Definition 1.4 – fields

A set F having *at least* two elements with two compositions, say addition and multiplication defined on it, is called a **field** if:

(i) F, w.r.t. the additive and multiplicative composition, is a commutative ring with identity; and
(ii) every non-zero element of F is invertible w.r.t. multiplication.

It is then immediate that F^*, the set of all non-zero elements of F, is an Abelian group w.r.t. multiplication. Observe that $\mathbb{Q}$, the set of all rational numbers, w.r.t. the addition and multiplication of the number system, is a field, and one having an infinite number of elements. However, there do exist fields having only a finite number of elements; these are called finite or **Galois fields**.

If p is a prime integer, consider the set $F_p = \{0, 1, 2, \ldots, p-1\}$ of p elements in which addition $\oplus$ and multiplication $\bigcirc$ are defined modulo p. Explicitly, for a, $b \in F_p$, $a \oplus b = c$ and $a \bigcirc b = d$ where c, d are respectively the least non-negative remainders when the sum $a + b$ and product ab of the integers a, b are divided by p. When there is no danger of confusion, we also write $a + b$ for $a \oplus b$ and ab (or $a \times b$) for $a \bigcirc b$. For $p = 2$, we denote the field F_2 by $\mathbb{B}$ ('b' for binary). Thus $\mathbb{B} = \{0, 1\}$ with addition and multiplication defined by $0 + 0 = 0$, $1 + 0 = 0 + 1 = 1, 1 + 1 = 0, 0 \times 0 = 1 \times 0 = 0 \times 1 = 0, 1 \times 1 = 1$. Throughout this chapter, we are concerned with the field $\mathbb{B}$ of two elements. (There also exist other finite fields and we will study these later.)

Let $\mathbb{B}^n$, where n is a positive integer, denote the set of all ordered n-tuples or sequences of length n with entries belonging to the field $\mathbb{B}$. Define sum of two sequences of length n component-wise, i.e. if $a = a_1 \text{---} a_n$, $b = b_1 \text{---} b_n$, then $a + b = c_1 \text{---} c_n$, where $c_i \in \mathbb{B}$ are defined by $c_i = a_i + b_i$, $1 \leq i \leq n$. Observe that with this composition $\mathbb{B}^n$ becomes an Abelian group. The zero sequence (the sequence of length n with every component or entry zero) is the identity of $\mathbb{B}^n$ and every element of $\mathbb{B}^n$ is its own inverse.

Definition 1.5 – code words

A **binary block** (m, n)**-code** consists of an **encoding function** E: $\mathbb{B}^m \rightarrow \mathbb{B}^n$ and a **decoding function** D: $\mathbb{B}^n \rightarrow \mathbb{B}^m$. The elements of $\operatorname{Im} E$ (image of E) are called **code words**.

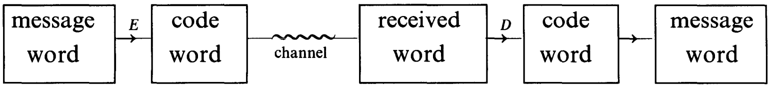
One of the earliest examples (although outdated) of codes that we might come across in our daily life is **Morse code**. To send a message telegraphically, the message to be conveyed is first written as English text. In the Morse code, each character of the English language is identified by a sequence of dots and dashes. Using this coding system, the message is converted into a sequence of dots and dashes and there results a code word which is then transmitted through the machine to another (say in Delhi). The operator working on the machine in Delhi receives a sequence of dots and dashes which is then translated, using the inverse process, back into English text. Some of the dots and dashes transmitted may have been wrongly received in Delhi due to disturbance in the channel. The operator in Delhi will then have to depend on his vocabulary in English to decipher the message as accurately as possible.

Another example in daily life is provided by telephone. When we speak into the microphone, the speech is converted into an electric wave. To ensure that this electric wave does not become very weak when it travels through the wire over a long distance or due to resistance of the wire, this wave is superimposed by a (modulating) electric wave (the net result becoming a code word). On the receiving end, the added wave is filtered out (i.e. the code word is decoded) and the rest of the electric wave which represents the original speech is then converted into sound wave and received by the receiver.

In coding theory, we are concerned with devising methods of encoding and decoding so that the errors which occur due to disturbance in channel, if not altogether eliminated, are minimized. One of the assumptions about the channel is that it does not increase or decrease the length of the sequence that passes through it, although it may garble a few dots and dashes (or 0s and 1s). The second assumption is that the probability of a 0 being garbled (or not) is the same as that of 1 being garbled (or not), i.e. it is a binary symmetric channel.

Since we are mostly concerned with binary codes, we suppress the word binary and, unless stated otherwise by a code, we shall mean a binary code throughout this chapter.

Also the decoding function D is not, in general, defined from $\mathbb{B}^n \to \mathbb{B}^m$ but is defined from $\mathbb{B}^n \to C$, where C is the set of all code words. The process/problem of coding theory may be pictorially depicted as:



Definition 1.6 – distance function

If a, $b \in \mathbb{B}^n$, we define the **distance** $d(a,b)$ between a and b by

$$d(a,b) = \sum_{i=1}^{n} x_i \qquad \begin{cases} x_i = 0 & \text{if } a_i = b_i \\ x_i = 1 & \text{if } a_i \neq b_i \end{cases}$$

where $a = a_1 a_2 \text{---} a_n$ and $b = b_1 b_2 \text{---} b_n$.

For example:

(i) if $a = 10011011$ and $b = 11001101$, then $a_1 = b_1$, $a_2 \neq b_2$, $a_3 = b_3$, $a_4 \neq b_4$, $a_5 = b_5$, $a_6 \neq b_6$, $a_7 \neq b_7$, $a_8 = b_8$ and so $d(a,b) = 4$

(ii) if $a = 111001$ and $b = 101010$, then $d(a,b) = 3$

Observe that $d(a,b) = d(b,a) \forall a, b \in \mathbb{B}^n$.

Definition 1.7 – weight function

If $a \in \mathbb{B}^n$, we define the **weight** $\mathrm{wt}(a)$ of a as the number of non-zero components of the sequence a.

For example:

(i) if $a = 10011011$, then $\mathrm{wt}(a) = 5$
(ii) if $a = 11001101$, then $\mathrm{wt}(a) = 5$
(iii) if $a = 111001$, then $\mathrm{wt}(a) = 4$
(iv) if $a = 101010$, then $\mathrm{wt}(a) = 3$

Lemma 1.1

If $a, b \in \mathbb{B}^n$, then $d(a, b) = \mathrm{wt}(a + b)$.

Proof

Let $a = a_1 a_2 \text{---} a_n$ and $b = b_1 b_2 \text{---} b_n$. For any i, $1 \leq i \leq n$, $a_i + b_i = 1$ iff $a_i \neq b_i$. Hence the pair (a_i, b_i) contributes 1 to $\mathrm{wt}(a + b)$ iff it contributes 1 to $d(a, b)$. Therefore, $d(a, b) = \mathrm{wt}(a + b)$.

Corollary

If a, b, $c \in \mathbb{B}^n$, then $d(a + c, b + c) = d(a, b)$.

Lemma 1.2

If a, b, $c \in \mathbb{B}^n$, then $d(a, b) \leq d(a, c) + d(b, c)$.

Proof

Let $a = a_1 a_2 \text{---} a_n$, $b = b_1 b_2 \text{---} b_n$ and $c = c_1 c_2 \text{---} c_n$. For any i, $1 \leq i \leq n$, define

$$d(a_i, b_i) = \begin{cases} 1 & \text{if } a_i \neq b_i \\ 0 & \text{if } a_i = b_i \end{cases}$$

Similarly define $d(a_i, c_i)$ and $d(b_i, c_i)$. Then

$$d(a, b) = \sum_{i=1}^{n} d(a_i, b_i)$$

If $a_i = b_i$, then $d(a_i, b_i) \leq d(a_i, c_i) + d(b_i, c_i)$ trivially. Suppose that $a_i \neq b_i$. Then $d(a_i, b_i) = 1$. If $a_i = c_i$, then necessarily $b_i \neq c_i$ while if $b_i = c_i$, then $a_i \neq c_i$. Thus, in either case

$$d(a_i, b_i) \leq d(a_i, c_i) + d(b_i, c_i)$$

and therefore

$$\begin{aligned} d(a, b) &= \sum_{i=1}^{n} d(a_i, b_i) \\ &\leq \sum_{i=1}^{n} d(a_i, c_i) + \sum_{i=1}^{n} d(b_i, c_i) \\ &= d(a, c) + d(b, c) \end{aligned}$$

Definition 1.8 – nearest-neighbour decoding principle

The **nearest-neighbour decoding principle** states that if a word $r \in \mathbb{B}^n$ is received and it happens to be a code word, we put $D(r) = r$. If r is not a code word, we find the distance of r from all the code words and among these distances, we find the least. Suppose that d is the least distance. Then there exists a code word a (say) such that $d(a, r) = d$. If a is the only code word with $d(a, r) = d$, we put $D(r) = a$. If there is more than one code word with distance from r equal to d, we say that there is a **decoding failure**.

In the case of binary symmetric channel (as in our case) in which all code words are equally likely to be transmitted and the single-symbol error probability is less than $\frac{1}{2}$, the principle is called the **maximum likelihood decoding principle**.

Definition 1.9 – detected and undetected errors

We say that an error vector (or word) e is **detected** by a code if $a + e$ is not a code word for any code word a. If, for some code word a, $a + e$ is again a code word, we say that the error vector (word or pattern) e goes **undetected**.

When a word of length n is transmitted and k of these entries are incorrectly received, we say that k transmission errors have occurred. By taking 1 at positions corresponding to k errors and zero everywhere else, we create a word (error word/vector) e of weight k. Also, the k non-zero positions of an error vector of weight k give a set of k errors. This gives a one-to-one correspondence between error words of weight k and the sets of k errors. We say that a set of k errors is detected if the corresponding error word of weight k is detected.

Theorem 1.1

For a code to detect all sets of k or fewer errors, it is necessary and sufficient that the minimum distance between any two code words be $k + 1$ or more.

Proof

Let C be the set of all code words (of length n) of the given code. Suppose that $\forall b, b' \in C, d(b, b') \geq k + 1$. Let $b \in C$ be transmitted and suppose that the channel introduces an error word $e = e_1 e_2 \text{---} e_n$ with

$$\mathrm{wt}(e) = \left(\sum_{i=1}^{n} e_i \right) \leq k$$

Then the received word is $b + e$ and

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

This shows that $b + e$ is not a code word. As such, we know that some transmission error has occurred and this proves that the error vector e is detected.

Conversely, suppose that the code is able to detect all sets of k or fewer errors. This means that whatever word e with $\mathrm{wt}(e) \leq k$ and code word b be given, $b+e$ is not a code word. Suppose that b, $b' \in C$ and that $d(b, b') \leq k$. Let $e = b + b'$. Then $\mathrm{wt}(e) \leq k$. Also $b + e = b + b + b' = b'$ – a code word. This proves that the error vector e goes undetected. This contradiction proves that

$$d(b, b') \geq k + 1 \forall b, b' \in C$$

Definition 1.10 – corrected errors

We say that an error word e is **corrected** by a code if the decoding function D of the code is such that $D(b+e) = b$ for every code word b. We also say that a set of k errors is **corrected** if the corresponding error word of weight k is corrected.

Theorem 1.2

For a code (D, E) to correct all sets of k or fewer errors, it is necessary that the minimum distance between code words be at least $2k+1$ (it being given that the nearest neighbour decoding principle holds).

Proof

Suppose that $b = b_1 \text{---} b_n$, $a = a_1 \text{---} a_n$ are two code words with $d(a, b) \leq 2k$. Then $\mathrm{wt}(a+b) = l \leq 2k$. We can then find words e, e' such that $\mathrm{wt}(e) \leq k$, $\mathrm{wt}(e') \leq k$ and $a + b = e + e'$. Then $a + e = b + e'$.

Case (i): $l = 2t + 1$

Then we can suppose that $\mathrm{wt}(e) = t$ and $\mathrm{wt}(e') = t + 1 \leq k$. Therefore,

$$d(a+e, a) = \mathrm{wt}(a+e+a) = \mathrm{wt}(e) < \mathrm{wt}(e') = d(b+e', b)$$

Suppose that b is the code word transmitted and the channel adds to it the error vector e'. Then

$$d(b+e', a) = d(a+e, a) < d(b+e', b)$$

and by the nearest neighbour decoding principle, $b+e'$ will be decoded into a or some other code word but never b. Thus, the error vector e' with $\mathrm{wt}(e') \leq k$ is not corrected. Hence, the distance between any two code words must be at least $2k+1$.

Case (ii): $l = 2t$

We can then suppose that $\mathrm{wt}(e) = \mathrm{wt}(e') = t$. Now, $d(a+e, a) = \mathrm{wt}(e) = t = w(e')$ $= d(b+e', b)$ so that the received word $a+e$ is equidistant from two code words a and b and either the received word $b+e'$ is decoded into a word different from b or the code is unable to decide about the transmitted word – again a contradiction.