedom in $R_j(x)$, and it can be either trivial (Example 10) or non-trivial (Example 11). Examples 5(Continued) and 12 indicate that the divisibility of $R_j(x)$ still hold even if Z_j is not self-dual.

Example 10. Consider the $[[8,3,2]]$ color code [Cam16, RCNP20], CSS$(X, \langle\ \underline{1}_8\rangle$; $Z, \mathrm{RM}(1,3))$, and the $[[15,1,3]]$ punctured quantum Reed-Muller code [BK05,RCNP20],

$\mathrm{CSS}(X, C_2; Z, C_1^{\perp})$, where C_2 is generated by the degree one monomials, x_1, x_2, x_3, x_4, and

$$C_1^{\perp} = \langle x_1, x_2, x_3, x_4, x_1x_2, x_1x_3, x_1x_4, x_2x_3, x_2x_4, x_3x_4 \rangle,$$

with the first coordinate removed in both C_2 and $C_1^{\perp}$. Since the signs of all stabilizers are positive, Theorem 5 and Remark 7 imply that both are invariant under transversal T but not transversal $\sqrt{T}$ ([RCNP20]). There are 15 non-zero X-stabilizers in the $[[15, 1, 3]]$ code, and in each case the corresponding Z_j is $\mathrm{RM}(1, 3)$. There is a single X-stabilizer $a_1 = \underline{1}_8$ in the $[[8, 3, 2]]$ code, and again the corresponding Z_1 is $\mathrm{RM}(1, 3)$, with weight enumerator

$$A_{\mathrm{RM}(1,3)}(x) = 1 + 14x^4 + x^8. \tag{8.78}$$

The character $\epsilon_v = 1$ for all $v \in \mathrm{RM}(1, 3)$ so $R_1(x)$ is given by

$$R_1(x) = -4(x^2 - 2x^4 + x^6) = x^2(x - 1)^2(x + 1)^2 h(x), \tag{8.79}$$

where $h(x) = -4$. Note that $h(x)$ must be constant for any length 8 code Z_j arising from a stabilizer code invariant under transversal T.

Example 11. We construct a $[[16, 7, 2]]$ code by removing half of the degree two monomials in Z-stabilizers from the $[[16, 4, 2]]$ code presented in Example 5. This yields the $\mathrm{CSS}(X, \underline{1}_{16}; Z, \mathrm{RM}(1.5, 4))$ code with the signs of all stabilizers being positive, where $\mathrm{RM}(1.5, 4)$ is the self-dual code generated by $\underline{1}_{16}$, all the degree 1 monomials, and the degree 2 monomials x_1x_2, x_1x_3, x_1x_4. It is invariant under transversal T but not under transversal $\sqrt{T}$, i.e., $l_{\max} = 3$. The weight enumerator of the only

$Z_1 = \mathrm{RM}(1.5, 4)$ of $[[16, 7, 2]]$ is

$$A_{Z_1}(x) = 1 + 28x^4 + 198x^8 + 28x^{12} + x^{16}. \tag{8.80}$$

Note that $\epsilon_v = 1$ for all $v \in Z_1$, we simplify $R_1(x)$ as

$$R_1(x) = -8(x^2 + 7x^6 - 16x^8 + 7x^{10} + x^{14}) = x^2(x-1)^2(x+1)^2 h(x), \tag{8.81}$$

where $h(x) = -8(x^8 + 2x^6 + 10x^4 + 2x^2 + 1) = -8\left[(x^2+1)^4 - 2x^2(x-1)^2(x+1)^2\right]$, which is non-trivial.

Example 12. The $[[16, 3, 2]]$ code is a $\mathrm{CSS}(X, C_2; Z, C_1^\perp)$ code constructed in [RCNP20], where $C_2 = \langle \underline{1}_{16}, x_1, x_2 \rangle$ and $C_1^\perp = \langle \underline{1}_{16}, x_1, x_2, x_3, x_4\ , x_1x_2, x_1x_3, x_1x_4, x_2x_3, x_2x_4 \rangle$. By verifying the three conditions in Remark 7, we know that the codespace is preserved by transversal T. Note that $\tan \frac{2\pi}{2^4}$ does not satisfy (2.31), so the codespace is not preserved by transversal $\sqrt{T}$. There are two types of Z_j among the 7 non-zero X-stabilizers a_j. The first $Z_1 = C_1^\perp$ is corresponding to $a_1 = \underline{1}_{16}$ is not self-dual as $C_1 = \langle \underline{1}_{16}, x_1, x_2, x_3, x_4, x_1x_2 \rangle$. By symmetry of monomials with the same order, the remaining $Z_2, \cdots, Z_7$ are all $\mathrm{RM}(1, 3)$, which was already discussed in Example 10. The weight distribution of Z_1 is

$$A_{Z_1}(x) = 1 + 76x^4 + 192x^6 + 486x^8 + 192x^{10} + 76x^{12} + x^{16}. \tag{8.82}$$

With the trivial signs, (8.15) becomes

$$R_1(x) = -8(x^2 - 6x^4 + 31x^6 - 52x^8 + 31x^{10} - 6x^{12} + x^{16}) = x^2(x-1)^2(x+1)^2 h(x), \tag{8.83}$$

where $h(x) = -8(x^8 - 4x^6 + 22x^4 - 4x^2 + 1) = -8[(x^2+1)^4 - 8x^2(x-1)^2(x+1)^2]$.

Example 1 (revisited)**.** Recall the $[[16, 4, 2]]$ CSS code with X-stabilizer $\langle \underline{1}_{16} \rangle$ and Z-stabilizer RM$(2, 4)$. The dual of RM$(2, 4)$ is RM$(1, 4)$, which means that the only $Z_1 = \text{RM}(2, 4)$ corresponding to the $a_1 = \underline{1}_{16}$ is not self-dual. As verified in Section 3, we know that the code is invariant under the application of transversal $\frac{\pi}{2^l}$ with $l \leq 4$. Note that for all $v \in Z_1$, $\epsilon_v = 1$. It follows from the weight enumerator in (4.10) that

$$R_1(x) = -8(x^2 - 14x^4 + 63x^6 - 100x^8 + 63x^{10} - 14x^{12} + x^{14}) \tag{8.84}$$

$$= -8x^2(p_3(x))^2(p_3(-x))^2(p_4(x))^2(p_4(-x))^2, \tag{8.85}$$

where $p_3(x) = x-1$, $p_3(-x) = -x-1$, $p_4(x) = x^2+2x-1$, and $p_4(-x) = x^2-2x-1$ are the minimal polynomials of $\tan\frac{2\pi}{2^3}$, $\tan -\frac{2\pi}{2^3}$, $\tan\frac{2\pi}{2^4}$, and $\tan -\frac{2\pi}{2^4}$ respectively. Here, we have $h(x) = (p_4(x)p_4(-x))^2 = (x^2+1)^4 - 16x^2(x-1)^2(x+1)^2$.

It is interesting to see in Example 5 that the square of the product of minimal polynomials of $\tan\frac{2\pi}{2^4}$ and $-\tan\frac{2\pi}{2^4}$, i.e., $(p_4(x)p_4(-x))^2$, divides $R_1(x)$. In this vein, we also computed $R_j(x)$ corresponding to the only $Z_j = \text{RM}(3, 5)$ associated with the $[[32, 5, 2]]$ CSS$(X, \langle \underline{1}_{32} \rangle; Z, \text{RM}(3, 5))$ code in the QRM $[[2^m, \binom{m}{1}, 2]]$ family constructed in [RCNP20]. We know from [RCNP20, Theorem 19] that the code space is fixed under transversal $T^{\frac{1}{4}}$ ($\frac{\pi}{2^5}$ Z-rotation), i.e., $l_{\max} = 5$. The polynomial $R_j(x) = -16x^2\prod_{l=3}^{5}(p_l(x)p_l(-x))^2$ continues to be divisible by squares.

We may get some intuition about the appearance of the squares in the minimal polynomials from a physical perspective. If a stabilizer code is invariant under transversal $\pi/2^{l_{\max}}$ Z-rotation, then it is also preserved by transversal $i\pi/2^{l_{\max}}$ Z-rotation for $i = 0, ..., 2^l - 1$. It follows from Theorem 5 that $R_j(x)$ has roots $\tan(2k\pi/2^{l_{\max}})$ for $k = \{0, 1, ..., 2^{l_{\max}} - 1\} \setminus \{2^{l_{\max}-2}, 3 \cdot 2^{l_{\max}-2}\}$. Note that $\tan x$

has period of π, which implies that $\tan(2k\pi/2^{l_{\max}}) = \tan(2(k + 2^{l_{\max}-1})\pi/2^{l_{\max}})$. The physical $i\pi/2^{l_{\max}}$ and $(i + 2^{l_{\max}-1})\pi/2^{l_{\max}}$ Z-rotations are different, which indicates that each of the roots $\tan(2k\pi/2^{l_{\max}})$ with $k = \{0, 1, ..., 2^{l_{\max}-1} - 1\} \setminus \{2^{l_{\max}-2}\}$ in $R(x)$ appears twice. Mathematically, if $\tan(2k\pi/2^{l_{\max}})$ is a root of $R(x)$, then $\tan(2(k+2^{l_{\max}-1})\pi/2^{l_{\max}})$ is automatically a root, which means that we need to come up with a new way to show the existence of squares.

If we could show that the multiplicity of roots corresponding to each of the minimal polynomials $p_l(x), p_l(-x)$, with $l = 3, ..., l_{\max}$, are at least 2, then

$$x^2 \prod_{i=3}^{l_{\max}} (p_i(x)p_i(-x))^2 \tag{8.86}$$

divides $R_j(x)$. We also know that $\deg(x^2 \prod_{i=3}^{l_{\max}} (p_i(x)p_i(-x))^2) = 2^{l_{\max}} - 2 \leq \deg R_j(x) \leq m_j - 2$. Thus, when $m_j = 2^{l_{\max}}$, we conjecture that $R_j(x) = x^2 \prod_{i=3}^{l_{\max}} (p_i(x)p_i(-x))^2$ up to some constant and the weight enumerator of Z_j is restricted, as follows.

Conjecture 1. *Assume S defines a stabilizer code $V(S)$ which is preserved by finitely many transversal applications of $\exp(\frac{i\pi}{2^l}\sigma_Z)$, with $l \leq l_{\max}$. If there is a Z_j with $m_j = 2^{l_{max}}$, then the signs of Z-stabilizers in Z_j are trivially one and the weight distribution of Z_j is fixed once the dimension of Z_j is fixed.*

Here, we show that the special case $l_{\max} = 3$ of Conjecture 1 holds true.

Proof of Conjecture 1 when $l_{\max} = 3$. Let $V(S)$ be a stabilizer code which is invariant under the application of transversal T but is not invariant under application of transversal $\exp(\frac{\pi}{2^l}\sigma_Z)$ with $l \geq 4$. Let Z_j be the space of Z-stabilizers supported on a nonzero X stabilizer with weight 8, i.e., $m_j = 2^3$. Note that $\deg R_j(x) \leq m_j - 2 = 6$.

It follows from Theorem 5, Theorem 26, and Lemma 29 that

$$R_j(x) = \sum_{t=0}^{4} \left[\sum_{v \in Z_j(2t)} \epsilon_v \, (-1)^t - \binom{4}{t} \right] x^{2t} = cx^2(x-1)^2(x+1)^2 = c(x^2 - 2x^4 + x^6), \tag{8.87}$$

for some constant $c \in \mathbb{Q}$, where $Z_j(2t)$ is the set of vectors in Z_j with Hamming weight $2t$. Let $\gamma = \dim Z_j$. If ϵ_v are half 1 and half -1 for $v \in Z_j$, then we have the following system of equations

$$\begin{cases} \frac{-\sum_{v \in Z_j(2)} \epsilon_v - \binom{4}{1}}{\sum_{v \in Z_j(4)} \epsilon_v - \binom{4}{2}} = \frac{-(p_2 - n_2) - 4}{(p_4 - n_4) - 6} = -\frac{1}{2} \\ 2p_2 + p_4 = 2^{\gamma-1} - 2 \\ 2n_2 + n_4 = 2^{\gamma-1} \end{cases}, \tag{8.88}$$

where p_k (resp., n_k) are the number of vectors with Hamming weight k in Z_j associating with positive signs (resp., negative signs). After solving for (8.88), we have $p_2 - n_2 = -4$ and $p_4 - n_4 = 6$, which leads to $R(x) = 0$, contradicting to the fact that S is invariant under finitely many applications of transversal small angle Z-rotations. Thus, the only valid case is that $\epsilon_v = 1$ for all $v \in Z_j$, then we have

$$\frac{-\sum_{v \in Z_j(2)} \epsilon_v - \binom{4}{1}}{\sum_{v \in Z_j(4)} \epsilon_v - \binom{4}{2}} = \frac{-Z_j(2) - 4}{Z_j(4) - 6} = -\frac{1}{2}, \tag{8.89}$$

and

$$2Z_j(2) + Z_j(4) = 2^{\gamma} - 2, \tag{8.90}$$

which implies that $Z_j(2) = 2^{\gamma-2} - 4$, and $Z_j(4) = 2^{\gamma-1} + 6$. Thus, for a given dimension of Z_j, the weight enumerator of Z_j is fixed as $A_{Z_j}(x) = 1 + (2^{\gamma-2} - 4)x^2 + (2^{\gamma-1} + 6)x^4 + (2^{\gamma-2} - 4)x^8$ with the all-one signs of Z-stabilizer in Z_j. □

Remark 34. *To generalize the proof for $l_{\max} \geq 4$, first we need an argument for*

the squaring of the minimal polynomials for $l \geq 4$, and then we need to understand their signs. This we leave to future work. If the above conjecture is true, then it provides an explicit formula for the weight enumerators of Reed-Muller codes in the QRM $[[2^m, \binom{m}{1}, 2]]$ family [RCNP20] satisfying $m_j = 2^{l_{\max}}$ (i.e., weight of the all $1s$ X-stabilizer).

Chapter 9

Conclusion

In this thesis, we derived sufficient conditions on the Hamming weights and signs of Z-stabilizers for a stabilizer code to be invariant under the transversal application of $\exp(\imath\theta\sigma_Z)$ for all θ. Using the sufficient conditions we are able to construct a family of CSS codes with a good rate-distance tradeoff that provides a DFS towards coherent Z-errors. In future work, we will explore the realization of a universal set of fault-tolerant logical operations on these codes. Besides the specific family of CSS codes, the sufficient conditions could also help us check whether a general stabilizer code forms a Z-DFS. It remains open to find whether the necessary direction implies that every qubit is covered by some weight-2 Z-stabilizer, and whether the necessary conditions match our sufficient conditions. It also connects to generator coefficient framework [HLC22b], which may lead to the general diagonal error model.

To realize non-identity logical operators in third level or higher in the Clifford hierarchy, we also studied the stabilizer codes which are preserved by finitely many $\pi/2^l$ Z-rotations, for $l \leq l_{\max} < \infty$.

In this case, the identity (2.31) is reduced to a polynomial with factors including the minimal polynomials of $\tan\frac{2\pi}{2^l}, l \leq l_{\max}$. The polynomial provides information about the weight distribution and sign of the binary code formed by the Z-stabilizers supported on each non-zero X-component of stabilizers. When the binary code is self-dual, we made a tight connection to Gleason's theorem (Corollary 31).

Through the weight divisibility conditions in Sections 3 and 4, and the minimal polynomials derived in Theorem 26, we made new connections between quantum

information theory and classical coding theory. Along this direction, one of our main interests for future work is to generalize Corollary 31 by proving Conjecture 1 and/or by removing the self-dual assumption. Besides that, the other future direction is to find a general construction of stabilizer codes that are invariant under finitely many transversal $\frac{\pi}{2^l}$ Z-rotations. Since non-CSS constructions with such properties are extremely sparse in the literature, we think that our work could help break new ground in this regard. For the second direction, it is interesting to investigate whether the identities (2.31) and (2.32) imply the existence of a self-dual code inside Z_j satisfying (2.31) and (2.32), since this may provide us information on how different Z_j's interact with each other.

Bibliography

[ABC+01] G Alber, Th Beth, Ch Charnes, A Delgado, M Grassl, and M Mussinger. Stabilizing distinguishable qubits against spontaneous decay by detected-jump correcting quantum codes. *Phys. Rev. Lett.*, 86(19):4402, 2001.

[ABD+19] Ayomikun Adeniran, Steve Butler, Colin Defant, Yibo Gao, Pamela E Harris, Cyrus Hettle, Qingzhong Liang, Hayan Nam, and Adam Volk. On the genus of a quotient of a numerical semigroup. In *Semigroup Forum*, volume 98, pages 690–700. Springer, 2019.

[ABDB+19] Ayomikun Adeniran, Steve Butler, Galen Dorpalen-Barry, Pamela E Harris, Cyrus Hettle, Qingzhong Liang, Jeremy L Martin, and Hayan Nam. Enumerating parking completions using join and split. *arXiv preprint arXiv:1912.01688*, 2019.

[ACB12] Hussain Anwar, Earl T Campbell, and Dan E Browne. Qutrit magic state distillation. *New J. Phys.*, 14(6):063006, 2012.

[ADCP14] Jonas T Anderson, Guillaume Duclos-Cianci, and David Poulin. Fault-tolerant conversion between the Steane and Reed-Muller quantum codes. *Phys. Rev. Lett.*, 113(8):080501, 2014.

[Ax64] J. Ax. Zeroes of polynomials over finite fields. *Am. J. Math.*, 86:255–261, 1964.

[BH12] Sergey Bravyi and Jeongwan Haah. Magic-state distillation with low overhead. *Phys. Rev. A*, 86(5):052329, 2012.

[BK05] Sergey Bravyi and Alexei Kitaev. Universal quantum computation with ideal clifford gates and noisy ancillas. *Phys. Rev. A*, 71(2):022316, 2005.

[BMP+99] P Oscar Boykin, Tal Mor, Matthew Pulver, Vwani Roychowdhury, and Farrokh Vatan. On universal and fault-tolerant quantum computing: a novel basis and a new constructive proof of universality for shor's basis. In *40th Annu. Symp. Found. Comput. Sci. (Cat. No.99CB37039)*, pages 486–494. IEEE, 1999.

[Bor13] Yuri L Borissov. On Mceliece's result about divisibility of the weights in the binary Reed-Muller codes. In *Seventh International Workshop, Optimal Codes and related topics*, pages 47–52, 2013.

[BWG+18] Stefanie J. Beale, Joel J. Wallman, Mauricio Gutiérrez, Kenneth R. Brown, and Raymond Laflamme. Quantum Error Correction Decoheres Noise. *Phys. Rev. Lett.*, 121(19):190501, 2018.

[CAB12] Earl T Campbell, Hussain Anwar, and Dan E Browne. Magic-state distillation in all prime dimensions using quantum Reed-Muller codes. *Phys. Rev. X*, 2(4):041021, 2012.

[Cam16] Earl T Campbell. The smallest interesting colour code. *Blog post*, 2016.

[CGK17] Shawn X. Cui, Daniel Gottesman, and Anirudh Krishna. Diagonal gates in the Clifford hierarchy. *Phys. Rev. A*, 95(1):012329, 2017.

[CH17] Earl T Campbell and Mark Howard. Unified framework for magic state distillation and multiqubit gate synthesis with reduced resource cost. *Phys. Rev. A*, 95(2):022316, 2017.

[CS96] A. R. Calderbank and Peter W. Shor. Good quantum error-correcting codes exist. *Phys. Rev. A*, 54:1098–1105, Aug 1996.

[DHL+22] Elena Dimitrova, Jingzhen Hu, Qingzhong Liang, Brandilyn Stigler, and Anyu Zhang. Algebraic model selection and experimental design in biological data science. *Advances in Applied Mathematics*, 133:102282, 2022.

[DM76] P. Delsarte and R. J. McEliece. Zeros of functions in finite abelian group algebras. *Am. J. Math.*, 98:197–224, 1976.

[GC99] Daniel Gottesman and Isaac L Chuang. Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations. *Nature*, 402(6760):390–393, 1999.

[GG05] Solomon W Golomb and Guang Gong. *Signal design for good correlation: for wireless communication, cryptography, and radar*. Cambridge University Press, 2005.

[Gle71] Andrew M Gleason. Weight polynomials of self-dual codes and the macwilliams identities. In *Actes Congres Int. de Mathematique, 1970*. Gauthier-Villars, 1971.

[Gol49] M.J.E Golay. Notes on Digital Coding. *Proceedings of the IEEE*, page 657, 1949.

[Got98a] Daniel Gottesman. The Heisenberg representation of quantum computers. In *Intl. Conf. on Group Theor. Meth. Phys.*, pages 32–43. International Press, Cambridge, MA, 1998.

[Got98b] Daniel Gottesman. Theory of fault-tolerant quantum computation. *Phys. Rev. A*, 57(1):127, 1998.

[Got09] Daniel Gottesman. An Introduction to Quantum Error Correction and Fault-Tolerant Quantum Computation. *arXiv preprint arXiv:0904.2557*, 2009.

[Haa18] Jeongwan Haah. Towers of generalized divisible quantum codes. *Phys. Rev. A*, 97(4):042327, 2018.

[Ham50] Richard Hamming. Error detecting and error correcting codes. *The Bell System Technical Journal*, 1950.

[HDF19] Eric Huang, Andrew C. Doherty, and Steven Flammia. Performance of quantum error correction with coherent errors. *Phys. Rev. A*, 99(2):022313, 2019.

[HH18] Jeongwan Haah and Matthew B Hastings. Codes and protocols for distilling t, controlled-s, and toffoli gates. *Quantum*, 2:71, 2018.

[HLC21] Jingzhen Hu, Qingzhong Liang, and Robert Calderbank. Climbing the diagonal clifford hierarchy. *arXiv preprint arXiv:2110.11923*, 2021.

[HLC22a] Jingzhen Hu, Qingzhong Liang, and Robert Calderbank. Co-design of css codes and diagonal gates. In *2022 IEEE International Symposium on Information Theory (ISIT)*, pages 1229–1234. IEEE, 2022.

[HLC22b] Jingzhen Hu, Qingzhong Liang, and Robert Calderbank. Designing the quantum channels induced by diagonal gates. *Quantum*, 6:802, 2022.

[HLC22c] Jingzhen Hu, Qingzhong Liang, and Robert Calderbank. Divisible codes for quantum computation. *arXiv preprint arXiv:2204.13176*, 2022.

[HLRC21a] Jingzhen Hu, Qingzhong Liang, Narayanan Rengaswamy, and Robert Calderbank. Mitigating coherent noise by balancing weight-2 z-stabilizers. *IEEE Transactions on Information Theory*, 68(3):1795–1808, 2021.

[HLRC21b] Jingzhen Hu, Qingzhong Liang, Narayanan Rengaswamy, and Robert Calderbank. Css codes that are oblivious to coherent noise. In *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, pages 1481–1486, July 2021.

[IP20] Joseph K Iverson and John Preskill. Coherence in logical quantum channels. *New J. Phys.*, 22(7):073066, 2020.

[Kat08] Daniel J Katz. Sharp p-divisibility of weights in abelian codes over $z/p^d z$. *IEEE Trans. Inf. Theory*, 54(12):5354–5380, 2008.

[KBLW01] J. Kempe, D. Bacon, D. A. Lidar, and K. B. Whaley. Theory of decoherence-free fault-tolerant universal quantum computation. *Phys. Rev. A*, 63(4):042307, 2001.

[KLM01] Emanuel Knill, Raymond Laflamme, and Gerald J Milburn. A scheme for efficient quantum computation with linear optics. *nature*, 409(6816):46–52, 2001.

[KT19] Anirudh Krishna and Jean-Pierre Tillich. Towards low overhead magic state distillation. *Phys. Rev. Lett.*, 123(7):070507, 2019.

[LB17] Qingzhong Liang and Grant Bowling. Cyclic sieving of matchings. *arXiv preprint arXiv:1712.07812*, 2017.

[LC13] Andrew J Landahl and Chris Cesare. Complex instruction set computing architecture for performing accurate quantum z rotations with less

magic. *arXiv preprint arXiv:1302.3240*, 2013.

[Mac63] F.J. MacWilliams. A theorem on the distribution of weights in a systematic code. *Bell Syst. Tech. J.*, 42(1):79–94, January 1963.

[McE72] R. J. McEliece. Weight congruences for p-ary cyclic codes. *Discrete Math.*, 3:177–192, 1972.

[MS77] Florence Jessie MacWilliams and Neil James Alexander Sloane. *The theory of error correcting codes*, volume 16. Elsevier, 1977.

[Mul54] Bodegas De Muller. Application of Boolean algebra to switching circuit design and to error detection. *Transactions of the IRE professional group on electronic computers*, 1954.

[NC11] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2011.

[NH21] Sepehr Nezami and Jeongwan Haah. Classification of small triorthogonal codes. *arXiv preprint arXiv:2107.09684*, 2021.

[NRS06] Gabriele Nebe, Eric M Rains, and Neil James Alexander Sloane. *Self-dual codes and invariant theory*, volume 17. Springer, 2006.

[Ouy] Yingkai Ouyang. personal communication.

[Ouy20] Yingkai Ouyang. Avoiding coherent errors with rotated concatenated stabilizer codes. *arXiv preprint arXiv:2010.00538*, 2020.

[Ouy21] Yingkai Ouyang. Avoiding coherent errors with rotated concatenated stabilizer codes. *Npj Quantum Inf.*, 7(1):1–7, 2021.

[PDH+20] Kaitlyn Phillipson, Elena S Dimitrova, Molly Honecker, Jingzhen Hu, and Qingzhong Liang. Gröbner bases of convex neural code ideals. In *Advances in Mathematical Sciences: AWM Research Symposium, Houston, TX, April 2019*, pages 127–138. Springer, 2020.

[RCNP20] Narayanan Rengaswamy, Robert Calderbank, Michael Newman, and Henry D. Pfister. On optimality of CSS codes for transversal T. *IEEE J. Sel. Areas in Inf. Theory*, 1(2):499–514, 2020.

[RCP19] Narayanan Rengaswamy, Robert Calderbank, and Henry D. Pfister. Unifying the Clifford hierarchy via symmetric matrices over rings. *Phys. Rev. A*, 100(2):022304, 2019.

[RCPK18] N. Rengaswamy, R. Calderbank, H. D. Pfister, and S. Kadhe. Synthesis of logical Clifford operators via symplectic geometry. In *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, pages 791–795, June 2018.

[Ree54] I. Reed. A class of multiple-error-correcting codes and the decoding scheme. *Transactions of the IRE Professional Group on Information Theory*, 4(4):38–49, 1954.

[Rei05] Ben W Reichardt. Quantum universality from magic states distillation applied to CSS codes. *Quantum Inf. Process*, 4(3):251–264, 2005.

[Sha48] Claude Shannon. A Mathematical Theory of Communication. *The Bell System Technical Journal*, 1948.

[Sho96] Peter W Shor. Fault-tolerant quantum computation. In *Proc. - Annu. IEEE Symp. Found. Comput. Sci. FOCS*, pages 56–65. IEEE, 1996.

[Slo77] Neil JA Sloane. Error-correcting codes and invariant theory: new applications of a nineteenth-century technique. *Am. Math. Mon.*, 84(2):82–107, 1977.

[Ste96] A. M. Steane. Simple quantum error-correcting codes. *Phys. Rev. A*, 54(6):4741–4751, 1996.

[VB19] Christophe Vuillot and Nikolas P. Breuckmann. Quantum Pin Codes. *arXiv preprint arXiv:1906.11394*, 2019.

[War01] H. N. Ward. Divisible codes – a survey. *Serdica Math. J.*, 27 (4):263–278, 2001.

[ZR97] Paolo Zanardi and Mario Rasetti. Noiseless quantum codes. *Phys. Rev. Lett.*, 79(17):3306, 1997.

Biography

Qingzhong Liang received the B.S. degree (Hons.) in mathematics from the University of Michigan, Ann Arbor, USA, in 2017. During the undergraduate study, he worked with Professor Thomas Lam on cyclic sieving phenomenon in algebraic combinatorics. His work led to the publication [LB17]. He also participated in mathematical competitions and achieved high rankings. He ranked 15th in the nation and first in the department at the 37th Virginia Tech Regional Mathematics Contest. He also ranked 260.5th in the nation and third in the department at the 75th William Lowell Putnam Mathematical Competition. In 2018, he moved to Duke University to pursue the Ph.D. degree in mathematics with focus on quantum error correction, and quantum computing. His advisor is Professor Robert Calderbank. His research in quantum error correcting codes led to the publications [HLRC21a], [HLRC21b], [HLC22b], [HLC21], [HLC22c], and [HLC22a]. He also collaborated with researchers in the fields of parking functions, numerical semigroup, and applied algebraic geometry, and accomplished the publications [ABDB$^+$19], [ABD$^+$19], [DHL$^+$22], and [PDH$^+$20]. In addition, he served as an Officer in the Duke Student Chapter of SIAM (Society for Industrial and Applied Mathematics) and organized the 2021 Triangle Area Graduate Mathematics Conference (TAGMaC). He received certificate of recognition from SIAM for his outstanding efforts and accomplishments to the SIAM chapter at Duke University.