Chapter 27

MSMYP7 Coding Theory

(27.1) Basic Results

(27.1.1) Detecting & Correcting Errors

The purpose of a code is to send a message via a 'noisy' channel in which it is likely to be corrupted. The message should be received perfectly in tact*. Most of the elementary work in coding theory was conducted by Richard W. Hamming in the 1940s, the motivation being to detect and correct errors occurring in electrical computers, mostly caused by the failure of valves.

A simple code is a repetition code. When the message is "yes" or "no"—encoded as "0" and "1"—one may send the message "00000" for "no" and "11111" for "yes". If "11001" was received it would be corrected to "11111"—its nearest neighbour—and interpreted as "yes".

Definition 1 *A code is a set of codewords or vectors, each corresponding to a message. In a block code all the codewords are the same length.*

Definition 2 *The alphabet of a code is the set of all symbols used in the code. When this set has cardinality q the code is said to be a q-ary code.*

Most commonly the alphabet is a field, F_q, allowing arithmetic to be performed. A code of length n is then a subset of F_q^n giving the obvious interpretation of vector spaces. It is natural to ask as to how frequently errors occur. It is assumed that

- Each symbol in a codeword is equally likely to become erroneous and does so with probability $p < \frac{1}{2}$.
- Of the q symbols available, they each appear erroneously with equal probability.

Hence the probability that a particular position becomes erroneous with a particular symbol is $\frac{p}{q-1}$.

Consider the two codes given in Table 27.1.1. Code 1 gives a simple encoding of the message, but if an error occurs then it cannot be detected let alone corrected. Code 2 adds a check digit—the third digit is appended to make the number of 1s even. A single error can now be detected, but not corrected. Code 3 repeats the encoding after the check digit. A single error can now be detected since

- If the error is in the message the check bit will show which message is correct.
- If the check digit is incorrect then the messages will agree.

Definition 3 *The Hamming distance between vectors in F_q^n is the number of places in which they differ. For vectors $\mathbf{a}$ and $\mathbf{b}$ this is represented $d(\mathbf{a}, \mathbf{b})$.*

*Cryptographic codes are not discussed here.

Message	Code 1	Code 2	Code 3
N	00	000	00000
E	01	011	01101
S	10	101	10110
W	11	110	11011

Table 1: Using a check digit and repetition to encode a message.

Lemma 4 *The Hamming distance is a metric on the space F_q^n.*

Proof. Trivially $d(\mathbf{a},\mathbf{a}) \geqslant 0$ with equality precisely when $\mathbf{a} = \mathbf{b}$. Also, $d(\mathbf{a},\mathbf{b}) = d(\mathbf{b},\mathbf{a})$ which leaves the triangle inequality. Changing **a** to **b** then **b** to **c** must certainly take more changes than **a** directly to **c**. □

The Hamming distance formalises the concept of 'nearest neighbour' decoding. To what extent this works is dependent on the smallest distance between codewords: this is called the minimal distance of the code C and is denoted $d(C)$. Therefore

$$d(C) = \min \{ d(\mathbf{x},\mathbf{y}) \mid \mathbf{x},\mathbf{y} \in C,\ \mathbf{x} \neq \mathbf{y} \}$$

Theorem 5 1. *If $d(C) \geqslant s+1$ then C can detect s errors.*

2. *If $d(C) \geqslant 2t+1$ then C can correct t errors.*

Proof. 1. Let **c** be the codeword that is sent. If $s+1$ errors occur then it can change into a different codeword. Hence up to s errors can be detected.

2. Suppose $d(C) \geqslant 2t+1$ and that when **x** is transmitted the vector **y** is received in which at most t errors have occurred. Let **z** be another codeword then $d(\mathbf{x},\mathbf{z}) \geqslant 2t+1$. Furthermore $d(\mathbf{y},\mathbf{z}) \geqslant t+1$ since putting $d(\mathbf{y},\mathbf{z}) < t+1$ gives by the triangle inequality

$$d(\mathbf{x},\mathbf{z}) \leqslant d(\mathbf{x},\mathbf{y}) + d(\mathbf{y},\mathbf{z}) \leqslant t + t < 2t+1$$

contradicting the minimal distance. Hence **x** is the nearest codeword to **y**, meaning that up to t errors can be corrected. □

Even though in the first case it would be counterintuitive to detect s errors—being one away from a different codeword suggests a correction—this is the nature of error detection. Consider receiving "101" from the code $\{000, 111\}$. From this theorem it is evident that a code C with minimal distance d can be used either to detect $d-1$ errors or to correct $\left\lfloor \frac{d-1}{2} \right\rfloor$ errors, but not both.

(27.1.2) Features Of A Code

Parameters

It has already been seen that a code C of length n may be constructed from an alphabet of size q producing $M = |C|$ codewords. Where the minimum distance is d this may be called a $q-(n,M,d)$ code. Different values are required under different circumstances.

- d should be increased for better correction.
- Increasing n and q results in a higher cost and time taken to send messages.
- Increasing M allows more information to be sent.

The main problem in coding theory is to maximise M and d while minimising n and q. q, n, and d are generally dictated by hardware considerations, so it is usual to seek to maximise M.

Definition 6 *For a $q-(n, M, d)$ code, denote the maximum number of codewords attainable for given n and d by $A_q(n, d)$.*

Two trivial codes are the repetition code which has the greatest possible minimal distance. At the other extreme is the code with all possible codewords which, of course can neither correct nor detect errors. For this latter code $A_q(n, 1) = q^n$ is quite clear.

Example 7 *$A_q(n, n) = q$ if and only if the code in question is the (q-ary) repetition code.*

Proof. Solution If C is the repetition code then

$$C = \{(0, 0, \ldots, 0), (1, 1, \ldots 1), \ldots, (q-1, q-1, \ldots, q-1)\}$$

so that $M = q$ and $d = n$. Hence $A_q(n, d) = A_q(n, n)$ and trivially this is equal to q.

Conversely, suppose that $A_q(n, n) = q$ then $d = n$. If $M > q$ then two codewords must agree in at least one place, contradicting $d = n$ so $M = q$. Therefore all codewords must be different in all places, i.e. C must be the repetition code. □

Equivalence Of Codes

Provided the codewords of a code are changed in a consistent way, the properties of the code remain unchanged. In particular

- Re-ordering some of the co-ordinates in all of the codewords (e.g. swapping places 2 and 5).
- Re-labelling the alphabet for some of the co-ordinates for all of the codewords (e.g. switching the 1s and 0s in place 3).

Lemma 8 *Any code is equivalent to a code that contains the zero vector as a codeword.*

Proof. Take any code that does not contain the zero vector and choose some codeword $\mathbf{x} = (x_1, x_2, \ldots, x_n)$. Simply re-label the co-ordinates for which $x_i \neq 0$. □

This lemma can be quite useful as the zero vector can be a good place to start when proving other results.

Example 9 $A_2(5, 3) = 4$.

Proof. Solution To show this, try to construct a binary $(5, M, 3)$ code with $M \geqslant 4$. The idea is to deduce that M must be 4. Without loss of generality $\mathbf{0} = 00000 \in C^{\dagger}$ Now, the minimal distance is 3 and therefore any other codeword must have at least three 1s and so must be 00111 and 11100 which differ in 4 places and so both are acceptable. Finally, the only codeword of distance at least 3 from both of these is 11011, and hence the binary $(5, M, 3)$ code is determined uniquely as

$$C = \{00000, 00111, 11100, 11011\}$$ □

[†] The duality between codewords and vectors has already been drawn. Note that it is sometimes convenient to write "$x_1x_2 \ldots x_n''$ rather than $(x_1, x_2, \ldots, x_n)$ for a codeword. While in the context of coding theory the former is more correct the expression $q-1q-1\ldots q-1$ is unclear, hence the optional use of vector notation.

Definition 10 *Let* $F_2 = \{0, 1\}$ *so that* $F_2^n = \{x_1x_2 \dots x_n \mid x_i \in F_2\}$. *For* $\mathbf{x} \in F_2^n$ *and* $\mathbf{y} \in F_2^n$ *define*

$$\begin{aligned} \mathbf{x} + \mathbf{y} &= (x_1 + y_1, x_2 + y_2, \dots, x_n + y_n) \\ \mathbf{x} \cap \mathbf{y} &= (x_1y_1, x_2y_2, \dots, x_ny_n) \end{aligned}$$

where arithmetic is performed modulo 2.

Define also the weight of a vector, $w(\mathbf{x}) = \sum_{i=1}^{n} x_i$.

Note that while the definitions of addition and intersection generalise readily to $q > 2$ the weigh actually refers to the number of non-zero elements in the vector.

Lemma 11 *For a binary code* $d(\mathbf{x}, \mathbf{y}) = w(\mathbf{x} + \mathbf{y})$ *while in general* $d(\mathbf{x}, \mathbf{y}) = w(\mathbf{x} - \mathbf{y})$.

Proof. In the binary case $\mathbf{x}$ and $\mathbf{y}$ differ precisely in the places where $x_i + y_i = 1$. Hence the result.

Generally, $\mathbf{x}$ and $\mathbf{y}$ differ in the ith place $\Leftrightarrow x_i \neq y_i \Leftrightarrow x_i - y_i \neq 0 \Leftrightarrow \mathbf{x} - \mathbf{y} \neq \mathbf{0}$ in the ith place. □

Lemma 12 *For binary codes* $d(\mathbf{x}, \mathbf{y}) = w(\mathbf{x}) + w(\mathbf{y}) - 2w(\mathbf{x} \cap \mathbf{y})$.

Proof. The distance between $\mathbf{x}$ and $\mathbf{y}$ is precisely the number of places in which $\mathbf{x}$ and $\mathbf{y}$ differ. Now, $w(\mathbf{x}) + w(\mathbf{y}$ counts the number of places in which $\mathbf{x}$ is 1 and $\mathbf{y}$ is 1 and twice the number of places where both are 1. Hence the result. □

A parity check can be constructed by adding a 0 or 1 depending on whether the weight is even or odd. The weight of the new vector is now even and by the above lemma so is the distance between any two checked vectors.

Theorem 13 *When* d *is odd a binary* (n, M, d) *code exists if and only if a binary* $(n + 1, M, d + 1)$ *code exists.*

Proof. ($\Rightarrow$) Let C be a binary (n, M, d) code and let d be odd. Construct from C the code C' by appending an overall parity check to each codeword. Trivially C' has M codewords of length $n + 1$. Now, $w(\mathbf{x}')$ is even for all codewords $\mathbf{x}'$ in C' and hence by Lemma 12 $d(\mathbf{x}', \mathbf{y}')$ must be even for all $\mathbf{x}'$ and $\mathbf{y}'$ in C' and hence $d(C')$ is even. Certainly $d \leqslant d(C') \leqslant d + 1$ and since d is odd $d(C') = d + 1$.

($\Leftarrow$) Let C be a binary $n + 1, M, d + 1)$ code with d odd. Select codewords $\mathbf{x}$ and $\mathbf{t}$ such that $d(\mathbf{x}, \mathbf{y}) = d + 1$ and select one co-ordinate in which they differ. Remove this co-ordinate from all codewords, hence the result. □

Furthermore if d is even and C is a $2 - (n + 1, M, d + 1)$ code then choose two codewords at distance $d + 1$ and remove from all codewords one co-ordinate which differs in the chosen codewords. The result is a $2 - (n, M, d)$ code with odd d.

Corollary 14 $A_2(n, d) = A_2(n + 1, d + 1)$.

Proof. Let C be a $2 - (n, M, d)$ code with $M = A_2(n, d)$. From Theorem 13 there exists a $2 - (n + 1, M, d + 1)$ code with $M = A_2(n, d)$, but certainly $M \leqslant A_2(n + 1, d + 1)$ and therefore $A_2(n, d) \leqslant A_2(n + 1, d + 1)$.

Conversely, let C' be a $2 - (n + 1, M', d + 1)$ code with $M' = A_2(n + 1, d + 1)$ then there exists a $2 - (n, M', d)$ code with $M' = A_2(n + 1, d + 1)$. But then $M' \leqslant A_2(n, d)$ giving $A_2(n, d) \geqslant A_2(n + 1, d + 1)$. □

(27.1.3) Sphere Packing

If C is a t error correcting code and $\mathbf{b}$ and $\mathbf{x}$ be codewords. It is quite clear that

$$\{\mathbf{x} \mid d(\mathbf{b}, \mathbf{x}) \leqslant t\} \cap \{\mathbf{x} \mid d(\mathbf{c}, \mathbf{x}) \leqslant t\} = \varnothing \tag{15}$$

which has the obvious geometrical interpretation as spheres.

Definition 16 *Let* $\mathbf{c}$ *be a codeword, then the sphere of radius t centred at* $\mathbf{c}$ *is* $S_t(\mathbf{c}) = \left\{\mathbf{x} \in F_q^n \mid d(\mathbf{c}, \mathbf{x}) \leqslant t\right\}$.

Observe that since F_q^n is a discrete set a sphere is rather like an onion, with different vectors at different integer distances. Now,

- There is 1 point at distance 0 from $\mathbf{c}$.
- There are $n(q-1)$ points at distance 1 from $\mathbf{c}$.
- There are $\frac{1}{2}n(n-1)(q-1)^2$ points at distance 2 from $\mathbf{c}$.
- There are $\frac{n!}{(n-t)!t!}(q-1)^t$ points at distance t from $\mathbf{c}$.

Clearly this gives a total of

$$|S_t(\mathbf{c})| = \sum_{i=0}^{t} \binom{n}{i} (q-1)^i \tag{17}$$

Theorem 18 (The Sphere Packing Bound) $M \sum_{i=0}^{t} \binom{n}{i} (q-1)^i \leqslant q^n$.

Proof. From equations (15) and (17) it is clear that there are at least $M \sum_{i=1}^{n} \binom{n}{i} (q-1)^i$ vectors in F_q^n. However, some vectors may not be included in any sphere, so since $|F_q^n| = q^n$ the result is shown. □

Equivalently one may write

$$M \leqslant \frac{q^n}{\sum_{i=1}^{n} \binom{n}{i} (q-1)^i}$$

giving an upper bound on the number of codewords that can be used. Clearly larger spheres—which pack poorly but can correct many errors—yields a low upper bound on M. Generally $t < \frac{1}{4}n$ is acceptable.

Perfect Codes & Balanced Block Designs

A code is said to be perfect if $A_q(n, d)$ attains the sphere packing bound. Trivial examples of a perfect code are of course those with only 1 codeword, and those with all possible codewords. A less obvious example is the binary repetition code, provided it is of odd length (if even the vector that has as many 0s as 1s cannot be corrected). Other prefect codes may be constructed by means of a balanced block design.

Definition 19 *A balanced block design is a set S of points v and a family of b subsets of S, called blocks.*

1. *Each block contains k points.*
2. *Each point is contained in r blocks.*
3. *Any pair of points are in λ blocks.*

This is referred to as a (b, v, r, k, λ) design.

One such design may be constructed from the set $S = \{0, 1, 2, 3, 4, 5, 6\}$ by taking cyclic subsets

$$\{0,1,4\} \quad \{1,2,5\} \quad \{2,3,6\} \quad \{3,4,0\} \quad \{4,5,1\} \quad \{5,6,2\} \quad \{6,0,3\}$$

This is a $(7, 7, 3, 3, 1)$ design. Seven codewords in F_2^7 are now constructed so that the ith co-ordinate in codeword j indicates whether i is in block j. The complements of these are then taken, and $\mathbf{0}$ and $\mathbf{1}$ are added. This gives

$$\begin{array}{rlcrl}
\mathbf{a}_1 &= 1101000 & \qquad & \mathbf{b}_1 &= 0010111 \\
\mathbf{a}_2 &= 0110100 & & \mathbf{b}_2 &= 1001011 \\
\mathbf{a}_3 &= 0011001 & & \mathbf{b}_3 &= 1100110 \\
\mathbf{a}_4 &= 1001100 & & \mathbf{b}_4 &= 0110011 \\
\mathbf{a}_5 &= 0100110 & & \mathbf{b}_5 &= 1011001 \\
\mathbf{a}_6 &= 0010011 & & \mathbf{b}_6 &= 1101100 \\
\mathbf{a}_7 &= 1001001 & & \mathbf{b}_7 &= 0110110 \\
\mathbf{0} &= 0000000 & & \mathbf{1} &= 1111111
\end{array}$$

Clearly $M = 16$ and $n = 7$ so this would be a perfect code if $d = 3$ (giving $t = 1$). Observe that

$$d(\mathbf{x}, \mathbf{0}) \geqslant 3 \ \forall \mathbf{x} \in C \qquad d(\mathbf{x}, \mathbf{1}) \geqslant 3 \ \forall \mathbf{x} \in C \qquad d(\mathbf{a}_i, \mathbf{b}_i) = 7$$

Now, since $\lambda = 1$ any two points belong to precisely one block, so that $w(\mathbf{a}_i \cap \mathbf{a}_j) = 1$. But $w(\mathbf{a}_i) = 3 \ \forall i$ so by Lemma 12 $d\left(\mathbf{a}_i, \mathbf{a}_j\right) = 3 + 3 - 2 = 4$.

Since the $\mathbf{b}$s are equivalent to the $\mathbf{a}$s (as they are obtained by permuting symbols) it must be the case that $d\left(\mathbf{a}_i, \mathbf{a}_j\right) = d\left(\mathbf{b}_i, \mathbf{b}_j\right) = 4$.

Finally, $\mathbf{a}_i$ and $\mathbf{b}_j$ differ wherever $\mathbf{a}_i$ and $\mathbf{a}_j$ agree. But $\mathbf{a}_i$ and $\mathbf{a}_j$ differ in 4 places and therefore must agree in 3, giving $d\left(\mathbf{b}_i, \mathbf{b}_j\right) = 3$.

Hence the minimal distance for this code is $d = 3$. When $d = 3$ $t = 1$ so using the sphere packing formula gives lower bound 16 for M. But there are 16 codewords in the above code, hence it is perfect.

It turns out that some of the parameters of block designs are redundant in that they can be deduced from the others. Consider block-point pairs where the point is in the block.

- There are b blocks, each with k points in them, hence there are bk such pairs.
- There are v points and each point is a member of r blocks, hence there are vr such pairs.

Hence $bk = vr$. Moreover, consider all triples of a block and a pair of points in the block.

- There are b blocks, each with k points in them and thus containing ${}^kC_2$ pairs of points. This gives $\frac{k(k-1)}{2}b$ such triples.
- There are ${}^vC_2$ pairs of points and λ blocks containing any given pair, so there are $\frac{v(v-1)}{2}\lambda$ such triples.

Combining these relationships one may deduce that

$$r = \frac{v-1}{k-1}\lambda \quad \text{and} \quad b = \frac{v(v-1)}{k(k+1)}\lambda$$

so it is sufficient to specify only v, b, and λ. Popular designs include those where $b = v$ (and hence $r = k$) which are said to be symmetric. A symmetric $(n^2 + n + 1, n + 1, 1)$ design is called a projective plane of order n.

(27.1.4) Finite Fields

De nition & Results

As the alphabet of most codes is a finite field it is worth making a few points about finite fields. A field is a set F equipped with 2 binary operations, '+' and '·' say, and distinguished elements 0 and 1 such that

- '+' is associative, i.e. $(a + b) + c = a + (b + c)$ for all $a, b, c \in F$.
- '+' is commutative, i.e. $a + b = b + a$ for all $a, b \in F$.
- $a + 0 = 0 + a = a$ for all $a \in F$.
- $\forall a \in A\ \exists a' \in A$ such that $a + a' = a' + a = 0$.
- $\cdot$ is associative.
- $\cdot$ is commutative.
- $a \cdot 1 = 1 \cdot a = a$ for all $a \in F$.
- $\cdot$ has inverses except for the distinguished element 0.

Results about fields can be found in Chapter **??**.

Theorem 20 *Every finite field has prime power order and, moreover, for every prime power there exists a field of that order.*

When p is prime $\mathbb{Z}_p$ is a field. Clearly $\mathbb{Z}_4$ is not a field, but 4 is a prime power and so a field of order 4 does exist. This field is in fact the Galois field of order 4, denoted GF (4).

Example: The ISBN Code

Every book has a unique ISBN code, and these are constructed in such a way as to detect and correct certain errors. The code consists of 10 digits from GF (11) though 'X' is used instead of '10'. The last digit of the ISBN is calculated from the preceding 9 so that where the ISBN is $a_0 a_1 \ldots a_X$,

$$a_X = 0a_0 + 1a_1 + 2a_2 + \cdots + 9a_9 \pmod{11}$$

But working modulo 11, $-1 = X$ and hence for a valid ISBN

$$0a_0 + 1a_1 + 2a_2 + \cdots + 9a_9 + Xa_X \equiv 0 \pmod{11}$$

Theorem 21 *The ISBN code can detect single errors.*

Proof. Suppose that a correct ISBN is $a_0 a_1 \ldots a_X$ and that one of the digits, a_i is changed to a'_i. Hence using the checking formula

$$\begin{aligned} a_1 + 2a_2 + \cdots + ia'_i + \cdots + Xa_X &\equiv a_1 + 2a_2 + \cdots + Xa_X - ia_i + ia'_i \\ &\equiv i\left(a'_i - a_i\right) \pmod{11} \end{aligned}$$

Now, $i \neq 0$ modulo 11 and neither is $a'_i - a_i$ hence an error is detected. □

Theorem 22 *The ISBN code can detect transposition errors.*

Proof. Let $a_0a_1\dots a_X$ is a correct ISBN and then that a_i and a_j are transposed. Hence using the checking formula

$$\begin{aligned} a_1 + 2a_2 + \cdots + ia_j + \cdots + ja_i + \cdots + Xa_X &\equiv a_1 + 2a_2 + \cdots + Xa_X - ia_i - ja_j + ia_j + ja_i \\ &\equiv (i-j)(a_j - a_i) \pmod{11} \end{aligned}$$

Now, $i \neq j$ and it may be assumed that $a_i \neq a_j$ (as otherwise even when transposed no error has been made) all modulo 11. Hence the sum is non-zero meaning that an error has been detected. □

Theorem 23 *The ISBN code can reinstate a single illegible digit.*

Proof. Suppose $a_0a_1\dots a_X$ is an ISBN but that a_i is unknown (i is known). Then in the checking formula

$$\begin{aligned} -ia_i &\equiv a_1 + 2a_2 + \cdots + (i-1)a_{i-1} + (i+1)a_{i+1} + \cdots + Xa_X \\ &\equiv k \pmod{11} \quad \text{say. And hence} \end{aligned}$$

$$a_i \equiv (-i)^{-1}k \pmod{11}$$ □

(27.1.5) Vector Spaces

A vector space may be defined over a finite field rather than $\mathbb{R}$ or $\mathbb{C}$ has is 'usual'. Define

$$V(n,q) = \mathrm{GF}(q)^n = \{(x_1x_2\dots x_n)\ midx_i \in \mathrm{GF}(q)\}$$

where commas may be used or omitted at will. Addition and scalar multiplication are defined in the obvious way;

$$\begin{aligned} (x_1x_2\dots x_n) + (y_1y_2\dots y_n) &= (x_1+y_1, x_2+y_2, \dots, x_n+y_n) \\ \lambda\,(x_1x_2\dots x_n) &= (\lambda x_1, \lambda x_2, \dots \lambda x_n) \end{aligned}$$

where $\lambda \in \mathrm{GF}(q)$. Since the vector space is taken over a finite field it has in it a finite number of vectors—namely q^n. Other properties and results regarding vector spaces which should be recalled include

- A subspace C of $V(n.q)$ (written $C \leqslant V$) is a non-empty subset of $V(n,q)$ that is closed under addition and scalar multiplication.
- Vectors $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$ are linearly independent if $\sum_{i=1}^{n} \lambda_i \mathbf{v}_i = \mathbf{0}$ if and only if $\lambda_i = 0$ for all i.
- The set $\{\lambda_1\mathbf{v}_1 + \lambda_2\mathbf{v}_2 + \cdots + \lambda_k\mathbf{v}_k \mid \lambda_i \in \mathrm{GF}(q)\}$ is the subspace of $V(n,q)$ that is spanned by $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k\}$.
- If $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k\}$ span a subspace C of $V(n,q)$ and are linearly independent then they are said to be a basis for C. In this case $|C| = q^k$ because of linear independence.

(27.2) Linear Codes

(27.2.1) Benefits Of Linear Codes

A linear code is simply a subspace C of a vector space $V(n,q)$ and if C has dimension k then it is called a $[n,k,d]$ code where d is the minimal distance. Equivalently this is also referred to as an (n, q^k, d) code—note the use of parenthesis instead of brackets and note also the converse is not true. The minimal distance retains its definition, but for a linear code calculating the minimal distance is relatively easy.

Theorem 24 *If C is a linear code then $d(C) = w(C)$ where*

$$w(C) = \min \{w(\mathbf{x}) \mid \mathbf{x} \in C\}$$

Proof. First of all observe that $d(\mathbf{x}, \mathbf{y}) = w(\mathbf{x} - \mathbf{y})$ since these two codewords differ in precisely those places where $\mathbf{x} - \mathbf{y}$ is 1. Let $\mathbf{x}$ and $\mathbf{y}$ be codewords such that $d(\mathbf{x}, \mathbf{y}) = d(C)$ then

$$d(C) = d(\mathbf{x}, \mathbf{y}) = w(\mathbf{x} - \mathbf{y}) \geqslant w(C)$$

Conversely, there must exists a codeword $\mathbf{z}$ such that $w(C) = w(\mathbf{z})$. But then

$$w(C) = w(\mathbf{z}) = d(\mathbf{z}, \mathbf{0}) \geqslant d(C)$$

Hence $w(C) = d(C)$. □

This is a clear advantage of linear codes. Other advantages include the possibility of specifying a linear code by giving a basis for a subspace of $V(n, q)$, and encoding and decoding can be done using linear algebra.

The limitations of linear codes include the facility to work only over fields of prime power order, and the structure of linear codes has an impact on efficiency. For example, although a 2-(8,20,3) code exists, the best binary linear code of length 8 and minimal distance 3 has only 16 codewords.

(27.2.2) Encoding

Definition 25 *The generator matrix G for a linear code C is the matrix whose rows form a basis for $C \leqslant V(n, q)$.*

A generator matrix has more columns than rows, and it is preferable to put the matrix in 'standard form' where the left hand portion is in Echelon form. However, not all codes may have generator matrices in standard form, and the initial $k \times k$ portion of the generator matrix may have linearly dependent rows. A permutation of the columns will solve this, and although column operations are not permittable in a vector space they serve to produce an equivalent code and so in this situation can be tolerated.

Definition 26 *Two linear codes are equivalent if one can be obtained from the other by a combination of*

1. *Permuting columns in all codewords simultaneously. This corresponds to a permutation of the columns of the generator matrix.*
2. *Multiplying any particular co-ordinate in all codewords by a constant. This corresponds to multiplying a column of the generator matrix by a constant.*

Linear codes are always described in terms of their generator matrix. If C has dimension k then G will be a $k \times n$ matrix of full row rank (k) with rows $\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_k$. Hence C contains some q^k vectors, all of the form

$$\lambda_1 \mathbf{r}_1 + \lambda_2 \mathbf{r}_2 + \cdots + \lambda_k \mathbf{r}_k \qquad \lambda_i \in \mathrm{GF}\,(q)\ \ \forall i$$

This corresponds to a message vector $(\lambda_1, \lambda_2, \ldots, \lambda_k)$ and so the encoded message is

$$\sum_{i=1}^{k} \lambda_i \mathbf{r}_i = \breve{}G$$

Hence encoding is done by right multiplication by the generator matrix.

(27.2.3) Decoding

Decoding is not quite so easy as encoding, indeed it must incorporate some method for detecting and correcting errors. Consider a code

$$G = \begin{pmatrix} 1011 \\ 0101 \end{pmatrix} \quad \text{so} \quad C = \{0000, 1011, 0101, 1110\}$$

over GF (2) then if an error occurs in the first place then the received vector will be in the coset of C, $1000 + C$.

Definition 27 *A coset of a subspace C of a vector space $V(n,q)$ is the set*

$$\mathbf{a} + C = \{\mathbf{a} + \mathbf{c} \mid \mathbf{c} \in C\}$$

Theorem 28 *Let $C \leqslant V(n,q)$ have dimension k. Then*

1. *If $\mathbf{b} \in \mathbf{a} + C$ then $\mathbf{b} + C = \mathbf{a} + C$.*
2. *Every vector in $V(n,q)$ is in a coset of C.*
3. *Every coset has q^k vectors in it.*
4. *There are q^{n-k} cosets.*
5. *Distinct cosets are disjoint.*

Proof.

1. If $\mathbf{b} \in \mathbf{a} + C$ then $\mathbf{b} = \mathbf{a} + \mathbf{x}$ for some $\mathbf{x} \in C$. Say elements of $\mathbf{b} + C$ are of the form $\mathbf{b} + \mathbf{y}$ then $\mathbf{b} + \mathbf{y} = \mathbf{a} + \mathbf{x} + \mathbf{y}$ but $\mathbf{x} + \mathbf{y} \in C$ and hence $\mathbf{b} + \mathbf{y} \in \mathbf{a} + C$. Therefore $\mathbf{b} + C \subseteq \mathbf{a} + C$.

   Conversely, all elements of $\mathbf{a} + C$ are of the form $\mathbf{a} + \mathbf{z}$. Therefore $\mathbf{a} + \mathbf{z} = \mathbf{b} - \mathbf{x} + \mathbf{z}$ but since $\mathbf{x} - \mathbf{z} \in C$ this gives $\mathbf{a} + \mathbf{a} \in \mathbf{b} + C$. So $\mathbf{a} + C \subseteq \mathbf{b} + C$.
2. For any vector $\mathbf{v}$, $\mathbf{v} \in \mathbf{v} + C$ because $\mathbf{0} \in C$ because C is a vector space.
3. $|C| = q^k$ hence by additive cancellation the result holds.
4. Since every coset has q^k vectors in it and $|V(n,q)| = q^n$ it follows that there are q^{n-k} cosets.
5. If $\mathbf{a} \in \mathbf{b} + C$ and $\mathbf{a} \in \mathbf{c} + C$ then by part 1 $\mathbf{b} + C = \mathbf{a} + C = \mathbf{c} + C$. □

Decoding Using The Slepian Standard Array

Fir the code $C = \{0000, 1011, 0101, 1110\}$ the cosets may be tabulated as shown in Table 27.2.3. This is not a particularly good code as an error in the fourth place cannot be corrected.

	coset leader			
code	0000	1011	0101	1110
coset 1	1000	0011	1101	0110
coset 2	0100	1111	0001	1010
coset 3	0010	1001	0111	1100

Table 2: Slepian standard array for a simple code.

The tabulation as shown in Table 27.2.3 is called a Slepian standard array. It is used to decode a received vector by looking that vector up in the table and decoding to the vector at the top of the column. The error made in transmission was the addition of the vector at the start of the row.

When making such an array it is usual to choose as coset leaders the vectors of least weight—corresponding to the fewest errors made.

Clearly, listing all the codewords in this way is undesirable, and there is no clear cut way to choose the coset leaders—whether to choose 0100 or 0001 in coset 2 of Table 27.2.3. In such a case the code may be used only to detect errors in the second and fourth places rather than correct them. This is an example of incomplete decoding; only perfect codes can decode completely. For complete decoding of single errors, all single error vectors must be in different cosets. There are $n(q-1)$ single error vectors, so together with the original code (represented by the zero coset) there must be $1+n(q-1)$ cosets. For complete decoding there are precisely $1+n(q-1)$ cosets, and since there are q^k vectors in each coset

$$q^k = \frac{q^n}{1+n(q-1)}$$

Extending, if C detects t errors then all the codewords $\mathbf{x}$ with $w(\mathbf{x}) \leqslant t$ must each be in a different coset. There are

$$\sum_{i=0}^{t} \binom{n}{i} (q-1)^i$$

such vectors and since each coset contains q^k vectors this gives

$$q^n = q^k \sum_{i=0}^{t} \binom{n}{i} (q-1)^i$$

which is obtained if and only if C is perfect. The coset leaders are then precisely the codewords of weight at most t.

Probability Of Correct Decoding

The use of a code is no guarantee that the message sent is the same as the message received, though it does increase the chances of this. It is of interest, therefore, as to the probability that a codeword will be correctly decoded.

For the sake of simplicity consider only binary codes and let the symbol error probability be p. It is clear that the probability that a vector has weight i is $p^i(1-p)^{n-i}$.

Theorem 29 *Let α_i be the number of coset leaders of weight i then the probability that a received codeword is correctly decoded is*

$$p_c(C) = \sum_{i=1}^{n} \alpha_i p^i (1-p)^{n-i}$$

Only the errors that are the coset leaders can be corrected, so hence the result. A more useful measure of the accuracy of a code is the probability that a received codeword is decoded incorrectly, $p_e(C) = 1 - p_c(C)$.

Definition 30 *For a linear $[n,k]$ code (i.e. of length n and dimension k) the rate of the code is $R(C) = \frac{k}{n}$.*

The rate of a code represents the proportion of the information sent that is actual message—the other part provides information about how to correct errors.

Definition 31 *The capacity of a binary symmetric channel with symbol error probability p is*

$$\mathcal{C}(p) = 1 + p\log_2 p + (1-p)\log_2 (1-p)$$

Theorem 32 (Shannon's Theorem) *Suppose a binary symmetric channel has symbol error probability p. Then for any $R < \mathcal{C}(p)$ and any $\varepsilon > 0$ there exists—for sufficiently large n—a code of rate at least R and with error probability $p_c(C) < \varepsilon$.*

(27.2.4) Syndrome Decoding

Dual Codes

Consider a vector space with the standard inner product, i.e. $\mathbf{u} \cdot \mathbf{v} = \sum_{i=1}^{n} u_i v_i$.

Definition 33 *If $C \leqslant V(n,q)$ then define*

$$C^{\perp} = \{\mathbf{v} \in V(n,q) \mid \mathbf{u} \cdot \mathbf{v} = 0 \, \forall \mathbf{u} \in C\}$$

This definition requires some thought: it means all the vectors in $V(n,q)$ that are orthogonal to all the vectors in C. If $V(n,q)$ was over an infinite field then C and $C^{\perp}$ would be disjoint except for $\mathbf{0}$. However, modular arithmetic means that for finite fields this does not hold. Take for example

$$C = \{0000, 1100, 0011, 1111\} \quad \text{then} \quad C^{\perp} = \{\text{vectors of the form } v_1 v_1 v_3 v_3\} = C$$

Lemma 34 *If $\mathbf{v}$ is orthogonal to every vector in the basis of a code C then $\mathbf{v}$ is orthogonal to every vector of C.*

Proof. Let $\mathbf{u} \in C$ and let $\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_k$ be a basis for C. Therefore

$$\begin{aligned} \mathbf{u} &= \sum_{i=1}^{k} \lambda_i \mathbf{r}_i \\ \text{so } \mathbf{u} \cdot \mathbf{v} &= v \cdot \sum_{i=1}^{k} \lambda_i \mathbf{r}_i \\ &= \sum_{i=1}^{k} \lambda_i (\mathbf{v} \cdot \mathbf{r}_i) \\ &= 0 \end{aligned}$$

because by hypothesis $\mathbf{v} \cdot \mathbf{r}_i = 0$ for all i. □

This simpler condition can be further simplified as $\mathbf{v} \in C^{\perp} \Leftrightarrow \mathbf{v}G^T = \mathbf{0}$.

Lemma 35 *$C^{\perp}$ is a linear code.*

Proof. To show this it is necessary to show that $C^{\perp} \leqslant V(n,q)$ which is done in the usual way.
If $\mathbf{v}, \mathbf{w} \in C^{\perp}$ and $\mathbf{u} \in C$ then

$$(\lambda \mathbf{v} + \mathbf{w}) \cdot \mathbf{u} = \lambda(\mathbf{v} \cdot \mathbf{u}) + \mathbf{w} \cdot \mathbf{u} = 0$$

Therefore $\lambda \mathbf{v} + \mathbf{w} \in C^{\perp}$ showing that $C^{\perp}$ is closed under addition and scalar multiplication. $C^{\perp}$ is non-empty as $\mathbf{0} \in C^{\perp}$, and hence $C^{\perp}$ is a vector space. □

Theorem 36 *If $C \leqslant V(n,q)$ has dimension k then $C^{\perp}$ has dimension $n-k$.*

Proof. $C^{\perp}$ is the solution set to k independent homogeneous linear equations in n unknowns, namely $\mathbf{v}G^T = \mathbf{0}$. □

Corollary 37 $(C^{\perp})^{\perp} = C$.

Proof. Every vector in $C^{\perp}$ is orthogonal to every vector in C and so it must be the case that $C \leqslant (C^{\perp})^{\perp}$. Now, $\dim(C^{\perp})^{\perp} = n - (n-k) = k = \dim C$ so $C = (C^{\perp})^{\perp}$. □

Definition 38 *The generator matrix H for $C^{\perp}$ is called the parity check matrix for C.*

Since H is a generator matrix its rows must be elements of $C^{\perp}$ and so satisfy $\mathbf{r}_i G^T = \mathbf{0}$. Hence $HG^T = (0)$.

H is not called the parity check matrix without reason. Every row of H may be thought of as a parity check for the code C since for $\mathbf{v} \in C$, $\mathbf{v}H^T = \mathbf{0}$. So for example if

$$H = \begin{pmatrix} 1100 \\ 0011 \end{pmatrix}$$

then any vector $\mathbf{v} = (v_1 v_2 v_3 v_4) \in C$ must satisfy $v_1 + v_2 = 0$ and $v_3 + v_4 = 0$.

Theorem 39 *If a code C has generator matrix in the form $G = (I_k \mid A)$ then there exists a parity check matrix for C of the form $H = (-A^T \mid I_{n-k})$.*

Proof. Let G and H have the prescribed form, and suppose

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1,n-k} \\ a_{21} & a_{22} & \cdots & a_{2,n-k} \\ \vdots & \vdots & \ddots & \vdots \\ a_{k1} & a_{k2} & \cdots & a_{k,n-k} \end{pmatrix}$$

then H has the required size and its rows are linearly independent (by virtue of the presence of the identity matrix). Now, the inner product of the ith row of G with the jth row of H is $-a_{ij} + a_{ij} = 0$ meaning that the rows of H are in $C^{\perp}$. But there are $n-k$ such rows which are linearly independent and so must span the whole of $C^{\perp}$. Hence H is a parity check matrix for C. □

Syndromes

Definition 40 *Let $\mathbf{a}$ be any vector in a vector space $V(n,q)$. Let $C \leqslant V(n,q)$ be a code with parity check matrix H. Then the syndrome of $\mathbf{a}$ is defined as $S(\mathbf{a}) = \mathbf{a}H^T$.*

Syndromes are useful because they produce a one-to-one relationship with the cosets of C, hence eliminating the need to calculate the Slepian standard array.

Suppose $\mathbf{b} \in \mathbf{a} + C$ then $\mathbf{b} = \mathbf{a} + \mathbf{x}$ for some $\mathbf{x} \in C$. From Section 27.2.4 $\mathbf{y} \in C \Leftrightarrow \mathbf{y}H^T = 0$. Hence

$$\mathbf{b}H^T = (\mathbf{a} + \mathbf{x})H^T = \mathbf{a}H^T\mathbf{x}H^T = \mathbf{a}H^T$$

Conversely, if $\mathbf{b}H^T = \mathbf{a}H^T$ then $(\mathbf{b} - \mathbf{a})H^T = \mathbf{0}$ meaning that $\mathbf{b} - \mathbf{a} \in C$ and therefore $\mathbf{b} \in \mathbf{a} + C$. Hence the one-to-one correspondence between syndromes and cosets.

To use syndromes for decoding it is necessary to tabulate each coset leader and its syndrome. Decoding a vector $\mathbf{y}$ is then done by calculating $S(\mathbf{y})$ and using the table to find the coset leader $\mathbf{a}$ for which $S(\mathbf{y}) = S(\mathbf{a})$. The vector is then decoded as $\mathbf{y} - \mathbf{a}$.

If $d(C) = 2t + 1$ (or $2t + 2$) then up to t errors can be corrected, and the syndromes for these t errors are the only syndromes that need be calculated. If a received vector has any other syndrome then an error is detected, but cannot be corrected; in fact this must be an error in more than t places.

Returning to the example of the ISBN code, recall that for an ISBN $0a_0 + 1a_1 + \cdots + Xa_X = 0$ modulo 11, so clearly the parity check matrix is

$$H = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & X \end{pmatrix}$$

If a single error occurs with erroneous symbol in position k then the syndrome has value ka'_k. However, this provides only 1 equation in 2 unknowns, so k and a'_k cannot be found—the error cannot be corrected.

Example 41 *Modify the ISBN code to have parity check matrix*

$$H = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & X \end{pmatrix}$$

Show that 1 error can be corrected and that transposition errors can be detected.

Proof. Solution Suppose the vector $a_0a_1 \ldots a_X$ is sent and that a single error occurs so that in the ith position $a_i + k$ is received. The syndrome is then (k, ik) so that in general when the syndrome of a received vector is (p, q) the error is corrected by subtracting p from the $\frac{q}{p}$th co-ordinate.

However, if $(p, q) = (0, 0)$ then no error must have occurred. If only $A = 0$ then as for the ISBN code, a transposition error must have occurred. □

(27.3) Hamming Codes

(27.3.1) Binary Hamming Codes

De□nition & Basic Results

Clearly the properties of linear codes are highly desirable, regardless of the effect of the admission of so much structure on the efficiency of code achievable. Hamming codes are linear codes constructed in a certain way so as to have more such desirable properties. As will be shown, Hamming codes are perfect.

Definition 42 *Let C be a code of length n and dimension k. The redundancy, r, of C is defined as $r = n - k$.*

It is immediately clear that r is both the number of rows in the parity check matrix, and the dimension of the dual code.

Definition 43 *The binary Hamming code of redundancy r is defined as the code with parity check matrix whose columns are every non-zero vector of length r.*

A binary Hamming code has length $2^r - 1$ because there are $2^r - 1$ non-zero vectors of length r over GF (2). When writing down such a parity check matrix it is imperative to do so in a orderly fashion, to which end it is usual to write down as the columns the binary numbers from 1 to $2^n - 1$.

Lemma 44 *The binary Hamming code is a perfect code with minimal distance 3.*

Proof. For the minimal distance, that $d(C) \geqslant 3$ can be deduced by showing there are no codewords of weight 1 or 2.

If a codeword $\mathbf{y} \in C$ has weight 1 then because $\mathbf{a}H^T = \mathbf{0} \Leftrightarrow \mathbf{y} \in C$ it must be the case that H^T has a row of zeros, i.e. the zero vector is a column of H. This would contradict the definition of H and so there are not vectors of weight 1.

If a codeword $\mathbf{y} \in C$ has weight 2 then using $\mathbf{y}H^T = \mathbf{0} \Leftrightarrow \mathbf{y} \in C$ it must be the case that one row of H^T is the negative of another. But modulo 2 this means that 2 columns of H are the same, contradicting the definition of H, hence there are no vectors of weight 2 in C.

Now, if the columns of H are ordered as suggested then the first 3 are

$$\begin{matrix} 0 & 0 & 0 \\ \vdots & \vdots & \vdots \\ 0 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{matrix}$$

in which case $1110\ldots0 \in C$. Since there exists a codeword of weight 3 the minimal distance must, therefore, be 3.

The dimension of the code is $k = n - r$ so over $\mathrm{GF}\,(2)$ there must be 2^{n-r} vectors in the binary Hamming code of redundancy r. Calculating the sphere packing bound gives

$$M \leqslant \frac{2^n}{1+n} = \frac{2^n}{1+(2^r-1)} = 2^{n-r}$$

so the sphere packing bound is attained. □

Decoding With A Binary Hamming Code

Observe that since the columns of the parity check matrix H are simply the binary numbers from 1 to $2^n - 1$ then where

$$\mathbf{e}_i = \begin{pmatrix} 0 & 0 & \ldots & 0 & 1 & 0 & \ldots & 0 \end{pmatrix}$$

with the 1 in the ith position then $\mathbf{e}_i H^T$ is the binary representation of i.

The binary Hamming code can correct 1 error, so the coset leaders are all the vectors $\mathbf{e}_i$. Therefore if $\mathbf{y}$ is a received vector, $\mathbf{y}H^T$ must give the binary representation of the position in $\mathbf{y}$ where the error has occurred.

The Extended Binary Hamming Code

Any binary Hamming code may be extended by means of an overall parity check. By Theorem 13 the extended code has minimal distance 4, and trivially it has length 2^n.

Lemma 45 *Adding an overall parity check to a linear $[n,k]$ code C produces a linear $[n+1,k]$ code $\hat{C}$.*

Proof. Let $x_1x_2\ldots x_nx_{n+1} \in \hat{C}$ and $y_1y_2\ldots y_ny_{n+1} \in \hat{C}$ then $x_1x_2\ldots x_n \in C$ and $y_1y_2\ldots y_n \in C$ and moreover

$$x_{n+1} = \sum_{i=1}^{n} x_i \quad \text{and} \quad y_{n+1} = \sum_{i=1}^{n} y_i \quad \text{modulo 2}$$

Therefore

$$\sum_{i=1}^{n} (x_i + y_i) = x_{n+1} + y_{n+1}$$

meaning that $x_1 + y_1, x_2 + y_2, \ldots, x_{n+1}y_{n+1} \in \hat{C}$, so $\hat{C}$ is a vector space and so is a code. Clearly it has length $n+1$ and dimension k, so the result is shown. □

Where the binary Hamming code is a $[2^r, 2^r - 1 - r, 3]$ code the extended binary Hamming code is a $[2^r - 1, 2^r - 1 - r, 4]$ code. The effect of adding this parity check is the requirement that

$$x_1 + x_2 + \cdots + x_{n+1} = 0$$

This defines a new row in the parity check matrix, so that

$$\hat{H} = \begin{pmatrix} & & & & 0 \\ & & & & 0 \\ & & H & & \vdots \\ & & & & 0 \\ 1 & 1 & \dots & 1 & 1 \end{pmatrix}$$

Note that $\hat{H}$ is not in standard form. Decoding the extended binary Hamming code may be done as follows. Calculate the syndrome $S(\mathbf{y}) = (s_1 s_2 \dots s_{n+1})$.

- If $s_{n+1} = 1$—the value of the parity check $y_1 + y_2 + \cdots + y_{n+1}$—then there must be an odd number of errors. As before $s_1 s_2 \dots s_n$ is the binary representation of the position of the error, though if it represents the number 0 then the error must have occurred in the last place.
- If s_{n+1}=0 then there are evenly many errors.
  - If there no errors then $S(\mathbf{y}) = \mathbf{0}$.
  - If the syndrome is not zero then two errors are detected.
  - If there are 4 errors then another codeword has been reached and it is assumed that no error has occurred.

(27.3.2) General Hamming Codes

Construction & Properties

Lemma 46 *Let C be a code with parity check matrix H. C contains a word of length d if and only if a set of d columns of H are linearly dependent.*

Proof. Let $\mathbf{H}_1, \mathbf{H}_2, \dots, \mathbf{H}_n$ be the columns of H. Therefore

$$\begin{aligned} x_1 x_2 \dots x_n \in C \Leftrightarrow \begin{pmatrix} x_1 & x_2 & \dots & x_n \end{pmatrix} \begin{pmatrix} \mathbf{H}_1^T \\ \mathbf{H}_2^T \\ \vdots \\ \mathbf{H}_n^T \end{pmatrix} &= \mathbf{0} \\ \Leftrightarrow x_1 \mathbf{H}_1^T + x_2 \mathbf{H}_2^T + \cdots + x_n \mathbf{H}_n^T &= \mathbf{0} \\ \Leftrightarrow x_1 \mathbf{H}_2 + x_2 \mathbf{H}_2 + \cdots + x_n \mathbf{H}_n &= \mathbf{0} \end{aligned}$$

Now, if $\mathbf{x}$ has weight d this defines a linear dependence on d columns of H.

Conversely if $\lambda_i \mathbf{H}_i + \lambda_j \mathbf{H}_j + \cdots + \lambda_k \mathbf{H}_k = \mathbf{0}$ where there are d terms in this sum and where all other values of λ are 0 then

$$\sum_{i=1}^{n} \lambda_i \mathbf{H}_i = \mathbf{0}$$

and therefore ˇ $\in C$ is a vector of weight d. □

Theorem 47 *Let C be a code with parity check matrix H. Then $d(C)$ is equal to the minimum number of columns of H that are linearly dependent.*

Proof. Immediate from Lemma 46. □

This theorem may be restated in one of the following ways. $d(C) = d \Leftrightarrow$ every set of $d-1$ columns of H is linearly independent $\Leftrightarrow$ there exists a set of d columns of H that are linearly dependent.

The value of this theorem is it allows a parity check matrix to be constructed in such a way to define a code with a given minimal distance. Over GF $(()\,q)$ every column has $q-1$ non-zero multiples, so only one of these $q-1$ can be chosen for otherwise the minimal distance is 1—not much use. It also provides a method by which the minimal distance of a code may be found.

Definition 48 *The Hamming code over* GF (q) *with redundancy r,* Ham (r, q)*, is defined to have parity check matrix whose columns are all the vectors in $V(r, q)$ such that no two of them are linearly dependent.*

Of course there are many ways in which such vectors may be chose, but it is clear that any particular choice can be obtained from any other by a combination of multiplication of columns by scalars and column permutations. Ham (r, q) therefore specifies a family of equivalent codes, and it is unimportant which one is chosen.

When choosing the columns of a parity check matrix for a Hamming code it is usual to choose vectors beginning with zeros, then a 1, then other digits chosen so as not to cause linear dependence. For example, for a Ham $(3, 3)$ code the parity check matrix would be chosen as

$$H = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 2 & 2 & 2 \\ 1 & 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \end{pmatrix} \tag{49}$$

Now, there are $q^r - 1$ vectors in $V(r, q)$ and each has $q-1$ scalar multiples. Hence the parity check matrix for the q-ary Hamming code must have $\frac{q^r-1}{q-1}$ columns. Observe this is so in equation (49) where there are $3^0 + 3^1 + 3^2$ columns[‡]. From Theorem 47 this must give a

$$\left[\frac{q^r-1}{q-1}, \frac{q^r-1}{q-1} - r, 3\right]$$

code, called Ham (r, q). The minimal distance is 3 because by construction (from Theorem 47) $d(C) > 2$ but the linear dependence

$$\begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 1 \end{pmatrix}$$

will always exist and so the minimal distance must actually be 3.

Theorem 50 Ham (r, q) *is perfect.*

[‡] Recall the factorisation $x^n - 1 = (x-1)(x^{n-1} + x^{n-2} + \cdots + 1)$.

Proof. Ham (r, q) has dimension $n - r$ where $n = \frac{q^r-1}{q-1}$ and so there must be $M = q^{n-r}$ codewords. Now,

$$\begin{aligned} \frac{q^n}{\sum_{i=0}^{t} \binom{n}{i}(q-i)^i} &= \frac{q^n}{1+n(q-1)} \\ &= \frac{q^n}{1+\frac{q^r-1}{q-1}(q-1)} \\ \frac{q^n}{q^r} &= M \end{aligned}$$

Hence the sphere packing bound is attained, so Ham (r, q) is perfect. □

Decoding

Since Ham (r, q) is a perfect 1 error correcting code the coset leaders must be the vectors $\lambda \mathbf{e}_i$ which have all places zero bar for a λ in the ith place. Now, $\lambda \mathbf{e}_i H^T = \lambda \mathbf{H}_i$ and therefore for a received vector $\mathbf{y}$ with $S(\mathbf{y} = \mu \mathbf{H}_j$ the decoding is performed by subtracting μ from the jth column of $\mathbf{y}$. Clearly determining μ and j is not easy, and in practise binary Hamming codes are much preferred.

(27.3.3) Shortening Codes

A code may be shortened by removing a column from the parity check matrix. This has the effect of reducing the length of the code but maintaining the redundancy: consequently the dimension of the code has been decreased so there cannot be as many codewords.

Theorem 51 *When a code is shortened the minimal distance will either stay the same or increase.*

Proof. Say H has columns $\mathbf{H}_1, \mathbf{H}_2, \ldots, \mathbf{H}_n$ and that $\mathbf{x} = x_1 x_2 \ldots x_n \in C$.

$$\begin{aligned} \mathbf{x} \in C &\Leftrightarrow x_1\mathbf{H}_1 + x_2\mathbf{H}_2 + \cdots + x_n\mathbf{H}_n = \mathbf{0} \\ \text{giving } x_2x_3\ldots x_n \in C' &\Leftrightarrow \mathbf{H}_2 + x_2\mathbf{H}_3 + \cdots + x_n\mathbf{H}_n = \mathbf{0} \\ &\Leftrightarrow 0x_2x_3\ldots x_n \in C \end{aligned}$$

Hence all vectors in C' have in C a zero in the deleted position, and hence the minimal weight cannot decrease. Furthermore, if the vector(s) of minimal weight in C do not have zeros in the deleted position then the vector of minimal weight in C' must have greater minimal weight than those of C; hence the minimal distance may increase. □

The increasing of minimal distance is consistent with Theorem 47 because when a column is deleted the linear dependencies involving that column cease to be. This certainly ensures that the minimal distance cannot decrease. Shortening a $[n, k, d]$ code therefore produces a $[n-1, k-1, d']$ code with $d' \geqslant d$.

When shortening a code it is important not to create linear dependence in the rows of H as the redundancy is then decreased.

For example Ham $(2, 11)$ has parity check matrix

$$H = \begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & X \end{pmatrix}$$

Deleting the first two columns does not produce linear dependence in the rows, so this is a valid way to shorten the code. The minimal distance must be at least 3 for that is the minimal distance of a Hamming code. The minimal distance must in fact be 3 as the columns of H when treat as vectors occupy a 2 dimensional space, so any 3 must be linearly dependent, then apply Theorem 47.

(27.3.4) The Gilbert Bound

Using Theorem 47 a Hamming code may be constructed by choosing the columns for a generator matrix such that no 2 are linearly dependent. This is quite easy, but extending to 3 columns (giving a minimal distance of at least 4) the problem of choosing the next vector is not so simple. Clearly there is a limit to the number of vectors available.

Suppose it is required to choose columns so that no 3 are linearly dependent and that $m-1$ columns have already been chosen. The vectors that cannot be chosen for the mth column are

- All scalar multiples of the $m-1$ columns already chosen. There are at most $(q-1)(m-1)$ of these. Note that there are only $q-1$ multiples allowed because the zero vector cannot be a column of the parity check matrix.
- All linear combinations of 2 of the $m-1$ columns already chosen. There are at most $\binom{m-1}{2}(q-1)^2$ of these.

It is important to use the words 'at most' for in the first case some scalar multiples may be equal and in the second case there may be more than one way to express a given vector as a linear combination of 2 of the $m-1$ vectors chosen. Including the zero vector this shows that at most

$$(q-1)^2\binom{m-1}{2} + (q-1)(m-1) + 1$$

vectors must be avoided. Whenever this number is less than q^r it must, therefore, be possible to choose another vector. Note that the converse is false: even when equal to q^r it may be possible to choose another vector. Suppose that

$$(q-1)^2\binom{n-1}{2} + (q-1)(n-1) + 1 < q^r$$
$$\text{then } (q-1)^2\binom{m-1}{2} + (q-1)(m-1) + 1 < q^r \ \forall m \leqslant n$$

and so it is possible to choose another column for the parity check matrix.

Theorem 52 (Gilbert Bound) *If* $\sum_{i=0}^{d-2}(q-1)^i\binom{n-1}{i} < q^r$ *then there exists a* $q-[n,k,d]$ *code with* $k=n-r$.

Proof. If

$$\sum_{i=1}^{d-2}(q-1)^i\binom{n-1}{i} < q^r$$
$$\text{then } \sum_{i=1}^{d-2}(q-1)^i\binom{m-1}{i} < q^r \ \forall m \leqslant n$$

The left hand side of this expression counts the number of linear combinations of up to $d-2$ of the already chosen columns for the parity check matrix. Any other column chosen must be different from any of these linear combinations so there never exists a dependence between any $d-1$ columns. Hence by Theorem 47 $d(C) \geqslant d$, as required. □

The Gilbert bound may also be stated that if q is a prime power then $A_q(n,d)$ is at least the largest power of q less than

$$\frac{q^n}{\sum_{i=1}^{d-2}(q-1)^i\binom{n-1}{i}}$$

This is similar to the sphere packing bound except that the sphere packing bound gives an upper bound for $A_q(n,d)$ and is valid for arbitrary codes, not just linear codes.

(27.3.5) Golay Codes

Existence Of Perfect Codes

The sphere packing bound is a very strict condition on the integers q, n, t, M: If q is a prime power then the sum must be a power of q. The only known solutions are as follows.

- The trivial solutions, when $t = 0$ and $M = q^n$ or when $t = n$ and $M = 1$ corresponds to the code with all codewords and the repetition code.
- The Hamming code type solutions where $t = 1$ and $n = \frac{q^r-1}{q-1}$ for any r and $M = q^{n-r}$. This works for all q, but codes are only known when q is a prime power: Hamming codes in the binary case and generalised by Golay.
- Other sporadic solutions include
  - $t = 3, q = 2, n = 23, M = 2^{12}$ for which a code can be constructed.
  - $t = 3, q = 3, n = 11, M = 3^6$ for which a code can be constructed.
  - $t = 2, q = 2, n = 90, M = 2^{78}$ for which no code can be constructed.

Golay was interested only in linear codes, and using Theorem 47 the non-existence of a linear $2-[90,78,5]$ code can be shown.

Theorem 53 *There does not exist a linear* $2-[90,78,5]$ *code.*

Proof. Let $\mathbf{H}_1, \mathbf{H}_2, \ldots, \mathbf{H}_{90}$ be the columns of the parity check matrix, then by Theorem 47 any 4 of them must be linearly independent. Consider the set

$$X = \left\{\mathbf{0}, \mathbf{H}_i, \mathbf{H}_j + \mathbf{H}_k \mid 1 \leqslant i \leqslant 90, 1 \leqslant i < j \leqslant 90\right\}$$

then by the requirement of linear independence this must be a set of $1 + 90 + \binom{90}{2} = 4,096 = 2^{12}$ distinct vectors. Now, $90 - 78 = 12$ so the redundancy is 12, which is the number of rows of H. But there are precisely 2^{12} binary vectors of length 12, so these vectors must be all those vectors in $V(12,2)$ and exactly half of these (2^{11} of them) have odd weight.

The same number is now calculated by examining X. Suppose that of the columns of H, m have odd weight, so $90 - m$ have even weight. Of the $\binom{90}{2}$ vectors in X of the form $\mathbf{H}_j + \mathbf{H}_k$ there must be $m(90-m)$ of them with odd weight as for a sum of vectors to have odd weight one must have odd weight while the other must have even weight. For the sum to have even weight the two vectors must both have odd or both have even weight, so there are $\binom{m}{2} + \binom{90-m}{2}$ vectors of even weight. The total number of vectors in X that have odd weight is therefore

$$m + m(90-m) = m(91-m) = 2^{11}$$

but the left hand side of this expression is odd (unless $m = 1$) hence a contradiction, so no such code can exist. □

Theorem 54 *There does not exist a* $2-\left[90, 2^{78}, 5\right]$ *code.*

Proof. Let C be such a code and let K be the number of codewords of weight 5. Without loss of generality $\mathbf{0} \in C$ so there are no codewords of weight 3 in C, but there are $\binom{90}{3}$ such vectors of length 90. Each vector of weight 3 must be at distance at most 2 from exactly one codeword as C is perfect. These codewords must therefore be of weight 5.

Consider now the vectors of weight 3 that begin with two 1s. There must be $\binom{90-2}{1} = 88$ of these, and every vector of weight 5 that starts with two 1s must cover exactly 3 of these (there are 3 other 1s somewhere in the vector of weight 5). But 3 does not divide 88, hence a contradiction. □

Constructing The $2-[23, 12, 7]$ Golay Code

Instead of constructing the code directly it is easier to construct the extended $2-[24, 12, 8]$ code that has an overall parity check.

Definition 55 *The extended binary Golay code is (any code equivalent to)* $C = G_{24}$ *that has generator matrix* $G = (I_{12} \mid A)$ *where*

$$A = \begin{pmatrix}
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\
1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\
1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\
1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\
1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\
1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\
1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\
1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\
1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\
1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0
\end{pmatrix}$$

Row 1 is unique while the other rows share the first column and the remaining 11 elements are obtained by cycling the row above one place to the left.

Clearly G_{24} has length 24 and dimension 12. The task now is to show that the minimal distance is 8. This may be done using a computer to check every combination, but a more elegant theoretical method is now presented.

Lemma 56 $G_{24} = G_{24}^{\perp}$.

Proof. By Lemma 34 it is sufficient to show that any two rows of G are orthogonal, and that each row is orthogonal to itself, hence showing that $G_{24} \leqslant G_{24}^{\perp}$. Since $\dim G_{24} = 12 = 24 - 12 = \dim G_{24}^{\perp}$ the result is shown. Now,

$$\begin{aligned}
\mathbf{r}_1 \cdot \mathbf{r}_1 &= 12 \equiv 0 \mod 2\\
\mathbf{r}_i \cdot \mathbf{r}_i &= 8 \equiv 0 \mod 2 \text{ for } i > 1\\
\mathbf{r}_1 \cdot \mathbf{r}_i &= 6 \equiv 0 \mod 2 \text{ for } i > 1\\
\mathbf{r}_2 \cdot \mathbf{r}_3 &= 4 \equiv 0 \mod 2\\
\mathbf{r}_2 \cdot \mathbf{r}_4 &= 4 \equiv 0 \mod 2\\
\mathbf{r}_2 \cdot \mathbf{r}_5 &= 4 \equiv 0 \mod 2\\
\mathbf{r}_2 \cdot \mathbf{r}_6 &= 4 \equiv 0 \mod 2\\
\mathbf{r}_2 \cdot \mathbf{r}_7 &= 4 \equiv 0 \mod 2\\
\mathbf{r}_2 \cdot \mathbf{r}_8 &= 4 \equiv 0 \mod 2\\
\mathbf{r}_2 \cdot \mathbf{r}_9 &= 4 \equiv 0 \mod 2\\
\mathbf{r}_2 \cdot \mathbf{r}_{10} &= 4 \equiv 0 \mod 2\\
\mathbf{r}_2 \cdot \mathbf{r}_{11} &= 4 \equiv 0 \mod 2\\
\mathbf{r}_2 \cdot \mathbf{r}_{12} &= 4 \equiv 0 \mod 2
\end{aligned}$$

By symmetry of the rows the result is now shown. □

Lemma 57 *A second generator matrix for G_{24} is $(-A^T \mid I_{12})$.*

Proof. Since G_{24} has generator matrix $(I_{12} \mid A)$ the dual code $G_{24}^{\perp}$ has generator matrix $(-A^T \mid I_{12})$. But by Lemma 56 $G_{24} = G_{24}^{\perp}$ so this is a generator matrix for G_{24}. □

Lemma 58 *Every codeword in G_{24} has weight divisible by 4.*

Proof. Any vector in G_{24} is a linear combination of the rows of the generator matrix. All rows of the generator matrix have weight 8 or 12, so the result holds for linear combinations of 1 basis vector.

Suppose the result holds for a linear combination of t rows of the generator matrix. In this binary case,

$$w(\mathbf{r}_1 + \mathbf{r}_2 + \cdots + \mathbf{r}_t + \mathbf{r}_{t+1}) = w(\mathbf{r}_1 + \mathbf{r}_2 + \cdots + \mathbf{r}_t) + w(\mathbf{r}_{t+1}) \; - 2w((\mathbf{r}_1 + \mathbf{r}_2 + \cdots + \mathbf{r}_t) \cap (\mathbf{r}_{t+1}))$$

Now, the first two terms are divisible by 4 by hypothesis. Since the code is self dual the dot product of any two vectors must be even (0 mod 2) and therefore the intersection of any two vectors must have even weight. The factor of 2 in the last term ensures it is also divisible by 4. □

Lemma 59 *G_{24} has no codewords of weight 4.*

Proof. Suppose that a codeword $\mathbf{x}$ has weight 4. Then in the first 12 co-ordinates there must be 4,3,2,1, or 0 non-zero entries and correspondingly 0,1,2,3, or 4 in the last 12 co-ordinates.

- If there are 0 non-zero co-ordinates in the first 12 then the only way this can be achieved is if $\mathbf{x} = \mathbf{0}$.
- If there are 4 non-zero co-ordinates in the first 12 then there are no non-zero co-ordinates in the last 12. Therefore using the other generator matrix the same argument as above shows that $\mathbf{x} = \mathbf{0}$.
- If there is 1 non-zero entry in the first 12 co-ordinates then $\mathbf{x}$ must be the corresponding row of the generator matrix and therefore have weight other than 4.
- If there are 3 non-zero entries in the first 12 co-ordinates then there is 1 in the last 12. Therefore apply the above argument using the other generator matrix.

- If there are 2 non-zero entries in the first 12 co-ordinates of $\mathbf{x}$ and 2 in the last 12 then $\mathbf{x}$ must be the sum of 2 rows.

$$w(\mathbf{r}_1+\mathbf{r}_i) = w(\mathbf{r}_1)+w(\mathbf{r}_i)-2w(\mathbf{r}_1\cap\mathbf{r}_i) = 12+8-2.6 = 8$$
$$w(\mathbf{2}+\mathbf{i}) = 8+8-2.4 = 8 \text{ for } j>2$$

So none of these vectors have weight 4. □

Theorem 60 *The code G_{24} is a $2-[24,12,8]$ code.*

Proof. By construction the code has the required length and dimension. Lemma 58 and Lemma 59 shows the minimal distance to be 8. □

Definition 61 *The binary Golay code G_{23} is (any code equivalent to) the code obtained by deleting the last co-ordinate of all codewords of G_{24}.*

Equivalently the last column of the generator matrix of G_{24} may be deleted. Note this is distinct from shortening a code where a column is deleted from the parity check matrix. Removing a column from the generator matrix is called puncturing the code and generally decreases the minimal distance.

Theorem 62 *G_{23} is a perfect code.*

Proof. From the definition of G_{23} from G_{24} the code G_{23} must have parameters $n=23$, $M=2^{12}$, $d=7$ (so $t=3$). These meet the sphere packing bound since

$$2^{11}(1+23+253) = 2^{12}2^{11} = 2^{23} = q^n$$ □

(27.3.6) Automorphism Groups

The code G_{24} has the property that certain permutations of the co-ordinates leave the code unchanged. If the co-ordinates are numbered 1 to 24 then an example permutation is

$$\begin{pmatrix}1 & 13\end{pmatrix}\begin{pmatrix}2 & 14\end{pmatrix}\dots\begin{pmatrix}12 & 24\end{pmatrix}$$

Such permutations of a code are called automorphisms, and not surprisingly these form a group.

Definition 63 *The group of automorphisms of a code C is denoted* Aut(C).

The automorphism groups of G_{23} and G_{24} are the Mathieu groups M_{23} and M_{24} which have around 10 million and 250 million elements respectively. These are simple groups, having no non-trivial normal subgroups.

(27.3.7) t-designs

Suppose that C is a perfect t error correcting code that is not necessarily linear. Without loss of generality $\mathbf{0}\in C$ and $d(C)=2t+1$ so $w(C)=2t+1$. Consider codewords $vtrc_i$ of weight $2t+1$ and vectors $\mathbf{v}_i$ of weight $t+1$. Every vector of weight $t+1$ must be on the perimeter of one of the spheres (of radius t) around the codewords of weight $2t+1$. Hence the $t+1$ non-zero co-ordinates of the $\mathbf{v}_i$ are a subset of those of the codeword upon the sphere of which it lies. Thus each $\mathbf{c}_i$ has $\binom{2t+1}{t+1}$ such $\mathbf{v}_i$ on the surface of its sphere.

If the length of the code is n then there are $\binom{n}{t+1}$ $\mathbf{v}_i$s and therefore the number of codewords of weight $2t+1$ must be

$$\binom{n}{t+1} \div \binom{2t+1}{t+1}$$

That this is an integer is a very strong condition. However, more such conditions are readily available, as restricting attention to those vectors that begin with a 1 there are $\binom{n-1}{t}$ such $\mathbf{v}_i$ and there are $\binom{2t}{t}$ of these in each sphere centred on a codeword that starts with a 1. Hence the number of codewords starting with a 1 is

$$\binom{n-1}{t} \div \binom{2t}{t}$$

Indeed, the number of codewords that begin with i 1s is

$$\binom{n-i}{t+1-i} \div \binom{2t+1-i}{t+1-i}$$

For example for the binary Golay code G_{23} putting $i = 4$ gives, trivially, $\binom{19}{0} \div \binom{3}{0} = \frac{1}{1} = 1$ meaning that if any 4 of the 23 co-ordinates are chosen then there is exactly 1 codeword of weight 7 with 1s in these places.

Definition 64 *The support of a codeword* $\mathbf{c}_j$ *define* B_j*, the support of* $\mathbf{c}_j$*, to be the set* $\{i \in \mathbb{N} \mid (\mathbf{c}_j)_i = 1\}$.

The sets (or blocks) B_j have the property that every set of $t+1$ co-ordinates belong to exactly 1 block. Such a system of blocks is called a Steiner system, $S(t+1, 2t+1, n)$.

Generally a set blocks of size k of points from a set of v points such that every set of t points is in a unique block is called a Steiner system, $S(t, k, v)$.

More generally still if each set of t points in in λ blocks then a $t - (v, k, \lambda)$ design arrises. 2-designs have already been seen in Section 27.1.3, called (v, b, λ) or $b, v, r, k, \lambda)$ design.

By the above argument if there exists an $S(t, k, v)$ system then

$$\binom{v-i}{t-i} \div \binom{k-i}{t-i} \in \mathbb{Z} \quad \forall i < t$$

Here the first binomial coefficient is the number of sets of t points in which i are fixed. The second is the number of sets of t points that are in a block where i points are fixed.

(27.3.8) The Singleton Bound

Theorem 65 (Singleton Bound) $A_q(n, d) \leqslant q^{n-d+1}$.

Proof. If C is any $q - (n, M, d)$ code let C' be obtained from C by deleting the last $d-1$ co-ordinates, i.e. puncturing C some $d-1$ times. The M codewords in C' are distinct as $D(C) = d$ so $|C| = M = |C'| \leqslant q^{n-(d-1)}$. But this is true for any C, so $A_q(n, d) = \max_{\text{all codes}}\{M\} \leqslant q^{n-d+1}$. □

BCH Codes

Say $\mathrm{GF}\,(q) = \{a_1, a_2, \ldots, a_q\}$ and let A be the Vandermonde matrix

$$A = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ a_1 & a_1 & \ldots & a_q \\ a_1^2 & a_2^2 & \ldots & a_q^2 \\ \vdots & \vdots & \ddots & \vdots \\ a_1^{q-1} & a_2^{q-1} & \ldots & a_q^{q-1} \end{pmatrix}$$

It is not difficult to show (by induction on the size of A) that $\det A = \prod_{i>j} (a_i - a_j)$. Let

$$H = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ a_1 & a_1 & \ldots & a_n \\ a_1^2 & a_2^2 & \ldots & a_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ a_1^{d-2} & a_2^{d-2} & \ldots & a_n^{d-2} \end{pmatrix}$$

and let C be the code with parity check matrix H and taken over $\mathrm{GF}\,(q)$ then C has redundancy $d-1$. The dimension of the code is $n-(d-1) = n-d+1$. Any set of $d-1$ columns of H forms a Vandermonde matrix which has non-zero determinant meaning that no combination of any such $d-1$ columns are linearly dependent. Therefore C has minimal distance d and dimension $n-d+1$ so C reaches the Singleton bound.