Linear Binary Error Correcting Codes with Specified Minimum Hamming Distance

Associate prof. Dr.Abbas Fadhil Al-Hashemi
Baghdad College of Economic Sciences University

Abstract

A procedure for generating liner binary codes with minimum hamming distance specified is given. The work involves the theoretical aspect in the form of two new theorems. Thus the details are restricted to small values of minimum distance. An outline of the procedure for larger values is discussed. The method relies on the fact that a linear binary code is a subspace of the vector space of all n-tuples. The construction of the code makes use of the properties of the gray code.

Keyword: linear binary code, minimum hamming distance, and subspace of all binary n-tuples.

المستخلص

في عالمنا الحالي يتم تناقل البيانات الرقمية عبر شبكات الحاسوب بكميات هائلة. إرسال هذه البيانات يحتاج طاقة وبذلك كلف مالية عالية جداً . هنا التناقل يؤدي الى حدوث إخطاء في البيانات مما يؤدي إلى إعادة إرسالها وهدر مالي كبير .

كذلك في بعض الإستخدامات الهامة هذه الأخطاء تؤدي إلى نتائج كارثية (تحطيم بعض الأجهزة على سبيل المثال). لذلك وجب تصميم أنظمة رقمية خاصة لتصحيح الاخطاء.

هذه الأنظمة تصمم بإضافة بيانات خاصة إلى البيانات الأصلية تستخدم لتصحيح الأخطاء. هذه الإضافة أيضاً تؤدي إلى كلف مالية لذلك فأن تصميم هذه الأنظمة حاز إهتمام عدد كبير من الباحثين همهم هو إرسال أقل مايمكن من البيانات هذه المعرفة هدفها تقليل كلف الإرسال.

في هذا البحث نقدم طريقة مبتكرة تصب في تصميم أنظمة تصحيح الأخطاء ونقدم نظريتين جديدتين يساهمان في التصميم الكفوء لهذه الأنظمة. أهم صفاتها هو الحصول على أقل عدد من الرموز لقابلية تصحيح أخطاء معينة.

1. Introduction

An (n, M ,d) code may be defined [1] [2] ,and [6] of M binary vectors of length n, called words, such that any two codewords differ in at least d places. n is called the block length of the code and d is the minimum distance of the code. Since the principle of coding is such that a block of k information digits is encoded into a codeword of n digits for error correction, then $M=2^k$ for binary codes. Thus the code may also be symbolized by (n, k, d) or more usually (n, k). We shall use both of these notations.

1.1. The Coding Problem

According to the following theorem [1],[6]; "An (n, k, d) code can correct $t\leq |[\frac{d-1}{2}]$ errors (here $|[x]$ denotes the greatest integer$\leq x$)"; efficient code need to have small n (for speed), a large d(to correct many errors). These are conflicting aims. The coding theory problem is stated as follows [1] [13]:

"Given n and d, to find a code with the greatest possible M (alternatively, given n and M, to find a code with the greatest d)".

In this work we deal with the problem of obtaining linear binary codes, (i.e. n and k) for specified d. This is also emphasized by Macwilliams and Sloane [2], and Brower, et al [3] who state that “the function A (n, d) defined as the maximum number of codewords in a binary code of block length n and minimum distance at least d, is of considerable interest in coding theory”.

1.2. Related Work

The above problem has been dealt with by many. But the most notable is the early work of Plotkin [4]. His work establishes upper bounds for the number of n-digit points in codes of minimum distance d with certain properties. He also gives the systematic construction of codes for given n, d. But the codes that contain the greatest possible numbers of points are obtained only for some n, d. The method given is involved and implementation is not easy.

The work of Plotkin is also discussed by Quistorff [5] where the Plotkin upper bound on the maximal cardinality of a code with minimum distance at least d is applied to q-ary codes preserved with the Hamming metric as coincident. The Plotkin bound is given by $A(n, d) \leq 2m \leq 2d/2d-n$, if $2d > n$ where A (n, d) is the number of code points and m€N.

Macwilliams and Sloane [2] gave the equivalent bound $A (n, d) \leq 2$ |[d/2d-n] and Berlekamp gave the bound $Aq (n, d) \leq \frac{dq}{dq-n}$ if $dq > n$ for a q-ary code, depending on the Plotkin bound above.

Ward [7] discusses the problem of designing codes with guaranteed minimum distance, and gives as a consequence the Gilbert-Varshamov bounds, and the Plotkin upper bounds, with their derived asymptotic bounds as well as the Griesmer bound. [27] These bounds result from dealing with the problem. They are all derived by using rather complex algebraic techniques and do not give exact results in most cases. Codes which satisfy the equality of the bounds are called perfect codes. The above problem and related bounds and their importance are also indicated by Hall [8].

The algorithmic part of the problem of finding all codewords in a C within a Hamming distance d is discussed by Sudan [9]. The aim is to apply the result to bounded distance decoding. Spielman [10] presented the first known code that is encodable in linear time. However Vardy [11] suggest that in general one cannot compute d in polynomial time.

Spanning a code by its minimum-weigh vectors (a problem similar to finding a code C for a certain d) is discussed for specific codes by Ding and Key [12]. The similar problem for codes constructed from conference matrices is discussed by Gulliver and Harada [14], and for the perfect codes by Etzion and Vardy [15]. For self dual codes construction given a certain minimum weigh (minimum distance) bound is investigated by Sloane and Thompson [16], by Bachoc [17], [26],and[28].

An important related problem is the construction of constant weight codes. The maximum number of codewords in a binary code of block length n and constant weight w, is required for a give bound on the minimum distance d. This is investigated by Nguyen, Gyorfi, and Massey [18].

As noted above the error correction capability of a code depends on the minimum distance. A great deal of work exists related to bounds on the minimum distance for example [19]. [20],[21], and [30]. Finding specific codewords with specific weight has also been dealt with, for example [22], [23], [29].

1.3. Present Work

The work presented in essence deals with the problem of finding the minimum Hamming distance. As pointed out above, this is an important coding problem and most existing methods depend on involved mathematical tools. Thus making the algorithmic implementation rather complicated. Here we give new result which is rather straightforward and clean. This is achieved by dealing with the problem from basic principle, which of developing codes as a subspace of the binary vectors space of all binary n-tuples. Combinations of the vectors in this vector space are needed for the construction of the codes as well as establishing the subspace which facilitates the generation of groups of such combination. The construction of the codes use the well known gray code. The codes obtained are optimum i.e. for a given (n, d) the maximum k (information digits) are obtained.

2. The X(n) blocks

We give a theorem which deal with the generation of what we call the x(n) block defined below. These are obtained from the vector space of all possible n-tuples. The theorem is a general result and is used in investigating as a subspace the generated code.

2.1 The x(n) block

It two binary n-tuples say u and v are added mod. 2,then,

$u + v = z \quad \longrightarrow \quad u + z = v \quad \longrightarrow \quad v + z = u$ ———(1)

Considering all possible combinations of a binary n-tuple then all possible sets (u,v,z) are obtained. If we consider the element u then all possible sets having u as a common element constitute a block of such sets in each of which u appears for example having 2 as the common element and using decimal notation we have for n=4,

(2, 4, 6)
(2, 5, 7)
(2, 8, 10)
(2, 9, 11)
(2, 12, 14)
(2, 13, 15)

We shall refer to such blocks as 2(4) for this example and in general x(n), where x is the element u and n the size of the binary tuple.

Theorem (1)

The x(n) blocks are given by

(i) For $x=2^i$, $i=0,1,2,\ldots,r$
Such that $r=m \mid x=1$, and
$M= |[2^n -2|2x]$. ——— (2)

The sets generating the blocks are

$(x, v_x + j_x , v_x + k_x)$ ——— (3)

where, $v_x =2ux$, $u=1,2,\ldots, m$ ——— (4)

$j_x =0, \ldots, x-1$ ———(5)

and $k_x= j_x +x$ ———(6)

(ii) For x even $\neq 2^i$, i=0, 1, …, r

The sets generating the blocks are (x, $v_x + j_x$, $v_x + k_x$) where with

$$K=\begin{bmatrix} k_1 \\ k_i \\ k_l \end{bmatrix} \;,\; k_i = k_{x-a} \;, l = a \;, \qquad \text{———} (7)$$

$$A=\begin{bmatrix} A_1 \\ A_i \\ A_l \end{bmatrix} \;, A_i = (-1)^{i+1} B \;, i = 1, \ldots b \; with \;\; b = |j_x| \,/{}^{*} a \;,$$

$$B = \begin{bmatrix} b_1 \\ b_i \\ b_l \end{bmatrix} , b_i = a \,,\; l = a, \qquad \text{———} (8)$$

$$\text{and} \;\; T = \begin{bmatrix} t_1 \\ t_i \\ t_l \end{bmatrix} = K + A. \qquad \text{———} (9)$$

the sets are such that.

$$v_x = v_{x-a} \quad , \quad j_x = j_{x-a} \;, \text{ and } \; k_x = t_i \,, i = 1, \ldots, a \qquad \text{—} (10)$$

Where a is the least value taken from $a = 2^{z-1}$, z=2, 3, …., m such that x/a is an odd number ———(11)

(iii) For x odd, $x \geq 3$

The set generating the blocks are

$(x, v_x + j_x \;\;, \; v_x + k_x)$ ———(12)

Such that: $v_x = v_x - 1$ ———(13)

$j_x = j_x - 1$ ———(14)

$k_x = k_{x-1} + (-1)^{j_x}$ ———(15)

Proof: without loss of generality take n=4

For $x = 2^i \;, i = 0 \,, 1 \,, \ldots . , m$

Take $x = 1 \,, 2 \,, 4$ we have

2 , 3
4 , 5
.
.
14 , 15

4 , 6
5 , 7
.
9 , 13
.
13 , 15

8 , 12
.
.
.
11 , 15

Respectively ——— (16)

For $v_1 = 2 \,, v_2 = 4$, and $v_4 = 8$ we have

$^{*}|j_x|$ = number of elements in j_x

v_1, v_1+1
$v_1+2\,, v_1+3$
.
.
v_1+12, v_1+13

v_2, v_2+2
$v_2+1\,, v_2+3$
.
.
v_2+9, v_2+11

and

v_4, v_4+4
$v_4+1\,, v_4+5$
.
.
v_4+3, v_4+7

Respectively ———— (17)

Similar pattern is obtained for x=8, 18, … It is clear from above that;

(i) Extension to n >4 is obvious.

(ii) For $j_x = 0, \dots, x-1$ the second element in the block is $v_x + j_x$. The third element is $v_x + k_x$ with $k_x = j_x + x$

(iii) For u= 1, 2, …, m, the effect of x and n on the construction of the blocks is given by $m = |[2^n - 2/2x]$ ————(18)
and obviously $v_x = 2ux$.

For x odd

By Considering two successive n binary tuples x=w , and x=z say, where the corresponding. Decimals are even and odd, respectively. Obviously w and z differ only in the right most digit, being a 0 for w and a I for z. Thus replacing w by z affects only the third element in each set of the block for w, increasing or decreasing its decimal value by one according to whether the binary n-tuple corresponding to the second element in the set has a 0 (increasing) or a 1 (decreasing) as its right most digit. Clearly this applies for all values of n, Thus we have relations (13), (14), and (15) above.

For x even, $\boldsymbol{x \neq 2^i}$, $\boldsymbol{i = 0, 1, 2, \dots, m}$

Without loss of generality let us take n=5, then all the even decimal numbers for gray code are:
0, 1, 2, 4, 6, …, 26, 28, 30.
First consider x =6. Obviously:

$v_6 = v_4 = v_{6-2,}$

$j_6 = j_4 = j_{6-2,}$

$t_6 = k_6 =$ k4+ A4 , and T= K+A= $\begin{bmatrix} k_6 \\ \vdots \\ \cdot \\ k_6 \end{bmatrix}$ +A where A= $\begin{bmatrix}+2\\+2\\\cdots\\-2\\-2\end{bmatrix} = \begin{bmatrix}A_1\\A_2\end{bmatrix}, i.e.\ A_1 = B = \begin{bmatrix}2\\2\end{bmatrix}, A_2 = -B$

Thus $A_j = (-1)^{j+1}\,B, j = 1,2$

This is so since 4 and 6 as binary tuples differ only in the second digit from the right. These digits appear in the set of all possible n-tuples in group of two 0's and two 1's. For the 0's $\begin{bmatrix}k_6\\k_6\end{bmatrix}$ = $\begin{bmatrix}k_4\\k_4\end{bmatrix}$ +$\begin{bmatrix}2\\2\end{bmatrix}$, and for the 1's, cancellation results in k6 number of elements in A is |j4| and the number of sub matrices in A is given by: $\frac{|j6|}{2} = \frac{4}{2} = 2$. The same argument apply to numbers 10, 14, 18, …, i.e.

The difference lies in the construction of A whose elements depend on j_x and j=1, 2, …, b where b =$\frac{|jx|}{2}$, and taking a=2 we see that B = $\begin{bmatrix}a\\a\end{bmatrix}$, and that for any x, x-6 is a multiple of 4= 2a.now consider x= 12.

$v_{12} = v_8 = v_{12-4}$,

$j_{12} = j_8$,

$$\begin{bmatrix}k_{12}\\\vdots\\.\\k_{12}\end{bmatrix} = \begin{bmatrix}k_8\\\vdots\\.\\k_8\end{bmatrix} + A = T, where\ A = \begin{bmatrix}A_1\\A_2\end{bmatrix},\quad A_1 = \begin{bmatrix}4\\4\\4\\4\end{bmatrix},\quad A_2 = \begin{bmatrix}-4\\-4\\-4\\-4\end{bmatrix}$$

This is so because the corresponding binary tuples differ only in the third digit from the left. These digits appear in groups of four 0's and four1's in the set of all possible n-tuples. Following the same procedure as above we have:

j=1, 2, …,b, b=$\frac{|jx|}{a}$ $, a = 4$

B = $\begin{bmatrix}a\\a\\a\\a\end{bmatrix}$, for any x, x-12 a multiple of 8=2a, and T=$\begin{bmatrix}k_8\\\vdots\\.\\k_8\end{bmatrix}$+$\begin{bmatrix}A_1\\A_2\end{bmatrix}$

In summary,

1) a =2 starting x_1= 6
   a =4 starting x_1= 12
   a =8 starting x_1= 24 and so on. ———— (19)
2) The change is in k_x only, through A, whose construction depends entirely on x.
3) All the above result depend on a. Thus deciding a is crucial, since;
   $v_x = v_{x-a}$
   $j_x = j_{x-a}$
   $k_x = k_{x-a} + A_i$ =t$_i$, where A depends on B whose elements are a.

Let us start by x=x1. Thus for any positive integers x, and y:

x – x1 = y(2a) = 2ya ———— (20)

$\frac{x}{a} = 2y + \frac{x1}{a}$, obviously for (20) to hold, and from (19), $\frac{x1}{a}$ must be odd;but 2y is even.

Therefore $\frac{x}{a}$ must be odd. Thus we have the general result $\frac{x}{a}$ must be odd for the least $= 2^{z-1}$, i=1, 2, 3, …, m.

2.3. Examples

Two examples are given which cover all cases in theorem (1). Let n=5 for simplicity,

Example 1: x=14, now $x=14 \neq 2^i$ for any i. Thus we use equations (7), (8), (9) and (10). First we must find a.

Starting with z=1, i.e. a=2, which gives x/a = 14/2=7 (odd). Thus, by (9), (10), and (11) we get:

$v_{14} = v_{12}$ ———(21)

$j_{14} = j_{12}$ ———(22)

$k_{14} = t_{12}$ ———(23)

Now for (23) t_{12} has elements k12, and we only know $B = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$. To find j12, and k12:

here $x = 12 \neq 2^i$ for any i. Thus by (8), (9), and (10) we get:

$v_{12} = v_8$ ———(24)

$j_{12} = j_8$ ———(25)

$k_{12} = t_8$ ———(26)

And t_8 has k_8 as elements since z=3, a=4. To find j_8 and k_8:here x=3, i.e. i = 3. Therefore $m=|[\frac{30}{16}]=1$. Thus u = 1, v8 = 16, and by (5) and (6):

j_8 =0, 1, 2, 3, 4, 5, 6, 7 and

k_8 = 8, 9, 10, 11, 12, 13, 14, 15,.

Now from (26) and (8).

$$B = \begin{bmatrix} 4 \\ 4 \\ 4 \\ 4 \end{bmatrix} \text{, b= 8/4 = 2, and j = 1,2}$$

$$\text{Thus } t_{12} = \begin{bmatrix} 8 \\ 9 \\ 10 \\ 11 \\ 12 \\ 13 \\ 14 \\ 15 \end{bmatrix} + \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} = \begin{bmatrix} 12 \\ 13 \\ 14 \\ 15 \\ 8 \\ 9 \\ 10 \\ 11 \end{bmatrix}$$

From (23) and (8) we get:

$$B = \begin{bmatrix} 2 \\ 2 \end{bmatrix} \text{, b = 8/2 = 4, j= 1, 2, 3, 4,}$$

$$A_1 = A_3 = \begin{bmatrix} 2 \\ 2 \end{bmatrix}, A_2 = A_4 = \begin{bmatrix} -2 \\ -2 \end{bmatrix}. \text{ Thus } k_{14} = \begin{bmatrix} 14 \\ 15 \\ 12 \\ 13 \\ 10 \\ 11 \\ 8 \\ 9 \end{bmatrix}$$

The 14(5) block is therefore:

(14, 16, 30)
(14, 17, 31)
(14, 18, 28)
(14, 19, 29)
(14, 20, 26)
(14, 21, 27)
(14, 22, 24)
(14, 23, 25)

It is interesting to make a comparison between k_8, k_{12} and k_{14}. It is also interesting to note that:

$v_{14} = v_{12} = v_8$ and $j_{14} = j_{12} = j_8$

Note that infact we find blocks for x=8 and x=12 in finding the block for x=14. This is not always the case. It depends on how far is the value of x for the required block from an x given by x=2^i.

Example 2: x=13. Since x is odd then by (13), (14), and (15):

$v_{13} = v_{12}$ ———(27)
$j_{13} = j_{12}$ ———(28)

$k_{13} = k_{12} + (-1)^{j12}$ ———(29)
now from example 1, we have:
$v_{13} = v_8$ ———(30)
$j_{13} = j_8$ ———(31)
thus $k_{13} = k_{12} + (-1)^{j8}$ =13, 12, 15, 14, 9, 8, 11, 10.

The 13 (5) block. Is therefore:

(13, 16, 29)
(13, 17, 28)
(13, 18, 31)
(13, 19, 30)
(13, 20, 25)
(13, 21, 24)
(13, 22, 27)
(13, 23, 26)

3. Codes with minimum distance d=3 and d=4

Starting with a specific d various values for n and k can be decided since according to standard coding theory [24], a linear block code is a subspace of the vector space v_n of n binary vectors i.e. it is such that:

(i) It contains the zero vector (or codeword).

(ii) The sum modulo two of any two vectors (codewords) is another vector (codeword) in the subspace. Finding such a subspace is possible but containing the maximum number of vectors is not an easy task.

3.1.Gray Code:

The gray code is a reflected cod which can be constructed recursively [25], so that.

a) A 1 bit gray code has two codewords, 0 and 1.

b) The first 2^n codewords of an (n+1) bit gray code equal the codwords of an n bit gray code written in order with a leading 0 appended.

c) The last 2^n codewords of an (n+1) bit gray cod equal to the codewords of an n bit gray code, but written in reverse order with a leading 1 appended.

For example a 4 bit gray code in decimal notation is

0 1 3 2 6 7 5 4 12 13 15 14 10 11 9 8 ———(32)

From above it is clear that the hamming distance H for consecutive codewords is H=1.

also H = 2 for even (in distance) consecutive codewords. These properties play an important role in the work that follows.

In the following we shall use decimal notation, for this gives the result in a neat form. Otherwise working in binary is very tedious.

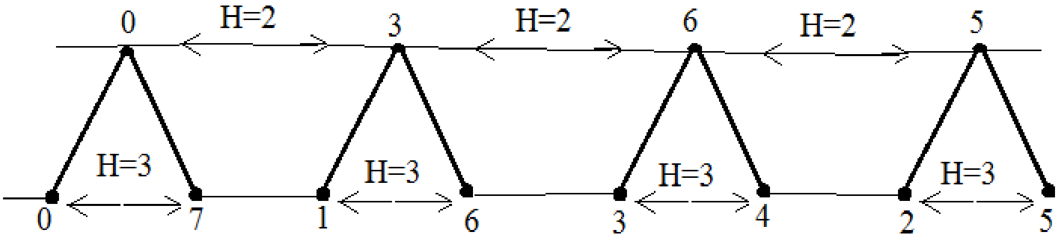
3.2.Codes having minimum hamming distance d = 3:

This is an important case, since some important codes exist having d =3; in particular the hamming code. But most importantly here the procedure leads to the development of the case for d >3.

To illustrate we use block lengths, n = 6,7. The first step is to partition the code block length n into two parts n_1, and n_2 . If n even then $n_1 = n_2$. If n is odd then $| n_1 - n_2| =1$

For n=6 , $n_1 = n_2 = 3$. The 3 bit gray code used is 0,1,3,2,6,7,5,4 . The first codeword is 00 (in binary 000000). From this the codeword 07 (in binary 000111) is derived, since the hamming distance between 0 and 7 is 3.

The structure of the codewords is shown in fig below.



Here H means hamming distance.

The 8 codewords are;

00 31 63 52
07 36 64 55
Or in binary ——(33)
000000 011001 110011 101010
000111 011110 110100 101101

Also note that

(i) $O \longrightarrow 1$, $1 \longrightarrow 3$, $3 \longrightarrow 2$ all have H = 1.

(ii) $7 \xrightarrow{H=1} 6$, $6 \xrightarrow{H=2} 4$, $4 \xrightarrow{H=1} 5$

The code is a subspace. This is so for using result from theorem (1) given by blocks in the appendix we get, for example,

$\begin{array}{c}36\\ \underline{52}\\ 64\end{array}$ $\begin{array}{c}55\\ \underline{31}\\ 64\end{array}$ $\begin{array}{c}63\\ \underline{31}\\ 52\end{array}$ by using the sets

(3, 5, 6) , (2, 4, 6) , (1, 4, 5) , and (1, 2, 3)

Notes:

(i) Since the sets of codewords (0, 7) , (0, 13) , (0, 14) ,and(0, 11) all have H=3 then any of them can be taken as codewords associated with the codeword 00.

(ii) The number of the codewords is the maximum possible. This is obvious because we are using the 3 bit gray code.

(iii) K (the number of information digits) is given by 2^k =number of codewords, here k = 3.

(iv) Since k and n are known, then we can obtain the generator matrix G. It is given by the codewords 46, 25 and 13. note that 4, 2, and 1 are chosen to given G in standard form. Thus

$$G = \begin{bmatrix}100 & 110\\ 010 & 101\\ 001 & 001\end{bmatrix}$$ This G is the same for an (6, 3) code with d=3 given by [24]

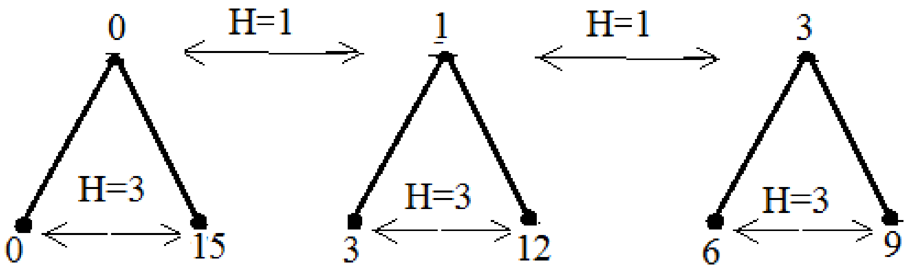
For n=7 , $n_1 = 4$, $n_2 = 3$. For n_1 we use the 4 bit gray code
0, 1, 3, 2, 6, 7, 5, 4, 12, 13, 15, 14, 10, 11, 9, 8.

For n_2 we use the 3 bit gray code. For the same n, k, d, many codes can be obtained.

For example:
00 31 63 52 126 157 105 94
07 36 64 55 121 150 102 93 ——(34)

Or the code may be
00 13 36 25 63 70 55 46
015 112 39 210 612 715 510 49 ——(35)

However the construction of both codes is similar and also to n=6. For the code in (35) the construction is shown in fig below for the first three codewords



Notes:

(i) The codewords are the maximum possible therefore the code is optimum. Also k =4 which is the maximum.

(ii)The code is a subspace. This can be checked using theorem (1)

(iii) Any of the sets (0,7) , (0,13) , (0, 14) and (0,11) can be used as the codewords associated with the codeword 00.

(iv) The generator matrix G can be obtained by

$$G=\begin{bmatrix}1000 & 110\\ 0100 & 101\\ 0010 & 011\\ 0001 & 111\end{bmatrix}\begin{matrix}46\\ 25\\ 13\\ 015\end{matrix} \qquad \text{———} (36)$$

This is the hamming code [24] which is known to be an optimum code i.e. having the maximum k for given n and d.

3.3. Theorem (2)

For d=3, 4 we have the following theorem for the construction of the code.

Theorem (2):

All the codewords of a linear binary code having minimum hamming distance d=3 or d =4 and block length n are generated (using decimal notation) as pairs (u, z) , and (u, v) by

$$u_1z_1 \,,\ \ldots\ u_i\, z_i \,,\ldots,\ u_m\, z_m$$
$$u_1v_1 \,,\ \ldots\ u_i\, v_i \,,\ldots,\ u_m\, v_m \qquad \text{———}(37)$$

with $u_i\ z_i$ and $u_i\ v_i$ are codewords such that $u_i=n_1$ bits, $z_i= v_i=n_2$ bits, $n_1+ n_2=n$ and if n even, $n_1= n_2$ and if n odd $|n_1 - n_2|=1$.

Where:

(i) u_i , z_i , v_i , i = 1,…,m are taken from the gray code of block length n_1 such that w $(u_i+ u_{i+1})=2$ ———(38)

and w $(z_i+ z_{i+1})=1$, where w is the weight of the codeword.

for d=3 , w $(z_i+ v_i)=3$, and ———(39)

for d=4 w $(z_i+ v_i)= 4$

Proof: for d=3

For the first row in (37). The u_i , i=1, …, m. are taken in sequence according to the sequence in the gray code such that consecutive codewords have H=2; i.e. $u_1=0$, $u_2=3$ and so on.

The z_i, i=1, …, m. are taken in sequence according to the sequence in gray code such H=1 for consecutive codewords, i.e. $z_1=0$, $z_2=1$, $z_3=3$ and so on. Thus $w\ (z_i+ z_{i+1})= \geq 1$ ———— (40)

Hence $w\ [\ (u_i\ ,\ z_i) + (u_{i+1}, z_{i+1})\ ] = w\ (u_i+ u_{i+1})+ w(z_i+ z_{i+1})\geq 3$.

For the second row in (37). Taking a codeword in the first row and its partner in the second row we have.

$w\ [(u_i\ z_i) + (u_i\ v_i)] = w\ (u_i+ u_i) + w\ (z_i+ v_i)=0+w(z_i+ v_i)=3$,

taking $w\ (z_i+ v_i)=3$, using theorem (1).

Now consider the codeword, $\begin{matrix} ab & de \\ ac & df \end{matrix}$ we need to show that (ab+de) is a codeword.

That is to show that the code obtained is a subspace. The first 4 codewords of the code are, $\begin{matrix} 00 & 31 \\ 0s & 3v \end{matrix}$ which gives $w\ (0+s)= w\ (1+v)= w(s)=3$ ——(41)

Let $t = (a+d)$, and $x=(b+e)$. Then by (41) $w\ (b+c) =w\ (e+f)=s$

We have $b + c=s$ and $e+f=s$. Hence $(b+e)+(c+f)=0$.

Implying $c+f=x$. thus $(a+d)(b+e)=tx$ and $(a+d)(c+f)=tx$

Let $y=(b+f)$ then by similar argument we have

$(a+d)\ (b+f)=ty$, $(a+d)\ (e+e)= ty$. Now we have to show $x+y=s$.

Since $b+e=x$, $b+f=y$ which gives $e+f=x+y=s$. thus tx and ty are codewords.

For d=4, we have for (40) $w\ (z_i+ z_{i+1})\geq 2$

Hence $w[(u_i\ ,\ z_i) + (u_{i+1}\ z_{i+1})] \geq 4$

Also $w(u_i\ \ z_i +u_i\ \ v_i)=4$ by taking $w\ (z_i+ v_i)=4$ from theorem (1). Now consider the codewords $\begin{matrix} aa & dd \\ ac & de \end{matrix}$ ————(42)

We need to show that (aa+dd) is a codeword which is obvious. Also we need to show ac de is a codeword. Consider the first 4 codewords $\begin{matrix} 00 & 33 \\ 0s & 3v \end{matrix}$ then , $w(0+s)= w(3+v)= w(s)=4$. ——(43)

Let $x=(a+c)$ then by (43) $w(a+c)=w(d+e)=s$, we get $(a+c) = (d+e) =s$, giving $a+d= c+e=x$ implying xx is a codeword.

Let $y =a+e$ then by (42) $a+e= c+d=y$. giving xy as a codeword thus the resulting codewords are $\begin{matrix} x & x \\ x & y \end{matrix}$.

4. Discussion

We shall briefly discuss extension to d > 4, and based on the ideas implied by theorems (1) and (2). Consider two codewords, for example, (u v) and (z x). Since for any minimum distance it is required that $w\ (\ (u\ \ v)+(z\ \ x)) \geq d$, then two main possibilities exist:

i. $u = z$, $w\ (v + x) =d$

ii. $w = (u + z) = q$, $w\ (v + x) \geq d- q$

q may be the d for a previously obtained code, for example, if q = 3 and present d = 7, then u and z are of length 3, while v and x are of length 4. All of these are previously obtained codewords for code, with d < 7. Since the 0 codeword is part of the code, then it is possible to start with code words (0 0) and (0 d) and choosing the succeeding codewords by a procedure similar form of previously obtained codes. As d increases, n and k increase and become large.

5. Conclusion

In attempting to design liner binary codes with specific minimum distance, it became necessary to guarantee that the codewords constitute a subspace. Thus investigation of all combinations of a binary n-tuple revealed a process embodied in theorem (1) by which set of three binary n-tuples, where in each set each element result from mod 2 addition of the other elements, are generated in blocks where all the elements of a block have a common element.

6. References

[1] N.J.A Sloane “A Short Course on Error Correcting Codes” International Center for Mechanical Sciences, Springer-Verlag Wien – New York, 1975.

[2] F.J. Macwilliams and N.J.A. Sloane “The Theory of Error Correcting Codes” , Amsterdam, North Holland, 1977.

[3] A.E. Brower, J.B. Shearer, N.J.A. Sloane. And W. D. Smith, “A New Table of Constant Code Weights”, IEEE Trans. Inform. Theory, vol. 36, Nov. 1990.

[4] M. Plotkin, “Binary Codes with Specific Minimum distance”, IEEE Transactions on Information Theory, Vol. 6, Sep.1960.

[5] J. Quistorff, “Some Remarks on the Plotkin Bound”, the Electronic Journal of Combinatorics, Vol. 10,2003.

[6] Berlekamp, E.R., “Algebraic Coding Theory”. McGraw-Hill, New York, 1968.

[7] H. N. Ward, “An Introduction to Algebraic Coding Theory”,

http://www.math.virginia.edu/1-hnw/codingtheory.pdf

[8] J.I. Hall, “Notes on Coding Theory”, Departments of Mathematics, Michigan State University, Jan.2003.

[9] M. Sudan, “Algorithmic Issues in Coding Theory’, lecture notes in computer science, The Laboratory for Computer Science, MIT,1998.

[10] D.A. Spielman, “Linear Time Encodable and Decodable Error Correcting Codes”, IEEE Transactions on Information Theory, 42(6),1996.

[11] A.Vardy, Algorithmic Complexity in Coding Theory and the Minimum Distance Problem”, proceedings of the twenty ninth annual ACM symposium on theory of computing, 1997.

[12] P.Ding and J.D. Key, “Minimum Weight Codewords as generators of generalized Reed-Muller Codes”, lecture notes in computers science, vol. 1346, 1997.

[13] S. Gao and J.D. Key, “ Bases of Minimum Weight Vectors Codes from Designs” Finite Fields Appl., vol. 4,1998.

[14] T.A. Gulliver and M.Harada, “On the minimum Weight of codes Over F5 Constructed from Certain Conference Matrices” , Design, and Cryptography, vol. 31, 2004

[15] T. Etzion and A. Vardy, “Perfect Binary Codes: Constructions, Properties, and Enumerations”, IEEE Transactions on Information Theory, vol. 40 no. 3,1994

[16] N.j.A. Sloane and J.G Thompson, “Cyclic Slef-Dual Codes”, IEEE Tranactions on Information Theory, vol. 29,1983.

[17] C.Bachoc, “Application of coding Theory to the Construction of Modular Lattices”, J.Combin. Theory, A78,1997.

[18] Q.A. Nguyen L.Gyorfi, and J.L. Massy, “Construction of Binary Constant Weight Cyclic Codes and Cyclically Permutable Codes”, IEEE Transactions on Information Theory, vol. 38, no. 4,1992.

[19] O. Moreno and P. Vijay Kumar, “Minimum Distance Bounds for Cyclic codes and Deligne’s Theorem” IEEE Transaction on Information Theory, vol. 39, no 5, Sep. 1993.

[20] J.H. Conway and N.J.A Sloane, “Anew Upper Bound on the Minimal Distance of Self-Dual Codes”, IEEE Transaction on Information Theory, vol. 36, Nov, 1990.

[21] N.J Calkin J.D. Key, and M.J.D.E. Resmini, “Minimum Weights and Dimension Formulas for Some Geometric Codes”, Design, and Cryptography, vol, 17, Sep, 1998.

[22] A.Canteaut and F. Chabaud, “A New Algorithm for Finding Minimum Weight Words in a Linear Code: Application to Primitive Sense BCH Code of Length 511”, INRIA, no. 2685, Oct. 1995.

[23] P.E. Allard S.G.S. Shiva, and S.E. Tavers, “A Note on the Decomposition of Cyclic Code into Cyclic Classes”, Information and Control 22, 1973.

[24] S. Lin, An Introduction to Error Correcting Codes”, International Center for Mechanical Sciences, Prentice-Hall, INC, 1970.

[25] R.M. Losee, “A Gray Code Ordering for Documents on Shelves”, Journal of the American Society for Information Science, 43 (4), 1992.

[26] B.Parhami, “Dependable Computing”, University of California, Santa Barbar.

[27] M.Pujol and M.villanueva, “ Computing the Minimum Hamming Distance”, University de Barcelona, 2012.

[28] S.Draft, “Coping with Bit Error using Error Correction Codes”, MIT Leeture Notes, Sep.32,2012.

[29] E. Pasalic, “Coding theory and Applications”, University of Primoreska. 2013.

[30] P.Srelatha et al, “Extended (10, 5) Binary Hamming Code Generator for Telecommanding Applications’, International Journal of Soft Computing and Engineering, volume-4, Issue-2, May 2014.

Appendix:

(A) The following x(n) blocks generated by theorem (1) for n = 5

1(5)
(1, 2, 3)
(1, 4, 5)
(1, 6, 7)
(1, 8, 9)
(1, 10, 11)
(1, 12, 13)
(1, 14, 15)
(1, 16, 17)
(1, 18, 19)
(1, 20, 21)
(1, 22, 23)
(1, 24, 25)
(1, 26, 27)
(1, 30, 31)

2(5)
(2, 4, 6)
(2, 5, 7)
(2, 8, 10)
(2, 9, 11)
(2, 12, 14)
(2, 13, 15)
(2, 16, 18)
(2, 17, 19)
(2, 20, 22)
(2, 21, 23)
(2, 24, 26)
(2, 28, 30)
(2, 29, 31)

3(5)
(3, 4, 7)
(3, 5, 6)
(3, 8, 11)
(3, 9, 10)
(3, 12, 15)
(3, 13, 14)
(3, 16, 19)
(3, 17, 18)
(3, 20, 23)
(3, 21, 22)
(3, 24, 27)
(3, 25, 26)
(3, 28, 31)
(3, 29, 30)

4(5)
(4, 8, 12)
(4, 9, 13)
(4, 10, 14)
(4, 11, 15)
(4, 16, 20)
(4, 17, 21)
(4, 18, 22)
(4, 19, 23)
(4, 24, 28)
(4, 25, 29)
(4, 26, 30)
(4, 27, 31)

5(5)
(5, 8, 13)
(5, 9, 13)
(5, 10, 13)
(5, 11, 13)
(5, 16, 13)
(5, 17, 13)
(5, 18, 13)
(5, 19, 13)
(5, 24, 13)
(5, 25, 13)
(5, 26, 13)
(5, 27, 13)

6(5)
(6, 8, 14)
(6, 9, 5)
(6, 10, 12)
(6, 11, 13)
(6, 16, 22)
(6, 17, 23)
(6, 18, 20)
(6, 19, 21)
(6, 24, 30)
(6, 25, 31)
(6, 26, 28)
(6, 27, 29)

7(5)
(7, 8, 15)
(7, 9, 14)
(7, 10, 10)
(7, 11, 11)
(7, 16, 14)
(7, 17, 15)
(7, 18, 18)
(7, 19, 19)
(7, 24, 22)
(7, 25, 23)
(7, 26, 26)
(7, 27, 30)

8(5)
(8, 16, 24)
(8, 17, 25)
(8, 18, 26)
(8, 19, 27)
(8, 20, 28)
(8, 21, 29)
(8, 22, 30)
(8, 23, 31)

9(5)
(9, 16, 25)
(9, 17, 24)
(9, 18, 27)
(9, 19, 26)
(9, 20, 29)
(9, 21, 28)
(9, 22, 31)
(9, 23, 30)

10(5)
(10, 16, 26)
(10, 17, 27)
(10, 18, 24)
(10, 19, 25)
(10, 20, 30)
(10, 21, 31)
(10, 22, 28)
(10, 23, 29)

11(5)
(11, 16, 27)
(11, 17, 26)
(11, 18, 25)
(11, 19, 24)
(11, 20, 31)
(11, 21, 30)
(11, 22, 29)
(11, 23, 28)

12(5)
(12, 16, 28)
(12, 17, 29)
(12, 18, 30)
(12, 19, 3)
(12, 20, 2)
(12, 21, 24)
(12, 22, 26)
(12, 23, 27)

13(5)
(13, 16, 27)
(13, 17, 27)
(13, 18, 27)
(13, 19, 27)
(13, 20, 27)
(13, 21, 27)
(13, 22, 27)
(13, 23, 27)

14(5)
(14, 16, 28)
(14, 17, 28)
(14, 18, 28)
(14, 19, 28)
(14, 20, 28)
(14, 21, 28)
(14, 22, 28)

15(5)
(15, 16, 31)
(15, 17, 30)
(15, 18, 29)
(15, 19, 28)
(15, 20, 27)
(15, 21, 12)
(15, 22, 25)

(B) To show that the code in (35) is a subspace by using the result for x(n) blocks in (A) above.

$$\begin{array}{cccccccc} 00 & 13 & 36 & 25 & 63 & 70 & 55 & 46 \\ 015 & 112 & 39 & 210 & 612 & 715 & 510 & 49 \end{array}$$

By construction; $\begin{array}{c} 0 \\ 15 \\ \hline 15 \end{array} \quad \begin{array}{c} 3 \\ 12 \\ \hline 15 \end{array} \quad \begin{array}{c} 6 \\ 9 \\ \hline 15 \end{array}$ similarly for (5, 10), (3, 12).

(0, 15) , (5, 10) , and (6, 9). For the rest of the codewords we have.

$$\begin{array}{cccccc} 13 & 13 & 13 & 13 & 13 & 13 \\ 36 & 63 & 55 & 39 & 612 & 510 \\ \hline 25 & 70 & 46 & 210 & 715 & 49 \end{array}$$

$$\begin{array}{cccccc} 112 & 112 & 112 & 112 & 112 & 112 \\ 36 & 25 & 63 & 70 & 55 & 46 \\ \hline 210 & 39 & 715 & 612 & 49 & 510 \end{array}$$

$$\begin{array}{cccc} 36 & 36 & 36 & 6 \\ 612 & 715 & 510 & 49 \\ \hline 510 & 49 & 612 & 715 \end{array}$$

$$\begin{array}{cccc} 36 & 36 & 36 & 36 \\ 612 & 715 & 510 & 49 \\ \hline 510 & 49 & 612 & 715 \end{array}$$

$$\begin{array}{cccc} 39 & 36 & 36 & 39 \\ 63 & 70 & 55 & 46 \\ \hline 510 & 49 & 612 & 715 \end{array}$$

And so on for all codewords.