BACHELOR'S DEGREE THESIS

Degree in Mathematics

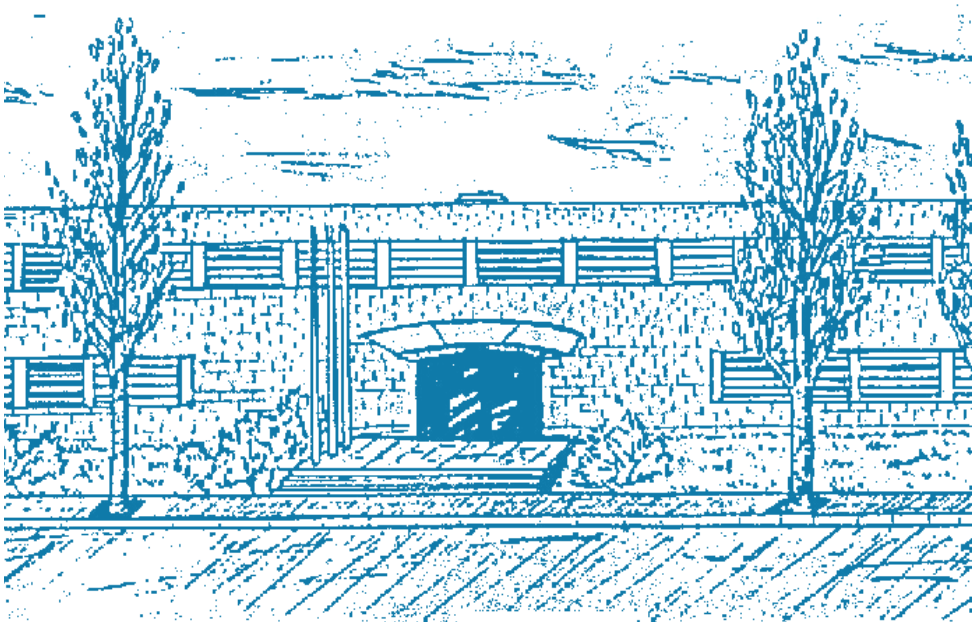
Title: The Geometry of Quantum Stabiliser Codes

Author: Aina Centelles Tarrés

Advisor: Simeon Ball

Department: Applied Mathematics IV

Academic year: 2019-2020



UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat de Matemàtiques i Estadística

The Geometry of Quantum Stabiliser Codes

A Degree Thesis submitted to
Facultat de Matemàtiques i Estadística
Universitat Politècnica de Catalunya

In partial fulfilment of the requirements for the
Bachelor's degree in Mathematics

Author
Aina Centelles Tarrés

Advisor
Simeon Michael Ball

Barcelona, May 2020

I would like to thank, first and foremost, professor Simeon Ball who introduced me to the field of quantum geometry. His guidance and encouragement were essential throughout the whole process. Secondly, my parents and my friends Gustavo, Mar, Laura and Oriol for their company and support.

The Geometry of Quantum Stabiliser Codes

Abstract

The aim of this project is to bring together quantum error-correcting codes theory and the study of finite geometries. A quantum code is used to protect quantum information from errors that may occur due to quantum decoherence. We give a geometric interpretation of the codes as sets of lines in certain finite projective spaces. We exploit the geometric aspect of codes to rewrite proofs in a more intuitive way and explore their properties through visualization. Some examples of stabiliser codes and their associated quantum sets of lines are presented. We also discuss how to build nonadditive codes as the union of stabiliser codes.
Finite geometry has proved to be a powerful tool to work on quantum error-correcting codes. Some of its applications include finding new codes or proving the non-existence of codes with certain parameters.

Key words: coding theory, stabiliser codes, finite geometry, quantum error-correction.

Contents

Introduction

The modern theory of quantum mechanics was introduced somewhere around 1920 by physicists of the time. Later on, the concept of a quantum computer was presented by Richard Feynman in 1982. Although for many problems quantum computers present little to no advantage to classical computers, the power of this often counter-intuitive discipline is undeniable. From breaking cryptosystems to simulating the behaviour of quantum physics, its limits are still blurry.

One of the main problems that the quantum computing field faces is quantum decoherence, that is, the unwanted interaction between quantum computers and their environment. Although immense progress in quantum computation has been achieved in the last few decades, a lot remains unknown. Combine this with its powerful applications and you get the motivation of many mathematicians and physicists who work on quantum error-correcting codes. An error-correcting code is a subspace where all errors up to a certain weight can be detected and corrected using a recovery map. Classical error-correcting codes are based on redundancy, however this technique does not work with quantum information by the no-cloning theorem.

In this project we will focus on the most common type of quantum error-correcting codes, stabiliser codes. A quantum stabiliser code with minimum distance d is able to detect and correct $\lfloor \frac{d-1}{2} \rfloor$ errors. We will discuss the dimension and minimum distance of a given stabiliser code and give a bijection between stabiliser codes and additive codes over $\mathbb{F}_4$.

The main purpose of this project is to find a way to translate the codewords of a stabiliser code into elements of a finite projective space over $\mathbb{F}_2$. Namely, each generator of a stabiliser group will translate into a line in the projective space. A set of lines in a projective space correspond to a stabiliser code –from here on called *quantum sets of lines*– if any co-dimension 2 subspace is skew to an even number of the lines. A method for asserting the minimum distance and dimension of code geometrically will also be presented. Moreover we create visualisations for the geometries of some codes using the Tikz package.

Translating stabiliser codes into quantum sets of lines opens the door to an extensive theory on finite geometries. Many applications are derived from this process such as finding new codes or proving the non-existence of codes with certain parameters. We will see an equivalent definition of a quantum set of lines: *any quantum set of lines is the sum modulo two of pencils of lines.* This provides an immediate way of generating new codes from a quantum set of lines by adding pencils modulo two. Another kind of error-correcting codes are nonadditive quantum codes. We will also discuss a way of finding a nonadditive quantum code as the union of several stabiliser codes.

More precisely, the objectives for this project are:

- Understanding the principles and behaviour of quantum mechanics and the errors caused by quantum decoherence.
- Delve into quantum error-correcting techniques, more precisely on quantum stabiliser codes.
- Understand and develop the geometrical translation of stabiliser codes into finite projective spaces based on [11] and [3].
- Apply these theoretical concepts to find the associated geometries to several stabiliser codes and vice-versa.

The structure of this project is the following. In Chapter 1, the background and definitions are explained. This includes both quantum error correction and mathematical concepts. In Chapter 2, we introduce quantum stabiliser codes and their parameters as well as their analogy as additive codes over $\mathbb{F}_4$ which will be the basis for Chapter 3, where the geometry is introduced. Chapter 3 contains the theory behind the translation of linear, additive and stabiliser codes into quantum sets of lines in certain finite projective spaces. Some examples of codes and their associated geometries are presented and explained through visualisations. In Chapter 4, a nonadditive quantum code is presented which is the union of 6 stabiliser codes, derived from [18]. Finally, we present the conclusions and ideas for further work in Chapter 5.

This project follows from the notes written by Felix Huber and Simeon Ball ([3]) for a graduate course on Quantum Error-Correcting Codes by the Barcelona Graduate School of Mathematics which took place in January 2020.

Chapter 1

Background and Definitions

In this chapter we introduce the background and context needed for this work. Most of the quantum mechanics concepts are taken from [16], which is a good reference for the reader interested in this topic. For anyone interested in the power and applications of quantum computation, reading Aaronson's article [1] is recommended.

1.1 Quantum Mechanics

Classical computation uses the fundamental concept of information: the bit, which can be in states 0 or 1. Quantum computation has an analogous concept: the quantum bit which we will call *qubit* for short. A qubit $|\psi\rangle$ can be in a state other than the two classic states $|0\rangle$ and $|1\rangle$. We will use the Dirac notation $|\cdot\rangle$ as the standard notation for qubit states. Apart from being in states $|0\rangle$ and $|1\rangle$ (which correspond to the classic 0, 1 for bits), qubits can be in a superposition of $|0\rangle$ and $|1\rangle$. What we call superposition of states, is in fact a linear combination of states $|0\rangle$ and $|1\rangle$:

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$$

Intuitively, we can think of a qubit $|\psi\rangle$ as being somewhere in between states $|0\rangle$ and $|1\rangle$.
When we perform a measurement on a qubit, it can only return either 0 or 1. As we will see in the next section, the probability of the measurement returning 0 or 1 depends on α and β.
However, once we measure a qubit and obtain a result, its state is automatically changed into the state of the result (that is, $|0\rangle$ or $|1\rangle$). For instance, consider the qubit

$$|\psi\rangle = \frac{1}{\sqrt{2}}\,|0\rangle + \frac{1}{\sqrt{2}}\,|1\rangle$$

Suppose we make a measurement on $|\psi\rangle$ and it returns 0, then the state of $|\psi\rangle$ will be $|0\rangle$ after this. The reason for this phenomenon is unknown but many physical systems can be used to realise qubits and various experiments confirm this strange behaviour. For instance, an electron orbiting an atom can be in so-called "ground" or "excited" states which can be thought of as the $|0\rangle$ and $|1\rangle$ states. The electron can be moved from state $|0\rangle$ to $|1\rangle$ and vice versa by shining light for an adequate period of time and with a certain energy. What is more, by shining such light for a shorter period of time, the electron can be moved to be in an intermediate state between $|0\rangle$ and $|1\rangle$.

So far we have seen that qubits can be in a superposition of states and that a measurement on any qubit (which can only return the results $|0\rangle$ or $|1\rangle$) will permanently change its state into the result state. Now, it is only natural to wonder how such a counter-intuitive behaviour is useful in terms of information theory.
Apparently, when quantum systems evolve in nature without measurements being performed, nature does keep track of all the αs and βs, so it seems that in a single state of a qubit, there is "extra information" hidden behind it. What is more, when the number of qubits is increased, this extra hidden information grows exponentially and the only way to retrieve α and β is if one was to measure an infinite number of identically prepared qubits.

1.2 Qubits and the Pauli Matrices

For our purposes we will think of qubits as an abstract mathematical object. That is, we use a vector in the two-dimensional complex vector space $\mathbb{C}^2$ to represent the state of a qubit. Namely any qubit is written as a linear combination of the two basis states $|0\rangle$ and $|1\rangle$, with

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

where

$$|0\rangle = \begin{pmatrix}1\\0\end{pmatrix} \in \mathbb{C}^2 \quad \text{and} \quad |1\rangle = \begin{pmatrix}0\\1\end{pmatrix} \in \mathbb{C}^2$$

such that $\alpha, \beta \in \mathbb{C}$ and $|\alpha|^2 + |\beta|^2 = 1$. When a qubit is measured for the first time, the probability of it being in state $|0\rangle$ is $|\alpha|^2$ and the probability of state $|1\rangle$ is $|\beta|^2$. As we mentioned before, when a measurement is performed, it can only return either 0 or 1, therefore since adding both probabilities should clearly sum to one, we have

$$|\alpha|^2 + |\beta|^2 = \overline{\alpha}\alpha + \overline{\beta}\beta = 1.$$

We will use the notation $|\psi\rangle$ to designate a column vector and $\langle\psi|$ to designate a row vector whose coordinates are the conjugate of the coordinates of ψ. Using this notation we can define the following inner product.

Definition 1.2.1 (Inner product on $\mathbb{C}^2$)**.** We define the inner product on $\mathbb{C}^2$ as

$$\begin{aligned}\langle\cdot|\cdot\rangle \colon \mathbb{C}^2 \times \mathbb{C}^2 &\to \mathbb{C}\\ (\psi_1, \psi_2) &\mapsto \langle\psi_1|\psi_2\rangle = \begin{pmatrix}\overline{\alpha_1} & \overline{\beta_1}\end{pmatrix}\begin{pmatrix}\alpha_2\\ \beta_2\end{pmatrix} = \overline{\alpha_1}\alpha_2 + \overline{\beta_1}\beta_2\end{aligned}$$

Definition 1.2.2 (Unitary transformation)**.** A unitary transformation in $\mathbb{C}^2$ is a non-singular matrix $U \in M_{2\times 2}(\mathbb{C})$ such that

$$\langle U\psi_1|U\psi_2\rangle = \langle\psi_1|\psi_2\rangle.$$

Remark. In particular, we have

$$\langle U\psi_1|U\psi_1\rangle = \langle\psi_1|\psi_1\rangle = \overline{\alpha}\alpha + \overline{\beta}\beta = 1.$$

Example 1.2.3. The Matrix $U = \left(\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right)$ is a unitary transformation since

$$\langle U\psi_1|U\psi_2\rangle = \langle \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \beta_1 \end{pmatrix} | \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} \alpha_2 \\ \beta_2 \end{pmatrix} \rangle = \overline{\alpha_1}\alpha_2 + (-\overline{\beta_1})(-\beta_2) = \langle \psi_1|\psi_2\rangle.$$

We can think of a unitary transformation U as an error on a qubit. That is, while still preserving the condition $\langle\psi_1|\psi_1\rangle = \overline{\alpha}\alpha + \overline{\beta}\beta = 1$, U alters α and/or β (unless $U = Id$).
Therefore, we will consider any 2×2 non-identity unitary transformation in $\mathbb{C}^2$ as an error on a qubit.

Definition 1.2.4 (Pauli matrices)**.** The Pauli matrices are the unitary 2×2 complex matrices

$$\sigma_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

The Pauli Matrices form a basis for all 2×2 unitary transformations in $\mathbb{C}^2$, so we can express any qubit error as a linear combination of $\sigma_0, \sigma_x, \sigma_y$ and σ_z.

Note that σ_x corresponds to the $|0\rangle \rightleftarrows |1\rangle$ bit flip

$$\sigma_x |\psi\rangle = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \beta \\ \alpha \end{pmatrix} = \beta |0\rangle + \alpha |1\rangle$$

σ_z corresponds to the $\alpha|0\rangle + \beta|1\rangle \rightleftarrows \alpha|0\rangle - \beta|1\rangle$ sign flip, and σ_y is a combination of both a bit and a sign flip, since $\sigma_y = i\sigma_x\sigma_z$.

Now, instead of working with single qubits, we will study error-correcting on systems of n qubits. A set of n qubits is interpreted as a vector in the 2^n-dimensional Hilbert Space $\mathcal{H}_n = \mathbb{C}^2 \otimes \cdots \otimes \mathbb{C}^2$. To simplify notation when working with more than one qubit, we will write $|\psi \cdots \psi\rangle$ instead of $|\psi\rangle \otimes \cdots \otimes |\psi\rangle$.
Remark. The Pauli matrices have some useful product properties:

$$\begin{aligned} \sigma_x\sigma_y &= -\sigma_y\sigma_x = i\sigma_z \\ \sigma_x\sigma_z &= -\sigma_z\sigma_x = i\sigma_y \\ \sigma_y\sigma_z &= -\sigma_z\sigma_y = i\sigma_x \end{aligned}$$

Definition 1.2.5 (Pauli group)**.** The Pauli group on 1 qubit $\mathcal{P}_1$ is the 16-element group generated by the Pauli matrices.

$$\mathcal{P}_1 := \{\pm\sigma_0, \pm i\sigma_0, \pm\sigma_x, \pm i\sigma_x, \pm\sigma_y, \pm i\sigma_y, \pm\sigma_z, \pm i\sigma_z\}$$

In general, the Pauli group on n qubits $\mathcal{P}_n$ is composed of the 4^{n+1} tensor products of the elements in $\mathcal{P}_1$.

Remark. For our purposes we will not work with all the 4^{n+1} elements in $\mathcal{P}_n$ but with the 4^n elements in $\mathcal{P}_n/\{\pm 1, \pm i\}$ instead. The reason for this will become apparent in Chapter 2.

Proposition 1.2.6. *All elements in $\mathcal{P}_n/\{\pm 1, \pm i\}$ either commute or anti-commute.*

Definition 1.2.7 (Hermitian matrix)**.** A matrix A is hermitian if $A = A^\dagger$ where $A^\dagger$ denotes the conjugate transpose of A.

Lemma 1.2.8. *Let H be a hermitian matrix, then:*

- *H can be diagonalised by a unitary matrix and all its eigenvectors with different eigenvalue are orthogonal.*
- *H^{-1} is also hermitian.*
- *If J is another hermitian matrix, then $H + J$ is hermitian.*
- *HJ is hermitian if $HJ = JH$.*

Proposition 1.2.9. *All the elements in $\mathcal{P}_n/\{\pm 1, \pm i\}$ are hermitian.*

Proof. Consider an element $\sigma_1 \otimes \cdots \otimes \sigma_n$ in $\mathcal{P}_n/\{\pm 1, \pm i\}$. We have

$$(\sigma_1 \otimes \cdots \otimes \sigma_n)^\dagger = \sigma_1^\dagger \otimes \cdots \otimes \sigma_n^\dagger$$

since for any two square matrices $\overline{A \otimes B} = \overline{A} \otimes \overline{B}$ and $(A \otimes B)^T = A^T \otimes B^T$.
It only remains to check that $\sigma_0^\dagger = \sigma_0$, $\sigma_x^\dagger = \sigma_x$, $\sigma_y^\dagger = \sigma_y$, $\sigma_z^\dagger = \sigma_z$ which is trivial. □

Definition 1.2.10 (Weight)**.** The weight of an operator $w \in \mathcal{P}_n/\{\pm 1, \pm i\}$ is the number of tensor factors which are not σ_0. For example, $w = \sigma_0 \otimes \sigma_x \otimes \sigma_x \otimes \sigma_z$ has weight 3.

The weight of an error is an indicator of how hard it is to detect and correct. In the next chapter, we will see that quantum error-correcting codes can detect and correct all errors up to a certain weight.

Definition 1.2.11 (Error-correcting code)**.** A quantum error-correcting code is a linear subspace of $\mathcal{H}_n$, where all errors up to a certain weight can be detected and corrected.

1.3 Tensor Products

We use tensor products to denote a set of more than one qubit. In this section we present some properties of tensor spaces and linear operators that we will use when considering how errors –the linear operators– act on sets of qubits.

Definition 1.3.1. Given two vector spaces V and W over a field K, we define the tensor product of V and W, $V \otimes W$ as the bigger vector space whose elements are of the form

$$\sum_i \lambda_i v_i \otimes w_i$$

where $v_i \in V$, $w_i \in W$ and $\lambda_i \in K$.

Proposition 1.3.2. *The dimension of $V \otimes W$ is nm where $\dim V = n$ and $\dim W = m$.*

A set of n qubits is represented as a vector in the 2^n-dimensional Hilbert Space $\mathcal{H}_n = (\mathbb{C}^2)^{\otimes n}$ which is the tensor product of $\mathbb{C}^2$ with itself n times.

Proposition 1.3.3. *Given two orthonormal bases* $\{v_1, \ldots, v_n\}$ *and* $\{w_1, \ldots, w_n\}$ *of* V *and* W *respectively,*

$$v_i \otimes w_j$$

is an orthonormal basis of $V \otimes W$*.*

In our case, we have the two basis states $|0\rangle$ and $|1\rangle$ in $\mathbb{C}^2$ so any set of n qubits can be written as a linear combination of

$$\{|0\ldots0\rangle, |0\ldots01\rangle, \ldots, |1\ldots1\rangle\}$$

where $|\psi\ldots\psi\rangle$ denotes $|\psi\rangle \otimes \ldots \otimes |\psi\rangle$.

Proposition 1.3.4. *Let* λ *be a scalar. For any* $v_1, v_2 \in V$ *and* $w_1, w_2 \in W$*, the following are true*

- $\lambda(v \otimes w) = (\lambda v) \otimes w = v \otimes (\lambda w)$
- $(v_1 + v_2) \otimes w = v_1 \otimes w + v_2 \otimes w$
- $v \otimes (w_1 + w_2) = v \otimes w_1 + v \otimes w_2$

Proposition 1.3.5. *Let* A *be a linear operator on* V *and* B *a linear operator on* W*. We can define a linear operator* $A \otimes B$ *on* $V \otimes W$ *as*

$$(A \otimes B)(v \otimes w) = Av \otimes Bw$$

for any $v \in V$ *and* $w \in W$*.*

In our case, we can consider the Pauli matrices as linear operators on any qubit in $\mathbb{C}^2$. For example

$$(\sigma_x \otimes \sigma_y)(|01\rangle) = \sigma_x |0\rangle \otimes \sigma_y |1\rangle = |1\rangle \otimes -i |0\rangle = -i |10\rangle .$$

1.4 Finite Fields

The points of a finite projective space are the 1-dimensional subspaces of a vector space. This means that in order to translate stabiliser codes into lines of certain finite projective spaces, we will consider errors on qubits as elements of $\mathbb{F}_4$. In Chapter 2 we will see a bijection between the Pauli group modulo scalars $\mathcal{P}_n/\{\pm1, \pm i\}$ and $\mathbb{F}_4^n$. In this section we present some basic aspects of finite fields. For more details, the reader is referred to Chapter 1 of [2].

Definition 1.4.1 (Finite field)**.** A finite field of size q, from now on noted as $\mathbb{F}_q$, is a finite set with two internal operations $\cdot$ and $+$ such that

- $(\mathbb{F}_q, +)$ is a commutative group with identity 0.
- $(\mathbb{F}_q \setminus \{0\}, \cdot)$ is a commutative group with identity 1.
- $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$ holds for any a, b, c in $\mathbb{F}_q$

Definition 1.4.2 (Order of a finite field)**.** We say that the order of $\mathbb{F}_q$ is q.

Proposition 1.4.3. *For any $x \in \mathbb{F}_q$, $x^q = x$.*

Proof. The case $x = 0$ is clear. If $x \neq 0$, since $\mathbb{F}_q \setminus \{0\}$ is a multiplicative group of order $q-1$, we have $x^{q-1} = 1$. Multiplying by x on both sides gives $x^q = x$. □

Proposition 1.4.4. *The order of a finite field is $q = p^h$ where p is a prime number.*

Definition 1.4.5 (Subfield)**.** $\mathbb{F}_p$ is a subfield of $\mathbb{F}_q$ if $\mathbb{F}_p$ is a field and $\mathbb{F}_p \subseteq \mathbb{F}_q$.

Lemma 1.4.6. *A finite field $\mathbb{F}_q$ is a vector space over $\mathbb{F}_p$ where $q = p^h$.*

Proof. Let u be an element in $\mathbb{F}_q$, we want to see that for all $\lambda \in \mathbb{F}_p$, $\lambda u \in \mathbb{F}_q$.
We can write

$$\lambda = 1 + \ldots + 1$$

so

$$\lambda u = (1 + \ldots + 1)u = u + \ldots + u$$

which is clearly in $\mathbb{F}_q$. □

1.5 Projective Spaces over Finite Fields

Definition 1.5.1 (Finite geometry)**.** A finite geometry $(\mathcal{P}, \mathcal{L}, \mathcal{I})$ consists of a finite set of points $\mathcal{P}$, a finite set of lines $\mathcal{L}$ and an incidence structure between them $\mathcal{I}$.

Definition 1.5.2 (Projective plane)**.** A finite projective plane is a finite incidence structure such that:

- Any two points are incident with exactly one line.
- Any two lines are incident with exactly one point.
- There are 4 points, such that any 3 of them are not collinear.

Definition 1.5.3 (Order of a projective plane)**.** Any line in a projective plane contains exactly $q+1$ points. We call q the order of the plane.

Proposition 1.5.4. *In a finite projective plane of order q, the following are true:*

- *There are $q^2 + q + 1$ points.*
- *There are $q^2 + q + 1$ lines.*
- *Any point is on exactly $q + 1$ lines.*
- *There are exactly $q + 1$ points on each line.*

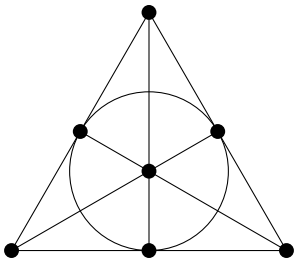


Figure 1.1: The projective plane of order two is called The Fano Plane. It has 7 points and 7 lines.

The concept of a projective plane can be extended to what is called a projective space. As taken from [4] a projective space is any incidence structure satisfying the following 3 axioms.

Definition 1.5.5 (Projective space)**.** A projective space is a finite incidence structure such that:

- Any two points are on exactly one line.
- Let A, B, C, D are 4 points such that any 3 of which are not collinear. If lines AB and CD intersect each other then AD and BC also intersect each other.
- Any line has at least 3 points on it.

Definition 1.5.6 ($PG(n-1,q)$)**.** Any finite projective space of at least 3 dimensions can be obtained in the following way. Let $V_n(\mathbb{F}_q)$ be a vector space of dimension n over the finite field $\mathbb{F}_q$. The points of the projective space –which we will denote by $PG(n-1,q)$– are the 1-dimensional subspaces of $V_n(\mathbb{F}_q)$. The lines are the 2-dimensional subspaces of $V_n(\mathbb{F}_q)$ and more generally, the d-dimensional subspaces of $PG(n-1,q)$ are the $(d+1)$-dimensional subspaces of $V_n(\mathbb{F}_q)$.
The incidence structure in $PG(n-1,q)$ is given by the subspace inclusion in $V_n(\mathbb{F}_q)$.

Definition 1.5.7 (Order and dimension of a projective space)**.** In a $PG(n,q)$ projective space, we call n the dimension of the space and q the order of the space.

Remark. For the purpose of this project, mainly $q = 2$ will be considered.

Definition 1.5.8 (Hyperplane)**.** A hyperplane is an $(n-2)$-dimensional subspace of $PG(n-1,q)$.

Lemma 1.5.9. *The number of sets of r linearly independent vectors of $V_n(\mathbb{F}_q)$ is*

$$(q^n-1)(q^n-q)\cdots(q^n-q^{r-1})$$

Proof. For the first vector, we can choose any vector in $V_n(\mathbb{F}_q)$ except for 0. That is, we have (q^n-1) choices. For the second one, we can choose any vector in $V_n(\mathbb{F}_q)$ except for the q multiples of the one we previously chose, giving us (q^n-q) choices. Repeating this argument for each of the r vectors proves the statement. □

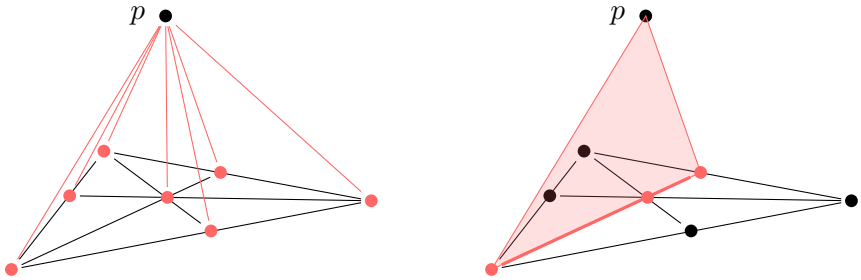
Proposition 1.5.10. *In a projective space $PG(n-1,q)$, the number of $(r-1)$-dimensional subspaces is*

$$\frac{(q^n-1)(q^n-q)\cdots(q^n-q^{r-1})}{(q^r-1)(q^r-q)\cdots(q^r-q^{r-1})}$$

Proof. The number of $(r-1)$-dimensional subspaces in $PG(n-1,q)$ is precisely the number of r-dimensional subspaces of $V_n(\mathbb{F}_q)$.
An r-dimensional subspace of $V_n(\mathbb{F}_q)$ is defined by a set of r linearly independent vectors of $V_n(\mathbb{F}_q)$. However, we must take into account that there are several sets of r linearly independent vectors we can take for each r-dimensional subspace. So, the number of r-dimensional subspaces of $V_n(\mathbb{F}_q)$ is the number of choices of r linearly independent vectors of $V_n(\mathbb{F}_q)$ divided by the number of sets of r linearly independent vectors of $V_r(\mathbb{F}_q)$.By the previous lemma, this proves the statement. □

Proposition 1.5.11. *In a projective space $PG(n-1,q)$, the number of $(r-1)$-dimensional subspaces containing a fixed $(s-1)$-dimensional subspace $(s \leq r)$ is equal to the number of $(r-s-1)$-dimensional subspaces in $PG(n-s-1,q)$. Namely*

$$\frac{(q^{n-s}-1)(q^{n-s}-q)\cdots(q^{n-s}-q^{r-s-1})}{(q^{r-s}-1)(q^{r-s}-q)\cdots(q^{r-s}-q^{r-s-1})}$$



(a) Each point in $PG(2,2)$ defines a line through p in $PG(3,2)$.

(b) Each line in $PG(2,2)$ defines a plane containing p in $PG(3,2)$.

Example 1.5.12. $PG(3,2)$ is the smallest three-dimensional projective space. It has 15 points, 35 lines and 15 planes. Each point is in 7 lines and 7 planes. Each line contains 3 points and is contained in 3 planes. Each plane contains 7 points and 7 lines.

Definition 1.5.13 (k-spread)**.** A k-spread in $PG(n-1,q)$ is a set of k-dimensional subspaces that partition $PG(n-1,q)$.

For example, a 1-spread in $PG(3,2)$ is a set of 5 skew lines.

For a more in depth view on projective geometry, the reader is referred to [2] by Ball and Chapter 9 of [8] by Cameron.

1.6 Linear and Additive Codes

In this section, Chapter 7 from [14] by Jones & Jones and Chapter 3 from [20] by van Lint have been used to develop an introduction to linear and additive codes. This will serve as a basis for stabiliser codes later on.

Definition 1.6.1 (Code)**.** Let A be a finite set which we call the alphabet. A code C of length n over A is a subset of A^n. The elements in C are called codewords.

Definition 1.6.2 (Minimum distance)**.** The minimum distance d of a code C is the minimum number of coordinates in which two codewords differ.

Example 1.6.3. Let C be a code of length 4 over the alphabet $A = \{1, 2, 3, 4\}$

$$C = \{1111, 1232, 3312, 4123\}$$

In this case, the minimum distance in C is $d = 3$.

Definition 1.6.4 (Linear code)**.** A k-dimensional linear code of length n over $\mathbb{F}_q$ is a subspace of dimension k of $\mathbb{F}_q^n$.

A k-dimensional linear code of length n over $\mathbb{F}_q$ with minimum distance d is denoted as $[n, k, d]_q$.

Definition 1.6.5 (Additive code)**.** An additive code is a linear code C closed under $+$, that is, $u + v \in C \quad \forall u, v \in C$.

Definition 1.6.6 (Weight)**.** The weight of a vector c of $\mathbb{F}_q^n$ is the number of coordinates that are different from zero.

Proposition 1.6.7. *The minimum distance d of an additive code C is equal to the minimum non-zero weight w in C.*

Proof. We will prove $w \leq d$ and $d \leq w$ to conclude $d = w$.

$d \leq w$
By definition, C is a subset of A^n, where A is a finite set with a commutative operation $+$ and a neutral element 0, and C is closed under $+$: $\forall u, v \in C, u + v \in C$. Thus, since A is finite, summing a codeword $u \in C$ enough times, will give us $0 = (0, \ldots, 0) \in C$. Suppose that $u \in C$ has weight w. By definition, the minimum distance is the minimum number of coordinates in which two codewords of C differ. Thus, since u and 0 are both in C and they differ in w coordinates, then the minimum distance can be at most w.

$w \leq d$
On the other hand, since $0 \in C$, we have $u \in C \Rightarrow -u \in C$. Now suppose $u, v \in C$ are two codewords that differ in exactly d coordinates, then $u - v \in C$ has weight d and so $w \leq d$. □

1.7 Quantum Error Detection and Correction

Classic error-correction theory relies on redundancy. An error can be detected by making copies of the original state and then taking the majority vote. However, the no-cloning theorem states that we cannot make identical copies of a certain quantum state. This forces us to rely on other properties to find quantum error-correcting codes. In this section we show a simple classic code, we prove the no-cloning theorem and we introduce the principles of quantum error-correction.

Example 1.7.1. Recall that a classical code of length n is a subset of A^n where A is a finite set called the alphabet. The simplest example of a classical error-correcting code is the repetition code,

where each element $a \in A$ is encoded to its repetition n times:

$$A \to A^n$$
$$a \mapsto (a, \cdots, a)$$

For example, suppose $A = \{0, 1\}$ and $n = 3$, then we encode $0 \mapsto (0, 0, 0)$ and $1 \mapsto (1, 1, 1)$. Now we can decode by taking the majority and this allows us to detect and correct up to $\lfloor \frac{n-1}{2} \rfloor$ errors, in this case 1:

$$(0,0,0), (0,0,1), (0,1,0) \text{ and } (1,0,0) \text{ are decoded as } 0$$

$$(1,1,1), (1,1,0), (1,0,1) \text{ and } (0,1,1) \text{ are decoded as } 1$$

Unfortunately, the repetition encoding is not valid for quantum codes. Consider such code, for example:

$$|\psi\rangle \mapsto |\psi\rangle \otimes |\psi\rangle$$

This code would have to be linear but that contradicts the following theorem.

Theorem 1.7.2 (No-cloning theorem)**.** *There is no linear map which takes $|\psi\rangle$ to $|\psi\rangle \otimes |\psi\rangle$ for all $|\psi\rangle \in \mathcal{H}_n$.*

Proof. Suppose such a map f exists. Then by linearity:

$$f(|\psi_1\rangle + |\psi_2\rangle) = f(|\psi_1\rangle) + f(|\psi_2\rangle)$$

But this is not true since

$$f(|\psi_1\rangle + |\psi_2\rangle) = (|\psi_1\rangle + |\psi_2\rangle) \otimes (|\psi_1\rangle + |\psi_2\rangle) \neq (|\psi_1\rangle \otimes |\psi_1\rangle) + (|\psi_2\rangle \otimes |\psi_2\rangle) = f(|\psi_1\rangle) + f(|\psi_2\rangle)$$

□

We have seen that we cannot build codes based on redundancy for quantum information. Instead, we can use orthogonal projections as measurement operators. This procedure is based on what is called the *projection postulate*. As taken from Davies & Betts [9], it reads

"Every observable can be represented by a Hermitian operator, the eigenvalues of which are the various possible values that would be obtained on measurement. Immediately after a measurement, the state of the system is the corresponding eigenstate associated with that eigenvalue."

Recall that a system of n qubits is represented as an element of $\mathcal{H}_n = \mathbb{C}^2 \otimes \cdots \otimes \mathbb{C}^2$. Let Q be a subspace of $\mathcal{H}_n$.

Let $Q^\perp = \{u \in \mathcal{H}_n : \langle u|v\rangle = 0 \quad \forall v \in Q\}$ denote the orthogonal complement of Q. Any vector $v \in Q$ can be uniquely written as

$$v = \Pi_Q(v) + \Pi_{Q^\perp}(v)$$

where $\Pi_Q(v) \in Q$ and $\Pi_{Q^\perp}(v) \in Q^\perp$.

Definition 1.7.3 (Orthogonal projection)**.** The map $v \mapsto \Pi_Q(v)$ is a linear map called the orthogonal projection onto Q.

Definition 1.7.4 (Detectable error)**.** A linear operator E on $\mathcal{H}_n$ is a detectable error if, for all $x, y \in Q$ such that $\langle x|y\rangle = 0$, we have that $\langle x| E |y\rangle = 0$.

Suppose we have n qubits in a state $|y\rangle \in Q$. If an error E has occurred then the n qubits are now in state $E|y\rangle$. Now suppose we make a measurement $\widehat{\Pi}_Q$ on $E|y\rangle$. Since Π_Q is a projection matrix, we have ${\Pi_Q}^2 = \Pi_Q$ so 0 and 1 are the only possible eigenvalues of Π_Q. This leaves us with two possible scenarios:

- Suppose the measurement returns 1. If E is a detectable error, then by definition $\Pi_Q E|y\rangle$ is orthogonal to all vectors $x \in Q$ that are orthogonal to y and therefore, $\Pi_Q E|y\rangle$ must be a multiple of $|y\rangle$. This means that if E is detectable we have retrieved the original state $|y\rangle$ up to a scalar factor.

- Suppose the measurement of $E|y\rangle$ returns 0. This means that $E|y\rangle \in \ker \Pi_Q = Q^{\perp}$, so we can be certain that some error has occurred.

If we are more careful about which measurements we make, then one can show that there is a recovery map which restores state $|y\rangle$ for any correctable error E and this is given by Nielsen and Chuang's Theorem 10.1 in [16].

Chapter 2

Quantum Stabiliser Codes

2.1 Definition and Examples

Definition 2.1.1 (Qubit stabiliser code)**.** Let S be a subgroup of $\mathcal{P}_n/\{\pm 1, \pm i\}$ generated by $n-k$ independent mutually commuting matrices $M_1, \ldots, M_{n-k}$ of $\mathcal{P}_n/\{\pm 1, \pm i\}$. A qubit stabiliser code $Q(S)$ is the joint eigenspace with eigenvalue 1 of $\langle M_1, \ldots, M_{n-k} \rangle$.

Note that to find $Q(S)$, it is enough to find the joint eigenspace of eigenvalue 1 of $M_1, \ldots, M_{n-k}$. If v is an eigenvector of eigenvalue 1 of $M_1, \ldots, M_{n-k}$, then for any $J \subseteq \{1, \ldots, n\}$

$$(\prod_{i \in J} M_i)v = v.$$

Recall that the elements M_i in $\mathcal{P}_n/\{\pm 1, \pm i\}$ are the n-tensor products of the Pauli Matrices:

$$\sigma_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Applying these 4 errors to our basis states $|0\rangle$, $|1\rangle$ gives:

$$\begin{aligned} \sigma_0 |0\rangle &= |0\rangle & \sigma_0 |1\rangle &= |1\rangle \\ \sigma_x |0\rangle &= |1\rangle & \sigma_x |1\rangle &= |0\rangle \\ \sigma_y |0\rangle &= i |1\rangle & \sigma_y |1\rangle &= -i |0\rangle \\ \sigma_z |0\rangle &= |0\rangle & \sigma_z |1\rangle &= - |1\rangle \end{aligned}$$

Example 2.1.2. Suppose $n = 3$ and $S = \langle M_1, M_2, M_3 \rangle$, where

$$\begin{aligned} M_1 &= \sigma_0 \otimes \sigma_x \otimes \sigma_x \\ M_2 &= \sigma_0 \otimes \sigma_y \otimes \sigma_y \\ M_3 &= \sigma_x \otimes \sigma_z \otimes \sigma_z \end{aligned}$$

One can easily check that M_1, M_2, M_3 are independent and they satisfy $M_i M_j = M_j M_i$ for all $i, j \in 1, 2, 3$. For example:

$$\begin{aligned} M_1 M_2 &= (\sigma_0 \otimes \sigma_x \otimes \sigma_x)(\sigma_0 \otimes \sigma_y \otimes \sigma_y) = (\sigma_0\sigma_0) \otimes (\sigma_x\sigma_y) \otimes (\sigma_x\sigma_y) \\ &= \sigma_0 \otimes i\sigma_z \otimes i\sigma_z = -\sigma_0 \otimes \sigma_z \otimes \sigma_z \\ M_2 M_1 &= (\sigma_0 \otimes \sigma_y \otimes \sigma_y)(\sigma_0 \otimes \sigma_x \otimes \sigma_x) = (\sigma_0\sigma_0) \otimes (\sigma_y\sigma_x) \otimes (\sigma_y\sigma_x) \\ &= \sigma_0 \otimes (-i\sigma_z) \otimes (-i\sigma_z) = -\sigma_0 \otimes \sigma_z \otimes \sigma_z \end{aligned}$$

Now we want to find the joint eigenspace of eigenvalue 1 to find the stabiliser code $Q(S)$. Let

$$v = \lambda_{000}|000\rangle + \lambda_{001}|001\rangle + \lambda_{010}|010\rangle + \lambda_{011}|011\rangle + \lambda_{100}|100\rangle + \lambda_{101}|101\rangle + \lambda_{110}|110\rangle + \lambda_{111}|111\rangle$$

be a vector of 2^n coordinates, in this case $2^n = 8$. It is enough to ensure that v is in the eigenspace of eigenvalue 1 of M_1, M_2 and M_3 for v to be in $Q(S)$.
Suppose $v \in E_1$, where E_i denotes the eigenspace of eigenvalue 1 of M_i.

$$\begin{aligned} M_1 v &= \lambda_{000}(\sigma_0 \otimes \sigma_x \otimes \sigma_x)|000\rangle + \ldots + \lambda_{111}(\sigma_0 \otimes \sigma_x \otimes \sigma_x)|111\rangle \\ &= \lambda_{000}(\sigma_0|0\rangle \otimes \sigma_x|0\rangle \otimes \sigma_x|0\rangle) + \ldots + \lambda_{111}(\sigma_0|1\rangle \otimes \sigma_x|1\rangle \otimes \sigma_x|1\rangle) \\ &= \lambda_{000}|011\rangle + \lambda_{001}|010\rangle + \lambda_{010}|001\rangle + \lambda_{011}|000\rangle + \lambda_{100}|111\rangle + \lambda_{101}|110\rangle + \lambda_{110}|101\rangle + \lambda_{111}|100\rangle \end{aligned}$$

Therefore, $v \in E_1$ if and only if

$$\lambda_{000} = \lambda_{011} \qquad \lambda_{001} = \lambda_{010} \qquad \lambda_{100} = \lambda_{111} \qquad \lambda_{101} = \lambda_{110}$$

On the other hand,

$$M_2 v = -\lambda_{000}|011\rangle + \lambda_{001}|010\rangle + \lambda_{010}|001\rangle - \lambda_{011}|000\rangle - \lambda_{100}|111\rangle + \lambda_{101}|110\rangle + \lambda_{110}|101\rangle - \lambda_{111}|100\rangle$$

So, $v \in E_2$ if and only if

$$\lambda_{000} = -\lambda_{011} \qquad \lambda_{001} = \lambda_{010} \qquad \lambda_{100} = -\lambda_{111} \qquad \lambda_{101} = \lambda_{110}$$

Finally,

$$M_3 v = \lambda_{000}|100\rangle - \lambda_{001}|101\rangle - \lambda_{010}|110\rangle + \lambda_{011}|111\rangle + \lambda_{100}|000\rangle - \lambda_{101}|001\rangle - \lambda_{110}|010\rangle + \lambda_{111}|011\rangle$$

Which means $v \in E_3$ if and only if

$$\lambda_{000} = \lambda_{100} \qquad \lambda_{001} = -\lambda_{101} \qquad \lambda_{010} = -\lambda_{110} \qquad \lambda_{011} = \lambda_{111}$$

Putting all three conditions together, $Q(S) = E_1 \cap E_2 \cap E_3$ is the one-dimensional subspace spanned by

$$|001\rangle + |010\rangle - |101\rangle - |110\rangle .$$

2.2 Dimension and Minimum Distance

In this section we prove some properties of quantum stabiliser codes.

Theorem 2.2.1. *The qubit stabiliser code $Q(S)$ with stabiliser group $S = \langle M_1, \ldots, M_{n-k} \rangle$ where $M_i \in \mathcal{P}_n/\{\pm 1, \pm i\}$, has dimension 2^k.*

Proof. Let P be the linear operator

$$P = \frac{1}{|S|} \sum_{M \in S} M$$

and consider the following properties of P:

- For any error $E \in S$, we have $EP = P$ since

$$EP = \frac{1}{|S|} \sum_{M \in S} EM = \frac{1}{|S|} \sum_{M \in S} M = P.$$

- On the other hand, $P^2 = P$:

$$P^2 = \left(\frac{1}{|S|} \sum_{E \in S} E \right) P = \frac{1}{|S|} \sum_{E \in S} (EP) = \frac{1}{|S|} \sum_{E \in S} P = \frac{1}{|S|} P \sum_{E \in S} 1 = P.$$

- P is hermitian since any $M \in S$ is hermitian and the linear combination of hermitian matrices is also hermitian. This means that P is diagonalisable and thus its minimal polynomial is $P^2 - P$. The only possible eigenvalues for P are $\lambda = 1$ or $\lambda = 0$. By the primary decomposition theorem, we have $\mathbb{C}^{2^n} = E_0 \oplus E_1$ where E_0 and E_1 denote P's eigenspace of eigenvalue 0 and 1 respectively.

- $E_1 = Im(P)$

  $u \in E_1 \Rightarrow Pu = u \Rightarrow u \in Im(P)$

  $u \in Im(P) \Rightarrow Pv = u$ for some $v \Rightarrow Pu = P(Pv) = P^2 v = Pv = u \Rightarrow u \in E_1$

- $Q(S) = Im(P)$

  $u \in Q(S) \Rightarrow Mu = u$ for all $M \in S \Rightarrow Pu = \frac{1}{|S|} \sum_{M \in S} Mu = \frac{1}{|S|} \sum_{M \in S} u = \frac{1}{|S|} u \sum_{M \in S} 1 = u$

  $\Rightarrow u \in Im(P)$

  $u \in Im(P) \Rightarrow Pv = u$ for some $v \Rightarrow \forall M \in S, Mu = M(Pv) = (MP)v = Pv = u \Rightarrow u \in Q(S)$

We have seen $Q(S) = Im(P) = E_1$ so $dim(Q(S)) = dim(E_1)$.
Again by the primary decomposition theorem $tr(P) = dim(E_1)$, so we have:

$$dim(Q(S)) = dim(E_1) = tr(P) = \frac{1}{|S|} \sum_{M \in S} tr(M)$$

and

$$tr(M) = \begin{cases} 2^n & \text{if } M = Id \\ 0 & \text{otherwise} \end{cases}$$

This implies that since $|S| = 2^{n-k}$

$$dim(Q(S)) = \frac{1}{|S|} \sum_{M \in S} tr(M) = \frac{1}{2^{n-k}} 2^n = 2^k.$$

□

Definition 2.2.2 (Centraliser)**.** Given a group G and a subset S of G, the centraliser of S in G is the subgroup

$$C(S) = \{x \in G |\ xy = yx \text{ for all } y \in S\}.$$

In our case, we will consider $C(S)$, the centraliser of $S = \langle M_1, \ldots, M_{n-k} \rangle$ in $\mathcal{P}_n/\{\pm 1, \pm i\}$. The importance of this subgroup is made clear with the following theorem about the minimum distance of a stabiliser code.

Theorem 2.2.3. *The minimum distance d of a quantum stabiliser code $Q(S)$ is equal to the minimum weight of the errors in $C(S) \setminus S$, where $S = \langle M_1, \ldots, M_{n-k} \rangle$ and $C(S)$ denotes the centraliser of S.*

Proof. Consider the minimum weight w of all undetectable errors, this means we can correct all errors with weight less than w. We will prove that if E is undetectable then $E \in C(S) \setminus S$. This means that $Q(S)$ can detect all errors with weight less than the minimum weight in $C(S) \setminus S$.

- Suppose E undetectable and $E \notin C(S)$.
  The errors we consider are in $\mathcal{P}_n/\{\pm 1, \pm i\}$, and all elements in $\mathcal{P}_n$ either commute or anti-commute, as we saw in Proposition 1.2.6. By definition, $C(S)$ contains all the elements of $\mathcal{P}_n/\{\pm 1, \pm i\}$ that commute with all elements of S. Thus, since $E \notin C(S)$ this means E anti-commutes with some element of S. Let $M \in S$ denote such element, then

  $$ME = -EM$$

  Now, recall that the elements in $Q(S)$ are the eigenvectors of eigenvalue 1 of all the elements in S. Therefore, $\forall x, y \in Q(S)$ such that $\langle x|y\rangle = 0$ we have:

  $$ME\,|y\rangle = -EM\,|y\rangle = -E\,|y\rangle$$

  We have seen that both $|x\rangle$ and $E\,|y\rangle$ are eigenvectors of M of eigenvalue 1 and -1 respectively, therefore since $|x\rangle$ and $E\,|y\rangle$ have different eigenvalues, they have to be orthogonal, so

  $$\langle x|\,E\,|y\rangle = 0.$$

  A contradiction with the fact that E is undetectable.

- Suppose E undetectable and $E \in S$.
  The fact that E is in S means that $E\,|x\rangle = |x\rangle$ for all $|x\rangle \in Q(S)$. Thus, no error has occurred.

□

Definition 2.2.4 (Minimum distance of a stabiliser code)**.** The minimum distance of a quantum stabiliser code $Q(S)$ with stabiliser group S is the minimum weight of the elements in $C(S) \setminus S$.

We will denote a quantum stabiliser code $Q(S)$ with $S = \langle M_1, \ldots, M_{n-k} \rangle$ as $[[n, k, d]]_2$, where d is the minimum distance.

Definition 2.2.5 (Pure and impure stabiliser codes)**.** We say that a stabiliser code $Q(S)$ is impure if S contains errors of weight less than the minimum distance. Otherwise we say it is a pure code.

2.3 The Shor Code

The Shor Code is an impure 9 qubit stabiliser code which can correct arbitrary single qubit errors. It encodes a qubit $|\psi\rangle$ as $|\psi_S\rangle$ in the following way:

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle \mapsto |\psi_S\rangle = \alpha\,|0_S\rangle + \beta\,|1_S\rangle$$

where

$$|0_S\rangle = (|000\rangle + |111\rangle) \otimes (|000\rangle + |111\rangle) \otimes (|000\rangle + |111\rangle)$$
$$|1_S\rangle = (|000\rangle - |111\rangle) \otimes (|000\rangle - |111\rangle) \otimes (|000\rangle - |111\rangle)$$

With this encoding, a single error of any kind can be detected and corrected. As we saw in Section 1.2, the Pauli Matrices form a basis for all 2×2 unitary transformations in $\mathbb{C}^2$. It is enough to be able to correct the errors corresponding to σ_x, σ_y and σ_z to be able to correct all errors. The reason for this for this can be found in Section 10.3.1 of [16] by Nielsen and Chuang.

First, suppose a bit flip error σ_x has occurred on the 7th bit for example. This would give us:

$$\begin{aligned}|\alpha_L\rangle = {} & \alpha_0(|000\rangle + |111\rangle) \otimes (|000\rangle + |111\rangle) \otimes (|100\rangle + |011\rangle) \\ & + \alpha_1(|000\rangle - |111\rangle) \otimes (|000\rangle - |111\rangle) \otimes (|100\rangle - |011\rangle)\end{aligned}$$

Now by taking the majority decision on the 7th, 8th and 9th bit, we decode $(|100\rangle + |011\rangle)$ and $(|100\rangle - |011\rangle)$ as $(|000\rangle + |111\rangle)$ and $(|000\rangle - |111\rangle)$ respectively.

Now suppose a phase error σ_z has occurred on the 5th bit. This would give us:

$$\begin{aligned}|\alpha_L\rangle = {} & \alpha_0(|000\rangle + |111\rangle) \otimes (|000\rangle - |111\rangle) \otimes (|000\rangle + |111\rangle) \\ & + \alpha_1(|000\rangle - |111\rangle) \otimes (|000\rangle + |111\rangle) \otimes (|000\rangle - |111\rangle)\end{aligned}$$

By taking the majority decision on the signs, we can detect the error and correct it.

Finally, imagine a σ_y error has occurred, $\sigma_y = i\sigma_x\sigma_z$ which means that both a bit flip and a phase error have occurred. Suppose this happened on the second bit, then we would have:

$$\begin{aligned}|\alpha_L\rangle = {} & \alpha_0(|010\rangle - |101\rangle) \otimes (|000\rangle + |111\rangle) \otimes (|000\rangle + |111\rangle) \\ & + \alpha_1(|010\rangle + |101\rangle) \otimes (|000\rangle - |111\rangle) \otimes (|000\rangle - |111\rangle)\end{aligned}$$

We can use the same reasoning as the two previous cases and independently correct both the sign and the bit flip by taking the majority decision.

Remark. The Shor Code is a stabiliser code where S is the subgroup generated by the following 8 matrices:

$$M_1 = \sigma_z \otimes \sigma_z \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0$$
$$M_2 = \sigma_0 \otimes \sigma_z \otimes \sigma_z \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0$$
$$M_3 = \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_z \otimes \sigma_z \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0$$
$$M_4 = \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_z \otimes \sigma_z \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0$$
$$M_5 = \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_z \otimes \sigma_z \otimes \sigma_0$$

$$M_6 = \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_z \otimes \sigma_z$$
$$M_7 = \sigma_x \otimes \sigma_x \otimes \sigma_x \otimes \sigma_x \otimes \sigma_x \otimes \sigma_x \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0$$
$$M_8 = \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_x \otimes \sigma_x \otimes \sigma_x \otimes \sigma_x \otimes \sigma_x \otimes \sigma_x$$

Recall that the M_i's have to be $n-k$ independent mutually commuting elements of $\mathcal{P}_n/\{\pm 1, \pm i\}$. In this case $n = 9$ and $k = 1$. It is easy to see that these matrices are independent and M_i and M_j commute for any i and j.

Proposition 2.3.1. *The 9-qubit Shor Code has minimum distance 3. That is, it allows us to detect and correct one error.*

Proof. As seen in Theorem 2.2.3, the minimum distance is the minimum weight of the elements in $C(S) \setminus S$: the elements in $\mathcal{P}_9/\{\pm 1, \pm i\}$ that commute with all the elements in $S = \langle M_1, \ldots, M_8 \rangle$ but are not in S.

A case by case reasoning is enough to see that if a matrix M commutes with all the elements in S, then the weight of M is at least 3 which means M has at least 3 tensor products different from σ_0.

Suppose M has weight 1

If the only tensor factor different from σ_0 is a σ_x or a σ_y, then it does not commute with one of $M_1, \ldots, M_6$. On the other hand, if it is a σ_z, then M does not commute with either M_7 or M_8.

Suppose M has weight 2

Suppose there is a σ_x or σ_y on the first system. Then there must be a σ_x or σ_y on the second system so that M commutes with M_1, but then there must also be another σ_x or σ_y on the third system so that M commutes with M_2. So M would have weight ≥ 3. The same reasoning can be used for a σ_x or σ_y in any of the other systems, forcing M to commute with $M_1, \ldots, M_6$ implies that M must have weight at least 3.

Now suppose there is a σ_z on the first system. We have seen that if the other tensor different from σ_0 is a σ_x or σ_y, then M needs another σ_x or σ_y and thus M has weight ≥ 3. Therefore, the last 2-weight option to be considered is two σ_z and seven σ_0. In order to commute with M_8 both σ_z have to be in positions $\{4, \ldots, 9\}$ or $\{1, 2, 3\}$. On the other hand, to commute with M_7, they both have to be in positions $\{1, \ldots, 6\}$ or $\{7, 8, 9\}$, which leaves us with the following possible positions for the two σ_z: $\{12, 23, 13, 45, 56, 46, 78, 89, 79\}$ but these are equal to $\{M_1, M_2, M_1M_2, M_3, M_4, M_3M_4, M_5, M_6, M_5M_6\}$ respectively which are all in S.

All possible 2-weight matrices have been considered, thus we can conclude $\nexists M \in C(S) \setminus S$ with weight 2.

M has weight 3

Finally, consider $M = \sigma_x \otimes \sigma_x \otimes \sigma_x \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_0$. M has weight 3 and it is in $C(S)$ but is not in S. Therefore, we can conclude that the minimum distance of the Shor Code is 3. □

Remark. Note that the 9-qubit Shor Code is an impure code since S contains errors of weight 2 ($M_1, \ldots, M_6, M_1M_2, M_3M_4$ and M_5M_6) but the minimum distance is 3.

2.4 Quantum Stabiliser Codes as Additive Codes over $\mathbb{F}_4$

We can find a bijection between $\mathcal{P}_n/\{\pm 1, \pm i\}$ and $\mathbb{F}_4^n$ by finding a map between $\mathcal{P}_1 = \{\sigma_0, \sigma_x, \sigma_y, \sigma_z\}$ and $\mathbb{F}_4$ and then extending it to $\mathcal{P}_n/\{\pm 1, \pm i\}$. This is very useful for us, since it gives us a way of treating qubit stabiliser codes as additive codes over $\mathbb{F}_4$.
Let $\mathbb{F}_4 = \{0, 1, e, e^2\}$ be the finite filed of order 4, where $e^2 = e + 1$. Addition in $\mathbb{F}_4$ is modulo two, that is $x + x = 0$ for all $x \in \mathbb{F}_4$. Due to this, it is isomorphic to addition in $\mathbb{F}_2^2$ when we identify 0 with $(0, 0)$, 1 with $(1, 0)$, e with $(0, 1)$ and e^2 with $(1, 1)$. Consider the map

$$\begin{aligned} \theta : \{\sigma_0, \sigma_x, \sigma_y, \sigma_z\} &\to \mathbb{F}_4 \\ \sigma_0 &\mapsto 0 \\ \sigma_x &\mapsto 1 \\ \sigma_z &\mapsto e \\ \sigma_y &\mapsto e^2 \end{aligned}$$

Now we can extend it to a map between $\mathcal{P}_n/\{\pm 1, \pm i\}$ and $\mathbb{F}_4^n$ in the following way:

$$\begin{aligned} &\tau : \mathcal{P}_n/\{\pm 1, \pm i\} \to \mathbb{F}_4^n \\ &\tau(\sigma_1 \otimes \ldots \otimes \sigma_n) \mapsto (\theta(\sigma_1), \ldots, \theta(\sigma_n)) \end{aligned}$$

For example, $\tau(\sigma_x \otimes \sigma_x \otimes \sigma_y \otimes \sigma_0 \otimes \sigma_z) = (1, 1, e^2, 0, e)$.

The following lemma and its proof make apparent why instead of working in $\mathcal{P}_n$ we work in $\mathcal{P}_n/\{\pm 1, \pm i\}$.

Lemma 2.4.1. *For all $M, N \in \mathcal{P}_n/\{\pm 1, \pm i\}$,*

$$\tau(MN) = \tau(M) + \tau(N)$$

Proof. Suppose $M = \sigma_1 \otimes \ldots \otimes \sigma_n$ and $N = \varsigma_1 \otimes \ldots \otimes \varsigma_n$. Then

$$\tau(MN) = \tau(\sigma_1\varsigma_1 \otimes \ldots \otimes \sigma_n\varsigma_n) = (\theta(\sigma_1\varsigma_1), \ldots, \theta(\sigma_n\varsigma_n))$$

On the other hand,

$$\tau(M) + \tau(N) = (\theta(\sigma_1) + \theta(\varsigma_1), \ldots, \theta(\sigma_n) + \theta(\varsigma_n))$$

Thus we just have to check coordinate by coordinate that

$$\theta(\sigma\varsigma) = \theta(\sigma) + \theta(\varsigma)$$

This becomes clear when looking at the operation tables in $\mathcal{P}_1/\{\pm 1, \pm i\}$ and $\mathbb{F}_4$.

$\cdot$	σ_0	σ_x	σ_z	σ_y
σ_0	σ_0	σ_x	σ_z	σ_y
σ_x	σ_x	σ_0	σ_y	σ_z
σ_z	σ_z	σ_y	σ_0	σ_x
σ_y	σ_y	σ_z	σ_x	σ_0

$+$	0	1	e	e^2
0	0	1	e	e^2
1	1	0	e^2	e
e	e	e^2	0	1
e^2	e^2	e	1	0

□

The above lemma implies that we have also established a bijection between subgroups of $\mathcal{P}_n/\{\pm 1, \pm i\}$ and subspaces of $\mathbb{F}_4^n$. We now go on to relate this to stabiliser codes.
First, for $u, v \in \mathbb{F}_4^n$ consider the following twisted alternating form:

$$(u, v)_t = \sum_{j=1}^{n} (u_j v_j^2 - v_j u_j^2)$$

Lemma 2.4.2. *For all $M, N \in \mathcal{P}_n/\{\pm 1, \pm i\}$,*

$$MN = (-1)^{(\tau(M), \tau(N))_t} NM$$

Proof. We know that all elements in $\mathcal{P}_n/\{\pm 1, \pm i\}$ either commute or anti-commute. We have to check that $(\tau(M), \tau(N))_t = 0$ iff M, N commute and $(\tau(M), \tau(N))_t = 1$ iff they anti-commute. Since any two Pauli matrices either commute or anti-commute, for M, N to commute, we need an even number of their systems to anti-commute.

Suppose $u = \tau(M)$ and $v = \tau(N)$ and consider the following table, for each of the j-th terms of the alternating form.

$u_j v_j (v_j - u_j)$	0	1	e	e^2
0	0	0	0	0
1	0	0	1	1
e	0	1	0	1
e^2	0	1	1	0

$(u, v)_t^j = 0$ if and only if either:

- u_j or $v_j = 0$, which corresponds to the j-th system of M or N being σ_0, which is the identity and therefore commutes with any other matrix.
- $u_j = v_j$. If the j-th system of both M, N are the same matrix, then clearly they will commute.

□

Lemma 2.4.3. *S is a subgroup of $\mathcal{P}_n/\{\pm 1, \pm i\}$ if and only if $\tau(S)$ is an additive subspace of $\mathbb{F}_4^n$.*

Proof. $\boxed{\Rightarrow}$ Let u, v be two elements of $\tau(S)$. By definition of τ there exist $M, N \in S$ such that $u = \tau(M)$ and $v = \tau(N)$. Since S is a subgroup, $MN \in S$. Finally, by Lemma 2.4.1, $u + v = \tau(M) + \tau(N) = \tau(MN)$ and therefore, $u + v \in \tau(S)$.

$\boxed{\Leftarrow}$ Let M, N be two matrices of S. By definition of τ, there exist $u, v \in \tau(S)$ such that $u = \tau(M)$ and $v = \tau(N)$. $\tau(S)$ is an additive subspace of $\mathbb{F}_4^n$ and by Lemma 2.4.1, $u + v = \tau(M) + \tau(N) = \tau(MN) \in \tau(S)$, which implies $MN \in S$. □

Definition 2.4.4. Let C be an additive subspace of $\mathbb{F}_4^n$. We define $C^{\perp_t}$ as

$$C^{\perp_t} = \{v \in \mathbb{F}_4^n | (u, v)_t = 0 \text{ for all } u \in C\}.$$

As we saw in Lemma 1.4.6, an additive subspace of $\mathbb{F}_q$ is a subspace of $\mathbb{F}_p$, where $q = p^h$ and p is prime. In this case, an additive subspace of $\mathbb{F}_4^n$ is not necessarily a subspace of $\mathbb{F}_4^n$ but it is a subspace over $\mathbb{F}_2$.

Note that $C^{\perp_t}$ is also an additive subspace of $\mathbb{F}_4^n$. Suppose $v, w \in C^{\perp_t}$, since the sum in $\mathbb{F}_4^n$ is modulo two, we have for any $u \in C$:

$$\begin{aligned}(u, v + w)_t &= \sum_{j=1}^{n}(u_j(v_j + w_j)^2 - (v_j + w_j)u_j^2) = \sum_{j=1}^{n}(u_j(v_j^2 + w_j^2) - v_j u_j^2 - w_j u_j^2) \\ &= \sum_{j=1}^{n}(u_j v_j^2 - v_j u_j^2) + \sum_{j=1}^{n}(u_j w_j^2 + w_j u_j^2) = (u, v)_t + (u, w)_t = 0 + 0 = 0\end{aligned}$$

And so, $v + w \in C^{\perp_t}$

Theorem 2.4.5. *S is a subgroup of $\mathcal{P}_n/\{\pm 1, \pm i\}$ generated by $n - k$ independent mutually commuting elements if and only if $C = \tau(S)$ is an additive subspace of $\mathbb{F}_4^n$ for which $C \subseteq C^{\perp_t}$ such that $|C| = 2^{n-k}$. What is more, τ is a bijection between C and S.*

Proof. Most of the statements of this theorem have already been proved with the previous lemmas. It remains to see why $C \subseteq C_t^{\perp}$ and $|C| = 2^{n-k}$. On the one hand, by Lemma 2.4.2, two elements in $\mathcal{P}_n/\{\pm 1, \pm i\}$ commute if and only if $(\tau(M), \tau(N))_t = 0$. This is equivalent to saying that for any two elements $u = \tau(M) \in C$ and $v = \tau(N) \in C$ we have $(u, v)_t = 0$, which implies $C \subseteq C_t^{\perp}$. On the other hand, since $|S| = 2^{n-k}$ and τ is a bijection, $|C| = |\tau(S)| = |S| = 2^{n-k}$. □

Theorem 2.4.6. *There is a $[[n, k, d]]_2$ qubit stabiliser code if and only if there is an $(n, 2^{n-k}, d)$ additive code C over $\mathbb{F}_4$ such that $C \subseteq C^{\perp_t}$ and the minimum non-zero weight in $C^{\perp_t} \setminus C$ is d.*

Proof. An $[[n, k, d]]_2$ qubit stabiliser code is the joint eigenspace with eigenvalue 1 of $n-k$ independent mutually commuting elements of $\mathcal{P}_n/\{\pm 1, \pm i\}$. The previous theorem states the existence, so it remains to check the minimum distance statement. In Theorem 2.2.3, we saw that the minimum distance of a stabiliser code with stabiliser group S, is the minimum non-zero weight of the errors in $C(S) \setminus S$, where $C(S)$ denotes the centraliser of S. If we prove that $\tau(C(S)) = C^{\perp_t}$, then since $C = \tau(S)$, we have that the minimum distance of $(n, 2^{n-k}, d)$ is the minimum non-zero weight in $C^{\perp_t} \setminus C$. To see $\tau(C(S)) = C^{\perp_t}$, we will use double inclusion as follows

$\underline{\tau(C(S)) \subseteq C^{\perp_t}}$

Let N be an element of $C(S)$ and consider $v = \tau(N)$. Since $N \in C(S)$ we have that $MN = NM$ for all M in S. In Lemma 2.4.2 we saw:

$$MN = (-1)^{(\tau(M), \tau(N))_t} NM$$

This implies that $(\tau(M), \tau(N))_t = 0$ for all M in S. We have seen that $(v, u)_t = (\tau(N), \tau(M))_t = 0$ for all $u \in \tau(S) = C$, and thus $v \in C^{\perp_t}$.

$\underline{C^{\perp_t} \subseteq \tau(C(S))}$

Let v be an element of $C^{\perp_t}$ and consider $N \in S$ such that $v = \tau(N)$. By the definition of $C^{\perp_t}$, we have that $(v, u)_t = 0$ for all $u = \tau(M) \in C$, for some $M \in S$. Again by Lemma 2.4.2 this means that $MN = NM$ for all $M \in S$ and thus, $N \in C(S)$ which means that $v = \tau(N) \in \tau(C(S))$. □

Remark. It is technically not necessary to extend the map to $\mathbb{F}_4$. Instead, we can consider a map between $\mathbb{F}_2^2$ and $\{\sigma_0, \sigma_x, \sigma_y, \sigma_z\}$ defined by

$$\begin{aligned} \theta^* : \{\sigma_0, \sigma_x, \sigma_y, \sigma_z\} &\to \mathbb{F}_2^2 \\ \sigma_0 &\mapsto (0,0) \\ \sigma_x &\mapsto (1,0) \\ \sigma_z &\mapsto (0,1) \\ \sigma_y &\mapsto (1,1) \end{aligned}$$

Now we can extend it to $\tau*$, a map between $\mathcal{P}_n/\{\pm 1, \pm i\}$ and $\mathbb{F}_2^{2n}$ applying θ^* coordinate-wise where the image of the j-th system is split between the j and $(j+n)$-th coordinate. For example

$$\tau^*(\sigma_x \otimes \sigma_x \otimes \sigma_y \otimes \sigma_0 \otimes \sigma_z) = (1,1,1,0,0|0,0,1,0,1)$$

The line between the n and $n+1$ coordinates is drawn for readability.
Note that all the lemmas and propositions we have seen in this section can be rewritten in terms of θ^*. In the upcoming sections, we will use the $\mathbb{F}_4$ notation for simplicity reasons. However some proofs will be done in $\mathbb{F}_2^2$ and then later translated to $\mathbb{F}_4$.

2.5 Syndrome Decoding

In order to use quantum codes, successful methods for encoding, error-correcting and decoding are essential. In this section, we present syndrome decoding, an algorithm for decoding quantum stabiliser codes. It allows us to detect and correct any error of weight $w \leq \lfloor \frac{d-1}{2} \rfloor$.

Take a qubit in $|\psi\rangle \in Q(S)$, a stabiliser code. Now suppose it has been altered into $E|\psi\rangle$ by some error E of weight w. For each $j \in \{1, \ldots, n-k\}$ we apply M_j as

$$M_j E |\psi\rangle .$$

When a measurement is performed on $M_j E|\psi\rangle$ it will return an eigenvalue of M_j and leave the state in the correspondent eigenvector.
We know that E and M_j either commute or anti-commute.

- If M_j and E commute, then $M_j E|\psi\rangle = E M_j |\psi\rangle = E|\psi\rangle$ since $|\psi\rangle$ is in the eigenspace of eigenvalue 1 of all the $M_1 \ldots, M_{n-k}$. In this case, the measurement returns 1.

· If M_j and E anti-commute, then $M_jE|\psi\rangle = -EM_j|\psi\rangle = -E|\psi\rangle$. In this case, the measurement returns -1.

A table is created of the measurements performed on $M_1E, \ldots, M_{n-k}E$ for each detectable E of weight w. Now given the measurements, one can retrieve the error E given that the measurements for each E will be pairwise distinct which we prove in Proposition 2.5.2. The following is an example of syndrome decoding for a $[[5,1,3]]_2$ code.

Example 2.5.1. Consider the $[[5,1,3]]_2$ code whose stabiliser group is generated by

$$M_1 = \sigma_x \otimes \sigma_z \otimes \sigma_z \otimes \sigma_0 \otimes \sigma_y$$

$$M_2 = \sigma_z \otimes \sigma_x \otimes \sigma_0 \otimes \sigma_z \otimes \sigma_y$$

$$M_3 = \sigma_0 \otimes \sigma_z \otimes \sigma_x \otimes \sigma_z \otimes \sigma_y$$

$$M_4 = \sigma_z \otimes \sigma_0 \otimes \sigma_z \otimes \sigma_x \otimes \sigma_y$$

The minimum distance is 3, so we can detect and correct any error of weight 1.
The correspondence table between errors of weight one and the performed measurements is

E	M_1	M_2	M_3	M_4
$XIIII$	$+1$	-1	$+1$	-1
$IXIII$	-1	$+1$	-1	$+1$
$IIXII$	-1	$+1$	$+1$	-1
$IIIXI$	$+1$	-1	$+1$	-1
$IIIIX$	$+1$	$+1$	-1	-1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$IIIIZ$	-1	-1	-1	-1

Now suppose we perform the four measurements on M_1E, M_2E, M_3E, M_4E and they return $+1, +1$, $+1, +1$, this means that no error has occurred. However, if one of them returns -1, then we refer to the table to find out which error has occurred. For example, suppose the measurements return $+1, +1, -1, -1$ respectively. Since all the errors are pairwise distinct, we know for sure that the only possible error is a bit flip in the fifth position.

Proposition 2.5.2. *The measurements performed on errors of weight $\leq \lfloor \frac{d-1}{2} \rfloor$ are all pairwise distinct.*

Proof. Let E, E' be two different errors of weight $\leq \lfloor \frac{d-1}{2} \rfloor$, and suppose they return the same measurement. This means that they commute or anti-commute with the same M_j's from the stabiliser group. Now consider the error EE'. For any $M \in S$ we have:

· If E and E' commute with M, then $MEE' = EE'M$.

· If E and E' anti-commute with M, then $MEE' = -EME' = EE'M$

In any case, EE' commutes with all elements in S, which means $EE' \in C(S)$ where $C(S)$ denotes the centraliser. In Theorem 2.2.3 we saw that the minimum distance is equal to the minimum weight of the errors in $C(S) \setminus S$, so

$$wt(EE') \geq d.$$

On the other hand, $wt(E), wt(E') \leq \lfloor \frac{d-1}{2} \rfloor$ so $wt(EE') \leq wt(E) + wt(E') \leq d-1$, a contradiction. □

Chapter 3

The Geometry of Quantum Stabiliser Codes

We will start by finding a way to relate classic codes (both linear and additive) to finite projective spaces over finite fields, and then using the bijection between stabiliser codes and additive codes we found in Chapter 2, the geometry of stabiliser codes will be discussed, as is done in [11].

3.1 Linear and Additive Codes

In Theorem 2.4.6, we found that an additive code exists if and only if a qubit stabiliser code exists and we saw what role n, k, d play in each of them. Using this, we can find a connection between linear codes and finite projective spaces and then adapt it to stabiliser codes.

3.1.1 The Geometry of Linear Codes over $\mathbb{F}_q$

Let C be an $[n, k, d]_q$ code i.e. C is a k-dimensional subspace of $\mathbb{F}_q^n$. Let G be a matrix whose rows are a basis for C. G will have dimensions $k \times n$ since the codewords are of length n and C has dimension k. We can express any element $u \in C$ in terms of this basis:

$$u = aG$$

where $a = (a_1, \ldots, a_k) \in \mathbb{F}_q^k$.

Instead of looking at G's rows, let's take a look at its columns. Let $\mathcal{X}$ be the set of columns of G. Note that it is possible that two columns could be repeated, so G is possibly a multi-set of n vectors in $\mathbb{F}_q^k$.

Remark. The j-th coordinate of $u \in C$ is zero if and only if:

$$u_j = (aG)_j = a_1 z_1^j + \ldots + a_k z_k^j = 0$$

where $(z_1^j, \ldots, z_k^j)$ is the j-th column of G. Note also that this condition is unaltered if instead of z^j we consider a non-zero multiple of z^j.

We will consider $\mathcal{X}$ as a set of n points in $PG(k-1,q)$. Now we want to assert what does it mean in $PG(k-1,q)$ that the minimum distance of C is d.

Theorem 3.1.1. *An $[n,k,d]_q$ linear code over $\mathbb{F}_q$ is equivalent to a set of points $\mathcal{X}$ in $PG(k-1,q)$ where every hyperplane of $PG(k-1,q)$ is incident with at least $n-d$ points of $\mathcal{X}$ and there is a hyperplane which is incident with exactly $n-d$ points of $\mathcal{X}$.*

Proof. Let G be the $n \times k$ matrix whose rows are a basis for an $[n,k,d]_q$ linear code C and let $\mathcal{X}$ be the set of columns of G, which we identify with n points in $PG(k-1,q)$. In $PG(k-1,q)$, any hyperplane is defined as the kernel of a linear form:

$$H_a \equiv a_1 X_1 + \ldots + a_k X_k$$

where $a_i \in \mathbb{F}_q$.
Consider a codeword $u \in C$ with weight w. By the previous remark, such codeword will have a 0 in its j-th coordinate if and only if:

$$a_1 z_1^j + \ldots + a_k z_k^j = 0$$

This means that for each codeword $u = aG$, we have an associated hyperplane $H_a = a_1 X_1 + \ldots + a_k X_k$ which will be incident with as many points as zero coordinates of u. Thus, since the weight w is the number of non-zero coordinates of u, u will have $n-w$ zero coordinates. Therefore, H_a will be incident with $n-w$ points of $\mathcal{X}$, and by Proposition 1.6.7 $n-d \leq n-w$.
Now, we have seen that the minimum distance of the code is equal to the minimum non-zero weight of all codewords, so there must be a codeword in C with exactly $n-d$ zero coordinates and thus, its associated hyperplane in $PG(k-1,q)$ will be incident with exactly $n-d$ points of $\mathcal{X}$. □

3.1.2 The Geometry of Additive Codes over $\mathbb{F}_q$

By Lemma 1.4.6, an additive code C over $\mathbb{F}_q$ is a linear code over $\mathbb{F}_p$ where $q = p^h$ and p is a prime number. This means that there exists a generator element $e \in \mathbb{F}_q$ such that $\{e, e^p, \ldots, e^{p^{h-1}}\}$ is a basis for $\mathbb{F}_q$ over $\mathbb{F}_p$. That is, any element $x \in \mathbb{F}_q$ can be written as

$$x = x_0 e + \ldots + x_{h-1} e^{p^{h-1}} \text{ where } x_i \in \mathbb{F}_p.$$

Therefore $|C| = p^r$ for some r.

The following theorem is the equivalent of Theorem 3.1.1 but for additive codes instead of linear. The idea is the same: considering the columns of a generating matrix G as elements in a projective space. However, where we had $u = aG$ with $a_i \in \mathbb{F}_q$, now the a_i's are in $\mathbb{F}_p$, which slightly complicates the reasoning.

Theorem 3.1.2. *An (n, p^r, d) additive code over $\mathbb{F}_q$ is equivalent to a set $\mathcal{X}$ of $\leq (h-1)$-dimensional subspaces of $PG(r-1,p)$ where every hyperplane of $PG(r-1,p)$ contains at most $n-d$ subspaces of $\mathcal{X}$ and there is a hyperplane which contains exactly $n-d$ subspaces of $\mathcal{X}$.*

Proof. Consider the $r \times n$ matrix G whose rows are a basis for C over $\mathbb{F}_p$. Any element $u \in C$ can be written as

$$u = aG \text{ with } a \in \mathbb{F}_p^r.$$

As before, let $\mathcal{X}$ be the set of columns of G. The elements of $\mathcal{X}$ are vectors in $\mathbb{F}_q$ of length r. However, we will not consider them as points in $PG(r-1,q)$ but as subspaces in $PG(r-1,p)$.
Let $x \in \mathbb{F}_q^r$ be an element of $\mathcal{X}$, i.e. a column of G and take $e \in \mathbb{F}_q$ such that $\{e, e^p, \ldots, e^{p^{h-1}}\}$ is a basis for $\mathbb{F}_q$ over $\mathbb{F}_p$, which we know exists from Lemma 1.4.6.
Now x can be written as

$$x = \sum_{j=0}^{h-1} x_j e^{p^j} \text{ where } x_j \in \mathbb{F}_p^r.$$

We can associate x with the subspace spanned by $x_0, \ldots, x_{h-1}$ in $PG(r-1,p)$ which we denote by l_x. Clearly, l_x has dimension at most $h-1$ in $PG(r-1,p)$.
We can use the same argument as in Theorem 3.1.1 to see that any hyperplane of $PG(r-1,p)$ contains at most $n-d$ subspaces of $\mathcal{X}$ and there is a hyperplane which contains exactly $n-d$ of the subspaces of $\mathcal{X}$. Let $x \in \mathcal{X}$ be the i-th column of G. A codeword $u = aG$ will have a zero in the $i-th$ coordinate if and only if $a \cdot x = 0$, which when we think of x as l_x, is equivalent to saying that l_x is contained in the hyperplane

$$H_a \equiv a_1 X_1 + ... + a_r X_r = 0.$$

Since u has at most $n-d$ zero-coordinates, then the hyperplane $H_a \equiv aX = 0$ will contain at most $n-d$ subspaces of $\mathcal{X}$.
Finally, since the minimum distance d is equal to the minimum non-zero weight of the codewords in C, there must be a codeword which has exactly $n-w = n-d$ zero-coordinates and thus, the hyperplane associated to this codeword will contain exactly $n-d$ subspaces of $\mathcal{X}$. □

3.2 The Geometry of Quantum Stabiliser Codes

In this section, we describe the geometry of stabiliser codes by treating them as additive codes over $\mathbb{F}_4$. The problem of constructing $[[n,k,d]]_2$ stabiliser codes is reduced to the geometrical problem of finding sets of lines $\mathcal{X}$ in $PG(n-k-1,2)$ such that each co-dimension 2 subspace of $PG(n-k-1,2)$ is skew to an even number of the lines in $\mathcal{X}$. A method for asserting the minimum distance geometrically will also be explained.

Recall Theorem 2.4.6 said:

"There is a $[[n,k,d]]_2$ qubit stabiliser code if and only if there is an $(n, 2^{n-k}, d)$ additive code C over $\mathbb{F}_4$ such that $C \subseteq C^{\perp_t}$ and the minimum distance of $C^{\perp_t} \setminus C$ is d."

Now let's try to apply Theorem 3.1.2 to qubit stabiliser codes. We are restricting to the case $q = 4$, $p = 2$, $h = 2$ and $r = n-k$. In this case, the set $\mathcal{X}$ of columns of G gives us a set of $\leq$ 1-dimensional subspaces (that is, either lines or points) in $PG(n-k-1,2)$.

Definition 3.2.1. Two subspaces of a projective space $PG(s,q)$ are skew if they don't intersect.

Definition 3.2.2. A co-dimension m subspace of $PG(s,q)$ is a subspace of dimension $s-m$.

Example 3.2.3. In $PG(2,2)$, a co-dimension 2 subspace is a subspace of dimension 0, a point. similarly, in $PG(3,2)$ a co-dimension 2 subspace is a line.

The following theorem gives a geometrical interpretation of the condition $C \subseteq C^{\perp_t}$.

Theorem 3.2.4. *There is a stabiliser code $Q(S)$, where S is generated by $n-k$ independent commuting elements of $\mathcal{P}_n/\{\pm 1, \pm i\}$ if and only if there is a set of lines $\mathcal{X}$ in $PG(n-k-1,2)$ such that every co-dimension 2 subspace of $PG(n-k-1,2)$ is skew to an even number of the lines in $\mathcal{X}$.*

Proof. $\boxed{\Rightarrow}$ As before, let G be a $(n-k)\times n$ matrix whose rows are a basis for $C = \tau(S)$ over $\mathbb{F}_2$. This means that any codeword $u \in C$ can be written as

$$u = aG \text{ where } (a_1, \ldots, a_{n-k}) \in \mathbb{F}_2^{n-k}$$

Recall the definition of $C^{\perp_t} := \{v \in C | (u,v)_t = 0 \text{ for all } u \in C\}$ where

$$(u,v)_t = \sum_{j=1}^{n} u_j v_j^2 + v_j u_j^2.$$

Thus, $C \subseteq C^{\perp_t}$ if and only if $(u,v)_t = 0 \ \forall u, v \in C$.

The sum in $\mathbb{F}_4$ is modulo two which implies that we can change $-$ for $+$ in the expression of $(u,v)_t$ and that for $(u,v)_t$ to be zero we need an even number (possibly zero) of the terms to be different from 0.
Consider now any two codewords $u = aG$ and $v = bG$ in C. Let $x \in \mathcal{X}$ be the j-th column of G so we can write $u_j = ax$ and $v_j = bx$ for the j-th coordinates of u and v. Rewritting the terms of $(u,v)_t$ gives

$$u_j v_j^2 + v_j u_j^2 = (a \cdot x)(b \cdot x)^2 + (b \cdot x)(a \cdot x)^2$$

Now take a basis $\{e, e^2\}$ for $\mathbb{F}_4$ over $\mathbb{F}_2$. This means any element $x \in \mathbb{F}_4^{n-k}$ can be expressed as a linear combination of e and e^2. Thus, we can express the j-th column of G as

$$x = x_0 e + x_1 e^2 \text{ with } x_0, x_1 \in \mathbb{F}_2^{n-k}.$$

Substituting this new expression of x and using that the sum in $\mathbb{F}_2$ is modulo two, we get

$$\begin{aligned}
&u_j v_j^2 + v_j u_j^2 = [a(x_0 e + x_1 e^2)][b(x_0 e + x_1 e^2)]^2 + [b(x_0 e + x_1 e^2)][a(x_0 e + x_1 e^2)]^2 \\
&= [(a \cdot x_0)e + (a \cdot x_1)e^2][(b \cdot x_0)^2 e^2 + (b \cdot x_1)^2 e] + [(b \cdot x_0)e + (b \cdot x_1)e^2][(a \cdot x_0)^2 e^2 + (a \cdot x_1)^2 e] \\
&= (a \cdot x_0)(b \cdot x_0) + (a \cdot x_1)(b \cdot x_0)e + (a \cdot x_0)(b \cdot x_1)e^2 + (a \cdot x_1)(b \cdot x_1) \\
&\quad + (b \cdot x_0)(a \cdot x_0) + (b \cdot x_1)(a \cdot x_0)e + (b \cdot x_0)(a \cdot x_1)e^2 + (b \cdot x_1)(a \cdot x_1) \\
&= (a \cdot x_1)(b \cdot x_0)(e + e^2) + (a \cdot x_0)(b \cdot x_1)(e + e^2) \\
&= (a \cdot x_1)(b \cdot x_0) + (a \cdot x_0)(b \cdot x_1)
\end{aligned}$$

This will be zero iff the matrix

$$\begin{pmatrix} a \cdot x_0 & a \cdot x_1 \\ b \cdot x_0 & b \cdot x_1 \end{pmatrix}$$

has determinant zero which is equivalent to say that $\exists \lambda_0, \lambda_1 \in \mathbb{F}_2$ such that

$$\begin{aligned}
a \cdot (\lambda_0 x_0 + \lambda_1 x_1) = 0 \\
b \cdot (\lambda_0 x_0 + \lambda_1 x_1) = 0
\end{aligned}$$

The last two equations are equivalent to saying that the point $\lambda_0 x_0 + \lambda_1 x_1$ which is on the line l_x spanned by x_0, x_1, is in both hyperplanes $\pi_a \equiv a \cdot X = 0$ and $\pi_b \equiv b \cdot X = 0$ in $PG(n-k-1,2)$. Finally, any co-dimension 2 subspace of $PG(n-k-1,2)$ can be realised as the intersection of two hyperplanes.

$\boxed{\Leftarrow}$ The fact that τ is a bijection, which we saw in Lemma 2.4.1, proves the backwards implication. □

Definition 3.2.5 (Quantum set of lines)**.** $\mathcal{X}$ is a quantum set of lines of $PG(n-k-1,2)$ if every co-dimension 2 subspace of $PG(n-k-1,2)$ is skew to an even number of the lines in $\mathcal{X}$.

Definition 3.2.6 (Minimum distance of a quantum set of lines)**.** Given a quantum set of lines $\mathcal{X}$ of $PG(n-k-1,2)$, we define $d(\mathcal{X})$ to be

- *($k = 0$)* The minimum over all hyperplanes h in $PG(n-1,2)$ of the number of lines of $\mathcal{X}$ not contained in h.

- *($k \neq 0$)* The minimum size of a dependent set of points in $\mathcal{X}$, discounting the dependencies such that the lines of $\mathcal{X}$ not giving the dependent points are contained in a hyperplane of $PG(n-k-1,2)$ containing the dependent points.

Theorem 3.2.7. *There is an $[[n,k,d]]_2$ stabiliser code if and only if there is a quantum set of lines $\mathcal{X}$ in $PG(n-k-1,2)$ such that $d(\mathcal{X}) = d$.*

Proof. We have proven the existence of $\mathcal{X}$ in the previous theorem, so we only have to check that the minimum distance d is equal to $d(\mathcal{X})$.

$\underline{k = 0}$
Let C be an $[[n,0,d]]_2$ stabiliser code. In this case, the minimum distance is simply the minimum non-zero weight of the codewords in C.
Let G be the $n \times n$ matrix with elements of $\mathbb{F}_4$ such that any codeword $u \in C$ can be written as

$$u = aG \text{ where } a \in \mathbb{F}_2^n$$

Let x_i denote the i-th column of G. A codeword $u = aG$ will have a zero in its i-th coordinate iff $u_i = a x_i = 0$ which means that the line associated to the column x_i will be contained in the hyperplane $aX = 0$. This means that a codeword u has weight w (that is, w of its coordinates are different from zero) iff w of the lines associated with the columns of G are not contained in the hyperplane $aX = 0$ where a comes from $u = aG$. Thus, since the minimum distance is the minimum weight over all codewords in C, $d(\mathcal{X})$ is the minimum over all hyperplanes h of $PG(n-1,2)$ of the number of lines in $\mathcal{X}$ not contained in h.

$\underline{k \neq 0}$
Let C be an $[[n,k,d]]_2$ stabiliser code. In Proposition 1.6.7 we saw that the minimum distance of C is equal to the minimum weight in $C^{\perp_t} \setminus C$.
As before, let G be the $(n-k) \times n$ matrix with elements from $\mathbb{F}_4$ such that any codeword $u \in C$ can be written as

$$u = aG \text{ with } a \in \mathbb{F}_2^{n-k}$$

Consider the j-th column of G, x_j. As before, we can take e, a generating element of $\mathbb{F}_4$ over $\mathbb{F}_2$ such that any column of G can be expressed in the basis $\{e, e^2\}$.

$$x_j = y_j e + z_j e^2$$

where $y_j, z_j \in \mathbb{F}_2^{n-k}$.
For the forward implication, take a codeword $v \in C^{\perp_t}$ with weight d. Let D be the set of non-zero coordinates of v, so by definition since the weight of a codeword is the number of non-zero coordinates, we have $|D| = d$. On the other hand, since $v \in C^{\perp_t}$, we have

$$\sum_{j \in D} (v_j^2 x_j + x_j^2 v_j) = 0.$$

Since the sum is modulo 2 and any $\alpha \in \mathbb{F}_2$ satisfies $\alpha^2 = \alpha$, substituting $x_j = y_j e + z_j e^2$ gives

$$\begin{aligned}
&\sum_{j \in D} v_j^2 (y_j e + z_j e^2) + v_j (y_j e + z_j e^2)^2 = \sum_{j \in D} v_j^2 (y_j e + z_j e^2) + v_j (y_j^2 e^2 + z_j^2 e^4) \\
&= \sum_{j \in D} (v_j^2 e + v_j e^2) y_j + (v_j^2 e^2 + v_j e) z_j = 0.
\end{aligned}$$

For each $j \in D$, the summand is a point on the line spanned by y_j and z_j which is the line l_j corresponding to the j-th column of G. This implies that there are d dependent points on the lines $\{l_{x_j} | j \in D\}$.
However, we must not consider the minimum weight in $C^{\perp_t}$ but the minimum weight in $C^{\perp_t} \setminus C$. The codeword v is in C if and only if there is an $a \in \mathbb{F}_2^{n-k}$ such that $v = aG$. For each coordinate i such that $v_i = 0$, we have $a x_i = 0$ where x_i is the i-th column of G. Thus, we have that the lines $\{l_i | i \in \{1, \ldots, n\} \setminus D\}$ are all contained in the hyperplane $aX = 0$. On the other hand, the hyperplane $aX = 0$ must be incident with the lines $\{l_j | j \in D\}$ in the dependent points since each dependent point $v_j^2 x_j + x_j^2 v_j$ is in $aX = 0$ if and only if

$$a(v_j^2 x_j + x_j^2 v_j) = 0 \Leftrightarrow v_j^2 (a x_j) + v_j (a x_j^2) = 0 \Leftrightarrow v_j (a x_j) + a^2 x_j^2 = 0 \Leftrightarrow v_j = a x_j \qquad j \in D$$

since the sum is modulo two and $a \in \mathbb{F}_2^{n-k}$.
For the other implication, if we take $d(\mathcal{X})$, i.e. if we find the minimum set of points which give us the dependency as described in the definition of $d(\mathcal{X})$, that is, the $(v_j^2 e + v_j e^2) y_j + (v_j^2 e^2 + v_j) z_j$'s, then given all the $\lambda_j = (v_j^2 e + v_j e^2)$ and $\mu_j = (v_j^2 e^2 + v_j)$ the v_j's can be determined.
We have proved that the minimum weight in $C^{\perp_t} \setminus C$ is equal to the minimum number of dependent points in the lines of $\mathcal{X}$ such that the lines not giving such dependencies are not contained in a hyperplane of $PG(n-k-1, 2)$ that contains the dependent set of points. □

Note that the proof still works in the case $d = 2$ where some lines can have multiplicity greater than 1.

Definition 3.2.8. A planar pencil of lines in $PG(n-k-1, q)$ is a set of lines which are contained in a plane and are incident with one point in that plane.

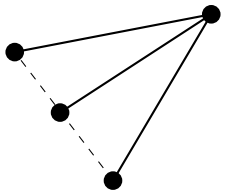


Figure 3.1: A planar pencil of lines.

In fact, a set of lines $\mathcal{X}$ is a quantum set of lines if and only if it is the sum modulo two of pencils of lines. The sum modulo two of two pencils of lines P_1 and P_2 is defined as the union of all lines in P_1 and P_2 removing those that appear an even number of times.

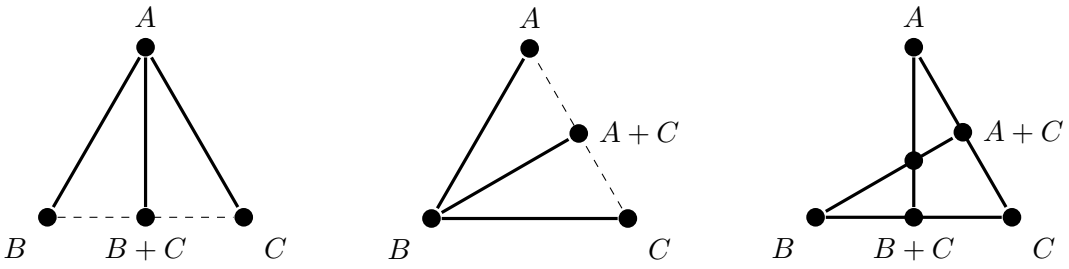


Figure 3.2: The sum modulo two of pencils (a) and (b) gives (c).

We go on to prove a few lemmas before showing that any quantum set of lines is the sum modulo two of pencils of lines.

Lemma 3.2.9. *A planar pencil of lines $\mathcal{X}$ in $PG(n, q)$ is a quantum set of lines.*

Proof. We have to see that any co-dimension 2 subspace V is skew to an even number of the lines in $\mathcal{X}$. By definition, V has dimension $n-2$ and a planar pencil of lines has dimension 2, therefore

$$V \cap \mathcal{X} \neq \emptyset.$$

Since the intersection of $\mathcal{X}$ and V is not empty, V cannot be skew to the three lines in the pencil. Suppose that V is skew to only one of the lines. Without loss of generality, suppose it is skew to $\langle A, B\rangle$ in the pencil (a) from Figure 3.2, then $B + C$ and C are in V and since V is a subspace, this means that $B = (B + C) + C$ is also in V, a contradiction. This implies that V can only be skew to either 0 or 2 lines of $\mathcal{X}$, therefore $\mathcal{X}$ is a quantum set of lines. □

Lemma 3.2.10. *The sum modulo two of quantum sets of lines is a quantum set of lines.*

Proof. Suppose $\mathcal{X}$ and $\mathcal{X}'$ are two quantum sets of lines. There are two cases to consider.
In the first place, if $\mathcal{X}$ and $\mathcal{X}'$ have no lines in common, then any co-dimension two subspace will be skew to an even number of lines of $\mathcal{X}$ (because $\mathcal{X}$ is a quantum set of lines) and also skew to an even number of lines of $\mathcal{X}'$ (by the same reason), since the sum of two even numbers is also an even number, we are done.

On the other hand, if $\mathcal{X}$ and $\mathcal{X}'$ have some line in common, then we risk that a co-dimension two subspace V is skew to an odd number of lines. However, this is not a problem since we are summing $\mathcal{X}$ and $\mathcal{X}'$ modulo two, so the lines in common of $\mathcal{X}$ and $\mathcal{X}'$ will disappear, thus leaving us with an even number of lines that are skew to V. □

Definition 3.2.11 (r-sputnik)**.** An r-sputnik is a set of $(r+1)$ concurrent lines in an $r-$dimensional subspace π such that any r of them span π.

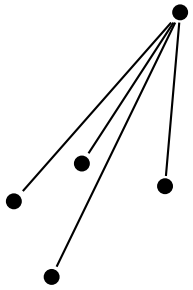


Figure 3.3: A 3-sputnik.

Note that a 2-sputnik is a set of 3 concurrent lines in a plane such that any two of the lines span the plane, which is precisely a planar pencil of lines.

Lemma 3.2.12. *An r-sputnik is the sum modulo two of pencils of lines.*

Proof. To see this we will take an r-sputnik and proceed to add pencils of lines until we are left with a single planar pencil of lines. Then we can reverse the process and build the r-sputnik by adding pencils of lines modulo two, proving the statement.
Suppose that $\mathcal{X}$ is an r-sputnik, that is, a set of $(r+1)$ concurrent lines spanning an r-dimensional subspace π. Take any two lines ℓ, ℓ' in $\mathcal{X}$. Consider, on the one hand, the plane δ spanned by ℓ and ℓ' and on the other hand, $\gamma := \mathcal{X} \setminus \{\ell, \ell'\}$ which is an $(r-1)$ subspace since removing one line of $\mathcal{X}$ leaves us with an r-dimensional subspace. γ and δ intersect in a line ℓ'' since

$$\dim(\gamma \cap \delta) = \dim\gamma + \dim\delta - \dim(\gamma \cup \delta) = (r-1) + 2 - r = 1$$

The set $\{\ell, \ell', \ell''\}$ is the first pencil of lines we will add. Adding this pencil leaves us with $\mathcal{X}_1 = \mathcal{X} \setminus \{\ell, \ell'\} \cup \ell''$. Note that ℓ'' is contained in γ and so $\mathcal{X}_1$ is an $(r-1)$-sputnik since it is a set of r lines with

$$\dim(\mathcal{X}_1) = \dim\gamma + \dim\ell'' - \dim\gamma \cap \ell'' = (r-1) + 1 - 1 = r - 1$$

We can continue adding pencils of lines modulo two in this way until we are left with a 2-sputnik, which is a planar pencil of lines. Finally, by reversing the process we can find an expression of $\mathcal{X}$ as the sum modulo two of these pencils of lines. □

Note that in particular, by Lemma 3.2.9 a planar pencil is a quantum set of lines and by Lemma 3.2.10 the sum modulo two of quantum sets of lines is a quantum set of lines, so we just proved that an r-sputnik is a quantum set of lines.

Lemma 3.2.13. *Let $\mathcal{X}$ be a quantum set of lines. There is a set of dependent points D such that each point of D is incident with a different line of $\mathcal{X}$.*

Proof. Consider π, the subspace spanned by the lines of $\mathcal{X}$ and take any line $\ell \in \mathcal{X}$. Let π' be the subspace spanned by the lines of $\mathcal{X} \setminus \{\ell\}$. Either $\ell \cap \pi' = \{\emptyset\}$, $l \cap \pi' = \{x\}$ where x is a point or $\ell \cap \pi' = \{\ell\}$. In the first case, π' is a co-dimension 2 subspace of π and it is only skew to $\ell \in \mathcal{X}$, a contradiction with the definition of a quantum set of lines.
In any of the other two cases there is a point of ℓ in π'. Let x be this point. Since π' is spanned by the lines of $\mathcal{X} \setminus \ell$, it is the sum of points of these lines. Suppose that two points y, y' of this dependent set are in the same line ℓ' of $\mathcal{X}$, then we know that there is always a third point $y'' \in \ell'$ that we can take instead of y and y'. We don't have to worry about y'' being in the dependent set of points because since y, y' and y'' are on the same line, they sum to zero.

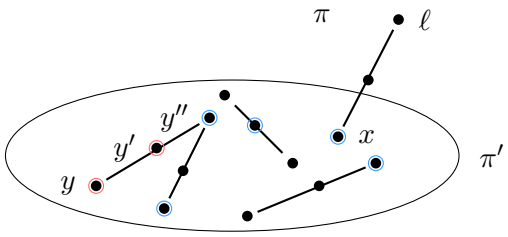


Figure 3.4: The subspace π.

Hence, we have found a dependent set of points such that each point is in a different line of $\mathcal{X}$. □

Lemma 3.2.14. *A quantum set of three lines is a planar pencil of lines.*

Proof. Let $\mathcal{X} = \{\ell_1, \ell_2, \ell_3\}$ be a quantum set of lines. The lines of $\mathcal{X}$ span at least a plane and at most a 5-dimensional space.

Suppose the three lines of $\mathcal{X}$ span $PG(5, 2)$

This means that the lines have to be skew. The subspace spanned by ℓ_1 and ℓ_2 is a co-dimension 2 subspace of $PG(5, 2)$ since it has dimension 4. But this is a contradiction since this co-dimension 2 subspace is only skew to ℓ_3, and thus an odd number of the lines in $\mathcal{X}$.

Suppose the lines of $\mathcal{X}$ span $PG(4, 2)$

By the same reasoning as before, the subspace spanned by ℓ_1 and $x \in \ell_2$ is a co-dimension 2 subspace of $PG(4, 2)$ and it is only skew to ℓ_3 therefore a contradiction with the definition of a quantum set of lines.

Suppose the lines of $\mathcal{X}$ span $PG(3, 2)$

Since the lines span $PG(3, 2)$ and not $PG(2, 2)$, they must be not be co-planar. It is easy to see that if the lines are not mutually skew, then there is some co-dimension 2 subspace (line) that is skew to either 3 or 1 line. Thus, we are left with 3 mutually skew lines in $PG(3, 2)$. Now, since the lines of $PG(3, 2)$ contain 3 points each, there are 9 points in $\mathcal{X}$. A co-dimension 2 subspace of $PG(3, 2)$ is a line, so any line that is incident with two of the lines in $\mathcal{X}$ must also be incident with the third. Take a look at Figure 3.5, where the lines of $\mathcal{X}$ are in black. Since there is a line joining any two points (by definition of a projective space), we have that for each point in a line of $\mathcal{X}$ there is a line joining it to the other six points of $\mathcal{X}$. In Figure 3.5, these are the lines in blue. Finally, we know

that any point in $PG(3,2)$ is in 7 lines, so the remaining lines are painted in red. This configuration gives us a total of 39 lines (3 in black, 9 in blue and 27 in red), but there are only 35 lines in $PG(3,2)$!

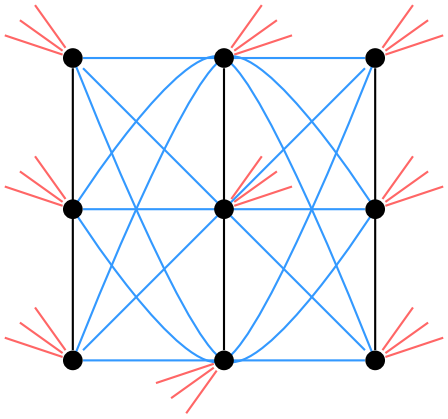


Figure 3.5: Configuration of the lines in $PG(3,2)$.

All cases are ruled out except that the lines of $\mathcal{X}$ span $PG(2,2)$. For them to be a quantum set of lines, we need that any point (which is a co-dimension two subspace in $PG(2,2)$) is skew to an even number of the lines. This means that any point has to be either in one of the lines or in three of them which implies that our lines are incident with all the points of $PG(2,2)$ and thus they must be concurrent ($PG(2,2)$ has only 7 points).
We are left with three concurrent lines in $PG(2,2)$ which is precisely a planar pencil of lines. □

Theorem 3.2.15. *Any quantum set of lines is the sum modulo two of pencils of lines.*

Proof. Let $\mathcal{X}$ be a quantum set of $r+1$ lines. To see that $\mathcal{X}$ is the sum modulo two of a series of pencils of lines, we will find an r-sputnik $\mathcal{X}'$ and a set of r planar pencils $\mathcal{L}_1, \ldots, \mathcal{L}_r$, such that adding modulo two $\mathcal{X}$, $\mathcal{X}'$ and $\mathcal{L}_1, \ldots, \mathcal{L}_r$ results in a quantum set of $|\mathcal{X}| - 1 = r$ lines.
By Lemma 3.2.12, $\mathcal{X}'$ is the sum modulo two of some pencils of lines. Repeating this process, we will find a set of pencils of lines such that adding them modulo two to $\mathcal{X}$ will give us a quantum set of 3 lines. By Lemma 3.2.14 these 3 remaining lines are a planar pencil of lines and so $\mathcal{X}$ is the sum modulo two of some planar pencils of lines.
First of all, let $\{\ell_1, \ldots, \ell_{r+1}\}$ denote the lines of $\mathcal{X}$. By Lemma 3.2.13, there is a set of $r+1$ dependent points $\{x_1, \ldots, x_{r+1}\}$, such that each dependent point x_i is incident with ℓ_i for $i = 1, \ldots, r+1$.
Let x_0 denote a point in $\ell_{r+1} \setminus \{x_{r+1}\}$, and consider the lines $l'_1, \ldots, l'_r$ where

$$l'_i = \langle x_0, x_i \rangle.$$

Let $\mathcal{X}'$ be the r-sputnik given by

$$\mathcal{X}' = \{\ell'_1, \ldots, \ell'_r\} \cup \{\ell_{r+1}\}.$$

Now for each $i \in \{1, \ldots, r\}$ consider the pencil $\mathcal{L}_i = \{\ell_i, \ell'_i, \alpha_i\}$ where α_i is the line that completes

the pencil given by ℓ_i and ℓ'_i. We are left with these sets of lines:

$$\begin{aligned}\mathcal{X} &= \{\ell_1, \ldots, \ell_{r+1}\}\\ \mathcal{X}' &= \{\ell'_1, \ldots, \ell'_r, \ell_{r+1}\}\\ \mathcal{L}_1 &= \{\ell_1, \ell'_1, \alpha_1\}\\ &\vdots\\ \mathcal{L}_r &= \{\ell_r, \ell'_r, \alpha_r\}\end{aligned}$$

Note that all the lines appear twice except for $\alpha_1, \ldots, \alpha_r$ so if we sum modulo two $\mathcal{X}$, $\mathcal{X}'$ and $\mathcal{L}_1, \ldots, \mathcal{L}_r$ we are left with a set of r lines. □

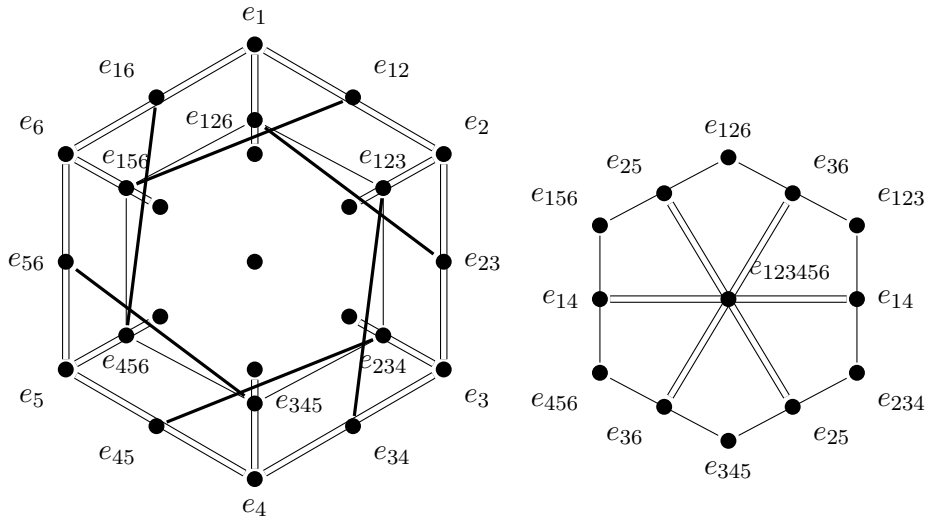


Figure 3.6: The quantum set of lines (the thicker lines) giving a $[[6, 0, 4]]_2$ code is the sum modulo two of 16 pencils of lines.

3.3 Examples

The previous sections describe the algorithm to translate linear, additive and quantum stabiliser codes into sets of lines in a finite projective space and vice-versa. In this section, a set of examples are presented.

3.3.1 From Stabiliser Codes to Quantum Sets of Lines

Example 3.3.1 (The Shor Code)**.** In Section 2.3 we saw the 8 elements of $\mathcal{P}_9/\{\pm 1, \pm i\}$ that generate S for Shor's 9-qubit stabiliser code. We also established in Proposition 2.3.1 that the minimum distance is 3 and thus, this is a $\mathbf{[[9, 1, 3]]_2}$ code.

To find the corresponding quantum set of lines, first we have to find the $(n-k) \times n$ generating matrix G whose row span over $\mathbb{F}_2$ is $C = \tau(S)$. In this case

$$G = \begin{pmatrix} e & e & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & e & e & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & e & e & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & e & e & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & e & e & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & e & e \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Now if we take $e \in \mathbb{F}_4 \setminus \mathbb{F}_2$ we can express each column x^i of G as $x^i = x^i_0 e + x^i_1 e^2$. Let $e_1, \ldots, e_8$ denote the vectors in the canonical basis of $\mathbb{F}_2^8$.

$$x^1 = (e_1 + e_7)e + (e_7)e^2 \qquad x^6 = (e_4 + e_7 + e_8)e + (e_7 + e_8)e^2$$
$$x^2 = (e_1 + e_2 + e_7)e + (e_7)e^2 \qquad x^7 = (e_5 + e_8)e + (e_8)e^2$$
$$x^3 = (e_2 + e_7)e + (e_7)e^2 \qquad x^8 = (e_5 + e_6 + e_8)e + (e_8)e^2$$
$$x^4 = (e_3 + e_7 + e_8)e + (e_7 + e_8)e^2 \qquad x^9 = (e_6 + e_8)e + (e_8)e^2$$
$$x^5 = (e_3 + e_4 + e_7 + e_8)e + (e_7 + e_8)$$

The resulting quantum set of lines is

$$\{\langle e_1, e_7\rangle, \langle e_1 + e_2, e_7\rangle, \langle e_2, e_7\rangle, \langle e_3, e_7 + e_8\rangle, \langle e_3 + e_4, e_7 + e_8\rangle, \langle e_4, e_7 + e_8\rangle, \langle e_5, e_8\rangle, \langle e_5 + e_6, e_8\rangle, \langle e_6, e_8\rangle\}$$

which are the 9 lines of the form $\langle x^i_0, x^i_1\rangle$.

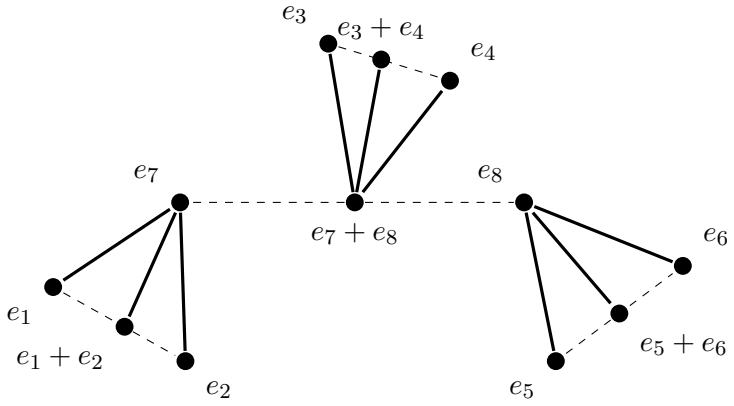


Figure 3.7: The quantum set of lines corresponding to the Shor code in $PG(7, 2)$.

These lines in $PG(7, 2)$ are certainly a quantum set of lines, since they can be obtained as the sum modulo two of three pencils of lines: $\{\langle e_1, e_7\rangle, \langle e_1 + e_2, e_7\rangle, \langle e_2, e_7\rangle\}$, $\{\langle e_3, e_7 + e_8\rangle, \langle e_3 + e_4, e_7 + e_8\rangle, \langle e_4, e_7 + e_8\rangle\}$ and $\{\langle e_5, e_8\rangle, \langle e_5 + e_6, e_8\rangle, \langle e_6, e_8\rangle\}$.
In Proposition 2.3.1 we saw that the minimum distance for the Shor Code is 3. Geometrically, we can check that $d(\mathcal{X}) = 3$.
We have a three point dependency (we can draw a line that intersects the lines of $\mathcal{X}$ in 3 points on different lines). For example, without loss of generality, consider the line $\langle e_1, e_2\rangle$ which intersects

$\mathcal{X}$ in $\{e_1, e_1+e_2, e_2\}$, each one in a different quantum line.
Is this the minimum number of dependent points we should consider? yes. We also have a dependency on 2 points (there are two lines in $\mathcal{X}$ that intersect), but we must not take them into account when calculating $d(\mathcal{X})$. Suppose without loss of generality, that the dependent points are $\{e_7\}$ coming from lines $\langle e_7, e_1\rangle$ and $\langle e_7, e_1+e_2\rangle$. The 7 lines not giving this two point dependency are all contained in the hyperplane $\langle e_2, \ldots, e_8\rangle$. We can conclude that $d(\mathcal{X})$ is 3.

Remark. If we interchange e and 1 in the previous G matrix, the resulting quantum lines are the same. Does this mean that there is more than one quantum code for each quantum set of lines? The answer is no. The matrix G depends on $\tau(S)$ and our only restriction when defining τ was that $\theta(\sigma\varsigma) = \theta(\sigma) + \theta(\varsigma)$ for all $\sigma, \varsigma \in \{\sigma_0, \sigma_x, \sigma_y, \sigma_z\}$.
In particular we chose θ to be

$$\begin{aligned} \theta : \{\sigma_0, \sigma_x, \sigma_y, \sigma_z\} &\to \mathbb{F}_4 \\ \sigma_0 &\mapsto 0 \\ \sigma_x &\mapsto 1 \\ \sigma_z &\mapsto e \\ \sigma_y &\mapsto e^2 \end{aligned}$$

However, other possible θ could be used that also satisfy this condition. These other choices of theta give us different G matrices, all resulting in the same quantum code and the same quantum set of lines.

Example 3.3.2. There is a $[[\mathbf{5, 0, 3}]]_\mathbf{2}$ stabiliser code where S is generated by

$$\begin{aligned} M_1 &= \sigma_x \otimes \sigma_z \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_z \\ M_2 &= \sigma_z \otimes \sigma_x \otimes \sigma_z \otimes \sigma_0 \otimes \sigma_0 \\ M_3 &= \sigma_0 \otimes \sigma_z \otimes \sigma_x \otimes \sigma_z \otimes \sigma_0 \\ M_4 &= \sigma_0 \otimes \sigma_0 \otimes \sigma_z \otimes \sigma_x \otimes \sigma_z \\ M_5 &= \sigma_z \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_z \otimes \sigma_x \end{aligned} \qquad G = \begin{pmatrix} 1 & e & 0 & 0 & e \\ e & 1 & e & 0 & 0 \\ 0 & e & 1 & e & 0 \\ 0 & 0 & e & 1 & e \\ e & 0 & 0 & e & 1 \end{pmatrix}$$

The quantum set of lines $\mathcal{X}$ of this code are lines in $PG(4, 2)$. Precisely, let $e_1, \ldots, e_5$ denote the elements of the canonical basis for $\mathbb{F}_2^5$

$$\mathcal{X} = \{\langle e_2+e_5, e_1\rangle, \langle e_1+e_3, e_2\rangle, \langle e_2+e_4, e_3\rangle, \langle e_3+e_5, e_4\rangle, \langle e_1+e_4, e_5\rangle\}$$

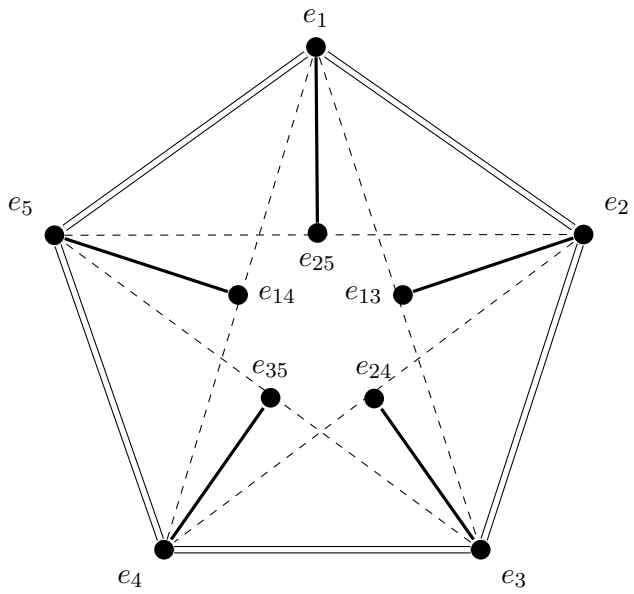


Figure 3.8: The quantum set of lines for the $[[5,0,3]]_2$ code is the sum modulo two of five pencils of lines.

The resulting set of lines $\mathcal{X}$ is a quantum set of lines, since we can write $\mathcal{X}$ as the sum modulo two of these 5 pencils:

$$\{\langle e_1, e_5\rangle, \langle e_1, e_2 + e_5\rangle, \langle e_1, e_2\rangle\}$$
$$\{\langle e_2, e_1\rangle, \langle e_2, e_1 + e_3\rangle, \langle e_2, e_3\rangle\}$$
$$\{\langle e_3, e_2\rangle, \langle e_3, e_2 + e_4\rangle, \langle e_3, e_4\rangle\}$$
$$\{\langle e_4, e_3\rangle, \langle e_4, e_3 + e_5\rangle, \langle e_4, e_5\rangle\}$$
$$\{\langle e_5, e_4\rangle, \langle e_5, e_1 + e_4\rangle, \langle e_5, e_1\rangle\}$$

We have $k = 0$ so the minimum distance $d(\mathcal{X})$ is defined as the minimum over all hyperplanes h in $PG(4,2)$ of the number of lines in $\mathcal{X}$ not contained in h. Any 3 lines in $\mathcal{X}$ span a 5-dimensional subspace so they cannot be contained in a hyperplane of $PG(4,2)$ which is a 4-dimensional subspace. We can conclude that $d(\mathcal{X}) = 3$.

Example 3.3.3. A $[[\mathbf{5,1,3}]]_\mathbf{2}$ stabiliser code which will give us a set of lines in $PG(3,2)$.

$$\begin{aligned} M_1 &= \sigma_x \otimes \sigma_z \otimes \sigma_z \otimes \sigma_x \otimes \sigma_0 \\ M_2 &= \sigma_0 \otimes \sigma_x \otimes \sigma_z \otimes \sigma_z \otimes \sigma_x \\ M_3 &= \sigma_x \otimes \sigma_0 \otimes \sigma_x \otimes \sigma_z \otimes \sigma_z \\ M_4 &= \sigma_z \otimes \sigma_x \otimes \sigma_0 \otimes \sigma_x \otimes \sigma_z \end{aligned} \qquad G = \begin{pmatrix} 1 & e & e & 1 & 0 \\ 0 & 1 & e & e & 1 \\ 1 & 0 & 1 & e & e \\ e & 1 & 0 & 1 & e \end{pmatrix}$$

The resulting set of lines $\mathcal{X}$ is

$$\{\langle e_1 + e_3, e_4\rangle, \langle e_1, e_2 + e_4\rangle, \langle e_3, e_1 + e_2\rangle, \langle e_1 + e_4, e_2 + e_3\rangle, \langle e_2, e_3 + e_4\rangle\}$$

These lines are a spread of 5 lines in $PG(3,2)$, which are the sum modulo two of the 5 pencils of lines in Figure 3.9, and thus they are a quantum set of lines.

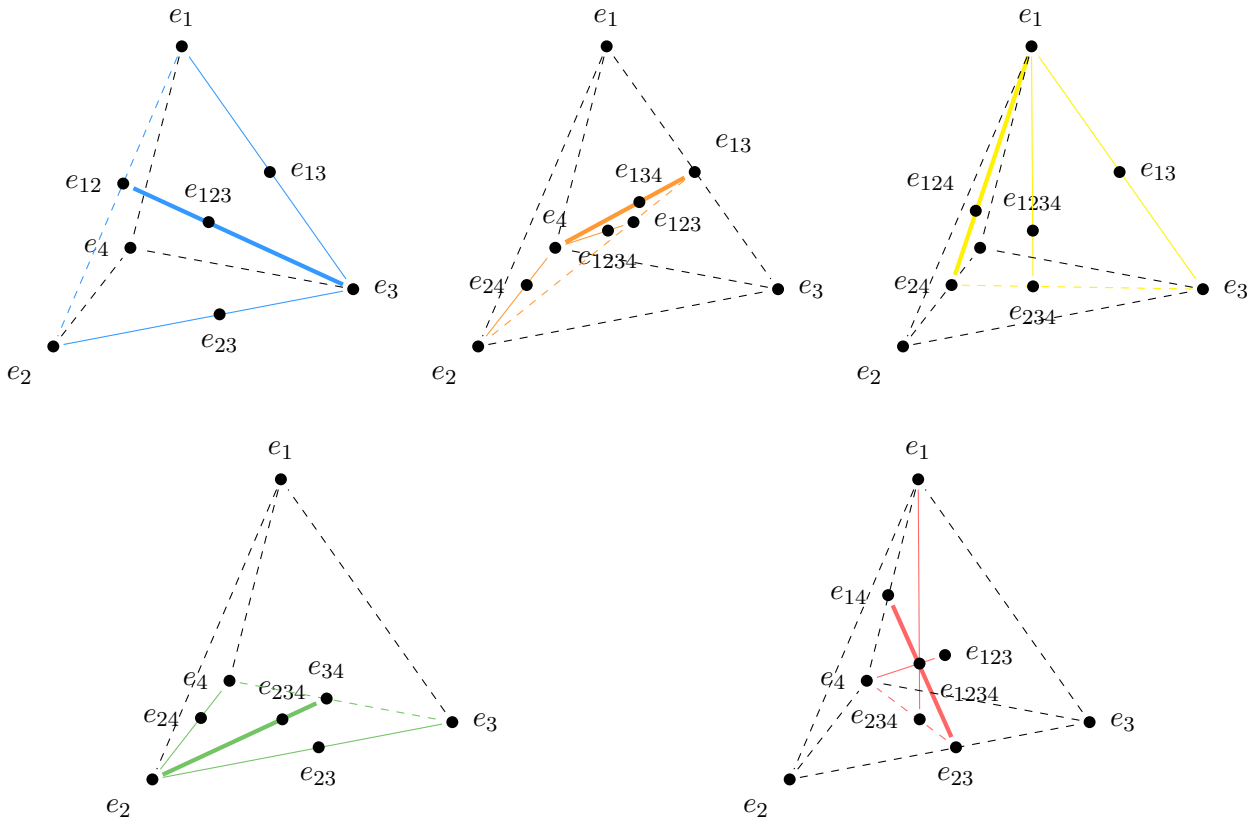


Figure 3.9: Five pencils giving the $[[5,1,3]]_2$.

Adding these pencils modulo two results in

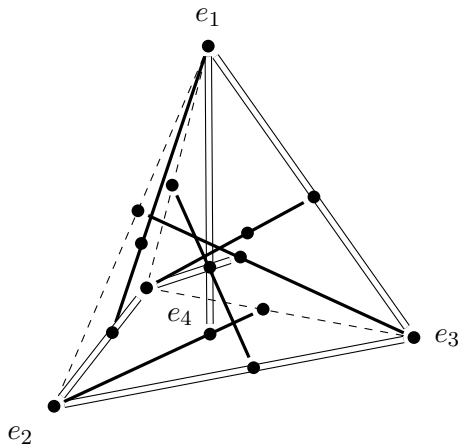


Figure 3.10: The quantum set of lines for the $[[5,1,3]]_2$ code is a spread in $PG(3,2)$.

There is a three point dependency in $\mathcal{X}$: any line of $PG(3,2)$ has three points, each one on a different line of $\mathcal{X}$. For instance consider the three point dependency given by points $\{e_1, e_{13}, e_3\}$. The lines of $\mathcal{X}$ not containing these points are $\langle e_2, e_{34}\rangle$ and $\langle e_{23}, e_{14}\rangle$ which span a 3-dimensional subspace and thus, they are not contained in a hyperplane. Since the lines of $\mathcal{X}$ are skew, there is no two point dependency, so we can conclude $d(\mathcal{X}) = 3$.

Example 3.3.4 (Steane Code)**.** The Steane Code is a $\mathbf{[[7,1,3]]_2}$ stabiliser code, whose stabiliser group S is generated by:

$$
\begin{aligned}
M_1 &= \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_x \otimes \sigma_x \otimes \sigma_x \otimes \sigma_x \\
M_2 &= \sigma_0 \otimes \sigma_x \otimes \sigma_x \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_x \otimes \sigma_x \\
M_3 &= \sigma_x \otimes \sigma_0 \otimes \sigma_x \otimes \sigma_0 \otimes \sigma_x \otimes \sigma_0 \otimes \sigma_x \\
M_4 &= \sigma_0 \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_z \otimes \sigma_z \otimes \sigma_z \otimes \sigma_z \\
M_5 &= \sigma_0 \otimes \sigma_z \otimes \sigma_z \otimes \sigma_0 \otimes \sigma_0 \otimes \sigma_z \otimes \sigma_z \\
M_6 &= \sigma_z \otimes \sigma_0 \otimes \sigma_z \otimes \sigma_0 \otimes \sigma_z \otimes \sigma_0 \otimes \sigma_z
\end{aligned}
\qquad
G = \begin{pmatrix}
0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 1 & 1 & 0 & 0 & 1 & 1 \\
1 & 0 & 1 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & e & e & e & e \\
0 & e & e & 0 & 0 & e & e \\
e & 0 & e & 0 & e & 0 & e
\end{pmatrix}
$$

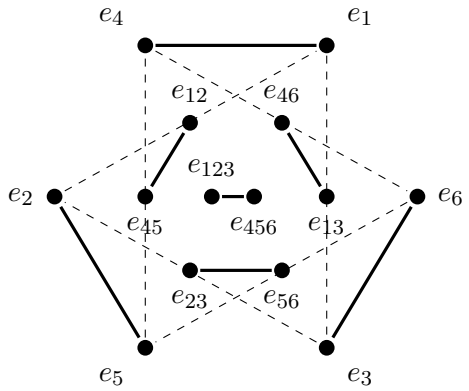


Figure 3.11: The 6 quantum lines for the Steane Code in $PG(5,2)$.

These lines can be written as the sum modulo two of these 11 pencils:

$\{\langle e_{123}, e_6\rangle, \langle e_{123}, e_{46}\rangle, \langle e_{123}, e_4\rangle\}$
$\{\langle e_{123}, e_5\rangle, \langle e_{123}, e_{56}\rangle, \langle e_{123}, e_6\rangle\}$
$\{\langle e_{123}, e_4\rangle, \langle e_{123}, e_{45}\rangle, \langle e_{123}, e_5\rangle\}$
$\{\langle e_1, e_4\rangle, \langle e_1, e_{456}\rangle, \langle e_1, e_{56}\rangle\}$
$\{\langle e_2, e_5\rangle, \langle e_2, e_{46}\rangle, \langle e_2, e_{456}\rangle\}$
$\{\langle e_3, e_45\rangle, \langle e_3, e_{456}\rangle, \langle e_3, e_6\rangle\}$
$\{\langle e_{56}, e_1\rangle, \langle e_{56}, e_{23}\rangle, \langle e_{56}, e_{123}\rangle\}$
$\{\langle e_{456}, e_1\rangle, \langle e_{456}, e_{23}\rangle, \langle e_{456}, e_{123}\rangle\}$
$\{\langle e_{46}, e_2\rangle, \langle e_{46}, e_{123}\rangle, \langle e_{46}, e_{13}\rangle\}$
$\{\langle e_{45}e_{12}\rangle, \langle e_{45}, e_3\rangle, \langle e_{45}, e_{123}\rangle\}$
$\{\langle e_{456}, e_2\rangle, \langle e_{456}, e_{23}\rangle, \langle e_{456}, e_3\rangle\}$

Concerning the minimum distance, we have a three point dependency (for example $\{e_1, e_{12}, e_2\}$). The four remaining lines not containing such dependent points are $\{\langle e_{123}, e_{456}\rangle, \langle e_{46}, e_{13}\rangle, \langle e_3, e_6\rangle, \langle e_{23}, e_{56}\rangle\}$. Clearly, these four lines cannot be contained in a hyperplane since they span a 6-dimensional subspace. This is the minimum number of dependent points because our lines are skew so we don't have a two point dependency and thus $d(\mathcal{X}) = 3$.

3.3.2 From Quantum Sets of Lines to Stabiliser Codes

In Section 3.2 we found two different definitions of a quantum set of lines $\mathcal{X}$. First, we could check that any co-dimension 2 subspace is skew to an even number of the lines of $\mathcal{X}$. Alternatively, we could also find an expression of the lines in $\mathcal{X}$ as the sum modulo two of pencils of lines. In this section, both definitions are used to present quantum sets of lines and later an explicit form of their associated codes is given. Most of the examples are taken from [11].

Example 3.3.5. Consider all the lines going through a point p_0 in $PG(2,2)$. This is a quantum set of lines because a co-dimension 2 subspace in $PG(2,2)$ is a point and all points of $PG(2,2)$ are skew to an even number of the lines in $\mathcal{X}$. If we take the point p_0 that is in all the lines of $\mathcal{X}$, then p is skew to 0 lines of $\mathcal{X}$. Otherwise, any of the other point p in $PG(2,2)$ is exactly in 1 line of $\mathcal{X}$ because if it was in two different lines, then these two lines would intersect twice: in p and p_0. Therefore, since there are three lines in $\mathcal{X}$, p is skew to 2 lines in $\mathcal{X}$.
Another way of proving that these lines are a quantum set of lines is by expressing them as the sum modulo 2 of pencils of lines. In this case, the lines are a single pencil. Consider for example the quantum lines in Figure 3.12 and their labelling.

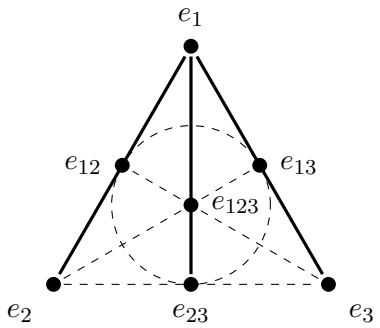


Figure 3.12: The quantum set of lines giving a $[[3,0,2]]_2$ code.

A possible generating matrix for the associated stabiliser code is

$$G = \begin{pmatrix} 1 & 1 & 1 \\ e & e & 0 \\ 0 & e & e \end{pmatrix}$$

For the minimum distance, since $k = 0$, we want to find the minimum over all hyperplanes (in $PG(2,2)$ a hyperplane is a line ℓ) of the number of lines of $\mathcal{X}$ not contained in ℓ. Take any line $\ell \in PG(2,2)$, if $\ell \in \mathcal{X}$ then the two other lines of $\mathcal{X}$ are not contained in ℓ. On the other hand, if $\ell \notin \mathcal{X}$ then all the three lines of $\mathcal{X}$ are not contained in ℓ. Since we have to take the minimum, $d(\mathcal{X}) = 2$. We have found a $\mathbf{[[3,0,2]]_2}$ quantum stabiliser code.

Example 3.3.6. Consider now $\mathcal{X}$ to be the 4 lines not going through a point p_0 in $PG(2,2)$.
$\mathcal{X}$ is a quantum set of lines since any point (co-dimension 2 subspace of $PG(2,2)$) is skew to an even number of the lines in $\mathcal{X}$. p_0 is skew to the 4 lines in $\mathcal{X}$ by definition, and thus it is skew to an even number of lines in $\mathcal{X}$. Any other point p in $PG(2,2)$ is in two lines of $\mathcal{X}$ since p is in 3 lines of $PG(2,2)$, one of which is the one joining p and p_0 so this line is not in $\mathcal{X}$. Thus, p is in two

lines of $\mathcal{X}$ and so it is skew to 2 other lines in $\mathcal{X}$.
For instance, consider $\mathcal{X}$ to be the lines in Figure 3.13.

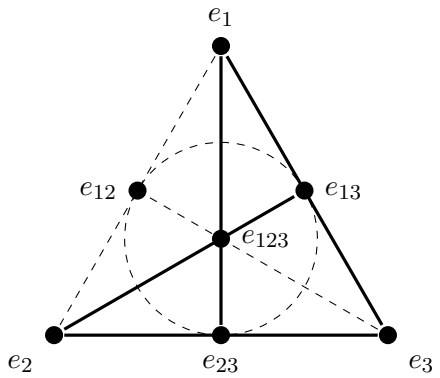


Figure 3.13: The quantum set of lines giving a $[[4, 1, 2]]$ code.

With this labelling, the quantum set of lines is

$$\mathcal{X} = \{\langle e_1, e_{23}\rangle, \langle e_{13}, e_2\rangle, \langle e_1, e_3\rangle, \langle e_2, e_3\rangle\}$$

We can also prove that it is a quantum set of lines by seeing that $\mathcal{X}$ is the result of adding together two pencils of lines modulo two:

$$\{\langle e_1, e_2, \rangle, \langle e_1, e_{23}\rangle, \langle e_1, e_3\rangle\} \text{ and } \{\langle e_2, e_1, \rangle, \langle e_2, e_{13}\rangle, \langle e_2, e_3\rangle\}.$$

A possible generating matrix G is

$$G = \begin{pmatrix} 1 & 1 & e & 0 \\ e & 0 & 1 & 1 \\ e & e & e & e \end{pmatrix}$$

The minimum distance $d(\mathcal{X})$ is 2 since we have a two point dependency (any two lines of $\mathcal{X}$ intersect in a point). This proves that this is a $\mathbf{[[4, 1, 2]]_2}$ code.

Example 3.3.7. A $[[6, 1, 3]]_2$ stabiliser code can be obtained from a spread of five lines $\mathcal{Z}$ in $PG(3, 2)$, which as we saw in Example 3.3.2 gives a $[[5, 0, 3]]_2$ code. Consider such spread $\mathcal{Z}$ and embed it in $PG(4, 2)$ so that the $PG(3, 2)$ is a hyperplane of $PG(4, 2)$. Adding a planar pencil of lines modulo two to $\mathcal{Z}$ will give a quantum set of lines $\mathcal{X}$ by Theorem 3.2.10. Let P be a pencil of lines containing one of the lines in $\mathcal{Z}$ but not contained in the $PG(3, 2)$. Now adding P modulo two gives us our quantum set of 6 lines in $PG(4, 2)$.

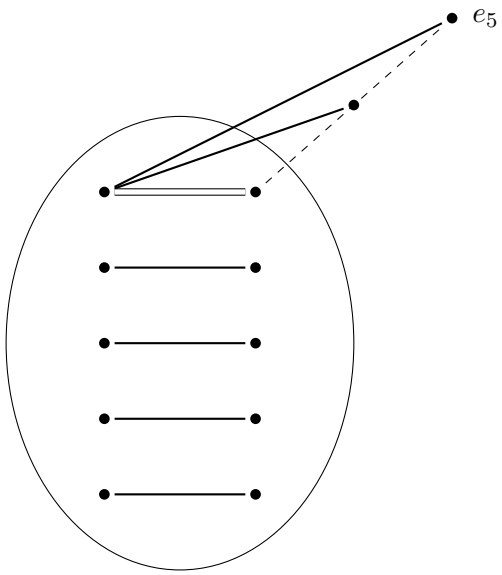


Figure 3.14: The quantum set of lines coming from a spread in $PG(3,2)$ embedded as a hyperplane in $PG(4,2)$ plus a planar pencil modulo two.

Regarding the minimum distance, the only two lines in $\mathcal{X}$ that intersect are the two lines in the pencil we have added. However, we should not take into account this dependent set of points of size two, since the lines in $\mathcal{X}$ that don't give this two point dependency are all contained in the hyperplane $PG(3,2)$ and thus, they correspond to words in C. On the other hand, we have a three point dependency since the rest of lines are all skew so $d(\mathcal{X}) = 3$.

Definition 3.3.8. A hyperoval in $PG(2,q)$ is a set of $q+2$ points no three of which are collinear.

Example 3.3.9. A $[[\mathbf{6,0,4}]]_\mathbf{2}$ stabiliser code can be obtained from a hyperoval O in $PG(2,4)$. Consider the hyperoval given by the following matrix, where each column represents a point of the hyperoval.

$$O = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & e & e^2 \\ 0 & 0 & 1 & 1 & e^2 & e \end{pmatrix}$$

The points in $PG(2,4)$ are given by elements in $\mathbb{F}_4^3$. Now we can perform a translation into $PG(5,2)$ whose points are elements of $\mathbb{F}_2^6$. Each point in the hyperoval will result in a line in $PG(5,2)$ resulting in a quantum set of lines.

Take any of the points of the hyperoval, that is any column of O, $p = (a,b,c)$. Consider $(a, ea, b, eb, c, ec)^t$ which is an element of $\mathbb{F}_4^6$. Now each of the elements in this form results in a line of $PG(5,2)$ by writing in terms of the basis $\{1, e\}$ as we have done before.

In this case, this process results in the stabiliser code given by

$$G = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 1 \\ e & 0 & 0 & e & e & e \\ 0 & 1 & 0 & 1 & e & e^2 \\ 0 & e & 0 & e & e^2 & 1 \\ 0 & 0 & 1 & 1 & e^2 & e \\ 0 & 0 & e & e & e & e^2 \end{pmatrix}$$

Whose lines are

$$\mathcal{X} = \{\langle e_1, e_2\rangle, \langle e_3, e_4\rangle, \langle e_5, e_6\rangle, \langle e_{135}, e_{246}\rangle, \langle e_{145}, e_{23456}\rangle, \langle e_{1346}, e_{2356}\rangle\}$$

It is easy to see that this is a stabiliser code by looking at the stabiliser matrix since all the rows are independent and pairwise commute.

Finally, we can check the minimum distance for this stabiliser code. The minimum distance $d(\mathcal{X})$ is given by the minimum over all hyperplanes h of $PG(5,2)$ of the number of lines in $\mathcal{X}$ not contained in h. One can check that any 4-dimensional subspace (hyperplane) of $PG(5,2)$ contains at most two of the lines in $\mathcal{X}$.

Chapter 4

A Nonadditive Quantum Code

So far we have only talked about linear, additive and stabiliser codes. It is true that these are the most common type of error-correcting codes. However in [18] a $((5, 6, 2))$ quantum non-stabiliser error-correcting code which is better than any other minimum distance two code (up until 1997) is presented.
With this code, 6 quantum states are encoded in 5 qubits with minimum distance two, which means that it can detect an error on any single qubit. The best stabiliser code of length 5 and minimum distance 2 is a $[[5, 2, 2]]$ code, that is, it encodes $2^k = 4$ quantum states in 5 qubits. In fact, for length 5, and minimum distance 2, the dimension $K = 6$ is extremal and cannot be improved. The nonadditive code presented is the union of six $[[5, 0, 3]]_2$ stabiliser codes.
Consider a $[[5, 0, 3]]_2$ code with stabiliser group $S_0 = \langle M_1, \ldots, M_5 \rangle$. As we have seen, the stabiliser code $Q(S_0)$ will have dimension 2^k, in this case $dim(Q(S_0)) = 1$.

Now consider S_1 to be $S_1 := \langle -M_1, M_2, \ldots, M_5 \rangle$ and its stabiliser code $Q(S_1)$ will also have dimension 1. When negating one of the generators of the stabiliser group, a new stabiliser code is obtained with the property that it is orthogonal to the original stabiliser code.

Lemma 4.0.1. *$Q(S_0)$ and $Q(S_1)$ are orthogonal.*

Proof. Let $v_0 \in Q(S_0)$ and $v_1 \in Q(S_1)$. We have

$$M_1 v_0 = v_0 \text{ and } -M_1 v_1 = v_1$$

The second equation implies

$$M_1 v_1 = -v_1$$

This means that v_0 and v_1 are both eigenvectors of M_1 with eigenvalues 1 and -1 respectively. Since any element in $\mathcal{P}_n$ is hermitian, all the elements in S_0 and S_1 are hermitian. Now, by Lemma 1.2.8 the eigenvectors of different eigenvalue of a hermitian matrix are orthogonal, so

$$\langle v_0, v_1 \rangle = 0$$

□

Similarly, consider the five stabiliser codes $Q(S_i), i = 1, \ldots, 5$ with stabiliser groups S_i generated by $M_1, \ldots, M_5$ with M_i negated. By the same reasoning as the previous lemma, we know that $Q(S_0)$ and $Q(S_i)$ are all mutually orthogonal. The new code Q is the union of $Q(S_i)$ $i = 0, \ldots, 5$.

Lemma 4.0.2. *The code*

$$Q = \bigoplus_{i=0}^{5} Q(S_i)$$

where S_0 is the stabiliser group of a $[[5, 0, 3]]_2$ stabiliser code and S_i is the stabiliser group S_0 but with M_i negated, is a quantum error-correcting code of dimension 6.

Proof. The six stabiliser codes considered are disjoint: without loss of generality, suppose $v \in Q(S_0) \cap Q(S_i)$ for some $i \in \{1, \ldots, 5\}$. Then $-M_i v = v$ because $v \in Q(S_i)$, but at the same time, since $v \in Q(S_0)$, we have $M_i v = v$ which implies $v = -v$, a contradiction with the definition of eigenvector. On the other hand, if we take any two vectors $v_i \in Q(S_i)$ and $v_j \in Q(S_j)$, v_i and v_j are linearly independent since they are eigenvectors of different eigenvalue (v_i is an eigenvector of eigenvalue -1 of M_i whereas v_j is an eigenvector of eigenvalue 1 of M_i).
Finally, any of the $Q(S_i)$ is a $[[5, 0, 3]]_2$ stabiliser code, and thus it has dimension 1. This means that

$$dim(Q) = \sum_{i=0}^{5} dim(Q(S_i)) = 6$$

□

An error-correcting code can correct any error of less weight than half the minimum weight of an undetectable error. In Section 2.2, we saw that for stabiliser codes, E was undetectable if and only if it was in $C(S) \setminus S$. We concluded that the minimum distance was the minimum non-zero weight of the errors in $C(S) \setminus S$. In this case, Q is not a stabiliser code, so the same statement doesn't hold. However, a similar reasoning can be used to find the minimum distance.
Let S_{ijk} denote the stabiliser group generated by M_i, M_j, M_k.

Proposition 4.0.3. *The minimum distance of Q is the minimum non-zero weight of errors in $C(S_{ijk}) \setminus S_{ijk}$ for all distinct $i, j, k \in \{1, \ldots, 5\}$.*

Proof. We will prove that if E is an undetectable error then $E \in C(S_{ijk}) \setminus S_{ijk}$ for any distinct $i, j, k \in \{1, \ldots, 5\}$ where $C(S_{ijk})$ denotes the centraliser of S_{ijk} and $S_{ijk} = \langle M_i, M_j, M_k \rangle$. Thus, Q can correct any error of weight less than the minimum weight of errors in $C(S_{ijk}) \setminus S_{ijk}$. To do this, suppose $E \notin C(S_{ijk}) \setminus S_{ijk}$ and we will see that E is detectable.
$E \notin C(S_{ijk}) \setminus S_{ijk}$ implies that E doesn't commute with at least three of the generating matrices of S_0. Since $E \in \mathcal{P}_n$, we have seen that E either commutes or anti-commutes with any element of $\mathcal{P}_n$. Thus, without loss of generality, suppose E anti-commutes with M_i, M_j, M_k.
The definition of a detectable error is that for any two elements $|x\rangle, |y\rangle \in Q$ such that $\langle x|y\rangle = 0$, $\langle x| E |y\rangle = 0$. It is enough to fix an orthogonal basis of $Q = \{x_0, \ldots, x_5\}$ and check that $\langle x_t| E |x_z\rangle = 0$ for any two elements of the basis.
In this case, we can take $x_i \in Q(S_i), i = \{0, \ldots, 5\}$ as a basis for Q and by Lemma 4.0.1, it is orthogonal.

Take any two elements of the basis x_i, x_k, we can choose an M_j such that $M_j \neq M_i, M_k$ and M_j anti-commutes with E

$$EM_j = -M_jE$$

Then we can write

$$\langle x_i| E |x_k\rangle = \langle x_i| E |M_j x_k\rangle = -\langle x_i| M_j E |x_k\rangle = -\langle x_i| E |x_k\rangle$$

since $x_i \in Q(S_i)$ and $x_k \in Q(S_k)$ (they are both an eigenvector of eigenvalue 1 of M_j).
This implies that $\langle x_i| E |x_k\rangle = 0$ and thus, E is detectable. □

In the case of the non-additive code which is the union of six $[[5,0,3]]_2$ stabiliser codes, one can check that if any three rows S_{ijk} of the stabiliser matrix are taken, the minimum weight of errors in $C(S_{ijk}) \setminus S_{ijk}$ is 2.

The geometrical idea behind this method is to fix a basis of points in $PG(n-k-1,2)$ such that projecting the quantum lines $\mathcal{X}$ from any two points of the basis onto $PG(n-k-3,2)$ gives a quantum set of lines $\mathcal{X}'$ of the same size as the original one. Computing the minimum distance of $\mathcal{X}'$ gives the minimum distance of the resulting nonadditive code.

Consider the Example of the $[[5,0,3]]_2$ code resulting in a $((5,6,2))$ code. In Example 3.3.2 we saw

$$G = \begin{pmatrix} 1 & e & 0 & 0 & e \\ e & 1 & e & 0 & 0 \\ 0 & e & 1 & e & 0 \\ 0 & 0 & e & 1 & e \\ e & 0 & 0 & e & 1 \end{pmatrix} \qquad G' = \begin{pmatrix} 1 & e & e^2 & e^2 & e \\ e & 1 & e & e^2 & e^2 \\ e^2 & e & 1 & e & e^2 \\ e^2 & e^2 & e & 1 & e \\ e & e^2 & e^2 & e & 1 \end{pmatrix}$$

Note that if we replace 0 by e^2, G' this is still the generating matrix of a $[[5,0,3]]_2$ code since the rows are independent and pairwise commute. Now the quantum set of lines associated to G' is

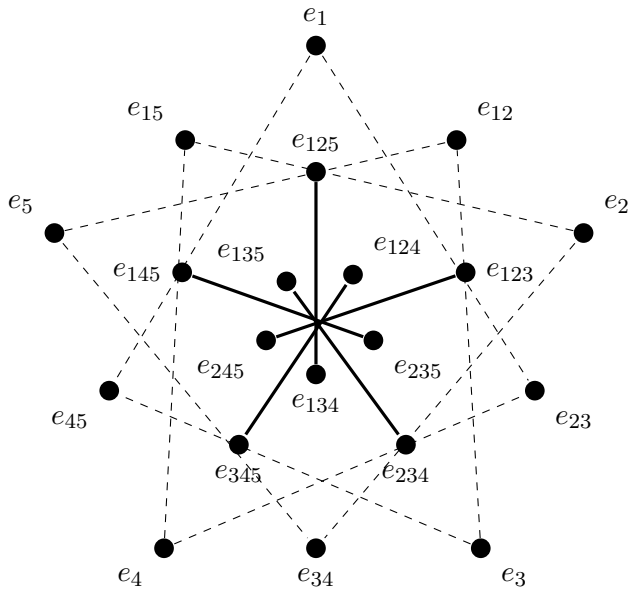


Figure 4.1: The quantum set of lines of the $[[5,0,3]]_2$ code associated to matrix G'.

We can fix the points $\{e_1, \ldots, e_5\}$ as a basis. If we project from any two points of this basis, we obtain 5 lines in $PG(2,2)$. Some lines might have multiplicity larger than one but this is not an issue. For example, consider the projection from points $\{e_4, e_5\}$. The resulting lines in $PG(2,2)$ are a quantum set of lines with minimum distance two (the minimum size of a dependent set of points is 2 since the lines intersect). The resulting quantum set of lines is represented in Figure 4.2.

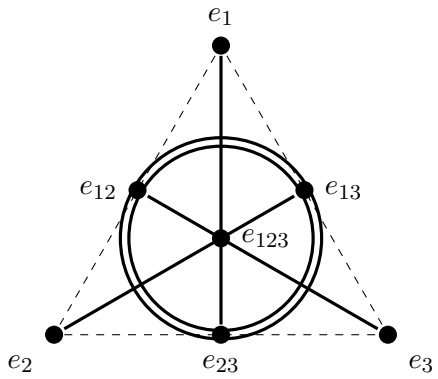


Figure 4.2: The quantum set of lines in Figure 4.1 projected from points $\{e_4, e_5\}$ onto $PG(2,2)$.

Example 4.0.4. Consider the $[[7,1,3]]_2$ code we saw in Example 3.3.4. As with the $[[5,0,3]]_2$ code we have just discussed, we can consider the union of eight $[[7,1,3]]_2$ codes

$$Q = \bigoplus_{i=0}^{7} Q(S_i)$$

where S_0 is the stabiliser group of a $[[7,1,3]]_2$ stabiliser code and S_i is S_0 but with M_i negated. The dimension of Q is

$$\dim(Q) = \sum_{i=0}^{7} \dim Q(S_i) = \sum_{i=0}^{7} 2^1 = 14$$

To assert the minimum distance of Q, consider the quantum set of lines associated to G'.

$$G' = \begin{pmatrix} e^2 & e^2 & e^2 & 1 & 1 & 1 & 1 \\ e^2 & 1 & 1 & e^2 & e^2 & 1 & 1 \\ 1 & e^2 & 1 & e^2 & 1 & e^2 & 1 \\ e^2 & e^2 & e^2 & e & e & e & e \\ e^2 & e & e & e^2 & e^2 & e & e \\ e & e^2 & e & e^2 & e & e^2 & e \end{pmatrix}$$

One can easily check that the rows of G' are independent and pairwise commute. In the same way as the $[[5,0,3]]_2$, we fix the basis $\{e_1, \ldots, e_6\}$ and project from any two points of the basis. For instance, suppose we project from points $\{e_5, e_6\}$. This will give 7 lines in $PG(3,2)$. Namely the projected lines will be

$$\ell_1 = \langle e_{1234}, e_3 \rangle \qquad \ell_5 = \langle e_{123}, e_{24} \rangle$$
$$\ell_2 = \langle e_{1234}, e_2 \rangle \qquad \ell_6 = \langle e_{123}, e_{34} \rangle$$
$$\ell_3 = \langle e_{1234}, e_{14} \rangle \qquad \ell_7 = \langle e_{123}, e_4 \rangle$$
$$\ell_4 = \langle e_{123}, e_{14} \rangle$$

There is a two point dependency since the 7 lines are not skew. For instance consider the two point dependency on e_{14}. The lines not containing e_{14} are $\ell_1, \ell_2, \ell_5, \ell_6$ and ℓ_7 and these lines span a 3-dimensional subspace so they cannot be contained in a hyperplane of $PG(3,2)$ which is a 2-dimensional subspace. This means that the minimum distance is 2. Thus, the union of these eight $[[7,1,3]]_2$ codes is a $((7,14,2))$ code.

Chapter 5

Conclusions and Further Work

The aim of this project was to study the geometry associated to quantum stabiliser codes used for quantum error detection and correction. More precisely, we have successfully achieved the following

1. *Understood the behaviour of quantum mechanics and the errors caused by quantum decoherence as well as the principles of quantum error-correcting theory.* In Chapter 1 we introduced the principles behind quantum mechanics and quantum error-correcting codes: superposition, qubit measurements and the no-cloning theorem. We also defined essential concepts of coding theory such as the Pauli Group –giving a basis for errors on sets of qubits– or the minimum distance of a code, which is the maximum weight of the errors that can be detected and corrected.

2. *Explored Quantum Stabiliser Codes and their parameters.* We have focused on stabiliser codes, which are the most common type of quantum error-correcting codes. In Chapter 2 we have defined them and given proofs of their parameters, that is, their dimension and minimum distance. We introduced the $[[9, 1, 3]]_2$ Shor Code which can correct arbitrary single qubit errors.

3. *Given a bijection between stabiliser codes and additive codes over $\mathbb{F}_4$.* We presented a map between elements of the Pauli group and $\mathbb{F}_4$ which allows us to consider errors of length n as elements of $\mathbb{F}_4^n$. This allowed us to treat stabiliser codes as additive codes over $\mathbb{F}_4$. An essential part of error-correcting codes is decoding; once the errors have been detected, one must be able to retrieve the original code. We presented a method for this in Section 2.5 as well as an example using a $[[5, 1, 3]]_2$ stabiliser code.

4. *Developed the geometrical translation of linear, additive and stabiliser codes as quantum sets of lines based on [11] and [3].* Using the bijection between stabiliser codes and additive codes we saw in Chapter 2, in Chapter 3 we found the conditions under which certain sets of lines in $PG(n - k - 1, 2)$ correspond to stabiliser codes with parameters $[[n, k, d]]$. We exploited the geometric aspect of codes to rewrite the proofs in [11] in a more intuitive way and explore their properties through visualization. Two equivalent definitions of a quantum set of lines were given which also allowed us to generate new codes from known quantum sets of lines by adding pencils of lines modulo two.

5. *Worked out the associated geometries to several stabiliser codes with various parameters.* We translated several stabiliser codes from Grassl's database [13] into quantum sets of lines and found their minimum distance using the theory we developed in Chapter 3. On the other hand, using the two geometrical definitions of a quantum set of lines $\mathcal{X}$ *–any co-dimension two subspace is skew to an even number of the lines in* $\mathcal{X}$ and $\mathcal{X}$ *is the sum modulo two of planar pencils of lines–* we have found some sets of lines that correspond to stabiliser codes and also asserted their minimum distance. Moreover we created visualisations for the geometries of these codes using the Tikz package.

6. *Explored nonadditive codes coming from the union of several stabiliser codes based on [18].* A nonadditive $((5, 6, 2))$ code is presented in [18] which is better than any stabiliser code of minimum distance 2. In Chapter 4 we explained how this nonadditive code is constructed as the union of six $[[5, 0, 3]]_2$ stabiliser codes and gave a generalisation of this method as well as a proof of their minimum distance.

Some aspects that were not tackled by this project and that remain as future work are:

· Use the method developed in [18] and Chapter 4 to find nonadditive codes with minimum distance 3 or more coming from the union of stabiliser codes.

· Use incidence geometry theory to prove the existence –or non-existence– of some stabiliser codes with certain parameters that remain unknown, for example $[[14, 3, 5]]_2$, $[[16, 5, 5]]_2$ or $[[24, 0, 10]]_2$. The non-existence of a $[[13, 5, 4]]_2$ was proved in this way in [5] in 2009.

Bibliography

[1] Scott Aaronson. The limits of quantum. *Scientific American*, 298(3):62–69, 2008.

[2] Simeon Ball. *Finite geometry and combinatorial applications*, volume 82. Cambridge University Press, 2015.

[3] Simeon Ball and Felix Huber. *Quantum error-correcting codes and their geometries*. Preprint, 2020.

[4] Albrecht Beutelspacher and Ute Rosenbaum. *Projective Geometry: From Foundations to Applications*. Cambridge University Press, 1998.

[5] J. Bierbrauer, S. Marcugini, and F. Pambianco. The non-existence of a [[13,5,4]] quantum stabilizer code, 2009.

[6] R. Calderbank, E. M. Rains, P. Shor, and N. J. A. Sloane. Quantum error correction via codes over gf(4). 04 1997.

[7] Robert Calderbank and Peter Shor. Good quantum error-correcting codes exist. *Physical Review A*, 54, 1998.

[8] Peter J. Cameron. *Combinatorics: Topics, Techniques, Algorithms*. Cambridge University Press, 1994.

[9] P.C.W. Davies and D.S. Betts. *Quantum Mechanics, Second edition*. Physics and its applications. Taylor & Francis, 1994.

[10] Giorgio Faina, Massimo Giulietti, Stefano Marcugini, and Fernanda Pambianco. The geometry of quantum codes. *Innovations in Incidence Geometry [electronic only]*, 6/7, 01 2007.

[11] D. Glynn, T. A. Gulliver, J. G. Maks, and M. Gupta. *The geometry of additive quantum codes*. 01 2004.

[12] Daniel Gottesman. Stabilizer codes and quantum error correction, 1997.

[13] Markus Grassl. Bounds on the minimum distance of linear codes and quantum codes. Online available at `http://www.codetables.de`, 2007. Accessed on 2020-04-02.

[14] Gareth A. Jones and J. Mary Jones. Information and coding theory. 01 2000.

[15] Andy Matuschak and Michael A. Nielsen. Quantum computing for the very curious. Available at `https://quantum.country/qcvc`, 2019.

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

[17] Alexander Pott. *Finite Geometry and Character Theory*, volume 1601. 01 1995.

[18] E. M. Rains, R. H. Hardin, P. W. Shor, and N. J. A. Sloane. A nonadditive quantum code. *Physical Review Letters*, 79(5):953–954, Aug 1997.

[19] Andrew Steane. Simple quantum error correcting codes. *Physical review. A*, 54:4741–4751, 01 1997.

[20] Jacobus van Lint. *Introduction to Coding Theory*. Graduate Texts in Mathematics. Springer Berlin Heidelberg, 1998.