School of Computer Science, Physics and Mathematics

Master Thesis

## Classification of perfect codes and minimal distances in the Lee metric

Naveed Ahmed<br>1982-06-20<br>Waqas Ahmed<br>1982-09-13<br>Subject: Mathematics<br>Level: Advance<br>Course code: 5MA12E


#### Abstract

Perfect codes and minimal distance of a code have great importance in the study of theory of codes. The perfect codes are classified generally and in particular for the Lee metric. However, there are very few perfect codes in the Lee metric. The Lee metric has nice properties because of its definition over the ring of integers residue modulo $q$. It is conjectured that there are no perfect codes in this metric for $q>3$, where $q$ is a prime number.

The minimal distance comes into play when it comes to detection and correction of error patterns in a code. A few bounds on the number of codewords and minimal distance of a code are discussed. Some examples for the codes are constructed and their minimal distance is calculated. The bounds are illustrated with the help of the results obtained.


Key-words: Hamming metric; Lee metric; Perfect codes; Minimal distance

## Acknowledgments

This thesis is written as a part of the Degree of Master of Science in Mathematics at Linnaeus University, Sweden. It is written at the School of Computer Science, Physics and Mathematics.

We would like to thank our supervisor Per-Anders Svensson who has been very supporting throughout the whole period. We are also obliged to him for giving us chance to work under his supervision. We would also like to thank our Program Manager Marcus Nilsson who has been very helpful and very cooperative within all matters, and also all the respectable faculty members of the department. We would also like to thank our family members and friends for their support and encouragement during the whole period.

## Contents

1 Introduction ..... 1
2 Preliminaries ..... 2
2.1 Groups ..... 2
2.2 Subgroups ..... 3
2.3 Cyclic groups ..... 4
2.4 Lagrange's theorem ..... 5
2.5 Rings and fields ..... 6
2.6 Vector spaces ..... 9
2.7 Subspaces ..... 11
2.8 Dimension and basis ..... 11
3 Introduction to coding theory ..... 13
3.1 Basic assumptions ..... 13
3.2 Correcting and detecting error patterns ..... 16
3.3 Maximum likelihood decoding ..... 18
4 Finite fields ..... 19
4.1 Introduction ..... 19
4.2 Prime fields ..... 19
4.3 Polynomials ..... 20
4.4 Primitive elements of finite fields ..... 22
4.5 Splitting field ..... 22
4.6 Structure of finite fields ..... 24
5 Linear Codes ..... 26
5.1 Linear codes ..... 26
5.2 Bases for a Code Generated by $S$ and its Dual Code ..... 28
5.3 Generating matrices ..... 29
5.4 Parity- check matrices ..... 30
5.5 Equivalent codes ..... 32
5.6 Distance of a linear code ..... 33
6 Perfect codes ..... 34
6.1 Hamming codes ..... 36
6.2 Golay codes ..... 37
6.3 Ternary Golay code ..... 40
6.4 Reed- Solomon codes (a non- perfect code) ..... 41
7 Perfect codes in Lee metric ..... 43
7.1 Sphere-packing bound in the Lee metric ..... 45
7.2 Berlekamp codes ..... 47
8 Minimal distance ..... 48
8.1 Singleton bound ..... 48
8.2 Gibert- Varshamov bound ..... 48
9 Minimal distance in Lee Metric ..... 52
9.1 Gilbert-Varshamov bound in the Lee metric ..... 52
9.2 Plotkin's low-rate average distance bound ..... 54
10 Conclusion ..... 56
References ..... 57

## 1 Introduction

Coding theory is relatively a new field. It was first studied in the 1940s. When the data is transmitted through a channel, there is a possibility of error occurance. These errors may be caused by the noise produced by the channel such as bad weather, poor telephone lines, etc. The aim of coding theory is to study the methods to detect and correct these errors.

There are several types of metrics that are used in construction of codes. We only consider two of them, namely the Hamming metric and the Lee metric. The main difference in the structure of these metrics is the way we calculate the distance between any two codewords. While we are using the Hamming metric, an error value (non zero) is not given the importance but error means a change of an entry in the transmitted codeword. The same weight is given to all (non zero) error values even in the $q$-ary symmetric channel. The Lee metric is defined over the ring of integers residue modulo $q$. In this metric change of an entry by $\pm 1$ is considered as one error. This type of error is often found in channels using phase shift key modulation. This is the reason that the Lee metric is more suitable for such channels. The Hamming metric and the Lee metric are the same if they are taken over $G F(2)$ or $G F(3)$, where $G F$ is so called Galois field.

A perfect code means a code satisfying the Hamming bound (discussed in chapter 6) with equality. There exists perfect codes in both the Hamming and the Lee metric over $G F(2)$ and $G F(3)$. The Hamming codes, the Golay codes and the extended Golay codes are the examples of perfect codes. But according to [1] and [4] it is conjectured that there are no more perfect codes in the Lee metric for $G F(q)$ where $q>3$. The minimal distance of code $C$ is the smallest possible distance between any two different codewords of $C$. The minimal distance gives us the idea about the ability of a code to detect and possibly correct errors.

To deal with the above mentioned things, one needs to know quiet a few mathematical concepts of algebraic structures which are discussed in chapter 2. In chapter 3, an introduction to the coding theory is given. We discuss basic things related to the coding theory including nearest neighbour decoding principle, detecting and correcting error patterns and parity check code, etc. Finite fields have great use in coding theory. Chapter 4 is about finite fields. Theorems about structure of finite fields are given with basic definitions. An important class of codes, called linear codes, are studied in chapter 5. The structure of generating and parity check matrices is discussed. The generating matrix is used to generate the Hamming codes and the Golay codes, for instance. To correct the error patterns, parity check matrices play an important role.

The last four chapters are the core of the whole work. We classify the perfect codes generally in chapter 6. They include the Hamming codes and the Golay codes. As an example of a non perfect code, we discuss Reed-Solomon code. Chapter 8 discusses bounds on the number of codewords and the minimal distance of a code. The Hamming bound (Sphere packing bound), Singleton bound and the Gelbert-Varshamov bound is given with examples. Finally, chapter 7 and 9 are focused on the Lee metric. The perfect codes in the Lee metric are discussed in chapter 7. The minimal distance in the Lee metric is given in the last chapter. In this chapter few bounds with proofs are discussed.

The aim of this thesis is to classify perfect codes and finding the minimal distance of a code in general (the Hamming metric) and particularly in the Lee metric. The classification of perfect codes and finding minimal distance of a code in the Lee metric is the main task of this work.

## 2 Preliminaries

In this chapter we are going to define some basic and special type of algebraic structures, like groups, rings and fields. These algebraic structures have great importance in coding theory. In the next section we will see the uses and applications of these algebraic structures. The material in the chapter is taken from [2], [8] and [5].

### 2.1 Groups

In this section of groups, we will define groups, subgroups, also some properties of groups. We will also discuss some important theorems on groups.

## Definition 2.1. Groups

Let $G$ be a non empty set and $*$ be a binary operation on $G$. Then $G$ with binary operation * is called group if it satisfies the following properties:
i. $*$ is associative in $G$,i.e. for all $a, b, c \in G$, we have

$$
(a * b) * c=a *(b * c) .
$$

ii. $G$ contains an identity element with respect to $*$. For any $a \in G$ there exist $e \in G$ such that

$$
a * e=e * a=a .
$$

iii. Every element in $G$ has it's inverse in $G$, with respect to $*$. For any $a \in G$ there exist $a^{\prime} \in G$ such that

$$
a * a^{\prime}=a^{\prime} * a=e .
$$

If all above properties are fulfilled then $G$ is called group under the binary operations * and it is written as $(G, *)$.

## Definition 2.2. Order of a group

If $G$ is group, then the number of elements in $G$, if $G$ is finite, is called the order of the group and it is denoted as $|G|$.

## Definition 2.3. Abelian Group

If the binary operation $*$ is commutative, then $(G, *)$ is called an abelian group.
Now we consider an example to illustrate the definition of the group in a simple way.
Example 2.1. Let us consider the set $\mathbb{Q}^{+}$and the binary operation $*$ defined by $a * b=\frac{a b}{3}$. Then
i. For all $a, b, c \in \mathbb{Q}^{+}$

$$
(a * b) * c=a *(b * c)=\frac{a b c}{9}
$$

which shows that $*$ is associative
ii. 3 is identity element.
iii. The inverse of any $a \in \mathbb{Q}^{+}$is $a^{\prime}=\frac{3}{a}$

Hence $\mathbb{Q}^{+}$is group with respect to the binary operation $*$.
$(\mathbb{Z},+),(\mathbb{Q},+),(\mathbb{R},+)$ are all examples of groups. All of them are abelian groups.

### 2.1.1 Properties of groups

Now we discuss some properties of groups and state some theorems about groups. In the first theorem we will establish cancellation laws and to prove this theorem, the thing which we know is the definition of the group. So we will use this definition to prove our first theorem.

Theorem 2.1. If $G$ is a group with binary operation *, the left and right cancellation laws hold in $G$, that is for all $a, b, c \in G$

$$
a * b=a * c \Longrightarrow b=c,
$$

and

$$
b * a=c * a \Longrightarrow b=c .
$$

Proof. Suppose $a * b=a * c$. Then by iii, there exists $a^{\prime}$, and

$$
a^{\prime} *(a * b)=a^{\prime} *(a * c) .
$$

By using associative law,

$$
\left(a^{\prime} * a\right) * b=\left(a^{\prime} * a\right) * c .
$$

By using iii, we get

$$
e * b=e * c .
$$

By the definition of $e$ in ii,

$$
b=c .
$$

Similarly we can do for $b * a=c * a$.
Theorem 2.2. If $G$ is a group with binary operation $*$, and if a and $b$ are any elements of $G$, then the linear equations $a * x=b$ and $y * a=b$ have unique solutions $x$ and $y$ in $G$.

The next theorem is about the uniqueness of inverse and identity element in a group.
Theorem 2.3. If $G$ is group and $*$ is a binary operation, then the identity element in $G$ is unique such that

$$
e * a=a * e=a \text {, }
$$

for all $a \in G$.
Theorem 2.4. If $G$ is group and $*$ is a binary operation, then the inverse of each element in $G$ is unique such that

$$
a^{\prime} * a=a * a^{\prime}=e,
$$

for all $a \in G$.

### 2.2 Subgroups

From the theory of groups, sometimes it is noticed that smaller groups are contained within larger groups. For example, the group $\mathbb{Z}$ under addition is contained within $(\mathbb{Q},+)$, which in turn contained in $(\mathbb{R},+)$. In this case of a group contained in the other group, means that the smaller group satisfy all the properties of the group under the same binary operation defined on the larger one. For $(\mathbb{Z},+)$, which is contained in $(\mathbb{R},+)$, it is very important to notice that the operation + on integers $a$ and $b$ as elements of $(\mathbb{Z},+)$ produces the same element $a+b$ as would result if you were to think of $a$ and $b$ as elements in $(\mathbb{R},+)$.

Definition 2.4. Let $G$ be a group with the binary operation $*$ and $H$ be subset of $G$. If $H$ with the induced operation from $G$ is itself a group, then $H$ is called a subgroup of $G$. It is denoted as $H \leq G$.

## Examples:

The group $\mathbb{Z}$ under addition is the subgroup of $\mathbb{Q}$ and $\mathbb{Q}$ is a subgroup of $\mathbb{R}$, further $\mathbb{R}$ is subgroup of $\mathbb{C}$. All these example are about the groups under addition. It is written as

$$
\mathbb{Z} \leq \mathbb{Q} \leq \mathbb{R} \leq \mathbb{C} .
$$

### 2.3 Cyclic groups

The goal of this section is to define cyclic groups, and present some properties of cyclic groups. We have defined the order of a finite group before. Here the order of an element will be defined.

Definition 2.5. A group $G$ is called cyclic if its every element is generated by some element $a \in G$, i.e for some $a \in G$ if $G=\langle a\rangle=\left\{a^{n} \mid n \in \mathbb{Z}\right\}$. The element $a$ is called generator of the cyclic group $G$.

Example 2.2. Consider the group $\mathbb{Z}_{5}^{*}=\{1,2,3,4\}$. We can see that

$$
3^{1}=3,3^{2}=4,3^{3}=2 \text { and } 3^{4}=1
$$

This shows that group $\mathbb{Z}_{5}^{*}$ is cyclic and 3 is a generator. It can be written as $\langle 3\rangle=\mathbb{Z}_{5}^{*}$.

## Definition 2.6. Order of an Element

Let $a$ be an element of a cyclic group $G$. If the cyclic subgroup generated by this element, i.e. $\langle a\rangle$ is a finite subgroup of $G$ then the order of $a$, denoted by $o(a)$, is the order $|\langle a\rangle|$ of this cyclic subgroup.

### 2.3.1 Properties of cyclic groups

Cyclic groups have some properties, here we will only state those properties where the proofs of these properties can be found in Fraleigh [2].

Theorem 2.5. If $G$ is a group and if it is cyclic, then it is abelian.

Theorem 2.6. Let $G$ be a cyclic group. Then a subgroup of $G$ is cyclic.
Theorem 2.7. For a cyclic group $G=\langle a\rangle$ of order $n$

$$
o\left(a^{k}\right)=\frac{n}{\operatorname{gcd}(n, k)}
$$

for all integers $k$.
Theorem 2.8. If $G$ is a cyclic group of $n$ elements and $d$ is a positive divisor of $n$, then there is a uniquely determined subgroup of order $d$ in $G$.

### 2.4 Lagrange's theorem

In this section a very important theorem of Lagrange, which is of great importance and have vast application even in coding theory, will be stated. Before going to the proof of the theorem we will define the concept of cosets and some important lemmas. These will help us to prove this theorem.

## Definition 2.7. Cosets

Let $G$ be a group and $H$ be subgroup of $G$ then for each $a \in G$ the set $a H=\{a h \mid h \in H\}$ is called the left coset of $H$ that contains $a$. Similarly the set $H a=\{h a \mid h \in H\}$ is called the right coset of $H$ containing $a$.

Before going to the Lagrange theorem, we need to state a lemma which states that the number of elements in any left coset of a subgroup $H$ are the same as the number of elements in $H$. The lemma is stated as

Lemma 2.9. Let $G$ be a group of finite order and $H$ be a subgroup of $G$. then for each $a \in G$,

$$
|a H|=|H| .
$$

Theorem 2.10. Let $H$ be a subgroup of $G$. Let the relation $\sim_{L}$ be defined on $G$ by

$$
a \sim_{L} b \quad \text { if and only if } a^{-1} b \in H .
$$

Let $\sim_{R}$ be defined by

$$
a \sim_{R} b \quad \text { if and only if } \quad a b^{-1} \in H .
$$

Then $\sim_{L}$ and $\sim_{R}$ are both equivalence relation on $G$, and the corresponding equivalence classes are exactly the left and right cosets, respectively.

Remark: The equivalence relation $\sim_{L}$ in Theorem 2.10 defines a partition of $G$.

## Theorem 2.11. (Langrange)

Let $G$ be a group of finite order and $H$ be a subgroup of $G$. Then the order of the subgroup divides the order of the group, i.e. $|H|$ divides $|G|$.
Proof. Let $G$ be a group of order $n$ and $H$ be a subgroup of order $m$. Let $a_{1} H, a_{2} H, \cdots a_{r} H$ be the different left cosets of $H$, We may write $G$ as

$$
G=a_{1} H \cup a_{2} H \cup \cdots \cup a_{r} H .
$$

As the left cosets according to the Theorem 2.10 form a partition in $G$, therefore the left cosets are pairwise disjoint, so we have

$$
|G|=\left|a_{1} H\right|+\left|a_{2} H\right|+\cdots+\left|a_{r} H\right| .
$$

By using the above lemma, we get

$$
|G|=|H|+|H|+\cdots+|H| .
$$

This implies

$$
|G|=r \cdot|H| \Longleftrightarrow n=r \cdot m,
$$

which gives the proof.
Corollary. Let $G$ be a finite group. Then the order of an element in the group divides the order of the group, i.e o(a) divides $|G|$ for all $a \in G$.

Proof. As we know that order of an element of a cyclic group is equal to the order of the cyclic group generated by that element. Therefore by Langrange's theorem order of that element divides the order of the group.

### 2.5 Rings and fields

From the study of groups, we observe that on groups a single binary operation is defined and this binary operation satisfies all the properties of group which are stated in the definition of group.

In this section we are going to define an algebraic structure on which two binary operations namely addition and multiplication are defined. This algebraic structure is called a ring.

## Definition 2.8. Ring

Let $R$ be a set, and let + and $\cdot$ be two binary operation on $R$. A ring is a triple ( $R,+, \cdot)$, such that
i. $(R,+)$ is an abelian group.
ii. multiplication is associative in $R$, i.e. for all $a, b, c \in R$

$$
a \cdot(b \cdot c)=(a \cdot b) \cdot c
$$

iii. • is distributive over +. This means that the left distributive law

$$
a \cdot(b+c)=a \cdot b+a \cdot c,
$$

and the right distributive law

$$
(b+c) \cdot a=b \cdot a+c \cdot a
$$

both hold.
Example 2.3. The set of integers $\mathbb{Z}$ under the ordinary addition and multiplication of numbers is a ring. Also the set of rational numbers $(\mathbb{Q})$ and set of real numbers $(\mathbb{R})$ are rings under the same binary operations as $\mathbb{Z}$.

Example 2.4. Let $F$ be the set of all functions $f: \mathbb{R} \rightarrow \mathbb{R}$. Then $F$ under the binary operation + , where + is usual function addition

$$
(f+g)(x)=f(x)+g(x),
$$

is an abelian group. The multiplication on $F$ is defined as

$$
(f g)(x)=f(x) g(x)
$$

Then $F$ under the binary operations usual addition and multiplication of functions is a ring.

### 2.5.1 Properties of rings

Theorem 2.12. If $R$ is a ring with additive identity 0 , then for all $a, b \in R$ we have
i. $0 a=a 0=0$,
ii. $a(-b)=(-a) b=-(a b)$,
iii. $(-a)(-b)=a b$.

Proof. i. We first show that $a 0=0$

$$
\begin{aligned}
a 0+a 0 & =a(0+0) \quad \text { left distributive law } \\
& =a 0=a 0+0 \quad 0 \text { is additive identity }
\end{aligned}
$$

As we know that $R$ with respect to addition is group, therefore cancellation laws holds in $R$. Applying the left cancellation law on the above equation yields

$$
a 0=0,
$$

as required. Similarly we can show that $0 a=0$.
ii. We first show that $a(-b)=-a b$, by definition $-a b$ is the element when added to $a b$ gives 0 . By the left distributive law yields,

$$
\begin{aligned}
a(-b)+a b & =a(-b+b) \\
& =a 0 \quad-b \text { is inverse of } b \\
& =0 \quad \text { from }(i)
\end{aligned}
$$

which shows that $a(-b)$ is the inverse of $a b$. But we know that $-a b$ is also the inverse of $a b$, we also know that inverse is always unique in a group. Therefore $a(-b)=-a b$. Similarly we can show that $(-a) b=-a b$.
iii. Now we prove that $(-a)(-b)=a b$. Consider

$$
(-a)(-b)-a b=(-a)(-b)+(-a) b \quad \text { due to (ii) }
$$

By using left distributive law the above equation becomes

$$
\begin{aligned}
(-a)(-b)-a b & =(-a)(-b+b) \\
& =(-a) 0 \quad-b \text { is inverse of } b \\
& =0
\end{aligned}
$$

which shows that $(-a)(-b)$ is an inverse of $-a b$. But we know that $a b$ is also inverse of $-a b$. Since inverse is always unique therefore $(-a)(-b)=a b$.

From the rings which we have studied so far, we observe that the ring $R$ contains identity element w.r.t addition but it does not need to contain identity element w.r.t multiplication. So by adding more things in the definition of ring, we define a new ring with a richer structure.

## Definition 2.9. Ring with unity

If a ring $R$ contains an identity element $1 \neq 0$ w.r.t multiplication, then this element is called a unity. The ring is then called a ring with unity.

## Definition 2.10. Commutative ring

A ring $R$ is said to be commutative, if for all $a, b \in R$

$$
a b=b a .
$$

### 2.5.2 Polynomial rings

Now we are going to define another type of rings, that is rings of polynomials. A polynomial ring is a set $R[x]$ of all polynomials in one or more variables with coefficients in another ring $R$ under the usual operations of polynomial addition and multiplication, these operations will be defined later. Before going to the definition of polynomial rings we need to define polynomial.

## Definition 2.11. Polynomial

Let $R$ be a ring. Then by a polynomial $f(x)$ with coefficients in the ring $R$ we mean an infinite formal sum

$$
\sum_{i=0}^{\infty} a_{i} x^{i}=a_{0}+a_{1} x^{1}+\cdots+a_{n} x^{n}+\cdots
$$

where the coefficients $a_{i}$ for all $i$ are elements of the ring and where $a_{i}=0$ for all but a finite number of values of $i$. The set of such polynomials is denoted by $R[x]$.

The degree of a polynomial $f(x)$ in $R[x]$ is the largest $i$ for which $a_{i} \neq 0$ and the degree of a polynomial $f(x)$ is denoted by $\operatorname{deg} f(x)$. Usually a polynomial of degree $n$ is written as

$$
f(x)=a_{0}+a_{1} x^{1}+\cdots+a_{n} x^{n} .
$$

Now we define the operations of polynomial addition and multiplication. If we have two polynomials $f(x)=\sum_{i=0}^{\infty} a_{i} x^{i}$ and $g(x)=\sum_{i=0}^{\infty} b_{i} x^{i}$, the sum of these two polynomials is defined as:

$$
\begin{equation*}
f(x)+g(x)=\sum_{i=0}^{\infty} c_{i} x^{i} \quad \text { where } c_{i}=a_{i}+b_{i} \quad \forall i \tag{2.1}
\end{equation*}
$$

and the product is defined as:

$$
\begin{equation*}
f(x) g(x)=\sum_{i=0}^{\infty} d_{i} x^{i} \quad \text { where } d_{i}=\sum_{k=0}^{i} a_{k} b_{i-k} \quad \forall i . \tag{2.2}
\end{equation*}
$$

Observe that both $c_{i}$ and $d_{i}$ are 0 for all but a finite number of values of $i$. Also $\sum_{k=0}^{i} a_{k} b_{i-k}$ need not equal $\sum_{k=0}^{i} b_{k} a_{i-k}$ if $R$ is not commutative. With the definition (2.1) and (2.2) we have the following theorem.

Theorem 2.13. The set of all polynomials $R[x]$ over a ring $R$ with coefficients in $R$ is a ring with respect to the operation defined in (2.1) and (2.2). If $R$ is ring with unity, then the unity of $R$ is the unity in $R[x]$. Furthermore if $R$ is commutative, then $R[x]$ is also commutative.

## Definition 2.12. Polynomial ring

The ring $R[x]$ defined in the above theorem is called a polynomial ring over a field $R$.

## Definition 2.13. Division algorithm

Let $F$ be a field. Suppose $f(x)$ and $g(x)$ are two polynomials in $F[x]$, where $g(x) \neq 0$. there are uniquely determined polynomials $q(x), r(x) \in F[x]$ such that

$$
f(x)=g(x) q(x)+r(x),
$$

where $r(x)=0$ or the degree of $r(x)$ is less than the degree of $g(x)$.

## Theorem 2.14. Factor theorem

An element of a field $F$ is a zero of $f(x) \in F[x]$ iff $(x-a) \mid f(x)$, where $a \in F$.

## Definition 2.14. Subring

Let $R$ be a ring. A subset $S$ of $R$ is called a subring of $R$, if $S$ is a ring with respect to the restriction of the addition and the multiplication on $R$ to $S$. It can be written as $S \leq R$.

## Definition 2.15. Ideal

Let $R$ be a ring and $I$ a subring of $R$. We say that $I$ is an ideal of $R$, is both ar and $r a$ belong to $I$, for all $a \in I$ and $r \in R$.

## Definition 2.16. Principal Ideals

Let $I$ be an ideal in a commutative ring with unity, then $I$ is called a principal ideal if there is an $a \in R$ such that

$$
I=\langle a\rangle=\{r a \mid r \in R\} .
$$

## Definition 2.17. Maximal Ideal

Let $R$ be a ring. An ideal $M$ of $R$ is said to be maximal, if $M \neq R$, and if there are no ideals $I$ such that $M \subset I \subset R$.

### 2.5.3 Fields

We have almost what is required for defining the field, but still we need to define one more thing and that is unit. In the next definition we define unit and then we will be able to define field.

## Definition 2.18. Unit

Let $R$ be a ring with unity. If an element $a$ of $R$ has an inverse with respect to multiplication, then the element $a$ is called a unit. Mathematically it is written as there exist $b$ in $R$ such that

$$
a b=b a=1 .
$$

## Definition 2.19. Field

A commutative ring with unity, in which all non zero elements in the ring are units is called field.

Example 2.5. The rings $\mathbb{Q}, \mathbb{R}, \mathbb{C}$ are fields but the ring $\mathbb{Z}$ is not field.
If we consider the ring $\mathbb{Z}_{n}$ of residue classes modulo $n$, we have the following theorem:
Theorem 2.15. The ring $\mathbb{Z}_{n}$ is a field, if and only if $n$ is a prime.

### 2.6 Vector spaces

In this section, one of the important fundamentals of linear algebra will be discussed. This is called vector spaces and this is of great importance and have many applications. We can see its importance from the fact that even linear algebra is generally considered as the theory of vector spaces. Details can be found in [5] or [8].

Definition 2.20. Vector Spaces Let $F$ be a field and $V$ be a non empty set. We assume that $V$ is closed w.r.t addition, i.e. for any $u, v \in V$ implies that $u+v \in V$. We also assume that for any $a \in F$ and $u \in V, a u \in V$. The set $V$ is called vector space over $F$ is the following conditions hold:
i. $V$ is an abelian group with respect to addition.
ii. For any $a \in F$ and $u, v \in V$,

$$
a(u+v)=a u+a v .
$$

iii. For any $a, b \in F$ and $v \in V$,

$$
(a+b) v=a v+b v
$$

iv. For any $a, b \in F$ and $v \in V$,

$$
a(b v)=(a b) v .
$$

v. There exist identity element in $F$ with respect to multiplication such that,

$$
1 v=v \quad \text { for any } v \in V
$$

Here the elements of field are called scalars and the elements of $V$ are called vectors. From (i) it follows that the identity element is unique and every element of $V$ has its inverse in $V$ with respect to addition of vectors. The multiplication of a vector by a scalar is called scalar multiplication. The left side of (iv) reveals that it is scalar multiplication and in the parentheses on right hand side is the multiplication of field elements. The conditions (ii) and (iii) shows that scalar multiplication is distributive over addition of vectors and scalars, as in the condition (ii) on the left hand side in parentheses is addition of vectors and in (iii) is addition of scalars.

Example 2.6. Let $F$ be a field and $n$ be any positive integer. Consider the set of all ordered $n$-tuples denoted by $F^{(n)}$ over field $F$, that is

$$
F^{(n)}=\left\{\left(a_{1}, a_{2}, a_{3}, \cdots, a_{n}\right) \mid a_{i} \in F, i=1,2,3, \cdots, n\right\} .
$$

The vector addition and scalar multiplication of a vector by an element in $F$ is introduced as follows:

$$
\begin{gathered}
\left(a_{1}, a_{2}, a_{3}, \cdots, a_{n}\right)+\left(b_{1}, b_{2}, b_{3}, \cdots, b_{n}\right)=\left(a_{1}+b_{1}, a_{2}+b_{2}, a_{3}+b_{3}, \cdots, a_{n}+b_{n}\right), \\
c\left(a_{1}, a_{2}, a_{3}, \cdots, a_{n}\right)=\left(c a_{1}, c a_{2}, c a_{3}, \cdots, c a_{n}\right), \quad c \in F .
\end{gathered}
$$

Then $F^{(n)}$ is a finite dimensional vector space over $F$ with respect to addition and scalar multiplication defined above.
Example 2.7. Let $F[x]$, the set of all polynomials with coefficients from $F$. Also let

$$
p(x)=a_{0}+a_{1} x+\cdots+a_{n} x^{n} \in F[x],
$$

and

$$
g(x)=b_{0}+b_{1} x+\cdots+b_{n} x^{n} \in F[x],
$$

The usual addition of polynomials and the scalar multiplication is defined as:

$$
p(x)+g(x)=c_{0}+c_{1} x+\cdots+c_{n} x^{n} \in F[x],
$$

where $c_{n}=a_{n}+b_{n}$. And for $c \in F$ :

$$
c p(x)=c a_{0}+c a_{1} x+\cdots+c a_{n} x^{n} .
$$

Then $F[x]$ is vector space over $F$.
Theorem 2.16. Let $V$ be a vector space over a field $F$. Then the following conditions hold:
i. For all $v \in V, 0 v=0$.
ii. For all $c \in F, c 0=0$.
iii. $(-1) v=-v$, for all $v$ in $V$.
iv. for all $c$ belonging to $F$ and $v$ belonging to $V, c v=0$ implies that either $c=0$ or $v=0$.

### 2.7 Subspaces

## Definition 2.21. Subspace

A subset of a vector space over a field $F$ is a subspace of the vector space, if the subset itself is a vector space under the operation induced from the vector space.

The next theorem provides a simple criteria for a non empty subset of a vector space to be a vector space.

Theorem 2.17. For a vector space $V$ over a field $F$, the non empty subset $W$ of $V$ is subspace iff the following condition hold:
i. For any $w_{1}, w_{2}$ in $W, w_{1}+w_{2} \in W$.
ii. For any $c$ in $F$ and $w$ in $W, c w \in W$.

Proof. First we suppose that $W$ is a subspace, then the conditions $(i)$ and (ii) clearly hold.
Conversely, we suppose that $W$ is a non empty subset of a vector space $V$ over a field $F$ such that $(i)$ and (ii) hold. For $-1 \in F,-w=(-1) w \in W$, then

$$
0=w-w,
$$

which implies that $0 \in W$. Also for any $w \in W$

$$
-w=1 \cdot 0+(-1) w .
$$

Which shows that $W$ has additive identity element and every element of $W$ has its inverse. All the properties of addition and scalar multiplication hold in $W$, and it is easy to verify. Hence $W$ is subspace of a vector space $V$.

Now we state two important results. First about the intersection of collection of subspaces is a subspace, and for any two subspaces of a vector space their union is again a subspace.

Proposition 2.18. Let $V$ be a vector space and let $\left\{W_{i} \mid i \in I\right\}$ be a collection of of subspaces of $V$. Then their intersection $\cap_{i \in I} W_{i}$ is a subspace.

Proposition 2.19. For any two subspaces $W_{1}$ and $W_{2}$ of a vector space $V$, the union $W_{1} \cup W_{2}$ is a subspace of $V$ iff $W_{1} \subseteq W_{2}$ or $W_{2} \subseteq W_{1}$.

### 2.8 Dimension and basis

In this section we will define the dimension of a vector space and basis. Before defining basis we need to define two more things, the linearly independence and spanning property. So these two things will be defined before basis.

## Definition 2.22. Linearly Dependent

Let $V$ be a vector space over a filed $F$. The $n$ vectors $x_{1}, x_{2}, \cdots, x_{n} \in V$ are called linearly dependent if there exist $n$ elements $c_{1}, c_{2}, \cdots, c_{n} \in F$ which are not all equal to zero such that

$$
c_{1} x_{1}+c_{2} x_{2}+\cdots+c_{n} x_{n}=0
$$

## Definition 2.23. Linearly independent

Let $V$ be a vector space over field $F$. The $n$ vectors $x_{1}, x_{2}, \cdots, x_{n} \in V$ are called linearly independent if there exist $n$ elements $c_{1}, c_{2}, \cdots, c_{n} \in F$ such that

$$
c_{1} x_{1}+c_{2} x_{2}+\cdots+c_{n} x_{n}=0,
$$

implies that

$$
c_{1}=c_{2}=\cdots=c_{n}=0 .
$$

## Definition 2.24. Spanning property

Let $V$ be a vector space over a field $F$. The set $\left\{v_{1}, v_{2}, \cdots, v_{n}\right\}$ spans $V$, if for every $x \in V$ there are $c_{1}, c_{2}, \cdots, c_{n} \in F$ such that

$$
\begin{equation*}
x=c_{1} v_{1}+\cdots+c_{n} v_{n} . \tag{2.3}
\end{equation*}
$$

In other word it can be defined as the set $\left\langle v_{1}, v_{2}, \cdots, v_{n}\right\rangle$. of all linear combinations of the vectors in a given set $\left\{v_{1}, v_{2}, \cdots, v_{n}\right\}$.

Definition 2.25. A vector space $V$ over a field $F$ is said to be a finite dimensional if there are finite number of elements $x_{1}, x_{2}, \cdots, x_{n}$ in $V$ such that $V=\left\langle x_{1}, x_{2}, \cdots x_{n}\right\rangle$. If no such finite set of elements exists then $V$ is called infinite dimensional.

## Definition 2.26. Basis

Let $V$ be a vector space over a field $F$. A subset $B$ of $V$ is a basis of $V$ if the following conditions hold:
i. $B$ spans $V$.
ii. $B$ is linearly independent subset of $V$.

The numbers $c_{i}$ in (2.3) are called the coordinates of the vector $x$ with respect to the basis $B$, and by the second property it is clear that they are uniquely determined.

Proposition 2.20. A finite subset $B$ of a vector space over a field $F$ is a basis of $V$ iff every element of $V$ is a unique linear combination of the elements of $B$.

## 3 Introduction to coding theory

Some methods are used to send information from one place to another. It is quite natural that error arises during the transmission. The study of methods to transmit the information more efficiently and precisely is called the coding theory. It helps us to minimize the noise and to provide more accurate information. It helps us to control the noise in compact disc recordings, the transmission of information through telephone lines, the transmission of data from one computer to another or the information received from the distant sources like a satellite. In the begning of the history of coding theory, it was used in the Voyager spacecraft to send noise pictures of Jupiter and Saturn. The material in this chapter is taken from [6].

We need a source to send and receive information from one place to another. Such source is said to be a channel. Examples of channels are telephone lines and the atmosphere. Sometimes it happens that the sent information is different from the received one, it is because of the noise on the channel. There may be several reasons behind the creation of noise. Some of the reasons are weather, sunlight, poor typing, poor telephone lines, etc.

The aim of the study of coding theory is to detect and correct the errors caused by the noise on the channel. The diagram below gives the general idea about information transmission system. The main and important part of study is the noise on the channel

and without noise, the study of coding theory is aimless. After handling the noise in a proper way we look forward to maintain the following things.

1. fast encoding,
2. easy transmission of data,
3. easy error detection and correction, and
4. fast decoding.

In our daily life, we use words in written or spoken form. We use these words for communication or to send information which we already have. In this case the channels are the space and the paper. During communication poor speech, bad grammar, bad hearing and bad hand writing causes the noise on the channel.

### 3.1 Basic assumptions

We state some fundamental definitions and assumptions which we will apply throughout the text.

Definition 3.1. fields: A set $F$ having at least two elements and two binary operations namely addition and multiplication defined on it, is called field if the following conditions are satisfied.
i. $F$ is a commutative ring with respect to addition and multiplication and contains multiplicative identity; and
ii. every non-zero element of $F$ is invertible w.r.t. multiplication.

If $p$ is a prime integer, consider the set $F_{p}=\{0,1,2, \cdots p-1\}$ of $p$ elements in which addition $\bigoplus$ and multiplication $\odot$ are defined modulo $p$. For $p=2$, we denote the field $F_{2}$ by $B$ ('b' for binary). Thus $B=\{0,1\}$ with binary operations addition and multiplication defined by $0+0=0,1+0=1=1+0,1+1=0,1 \times 1=1,0 \times 1=0 \times 0=0$. Throughout this chapter, we are concerned with the field $B$ of two elements. There also exist other finites fields and we will study these later.

Let $B^{n}$, where $n$ is a positive integer, denote the set of all ordered n-tuples or sequences of length $n$ i.e. for $a \in B, a=a_{1}, \cdots a_{n}$, with $a_{i}$ belonging to the field $B$.

In many cases, the information to be sent is transmitted by a sequence of zeros and ones. We call a 0 or a 1 a digit.
Definition 3.2. code words: Let $E: B^{m} \rightarrow B^{n}$ be an encoding function and $D: B^{n} \rightarrow B^{m}$ be the decoding function. The elements of the set $B^{n}$ are called codewords and the elements in the set $B^{m}$ are called simply the message words.

A binary code is a set $C$ of words. The code consisting of all words of length two is

$$
C=\{00,10,01,11\} .
$$

A code having all its words of the same length is called block code. The number of digits in a codeword is called its length. The number of codewords in a code $C$ is denoted by $|C|$. In the study of coding theory, need certain assumptions about the channel. The whole study is based on these assumptions.

The first assumption is that a code word of length $n$ consisting of 0 's and 1 's is received as a word of length $n$ consisting of 0 's and 1 's, however this is not necessary that the same word is received as it was sent. The second is that there should be no difficulty to understand the meaning of the first word transmitted. For example, if we are using codewords of length 2 and receive 011011 , we know that the words received are, in order, $01,10,11$. This assumption assures that the word 011 can not be received because a digit has been lost during the transmission. The third and the final assumption is that the probability of any one digit wrongly transmitted does not effect the other digits.

If the channel is perfect and have no noise then the digit received will be the same as the digit sent. It would be nice if the channel is perfect and noiseless, because for the perfect channel there would be no need for the coding theory. But every channel has some sort of noise, some are less noisy and some are more noisy.

If we send 0 and 1 in the channel and the probability of receiving the correct digit is independent of the digits 0 and 1 . Then, such channel is called binary symmetric channel (BSC). The probability that the digit sent is the digit received, is denoted by $p$ and $1-p$ denotes the probability that the digit received is not the digit sent, $0 \leq p \leq 1$. The probability $p$ shows the reliability of a binary symmetric channel.

The diagram 3.1 may clarify how a BSC operates:
A channel is more reliable than another if it has the higher probability than the other. For $p=1$, the sent and the received information will be exactly the same. In such case, we say that the channel is perfect and of no interest to us. Also a channel with $p=0$ is of no interest. It is easy to convert a channel with $0 \leq p \leq 1 / 2$ into a channel with $1 / 2 \leq p \leq 1$. Henceforth we will always assume that we are using a BSC with probability $p$ satisfying $1 / 2 \leq p \leq 1$. Now we introduce two important terms distance and weight of a code word:


Figure 3.1:

## Definition 3.3. Distance

Let $a$ and $b$ be the words of length $n$. The Hamming distance, or simple distance, between $a$ and $b$ is the number of places in which $a$ and $b$ are different. We denote the distance between $a$ and $b$ by $d(a, b)$. Mathematically it can be defined as

$$
d(a, b)=\sum_{i=1}^{n} x_{i} \quad \begin{cases}x_{i}=0, & \text { if } a_{i}=b_{i} \\ x_{i}=1, & \text { if } a_{i} \neq b_{i}\end{cases}
$$

where $a=a_{1} a_{2} \cdots a_{n}$ and $b=b_{1} b_{2} \cdots b_{n}$.
Example 3.1. If $a=01101$ and $b=00111$, then $d(a, b)=2$.
Example 3.2. If $a=10011011$ and $b=11001101$, then $d(a, b)=4$.

## Definition 3.4. Weight

Let $a$ be a word of length $n$, we define the weight $\mathrm{wt}(a)$ of $a$ as the number of times the digit 1 occurs in $a$. In general the number of non zeros entries in a codeword if it is not binary, is called the weight of the word.

Example 3.3. If $a=110101$, then $\operatorname{wt}(a)=4$.
Example 3.4. If $a=10011011$, then $\operatorname{wt}(a)=5$.
Note that for the binary codes, the distance between $a$ and $b$ is the same as the weight of the error pattern $c=a+b$ :

Lemma 3.1. If $a$ and $b$ are words of length $n$, then $d(a, b)=w t(a+b)$
Proof. Let $a=a_{1} a_{2} \cdots a_{n}$ and $b=b_{1} b_{2} \cdots b_{n}$ be any two words of length $n$. For any $i$, $1 \leq i \leq n, a_{i}+b_{i}=1$ iff $a_{i} \neq b_{i}$. Hence the pair $\left(a_{i}, b_{i}\right)$ contributes 1 to $\mathrm{wt}(a+b)$ iff it contributes 1 to $d(a, b)$. Therefore, $d(a, b)=w t(a+b)$

Corollary. If $a$ and $b$ are codewords of length $n$, then $d(a+c, b+c)=d(a+b)$.
Lemma 3.2. If $a, b, c \in B^{n}$, then $d(a, b) \leq d(a, c)+d(b, c)$.

## Definition 3.5. Nearest-neighbor decoding principle

The nearest-neighbor decoding principle means that if we receive a word which is not a codeword, then we wish to replace that word with nearest codeword. We do this by calculating the distances of the received word with all codewords and replace the word with a codeword having minimum distance, if such codeword is unique. In a case of two codewords with same minimum distance from the received word, we say that this is a decoding failure.

In the case of binary symmetric channel in which all code words are equally likely to be transmitted and the single-symbol error probability is less than 1 , the principle is called the maximum likelihood decoding principle.

### 3.2 Correcting and detecting error patterns

In this section we consider the possible ways of correcting and detecting errors. The basic concepts that are involved in the process of correcting and detecting errors will also be discussed. Formally it will be done in next sections.

If a codeword is sent and the received word is not a codeword, it means that during the transmission some error has occurred, so we have detected an error. And if the received word is a codeword then perhaps no error has occurred during the transmission. In this case the error can not be detected.

Example 3.5. Let $C=\{0000,0101,1011,1111\}$ be a code. We suppose that 1101 is the received word. It is clear that the received word is not a codeword, this means that some error has occurred. Since 1101 is not a codeword, we can form the codeword 1011 by changing one digit. All other codewords can be formed but to do this we have to change more than one digit. Therefore we say that the transmitted codeword was 1111, so we correct 1101 to 1111 by replacing 1 instead of 0 .

## Definition 3.6. Detected and undetected errors

Let $a$ be any codeword and $e$ be an error. Then we say that an error $e$ is detected if $a+e$ is not a code word. And if $a+e$ is again a codeword, then we say that the error is undetected.

If a word is transmitted having length $n$, and if $k$ are the number of entries that are received incorrectly then we say that $k$ errors has occurred. We create a word $e$ of the weight $k$ by putting 1 at positions that are corresponding to $k$ errors and by putting 0 on the other places. So we create a word $e$ of weight $k$. The non-zero positions in the error vector where we put 1 gives a set of error having weight $k$. Clearly we can say that a set of $k$ errors can be detected if we detect an error word of weight $k$.

Theorem 3.3. The necessary and sufficient condition for a code $C$ to detect all $k$ or fewer errors is that the minimum distance between any two codewords be $k+1$ or more, i.e $d(C) \geq k+1$.

Proof. Let $C$ be the set of all code words (of length $n$ ) of the given code. Suppose that $\forall b, b^{\prime} \in C$ such that $b \neq b^{\prime}, d\left(b, b^{\prime}\right) \geq k+1$. Let $b \in C$ be transmitted and $e=e_{1} e_{2} \cdots e_{n}$ be the error word caused by the channel during the transmission with

$$
w t(e)=\left(\sum_{i=1}^{n} e_{i}\right) \leq k .
$$

Then the received word is $b+e$ and

$$
d(b+e, b)=w t(b+e+b)=w t(e+2 b)=\mathrm{wt}(e) \leq k .
$$

According to our supposition distance between any two words in $C$ is at least $k+1$. Therefore $b+e$ is not a codeword, so we say that an error $e$ is detected.

Conversely, we suppose that the code is able to detect all sets of $k$ or fewer errors. This mean that $b+e$ is not a codeword for any code word $b$ and a word $e$ with $\operatorname{wt}(e) \leq k$. Suppose that $b$ and $b^{\prime}$ be any two codewords such that $d\left(b, b^{\prime}\right) \leq k$. Let $e$ be an error word such that $e=b+b^{\prime}$. Then $\mathrm{wt}(e) \leq k$. Also $b+e=b+b+b^{\prime}=b^{\prime}-$ a code word. This shows that the error vector $e$ is undetected. This is a contradiction, hence

$$
d\left(b, b^{\prime}\right) \geq k+1 \quad \forall b, b^{\prime} \in C \quad b \neq b^{\prime} .
$$

Corrected errors can also be defined as:
Definition 3.7. A code $C$ can correct an error word $e$ if for every code word $b, D(b+e)=b$ where $D$ is the decoding function of the code. We also say that a set of $k$ errors is corrected if the corresponding error word of weight $k$ is corrected.

## Definition 3.8. Parity check code

The parity check code ( $m, m+1$ ), is the code given by the encoding function $E: B^{m} \rightarrow$ $B^{m+1}$ defined by

$$
E\left(a_{1} a_{2} \cdots a_{m}\right)=a_{1} a_{2} \cdots a_{m} a_{m+1}
$$

where

$$
a_{m+1}= \begin{cases}1, & \text { if } \mathrm{wt}(a)=\mathrm{wt}\left(a_{1} a_{2} \cdots a_{m}\right) \text { is odd } \\ 0, & \text { if } \mathrm{wt}(a) \text { is even }\end{cases}
$$

Example 3.6. Consider the $(2,3)$ parity check code. Then the encoding function $E$ is

$$
00 \rightarrow 000,01 \rightarrow 011,10 \rightarrow 101,11 \rightarrow 110
$$

Thus the set of all code words is

$$
C=\{000,011,101,110\}
$$

There are only three possible error vectors of weight 1 , which are given by

$$
001,010,100
$$

and if any one of the error vectors given above added to any code word does not yield a code word. Thus every single error is detected by this code.

Now suppose that the code word 101 is transmitted and the channel adds to it the error vector 010. The received word is then 111. But it is clear that the same word will be received if 110 or 011 were transmitted and the channel had added the error vector 001 or 100 respectively. It is also obvious that the received word is equidistant from three code words, so this error is not corrected.

Theorem 3.4. A binary code with minimum distance $d=2 k+1$ can correct up to $k$ or fewer errors.
Proof. We consider a code $C$ of length $n$ with the minimum distance $d=2 k+1$ and let $e$ be an error word of weight at most $k$. Let $b$ be the transmitted code word and the received word is $r=b+e$. If $b^{\prime}$ is any code word other than $b$ i.e $b \neq b^{\prime}$, then

$$
\begin{aligned}
d\left(r, b^{\prime}\right) & =d\left(b+e, b^{\prime}\right) \\
& =\mathrm{wt}\left(b+e+b^{\prime}\right) \\
& =\mathrm{wt}\left(b+b^{\prime}+e\right)
\end{aligned}
$$

Now

$$
\mathrm{wt}\left(b+b^{\prime}\right)=d\left(b, b^{\prime}\right) \geq 2 k+1
$$

and $\mathrm{wt}(e) \leq k$. Therefore

$$
\mathrm{wt}\left(b+b^{\prime}+e\right) \geq k+1
$$

i.e.

$$
d\left(r, b^{\prime}\right) \geq k+1
$$

Also

$$
d(r, b)=\mathrm{wt}(r+b)=\mathrm{wt}(b+b+e)=\mathrm{wt}(e) \leq k
$$

Hence $b$ is the nearest code word to $r$, the received word and by the maximum likelihood decoding principle $D(r)=b$. Thus, the error vector $e$ with $\mathrm{wt}(e) \leq k$ is corrected.

### 3.3 Maximum likelihood decoding

We have two basic problems of coding theory and now we are going to give more precise formulation of these problems. We assume that we are at the receiving end of a binary symmetric channel (BSC) and we want to receive a message from the transmitter at the other end. In fact the design of the transmitter is now of the basic problems.

We have no control on two quantities. One is the probability $p$ of correctly transmitted digit and the second is the possible number of transmitted messages.

### 3.3.1 Encoding

We make some choices to determine a code which can be used for sending the message. Firstly a positive integer $k$ which is the length of each binary word corresponding to a message is chosen. Since a binary code of length $k$ can have maximum $2^{k}$ elements, therefore $k$ must be chosen such that $|M| \leq\left|B^{k}\right|$, where $|M|$ is the number of elements in the set $M$. Secondly, the addition of parity check digits to each word of length $k$ is done to make sure that the certain number of errors can be corrected or detected. In other words the transmitter transmits the codeword of length $n$ when he selects the message word of length $k$.

### 3.3.2 Decoding

Suppose a word $a$ in $\mathbb{B}^{n}$ is received. A method called maximum likelihood decoding (MLD), for deciding which word $b$ in $C$ was sent actually is described as follows.

1. Complete Maximum Likelihood Decoding (CMLD)

If there is one and only one word $b$ in $C$ closer to $a$ than any other word in $C$, we decode $a$ as $b$. That is, if $d(b, a)<d\left(b_{1}, a\right)$ for all $b_{1}$ in $C, b_{1} \neq b$, then decode $a$ as $b$. If there are several words in $C$ close to $a$, i.e. at the same distance from $a$, then we select arbitrarily one of them and conclude that it was the codeword sent.
2. Incomplete Maximum Likelihood Decoding (IMLD)

Again if there is one and only one word $b$ in $C$ closest to $a$, then we decode $a$ as $b$. But if there are several words in $C$ close to $a$, then we request a retransmission. In some cases we might even ask for a retransmission if the received word $a$ is too far from any word in the code.

## 4 Finite fields

### 4.1 Introduction

In this chapter, we try to get acquainted with algebraic concept of finite fields. Finite fields are used heavily in the study of coding theory. This concept will serve as our primary tools for constructing codes, for instance, a construction of a double-error correcting binary code, whose description and analysis uses finite fields. A few basic definitions of algebraic structures are discussed in the first chapter.

Some important definitions related to finite fields are given in the beginning of the chapter. It will be shown that the multiplicative group of finite fields is cyclic. We also prove that the size of any finite field must be a power of a prime and that this necessary condition is sufficient, that is, every power of a prime is a size of some finite field.

## Definition 4.1. Extension of a field

Let $K$ be a field and $F$ be a subfield of $K$. Then $K$ is called an extension of the field $F$. It is denoted by $\left.K\right|_{F}$. The field $K$ can be regarded as a vector space over $F$ by using the multiplication in $K$. The dimension of the vector space $K$ over $F$ is called the degree of extension $K$ of $F$ and is denoted by $[K: F]$.
Definition 4.2. The intersection of all subfields of any field is called the prime subfield.
The prime field is unique and is the smallest subfield of the field. Generally, a field having no proper subfield is called a prime field.

### 4.2 Prime fields

For a prime $p$, we let $\operatorname{GF}(p)$ (Galois field of size $p$ ) denote the ring of integer residues modulo $p$ (the ring is also denoted by $\mathbb{Z}_{p}$ ).
By Euclid's algorithm for integers, for every integer $a \in\{1,2, \cdots, p-1\}$ there exist integers $s$ and $t$ such that

$$
s a+t p=1 .
$$

The integer $s$, taken modulo $p$, is the multiplicative inverse $a^{-1}$ of $a$ in $\mathrm{GF}(p)$. Therefore, $\mathrm{GF}(p)$ is a field.

Example 4.1. In GF(7) we have

$$
2 \cdot 4=3 \cdot 5=6 \cdot 6=1 \cdot 1=1 .
$$

Also, $a^{6}=1$ for every non-zero $a \in \mathrm{GF}(7)$. The multiplication group of $\mathrm{GF}(7)$ is cyclic, and the elements 3 and 5 are the generators of this field, since

$$
\begin{array}{cc}
3^{0}=1=5^{0} & 3^{3}=6=5^{3} \\
3^{1}=3=5^{5} & 3^{4}=4=5^{2} \\
3^{2}=2=5^{4} & 3^{5}=5=5^{1} .
\end{array}
$$

Let $F$ be a field. The symbols 0 and 1 stand for the additive and multiplicative identity elements of $F$, respectively. The multiplicative group of $F$ will be denoted by $F^{*}$, and the multiplicative order of an element $a \in F^{*}$, if it exists, will be denoted by $O(a)$. In particular, this order exists whenever $F$ is finite, in which case we get the following result.
Proposition 4.1. Let $F$ be a finite field. For every $a \in F, a \neq 0$

$$
a^{|F|-1}=1
$$

Proof. It is obvious that $0^{|F|}=0$. For $a \in F^{*}$, we know by Lagrange's Theorem that $O(a)$ divides $\left|F^{*}\right|$. Hence $a^{\left|F^{*}\right|}=1$.

### 4.3 Polynomials

We talk about the polynomial over a field $F$. The definitions including polynomial, zero polynomial, degree of polynomial and some examples of irreducible polynomials will be given. We recall the definition of polynomial from the chapter 2.

Let $F$ be a field. An expression of the form

$$
a(x)=a_{0}+a_{1} x+a_{2} x^{2} \cdots+a_{n} x^{n}
$$

is called a polynomial of degree $n$ over $F$ in the indeterminate $x$, where $a_{0}, a_{1}, a_{2}, \cdots a_{n} \in$ $F$. The $a_{i}^{\prime} s$ are called coefficients of the polynomial $a(x)$. The degree of a polynomial $a(x)$ is defined as the highest power of the variable $x$ with non zero coefficients and is denoted by deg $a(x)$. We denote the set of all polynomials over a field $F$ by $F[x]$.

## Definition 4.3. Zero polynomial

A polynomial whose all coefficients are zero is said to be zero polynomial.

## Definition 4.4. Monic polynomial

Let $a(x)$ be a non zero polynomial over the field $F$. We say that $a(x)$ is monic if the coefficient of $x^{\operatorname{deg} a}$ is 1 .

Let $a(x)$ and $b(x)$ be two polynomials over $F$ and $a(x) \neq 0$. We can find unique polynomials $q(x)$, the quotient, and $r(x)$, the remainder, by using the long division such that

$$
b(x)=a(x) q(x)+r(x)
$$

where $\operatorname{deg} r<\operatorname{deg} a$.
By

$$
b(x) \equiv c(x) \quad(\bmod a(x))
$$

we mean that $b(x)$ is congruent to $c(x)$. It means, in other words, that $a(x)$ divides $b(x)-$ $c(x)$. Euclid's algorithm can also be applied to polynomials, in a same manner an we do for integers, to find their greatest common divisor (gcd).

## Definition 4.5. Zero of a polynomial

Let $a(x)$ be a polynomial over $F$. An element $\alpha \in F$ is said to be zero of $a(x)$, if $a(\alpha)=0$.

For example let $a(x)=x^{2}+1 \in \mathbb{F}_{2}$. Then $1 \in F$ is zero of $a(x)$, since $a(1)=0$.

## Definition 4.6. Irreducible polynomial

A non constant polynomial $f(x) \in F[x]$, where $F$ is a field, is called irreducible if

$$
f(x)=g(x) h(x) \quad \text { for } g(x), h(x) \in F[x] \Longrightarrow \operatorname{deg} g(x)=0 \text { or } \operatorname{deg} h(x)=0
$$

The polynomial $f(x)$ is called reducible if it does not satisfy the above condition.
Theorem 4.2. Let $F$ be a field and $f(x) \in F[x]$ be an irreducible polynomial. Then $F[x] /\langle f(x)\rangle$ is a field.

Proof. Let $I$ denote the ideal $\langle f(x)\rangle$ of $F[x]$ generated by $f(x)$. If $I=F[x]$, then $1=$ $f(x) g(x)$ for some $g(x) \in F[x]$. By compairing the degrees of the both sides we see that $f(x)$ is a constant polynomial which is not true. So $F[x] / I$ has at least two elements and
it is a commutative ring with identity.
Let $g(x) \in F[x], g(x) \notin I$. Then

$$
J=\{a(x) f(x)+b(x) g(x): a(x), b(x) \in F[x]\}
$$

is an ideal of $F[x]$ and there exist $h(x) \in F[x]$ such that $J=\langle h(x)\rangle$. since one can show that every ideal in a polynomial ring over a field is a principal ideal. Now

$$
f(x)=1 \times f(x)+0 \times g(x)
$$

is in $J$ and therefore

$$
f(x)=a(x) h(x) \quad \text { for some } a(x) \in F[x]
$$

Irreducibility of $f(x)$ shows that $\operatorname{deg} h(x)=0$ or $\operatorname{deg} a(x)=0$. If $\operatorname{deg} a(x)=0$, then $a(x)$ is a unit in $F[x]$ so that $h(x) \in I$ which implies that $J=I$. This is a contradiction because $g(x) \in J$ but $g(x) \notin I$. Hence $h(x)$ is a unit in $F[x]$ and $J(x)=F[x]$. Therefore

$$
1=a(x) f(x)+b(x) g(x) \quad \text { for some } a(x), b(x) \in F[x]
$$

and

$$
1+I=(b(x)+I)(g(x)+I)
$$

Thus $g(x)+I$ is invertible and $F[x] / I$ is a field.
Lemma 4.3. Let $a(x), b(x)$, and $c(x)$ be polynomials over $F$ such that $c(x) \neq 0$ and $\operatorname{gcd}(a(x), c(x))=1$. Then

$$
c(x)|a(x) \cdot b(x) \Longrightarrow c(x)| b(x) .
$$

Proof. It is given that $\operatorname{gcd}(a(x), c(x))=1$, therefore by Euclidean algorithm there are polynomials $s(x)$, and $t(x)$ over $F$ such that

$$
s(x) \cdot a(x)+t(x) \cdot c(x)=\operatorname{gcd}(a(x), c(x))=1
$$

Since we are to divide by $c(x)$ therefore, we can write

$$
s(x) \cdot a(x) \equiv 1(\bmod c(x)) .
$$

Now multiplying both sides by $b(x)$, we get

$$
s(x) \cdot a(x) \cdot b(x) \equiv b(x)(\bmod c(x))
$$

Now, since

$$
s(x) \cdot a(x) \cdot b(x) \equiv 0(\bmod c(x))
$$

we must have

$$
b(x) \equiv 0(\bmod c(x)) .
$$

This shows that $c(x) \mid b(x)$.

### 4.4 Primitive elements of finite fields

A very important property of the multiplicative group of a finite field is that this group is cyclic. This will be discussed in this section. A finite field can be generated by a single element of the field. There may be more elements such that the elements of a finite field can be written as powers of these elements.

## Definition 4.7. Primitive element

Let $F$ be a finite field and let $a \in F$. Then $a$ is said to be primitive element of cyclic group $F^{*}$ if $a$ generates $F^{*}$, i.e each element of $F$ can be written as a power of $a$. The element $a$ will generate all elements of $F^{*}$.
The definition of primitive element can also be stated as follows:
If

$$
F^{*}=\langle a\rangle=\left\{a^{n} \mid n \in \mathbb{Z}\right\} \quad \text { for some } a \in F,
$$

then $a$ is called a primitive element of $F$ if $F=\langle a\rangle \cup\{0\}$.
If $a$ is a primitive element in $F$, then $a^{i}$ is primitive element if and only if $\operatorname{gcd}\left(i,\left|F^{*}\right|\right)=1$. Hence the number of primitive elements in $F=G F(q)$ is $\phi(q-1)$, where $\phi$ is Euler function defined below.
Let $\mathbb{Z}$ be set of positive integers and let $\phi: \mathbb{Z}^{+} \rightarrow \mathbb{Z}$ be the Euler function. The $\phi(n)$ is the number of elements in the set $\{1,2, \cdots, n\}$ that are relatively prime to $n$.

Example 4.2. We have $F_{7}^{*}=\{1,2 \cdots, 6\}$. By trial and error, we find that each element of $F_{7}^{*}$ can be written as powers of $3(\bmod 7)$, so 3 is a primitive element of $F_{7}^{*}$.

Example 4.3. Let $P(x)=x^{4}+x+1$. We construct the field $\mathrm{GF}\left(2^{4}\right)$ over $F=\mathrm{GF}(2)$ modulo the polynomial $P(x)$. It will contain 16 elements. Since, we are taking $\bmod \left(x^{4}+\right.$ $x+1)$, we have $x^{4}=x+1$. The sixteen elements of $G F\left(2^{4}\right)$ are listed below

$$
\begin{aligned}
G F\left(2^{4}\right)= & \left\{0,1, x, x^{2}, x^{3}, 1+x, x+x^{2}, x^{2}+x^{3}, 1+x+x^{3}, 1+x^{2}, x+x^{3}, 1+x+x^{2},\right. \\
& \left.x+x^{2}+x^{3}, 1+x+x^{2}+x^{2}, 1+x^{2}+x^{3}, 1+x^{3}\right\}
\end{aligned}
$$

In the above finite field, each element can be expressed as a power of $x$ modulo the polynomial $\left(x^{4}+x+1\right)$. Hence $x$ is the primitive element.

Example 4.4. Let $F$ be the field of 5 elements. Here $f(x)=x^{2}+2$ is irreducible in $F[x]$, since no element of $F$ is a root of $f(x)$. Hence

$$
K=F[x] /\langle f(x)\rangle
$$

is of the order 25 . An arbitrary element of $K$ will be of the following form

$$
a x+b+\langle f(x)\rangle \quad a, b \in F
$$

We write $x+\langle f(x)\rangle=y$. Then $y^{2}=3, y^{3}=3 y, y^{4}=4, y^{5}=4 y, y^{6}=2, y^{7}=2 y, y^{8}=1$. We see that $y$ does not generate all elements of $K$, so $y$ is not a primitive element of $K$.

### 4.5 Splitting field

We give the definition of the splitting field along with some theorems about splitting fields.

## Definition 4.8. Splitting field

Let $F$ be a field and $f(x) \in F[x]$. An extension field $K$ of $F$ is called a splitting field of $f(x)$ if
(i) $f(x)$ factors as a product of linear factors over $K$ and
(ii) if $\alpha_{1}, \alpha_{2}, \cdots \alpha_{m}$ are zeros of $f(x)$ then $K=F\left(\alpha_{1}, \alpha_{2}, \cdots \alpha_{m}\right)$.

Proposition 4.4. Let $f(x)$ be an irreducible polynomial over a field $F$. Then there exist an extension $K$ of $F$ in which $f(x)$ has a zero.

Theorem 4.5. [7, p62]
Let $F$ be a field and $f(x) \in F[x]$. Then there exists a splitting field of $f(x)$ over $F$.
Proof. We prove this theorem by induction on the degree of $f(x)$. If $\operatorname{deg} f(x)=1$, then obviously $F$ itself is splitting field of $f(x)$. Suppose now that $\operatorname{deg} f(x)=n \geq 2$. Then by Proposition 4.4 it follows that there is an extension of $F$ which $f(x)$ has a zero $\alpha_{1}$. Let

$$
F_{1}=F\left(\alpha_{1}\right)
$$

Then

$$
f(x)=\left(x-\alpha_{1}\right) g(x)
$$

where $g(x) \in F_{1}[x]$ and $\operatorname{deg} g(x)=n-1$. By our hypothesis $g(x)$ has a splitting field $K$ over $F_{1}$, i.e. $g(x)$ factors as a product of linear factors over $K$ and

$$
K=F_{1}\left(\alpha_{2}, \alpha_{3}, \cdots \alpha_{n}\right)
$$

where $\alpha_{2}, \alpha_{3}, \cdots \alpha_{n}$ are the roots of $g(x)$. But then $\alpha_{1}, \alpha_{2}, \alpha_{3}, \cdots \alpha_{n}$ are the roots of $f(x)$, $K=F\left(\alpha_{1}, \alpha_{2}, \alpha_{3}, \cdots \alpha_{n}\right)$ and $f(x)$ factors as a product of linear factors over $K$. Hence $K$ is a splitting field of $f(x)$.

Theorem 4.6. For any non constant polynomial $f(x)$ over a field $F$, there exists an extension $K$ of $F$ in which $f(x)$ has a zero.

Proof. For an irreducible polynomial over a field $F$, there is an extension $K$ of $F$ in which that polynomial has a root.

Let $g(x) \in F[x]$ be an irreducible factor of $f(x)$. Then there is an extension $K$ of $F$ in which $g(x)$ has a root $\alpha$. But then $\alpha$ is also a root of $f(x)$.

## Definition 4.9. Characteristic of a field

Let $F$ be a field. The characteristic of $F$, denoted by $c(F)$, is the order of the element 1 in the additive group of $F$, provided that this order is finite. If 1 does not have a finite additive order, then $c(F)$ is defined to be zero.
For example, $c(G F(7))=7$, and $c(\mathbb{R})=0$.
Proposition 4.7. [4, p 63]
If $F$ is a field with $c(F)>0$, then $c(F)$ is a prime.
Proof. Suppose that $c(F)=m n$ for some positive integers $m$ and $n$. Then

$$
0=\sum_{i=1}^{m n} 1=\left(\sum_{i=1}^{m} 1\right)\left(\sum_{i=1}^{n} 1\right) .
$$

Therefore, either $\left(\sum_{i=1}^{m} 1\right)=0$ or $\left(\sum_{i=1}^{n} 1\right)=0$, which implies that either $m$ or $n$ is at least $c(F)(=m n)$. But this is possible only when either $n$ or $m$ equals 1 .

### 4.6 Structure of finite fields

## Definition 4.10. Minimal polynomials

Let $K$ be an extension of a field $F$. An element $\alpha \in K$ is called algebraic over $F$ if there exists a polynomial $f(x) \in F[x]$ which has $\alpha$ as a zero.

Let $\alpha \in K$ be algebraic over $F$ and we consider

$$
A=\{f(x) \in F[x]: f(\alpha)=0\} .
$$

Then $A$ is an ideal of $F[x]$ and $F[x]$ is a principal ideal domain. Let $m_{1}(x) \in F[x]$ be a generator of the ideal $A$. If $a$ is the coefficient of the highest power of $x$ in $m_{1}$, then $m(x)=$ $a^{-1} m_{1}(x)$ is a monic polynomial with $\operatorname{deg} m(x)=\operatorname{deg} m_{1}(x)$ which is also generator of A. If

$$
m(x)=r(x) s(x) \quad \text { for some } \quad r(x), s(x) \in F[x]
$$

then either $r(\alpha)=0$ or $s(\alpha)=0$. It means that $m(x) \mid r(x)$ or $m(x) \mid s(x)$. But

$$
\operatorname{deg} m(x)=\operatorname{deg} r(x)+\operatorname{deg} s(x)
$$

and therefore either $\operatorname{deg} r(x)=0$ or $\operatorname{deg} s(x)=0$. Hence $m(x)$ is irreducible. Also $m(x)$ is monic, irreducible polynomial of the least degree which has $\alpha$ as a root,by its choice. Such polynomial with these properties is called minimal polynomial of $\alpha$ over $F$.

Theorem 4.8. The order of a finite field is $p^{n}$ where $p$ is a prime and $n$ is a positive integer.

Proof. Let $F$ be a finite field. Of course $F$ is a finite group with respect to addition, therefore we can apply Lagrange's theorem on finite groups, we have that $O(F) a=0$ for all $a \in F$. This is similar to say that we add $a$, order of $F$ times. We choose $m$ to be the smallest positive integer with $m a=0$ for every $a \in F$. Let $e$ denote the identity element of $F$. Suppose that $m=r s$, where $r, s$ are integers with $r>1, s>1$. Then

$$
0=m e=(r s) e=(r e)(s e)
$$

Therefore either $r e=0$ or $s e=0$. For any $a \in F$ and positive integer $k, k a=(k e) a$. Therefore, either $r a=0$ for all $a \in F$ or $s a=0$ for all $a \in F$, which is contradiction to the choice of $m$. Hence $m$ must be a prime $p$. It can be easily seen that

$$
K=\{r e: 0 \leq r<p\}
$$

is a subfield of $F$. Since $F$ is finite, $F$ is a finite dimensional vector space over $K$. If $\operatorname{dim}_{K} F=n$, and $x_{1}, x_{2}, \cdots, x_{n}$ is a basis of $F$ over $K$, then every element of $F$ can be uniquely written as

$$
a_{1} x_{1}+a_{2} x_{2}+\cdots+a_{n} x_{n}, \text { where } a_{i} \in K
$$

Since each $a_{i}$ has exactly $p$ choices and choice of any $a_{j}$ is independent of the choice of other $a_{i}$, there are exactly $p^{n}$ elements in $F$.

Let $F$ be a field of order $p^{n}$. Then $F^{*}=F-\{0\}$ is a multiplicative group of order $p^{n}-1$. Therefore, by Lagrange theorem we have,

$$
a^{p^{n}-1}=1 \quad \text { for all } a \in F^{*}
$$

Remark. A finite field $F$ is called a Galois field and if $F$ is a field of order $p^{n}$, we write $F=\mathrm{GF}\left(p^{n}\right)$.

Lemma 4.9. Let $G$ be a non cyclic Abelian group of finite order, say $m$. Then there is a proper divisor $k$ of $m$ such that $x^{k}=1$ for all $x \in G$.

Theorem 4.10. In any finite field $F=G F\left(p^{n}\right)$, the multiplicative group $F^{*}$ of all non zero elements is cyclic.

Proof. We know that $F^{*}$ is an Abelian group of order $q-1$, where $q=p^{n}$. If we consider that $F^{*}$ is not cyclic, then there exists an integer $r$, with $1<r<q-1$ such that $a^{r}=1$ for every $a \in F^{*}$ by lemma 4.9. Hence every $a \in F$ will be a root of the polynomial $x^{r+1}-x$ and therefore

$$
x-a \mid x^{r+1}-x \quad \text { for all } a \in F
$$

Also for $a, b \in F$ and $a \neq b, x-a, x-b$ are relatively coprime. Therefore

$$
\prod_{a \in F}(x-a) \mid x^{r+1}-x
$$

But

$$
\operatorname{deg} \prod_{a \in F}(x-a)=O(F)=q \text { and } r+1<q-1+1=q
$$

From above statement, we see that a polynomial of degree $q$ divides a polynomial of degree less than $q$, which is a contradiction. Hence there is no $r$ with $1<r<q-1$ such that $a^{r}=1$ for all $a \in F$, and so $F^{*}$ is cyclic.

Corollary. Every finite field contains a primitive element.

## 5 Linear Codes

In this section we introduce a new and broad class of codes called linear codes. We will also discuss some powerful mathematical tools that will help us to resolve some of the previously discussed problems of coding theory when applied to codes in this class.

### 5.1 Linear codes

Definition 5.1. A code $C$ is called a linear code if for any two codewords $u$ and $v$ in $C$, $u+v$ is in $C$.

In other words we can say that linear code is a code which is closed under addition of codewords. For example if we consider a code $C=\{00,11\}$, then all the four sums

$$
00+00 \quad 00+11 \quad 11+00 \quad 11+11 .
$$

are in $C$, which shows that $C$ is a linear code. On the other hand, consider a code $C_{1}=$ $\{000,001,101\}$. This is not linear code because $001+101$ is not in $C_{1}$. According to the closure law under addition for a code $C$ to be linear the sum $u+u=0$ must be in $C$ for any $u \in C$. Therefore a linear code must contain the zero word. But some times if a code contain the zero word, then it does not guarantee that the code is linear as in case of a code $C_{1}$. We observe from the linear and non linear codes that a linear code has one advantage over non linear code. The advantage is that in the case of linear codes it is easy to find the distance as compare to the non linear codes. The distance is defined in chapter 3.

In the following section we will see that the linear codes have many advantages over the arbitrary codes and are highly structured. Here are some problems, difficult to solve in general, but relatively easy for linear codes:
i. There is a procedure MLD that is simpler and faster for linear codes.
ii. For a linear code, encoding process is faster and it requires less storage space than for arbitrary non-linear codes.
iii. It is easy to describe the set of error patterns that a linear code will detect.
iv. It is much easier to describe the set of error patterns a linear code will correct than it is for arbitrary non-linear codes.

For studying linear codes the most important tools come from linear algebra. We will discuss in this chapter some basic facts from linear algebra and we will try to show their relevance to coding theory.

In sec:3.1, the field $F_{2}$ is denoted as $B=\{0,1\}$ which contains only two elements 0 and 1. It is also mentioned that $B^{n}$ is the set of all ordered $n$-tuples or sequences of length $n$. So we can define a vector space $B^{n}$ on a field $B$, with the operations of vector addition and scalar multiplication, which satisfy all the properties of a vector space. In the section 2.6 of vector spaces, we have already discussed about vector spaces, subspace, and also some basic definitions.

Now we consider two important subspaces of the vector space $B^{n}$ which will provide interesting examples of linear codes. If $S$ is non empty subset of a vector space $V$, then the linear span of $S$ is defined as the set of linear combinations of a vectors in a given set $S=\left\{v_{1}, v_{2}, \cdots, v_{n}\right\}$, ans it is denoted as $\langle S\rangle$. If $S$ is empty the we define $\langle S\rangle=\{0\}$.

From the theory of linear algebra it is clear that for any subset $S$ of a vector space $V$, the linear span of $S$ is a subspace of the vector space and it is called the subspace generated by $S$. For the vector space $B^{n}$, since $\langle S\rangle$ is subspace of $B^{n}$, so we call $\langle S\rangle$ the subspace of linear codes generated by $S$. We have the following theorem for the vector space $B^{n}$.

Theorem 5.1. For any subset $S$ of $B^{n}$, the code $C=\langle S\rangle$ generated by $S$ consists on the following words: the zero word, all words in $S$, and all sums of two or more words in $S$.

Example 5.1. Let $S=\{010,011,111\}$. The the code generated by $S$ consists of

$$
C=\langle S\rangle=\{000,010,011,111,001,100,101,110\} .
$$

## Definition 5.2. Scalar or dot Product

For any two vectors $a=\left\{a_{1}, a_{2}, \cdots a_{n}\right\}$ and $b=\left\{b_{1}, b_{2}, \cdots b_{n},\right\}$ in $B^{n}$. Then the scalar product of $a$ and $b$ is denoted as $a \cdot b$ and is defined as

$$
(a \cdot b)=a_{1} b_{1}+\cdots+a_{n} b_{n}
$$

The two vectors are orthogonal if there dot product is zero, i.e for any two vectors $a$ and $b$ implies that $a \cdot b=0$. For example consider 011 and 111 belonging to $B^{3}$, as in the previous example. The dot product is then

$$
011 \cdot 111=0+1 \cdot 1+1 \cdot 1=0+1+1=0 .
$$

Hence the vectors 011 and 111 are orthogonal.

## Definition 5.3. Orthogonal complement

Let $S$ be set of vectors in $B^{n}$. A vector $u \in B^{n}$ is orthogonal to a set $S$, if $u$ is orthogonal to every vector in $S$. The set of all vectors orthogonal to every vector in $S$ is called orthogonal complement of $S$ and is denoted by $S^{\perp}$. For the vector space $B^{n}$, if $C=\left\langle S^{\perp}\right\rangle$, then we write $\langle C\rangle=\langle S\rangle$ and $\langle C\rangle$ is called the dual code of $C$. A code $C$ is called self-dual if $C=C^{\perp}$.

Now we will check the linear independence of linear codes, basis and dimension will also be discussed. Here just the examples of the linear codes will be discussed where these things will be applied. Linear independence, basis and dimension are defined in section 2.6.

Example 5.2. In this example, we check the linear independence of a code. Consider $S=\{1011,1101,1011\}$. Let $a_{1}, a_{2}$ and $a_{3}$ be the scalars, such that

$$
a_{1}(1001)+a_{2}(1101)+a_{3}(1011)=0000
$$

By equating the components on both sides, we get

$$
\begin{gathered}
a_{1}+a_{2}+a_{3}=0 \\
a_{2}=0 \\
a_{3}=0 \\
a_{1}+a_{2}+a_{3}=0
\end{gathered}
$$

by solving for $a_{1}, a_{2}$ and $a_{3}$ yields $a_{1}=a_{2}=a_{3}=0$. Which shows that $S$ is linearly independent.

Similarly, we can see that the code $S=\{110,011,101,111\}$ is linearly dependent. There exist constants $a_{1}=1, a_{2}=1, a_{3}=1$ and $a_{4}=0$ such that

$$
1(110)+1(011)+1(101)+0(111)=000
$$

### 5.2 Bases for a Code Generated by $S$ and its Dual Code

In this section we will find methods to find bases for a linear code $C$ and its dual $\langle C\rangle$. In the theory of linear codes these methods have great importance.

Consider that $S$ is a non empty subset of $B^{n}$. In the next two algorithms, we will see how to find basis for a linear code generated by $S$. All the algorithms which are going to be discussed are taken from Ref. [3].

Algorithm 5.2. If a matrix whose rows are the words in $S$. The number of non zero rows of a matrix when it is transformed to row echelon form (REF) by applying elementary row operations, is called basis for $C=\langle S\rangle$.

Example 5.3. Let $S=\{11101,10110,01011,11010\}$. We find basis for the linear code $C=\langle S\rangle$.

$$
A=\left(\begin{array}{l}
11101 \\
10110 \\
01011 \\
11010
\end{array}\right) \rightarrow\left(\begin{array}{l}
11101 \\
01011 \\
01011 \\
00111
\end{array}\right) \rightarrow\left(\begin{array}{l}
11101 \\
01011 \\
00111 \\
00000
\end{array}\right) . \quad[3, p 38]
$$

If we see from the above example, the last matrix is REF form of the matrix $A$. Hence $\{11101,01011,00111\}$ is a basis of the linear code $C$, since the rows of a matrix are words in $C$. By applying elementary row operations, we interchange rows (words) or simply replace. Then as a result we get the rows or word which are again in $C$. Therefore the algorithm works. It is also clear from linear algebra that the non zero rows in a matrix in REF are linearly independent.

In the previous algorithm it was considered that rows of a matrix are words in $C$. In next algorithm we will see that if columns of a matrix are words in $C$, then it is also possible to find basis for a linear code.


#### Abstract

Algorithm 5.3. Form a matrix A whose columns are the words in S. Transform the matrix to row echelon form (REF) by applying elementary row operations and locate the leading columns in REF. Then the original columns of A corresponding to these leading columns form a basis for $C=\langle S\rangle$.


Now we need to find basis of the dual code $C^{\perp}$. The next algorithm will tell us the method to find basis for the dual code.


#### Abstract

Algorithm 5.4. From the matrix $A$ whose rows are the words in $S$, use elementary row operations to place $A$ in reduced row echelon form (RREF). Let $G$ be the $k \times n$ matrix consisting of all the non zero rows of the RREF. Let $X$ be the $k \times(n-k)$ matrix obtained from $G$ by deleting the leading columns of $G$. Let $H$ be an $n \times(n-k)$ matrix formed as follows:


i. in the rows of $H$ corresponding to the leading columns of $G$, place, in order, the rows of $X$;
ii. in the remaining $n-k$ rows of $H$, place, in order the rows of the $(n-k) \times(n-k)$ identity matrix I.

Then the columns of $H$ form a basis for $C^{\perp}$.

Example 5.4. Let $S=\{11101,10110,01011,11010\}$. We form a matrix $A$ whose columns are words in $S$ and transform it into RREF form.
$A=\left(\begin{array}{lllll}1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0\end{array}\right) \rightarrow\left(\begin{array}{lllll}1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0\end{array}\right) \rightarrow\left(\begin{array}{lllll}1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0\end{array}\right) \rightarrow\left(\begin{array}{lllll}1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0\end{array}\right)$,
which is in RREF. Now we write the matrix $G$ as $G=\left(I_{3} X\right)$ where

$$
X=\left(\begin{array}{ll}
0 & 1 \\
1 & 1 \\
1 & 1
\end{array}\right)
$$

The $5 \times(5-3)$ matrix $H$ is

$$
H=\left(\begin{array}{ll}
0 & 1 \\
1 & 1 \\
1 & 1 \\
1 & 0 \\
0 & 1
\end{array}\right)
$$

By algorithm 5.4, the columns of $H$ form a basis for $C^{\perp}$.

### 5.3 Generating matrices

In this section we will find a generating matrix of a linear code $C$ and we will see how messages can be transmitted by using this matrix.

## Definition 5.4. Rank

The rank of a matrix $A$ over $B$ is the number of non zero rows when it is transformed into row echelon form.

The dimension of the code $C$ is the dimension of $C$, as a subspace of $B^{n}$. A linear code if it has length $n$, dimension $k$ and distance $d$ is denoted by $(n, k, d)$.

## Definition 5.5. Generating matrix

Any matrix whose rows form a basis for $C$ is called the generating matrix, where $C$ is the linear code of length $n$ and dimension $m$.

The generating matrix can also be defined in the following way:
Theorem 5.5. Let $G$ be a $k \times n$ binary matrix. Then the mapping $E: A^{k} \rightarrow A^{n}$ defined by

$$
E(x)=x G
$$

will do as an encoding mapping for a linear ( $n, k$ )-code, if and only if every $m=1,2, \cdots, k$ we have that if $a_{1}, a_{2}, \cdots, a_{m}$ are $m$ different rows of $G$, then

$$
a_{1} b_{1}+a_{2} b_{2}+\cdots+a_{m} b_{m}=0 \Longrightarrow b_{i}=0 \quad \forall i,
$$

where $b_{i} \in\{0,1\}$.
Definition 5.6. A binary $k \times n$ matrix $G$ that fulfills the conditions of the previous theorem is a called a generating matrix of a linear $(n, k)$-code.

The important thing to remember about generating matrix for an $(n, k)$-code is that it must have $k$ rows and $n$ columns.

Theorem 5.6. A matrix $G$ is a generator matrix for some linear code $C$ if and only if the rows of $G$ are linearly independent; that is, if and only if the rank of $G$ is equal to the number of rows of $G$. [3, p 42].

Theorem 5.7. Any matrix which is row equivalent to the generator $G$ is also a generating matrix of a linear code $C$.

Example 5.5. Consider a linear code $C=\{0000,1110,0111,1001\}$. We find the generating matrix for $C$. By using algorithm 5.2,

$$
A=\left(\begin{array}{l}
0000 \\
1110 \\
0111 \\
1001
\end{array}\right) \rightarrow\left(\begin{array}{l}
1110 \\
0111 \\
1001 \\
0000
\end{array}\right) \rightarrow\left(\begin{array}{l}
1110 \\
0111 \\
0111 \\
0000
\end{array}\right) \rightarrow\left(\begin{array}{l}
1110 \\
0111 \\
0000 \\
0000
\end{array}\right)
$$

So

$$
G=\binom{1110}{0111}
$$

is a generating matrix for $C$. By using the algorithm (5.4), reduced row echelon form of the matrix $A$ is

$$
\left(\begin{array}{l}
1001 \\
0111 \\
0000 \\
0000
\end{array}\right)
$$

therefore the matrix

$$
G_{1}=\binom{1001}{0111}
$$

is also a generator matrix for $C$.

### 5.4 Parity- check matrices

In this section we will discuss another method to find the matrix that is associated with a linear code and it is closely connected with the generator matrix. This new matrix will be helpful and of great importance in designing decoding scheme.

Definition 5.7. If columns of a a matrix $H$ form a basis for the dual code $C^{\perp}$, then $H$ is called parity- check matrix.

The parity check matrix can also be defined as
Definition 5.8. Let $G$ be a generating matrix of type $(k, n)$ for a linear $(n, k)$ - code. A binary matrix $H$ of type $n \times(n-k)$ such that

$$
G H \equiv O \quad(\bmod 2),
$$

where $O$ is the zero matrix, is called a parity-check matrix.
If the length of the linear code $C$ is $n$ and dimension $m$, then any parity- check matrix for $C$ must have $n$ rows, $n-m$ columns and rank $n-m$, since the sum of the dimensions of $C$ and $C^{\perp}$ is $n$.

Theorem 5.8. For a linear code $C$ a matrix $H$ is a parity-check matrix if and only if the columns of the matrix $H$ are linearly independent.
Theorem 5.9. Let $C$ be a linear code and $n$ be the length of $C$. Let $H$ be the parity-check matrix, then $C$ consists of all words $u \in B^{n}$ such that $u H=0$.

By using algorithm (5.4) we can find a parity-check matrix for a linear code $C$ if a generating matrix for $C$ is given. The matrix $H$ which was constructed in algorithm (5.4) is the parity- check matrix, since from the same algorithm, the columns of $H$ form a basis for $C^{\perp}$.

Example 5.6. Consider a $3 \times 5$ generating matrix

$$
G=\left(\begin{array}{lllll}
1 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 1 & 0
\end{array}\right)=\left(I_{3} A\right),
$$

where

$$
A=\left(\begin{array}{ll}
0 & 1 \\
1 & 1 \\
1 & 0
\end{array}\right)
$$

The parity-check matrix is

$$
H=\left(\begin{array}{ll}
0 & 1 \\
1 & 1 \\
1 & 0 \\
1 & 0 \\
0 & 1
\end{array}\right)
$$

Example 5.7. In example (5.5) we found the generator matrix for the code

$$
C=\{0000,1110,0111,1001\} .
$$

We found that

$$
G_{1}=\left(\begin{array}{llll}
1 & 0 & 0 & 1 \\
0 & 1 & 1 & 1
\end{array}\right)=(I X)
$$

is a generator matrix for $C$ which is RREF. By algorithm (5.4), we construct matrix $H$ as:

$$
H=\binom{X}{I}=\left(\begin{array}{ll}
0 & 1 \\
1 & 1 \\
1 & 0 \\
0 & 1
\end{array}\right)
$$

is a parity check matrix for $C$.
Now we try to summarize our discussion about generator matrix and parity-check matrix for a linear code in the following theorem. We also try to characterize the relationship between a generator matrix and parity-check matrix, and the relationship between these matrices for a linear code and its dual code.
Theorem 5.10. For a linear code $C$, the matrix $G$ is generating matrix and $H$ is a paritycheck matrix if and only if
i. the rows of $G$ are linearly independent,
ii. the columns of $H$ are linearly independent,
iii. If $G$ is of type $k \times n$ then $H$ is of type $n \times(n-k)$ and
iv. $G H=0$.

### 5.5 Equivalent codes

Let $G$ be a $m \times n$ matrix with $m<n$. suppose the first $m$ columns form the $m \times m$ identity matrix $I_{m}$, so

$$
G=\left(I_{m}, X\right) .
$$

Then the matrix $G$ is in RREF and has linearly independent rows. Thus we can say the matrix $G$ is a generator matrix for some linear code of length $n$ and dimension $m$. The matrix of this kind is said to be in standard form.

## Definition 5.9. Systematic code

A code $C$ generated by the generator matrix $G$ defined by $G=\left(I_{m}, X\right)$ is called a systematic code.

A code word in standard form has many advantages. From the following theorem, we can see the advantage of having generator matrix in standard form.

Theorem 5.11. If $G$ is a generator matrix of a linear code of dimension $m$ and length $n$, and if $G$ is in standard form, then the first $m$ digits in the code word $v=u G$ form the word $u$ in $B^{m}$. [3, p 49].

Example 5.8. Let

$$
G=\left(I_{4} X\right),
$$

where

$$
I_{4}=\left(\begin{array}{l}
1000 \\
0100 \\
0010 \\
0001
\end{array}\right)
$$

and

$$
X=\left(\begin{array}{l}
101 \\
100 \\
110 \\
011
\end{array}\right)
$$

If the message is $u=0111$, then $u G=0111001=[u 001]$. And if $u=1011$, then $u G=$ 1011000

If we have a code word $v=u G$ according to the theorem 5.11, the first $m$ digits of the code word are called information digits, since they actually contain the message $u$, while the last $n-m$ digits of $v=u G$ are called redundancy or parity-check digits.

We have seen that a linear code with generator matrix in standard form have advantages. Now it can also happen that a linear code $C$ does not have a generator matrix in standard form, then in this situation what we can do?

Let us consider a code $C$ having four number of words and length of code word is let say three, and also $C$ has no generator matrix in standard form. We do the following that we transform the code into the other code. It can be done in different ways, for example we fix first digit in a codeword and change the other two. We so the same thing with all codeword, by doing this the code $C$ will be transformed into the code, say $C^{\prime}$.

The code $C^{\prime}$ will be different from the word $C$, but they share many properties with each other. Both are linear, have same length and will have the same dimension. The both codes will also have same distance. But here the code $C^{\prime}$ has an advantage over $C$, the advantage is that $C^{\prime}$ has generating matrix in standard form. If $G$ is the generator matrix
of $C$, then the generator matrix for $C^{\prime}$ can be obtained by dong the same operation what we did on $C$ to get $C^{\prime}$.

Similarly if a code $C$ has a length $n$, we can obtain a new code $C^{\prime}$ of the same length. The resulting code $C^{\prime}$ is said to be equivalent to $C$. Now we summarize the above discussion in the following theorem.

Theorem 5.12. Given $C$ (linear) a code $C^{\prime}$ can be found having a standard generating matrix such that $C$ and $C^{\prime}$ are equivalent.

Proof. If $G$ is a generator matrix for $C$, transform $G$ in reduced row echelon form. Rearrange the columns of the reduced row echelon form so that the leading columns come first and form an identity matrix. The result is a matrix $G^{\prime}$ in standard form which is a generator matrix for a code $C^{\prime}$ equivalent to $C$.

### 5.6 Distance of a linear code

The distance of a linear code is the minimum weight of any nonzero codeword. The distance of a linear code can also be determined from a parity-check matrix for the code.

Theorem 5.13. Let $H$ be a parity-check matrix for a linear codeC. Then $C$ has distance $d$ iff any set of $d-1$ rows of $H$ is linearly independent, and at least one set of $d$ rows of $H$ is linearly independent.

## 6 Perfect codes

Definition 6.1. A vector having minimum Hamming weight in a coset of a linear code $C$ is called a coset leader.

The syndrome of a vector $u$ is defined to be $S(u)=u H^{T}$ where $H$ is the parity check matrix. The following lemma allows us to determine the coset easily.

Lemma 6.1. Two vectors $u$ and $v$ belong to the same coset of a linear code $C$ if and only if they have the same syndrome.

Proof. Two vectors $u$ and $v$ to belong to the same coset if and only if their difference belongs to the code $C$; that is, $u-v \in C$. This happens if and only if $(u-v) H^{T}=0$, which is equivalent to $S(u)=u H^{T}=v H^{T}=S(v)$.

## Definition 6.2. Hamming sphere

A Hamming sphere of radius $k$ centered at a codeword $c$ is denoted by $B_{H}(c, k)$ and is defined as

$$
B_{H}(c, k)=\left\{x \mid d_{H}(x, c) \leq k\right\} .
$$

i.e all vectors having Hamming distance $k$ or less to the codeword $c$.

Here we have used the notation $d_{H}$ for the Hamming distance. The reason of using this notation is that in the next chapters we will introduce another metric, so called the Lee metric. To avoid the confusion for the reader we used this notation instead.

In the following lemma, we calculate the number of vectors in the Hamming sphere $B_{H}(c, k)$.

Lemma 6.2. A sphere $B_{H}(c, k)$ in n-dimensional q-ary space has

$$
\binom{n}{0}+\binom{n}{1}(q-1)+\cdots\binom{n}{r}(q-1)^{r}
$$

elements.
Proof. We calculate two things. First the number of vectors having distance 1 from $c$. Then the number of vectors having distance $m$ from $c$.

The vectors at distance 1 from $c$ are those ones that are different from $c$ in exactly one location. Since, here are $n$ possible locations and due to $q$-ary we have $q-1$ ways to make an entry different. So the total number of vectors at Hamming distance 1 is $n(q-1)$.

Now there are $\binom{n}{m}$ ways in which we can choose $m$ locations to differ from the values of $c$. For all of such $m$ locations, we have $q-1$ choices of symbols different from the corresponding symbol of $c$. Therefore, we have

$$
\binom{n}{m}(q-1)^{m}
$$

vectors having Hamming distance $m$ from $c$. We also have to include the vector $c$ too, we know that $\binom{n}{0}=1$. Hence our required result will be

$$
\binom{n}{0}+\binom{n}{1}(q-1)+\cdots\binom{n}{r}(q-1)^{r} .
$$

## Theorem 6.3. Hamming bound

Let C be a q-ary $(n, M, d)$ code with $d \geq 2 k+1$. Then

$$
\begin{equation*}
M \leq \frac{q^{n}}{\sum_{j=0}^{k}\binom{n}{j}(q-1)^{j}} . \tag{6.1}
\end{equation*}
$$

Proof. We draw a Hamming sphere of radius $k$ centered at each codeword. According to the statement, the minimum distance $d \geq 2 k+1$. Therefore these spheres will be disjoint. We are working with $q$ symbols so the total number of vectors in all the Hamming sphere can not exceed $q^{n}$, where $n$ is of course the length of a codeword. Therefore, we get

$$
\left(\text { number of codewords)(number of elements per sphere) }=M \sum_{j=0}^{k}\binom{n}{j}(q-1)^{j} \leq q^{n} .\right.
$$

Example 6.1. We compute an upper bound for the size or dimension $M$ of a linear code $C$ of length $n=6$ and distance $d=3$. From $d=3=2 k+1$ we get $k=1$. The Hamming bound gives

$$
|C| \leq \frac{2^{6}}{\binom{6}{0}+\binom{6}{1}}=\frac{64}{1+6}=\frac{64}{7} .
$$

But $|C|$ must be a power of 2 , so $|C| \leq 8$, and $M \leq 3$.

## Definition 6.3. Perfect codes

A code $C$ of length $n$ and minimal Hamming distance $d=2 k+1$ is called perfect if it satisfies the Hamming bound (6.1) with equality.

Now we give two important examples on perfect codes. The codes in these examples are called trivial perfect codes. Therefore they are not important.

Example 6.2. Let $k=0$. Then

$$
\binom{n}{0}=1=2^{0},
$$

so

$$
|C|=2^{n} /\binom{n}{0}=2^{n} .
$$

The only code having length $n$ and with $2^{n}$ codewords is $C=K^{n} . K^{n}$ is a perfect code.
Example 6.3. Let the length of the code is $n=2 t+1$. Then

$$
\binom{n}{n-i}=\frac{n!}{(n-1)!(n-(n-i))!}=\frac{n!}{(n-i)!i!}=\binom{n}{i} .
$$

Thus it can be written as

$$
\binom{n}{0}=\binom{n}{n},\binom{n}{1}=\binom{n}{n-1}, \cdots,
$$

and, for $n=2 t+1$ implies

$$
\binom{n}{t}=\binom{n}{n-t}=\binom{n}{t+1} .
$$

Therefore

$$
\binom{n}{0}+\cdots+\binom{n}{t}=\frac{1}{2}\left(\binom{n}{0}+\cdots+\binom{n}{n}\right)=\frac{1}{2} \cdot 2^{n}=2^{n-1} .
$$

Hence by using (6.1), we get

$$
|C|=\frac{2^{n}}{\binom{n}{0}+\cdots+\binom{n}{t}}=\frac{2^{n}}{2^{n-1}}=2 .
$$

The above equation shows that any perfect code having length and distance $2 k+1$ has exactly two codewords.

### 6.1 Hamming codes

One important class of single error correcting codes is Hamming codes. These codes can easily be encoded and decoded. In the beginning these codes were used to control errors in long distance calls. The parameters for binary Hamming codes are codes of length $n=2^{m}-1$, dimension $k=2^{m}-m-1$ and minimum distance $d=3$.

We can easily describe the Hamming codes by a parity check matrix. To construct a binary Hamming code of length $n=2^{m}-1$, we construct a matrix of order $m \times n$ in which all non zero binary $m$ - tuples are written in columns.

Example 6.4. Let us find the parity check matrix $H$ for a $[15,11]$ binary Hamming code. We list all non-zero 15 -tuple as the columns of the matrix

$$
\left(\begin{array}{lllllllllllllll}
0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\
1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\
1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0
\end{array}\right)
$$

The above matrix is a parity check matrix, by interchanging columns in the above matrix so that we get $4 \times 4$ identity matrix. This matrix will generate a systematic code.

$$
H=\left(\begin{array}{lllllllllllllll}
0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\
1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\
0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\
1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1
\end{array}\right)
$$

One can now easily calculate a generator matrix from the parity check matrix $H$. The decoding algorithm for Hamming code that corrects up to single bit error is as below. Compute the syndrome $s=y H^{T}$ for any received vector $y$. If $s=0$, it means that there are no errors. If $s \neq 0$, it will determine the position of the column of parity check matrix $H$ that is the transpose of the syndrome. In this case we change the determined bit in received word or vector and get the code.

Example 6.5. From the previous example, assume that we receive the word

$$
\left.y=\left(\begin{array}{llll}
0 & 0 & 0 & 0
\end{array}\right) 00000011001\right) .
$$

The syndrome $s=y H^{T}$ will be $s=(0110)$. This is the transpose of the second column of $H$, so we change the second bit of $y$ and get the code word as (010000000011001).

We can conclude that we have found the error and also corrected it.

### 6.2 Golay codes

The Golay codes $G_{23}$ and $G_{24}$ are very famous binary codes. Infact, the extended Golay code $G_{24}[24,12,8]$ was used by the Voyager I and Voyager II space crafts to provide very nice photographs of Jupiter and Saturn in 1980's. The Golay code $G_{23}[23,12,7]$ is very closely related to $G_{24}$. We first analyze the construction of $G_{24}$ and then modify it to get $G_{23}$. We see how the generating matrix is constructed for $G_{24}$.

### 6.2.1 The extended Golay codes

The generating matrix for $G_{24}$ is $12 \times 24$ matrix given below

$$
G=\left(\begin{array}{llllllllllllllllllllllll}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1
\end{array}\right) .
$$

All the entries of above matrix are integers mod 2. Obviously the first 12 columns are identity matrix of order 12 . We need to think a little bit when it comes to the remaining columns of $G$. The last eleven columns are obtained in the following way. We take the squares of the elements of $\mathbb{Z}=\{0,1 \cdots, 10\} \bmod 11$. The squares are $0,1,3,4,5,9$. We set this vector as the last 11 entries in the first row of $G$, and for the remaining rows, except the last one, we just cyclically permute the entries of this vector. The reason for using squares mod 11 is to determine which place will get 1 . The 13th column and the 12th row in above matrix are included because they increase the number of codewords and the minimum distance. Two of the properties of $G_{24}$ are given in the following theorem.

Theorem 6.4. The extended Golay code $G_{24}$ is a self dual [24, 12] binary code.
Proof. The length of the rows of $G$ is 24 . Since $G$ contains the $12 \times 12$ identity matrix, this means that the 12 rows of $G$ are linearly independent. Hence $G_{24}$ has dimension 12 and we can write it as $[24,12, d]$ code for some $d$. We will later prove that it has minimum distance $d=8$.

Let $r_{1}$ and $r$ such that $r_{1} \neq r$, where $r_{1}$ is the first row of $G$ and $r$ is any row of the remaining 11 rows. An easy check show that that $r_{1}$ has exactly four 1 s common with $r$, and each has four 1 s that are matched with 0 s in the other vector. If we add $r_{1}$ and $r$ then in the sum $r_{1}+r$, the four 1 s which are common cancel ( give zero) mod 2 , and the remaining four 1 s from each which are matched with 0 s give total of eight 1 s . Therefore the weight of $r_{1}+r$ is 8 . Also, the common four 1 s contribute to the dot product of $r_{1}$ and $r$ i.e $r_{1} \cdot r$, so

$$
r_{1} \cdot r=1 \cdot 1+1 \cdot 1+1 \cdot 1+1 \cdot 1=4 \equiv 0 \quad(\bmod 2) .
$$

Now consider any two distinct rows of $G$, say $u$ and $v$, other than the last row. The first 12 and the last 11 entries of $v$ are obtained by permuting the corresponding entries of $u$ and also of the first row cyclically. The weight of the vectors of the values of the dot
product remains fixed, if we permute the entries cyclically, the calculations of sum and dot product which has been done before for $r_{1}$ and $r$ can be applied to $u$ and $v$. Therefore,

1. $w t(u+v)=8$
2. $u \cdot v \equiv 0(\bmod 2)$.

The conditions (1) and (2) holds for any two distinct rows $u, v$ of $G$, even if any of $u$ and $v$ is the last row of $G$. We note that every row of $G$ has an even number of 1 s , so condition (2) holds even if $u=v$.

Let $c_{1}$ and $c_{2}$ be any elements in $G_{24}$ then both of them are linear combination of rows of $G$, so $c_{1} \cdot c_{2}$ is a linear combination of numbers of the form $u \cdot v$. But since (2) holds for all such $u$ and $v$, therefore $C \subseteq C^{\perp}$. Also $C$ and $C^{\perp}$ have same dimension because $C$ is a 12-dimensional subspace of 24-dimensional space and $C^{\perp}$ has dimension $24-12=12$. Hence $C=C^{\perp}$.

Lemma 6.5. For any two binary operators $v_{1}, v_{2}$ of same length

$$
\begin{equation*}
w t\left(v_{1}+v_{2}\right)=w t\left(v_{1}\right)+w t\left(v_{2}\right)-2\left(v_{1} \cdot v_{2}\right), \tag{6.2}
\end{equation*}
$$

where $\left(v_{1} \cdot v_{2}\right)$ means the usual dot product of two vectors.
Proof. Here we are adding $v_{1}$ and $v_{2}(\bmod 2)$. Therefore $v_{1}+v_{2}$ will have entries 1 only when one of the vectors $v_{1}, v_{2}$ has entry 1 and the other has entry 0 at the corresponding positions. On the other hand $w t\left(v_{1}\right)$ equals number of places $v_{1}$ has 1 s and similarly $w t\left(v_{2}\right)$ equals number of places $v_{2}$ has 1 s . For this reason, $w t\left(v_{1}\right)+w t\left(v_{2}\right)$ will include those 1 s that cancels each other when we calculated $w t\left(v_{1}+v_{2}\right)$. There are actually $\left[v_{1} \cdot v_{2}\right]$ entries in $v_{1}$ and also the same number in $v_{2}$ which are included in $w t\left(v_{1}\right)+w t\left(v_{2}\right)$ but are not included in $w t\left(v_{1}+v_{2}\right)$.

Summing up all things together in equation (7.3), we get the equation (7.3) satisfied.

We are now able to prove that the weight of all vectors in $G_{24}$ is multiple of 4 . We do this in the following theorem.

Theorem 6.6. The weight of any word in $G_{24}$ is a multiple of 4 .
Proof. We take a word $g$ in $G_{24}$. We can write it as the following sum

$$
g \equiv u_{1}+u_{2}+\cdots+u_{k} \quad(\bmod \bmod 2),
$$

where $u_{1}, u_{2}, \cdots, u_{k}$ are distinct rows of the generating matrix $G$. To prove the theorem, we show will that $w t(g) \equiv 0(\bmod 4)$ by induction on $k$.

The case $k=1$ is true, since the weights of all rows of $G$ are multiples of 4 . Now we suppose that all the vectors which can be written as a sum of $k-1$ rows of $G$ have weight $\equiv 0(\bmod 4)$, i.e any vector $u$ in the form

$$
u=u_{1}+u_{2}+\cdots+u_{k-1}
$$

has a multiple of 4 as its wight. By using lemma 6.5

$$
w t(g)=w t\left(u+u_{k}\right)=w t(u)+w t\left(u_{k}\right)-2\left(u \cdot u_{k}\right) \equiv 0+0-2\left(u \cdot u_{k}\right) \quad(\bmod 4) .
$$

But $u \cdot u_{k} \equiv 0(\bmod 2)$. Therefore, $2\left(u \cdot u_{k}\right) \equiv 0(\bmod 4)$.
We have shown that if $g$ is a sum of $k$ rows of $G$ then $w t(g) \equiv 0(\bmod 4)$. Hence by the principal of induction all sums of rows of $G$ have weight a multiple of 4 . This proves the theorem.

Discussion: We form two matrices from $G$ in the following way, where $G$ is the matrix given in the section 6.2.1. A matrix $B$ consists of the last 12 columns of $G$.

$$
B=\left(\begin{array}{llllllllllll}
1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\
1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\
1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\
1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\
1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\
1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\
1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\
1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1
\end{array}\right) .
$$

The rows of $B$ are linearly independent $\bmod 2$. The matrix $A$ below is $11 \times 11$ matrix formed from the last 11 elements of the first 11 rows of $G$. These are linearly dependent mod 2 . If we sum all 11 rows, we obtain $0 \bmod 2$. So this is the linearly dependent relation.

$$
A=\left(\begin{array}{lllllllllll}
1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\
0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\
1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\
1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\
1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\
1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\
0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\
1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1
\end{array}\right) .
$$

We will use the matrices $A$ and $B$ in the next theorem.
Theorem 6.7. The minimum weight or minimum distance in $G_{24}$ is 8 .
Proof. Let $g$ be a codeword in $G_{24}$. Then $g$ will have 1s in $i$ th, $j$ th and so on $k$ th positions if $g$ is the sum of $i$ th, $j$ th and so on $k$ th rows, $1 \leq i, j, k \leq 12$. It is because the first 12 columns of $G$ make an identity matrix. Therefore $w t(g) \geq s$ if $g$ is a sum of $s$ rows.

Let us suppose that $w t(g)=4$. Then $g$ is sum of at most 4 rows of $G$. Since the weight of each row in $G$ is at least 8 , it is obvious that $g$ cannot be a single row of $G$. And we proved earlier that $w t(g)=8$ if it is the sum of two rows of $G$.

If $g$ is sum of three rows then there are two cases to consider.

1) The last row of $G$ is not part of the sum of $g$.
2) The last row of $G$ is part of the sum of $g$.

In the first case, three 1 s from the 13 th column are used. Three more 1 s contribute to $g$ from the first 12 positions for each of these rows, since $w t(g)=4$. Therefore that last 11 entries of $g$ are 0 . By the above discussion, the sum of only three rows of the matrx $A$ cannot be zero. We conclude that this case is not possible.

In the second case, the last 11 entries of $g$ are obtained from two rows and the vector $(1,1, \cdots, 1)$. Since the weight of the sum of two rows of $G$ is 8 and have contribution of 2 to the weight from 13th column implies that rows of $A$ has weight 6 . When the vector $(1,1, \cdots, 1)$ is added $\bmod 2$. It change all 1 s to 0 s and 0 s to 1 s . Therefore the weight of last 11 entries of $g$ is 15 . This gives the impossibility of this case too, since $w t(g)=4$.

The last thing that we consider is that $g$ is sum of 4 rows of $G$. Then the first 12 entries of $g$ have four 1s and last 12 entries of $g$ are 0 . This is also not possible by the same argument given before.

So we have proved that there is no codeword having weight 4 . By previous theorem we know that weights are multiple of 4 and the smallest such possibly is 8 . Hence minimum weight of $G_{24}$ is 8 .

### 6.2.2 The (non- extended) Golay code $G_{23}$

If we remove the last entry of each codeword in $G_{24}$, we get the non-extended Golay code $G_{23}$. The generating matrix for this code is obtained if the last column of the generating matrix $G$ for $G_{24}$ is removed. The generating matrix $G^{\prime}$ is given below:

$$
G^{\prime}=\left(\begin{array}{lllllllllllllllllllllll}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1
\end{array}\right) .
$$

Theorem 6.8. The non-extended Golay code $G_{23}$ is a linear [23, 12, 7] code.

Proof. Since the generating matrix $G^{\prime}$ for $G_{23}$ is obtained by deleting the last column from generating matrix $G$ for $G_{24}$, therefore each codeword will have length 23 . The set of all vectors in $G_{23}$ is linearly independent. The matrix $G^{\prime}$ contains the $12 \times 12$ identity matrix therefore the rows of $G^{\prime}$ are linearly independent as it was in the case of $G$. Because of the linear independence it spans a 12 -dimensional vector space. A codeword $g^{\prime}$ in $G_{23}$ can be obtained from a codeword $g$ in $G_{24}$ by deleting the last entry of $g$. Since $w t(g) \geq 8$ and $g^{\prime}$ has one less entry than $g$ so $w t\left(g^{\prime}\right) \geq 7$.

### 6.3 Ternary Golay code

Another example of a perfect code is the ternary Golay code. Ternary Golay codes are also of two types, known as ternary Golay code and the extended Golay code. The ternary Golay code is a perfect $[11,6,5]$ ternary linear code. The extended ternary Golay code is obtained if we add a zero check sum check digit to the codewords of [11, 6,5] code.

The generating matrix $G$ for the ternary Golay code [11, 6, 5] is given by

$$
G=\left(\begin{array}{lllllllllll}
1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 2 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 2 & 2 \\
0 & 0 & 0 & 1 & 0 & 0 & 2 & 1 & 0 & 1 & 2 \\
0 & 0 & 0 & 0 & 1 & 0 & 2 & 2 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 1 & 2 & 2 & 1 & 0
\end{array}\right)
$$

All the entries of the above matrix are integers mod 3. The first six columns are the identity matrix of order 6 . The last five entries of the first row are simply 1 s . The last five entries of the second row are obtained from $\mathbb{Z}_{3}$ in the order $0,1,2,2,1$. The last five entries of last four rows are obtained by cyclically permuting the last five elements in the second row.

The extended ternary Golay code is obtained by adding a zero sum check digit to the simple ternary Golay code. The parameters for extended ternary Golay code are [12, 6, 6]. The generating matrix will look like

$$
G=\left(\begin{array}{llllllllllll}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 2 & 2 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 2 & 2 \\
0 & 0 & 0 & 1 & 0 & 0 & 1 & 2 & 1 & 0 & 1 & 2 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 & 2 & 2 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 2 & 2 & 1 & 0
\end{array}\right) .
$$

### 6.4 Reed- Solomon codes (a non- perfect code)

We now talk about a non-perfect code. Reed-Solomon codes are one of the most practical codes. They have been used in spacecraft communications and in compact discs.

Let $\mathbb{F}$ be a finite field having $q$ elements and $n=q-1$. We know that every finite field has a primitive root (see chapter 4). Let $\alpha$ be a primitive roof of $\mathbb{F}$ and choose $d$ such that $1 \leq d<n$ and let

$$
g(x)=(x-\alpha)\left(x-\alpha^{2}\right) \cdots\left(x-\alpha^{d-1}\right) .
$$

Then $g(x)$ generates code $C$ of length $n$ over the finite field $\mathbb{F}$. This generated code is called Reed-Solomon code.

Now we take a look at an example and apply the Hamming bound on the parameters of the code.

Example 6.6. Consider the field of seven elements i.e $\mathbb{F}=\mathbb{Z}_{7}=\{0,1, \cdots, 6\}$. We have therefore $q=7$ and $n=q-1=6$. Since 3 is a prime element of $\mathbb{F}$, we choose $\alpha=3$. We construct a Reed-Solomon code having minimal distance $d=5$. We have the following generating polynomial

$$
\begin{aligned}
g(x) & =(x-3)\left(x-3^{2}\right)\left(x-3^{3}\right)\left(x-3^{4}\right) \\
& =(x-3)(x-2)(x-6)(x-4) \\
& =x^{4}-3 x^{3}+6 x^{2}-3 x-4 \\
& =x^{4}+4 x^{3}+6 x^{2}+4 x+3 .
\end{aligned}
$$

Hence the generating matrix will be

$$
G=\left(\begin{array}{cccccc}
3 & 4 & 6 & 4 & 1 & 0 \\
0 & 3 & 4 & 6 & 4 & 1
\end{array}\right)
$$

There will be $7^{2}=49$ codewords in this code. The way to construct the generating matrix $G$ from the generating polynomial $g(x)$ is discussed in [6, page 430].

Since a code is said to be perfect if it satisfies Hamming bound with equality. The Hamming bound is

$$
M \leq \frac{q^{n}}{\sum_{j=0}^{k}\binom{n}{j}(q-1)^{j}} .
$$

Since $d=5$, this yields $k=2$.

$$
\begin{gathered}
49=M \leq \frac{7^{6}}{\binom{6}{0}+\binom{6}{1} 6+\binom{6}{2} 6^{2}}, \\
49=M \leq 203.8 \approx 204 .
\end{gathered}
$$

Since the equality does not hold, wo the code considered in this example is not a perfect code.

Example 6.7. If we take $d=3$ in the previous example. Then the generating polynomial will be

$$
g(x)=(x-3)\left(x-3^{2}\right)=x^{2}+2 x+6,
$$

and the generating matrix is

$$
G=\left(\begin{array}{llllll}
6 & 2 & 1 & 0 & 0 & 0 \\
0 & 6 & 2 & 1 & 0 & 0 \\
0 & 0 & 6 & 2 & 1 & 0 \\
0 & 0 & 0 & 6 & 2 & 1
\end{array}\right)
$$

There will be $7^{4}=2401$ codewords. Applying Hamming bound we get,

$$
2401=M \leq 3197.7 \approx 3180 .
$$

Again we see that the code is not perfect.

## 7 Perfect codes in Lee metric

In this and the last chapter, we consider the codes in the Lee metric. In 1950s Lee metric was described for the first time by Lee. We begin with the definition of the Lee distance and Lee weight and see how it is different with comparison to the Hamming metric and Hamming weight.

We already have defined Hamming metric, Hamming distance and Hamming weight in chapter 3. The material in this chapter is mainly taken from [4] and [1].

## Definition 7.1. Lee distance

Let $x, y$ be two words in $\mathbb{Z}_{q}$, where $q \geq 2$ is a prime and let $n$ be a positive integer. The Lee distance between $x$ and $y$ is denoted by $d_{L}(x, y)$ and defined as:

$$
\begin{equation*}
d_{L}(x, y)=\sum_{i=1}^{n} \min \left(\left|x_{i}-y_{i}\right|, q-\left|x_{i}-y_{i}\right|\right) . \tag{7.1}
\end{equation*}
$$

Just like Hamming distance, Lee distance also satisfies the following properties of a metric. For all $x, y, z$ belonging to $\mathbb{Z}_{q}^{n}$, we have
i) $d_{L}(x, y) \geq 0$ and $d_{L}(x, y)=0$ if and only if $x=y$,
ii) $d_{L}(x, y)=d_{L}(y, x)$,
iii) $d_{L}(x, y) \leq d_{L}(x, z)+d_{L}(z, y)$.

The minimum Lee distance of a code $C$ is the minimum of the distances between any two codewords, i.e

$$
d_{L}(C)=\min d_{L}\left(c_{1}, c_{2}\right) \quad \text { where } \quad c_{1} \neq c_{2} .
$$

Example 7.1. For codewords 1342 and 1025 in $\mathbb{F}_{7}^{4}$, we have

$$
\begin{aligned}
d_{L}(1342,1025) & =\min (0,7-0)+\min (3,7-3)+\min (2,7-2)+\min (3,7-3), \\
& =0+3+2+3, \\
& =8
\end{aligned}
$$

If $q=2$ or $q=3$, then the Lee metric and the Hamming metric are the same thing. In other words we say, that they coincides.

## Definition 7.2. Lee metric

Let $\alpha \in \mathbb{Z}_{q}$. The Lee weight of $\alpha$ takes non negative integer values and is defined as

$$
w_{L}(\alpha)=|\alpha|= \begin{cases}\alpha, & \text { if } \\ q-\alpha, & 0 \leq \alpha \leq q / 2 \\ \text { otherwise }\end{cases}
$$

For simplicity we consider that $\mathbb{Z}_{q}^{+}$contains the elements in $0 \leq \alpha \leq q / 2$ and $\mathbb{Z}_{q}^{-}$ denotes the remaining elements in $\mathbb{Z}_{q} \backslash\{0\}$.

Example 7.2. We consider an example for Lee weight. Let $q=7$, then we have $\mathbb{Z}_{7}=$ $\{0,1,2,3,4,5,6\}$, and the Lee weights of the elements of $\mathbb{Z}_{7}$ are given by

$$
w_{L}(0)=0, \quad w_{L}(1)=w_{L}(6)=1, \quad w_{L}(2)=w_{L}(5)=2, \quad w_{L}(3)=w_{L}(4)=3 .
$$

Let $c=\left(c_{1} c_{2} \cdots c_{n}\right)$ be a codeword in $\mathbb{Z}_{q}^{n}$. The Lee weight of $c$ is defined as the sum of Lee weights of its digits i.e

$$
w_{L}(c)=\sum_{i=1}^{n}\left|c_{i}\right| .
$$

The Lee distance defined above can also be defined in terms of Lee weight, i.e

$$
w_{L}(x-y)=d_{L}(x, y) .
$$

We can define Hamming weight and Lee weight over the alphabet having number of elements that is not prime or a power of a prime. But, while constructing codes in Lee metric, the restriction on number of elements in alphabet is crucial. It restricts us to be with in a finite field. We know that the order of a finite field is always of the form $q^{n}$, where $q$ is a prime and $n$ is a positive integer. Also the integer $\bmod q$ form a field if and only if $q$ is prime.

When it comes to Lee metric, it is more suitable to phase modulation schemes. In this kind of channels signals are transmitted through additive gaussian noise. There is great chance that the noise will corrupt the transmitted letter into another letter of slightly different phase but not into a letter of greatly different phase.

## Definition 7.3. Lee sphere

Let $C \subseteq \mathbb{F}_{p}^{n}$. A Lee sphere of radius $k$ centered at a codeword $c$ is denoted by $B_{L}(c, k)$ and is defined as

$$
B_{L}(c, k)=\left\{x \mid d_{L}(x, c) \leq k\right\} .
$$

i.e all vectors having Lee distance $k$ or less.

Now we give the proposition that describe the size of Lee sphere. In other words it tells us that how many elements the Lee sphere of certain radius contains.

Proposition 7.1. The number of elements in a Lee sphere in $\mathbb{Z}_{q}^{n}$ having radius $t<q / 2$ is

$$
V_{L}(n, t)=\sum_{i=0}^{n} 2^{i}\binom{n}{i}\binom{t}{i}
$$

where $\binom{t}{i}=0$ if $i>t$.
Proof. Consider a subset $J \subseteq\{1,2, \cdots, n\}$ and this set contains $i$ elements. Let the set $B_{J}$ consist of all the words in $B_{L}(0, t)$ such that an entry of a word in $B_{J}$ is indexed by some $j \in J$ if and only if it is nonzero. Let $B_{J}^{+}$denote the set of all those words in $B_{J}$ whose non zero entries are all from $\mathbb{Z}_{q}^{+}$. From $t<q / 2$ it follows that

$$
\left|B_{J}\right|=2^{i} \cdot\left|B_{J}^{+}\right| .
$$

Now, the number of elements in $S_{J}^{+}$is equal to the number of words of length $i$ over positive integers such that the sum of entries in each word is at most $t$. That is

$$
\left|S_{J}^{+}\right|=|Q(t, i)|,
$$

where $Q(t, i)$ will be of the following form

$$
Q(t, i)=\left\{\left(m_{1} m_{2} \cdots m_{i}\right) \mid m_{1}, m_{2}, \cdots, m_{i} \in \mathbb{Z}^{+}, \quad \sum_{s=i}^{i} m_{s} \leq t\right\}
$$

The size of the set $Q(t, i)$ will be equal to $\binom{t}{i}$ because $\left(m_{1} m_{2} \cdots m_{i}\right)$ ranges over the elements of $Q(t, i)$ and the set

$$
\left\{m_{1}, m_{1}+m_{2}, \cdots, m_{1}+m_{2}+\cdots+m_{i}\right\}
$$

ranges over all the subsets of $\{1,2, \cdots, t\}$ of size $i$. We thus have,

$$
\begin{aligned}
V_{L}(n, t) & =\left|B_{L}(0, t)\right|=\sum_{J \subseteq\{1,2, \cdots, n\}}\left|B_{J}\right|=\sum_{J} 2^{|J|} \cdot\left|B_{J}^{+}\right| \\
& =\sum_{i=0}^{n} \sum_{J:|J|=i} 2^{i}\binom{t}{i}=\sum_{i=0}^{n} 2^{i}\binom{n}{i}\binom{t}{i}
\end{aligned}
$$

### 7.1 Sphere-packing bound in the Lee metric

The proposition 7.1 is about the size of Lee sphere and it will appear in Sphere-packing bound, therefore this proposition is useful. The following theorem is about the SpherePacking bound.

## Theorem 7.2. (Sphere-Packing bound in the Lee metric)

Let $C$ be an $(n, m)$ code of size $M>1$ over $\mathbb{Z}_{q}$ and let $t=\left\lfloor\frac{B_{L}(C)-1}{2}\right\rfloor$. Then,

$$
\begin{equation*}
M \cdot V_{L}(n, t) \leq q^{n} . \tag{7.2}
\end{equation*}
$$

Proof. We consider two Lee spheres $B_{L}\left(c_{1}\right)$ and $B_{L}\left(c_{2}\right)$ centered at two distinct codewords $c_{1}, c_{2} \in C$. The spheres of radius $t=\left\lfloor\frac{B_{L}(C)-1}{2}\right\rfloor$ centered at two distinct codewords as shown in figure 7.1 must be disjoint i.e

$$
B_{L}\left(c_{1}\right) \cap B_{L}\left(c_{2}\right)=\phi .
$$



Figure 7.1: Spheres of radius $t$ centered at distinct code codewords $c_{1}$ and $c_{2}$

Hence, total volume of these spheres given by the left hand side of equation 7.2 is

$$
M \cdot V_{L}(n, t)=\sum_{c \in C}\left|B_{L}(c, t)\right|=\left|\bigcup_{c \in C} B_{L}(c, t)\right| \leq q^{n} .
$$



Figure 7.2: Lee sphere of radius 4 in two-dimensional space.

Theorem 7.3. For any given $t$, there exists a perfect $t$-Lee-error- correcting code of block length $n=2$ over the alphabet of integers $\bmod q=2 t^{2}+2 t+1$.

Proof. We can represent the error patterns $\left(E_{0}, E_{1}\right)$ for which $\left|E_{0}\right|+\left|E_{1}\right| \leq t$ in the figure 7.2. In this figure, we take the Lee sphere of radius 4.

The error patterns will be of the following forms

1. $E_{0} \geq 0, E_{1}>0$
2. $E_{0}<0, E_{1} \geq 0$
3. $E_{0} \leq 0, E_{1}<0$
4. $E_{0}>0, E_{1} \leq 0$
5. $E_{0}=E_{1}=0$

The number of error patterns in $1,2,3$ or 4 is

$$
\sum_{i=0}^{t} i=t(t+1) / 2
$$

so the total number of error patterns of Lee weight $\leq t$ is

$$
V_{t}^{2}=1+4 t(t+1) / 2=q .
$$

Therefore, any $t$-Lee-error-correcting code which has $q$ codewords of length $n=2$ over the alphabet of integers $\bmod q=2 t^{2}+2 t+1$ must be perfect.

Corollary. If $2 t^{2}+2 t+1$ divides $q$, there exist a perfect $t$-Lee error-correcting code of length $n=2$ over the alphabet of integers mod $q$. [1]

Proof. Since $q$ divides $2 t^{2}+2 t+1$, therefore there exist $k$ such that $q=k\left(2 t^{2}+2 t+1\right)$. We may take the set of codewords $k^{2}\left(2 t^{2}+2 t+1\right)$ of the form $\left[C_{0}+i\left(2 t^{2}+2 t+1\right), C_{1}+\right.$ $\left.j\left(2 t^{2}+2 t+1\right)\right]$, where $0 \leq i<k$ and $0 \leq j<k$ and $C=\left[C_{0}, C_{1}\right]$ is a codeword in the perfect $t$-Lee-error- correcting code over the alphabet of integers $\bmod 2 t^{2}+2 t+1$.

### 7.2 Berlekamp codes

We introduce another type of codes reffered to as Berlekamp codes. The Berlekamp code with certain parameters is one of the very few perfects codes in the Lee metric.

Let $F=G F(q)$, where $q$ is a prime, be a finite field. Also let $\Phi$ be a finite extension field of $F$. Take the elements $\beta_{1}, \beta_{2}, \cdots, \beta_{n}$ from $\Phi$ such that $\beta_{i} \neq 0$, for $1 \leq i \leq n$, and $\beta_{i}+\beta_{j} \neq 0$ for all $1 \leq i<j \leq n$.

Consider the linear $[n, n-\gamma]$ code $C$ over $\Phi$ with a parity check matrix

$$
H_{\mathrm{Ber}}=\left(\begin{array}{cccc}
\beta_{1} & \beta_{2} & \cdots & \beta_{n}  \tag{7.3}\\
\beta_{1}^{3} & \beta_{2}^{3} & \cdots & \beta_{n}^{3} \\
\beta_{1}^{5} & \beta_{2}^{5} & \cdots & \beta_{n}^{5} \\
\cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot \\
\beta_{1}^{2 \gamma-1} & \beta_{2}^{2 \gamma-1} & \cdots & \beta_{n}^{2 \gamma-1}
\end{array}\right)
$$

The intersection $C \cap F^{n}$ is called a Berlekamp code. It is denoted by $C_{\text {Ber }}$.
Theorem 7.4. Let $C_{\text {Ber }} \neq 0$ be defined over $F=G F(q), q$ is a prime, be the parity check matrix 7.3. If we have $\gamma<q / 2$ then

$$
d_{L}\left(C_{B e r}\right) \geq 2 \gamma+1
$$

Example 7.3. Let $F=G F(q)$ and $\Phi=G F\left(q^{n}\right)$ where $q$ is an odd prime and $m$ a positive integer. We construct the ( $n, k$ ) Berlekamp code $C$ over $F$ by the parity-check matrix 7.3 with $\gamma=1$ and $n=\frac{1}{2}\left(p^{m}-1\right)$. We have $n-k \leq m$ for this code. We have, by using the proposition 7.1

$$
\left|C_{\mathrm{Ber}}\right| \cdot V_{L}(n, 1)=\left|C_{\mathrm{Ber}}\right| \cdot(1+2 n) \geq q^{n-m} \cdot(1+2 n)=q^{n}
$$

Since $F=G F(q)$ contains $q^{n}$ elements, so $C_{\text {Ber }}$ in this case make the whole space. therefore we can say that Berlekamp code with $\gamma=1$ and $n=\frac{1}{2}(|\Phi|-1)$ is a perfect code in the Lee metric.

The Hamming metric is used in orthogonal modulation schemes. As an alternate to the Hamming metric, the Lee metric was developed. The Lee metric basically is useful for the channels that use phase-shift key modulation. Codes in the Lee metric are also considered good for the channels with synchronization errors.

We discussed earlier that Lee metric and Hamming metric coincide over $G F(2)$ and $G F(3)$. Therefore, all perfect codes in the Hamming metric over $G F(2)$ and $G F(3)$ are also perfect in the Lee metric. The Golay codes and the Hamming codes over these fields are the examples of such codes. unfortunately, there are no more known perfect codes for the Lee metric when it comes to $q>3$. It is conjectured that for $q>3$, there are no perfect codes in the Lee metric over $\mathbb{Z}_{q}$ with code length greater than 2 and minimum Lee distance greater than 3 .

## 8 Minimal distance

In chapter 3 we had a look at $(n, M, d)$-codes, where $n$ is the length of the codeword, $M$ the number of codewords and $d$ is the distance between any two different codewords. Such a code can correct up to $k$ errors if $d \geq 2 k+1$. Since correction of errors depends on $d$, we want $d$ to be large enough to correct maximum number of errors. By adding redundant digits to codewords, (for example repetition codes) the possibility if detecting and correcting errors increases. But, on the other hand it will take more time to transmit these codes.

The information rate or code rate for a code $C$ of length $n$ is given by

$$
R=\frac{\log _{2}|C|}{n} .
$$

The information ranges between 0 and 1 , since we can assume that $1 \leq|C|=M \leq 2^{n}$. If the number of codewords $M$ is large enough as well then the information rate will also be close to 1 . Hence, we wish $M$ to be large too. This would help the bandwidth to be used efficiently. If we increase $d$ it leads to increase $n$ or decrease $M$.

In this chapter, we study the restriction of $n, M$ and $d$. We won't care about practical matters. The material in this chapter in mainly taken from [6].

### 8.1 Singleton bound

We begin with a bound on the number of codewords $M$. This bound is called the Singleton bound. This result was given by R. Singleton in 1964.

Theorem 8.1. Let C be a q-ary $(n, M, d)$ code. Then

$$
M \leq q^{n-d+1}
$$

Proof. Let $c^{\prime}=\left(a_{1}, a_{2} \cdots, a_{d}\right)$ is a word of length $d$, consisting of the first $d$ bits of $c$ $c=\left(a_{1}, \cdots, a_{n}\right)$. If $c_{1}, c_{2}$ are two codewords such that $c_{1} \neq c_{2}$, then they differ in at least $d$ positions. Since $c_{1} \neq c_{2}$ and we have obtained $c_{1}^{\prime}$ and $c_{2}^{\prime}$ by removing first $d-1$ entries from $c_{1}$ and $c_{2}$. Therefore $c_{1}^{\prime}$ and $c_{2}^{\prime}$ must be different from each other in at least one place, so $c_{1}^{\prime} \neq c_{2}^{\prime}$. The number of $c^{\prime}$ s that can be obtained in this way is $M$, the number of codewords. There will be at least $q^{n-d+1}$ such vectors $c^{\prime}$ because these vectors have total $n-d+1$ positions. This gives us $M \leq q^{n-d+1}$, which is required.

Corollary. The information or code rate for a $q$ - ary $(n, m, d)$ code is at most $1-\frac{d-1}{n}$.
From this corollary, we observe that the code rate will be small if the relative minimal distance $d / n$ is large.

A code $C$ satisfying the Singleton bound with equality is called a maximum distance separable (MDS) code. The Singleton bound can also be written as

$$
q^{d} \leq q^{n+1} / M
$$

so an MDS code has the largest possible value of $d$ for a given $n$ and $M$.

### 8.2 Gibert- Varshamov bound

Definition 8.1. Let the alphabet $\mathscr{A}$ have $q$ elements. Given $n$ and $d$ with $d \leq n$,the largest $M$ such that an $(n, M, d)$ code exists is denoted by $A_{q}(n, d)$.

We can always find at least one $(n, M, d)$ code: Fix an element $a_{0}$ of $\mathscr{A}$. Let $C$ be the set of all vectors $\left(a, a, \cdots a, a_{0}, \cdots, a_{0}\right)$ (with $d$ copies of $a$ and $n-d$ copies of $a_{0}$ ) with $a \in \mathscr{A}$. There are $q$ such vectors, and they are at distance $d$ from each other, so we have an $(n, q, d)$ code. This gives the trivial lower bound $A_{q}(n, d) \geq q$.

We know discuss amother lower bound which is known as Gilbert- Varshamov bound.
Theorem 8.2. Given $n, d$ with $n \geq d$ for $A_{q}(n, d)$, there exists a $q$-ary $(n, M, d)$ code with

$$
M \geq \frac{q^{n}}{\sum_{j=0}^{d-1}\binom{n}{j}(q-1)^{j}} .
$$

This means that

$$
A_{q}(n, d) \geq \frac{q^{n}}{\sum_{j=0}^{d-1}\binom{n}{j}(q-1)^{j}} .
$$

[6, page 405]
Proof. Consider a vector $c_{1}$ and remove all vectors in $\mathscr{A}^{n}$ that are inside a Hamming sphere of radius $d-1$ about $c_{1}$. Now take another vector $c_{2}$ from the remaining vectors in $A^{n}$. Since we have removed all vectors which are at distance $d-1$ from $c_{1}, d\left(c_{2}, c_{1}\right) \geq d$. Now we do the same thing for $c_{2}$. Remove all the vectors having distance at most $d-1$ from $c_{2}$ and $c_{1}$ and choose another vector $c_{3}$ from the remaining ones. Since we have removed vector in both the cases so we will not have $d\left(c_{3}, c_{1}\right) \leq d-1$ or $d\left(c_{3}, c_{2}\right) \leq d-1$. We continue the process of choosing vectors and remove them from the remaining ones.

For one chosen vector,

$$
\sum_{j=0}^{d-1}\binom{n}{j}(q-1)^{j}
$$

vectors will be removed from the space. Therefore, for $M$ such chosen vectors the total number of removed vectors will be

$$
M \sum_{j=0}^{d-1}\binom{n}{j}(q-1)^{j} .
$$

By lemma 7.3, the total number of vectors for $q$-ary $(n, m, d)$ code is $q^{n}$. Hence the total number of vectors which can be removed cannot exceed $q^{n}$ i.e

$$
M \sum_{j=0}^{d-1}\binom{n}{j}(q-1)^{j} \geq q^{n} .
$$

This proves the result.
Therefore, there exists a code $\left\{c_{1}, \cdots, c_{M}\right\}$ with $M$ codedwords satisfying the preceding inequality. Since $A_{q}(n, d)$ is the largest such $M$, it also satisfies the inequality.

There is one minor technicality that should be mentioned. We actually have constructed an $(n, M, e)$ code with $e \geq d$. However, by modifying a few entries of $c_{2}$ if necessary, we can arrange that $d\left(c_{2}, c_{1}\right)=d$. The remaining vectors are then chosen by the above procedure. This produces a code where the minimal distance is exactly $d$.

Example 8.1. We again consider the example 6.6 given in chapter 6 . We apply both Singleton and Gilbert-Varshomov bound on the parameters and see whether it satisfies these bounds.

The generating matrix given in example 6.6 is

$$
G=\left(\begin{array}{llllll}
3 & 4 & 6 & 4 & 1 & 0 \\
0 & 3 & 4 & 6 & 4 & 1
\end{array}\right) .
$$

We list all 49 codewords below.

$$
\begin{gathered}
\{\{0,0,0,0,0,0\},\{0,3,4,6,4,1\},\{0,6,1,5,1,2\},\{0,2,5,4,5,3\},\{0,5,2,3,2,4\}, \\
\{0,1,6,2,6,5\},\{0,4,3,1,3,6\},\{3,4,6,4,1,0\},\{3,0,3,3,5,1\},\{3,3,0,2,2,2\}, \\
\{3,6,4,1,6,3\},\{3,2,1,0,3,4\},\{3,5,5,6,0,5\},\{3,1,2,5,4,6\},\{6,1,5,1,2,0\}, \\
\{6,4,2,0,6,1\},\{6,0,6,6,3,2\},\{6,3,3,5,0,3\},\{6,6,0,4,4,4\},\{6,2,4,3,1,5\}, \\
\{6,5,1,2,5,6\},\{2,5,4,5,3,0\},\{2,1,1,4,0,1\},\{2,4,5,3,4,2\},\{2,0,2,2,1,3\}, \\
\{2,3,6,1,5,4\},\{2,6,3,0,2,5\},\{2,2,0,6,6,6\},\{5,2,3,2,4,0\},\{5,5,0,1,1,1\}, \\
\{5,1,4,0,5,2\},\{5,4,1,6,2,3\},\{5,0,5,5,6,4\},\{5,3,2,4,3,5\},\{5,6,6,3,0,6\}, \\
\{1,6,2,6,5,0\},\{1,2,6,5,2,1\},\{1,5,3,4,6,2\},\{1,1,0,3,3,3\},\{1,4,4,2,0,4\}, \\
\{1,0,1,1,4,5\},\{1,3,5,0,1,6\},\{4,3,1,3,6,0\},\{4,6,5,2,3,1\},\{4,2,2,1,0,2\}, \\
\{4,5,6,0,4,3\},\{4,1,3,6,1,4\},\{4,4,0,5,5,5\},\{4,0,4,4,2,6\}\}
\end{gathered}
$$

We see that the minimal distance is $d=5$. It is also given that $n=6$ and $q=7$. By applying the singleton bound, we get

$$
M \leq 7^{6-5+1},
$$

which implies that

$$
M \leq 7^{2}=49
$$

Which shows that maximum number of codewords is 49 . Hence the Singleton bound is satisfied with equality.

Now we apply the Gilbert- Varshomov bound on the same parameters. The GilberVarshomov bound gives

$$
M \geq \frac{7^{6}}{\binom{6}{0}+\binom{6}{1} 6+\binom{6}{2} 6^{2}+\binom{6}{3} 6^{3}+\binom{6}{4} 6^{4}}=4.83
$$

this implies $M \approx 5$.
Example 8.2. We consider the example 8.1 again but with the minimum distance $d=3$. We first apply the Singleton bound. We have,

$$
M \leq 7^{6-3+1}
$$

which implies

$$
M \leq 7^{4}=2401
$$

Now we apply Gilbert-Varshamov bound, this gives

$$
M \geq \frac{7^{6}}{\binom{6}{0}+\binom{6}{1} 6+\binom{6}{2} 6^{2}}=203.89 \approx 204
$$

implies that $M \geq 204$.

Observation:- From examples 8.1 and 8.2, we observe that for small minimum distance we have enumerous increase in the number of codewords for both Singleton and Gilbert-Varshamov bounds. The codewords will be more dense because the minimum distance between them is small.

On the other hand, if the minimum distance is increased we have less number of codewords for both bounds. It may be noted here that Singleton and Gilbert-Varshamov bounds are upper and lower bounds respectively for the number of codewords in a code.

## 9 Minimal distance in Lee Metric

In this chapter, we will establish conditions on the parameters of codes in Lee metric. We develop some important bounds on the parameters of codes in the Lee metric. The bounds which will be discussed are the Gilbert-Varshamov bound and Plotkin's low-rate average distance bound. The material in this chapter is mainly taken from [1] and [4].

### 9.1 Gilbert-Varshamov bound in the Lee metric

In the next theorem we discuss Gilber-Varshamov bound. This bound is in the Lee metric over prime fields of odd size.

Theorem 9.1. (Gilbert-Varshamov bound in the Lee metric)
Let $p$ be an odd prime and let $n, k$, and $d$ be positive integers such that

$$
\begin{equation*}
\frac{p^{n-k+1}-1}{p-1}>\frac{V_{L}(n, d-1)-1}{2} . \tag{9.1}
\end{equation*}
$$

Then there exists a linear $(n, k)$ code $C$ over $F=G F(p)$ with $d_{L}(C) \leq d$.
Proof. Initially we choose $C_{0}=\{0\}$, we use iterations method to construct a sequence of codes $C_{1}, C_{2}, \cdots, C_{k}$ such that each code $C_{i}$ in the sequence is linear $(n, i)$ code over $F$ and has minimum Lee distance $d$ i.e $d_{L}\left(C_{i}\right) \geq d$.

We suppose that for some $i \leq k$, the codes $C_{1}, C_{2}, \cdots, C_{i-1}$ are constructed. Let the set $C_{i-1}(e)$ is given by

$$
C_{i-1}(e)=\left\{c+a \cdot e: c \in C_{i-1}, a \in F^{*}\right\},
$$

where $e$ is any word belonging to $F^{n} \backslash C_{i-1}$. Note that the sets $C_{i-1}(e)$ are distinct and must be disjoint because $C_{i-1}(e)$ is a union of $p-1$ distinct cosets of $C_{i-1}$ in $F^{n}$. For all $e \in F^{n} \backslash C_{i-1}$, the number of distinct sets $C_{i-1}(e)$ that can be created is given by

$$
\begin{equation*}
\frac{1}{p-1}\left(\frac{p^{n}}{\left|C_{i-1}\right|}-1\right)=\frac{p^{n-i+1}-1}{p-1}>\frac{V_{L}(n, d-1)-1}{2} \tag{9.2}
\end{equation*}
$$

The right-hand side of (9.2) shows the size of the set $B_{L}^{*}(0, d-1)$, which is the set of all nonzero words in $B_{L}(0, d-1)$ and has leading nonzero entry in $F^{+}$. As (9.2) shows that the number of distinct sets $C_{i-1}(e)$ is larger than $\frac{V_{L}(n, d-1)-1}{2}$, it follows that there exist at least one set $C_{i-1}\left(e_{0}\right)$ such that $C_{i-1}\left(e_{0}\right) \cap B_{L}^{*}(0, d-1)=\phi$.

Moreover

$$
c \in C_{i-1}\left(e_{0}\right) \Longleftrightarrow-c \in C_{i-1}\left(e_{0}\right),
$$

and

$$
0 \notin C_{i-1}\left(e_{0}\right) .
$$

So, it follows that

$$
C_{i-1}\left(e_{0}\right) \cap B_{L}(0, d-1)=\phi .
$$

Thus the union of $C_{i-1}$ and $C_{i-1}\left(e_{0}\right)$ form a linear $(n, i)$ code $C_{i}$, and this code has minimum Lee distance $d$.

Example 9.1. We construct a linear (4,2)-code over the field GF (5). All possible codewords by using the generating matrix

$$
G=\left(\begin{array}{llll}
2 & 3 & 1 & 0 \\
0 & 2 & 3 & 1
\end{array}\right)
$$

are listed below

$$
\begin{gathered}
\{\{0,0,0,0\},\{0,2,3,1\},\{0,4,1,2\},\{0,1,4,3\},\{0,3,2,4\}, \\
\{2,3,1,0\},\{2,0,4,1\},\{2,2,2,2\},\{2,4,0,3\},\{2,1,3,4\}, \\
\{4,1,2,0\},\{4,3,0,1\},\{4,0,3,2\},\{4,2,1,3\},\{4,4,4,4\}, \\
\{1,4,3,0\},\{1,1,1,1\},\{1,3,4,2\},\{1,0,2,3\},\{1,2,0,4\}, \\
\{3,2,4,0\},\{3,4,2,1\},\{3,1,0,2\},\{3,3,3,3\},\{3,0,1,4\}\} .
\end{gathered}
$$

First we calculate the minimal Lee distance by using the formula (7.1). For instance we calculate the Lee distance between first two codewords.

$$
\begin{aligned}
d_{L}(0000,0231)= & \min (|0-0|, 5-|0-0|)+\min (|0-2|, 5-|0-2|) \\
& +\min (|0-3|, 5-|0-3|)+\min (|0-1|, 5-|0-1|) \\
= & \min (0,5-0)+\min (2,5-2)+\min (3,5-3)+\min (1,5-1) \\
= & 0+2+2+1 \\
= & 5
\end{aligned}
$$

Similarly by calculating the Lee distance between all codewords, we see that the minimal Lee distance is $d_{L}=4$.

Now we apply the Gilbert-Varshamov bound in the Lee metric. Here we have $p=q=$ $5, n=4$ and $k=2$. To apply the expression 9.1 we need to calculate volume $V_{L}$ of the Lee sphere and the formula for the volume is given in the proposition 7.1. To calculate volume me must have $t<q / 2$ according to the proposition 7.1 but in this example we have $t=d_{L}-1=4-1=3$ which does not satisfy $t<q / 2$.

We consider another example with the different generating matrix and smaller minimal distance which satisfy the condition $t<q / 2$.

Example 9.2. We use the generating matrix

$$
G=\left(\begin{array}{llll}
1 & 0 & 3 & 4 \\
0 & 1 & 1 & 1
\end{array}\right)
$$

to construct a linear $(4,2)$-code over the field $G F(5)$. All possible codewords are given as

$$
\begin{aligned}
& \{\{0,0,0,0\},\{0,1,1,1\},\{0,2,2,2\},\{0,3,3,3\},\{0,4,4,4\}, \\
& \{1,0,3,4\},\{1,1,4,0\},\{1,2,0,1\},\{1,3,1,2\},\{1,4,2,3\}, \\
& \{2,0,1,3\},\{2,1,2,4\},\{2,2,3,0\},\{2,3,4,1\},\{2,4,0,2\}, \\
& \{3,0,4,2\},\{3,1,0,3\},\{3,2,1,4\},\{3,3,2,0\},\{3,4,3,1\}, \\
& \{4,0,2,1\},\{4,1,3,2\},\{4,2,4,3\},\{4,3,0,4\},\{4,4,1,0\}\} .
\end{aligned}
$$

The minimal Lee distance for this code is $d_{L}=2$ and there $t=d_{L}-1=1$. This value of $t$ satisfies the condition $t<q / 2$ given by the proposition 7.1. We calculate the volume of the Lee sphere and it is $V_{L}=9$. Here we have $p=q=5, n=4$ and $k=2$. Now we apply the Gilbert-Varshamov bound (9.1)

$$
\frac{5^{4-2+1}-1}{5-1}>\frac{9-1}{2} \Longrightarrow 31>4
$$

Hence the Gilbert-Varshamov bound is satisfied.

### 9.2 Plotkin's low-rate average distance bound

The Plotkin's bound for the Lee metric was first obtained by Graham and Wyner (1968). There is an observation according to Plotkin about the minimum distance that the average distance between all pairs of different codewords always exceeds the minimum distance between any two codewords. This observation forms basis for a bound on the minimum distance. This bound is taken from [1].

Suppose that the code $C$ contains $C^{(1)}, C^{(2)}, \cdots C^{(k)}$ codewords, where

$$
C^{(i)}=\left[C_{0}^{(i)}, C_{1}^{(i)}, \cdots, C_{n-1}^{(i)}\right],
$$

where $C_{0}^{(i)}, C_{1}^{(i)}, \cdots, C_{n-1}^{(i)}$ are the digits of $C^{(i)}$. The total distance $\left(d_{\mathrm{t}}\right)$ between pairs or codewords is given by

$$
\begin{aligned}
d_{\mathrm{t}} & =\sum_{i=1}^{k} \sum_{j=1}^{k} d\left(C^{(i)}, C^{(j)}\right) \\
& =\sum_{i=1}^{k} \sum_{j=1}^{k} \sum_{m=0}^{n-1} d\left(C_{m}^{(i)}, C_{m}^{(j)}\right) \\
& =\sum_{m=0}^{n-1} \sum_{i=1}^{k} \sum_{j=1}^{k} d\left(C_{m}^{(i)}, C_{m}^{(j)}\right)
\end{aligned}
$$

For $I=0,1, \cdots, q-1$, we let $J_{I}^{(m)}$ denote the number of occurances of the $I$ th channel letter among the letters $C_{m}^{(1)}, C_{m}^{(2)} \cdots, C_{m}^{(k)}$. The probability vector

$$
\mathrm{P}^{(m)}=\left[J_{0}^{(m)}, J_{1}^{(m)}, \cdots, J_{q-1}^{(m)}\right] / K,
$$

then denotes the composition of the $m$ th letter of a codeword selected from the code at random. We then have

$$
\begin{aligned}
\sum_{i=1}^{k} \sum_{j=1}^{k} d\left(C_{m}^{(i)}, C_{m}^{(j)}\right) & =K^{2} \sum_{i=0}^{q-1} \sum_{j=0}^{q-1} P_{i}^{m} P_{j}^{m} D_{i, j} \\
& =K^{2} P^{m} D P^{(m) t}
\end{aligned}
$$

where $D_{i, j}$ is the $i, j$ th entry of the matrix $D$. It is the distance between the $i$ th channel input letter and the $j$ th channel input letter. For example for $q=6$ we have the following matrix for the Lee metric

$$
D=\left(\begin{array}{llllll}
0 & 1 & 2 & 3 & 2 & 1 \\
1 & 0 & 1 & 2 & 3 & 2 \\
2 & 1 & 0 & 1 & 2 & 3 \\
3 & 2 & 1 & 0 & 1 & 2 \\
2 & 3 & 2 & 1 & 0 & 1 \\
1 & 2 & 3 & 2 & 1 & 0
\end{array}\right)
$$

The total distance is

$$
d_{t}=\sum_{m=0}^{n-1} P^{m} D P^{(m) t} .
$$

This total distance includes the sum over $K^{2}$ pairs of codewords. $K$ of these pairs consist of the same codewords, and $K^{2}-K$ pairs consist of different codewords. The average
distance among the $K(K-1)$ pairs of different codewords is therefore given by

$$
d_{\mathrm{av}}=\frac{K^{2}}{K(K-1)} \sum_{m=0}^{n-1} P^{(m)} D P^{(m) t} \leq \frac{n}{1-K^{-1}} \mathrm{PDP}^{t} .
$$

Let $P$ denote the vector, for which $P^{(m)} D P^{(m) t}$ is the largest. Since the minimum distance cannot exceed the average distance,

$$
d_{\min } \leq \frac{n}{1-K^{-1}} \mathrm{PDP}^{t} .
$$

## 10 Conclusion

In this thesis, we studied and analyze the perfect codes and the minimal distance of a code in general and for the Lee metric in particular. First, we tried to classify perfect codes in general and studied their properties. Then, we studied the bounds on minimal distance and number of code words in a code. A few bounds are discussed with examples.

In case of the Lee metric, first the perfect codes are classified. The perfect codes in the Lee metric are very few. There are perfect codes in the Lee metric over $G F(2)$ and $G F(3)$. According to [1] and [4], it is conjectured that there are no more perfect codes in the Lee metric for $G F(q), q>3$, with code length greater than 2 and minimum distance greater than 3 .

Finally, bounds in the Lee metric are given in the end. These bounds are related to the minimal distance of a code in the Lee metric.

## References

[1] Elwyan R. Berlekamp. Algebraic Coding Theory. McGraw-Hill Book Company, Second edition, 1968.
[2] John B. Fraleigh. A First Course in Abstract Algebra. Pearson education, Seventh edition, 2003.
[3] D.R. Hankerson, D.G. Hoffman, D.A. Leonard, C.C. Linder, K.T. Phelps, C.A. Rodger, and J.R Wall. Coding theory and Cryptography The Essentials. Marcel Dekker, INC, Second edition, 2000.
[4] Ron M. Roth. Introduction to Coding Theory. Cambridge University Press, Second edition, 2006.
[5] Vivek Sahai and Vikas Bist. Linear Algebra. Alpha Science International Ltd, 2002.
[6] Wade Trappe and Lawrence C. Washingtion. Introduction to Cryptography with Coding Theory. Pearson education, Second edition, 2002.
[7] Lekh R. Vermani. Elements of Algebraic Coding Theory. National Bureau of standards, First edition, 1996.
[8] Zhe-Xian Wan. Introduction to Abstract and Linear Algebra. Chartwell-Bratt Ltd, 1992.

Linnæus University
School of Computer Science, Physics and Mathematics
SE-351 95 Växjö / SE-391 82 Kalmar
Tel +46-772-28 8000
dfm@lnu.se
Lnu.se/dfm

