J. E. MacDonald

Design Methods for Maximum Minimum-Distance Error-Correcting Codes*

Abstract: In error-correcting codes for combating noisy transmission channels, a central concept is the notion of minimum distance. If a code can be constructed with minimum distance between code points of $2m+1$, then any number of errors per code word which does not exceed m can be corrected, thus increasing the reliability of transmission above that to be expected with no redundancy in the code.

An upper bound on minimum distance is derived which depends on g (the number of code points or messages required) and n (the number of binary symbols per code point). This bound is complementary to a bound due to Hamming and uses an argument which is essentially due to Plotkin.

Construction methods are presented for codes which actually achieve the upper bound on minimum distance for any g and an infinite class of integers n which depend on g. Sixteen code types are described: three for $g=2h-1$, six for $g=2h$, and seven for $g=2^k$.

Introduction

Much of the work on synthesis of error-correcting codes has been devoted to devising methods of achieving a certain minimum distance between the sequences of binary symbols which are the code points of the code. Hamming[1] defined distance between two code points as the number of correspondingly placed binary symbols which differ between the two code points. As shown by Hamming, a major advantage of being able to find a code structure with a certain minimum distance, min d, is as follows:

If min $d=2m+1$, any error pattern containing m or fewer errors can be corrected.

If min $d=2m+2$, any error pattern containing m or fewer errors can be corrected. In addition, any error pattern containing $(m+1)$ errors can be detected.

The correcting procedure is simply to change any received sequence of n symbols which is not a code point into that code point which is "nearest" to the received sequence. This procedure does not depend in any direct way on the type of noise which may corrupt the transmitted code points. However, one hopes that error patterns containing e errors are more probable than error patterns containing more than e errors. Whether this is true or not for the noise at hand, it is not difficult to see that, should we be able to correct all error patterns containing e errors, we would automatically be able to correct all error patterns containing $(e-1)$ or fewer errors if we chose to do so.

*This paper is based in part on a thesis submitted in partial fulfillment of the degree of Master of Electrical Engineering at Syracuse University awarded January, 1958.

Bounds on minimum distance

Suppose we have g code points, each comprised of an n-binary symbol-vector. Hamming has shown that, for fixed g and n, an upper bound on minimum distance can be given as follows. Define an integer m by:

$$\sum_{i=0}^{m}\binom{N}{i} \leqslant \frac{2^N}{g} < \sum_{i=0}^{m+1}\binom{N}{i},$$

$$\text{where } \binom{N}{i} = \frac{(N)!}{(i)!(N-i)!}. \qquad (1)$$

Then,

$$\begin{aligned} &\min d \leqslant 2m+1 \text{ for } n=N, \\ &\min d \leqslant 2m+2 \text{ for } n=N+1. \end{aligned} \qquad (2)$$

In Appendix I the following upper bound on minimum distance is derived:

$$\text{For } g=2h\text{: } \min d \leqslant \frac{ng}{2[g-1]}$$

$$\text{For } g=2h-1\text{: } \min d \leqslant \frac{n[g+1]}{2g} \qquad (3)$$

$$h=1, 2, 3, 4, 5, \ldots\ldots$$

(The argument used in Appendix I is essentially identical to an argument used by Plotkin[3], who found an upper bound on g for fixed minimum d and fixed n. Because of our present need for an upper bound on minimum d for fixed g and fixed n and also because of the limited availability of Plotkin's paper, the proof of (3) is given in full in Appendix I.)

The bound of (3) is sometimes larger, sometimes smaller than that of Hamming. Thus, the smaller of the two is a better bound than either one alone. In general, if the ratio $2^n/g$ is large, (3) provides a tighter bound than (2). For example, in his paper Hamming uses a form of (2) to predict the existence of a code with $n=7$, $g=4$, and min $d=5$ because

$$\binom{7}{0}+\binom{7}{1}+\binom{7}{2}=29 \leqslant \frac{2^7}{4}=32\,.$$

He then points out that it can be shown by trial and error that such a code does not exist. Using (3), it can be shown analytically that such a code does not exist because

$$5 > \frac{7(4)}{2(4-1)}=4^+\,.$$

The results of (2) and (3) are tabulated in Table 1 for several values of n and g.

A general property of maximum minimum-distance codes

For convenience, let us visualize an array of the code points of a code as a matrix with g rows and n columns. Selection of a particular message or code point to be transmitted then amounts to selection of a row from the matrix. In deriving (3) in Appendix I it is necessary to assume that for each column of the matrix we have the following property:

For $g=2h$ (g even): Each one of the n columns contains $g/2$ ones and $g/2$ zeroes.

For $g=2h-1$ (g odd): Each one of the n columns contains $(g+1)/2$ ones and $(g-1)/2$ zeroes (or alternatively, $(g-1)/2$ ones and $(g+1)/2$ zeroes).

Thus of all the myriad possible codes for a given n and g, we are immediately led to consider only those with the property just stated if we wish to attempt the construction of maximum minimum-distance codes.

Construction of max min d codes for $g=2h-1$ (g odd)

A natural question to ask in view of the previous section is: how many *different* columns can be formed which each consist of, say, $(g+1)/2$ ones and $(g-1)/2$ zeroes? The answer is:

$$N_{2h-1}=\binom{g}{(g+1)/2}=\frac{(g)!}{[(g+1)/2]![(g-1)/2]!}\,. \qquad (4)$$

Let us then examine a code where we set $n=N_{2h-1}$ and agree to use each of the N_{2h-1} types of columns exactly once in forming the code matrix. For any two code points, $A^{(e)}=(a_1^{(e)}, a_2^{(e)}, \ldots, a_n^{(e)})$ and $A^{(f)}=(a_1^{(f)}, a_2^{(f)}, \ldots, a_n^{(f)})$, $1 \leqslant e, f \leqslant g$, we have:

$$\text{distance from } A^{(e)} \text{ to } A^{(f)}=d(A^{(e)}, A^{(f)})=d(A^{(f)}, A^{(e)})$$

$$=\sum_{i=1}^{n} a_i^{(e)}-2\sum_{i=1}^{n} a_i^{(e)}a_i^{(f)}+\sum_{i=1}^{n} a_i^{(f)}\,. \qquad (5)$$

These terms are evaluated separately as follows. The first term is a count of the number of columns which contain a one in row e. This is:

$$\sum_{i=1}^{n} a_i^{(e)}=\binom{g-1}{(g-1)/2}=\frac{(g-1)!}{[(g-1)/2]![(g-1)/2]!}\,, \qquad (6)$$

since we remove a single "one" and a single row and form, once each, all possible combinations of $(g-1)/2$ ones and $(g-1)/2$ zeroes in the remaining $(g-1)$ rows. The third term is identical with the first. The second term is a count of the number of columns which contain ones in both row e and row f. This is:

$$\sum_{i=1}^{n} a_i^{(e)}a_i^{(f)}=\binom{g-2}{(g-3)/2}=\frac{(g-2)!}{[(g-3)/2]![(g-1)/2]!} \qquad (7)$$

because we remove two ones and two rows and form all possible combinations of $(g-3)/2$ ones and $(g-1)/2$ zeroes in $(g-2)$ rows. Use of (6) and (7) in (5) gives:

$$d(A^{(e)}, A^{(f)})=2\left[\binom{g-1}{(g-1)/2}-\binom{g-2}{(g-3)/2}\right]$$

$$=\binom{g}{(g+1)/2}\frac{[g+1]}{2g}\,. \qquad (8)$$

Clearly this result is independent of the choice of e and f, and hence all distances are equal and, of course, equal to the minimum distance for the code. Since we chose $n=N_{2h-1}$, we can see from (4), (8), and (3), that the codes just constructed are max min d codes. Such codes will henceforth be called *Type 1 codes.*

A variation of a Type 1 code will be designated a *Type 2 code* and is constructed as follows. Starting with Type 1 code, delete any one column. Thus n is diminished by one and the minimum distance is diminished by one:

$$n=\binom{g}{(g+1)/2}-1\,. \qquad (9)$$

$$\min d=\binom{g}{(g+1)/2}\frac{[g+1]}{2g}-1\,. \qquad (10)$$

Table 1 **Upper bound on minimum distance as a function of *n* and *g*.**

Note: Upper row from Eqs. (1) and (2) (Hamming). Lower row from Eq. (3).

g	*n* → 2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
2	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
3	1	2	3	4	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
	1	2	2	3	4	4	5	6	6	7	8	8	9	10	10	11	12	12	13
4	1	2	2	3	4	5	6	6	7	8	9	10	11	12	13	14	15	16	16
	1	2	2	3	4	4	5	6	6	7	8	8	9	10	10	11	12	12	13
5		1	2	3	4	4	5	6	7	8	9	10	10	11	12	13	14	15	16
	1	1	2	3	3	4	4	5	6	6	7	7	8	9	9	10	10	11	12
6		1	2	2	3	4	5	6	6	7	8	9	10	11	12	13	14	14	15
	1	1	2	3	3	4	4	5	6	6	7	7	8	9	9	10	10	11	12
7		1	2	2	3	4	4	5	6	7	8	8	9	10	11	12	13	14	15
	1	1	2	2	3	4	4	5	5	6	6	7	8	8	9	9	10	10	11
8		1	2	2	3	4	4	5	6	7	8	8	9	10	11	12	13	14	14
	1	1	2	2	3	4	4	5	5	6	6	7	8	8	9	9	10	10	11
9			1	2	3	4	4	5	6	6	7	8	9	10	11	12	12	13	14
	1	1	2	2	3	3	4	5	5	6	6	7	7	8	8	9	10	10	11
10			1	2	2	3	4	5	6	6	7	8	9	10	10	11	12	13	14
	1	1	2	2	3	3	4	5	5	6	6	7	7	8	8	9	10	10	11
11			1	2	2	3	4	5	6	6	7	8	9	10	10	11	12	13	14
	1	1	2	2	3	3	4	4	5	6	6	7	7	8	8	9	9	10	10
12			1	2	2	3	4	4	5	6	7	8	8	9	10	11	12	12	13
	1	1	2	2	3	3	4	4	5	6	6	7	7	8	8	9	9	10	10
13			1	2	2	3	4	4	5	6	7	8	8	9	10	11	12	12	13
	1	1	2	2	3	3	4	4	5	5	6	7	7	8	8	9	9	10	10
14			1	2	2	3	4	4	5	6	6	7	8	9	10	10	11	12	13
	1	1	2	2	3	3	4	4	5	5	6	7	7	8	8	9	9	10	10
15			1	2	2	3	4	4	5	6	6	7	8	9	10	10	11	12	13
	1	1	2	2	3	3	4	4	5	5	6	6	7	8	8	9	9	10	10
16			1	2	2	3	4	4	5	6	6	7	8	9	10	10	11	12	13
	1	1	2	2	3	3	4	4	5	5	6	6	7	8	8	9	9	10	10
17				1	2	2	3	4	5	6	6	7	8	9	10	10	11	12	13
	1	1	2	2	3	3	4	4	5	5	6	6	7	7	8	9	9	10	10
18				1	2	2	3	4	5	6	6	7	8	9	10	10	11	12	12
	1	1	2	2	3	3	4	4	5	5	6	6	7	7	8	9	9	10	10
19				1	2	2	3	4	4	5	6	7	8	9	10	10	11	12	12
	1	1	2	2	3	3	4	4	5	5	6	6	7	7	8	8	9	10	10
20				1	2	2	3	4	4	5	6	7	8	9	10	10	11	12	12
	1	1	2	2	3	3	4	4	5	5	6	6	7	7	8	8	9	10	10

The right side of (10) is necessarily an integer. The bound on minimum distance for Type 2 codes is found by using (9) in (3):

$$\min d \leqslant \left[\binom{g}{(g+1)/2} - 1\right] \frac{[g+1]}{2g}. \tag{11}$$

To show Type 2 codes are max min d we need only show the right side of (11) exceeds the right side of (10) by less than unity. Thus:

$$\begin{aligned}\Delta &= \left[\binom{g}{(g+1)/2} - 1\right] \frac{[g+1]}{2g} \\ &\quad - \left[\binom{g}{(g+1)/2} \frac{[g+1]}{2g} - 1\right] \\ &= 1 - \frac{g+1}{2g}.\end{aligned} \tag{12}$$

The last expression of (12) is clearly less than unity and Type 2 are thus max min d codes. (Equations quite similar to (9) through (12) can be used to prove the max min d property for other code types to be discussed in the remainder of the paper. For brevity, this will be designated henceforth the Δ *argument.*)

Another variation of a Type 1 code will be designated a *Type 3 code*. To form a Type 3 code delete any two columns from a Type 1 code. We then have:

$$n = \binom{g}{(g+1)/2} - 2 \tag{13}$$

$$\min d \geqslant \binom{g}{(g+1)/2} \frac{[g+1]}{2g} - 2. \tag{14}$$

The Δ argument shows Type 2 codes are max min d codes.

Construction of max min d codes for $g=2h$ (g even)

How many different columns can be formed which contain $(g/2)$ ones and $(g/2)$ zeroes? The answer is:

$$N_{2h} = \binom{g}{g/2}. \tag{15}$$

Consider then a code (Type 4) where we take $n=N_{2h}$ and agree to use each of the N_{2h} types of columns exactly once in forming the code matrix. For any two code points, $A^{(e)}$ and $A^{(f)}$, the terms of (5) are evaluated as follows. The first term is a count of the number of columns which contain a one in row e. Reasoning as we did for (6), this is:

$$\sum_{i=1}^{n} a_i^{(e)} = \binom{g-1}{(g-2)/2}. \tag{16}$$

The third term is identical with the first. The second is evaluated similarly to (7), giving:

$$\sum_{i=1}^{n} a_i^{(e)} a_i^{(f)} = \binom{g-2}{(g-4)/2}. \tag{17}$$

Use of (16) and (17) in (5) yields:

$$\begin{aligned}d(A^{(e)}, A^{(f)}) &= 2\left[\binom{g-1}{(g-2)/2} - \binom{g-2}{(g-4)/2}\right] \\ &= \binom{g}{g/2} \frac{g}{2[g-1]}.\end{aligned} \tag{18}$$

The value of (18) is clearly independent of e and f, and hence all distances are equal and, of course, equal to the minimum distance. Since we chose $n=N_{2h}$, the Δ argument proves Type 4 codes are max min d.

A variation of Type 4 codes is designated *Type 5* and is formed by deleting any one column of a Type 4 matrix. For Type 5 codes:

$$n = \binom{g}{g/2} - 1 \tag{19}$$

$$\min d = \binom{g}{g/2} \frac{g}{2[g-1]} - 1. \tag{20}$$

By the Δ argument, Type 5 codes also yield max min d codes.

Another variation of Type 4 is designated *Type 6* and is formed by deleting any two columns from a Type 4 code matrix. This gives for Type 6:

$$n = \binom{g}{g/2} - 2 \tag{21}$$

$$\min d \geqslant \binom{g}{g/2} \frac{g}{2[g-1]} - 2. \tag{22}$$

The Δ argument shows Type 6 are max min d codes.

Another variation for $g=2h$ can be formed from a Type 1 code designed for $g=2h-1$. We will designate such a variation as *Type 7* and will show later that it is very similar to a Type 4 code. Consider a Type 1 code with one more code point affixed. Let this code point consist of n zeroes: $\phi = (0, 0, 0, \ldots\ldots, 0, 0)$.

The distance from any of the code points, say $A^{(e)}$, of the original Type 1 code to the ϕ code point is given by (6) after we substitute $(g-1)$ for the g of (6). Thus,

$$d(\phi, A^{(e)}) = \binom{g-2}{(g-2)/2}. \tag{23}$$

The distance between any two code points both belonging to the original Type 1 code is given by (8) after substituting $(g-1)$ for g:

$$d(A^{(e)}, A^{(f)}) = \binom{g-1}{g/2} \frac{g}{2[g-1]}. \tag{24}$$

Further algebraic manipulation shows (23) and (24) are identical in value. Thus the minimum distance of a Type 7 code is given by either equation. Since we chose n as that value given by (4) except that $(g-1)$ should be substituted for g, for Type 7 codes we have:

$$n = \binom{g-1}{g/2}. \tag{25}$$

Table 2 **Summary of code types 1 to 9.**

Code Type	g	n	min d
1*	$2h-1$	$\binom{g}{(g+1)/2}$	$\binom{g}{(g+1)/2}\frac{g+1}{2g}$
2	$2h-1$	$\binom{g}{(g+1)/2}-1$	$\binom{g}{(g+1)/2}\frac{g+1}{2g}-1$
3	$2h-1$	$\binom{g}{(g+1)/2}-2$	$\binom{g}{(g+1)/2}\frac{g+1}{2g}-2$
4*	$2h$	$\binom{g}{g/2}$	$\binom{g}{g/2}\frac{g}{2[g-1]}$
5	$2h$	$\binom{g}{g/2}-1$	$\binom{g}{g/2}\frac{g}{2[g-1]}-1$
6	$2h$	$\binom{g}{g/2}-2$	$\binom{g}{g/2}\frac{g}{2[g-1]}-2$
7*	$2h$	$\binom{g-1}{g/2}$	$\binom{g-1}{g/2}\frac{g}{2[g-1]}$
8	$2h$	$\binom{g-1}{g/2}-1$	$\binom{g-1}{g/2}\frac{g}{2[g-1]}-1$
9	$2h$	$\binom{g-1}{g/2}-1$	$\binom{g-1}{g/2}\frac{g}{2[g-1]}-2$

*All distances equal to minimum distance.

Use of the Δ argument shows $\Delta=0$. Since this is so, we may derive a max min d code (*Type 8*) from Type 7 exactly as we did a Type 2 from a Type 1. Another variation of a Type 7 is called *Type 9* and is formed by dropping any two columns from a Type 7 code as we did in deriving a Type 3 from a Type 1. Again the Δ argument reveals Type 8 and Type 9 codes are max min d. The results derived so far are tabulated in Table 2.

Summary and generalization of max min *d* code Types 1 to 9

The results on code Types 1 through 9 can be applied directly to many cases where, for a given g, the error potential is such as to require additional redundancy, that is, a larger value of n, but it is still desired that the resultant code be a max min d code. Let

$$\{x\} = \text{greatest integer contained in } x. \tag{26}$$

The bounds of (3) may be written:

$$\begin{aligned}&\text{For } g=2h\text{: max min } d=\left\{\frac{ng}{2[g-1]}\right\}\\ &\text{For } g=2h-1\text{: max min } d=\left\{\frac{n[g+1]}{2g}\right\}.\end{aligned} \tag{27}$$

Suppose we have constructed a code with $g=2h$ and n_1 binary symbols per code point whose minimum distance, $(\min d)_1$, obeys:

$$(\min d)_1=\frac{n_1 g}{2[g-1]}. \tag{28}$$

Suppose we have a code with the same $g=2h$ as the first code and n_2 binary symbols whose minimum distance, $(\min d)_2$, obeys:

$$(\min d)_2=\left\{\frac{n_2 g}{2[g-1]}\right\}. \tag{29}$$

If now we form a code with $n=n_1+n_2$ symbols by adjoining the two codes above, we form a code matrix of g rows and n columns. For this new code:

$$\begin{aligned}\min d &\geqslant (\min d)_2+(\min d)_1\\ &=\left\{\frac{n_2 g}{2[g-1]}\right\}+\frac{n_1 g}{2[g-1]}\\ &=\left\{\frac{n_2 g}{2[g-1]}+\frac{n_1 g}{2[g-1]}\right\}=\left\{\frac{ng}{2[g-1]}\right\}.\end{aligned} \tag{30}$$

In view of (27), we have thus shown the new code to be a max min d code. The argument for $g=2h-1$ is identical.

This result means that certain values of n can be chosen which yield max min d codes by adjoining any number of codes whose minimum distance achieves equality in (3) to at most one code whose minimum distance merely obeys (27). This generalization for many values of n is given by Table 3. In Table 3 we have made use of the identity:

$$\binom{g}{g/2}=2\binom{g-1}{g/2-1}. \tag{31}$$

This identity means that for some values of n and $g=2h$ we have a choice of one copy of a Type 4 code or two copies of a Type 7 code. For example, suppose we have $g=6$ and $n=40$. Since $\binom{g-1}{g/2}=10$, we have $c=4$. Referring to Table 3, line 4, we see that a max min d code can be made: (1) by adjoining two copies of a Type 4 code, or (2) by adjoining one copy of a Type 4 and two copies of a Type 7, or (3) by adjoining four copies of a Type 7 code. It should be noted that Types 7, 8, and 9 contain the code point ϕ whereas all the others do not contain ϕ as a code point. In certain physical situations there may be a valid reason for desiring the inclusion or exclusion of ϕ as a code point.

It is interesting to note from Table 3 that we have solved the problem of constructing a max min d code for $g=2$, 3, or 4 and any value of n. The case $g=2$ is completely trivial since any two code points which are complementary [e.g., ϕ and $I=(1,1,\ldots 1,1)$] have as minimum distance n which satisfies (3). For $g=3$, reference to Table 3 shows we can construct a max min d code for $n=3c$ or $3c-1$ or $3c-2$. Since there is a c corresponding to any n, the max min d code can always be constructed as in Table 3. For $g=4$, reference to Table 3 again shows

we can construct a max min d code for $n=3c$ or $3c-1$ or $3c-2$, so there is a c corresponding to any n.

Some examples of actual code matrices are shown in Table 4.

Construction of max min d codes for $g=2^k$

In this and some following sections we shall derive some construction methods for $g=2^k$. In a previous section we discussed max min d code construction for $g=2h$ and of course $g=2^k$ is a special case of $g=2h$. However, the ensuing results are important in their own right, because the smallest value of n for which we can build a max min d code using Table 3 is often much larger than a designer needs to satisfy a certain minimum reliability. In the methods of construction to be discussed, we will be able to use a much smaller value of n for a given $g=2^k$ than construction methods for $g=2h$ would require.

The construction of max min d codes for $g=2^k$ will be made through the use of group codes, studied extensively by Slepian,[2] who calls them *group alphabets*. A brief review of the pertinent properties of group codes will serve as an introduction to our construction procedure.

Consider a set of k vectors, each consisting of n binary symbols:

$$A^{(1)}=(a_1^{(1)}, a_2^{(1)}, a_3^{(1)}, \ldots\ldots, a_n^{(1)})$$
$$A^{(2)}=(a_1^{(2)}, a_2^{(2)}, a_3^{(2)}, \ldots\ldots, a_n^{(2)})$$
$$A^{(4)}=(a_1^{(4)}, a_2^{(4)}, a_3^{(4)}, \ldots\ldots, a_n^{(4)})$$
$$\vdots$$
$$A^{(2^{k-1})}=(a_1^{(2^{k-1})}, a_2^{(2^{k-1})}, a_3^{(2^{k-1})}, \ldots. a_n^{(2^{k-1})}). \qquad (32)$$

Note that the superscript which identifies the vector is a power of 2. This will be convenient subsequently. Using the symbol $\dotplus$ to denote the sum modulo 2, define the $\dotplus$ operation on two vectors $A^{(e)}$ and $A^{(f)}$ to be:

$$A^{(e)}\dotplus A^{(f)}=(a_1^{(e)}\dotplus a_1^{(f)}, a_2^{(e)}\dotplus a_2^{(f)}, \ldots\ldots, a_n^{(e)}\dotplus a_n^{(f)}). \qquad (33)$$

For example:

$$A^{(e)}=(1\ 0\ 1\ 0\ 1\ 1\ 0\ 1)$$
$$A^{(f)}=(1\ 0\ 0\ 1\ 0\ 1\ 0\ 1)$$
$$A^{(e)}\dotplus A^{(f)}=(0\ 0\ 1\ 1\ 1\ 0\ 0\ 0).$$

Define a larger set of vectors by the relation:

$$A^{(e)}=b_1A^{(1)}\dotplus b_2A^{(2)}\dotplus b_4A^{(4)}\dotplus\ldots\ldots\dotplus b_{2^{k-1}}A^{(2^{k-1})}, \qquad (34)$$

where each b_h is either 0 or 1. For convenience let the b_h's be determined uniquely from e by writing e in the binary form:

$$e=\sum_{h=0}^{k-1} b_h^{(e)}\,2^h; \qquad b_h^{(e)}=0, 1.$$

This means, of course, that e obeys:

$$0\leqslant e\leqslant 2^k-1.$$

Table 3 **Generalization of code types 1 to 9.**

g=2h−1

n	Code Type 1	Type 2	Type 3
$c\left[\binom{g}{(g+1)/2}\right]$	c	0	0
$c\left[\binom{g}{(g+1)/2}\right]-1$	$c-1$	1	0
$c\left[\binom{g}{(g+1)/2}\right]-2$	$c-1$	0	1

g=2h

n	Code Type 4	Type 5	Type 6	Type 7	Type 8	Type 9	b
$c\left[\binom{g-1}{g/2}\right]$	b	0	0	$c-2b$	0	0	$0, 1, 2, \ldots, c/2$.
$c\left[\binom{g-1}{g/2}\right]-1$	b	0	0	$c-1-2b$	1	0	$0, 1, 2, \ldots, \frac{c-2}{2}$.
OR	b	1	0	$c-2-2b$	0	0	$0, 1, 2, \ldots, \frac{c-2}{2}$.
$c\left[\binom{g-1}{g/2}\right]-2$	b	0	0	$c-1-2b$	0	1	$0, 1, 2, \ldots, \frac{c-2}{2}$.
OR	b	0	1	$c-1-2b$	0	0	$0, 1, 2, \ldots, \frac{c-2}{2}$.

Since there are k different b_h's in (34), there are 2^k different vectors represented by $A^{(e)}$ in (34). These 2^k vectors will be the coded form of the 2^k different code points to be transmitted. In the terminology of the previous sections, then, the coding matrix consists of 2^k rows and n columns. The e row of the coding matrix is the vector $A^{(e)}$.

It should be noted for the sake of completeness that the k vectors of (32) must be linearly independent; i.e., no one of them can be written as a sum modulo 2 of some of the others. In our construction methods to be discussed, this problem will be treated as it arises for the various code types.

Slepian has shown that the 2^k vectors of the form of (34) form an Abelian group under addition modulo 2. A fundamental property of such mathematical objects is the following: if $A^{(e)}$ and $A^{(f)}$ belong to the group, then $(A^{(e)}+A^{(f)})$ also belongs to the group. Slepian defines the weight of a code point by:

$$W(A^{(e)}) = W_e = \sum_{i=1}^{n} a_i^{(e)}, \tag{35}$$

so that

$$W(A^{(e)}+A^{(f)}) = \sum_{i=1}^{n} (a_i^{(e)}+a_i^{(f)}) = d(A^{(e)}, A^{(f)}). \tag{36}$$

Since the group property mentioned above holds for any two members of the group, (36) gives the result: *The distance between* any two *code points of a group code is equal to the weight of some code point of the group.*

For example:

$$d(A^{(3)}, A^{(2)}) = W(A^{(3)}+A^{(2)}) = W(A^{(2)}+A^{(1)}+A^{(2)}) = W(A^{(1)}).$$

This result means that, if we construct group codes, the minimum distance of the code is given by the minimum nonzero weight of the code points. We must include the term "nonzero" because $\phi = A^{(0)} = (0, 0, 0, \ldots 0)$ is always a code point and corresponds to $e=f$ in (36).

In view of (33) and (34) it is apparent that we can write, for the binary symbol in row e and column h of the code matrix:

$$a_h^{(e)} = b_1 a_h^{(1)} + b_2 a_h^{(2)} + b_{4h}^{(4)} + \ldots + b_{2^{k-1}} a_h^{(2^{k-1})},$$

where $e = 2^0 b_1 + 2^1 b_2 + 2^2 b_4 + 2^3 b_8 + \ldots$. (37)

Table 4 **Examples of code matrices for $g=2h$ or $g=2h-1$.**

Type 1

$A^{(0)}$	1	1	1	1	0	0	0	0	0	0
$A^{(1)}$	1	0	0	0	1	1	1	0	0	0
$A^{(2)}$	0	1	0	0	1	0	0	1	1	0
$A^{(3)}$	0	0	1	0	0	1	0	1	0	1
$A^{(4)}$	0	0	0	1	0	0	1	0	1	1

$g=5 \quad n=2\cdot\binom{5}{2}-2=18$

Type 3

$A^{(0)}$	1	1	1	1	0	0	0	0
$A^{(1)}$	1	0	0	0	1	1	1	0
$A^{(2)}$	0	1	0	0	1	0	0	1
$A^{(3)}$	0	0	1	0	0	1	0	1
$A^{(4)}$	0	0	0	1	0	0	1	0

Type 1

$A^{(0)}$	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
$A^{(1)}$	1	1	1	1	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0
A$^{(2)}$	1	0	0	0	1	1	1	0	0	0	1	1	1	0	0	0	1	1	1	0
$A^{(3)}$	0	1	0	0	1	0	0	1	1	0	1	0	0	1	1	0	1	1	0	1
$A^{(4)}$	0	0	1	0	0	1	0	1	0	1	0	1	0	1	0	1	1	0	1	1
$A^{(5)}$	0	0	0	1	0	0	1	0	1	1	0	0	1	0	1	1	0	1	1	1

$g=6 \quad n=\binom{6}{3}=20$

Type 1

$A^{(0)}$	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$A^{(1)}$	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0	0
$A^{(2)}$	1	1	1	1	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	1	1	1	1	1	1	0	0	0	0	1	1	1	1	0
$A^{(3)}$	1	0	0	0	1	1	1	0	0	0	1	1	1	0	0	0	1	1	1	0	1	1	1	0	0	0	1	1	1	0	1	1	1	0	1
$A^{(4)}$	0	1	0	0	1	0	0	1	1	0	1	0	0	1	1	0	1	1	0	1	1	0	0	1	1	0	1	1	0	1	1	1	0	1	1
$A^{(5)}$	0	0	1	0	0	1	0	1	0	1	0	1	0	1	0	1	1	0	1	1	0	1	0	1	0	1	1	0	1	1	1	0	1	1	1
$A^{(6)}$	0	0	0	1	0	0	1	0	1	1	0	0	1	0	1	1	0	1	1	1	0	0	1	0	1	1	0	1	1	1	0	1	1	1	1

$g=7 \quad n=\binom{7}{4}=35$

We therefore find that there are exactly 2^k *different types* of columns corresponding to the choice of the a terms on the right side of (37). However, we note that if all a terms on the right of (37) are zero, then $a_h^{(e)}$ is identically zero for all e. This means the entire column consists of zeroes and contributes nothing to the distance between code points. Ruling out the all zeroes as useless, we are left with (2^k-1) types of columns. It can be shown that each of these types of columns consists of 2^{k-1} zeroes and 2^{k-1} ones and hence are of the kind found necessary for construction of max min d codes in Appendix I.

We have seen previously that a choice for e in (34) and hence in (37) completely determines the b terms of (37). In view of this and the definition of the weight of $A^{(e)}$, (37) tells us that the weight of any code point is completely determined by the number of each of the (2^k-1) possible column types used to form the coding matrix. As a matter of considerable notational convenience, let us say that a choice of a terms of (37) has resulted in a column of type j if we have:

$$j=1a^{(1)}+2a^{(2)}+4a^{(4)}+\ldots.+2^{k-1}a^{(2^{k-1})}. \tag{38}$$

Letting n_j denote the number of columns of type j, we have:

$$n=\sum_{j=1}^{2^k-1} n_j. \tag{39}$$

As an example to illustrate the notion of column types, consider the following code for $k=2$, $n=5$:

e	$b_2\ b_1$	$A^{(e)}$
0	0 0	0 0 0 0 0
1	0 1	1 0 1 1 0
2	1 0	0 1 1 1 1
3	1 1	1 0 0 0 1

For the left-most column of $A^{(e)}$ we have $a^{(1)}=1$, $a^{(2)}=0$. Thus the appropriate j label for this column is $j=1$ from (38). The next column has $a^{(1)}=0$, $a^{(2)}=1$, giving $j=2$. The third column has $a^{(1)}=1$, $a^{(2)}=1$, giving $j=3$. The fourth column has $a^{(1)}=1$; $a^{(2)}=1$, giving $j=3$. The last column has $a^{(1)}=0$, $a^{(2)}=1$, giving $j=2$. Thus the j labels are, from left to right:

1, 2, 3, 3, 2.

Hence, $n_1=1$; $n_2=2$; $n_3=2$; $n=n_1+n_2+n_3=5$.

Pursuing the example somewhat further we can see that the following relations are true:

$W_0=W(A^{(0)})=0$

$W_1=W(A^{(1)})=n_1+n_3=1+2=3$

$W_2=W(A^{(2)})=n_2+n_3=2+2=4$

$W_3=W(A^{(3)})=n_1+n_2=1+2=3$.

These expressions suggest the matrix multiplication:

$$\begin{bmatrix} W_0 \\ W_1 \\ W_2 \\ W_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ n_1 \\ n_2 \\ n_3 \end{bmatrix} \tag{40}$$

We may write (40) more compactly as:

$$[W^*]=[C^*][N^*]. \tag{41}$$

The $[W^*]$ of (41) is always a $(2^k\times 1)$ matrix, $[N^*]$ is likewise $(2^k\times 1)$, whereas $[C^*]$ is always $(2^k\times 2^k)$. Because of the way we assigned superscripts e to the code points and j labels to the column types, it becomes a very simple matter to find the $[C^*]$ matrix for $(k+1)$ from the $[C^*]$ matrix for k. This is illustrated in Table 5.

Observe that the $[C^*]$ matrix is symmetric. Further observe that, except for the first row and column, each row and column contains 2^{k-1} ones and 2^{k-1} zeroes. The first row and column are really unnecessary for computation of weights since we have already agreed to set $n_0=0$, and also we already know that $W_0=0$. This row and column are merely included to show more simply the expansion from the $[C^*]$ matrix for k to the $[C^*]$ matrix for $(k+1)$. For later convenience, then, let us designate $[C]$ as the reduced matrix obtained from $[C^*]$ by deleting the first row and first column. By deleting W_0 from $[W^*]$, we form $[W]$ and by deleting n_0 from $[N^*]$ we form $[N]$. Thus, we still have the matrix multiplication relation:

$$[W]=[C][N]. \tag{42}$$

By virtue of the fact that each column of the C matrix contains 2^{k-1} ones, we have the additional restraint noted by Slepian:

$$\sum_{i=1}^{2^k-1} W_i=W=n2^{k-1}. \tag{43}$$

It is of interest that $[C]$ possesses an inverse for any k. Denoting by c_{ij} the entry in row i and column j of $[C]$ and by c_{ij}^{-1} the entry in row i and column j of $[C^{-1}]$, it can be shown that:

$$c_{ij}^{-1}=\frac{2c_{ij}-1}{2^{k-1}}, \tag{44}$$

where $[N]=[C^{-1}][W]$.

Thus a proposed set of weights must not only satisfy (43) but must also yield an integer for every n_j according to (44). Slepian arrived at this result using the theory of modular representations of mathematical groups.

Our main concern, however, is with the implications of (42). We will derive several types of max min d codes from this starting point.

First of all, consider a code for which $g=2^k$ and $n=2^k-1$ where we agree to use exactly one each of the 2^k-1 types of columns in the $[C]$ matrix. In other words, our coding matrix is exactly the $[C]$ matrix plus an additional row of all zeroes (ϕ or $A^{(0)}$). Saying it another way, our coding matrix is exactly the $[C^*]$ matrix with the left-hand column of all zeroes deleted. This means we

Table 5 **Generation of [C] for (k+1) from [C] for k.**

$[C^*]$
$k=3$

0	0	0	0	0	0	0	0
0	1	0	1	0	1	0	1
0	0	1	1	0	0	1	1
0	1	1	0	0	1	1	0
0	0	0	0	1	1	1	1
0	1	0	1	1	0	1	0
0	0	1	1	1	1	0	0
0	1	1	0	1	0	0	1

Step 1: Copy $[C^*]$ for k in upper right, upper left, and lower left quadrants of $[C^*]$ for $(k+1)$.

Step 2: Copy the complement of $[C^*]$ for k into the lower right-hand quadrant of $[C^*]$ for $(k+1)$.

$[C^*]$
$k=4$

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0
0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0
0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0
0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0
0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0
0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1
0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0
0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1
0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1
0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0

$[C]$
$k=3$

1	0	1	0	1	0	1
0	1	1	0	0	1	1
1	1	0	0	1	1	0
0	0	0	1	1	1	1
1	0	1	1	0	1	0
0	1	1	1	1	0	0
1	1	0	1	0	0	1

To form $[C]$ *from* $[C^*]$:

Delete left column and top row.

$[C]$
$k=4$

1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0
1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0
0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0
1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	1
0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	1
1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0
0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	1
0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	1
1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	0
0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0
1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	0
0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	0
1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	1

chose $[N]$ such that:

$$[N]^T=[1, 1, 1, \ldots\ldots, 1]. \tag{45}$$

Inasmuch as each row of the $[C]$ matrix contains 2^{k-1} ones and $(2^{k-1}-1)$ zeroes, the W vector becomes:

$$[W]^T=[2^{k-1}, 2^{k-1}, 2^{k-1}, \ldots\ldots, 2^{k-1}]. \tag{46}$$

Obviously all weights are equal and thus equal to the minimum weight, and hence minimum distance is equal to 2^{k-1}. Using (3) we have:

$$\min d \leqslant \frac{ng}{2[g-1]} = \frac{[2^k-1]2^k}{2[2^k-1]} = 2^{k-1}. \tag{47}$$

Thus we achieve the equality of (3) for the code just described. Such codes will be designated *Type 10*.

By deleting any one column we again achieve a max min d code. The Δ argument is identical to that for Type 2 codes. Thus, we designate as *Type 11* max min d codes these codes for which:

$$n=2^k-2$$

$$\min d=2^{k-1}-1 \tag{48}$$

$[N]$ contains (2^k-2) ones and a single zero anywhere.

By deleting any two columns we again realize a max min d code. The Δ argument is identical to that used for Type 3. Thus, we designate as *Type 12 codes* those codes for which:

$$n=2^k-3$$

$$\min d=2^{k-1}-2 \tag{49}$$

$[N]$ contains (2^k-3) ones and 2 zeroes anywhere.

Further exploitation of the properties of a group can be used to obtain max min d codes for values of n even smaller than those given for Types 10, 11, and 12. Consider the effect of starting with a $[C]$ matrix and deleting (2^u-1) consecutive columns beginning at the left, where we put the restraint $u=2, 3, 4, \ldots, (k-1)$. Put another way, consider using as an $[N]$ vector:

$$[N]^T=[\underbrace{0, 0, \ldots, 0}_{2^u-1}, \underbrace{0, 1, 1, \ldots\ldots, 1, 1}_{2^k-2^u}]. \tag{50}$$

What will happen to the weights? Considering the construction of the $[C]$ matrix, it becomes apparent that the contribution to the weight of the various code points made by the (2^u-1) leftmost columns is either 0 or 2^{u-1}. Thus, by deleting these columns we have diminished the weights by not more than 2^{u-1}. Since all weights equal 2^{k-1} for a complete $[C]$ matrix (when $[N]$ consists of all ones), we have for the incomplete case:

$$\min W=\min d=2^{k-1}-2^{u-1}$$

$$n=(2^k-1)-(2^u-1)=2^k-2^u. \tag{51}$$

The codes described by (50) and (51) will be designated as *Type 13 codes*. The Δ argument shows them to be max min d codes.

It is not obvious from the preceding discussion of Type 13 codes that we actually obtain 2^k *different* sequences from the (2^k-2^u) rightmost colums of the C matrix. The simplest way to see this is to assume the converse. If two such sequences, say $A^{(e)}$ and $A^{(f)}$, are identical, then, since $A^{(e)}+A^{(f)}$ is also a member of the group code, there would have to exist a row corresponding to $(A^{(e)}+A^{(f)})$ in the original $[C]$ matrix which had 2^{k-1} ones in the $(2^{u-1}-1)$ leftmost positions. However, from the method of construction of the $[C]$ matrix, it is apparent that each row contains either no ones or 2^{u-1} ones in the (2^u-1) leftmost positions. This guarantees, then, that there are no duplications in the $2^k\times(2^k-2^u)$ coding matrix. This is equivalent to guaranteeing that there exist k rows which are linearly independent under addition modulo 2.

A simple variation of a Type 13 code can be formed by deleting any one column from a Type 13 code. We will designate such a code as *Type 14*. It is apparent that we have for a Type 14 code the following relations:

$$n=2^k-2^u-1$$

$$\min W=\min d=2^{k-1}-2^{u-1}-1. \tag{52}$$

Application of the Δ argument gives the result that Type 14 codes are max min d.

To show that the 2^k code points of a Type 14 code are all different, we use essentially the same argument as used for Type 13. If two code points, say $A^{(e)}$ and $A^{(f)}$ are identical, they must have differed only in the deleted position of the parent Type 13 code. This would require a row of the $[C]$ matrix corresponding to $(A^{(e)}+A^{(f)})$ to contain $(2^{k-1}-1)$ ones in the (2^u-1) leftmost positions and a single one in the (2^k-2^u) rightmost positions. Since the (2^u-1) leftmost positions contain either no ones or 2^{u-1} ones, this is impossible.

There is still another variation of Type 13 codes which yields max min d codes for still smaller values of n than those discussed previously. We will call the present variation *Type 15 codes*. They are to be constructed as follows: Start with a Type 13 code which has been derived from a $[C]$ matrix by deleting the first (2^v-1) columns, where $v=2, 3, 4, \ldots, k-2$. (Note that the upper limit on v is smaller than the upper limit u, so we certainly have a Type 13 code.)

As a preliminary, we prove no code point of the Type 13 code contains all ones in its (2^k-2^v) positions. Assume the converse. Then the row of the $[C]$ matrix corresponding to the code point of all ones would have $2^{k-1}-(2^k-2^v)=(2^v-2^{k-1})$ ones in the (2^v-1) leftmost positions. Since (2^v-2^{k-1}) is negative for the restriction placed on v, this is impossible. (Indeed, *if* we permitted $v=k-1$, not only could the code point of all ones exist in the Type 13 code, but it actually would exist.)

The code points of the Type 13 code are designated $A^{(0)}$ (or ϕ), $A^{(1)}$, $A^{(2)}$, $A^{(3)}$, $\ldots$, $A^{(2^k-1)}$ as before, with $A^{(e)}$ and $A^{(f)}$ as two typical code points. We have proved that $I=(1, 1, 1, \ldots, 1, 1, 1)$ is not a member of this group. This also guarantees that, if $A^{(e)}$ belongs to the

group, then $(I+A^{(e)})$ does not belong to the group. Now consider the group code containing the following members: $A^{(0)}$, $A^{(1)}$, $A^{(2)}$, $A^{(3)}, \ldots, A^{(2^k-1)}$, I, $I+A^{(1)}$, $I+A^{(2)}$, $I+A^{(3)}, \ldots I+A^{(2^k-1)}$. Since the weight of $A^{(e)}$ is$n/2$ [see (51)] the weight of $A^{(e)}+I$ is likewise $n/2$. Since the weight of I is n, the minimum weight and hence minimum distance of the new group code is $n/2$, where n is the same as for the parent Type 13 code. For the new code, however, we have doubled the number of code points. We have then the following associations:

Type 13	*Type 15*
k	$k+1$
$n=2^k-2^v$	$n=2^k-2^v$
min $d=n/2=2^{k-1}-2^{v-1}$	min $d=n/2=2^{k-1}-2^{v-1}$

Our discussion of Type 15 codes is simplified by replacing $k+1$ by k. Thus, for Type 15 codes, we have:

$$n=2^{k-1}-2^v;\ v=2, 3, 4, \ldots\ldots, k-3$$
$$\text{min } d=2^{k-2}-2^{v-2}. \tag{53}$$

Once again the Δ argument is used to show Type 15 codes are max min d.

It is perhaps not obvious that Type 15 codes are really different from Type 13 codes. In Type 13 codes, to prove that the code points were distinct and each had weight $2^{k-1}-2^{u-1}$, we had to restrict u to be not greater than $(k-1)$. This means the smallest value of n obtainable is:

$$n=2^k-2^u=2^k-2^{k-1}=2^{k-1}. \tag{54}$$

Thus (54) means we may form a Type 13 code by retaining the 2^{k-1} rightmost columns of the $[C]$ matrix for use in the coding matrix. By contrast, Type 15 codes realize the construction of max min d codes for values of n given by (53), which is obviously smaller than (54). In reality, then, Type 15 codes are constructed by utilizing in the coding matrix the $(2^{k-1}-2^v)$ rightmost columns of the $[C]$ matrix.

By deleting any one column of a Type 15 code, we again have a max min d code which we designate Type 16. The appropriate relations are:

$$n=2^{k-1}-2^v-1$$
$$\text{min } d=2^{k-1}-2^{v-1}-1. \tag{55}$$

Summary and generalization of max min *d* code Types 10 to 16

A summary of Codes 10 through 16 is given in Table 6. Some illustrative examples of these code types are given in Tables 7a and 7b.

In a previous section, we proved that max min d codes could be constructed by adjoining any number of codes which achieved the equality in (3) to at most one code which obeyed (27). This result naturally applies to code Types 10 through 16 as well as to previous codes. Indeed, when $g=2^k$, we may adjoin certain codes from the first set and certain codes from the second set. In the interests of simplicity, however, we will restrict our attention to adjoining codes only from Types 10 through 16. The resultant generalization of code Types 10 through 16 is given in Table 8.

Comparison of Types 1 to 16 max min *d* codes and other codes

Hamming[1] has described a construction method for codes which always achieve the maximum minimum-distance when n and $g=2^k$ are chosen such that the resultant max min d is 3 or 4. For the single error-correcting Hamming codes, using the notation of the present paper, the parameter n is chosen to be the smallest integer satisfying

$$\frac{2^n}{n+1} \geqslant 2^k. \tag{56}$$

Comparison of (56) with (1) and (2) shows the maximum minimum-distance under this value of n is always 3, and Hamming codes are thus max min d codes. It should be noted that Hamming codes are always "low redundancy" codes and that the bound on min d of (1) and (2) is generally much lower than the bound on min d of (3) for low-redundancy cases. As remarked previously, the lower of the two bounds is naturally a tighter bound than either one used alone. Hamming also describes a construction method which uses a value of n which is one higher than the smallest integer satisfying (56). These codes yield a min d of 4 and hence are also max min d codes in agreement with (1) and (2). Both Hamming codes are group codes.

Plotkin[3] has described a construction method for max min d codes for certain values of g and a particular value of n dependent on g. Reference 4 gives a description of Plotkin's construction methods. Plotkin chooses the number of code points, g, so that:

$$g=8h, \text{ where } 4h-1 \text{ is a prime number}$$
$$n=4h \tag{57}$$
$$\text{min } d=2h.$$

Plotkin codes formed in accordance with (57) are shown to be max min d by using the Δ argument.

Plotkin's codes coincide with the codes of the present paper in one instance. In particular, with a Type 13 code, choosing $u=k-1$, pertinent parameters can be written:

$$g=2^k=8(2^{k-3})=8h$$
$$n=2^{k-1}=4h \tag{58}$$
$$\text{min } d=2^{k-2}=2h.$$

Written in the above form, it is clear that a Type 13 code with $u=k-1$ is a Plotkin code provided $(2^{k-1}-1)$ is a prime number. The first few values for which this is true are $k=2$, 4, 6, 8. Of course, not all Type 13 codes with $u=k-1$ are Plotkin codes, but they are max min

d codes. Conversely, not all Plotkin codes are Type 13 with $u=k-1$, but they are max min d codes.

Reed[5] has described a construction method for group codes devised by D. E. Muller. Using the notation of the present paper, Reed-Muller codes may be summarized as follows:

$n=2^a$; $a=2, 3, 4, 5, \ldots$.

$$g=2^k, \text{ where } k=\sum_{i=0}^{r}\binom{a}{i};\ r=1, 2, 3, 4, \ldots, (a-2). \quad (59)$$

$$\min d=2^{a-r}.$$

Using the Δ argument, it can be shown that Reed-Muller codes are max min d if and only if $r=1$. When $r=1$, Reed-Muller codes are identical with Type 13 codes with $u=k-1$.

Conclusions

After describing an alternative upper bound on minimum distance to that given by Hamming, we have given complete construction methods for codes which achieve the upper bound. Such codes exist for any value of g (the desired number of code points) and an infinitely large class of integers n (the number of binary symbols per

Table 6 **Summary of code types 10 to 16.**

$g=2^k$

Code Type	n	min d	$[N]$
10	2^k-1	2^{k-1}	$\underbrace{[\ 1, 1, 1, \ldots\ldots\ldots, 1, 1, 1\ ]}_{2^k-1}$
11	2^k-2	$2^{k-1}-1$	$\underbrace{[\ 0, 1, 1, \ldots\ldots\ldots, 1, 1, 1\ ]}_{2^k-2}$ *Note: Single zero can be placed anywhere in vector* $[N]$.
12	2^k-3	$2^{k-1}-2$	$\underbrace{[\ 0, 0, 1, 1, \ldots\ldots\ldots 1, 1, 1\ ]}_{2^k-3}$ *Note: Two zeroes can be placed anywhere in vector* $[N]$.
13	2^k-2^u	$2^{k-1}-2^{u-1}$	$[\underbrace{0, 0, \ldots, 0}_{2^u-1}, \underbrace{0, 1, 1, \ldots\ldots\ldots, 1, 1}_{2^k-2^u}]$ *Note:* $u=2, 3, 4, \ldots\ldots, k-1$.
14	2^k-2^u-1	$2^{k-1}-2^{u-1}-1$	$[\underbrace{0, 0, \ldots, 0}_{2^u-1}, *, \underbrace{0, 0, 1, 1, \ldots\ldots, 1, 1}_{2^k-2^u-1}]$ *Note 1:* $u=2, 3, 4, \ldots\ldots\ldots, k-1$. *Note 2: Starred zero can be placed anywhere in* 2^k-2^u *rightmost positions of* $[N]$ *vector.*
15	$2^{k-1}-2^v$	$2^{k-2}-2^{v-1}$	$[\underbrace{0, 0 \ldots\ldots, 0}_{2^{k-1}+2^v-1}, \underbrace{0, 1, 1, \ldots\ldots, 1, 1}_{2^{k-1}-2^v}]$ *Note:* $v=2, 3, 4, \ldots\ldots, k-3$.
16	$2^{k-1}-2^v-1$	$2^{k-2}-2^{v-1}-1$	$[\underbrace{0, 0, \ldots\ldots, 0, 0}_{2^{k-1}-2^v-1}, *, \underbrace{0\ 1, 1, \ldots\ldots, 1, 1}_{2^{k-1}-2^v}]$ *Note 1:* $v=2, 3, 4, \ldots\ldots, k-3$. *Note 2: Starred zero can be placed anywhere in* $2^{k-1}-2^v$ *rightmost positions of* $[N]$ *vector.*

code point). The class of integers n for which construction of maximum minimum-distance codes is achieved depends on the desired value of g. In all, sixteen different code types have been described and the appropriate proof of the maximum minimum-distance property presented. Three code types are devoted to the case $g=2h-1$, six types to the case $g=2h$, and seven types to the case $g=2^k$. It has been shown that these construction methods coincide only in a very few special cases with the results of Hamming and those of Plotkin. The present results are thus complementary to the only previous contributions to the construction problem of which the author is aware.

Appendix I: Derivation of an upper bound on minimum distance of a code

The bound to be derived depends on the simple fact that the minimum distance cannot exceed the average distance. The average distance is, of course, the ratio of the sum of all distances to the number of distances. Let the g code points (each an n-binary symbol-vector) be designated by $A^{(j)}$; $1 \leqslant j \leqslant g$. The number of distances is equivalent to the combination of g things taken 2 at a time, or

$$\text{Number of distances} = \binom{g}{2} = g\frac{(g-1)}{2}. \qquad \text{(I.1)}$$

Table 7a **Examples of code types 10 to 16.**

$k=3$

Code Type	n	min d	*Remarks*
10	7	4	Single error correcting, double error detecting (Hamming)
11	6	3	Single error correcting (Hamming)
12	5	2	Single error detecting
13	4	2	$u=2$; Single error detecting (Parity Check)
14	3	1	$u=2$; No redundancy

Type 10

	Code Matrix							[N]
$A^{(0)}$	0	0	0	0	0	0	0	
$A^{(1)}$	1	0	1	0	1	0	1	1
$A^{(2)}$	0	1	1	0	0	1	1	1
$A^{(3)}$	1	1	0	0	1	1	0	1
$A^{(4)}$	0	0	0	1	1	1	1	1
$A^{(5)}$	1	0	1	1	0	1	0	1
$A^{(6)}$	0	1	1	1	1	0	0	1
$A^{(7)}$	1	1	0	1	0	0	1	1

Type 11

	Code Matrix						[N]
$A^{(0)}$	0	0	0	0	0	0	
$A^{(1)}$	1	0	1	0	1	0	1
$A^{(2)}$	0	1	1	0	0	1	1
$A^{(3)}$	1	1	0	0	1	1	1
$A^{(4)}$	0	0	0	1	1	1	1
$A^{(5)}$	1	0	1	1	0	1	1
$A^{(6)}$	0	1	1	1	1	0	1
$A^{(7)}$	1	1	0	1	0	0	0

Type 12

	Code Matrix					[N]
$A^{(0)}$	0	0	0	0	0	
$A^{(1)}$	1	0	1	0	1	0
$A^{(2)}$	1	0	0	1	1	0
$A^{(3)}$	0	0	1	1	0	1
$A^{(4)}$	0	1	1	1	1	1
$A^{(5)}$	1	1	0	1	0	1
$A^{(6)}$	1	1	1	0	0	1
$A^{(7)}$	0	1	0	0	1	1

Type 13

	Code Matrix				[N]
$A^{(0)}$	0	0	0	0	
$A^{(1)}$	0	1	0	1	0
$A^{(2)}$	0	0	1	1	0
$A^{(3)}$	0	1	1	0	0
$A^{(4)}$	1	1	1	1	1
$A^{(5)}$	1	0	1	0	1
$A^{(6)}$	1	1	0	0	1
$A^{(7)}$	1	0	0	1	1

Type 14

	Code Matrix			[N]
$A^{(0)}$	0	0	0	
$A^{(1)}$	1	0	1	0
$A^{(2)}$	0	1	1	0
$A^{(3)}$	1	1	0	0
$A^{(4)}$	1	1	1	0
$A^{(5)}$	0	1	0	1
$A^{(6)}$	1	0	0	1
$A^{(7)}$	0	0	1	1

The sum of all distances can be written:

$$D=\sum_{e,f}\left[\sum_{i=1}^{n}\left[a_i^{(e)}-2a_i^{(e)}\,a_i^{(f)}+a_i^{(f)}\right]\right], \qquad \text{(I.2)}$$

where the summation on e and f is for all e such that $1\leqslant e\leqslant g-1$ and all f such that $f>e$. Thus, the leftmost summation sign implies $\binom{g}{2}$ terms, one for each distance. By interchanging the order of summation (I.2) can be written:

$$D=\sum_{i=1}^{n}\left[\sum_{e,f}\left[a_i^{(e)}-2a_i^{(e)}\,a_i^{(f)}+a_i^{(f)}\right]\right]=\sum_{i=1}^{n}d_i\,. \qquad \text{(I.3)}$$

Table 7b **Examples of code types 10 to 16.**

$k=6$

Code Type	n	min d	*Remarks*
10	63	32	
11	62	31	
12	61	30	
13	60	30	$u=2$
	56	28	$u=3$
	48	24	$u=4$
	32	16	$u=5$
14	59	29	$u=2$
	55	27	$u=3$
	47	23	$u=4$
	31	15	$u=5$
15	28	14	$v=2$
	24	12	$v=3$
16	27	13	$v=2$
	23	11	$v=3$

Table 8 **Generalization of code types 10 to 16.**

$t=1, 2, 3, 4, \ldots$

n	*Code Type:* 10	11	12	13	14	15	16
$(2^k-1)t$	t	0	0	0	0	0	0
$(2^k-1)t-1$	$t-1$	1	0	0	0	0	0
$(2^k-1)t-2$	$t-1$	0	1	0	0	0	0
$(2^k-1)t-2^u$	$t-1$	0	0	0	1	0	0
$(2^k-1)t-2^{k-1}-2^v$	$t-1$	0	0	0	0	0	1

Note: $u=2, 3, 4, \ldots, k-1$
$v=2, 3, 4, \ldots, k-3$

Now each d_i in (I.3) is obviously non-negative, so we can write:

$$(\max D)\leqslant n(\max d_i)\,. \qquad \text{(I.4)}$$

Using the definition of average distance we have:

$$\text{maximum average distance} = (\max \overline{D})$$

$$=(\max D)\Big/\binom{g}{2}\leqslant n(\max d_i)\Big/\binom{g}{2}. \qquad \text{(I.5)}$$

Therefore:

$$\text{maximum minimum-distance}\leqslant n(\max d_i)\Big/\binom{g}{2}, \qquad \text{(I.6)}$$

where from (I.3):

$$d_i=\sum_{e,f}\left[a_i^{(e)}-2a_i^{(e)}\,a_i^{(f)}+a_i^{(f)}\right]. \qquad \text{(I.7)}$$

In the above expression, a term such as $a_i^{(h)}$, where $1\leqslant h\leqslant g$, will appear $(g-1)$ times since $A^{(h)}$ participates in $(g-1)$ distances. A term such as $-2a_i^{(p)}\,a_i^{(q)}$ will appear only once, but there will be $\binom{g}{2}$ terms of this form corresponding to the $\binom{g}{2}$ distances. Let column i of any coding matrix have x ones and $(g-x)$ zeroes in the i^{th} column. Realizing that each term in (I.7) such as $a_i^{(h)}$ or $a_1^{(p)}\,a_i^{(q)}$ is either zero or one, we can write (I.7) as:

$$d_i=(g-1)x-2\binom{x}{2}=gx-x^2\,,$$

$$\text{where } x=0, 1, 2, 3, \ldots, g\,. \qquad \text{(I.8)}$$

Thus we simply seek that integer x which will maximize $(gx-x^2)$. Two cases must be treated.

(1) If $g=2h$, then $(gx-x^2)$ has a simple maximum at $x=g/2$. Thus:

$$\max d_i=g\left[\frac{g}{2}\right]-\left[\frac{g}{2}\right]^2=\frac{g^2}{4}\,.$$

(2) If $g=2h-1$, then $(gx-x^2)$ attains a maximum value for $x=(g-1)/2$ or $x=(g+1)/2$. For either value of x, the maximum value is:

$$\max d_i=g\left[\frac{g-1}{2}\right]-\left[\frac{g-1}{2}\right]^2$$

$$=g\left[\frac{g+1}{2}\right]-\left[\frac{g+1}{2}\right]^2=\frac{g^2-1}{4}\,.$$

Using these results in (I.6) we have:

For $g=2h$:

$$\text{maximum minimum-distance}\leqslant\frac{ng^2}{4\binom{g}{2}}=\frac{ng}{2(g-1)}$$

For $g=2h-1$: (I.9)

$$\text{maximum minimum-distance}\,\frac{n(g^2-1)}{4\binom{g}{2}}=\frac{n(g+1)}{2g}\,.$$

References

1. R. W. Hamming, "Error Detecting and Error Correcting Codes," *Bell System Technical Journal,* **29,** 147-160 (1950).
2. D. Slepian, "A Class of Binary Signalling Alphabets," *Bell System Technical Journal,* **35,** 203-234 (1956).
3. M. Plotkin, "Binary Codes with Specified Minimum Distance," University of Pennsylvania Research Division Report 51-20, January, 1951.
4. D. D. Joshi, "A Note on Upper Bounds for Minimum Distance Codes," *Information and Control,* **1,** 289-295 (1959).
5. I. S. Reed, "A Class of Multiple-Error-Correcting Coding and Decoding Schemes," *IRE Trans. Information Theory,* **4,** 38-49 (1954).
6. L. Calabi and H. G. Haefeli, "A Class of Binary Systematic Codes, etc.," *IRE Trans. Circuit Theory,* **CT-6** (May 1959).
7. H. W. Kautz, "A Class of Multiple-Error-Correcting-Codes for Data Transmission and Recording," Tech. Report No. 5, CRI 2124, Stanford Research Institute, Menlo Park, Calif.
8. E. Prange, "Some Cyclic Error Correcting Codes," ASTIA Doc. No. AD152386, Air Force Cambridge Research Center, April 1958.
9. N. Honda, "The Sequential Error-Correcting Code," *Science Reports of the Research Institute, Tohoku University,* **Series B, 8,** No. 3, 1956.

Revised manuscript received September 17, 1959