birth, $x_7x_8x_9$ is a personal number and x_{10} and x_{11} are check digits defined by

$$x_{10} \equiv -(2x_9 + 5x_8 + 4x_7 + 9x_6 + 8x_5 + x_4 + 6x_3 + 7x_2 + 3x_1) \pmod{11}$$

and

$$x_{11} \equiv -(2x_{10} + 3x_9 + 4x_8 + 5x_7 + 6x_6 + 7x_5 + 2x_4 + 3x_3 + 4x_2 + 5x_1) \pmod{11}.$$

Write down a parity-check matrix for the code (regarded as a code over $GF(11)$). If the code is used only for error detection, will all double errors be detected? If not, which double errors will fail to be detected?

12 Cyclic codes

Cyclic codes form an important class of codes for several reasons. From a theoretical point of view they possess a rich algebraic structure, while practically they can be efficiently implemented by means of simple devices known as shift registers. Furthermore, many important codes, such as binary Hamming codes, Golay codes and BCH codes, are equivalent to cyclic codes.

Definition A code C is *cyclic* if (i) C is a linear code and (ii) any cyclic shift of a codeword is also a codeword, i.e. whenever $a_0a_1 \cdots a_{n-1}$ is in C, then so is $a_{n-1}a_0a_1 \cdots a_{n-2}$.

Examples 12.1 (i) The binary code $\{000, 101, 011, 110\}$ is cyclic.

(ii) The code of Example 2.23, which we now know as the Hamming code Ham $(3, 2)$, is cyclic. (Note that each codeword of the form $\mathbf{a}_i$ is the first cyclic shift of its predecessor and so is each $\mathbf{b}_i$.)

(iii) The binary linear code $\{0000, 1001, 0110, 1111\}$ is not cyclic, but it is *equivalent* to a cyclic code; interchanging the third and fourth coordinates gives the cyclic code $\{0000, 1010, 0101, 1111\}$.

(iv) Consider the ternary Hamming code Ham $(2, 3)$ with generator matrix $\begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 2 \end{bmatrix}$. From the list of codewords found in Exercise 5.7, we see that the code is not cyclic. But is Ham $(2, 3)$ equivalent to a cyclic code? The answer will be given in Example 12.13 (see also Exercise 12.22).

When considering cyclic codes we number the coordinate positions $0, 1, \ldots, n-1$. This is because it is useful to let a vector $a_0a_1 \cdots a_{n-1}$ in $V(n, q)$ correspond to the polynomial $a_0 + a_1x + \cdots + a_{n-1}x^{n-1}$.

Polynomials

From now on we will denote the field $GF(q)$ by F_q, or simply by F (with q understood). We denote by $F[x]$ the set of polynomials in x with coefficients in F. If $f(x) = f_0 + f_1 x + \cdots + f_m x^m$ is a polynomial with $f_m \neq 0$, then m is called the *degree* of $f(x)$, denoted $\deg f(x)$. (By convention the degree of the zero polynomial is $-\infty$.) The coefficient f_m is then called the *leading coefficient*. A polynomial is called *monic* if its leading coefficient is 1.

Polynomials in $F[x]$ can be added, subtracted and multiplied in the usual way. $F[x]$ is an example of an algebraic structure known as a *ring*, for it satisfies the first seven of the eight field axioms (see Chapter 3). Note that $F[x]$ is not a field since polynomials of degree greater than zero do not have multiplicative inverses. Observe also that if $f(x)$, $g(x) \in F[x]$, then $\deg(f(x)g(x)) = \deg f(x) + \deg g(x)$.

The division algorithm for polynomials

The division algorithm states that, for every pair of polynomials $a(x)$ and $b(x) \neq 0$ in $F[x]$, there exists a unique pair of polynomials $q(x)$, the quotient, and $r(x)$, the remainder, such that

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

where $\deg r(x) < \deg b(x)$.

This is analogous to the familiar division algorithm for the ring $\mathbb{Z}$ of integers. The polynomials $q(x)$ and $r(x)$ can be obtained by ordinary long division of polynomials.

For example, in $F_2[x]$, we can divide $x^3 + x + 1$ by $x^2 + x + 1$ as follows.

$$\begin{array}{r} x+1 \\ x^2+x+1 \;\overline{)\; x^3 \quad +x+1} \\ \underline{x^3+x^2+x} \\ \quad x^2 \quad +1 \\ \underline{x^2+x+1} \\ \quad x \end{array}$$

Hence $x^3 + x + 1 = (x+1)(x^2 + x + 1) + x$ is the desired expression of $x^3 + x + 1$ as $q(x)(x^2 + x + 1) + r(x)$.

The ring of polynomials modulo $f(x)$

The ring $F[x]$ of polynomials over F is analogous in many ways to the ring $\mathbb{Z}$ of integers. Just as we can consider integers modulo some fixed integer m to get the ring Z_m (see Chapter 3), we can consider polynomials in $F[x]$ modulo some fixed polynomial $f(x)$.

Let $f(x)$ be a fixed polynomial in $F[x]$. Two polynomials $g(x)$ and $h(x)$ in $F[x]$ are said to be *congruent modulo* $f(x)$, symbolized by

$$g(x) \equiv h(x) \pmod{f(x)},$$

if $g(x) - h(x)$ is divisible by $f(x)$.

By the division algorithm, any polynomial $a(x)$ in $F[x]$ is congruent modulo $f(x)$ to a unique polynomial $r(x)$ of degree less than $\deg f(x)$; $r(x)$ is just the principal remainder when $a(x)$ is divided by $f(x)$.

We denote by $F[x]/f(x)$ the set of polynomials in $F[x]$ of degree less than $\deg f(x)$, with addition and multiplication carried out modulo $f(x)$ as follows.

Suppose $a(x)$ and $b(x)$ belong to $F[x]/f(x)$. Then the sum $a(x) + b(x)$ in $F[x]/f(x)$ is the same as the sum in $F[x]$, because $\deg(a(x) + b(x)) < \deg f(x)$. The product $a(x)b(x)$ in $F[x]/f(x)$ is the unique polynomial of degree less than $\deg f(x)$ to which $a(x)b(x)$ (as a product in $F[x]$) is congruent modulo $f(x)$.

For example, let us calculate $(x+1)^2$ in $F_2[x]/(x^2 + x + 1)$. We have

$$(x+1)^2 = x^2 + 2x + 1 = x^2 + 1 \equiv x \pmod{x^2 + x + 1}.$$

Thus $(x+1)^2 = x$ in $F_2[x]/(x^2 + x + 1)$.

Just as Z_m is a ring, so also is $F[x]/f(x)$; it is called the *ring of polynomials (over F) modulo $f(x)$*.

If $f(x) \in F_q[x]$ has degree n, then the ring $F_q[x]/f(x)$ consists of polynomials of degree $\leq n-1$. Each of the n coefficients of such a polynomial belongs to F_q and so

$$|F_q[x]/f(x)| = q^n.$$

Example 12.2 The addition and multiplication tables for $F_2[x]/$

(x^2+x+1) are easily found to be:

+	0	1	x	$1+x$
0	0	1	x	$1+x$
1	1	0	$1+x$	x
x	x	$1+x$	0	1
$1+x$	$1+x$	x	1	0

·	0	1	x	$1+x$
0	0	0	0	0
1	0	1	x	$1+x$
x	0	x	$1+x$	1
$1+x$	0	$1+x$	1	x

We see that this is more than just a ring. Every non-zero element has a multiplicative inverse and so $F_2[x]/(x^2+x+1)$ is actually a field. In fact, we have precisely the field of order 4 given in Example 3.6(3), with x and $1+x$ corresponding to a and b respectively.

It is certainly not the case that $F[x]/f(x)$ is a field for any choice of $f(x)$; consider, for example, the multiplication table of $F_2[x]/(x^2+1)$ (see Exercise 12.2). The special property of $f(x)$ which makes $F[x]/f(x)$ a field is that of being 'irreducible', which we now define.

Definition A polynomial $f(x)$ in $F[x]$ is said to be *reducible* if $f(x)=a(x)b(x)$, where $a(x), b(x) \in F[x]$ and $\deg a(x)$ and $\deg b(x)$ are both smaller than $\deg f(x)$. If $f(x)$ is not reducible, it is called *irreducible*.

Just as any positive integer can be factorized uniquely into a product of prime numbers, any monic polynomial in $F[x]$ can be factorized uniquely into a product of irreducible monic polynomials.

The following simple observations are often useful when factorizing a polynomial.

Lemma 12.3

(i) A polynomial $f(x)$ has a linear factor $x-a$ if and only if $f(a)=0$.

(ii) A polynomial $f(x)$ in $F[x]$ of degree 2 or 3 is irreducible if and only if $f(a)\neq 0$ for all a in F.

(iii) Over any field, $x^n-1=(x-1)(x^{n-1}+x^{n-2}+\cdots+x+1)$ (the second factor may well be further reducible).

Proof (i) If $f(x)=(x-a)g(x)$, then certainly $f(a)=0$. On the other hand, suppose $f(a)=0$. By the division algorithm, $f(x)=$

$q(x)(x-a)+r(x)$, where $\deg r(x)<1$. So $r(x)$ is a constant, which must be zero since $0=f(a)=r(a)$.

(ii) A polynomial of degree 2 or 3 is reducible if and only if it has at least one linear factor. The result is now immediate from (i).

(iii) By (i), $x-1$ is a factor of x^n-1 and long division of x^n-1 by $x-1$ gives the other factor.

Example 12.4 (i) Factorize x^3-1 in $F_2[x]$ into irreducible polynomials.

(ii) Factorize x^3-1 in $F_3[x]$ into irreducible polynomials.

Solution By 12.3(iii), $x^3-1=(x-1)(x^2+x+1)$ over any field.

(i) By 12.3(ii), x^2+x+1 is irreducible in $F_2[x]$.

(ii) By 12.3(i), in $F_3[x]$, $x-1$ is a factor of x^2+x+1, and we get the factorization $x^3-1=(x-1)^3$.

The finite fields $GF(p^h)$, $h>1$

The property in $F[x]$ of a polynomial being irreducible corresponds exactly to the property in $\mathbb{Z}$ of a number being prime. We showed in Theorem 3.5 that the ring Z_m is a field if and only if m is prime and the following may be proved in exactly the same way.

Theorem 12.5 The ring $F[x]/f(x)$ is a field if and only if $f(x)$ is irreducible in $F[x]$.

Proof This is left to Exercise 12.3.

Although we do not show it here, it can be shown that for any prime number p and for any positive integer h, there *exists* an irreducible polynomial over $GF(p)$ of degree h. This result, together with Theorem 12.5, gives the existence of the fields $GF(p^h)$ for all integers $h\geq 1$. As we remarked in Theorem 3.2, these are essentially the only finite fields.

Back to cyclic codes

Returning from our excursion to look at fields of general order, we now fix $f(x)=x^n-1$ for the remainder of the chapter, for we

shall soon see that the ring $F[x]/(x^n-1)$ of polynomials modulo x^n-1 is the natural one to consider in the context of cyclic codes. For simplicity we shall write $F[x]/(x^n-1)$ as R_n, where the field $F=F_q$ will be understood.

Since $x^n \equiv 1 \pmod{x^n-1}$, we can reduce any polynomial modulo x^n-1 simply by replacing x^n by 1, x^{n+1} by x, x^{n+2} by x^2 and so on. There is no need to write out long divisions by x^n-1.

Let us now identify a vector $a_0a_1\cdots a_{n-1}$ in $V(n,q)$ with the polynomial

$$a(x)=a_0+a_1x+\cdots+a_{n-1}x^{n-1}$$

in R_n. We shall simultaneously view a code as a subset of $V(n,q)$ and as a subset of R_n. Note that addition of vectors and multiplication of a vector by a scalar in R_n corresponds exactly to those operations in $V(n,q)$. Now consider what happens when we multiply the polynomial $a(x)$ by x. In R_n, we have

$$\begin{aligned} x\cdot a(x) &= a_0x+a_1x^2+\cdots+a_{n-1}x^n \\ &= a_{n-1}+a_0x+\cdots+a_{n-2}x^{n-1}, \end{aligned}$$

which is the vector $a_{n-1}a_0\cdots a_{n-2}$. Thus *multiplying by x corresponds to performing a single cyclic shift*. Multiplying by x^m corresponds to a cyclic shift through m positions.

The following theorem gives the algebraic characterization of cyclic codes.

Theorem 12.6 A code C in R_n is a cyclic code if and only if C satisfies the following two conditions:

(i) $a(x), b(x)\in C \Rightarrow a(x)+b(x)\in C$,

(ii) $a(x)\in C$ and $r(x)\in R_n \Rightarrow r(x)a(x)\in C$.

[Note that (ii) does not just say that C must be closed under multiplication; it says that C must be closed under multiplication *by any element of* R_n. The reader who is familiar with ring theory will recognize that Theorem 12.6 says that cyclic codes are precisely the 'ideals' of the ring R_n.]

Proof Suppose C is a cyclic code in R_n. Then C is linear and so (i) holds. Now suppose $a(x)\in C$ and $r(x)=r_0+r_1x+\cdots+r_{n-1}x^{n-1}\in R_n$. Since multiplication by x corresponds to a cyclic shift, we have $x\cdot a(x)\in C$ and then $x\cdot(xa(x))=x^2a(x)\in C$ and

so on. Hence

$$r(x)a(x)=r_0a(x)+r_1xa(x)+\cdots+r_{n-1}x^{n-1}a(x)$$

is also in C since each summand is in C. Thus (ii) also holds.

Now suppose (i) and (ii) hold. Taking $r(x)$ to be a scalar, the conditions imply that C is linear. Taking $r(x)=x$ in (ii) shows that C is cyclic.

We now give an easy way of constructing examples of cyclic codes.

Let $f(x)$ be any polynomial in R_n and let $\langle f(x)\rangle$ denote the subset of R_n consisting of all multiples of $f(x)$ (reduced modulo x^n-1), i.e.

$$\langle f(x)\rangle=\{r(x)f(x)\mid r(x)\in R_n\}.$$

Theorem 12.7 For any $f(x)\in R_n$, the set $\langle f(x)\rangle$ is a cyclic code; it is called the code *generated* by $f(x)$.

Proof We check conditions (i) and (ii) of Theorem 12.6.

(i) If $a(x)f(x)$ and $b(x)f(x)\in\langle f(x)\rangle$, then

$$a(x)f(x)+b(x)f(x)=(a(x)+b(x))f(x)\in\langle f(x)\rangle.$$

(ii) If $a(x)f(x)\in\langle f(x)\rangle$ and $r(x)\in R_n$, then

$$r(x)(a(x)f(x))=(r(x)a(x))f(x)\in\langle f(x)\rangle.$$

Example 12.8 Consider the code $C=\langle 1+x^2\rangle$ in R_3 (with $F=GF(2)$). Multiplying $1+x^2$ by each of the eight elements of R_3 (and reducing modulo x^3-1) produces only four distinct codewords, namely $0, 1+x, 1+x^2$ and $x+x^2$. Thus C is the code $\{000, 110, 101, 011\}$ of Example 12.1(i).

We next show that the above easy way of constructing cyclic codes is essentially the *only* way, i.e. any cyclic code can be generated by a polynomial. (In the terminology of ring theory, this says that every ideal in R_n is a 'principal ideal'.)

Theorem 12.9 Let C be a non-zero cyclic code in R_n. Then

(i) there exists a unique monic polynomial $g(x)$ of smallest degree in C,

(ii) $C=\langle g(x)\rangle$,

(iii) $g(x)$ is a factor of x^n-1.

Proof (i) Suppose $g(x)$ and $h(x)$ are both monic polynomials in C of smallest degree. Then $g(x)-h(x) \in C$ and has smaller degree. This gives a contradiction if $g(x) \neq h(x)$, for then a suitable scalar multiple of $g(x)-h(x)$ is monic, is in C, and is of smaller degree then $\deg g(x)$.

(ii) Suppose $a(x) \in C$. By the division algorithm for $F[x]$, $a(x)=q(x)g(x)+r(x)$, where $\deg r(x) < \deg g(x)$. But $r(x)=a(x)-q(x)g(x) \in C$, by the properties of a cyclic code given in Theorem 12.6. By the minimality of $\deg g(x)$, we must have $r(x)=0$ and so $a(x) \in \langle g(x) \rangle$.

(iii) By the division algorithm,

$$x^n - 1 = q(x)g(x) + r(x),$$

where $\deg r(x) < \deg g(x)$. But then $r(x) \equiv -q(x)g(x) \pmod{x^n-1}$, and so $r(x) \in \langle g(x) \rangle$. By the minimality of $\deg g(x)$, we must have $r(x)=0$, which implies that $g(x)$ is a factor of x^n-1.

Definition In a non-zero cyclic code C the monic polynomial of least degree, given by Theorem 12.9, is called the *generator polynomial* of C.

Note that a cyclic code C may contain polynomials other than the generator polynomial which also generate C. For example, the code of Example 12.8 is generated by $1+x^2$, but its generator polynomial is $1+x$.

The third part of Theorem 12.9 gives a recipe for finding all cyclic codes of given length n. All we need is the factorization of x^n-1 into irreducible monic polynomials.

Example 12.10 We will find all the binary cyclic codes of length 3. By Example 12.4(i), $x^3-1=(x+1)(x^2+x+1)$, where $x+1$ and x^2+x+1 are irreducible over $GF(2)$. So, by Theorem 12.9, the following is a complete list of binary cyclic codes of length 3.

Generator polynomial	Code in R_3	Corresponding Code in $V(3,2)$
1	all of R_3	all of $V(3,2)$
$x+1$	$\{0, 1+x, x+x^2, 1+x^2\}$	$\{000, 110, 011, 101\}$
x^2+x+1	$\{0, 1+x+x^2\}$	$\{000, 111\}$
$x^3-1=0$	$\{0\}$	$\{000\}$

Lemma 12.11 Let $g(x)=g_0+g_1x+\cdots+g_rx^r$ be the generator polynomial of a cyclic code. Then g_0 is non-zero.

Proof Suppose $g_0=0$. Then $x^{n-1}g(x)=x^{-1}g(x)$ is a codeword of C of degree $r-1$, contradicting the minimality of $\deg g(x)$.

By definition, a cyclic code is linear. It would be handy if immediately from the generator polynomial $g(x)$ we could deduce the dimension of the code and also write down a generator matrix. The next theorem shows that we can do both.

Theorem 12.12 Suppose C is a cyclic code with generator polynomial

$$g(x)=g_0+g_1x+\cdots+g_rx^r$$

of degree r. Then $\dim(C)=n-r$ and a generator matrix for C is

$$G=\begin{bmatrix} g_0 & g_1 & g_2 & \cdots & g_r & 0 & 0 & \cdots & 0 \\ 0 & g_0 & g_1 & g_2 & \cdots & g_r & 0 & \cdots & 0 \\ 0 & 0 & g_0 & g_1 & g_2 & \cdots & g_r & & \vdots \\ \vdots & \vdots & & \ddots & \ddots & \ddots & & \ddots & 0 \\ 0 & 0 & \cdots & 0 & g_0 & g_1 & g_2 & \cdots & g_r \end{bmatrix}$$

Proof The $n-r$ rows of the above matrix G are certainly linearly independent because of the echelon of non-zero g_0s with 0s below. These $n-r$ rows represent the codewords $g(x)$, $xg(x)$, $x^2g(x), \ldots, x^{n-r-1}g(x)$, and it remains only to show that every codeword in C can be expressed as a linear combination of them. The proof of Theorem 12.9(ii) shows that if $a(x)$ is a codeword of C, then

$$a(x)=q(x)g(x)$$

for some polynomial $q(x)$, and that this is an equality of polynomials within $F[x]$, not requiring any reduction modulo x^n-1. Since $\deg a(x) < n$, it follows that $\deg q(x) < n-r$. Hence

$$\begin{aligned} q(x)g(x) &= (q_0+q_1x+\cdots+q_{n-r-1}x^{n-r-1})g(x) \\ &= q_0g(x)+q_1xg(x)+\cdots+q_{n-r-1}x^{n-r-1}g(x), \end{aligned}$$

which is the desired linear combination.

Example 12.13 Find all the ternary cyclic codes of length 4 and write down a generator matrix for each of them.

Solution Over $GF(3)$, the factorization of x^4-1 into irreducible polynomials is

$$x^4-1=(x-1)(x^3+x^2+x+1)=(x-1)(x+1)(x^2+1).$$

So there are $2^3=8$ divisors of x^4-1 in $F_3[x]$, each of which generates a cyclic code. By Theorem 12.9, these are the only ternary cyclic codes of length 4. The codes are specified below by their generator polynomials, and the corresponding generator matrices are given by Theorem 12.12. Note that neither of the two-dimensional codes has minimum distance 3 and so the ternary Hamming $[4, 2, 3]$-code is not cyclic, thus answering the question posed in Example 12.1(iv).

Generator polynomial	Generator matrix
1	$[I_4]$
$x-1$	$\begin{bmatrix} -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}$
$x+1$	$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$
x^2+1	$\begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$
$(x-1)(x+1)=x^2-1$	$\begin{bmatrix} -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix}$
$(x-1)(x^2+1)=x^3-x^2+x-1$	$[-1 \quad 1 \quad -1 \quad 1]$
$(x+1)(x^2+1)=x^3+x^2+x+1$	$[1 \quad 1 \quad 1 \quad 1]$
$x^4-1=0$	$[0 \quad 0 \quad 0 \quad 0]$

The check polynomial and the parity-check matrix of a cyclic code

The generator matrix of a cyclic code as given by Theorem 12.12 is not in standard form. Our usual method of writing down a parity-check matrix from the standard form of G (via Theorem 7.6) is therefore not appropriate for cyclic codes. However, there is a natural choice of parity-check matrix for a cyclic code. This is closely related to the so-called 'check polynomial', which we define first.

Let C be a cyclic $[n, k]$-code with generator polynomial $g(x)$. By Theorem 12.9, $g(x)$ is a factor of x^n-1 and so

$$x^n-1=g(x)h(x),$$

for some polynomial $h(x)$. Since $g(x)$ is monic, so also is $h(x)$. By Theorem 12.12, $g(x)$ has degree $n-k$ and so $h(x)$ has degree k. This polynomial $h(x)$ is called the *check polynomial* of C. The reason for this name is apparent from the following theorem.

Theorem 12.14 Suppose C is a cyclic code in R_n with generator polynomial $g(x)$ and check polynomial $h(x)$. Then an element $c(x)$ of R_n is a codeword of C if and only if $c(x)h(x)=0$.

Proof First note that, in R_n, $g(x)h(x)=x^n-1=0$.
Hence $c(x)\in C \Rightarrow c(x)=a(x)g(x)$, for some $a(x)\in R_n$,

$$\begin{aligned} \Rightarrow c(x)h(x) &= a(x)g(x)h(x) \\ &= a(x)\cdot 0 \\ &= 0. \end{aligned}$$

On the other hand, suppose $c(x)$ satisfies $c(x)h(x)=0$. By the division algorithm, $c(x)=q(x)g(x)+r(x)$, where $\deg r(x)<n-k$. Then $c(x)h(x)=0$ implies that $r(x)h(x)=0$, i.e. $r(x)h(x)\equiv 0 \pmod{x^n-1}$. But $\deg(r(x)h(x))<n-k+k=n$, and so $r(x)h(x)=0$ in $F[x]$. Hence $r(x)=0$, and then $c(x)=q(x)g(x)\in C$.

In view of Theorem 12.14 and the fact that $\dim(\langle h(x)\rangle)=n-k=\dim(C^\perp)$, we might easily be fooled into thinking that $h(x)$ generates the dual code $C^\perp$. In general this is not so. The point is that the product of $c(x)$ and $h(x)$ being zero in R_n is not the same thing as the corresponding vectors in $V(n, q)$ being orthogonal. In the next theorem, however, we see that the condition $c(x)h(x)=0$ in R_n does imply some useful orthogonality relations which lead to a natural choice of parity-check matrix.

Theorem 12.15 Suppose C is a cyclic $[n, k]$-code with check polynomial

$$h(x) = h_0 + h_1 x + \cdots + h_k x^k.$$

Then

(i) a parity-check matrix for C is

$$H = \begin{bmatrix} h_k & h_{k-1} & \cdots & h_0 & 0 & 0 & \cdots & 0 \\ 0 & h_k & h_{k-1} & \cdots & h_0 & 0 & \cdots & 0 \\ & \ddots & \ddots & & \ddots & & & \vdots \\ & & \ddots & \ddots & & \ddots & & 0 \\ 0 & \cdots & 0 & h_k & h_{k-1} & \cdots & & h_0 \end{bmatrix}$$

(ii) $C^\perp$ is a cyclic code generated by the polynomial

$$\bar{h}(x) = h_k + h_{k-1} x + \cdots + h_0 x^k.$$

Proof (i) By Theorem 12.14, a polynomial $c(x) = c_0 + c_1 x + \cdots + c_{n-1} x^{n-1}$ is a codeword if and only if $c(x)h(x) = 0$. Now for $c(x)h(x)$ to be zero, then in particular the coefficients of $x^k, x^{k+1}, \ldots, x^{n-1}$ must all be zero, i.e.

$$\begin{aligned} c_0 h_k + c_1 h_{k-1} + \cdots + c_k h_0 &= 0 \\ c_1 h_k + c_2 h_{k-1} + \cdots + c_{k+1} h_0 &= 0 \\ \ddots \qquad\qquad \ddots &\ \vdots \\ c_{n-k-1} h_k + \cdots + c_{n-1} h_0 &= 0. \end{aligned}$$

Thus any codeword $c_0 c_1 \cdots c_{n-1}$ of C is orthogonal to the vector $h_k h_{k-1} \cdots h_0 00 \cdots 0$ and to its cyclic shifts. So the rows of the matrix H given in the statement of the theorem are all codewords of $C^\perp$. We have already observed that $h(x)$ is monic of degree k and so $h_k = 1$; thus the echelon of 1s with zeros below in H ensures that the rows of H are linearly independent. The number of rows of H is $n - k$, which is the dimension of $C^\perp$. Hence H is a generator matrix of $C^\perp$, i.e. a parity-check matrix for C.

(ii) If we can show that $\bar{h}(x)$ is a factor of $x^n - 1$, then it will follow from Theorem 12.12 that $\langle \bar{h}(x) \rangle$ is a cyclic code whose generator matrix is the above matrix H, and hence that $\langle \bar{h}(x) \rangle = C^\perp$. We observe that $\bar{h}(x) = x^k h(x^{-1})$. Since $h(x^{-1})g(x^{-1}) = (x^{-1})^n - 1$, we have $x^k h(x^{-1}) x^{n-k} g(x^{-1}) = x^n(x^{-n} - 1) = 1 - x^n$, and so $\bar{h}(x)$ is indeed a factor of $x^n - 1$.

Remarks (i) The polynomial $\bar{h}(x) = x^k h(x^{-1}) = h_k + h_{k-1} x + \cdots + h_0 x^k$ is called the *reciprocal polynomial* of $h(x)$; its coefficients are those of $h(x)$ *in reverse order*.

(ii) We may regard $\bar{h}(x)$ as the generator polynomial of $C^\perp$, though strictly speaking, in the non-binary case, one ought to multiply it by the scalar h_0^{-1} to make it monic.

(iii) The polynomial $h(x^{-1}) = x^{n-k} \bar{h}(x)$ is a member of $C^\perp$.

We have not yet discussed the minimum distance of cyclic codes. There are some classes of cyclic codes for which useful lower bounds on the minimum distance are known. For example, cyclic BCH codes can be constructed to have 'designed minimum distance' while there are codes called quadratic residue codes which satisfy a 'square root bound'. These codes and bounds are well treated in several of the more advanced texts. We concentrate here on finding the minimum distances of two particularly interesting cyclic codes, namely the two Golay codes. Our methods, while aimed directly at the codes in hand, nevertheless provide some insights into the more general methods.

The binary Golay code

In Chapter 9, we proved the existence of a perfect binary $[23, 12, 7]$-code G_{23} by exhibiting a generator matrix. We now show that this Golay code can be constructed in a more natural way as a cyclic code. The only knowledge we shall assume in advance is the factorization of $x^{23} - 1$ over $GF(2)$. [There is a clever method of finding the factors of $x^n - 1$ over $GF(q)$ in general (see, for example, Chapter 7, §5, of MacWilliams and Sloane (1977)) but we shall not dwell on this here. Alternatively one may find the factors by consulting tables (see, e.g., the same reference for a list of factors of $x^n - 1$ over $GF(2)$ for $n \leq 63$).]

We begin then with the factorization

$$\begin{aligned} x^{23} - 1 &= (x - 1)(x^{11} + x^{10} + x^6 + x^5 + x^4 + x^2 + 1) \\ &\quad \times (x^{11} + x^9 + x^7 + x^6 + x^5 + x + 1) \\ &= (x - 1) g_1(x) g_2(x), \text{ say.} \end{aligned}$$

Let C_1 be the code $\langle g_1(x) \rangle$ and let C_2 be the code $\langle g_2(x) \rangle$. By Theorem 12.12, C_1 is a $[23, 12]$-code. The object of the next few pages is to show that the minimum distance of C_1 is 7.

We observe that the polynomials $g_1(x)$ and $g_2(x)$ are reciprocals of each other, and so C_2 is equivalent to C_1. Remarkably, the knowledge that $x^{23}-1=(x-1)g_1(x)\bar{g}_1(x)$, where $\bar{g}_1(x)$ denotes the reciprocal of $g_1(x)$, is all we need to show that $d(C_1)=7$; we do not actually need to know what $g_1(x)$ is.

Remark 12.16 Although we do not show it here, x^p-1 has a factorization over $GF(2)$ of the form $(x-1)g_1(x)g_2(x)$, where $\langle g_1(x)\rangle$ and $\langle g_2(x)\rangle$ are equivalent codes, whenever p is a prime number of the form $8m\pm1$. If p is of the form $8m-1$ we also have $g_2(x)=\bar{g}_1(x)$. For example,

$$x^7-1=(x-1)(x^3+x+1)(x^3+x^2+1)$$

and

$$x^{31}-1=(x-1)g(x)\bar{g}(x),$$

where $g(x)=1+x^3+x^8+x^9+x^{13}+x^{14}+x^{15}$.

In view of Remark 12.16, we prove the next two lemmas for p equal to a general odd prime number rather than just for $p=23$. We will denote the vector $1+x+x^2+\cdots+x^{p-1}$ consisting of all 1s by $\mathbf{1}$. Note that if $x^p-1=(x-1)g_1(x)g_2(x)$, then $g_1(x)g_2(x)=\mathbf{1}$.

Lemma 12.17 Suppose that $x^p-1=(x-1)g_1(x)g_2(x)$ over $GF(2)$, and that $\langle g_1(x)\rangle$ and $\langle g_2(x)\rangle$ are equivalent codes. Let $a(x)$ be a codeword of $\langle g_1(x)\rangle$ of *odd* weight w. Then

(i) $w^2\geqslant p$

(ii) if also $g_2(x)=\bar{g}_1(x)$, then $w^2-w+1\geqslant p$.

Proof (i) Since $\langle g_2(x)\rangle$ is equivalent to $\langle g_1(x)\rangle$, there is some codeword $b(x)$ in $\langle g_2(x)\rangle$ also of weight w. Now $a(x)b(x)$ is a multiple of $g_1(x)g_2(x)=\mathbf{1}$, and so $a(x)b(x)=0$ or $\mathbf{1}$. Since w is odd, we have $a(1)b(1)=w\cdot w\equiv 1 \bmod (2)$, and so we must have $a(x)b(x)=1+x+\cdots+x^{p-1}$. But $a(x)b(x)$ has at most w^2 non-zero coefficients and so $w^2\geqslant p$.

(ii) If $g_2(x)=\bar{g}_1(x)$, then the codewords of $\langle g_2(x)\rangle$ are just the reciprocals of the codewords of $\langle g_1(x)\rangle$. In particular we may take $b(x)$ to be $a(x^{-1})$ in the proof of (i) to get

$$a(x)a(x^{-1})=1+x+x^2+\cdots+x^{p-1}.$$

But w of the w^2 terms in the product $a(x)a(x^{-1})$ are 1 and so the maximum weight of $a(x)a(x^{-1})$ is w^2-w+1.

Corollary 12.18 If, with the hypotheses of Lemma 12.17, it is also known that the minimum distance d of $\langle g_1(x)\rangle$ is *odd*, then d satisfies the *square root bound*

$$d\geqslant\sqrt{p},$$

while if also $g_2(x)=\bar{g}_1(x)$, this can be improved to

$$d^2-d+1\geqslant p.$$

By Lemma 12.17(ii), our [23, 12]-code C_1 has no words of *odd* weight less than 7, because $5^2-5+1<23$. There is an ingenious way of showing that C_1, and more generally any so-called quadratic residue (QR) code (we do not define QR codes here, but simply remark that C_1 is an example of such a code), must have odd minimum distance and therefore must satisfy the square root bound. The argument, which involves showing that an extended QR code has a transitive automorphism group, is beyond the scope of the present book. As our main aim is merely to find the minimum distance of the Golay code C_1, the following lemma will suffice.

Lemma 12.19 Suppose p is an odd prime number and that, over $GF(2)$, $x^p-1=(x-1)g_1(x)\bar{g}_1(x)$. Let $a(x)$ be a codeword of $\langle g_1(x)\rangle$ of even weight w. Then

(i) $w\equiv 0 \pmod 4$

(ii) $w\neq 4$ unless $p=7$.

Proof (i) As in the proof of Lemma 12.17, we have $a(x)a(x^{-1})=0$ or $\mathbf{1}$. Since $a(x)$ has even weight, $a(1)=0$, and so $a(x)a(x^{-1})=0$. Suppose $a(x)=x^{e_1}+x^{e_2}+\cdots+x^{e_w}$. Then

$$a(x)a(x^{-1})=\sum_{i=1}^{w}\sum_{j=1}^{w}x^{e_i-e_j}=0$$

in R_p. Of the w^2 summands, w are equal to 1 (the terms with $i=j$), and these sum to $0 \pmod 2$. So the remaining w^2-w terms $x^{e_i-e_j}(i\neq j)$ must cancel each other out in pairs. Now if $x^{e_i-e_j}=x^{e_k-e_l}$ then $x^{e_j-e_i}=x^{e_l-e_k}$, and so the terms must actually cancel four at a time. Thus

$$w^2-w\equiv 0 \pmod 4 \qquad \text{and so} \qquad w\equiv 0 \pmod 4.$$

(ii) Suppose $w=4$. Without loss of generality (via a suitable

cyclic shift), suppose $a(x) = 1 + x^i + x^j + x^k$, where i, j, k are distinct and $1 < i, j, k < p$. Then $(1 + x^i + x^j + x^k)(1 + x^{-i} + x^{-j} + x^{-k}) = 0$.

Thus the six sets $\{i, -i\}$, $\{j, -j\}$, $\{k, -k\}$, $\{i - j, j - i\}$, $\{i - k, k - i\}$ and $\{j - k, k - j\}$ must split into three matching pairs, under congruence modulo p. By symmetry there is no loss in assuming i is congruent to one of $-j$, $j - i$ or $j - k$.

Case 1 Suppose $i \equiv j - k \pmod{p}$. Then $k \equiv j - i$ gives a second match and so the third match must be given by $j \equiv \pm(i - k)$. But $i \equiv j - k$ and $j \equiv i - k$ implies $2k \equiv 0 \pmod{p}$, which is a contradiction since p is an odd prime. Likewise, $i \equiv j - k$ and $j \equiv k - i$ implies $2i \equiv 0 \pmod{p}$, which is again a contradiction.

Case 2 Suppose $i \equiv -j \pmod{p}$. Since Case 1 has been ruled out, we must have $k \equiv i - k$ or $k \equiv j - k$ and as the two possibilities are essentially the same, we may assume $k \equiv i - k$, i.e. $i \equiv 2k$. The third match is then given by $i - j \equiv j - k$, which implies $k \equiv -3i \equiv -6k$. Thus $7k \equiv 0 \pmod{p}$, which is a contradiction unless $p = 7$.

Case 3 Suppose $i \equiv j - i \pmod{p}$. To avoid the cases above, we may assume the remaining matches are given by $j \equiv k - j$ and $k \equiv i - k$. But then $k \equiv 2j \equiv 4i \equiv 8k$, again giving $7k \equiv 0 \pmod{p}$.

Remark We observed in Remark 12.16 that $x^7 - 1$ has the form $(x - 1)g(x)\bar{g}(x)$, where $g(x) = x^3 + x + 1$. Since $\langle g(x) \rangle$ contains words of weight 4, the exclusion of case $p = 7$ in Lemma 12.19(ii) is essential.

We have now reached our goal:

Theorem 12.20 Let G_{23} be the binary cyclic code in R_{23} with generator polynomial $g(x) = 1 + x^2 + x^4 + x^5 + x^6 + x^{10} + x^{11}$. Then G_{23} is a perfect $[23, 12, 7]$-code.

Proof We have already observed that

$$x^{23} - 1 = (x - 1)g(x)\bar{g}(x).$$

By Lemma 12.17, the minimum *odd* weight w of G_{23} satisfies $w^2 - w + 1 \geq 23$, which implies that $w \geq 7$. By Lemma 12.19, G_{23} can have no words of *even* weight < 8. As $g(x)$ is a codeword of

weight 7, we have $d(G_{23}) = 7$. Since

$$2^{12}\left\{1 + 23 + \binom{23}{2} + \binom{23}{3}\right\} = 2^{23},$$

the sphere-packing condition (9.1) is satisfied and so G_{23} is perfect.

The code G_{23} is called the *binary Golay code*. It is equivalent to the Golay code as defined in Chapter 9 (cf. the remarks following Problem 9.9).

The ternary Golay code

We now show that the ternary Golay code G_{11} may also be constructed as a cyclic code. Our starting point is the factorization of $x^{11} - 1$ over $GF(3)$:

$$\begin{aligned} x^{11} - 1 &= (x - 1)(x^5 + x^4 - x^3 + x^2 - 1)(x^5 - x^3 + x^2 - x - 1) \\ &= (x - 1)g_1(x)g_2(x), \text{ say.} \end{aligned}$$

Note that $g_2(x) = -x^5 g_1(x^{-1})$ and so $\langle g_1(x) \rangle$ and $\langle g_2(x) \rangle$ are equivalent $[11, 6]$-codes. We shall show that the code $\langle g_1(x) \rangle$ has minimum distance 5.

Theorem 12.21 Let C be the ternary code $\langle g_1(x) \rangle$ in R_{11}, where $g_1(x) = x^5 + x^4 - x^3 + x^2 - 1$. Let D be the subcode of C generated by $(x - 1)g_1(x)$. Let $a(x) = a_0 + a_1 x + \cdots + a_{10}x^{10}$ be a codeword of C of weight w. Then

(i) $a(x) \in D$ if and only if $\sum_{i=0}^{10} a_i = 0$,
(ii) if $a(x) \in D$, then $w \equiv 0 \pmod{3}$,
(iii) if $a(x) \notin D$, then $w \equiv 2 \pmod{3}$,
(iv) if $a(x) \notin D$, then $w \geq 4$,
(v) $w \neq 3$,
(vi) $d(C) = 5$.

Proof (i) Given that $a(x)$ is in C and so is a multiple of $g_1(x)$, we have

$$\begin{aligned} a(x) \in D &\Leftrightarrow a(x) \text{ is a multiple of } (x - 1) \\ &\Leftrightarrow a(1) = 0 \\ &\Leftrightarrow \sum_{i=0}^{10} a_i = 0. \end{aligned}$$

(ii) First observe that, since $a_i^2 \equiv 1 \pmod 3$ for each non-zero coefficient a_i, we have $w \equiv \sum a_i^2 \pmod 3$. By Theorem 12.15(ii), the dual code $D^\perp$ of D is generated by the reciprocal polynomial of $g_2(x)$, which happens to be precisely $-g_1(x)$. Thus $D^\perp = \langle \bar{g}_2(x) \rangle = \langle -g_1(x) \rangle = C$. So D is contained in $D^\perp$, which means that D is self-orthogonal, i.e. the inner product of any two vectors of D is zero. In particular, if $a(x) \in D$, then the inner product of $a(x)$ with itself is zero, i.e. $\sum a_i^2 \equiv 0 \pmod 3$. Thus $a(x) \in D \Rightarrow w \equiv 0 \pmod 3$.

(iii) By Theorem 12.12, D is a code of dimension 5. Also D is contained within the 6-dimensional code C. Since $\mathbf{1} = 1 + x + \cdots + x^{10}$ is in C but not in D, C is the disjoint union of the three cosets $D, \mathbf{1} + D$ and $-\mathbf{1} + D$. Thus any codeword $a(x)$ of C which is not in D is of the form

$$a(x) = d(x) \pm \mathbf{1},$$

for some codeword $d(x) = d_0 + d_1 x + \cdots + d_{10} x^{10} \in D$.

Hence
$$\begin{aligned} w(a(x)) &\equiv \sum_{i=0}^{10} (d_i \pm 1)^2 \\ &\equiv \left(\sum_{i=0}^{10} d_i^2\right) + 11 \pm 2\left(\sum_{i=0}^{10} d_i\right) \\ &\equiv 11 \qquad \text{(by (i) and (ii))} \\ &\equiv 2 \pmod 3. \end{aligned}$$

(iv) Suppose $a(x) \notin D$. Now $a(x)a(x^{-1})$ is a multiple of $g_1(x)g_2(x) = \mathbf{1}$. By (i), $a(1) \neq 0$, and so $a(x)a(x^{-1}) = \pm\mathbf{1}$. Thus $a(x)a(x^{-1})$ has weight 11. But at most w^2 coefficients of $a(x)a(x^{-1})$ are non-zero and so $w^2 \geqslant 11$. Hence $w \geqslant 4$.

(v) Suppose, for a contradiction, that $w = 3$. Then, by a suitable cyclic shift, and multiplication by -1 if necessary, we may suppose $a(x) = 1 \pm x^i \pm x^j$. By (ii) and (iii), $a(x)$ must be in D and so, by (i), we must actually have $a(x) = 1 + x^i + x^j$. Also, $a(x) \in D$ implies that $a(x)a(x^{-1})$ is a multiple of

$$(x-1)g_1(x)g_2(x) = x^{11} - 1 = 0$$

in R_{11}. Thus

$$(1 + x^i + x^j)(1 + x^{-i} + x^{-j}) = 0,$$

giving
$$x^i + x^{-i} + x^j + x^{-j} + x^{j-i} + x^{i-j} = 0.$$

Since i and j are distinct and non-zero we must have $i \equiv -j \equiv j - i \pmod{11}$, which implies that $3j \equiv 0 \pmod{11}$, which is a contradiction.

(vi) It follows from (ii)–(v) that $d(C) \geqslant 5$ and since $g_1(x)$ itself has weight 5, $d(C) = 5$.

The [11, 6, 5] code C of Theorem 12.21 is called the *ternary Golay code*. It is a perfect code because

$$3^6\left\{1 + 2 \cdot 11 + 2^2\binom{11}{2}\right\} = 3^{11},$$

and it is equivalent to the ternary Golay code defined in Chapter 9.

Hamming codes as cyclic codes

We will show that the binary Hamming codes discussed in Chapter 8 are equivalent to cyclic codes. The proof will be incomplete in the sense that we shall assume results previously stated, but left unproved, in the text.

Theorem 12.22 The binary Hamming code Ham $(r, 2)$ is equivalent to a cyclic code.

Proof Let $p(x)$ be an irreducible polynomial of degree r in $F_2[x]$. Then, by Theorem 12.5, the ring $F_2[x]/p(x)$ of polynomials modulo $p(x)$ is actually a field of order 2^r. As was mentioned in Chapter 3, every finite field has a primitive element and so there exists an element α of $F_2[x]/p(x)$ such that $F_2[x]/p(x) = \{0, 1, \alpha, \alpha^2, \ldots, \alpha^{2^r-2}\}$. Let us now identify an element $a_0 + a_1 x + a_2 x^2 + \cdots + a_{r-1} x^{r-1}$ of $F_2[x]/p(x)$ with the column vector

$$\begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{r-1} \end{bmatrix}$$

and consider the binary $r \times (2^r - 1)$ matrix

$$H = [1 \ \alpha \ \alpha^2 \cdots \alpha^{2^r-2}].$$

Let C be the binary linear code having H as parity-check matrix.

Since the columns of H are precisely the distinct non-zero vectors of $V(r,2)$, C is a Hamming code Ham $(r,2)$. Putting $n=2^r-1$ we have

$$C=\{f_0f_1\cdots f_{n-1}\in V(n,2)\mid f_0+f_1\alpha+\cdots+f_{n-1}\alpha^{n-1}=0\}$$
$$=\{f(x)\in R_n\mid f(\alpha)=0 \text{ in } F_2[x]/p(x)\}. \quad (12.23)$$

If $f(x)\in C$ and $r(x)\in R_n$, then $r(x)f(x)\in C$ because $r(\alpha)f(\alpha)=r(\alpha)\cdot 0=0$. So, by Theorem 12.6, this version of Ham $(r,2)$ is cyclic.

Definition If $p(x)$ is an irreducible polynomial of degree r such that x is a primitive element of the field $F[x]/p(x)$, then $p(x)$ is called a *primitive polynomial*.

Theorem 12.24 If $p(x)$ is a primitive polynomial over $GF(2)$ of degree r, then the cyclic code $\langle p(x)\rangle$ is the Hamming code Ham $(r,2)$.

Proof If $p(x)$ is primitive, then (12.23) implies that

$$\text{Ham}\,(r,2)=\{f(x)\in R_n\mid f(x)=0 \text{ in } F_2[x]/p(x)\}$$
$$=\langle p(x)\rangle.$$

Example 12.25 The polynomial x^3+x+1 is irreducible over $GF(2)$ and so $F_2[x]/(x^3+x+1)$ is a field of order 8. Also, x is a primitive element of this field, for

$$F_2[x]/(x^3+x+1)$$
$$=\{0,1,x,x^2,x^3=x+1,x^4=x^2+x,x^5=x^2+x+1,x^6=x^2+1\}.$$

Thus a parity-check matrix for a cyclic version of the Hamming code Ham $(3,2)$ is

$$H=\begin{bmatrix}1&0&0&1&0&1&1\\0&1&0&1&1&1&0\\0&0&1&0&1&1&1\end{bmatrix},$$

wherein the columns represent $1,\alpha,\alpha^2,\ldots,\alpha^6$ as described in the proof of Theorem 12.22, with $\alpha=x$.

Since x^3+x+1 is a primitive polynomial, it is a generator polynomial for Ham $(3,2)$ and so, by Theorem 12.12, a generator matrix for the code is

$$G=\begin{bmatrix}1&1&0&1&0&0&0\\0&1&1&0&1&0&0\\0&0&1&1&0&1&0\\0&0&0&1&1&0&1\end{bmatrix}.$$

Remarks (1) It can be shown that there exists a primitive polynomial of degree r for any r.

(2) We saw in Example 12.13 that the ternary Hamming code Ham $(2,3)$ is not equivalent to a cyclic code. However, Ham (r,q) is equivalent to a cyclic code if r and $q-1$ are relatively prime (see, e.g., Blahut (1983), Theorem 5.5.1).

Concluding remarks on Chapter 12

(1) Cyclic codes were first studied by Prange (1957). Interest was further stimulated by the theorem of Bose and Ray-Chaudhuri (1960) which gave lower bounds on the minimum distance for a large class of cyclic codes. It was quickly discovered that almost every special linear code previously discovered (e.g. Hamming, Golay, Reed–Muller) could be made cyclic.

(2) For a comprehensive treatment of the theory of cyclic codes, see, e.g., MacWilliams and Sloane (1977). For details of the practical implementation of cyclic codes, including the associated circuitry, see, e.g., Blahut (1983) or Lin and Costello (1983).

Exercises 12

12.1 Is each of the following codes (a) cyclic, (b) equivalent to a cyclic code?

(i) the binary code {0000, 1100, 0110, 0011, 1001}
(ii) the binary code {00000, 10110, 01101, 11011}
(iii) the ternary code {0000, 1122, 2211}
(iv) the q-ary repetition code of length n
(v) the binary even-weight code E_n
(vi) the ternary code $\{\mathbf{x}\in V(n,3)\mid w(\mathbf{x})\equiv 0 \pmod 3\}$

(vii) the ternary code

$$\left\{x_1x_2\cdots x_n \in V(n,3) \;\middle|\; \sum_{i=1}^{n} x_i \equiv 0 \pmod 3\right\}$$

12.2 Write out the multiplication table for $F_2[x]/(x^2+1)$. Explain why $F_2[x]/(x^2+1)$ is not a field.

12.3 Write out a proof of Theorem 12.5.

12.4 Show that an irreducible polynomial over $GF(2)$ of degree $\geqslant 2$ has an odd number of non-zero coefficients.

12.5 To verify that a polynomial $p(x)$ is irreducible, why is it enough to show that $p(x)$ has no irreducible factor of degree $\leqslant \frac{1}{2}\deg p(x)$?

12.6 List the irreducible polynomials over $GF(2)$ of degrees 1 to 4. Construct a finite field of order 8.

12.7 Suppose p is a prime number.

(i) Factorize $x^p - 1$ into irreducible polynomials over $GF(p)$.

(ii) Factorize $x^{p-1} - 1$ into irreducible polynomials over $GF(p)$.

12.8 Factorize $x^5 - 1$ into irreducible polynomials over $GF(2)$ and hence determine all the cyclic binary codes of length 5.

12.9 Let $g(x)$ be the generator polynomial of a binary cyclic code which contains some codewords of odd weight. Is the set of codewords in $\langle g(x)\rangle$ of even weight a cyclic code? If so, what is the generator polynomial of this subcode?

12.10 Suppose $x^n - 1$ is the product of t distinct irreducible polynomials over $GF(q)$. How many cyclic codes of length n over $GF(q)$ are there?

12.11 Given that the factorization of $x^7 - 1$ into irreducible polynomials over $GF(2)$ is $(x-1)(x^3+x+1)(x^3+x^2+1)$, determine all the cyclic binary codes of length 7. Give a name or a concise description of each of these codes.

12.12 Factorize $x^8 - 1$ over $GF(3)$. How many ternary cyclic codes of length 8 are there?

12.13 Write down a check polynomial and a parity-check matrix for each of the ternary cyclic codes of length 4 (see Example 12.13).

12.14 Let $h(x)$ be the check polynomial of a cyclic code C. Is $\langle h(x)\rangle$ equal to $C^\perp$? Is $\langle h(x)\rangle$ equivalent to $C^\perp$?

12.15 Suppose C is a binary cyclic code of odd length. Show that C contains a codeword of odd weight if and only if $\mathbf{1}$ is a codeword of C.

12.16 Suppose a generator matrix G of a linear code C has the property that a cyclic shift of any row of G is also a codeword. Show that C is a cyclic code.

12.17 Show that 2 is a primitive element of $GF(11)$. Deduce that the $[10,8]$- and $[10,6]$-codes over $GF(11)$ of Examples 7.12 and 11.3 respectively are equivalent to cyclic codes.

12.18 Let G_{23} be the cyclic Golay code defined in the text. Prove that any two vectors in G_{23} of even weight have inner product equal to zero. Hence prove that the extended Golay code G_{24}, obtained by adding an overall parity-check to G_{23}, is self-dual.

12.19 Determine which of the irreducible polynomials over $GF(2)$ of degree 4 (found in Exercise 12.6) are primitive. Hence write down a generator polynomial for the binary Hamming code of length 15. Find the check polynomial for this code. Write down the corresponding parity-check matrix (using Theorem 12.15) and check that its columns are precisely the non-zero vectors of $V(4,2)$.

12.20 Let $g(x)$ be the generator polynomial of a cyclic binary Hamming code $\mathrm{Ham}(r,2)$, with $r \geqslant 3$. Show that $\langle (x-1)g(x)\rangle$ is a cyclic $[2^r-1, 2^r-r-2, 4]$-code.

12.21 An error vector of the form $x^i + x^{i+1}$ in R_n is called a *double-adjacent error*. Show that the code $\langle (x-1)g(x)\rangle$ of Exercise 12.20 is capable of correcting all single errors and all double-adjacent errors.

12.22 Let C be a $[q+1, 2, q]$-code over $GF(q)$, where q is odd. Show that C cannot be cyclic. Deduce that the Hamming code $\mathrm{Ham}(2,q)$ is not equivalent to a cyclic code when q is odd.

13 Weight enumerators

If C is a linear $[n, k]$-code, its *weight enumerator* is defined to be the polynomial

$$W_C(z) = \sum_{i=0}^{n} A_i z^i$$

$$= A_0 + A_1 z + \cdots + A_n z^n,$$

where A_i denotes the number of codewords in C of weight i.

Another way of writing $W_C(z)$ is

$$W_C(z) = \sum_{\mathbf{x} \in C} z^{w(\mathbf{x})}.$$

Examples 13.1 (i) Let C be the binary even-weight code of length 3; i.e. $C = \{000, 011, 101, 110\}$. Its dual code $C^\perp$ is $\{000, 111\}$. The weight enumerators of C and $C^\perp$ are

$$W_C(z) = 1 + 3z^2$$

$$W_{C^\perp}(z) = 1 + z^3.$$

(ii) The code $C = \{00, 11\}$ is self-dual and so

$$W_C(z) = W_{C^\perp}(z) = 1 + z^2.$$

We have already seen (Theorem 6.14) that knowledge of the weight enumerator of a code enables us to calculate the probability of undetected errors when the code is used purely for error detection.

The main result of this chapter is a remarkable formula of MacWilliams (1963), which enables the weight enumerator of any linear code C to be obtained from the weight enumerator of its dual code $C^\perp$.

For simplicity we shall prove this result, known as the MacWilliams identity, only for binary codes (Theorem 13.5), although the general result will be stated afterwards (Theorem 13.6).

The following three lemmas are required only for the proof of

the MacWilliams identity. The less mathematically minded reader, who is happy to accept the validity of the formula without proof, may skip these lemmas, and also the proof of Theorem 13.5, without any great loss; the subsequent examples and exercises make use only of the formula and not of its proof.

Lemma 13.2 Let C be a binary linear $[n, k]$-code and suppose $\mathbf{y}$ is a fixed vector in $V(n, 2)$ which is not in $C^\perp$. Then $\mathbf{x} \cdot \mathbf{y}$ is equal to 0 and 1 equally often as $\mathbf{x}$ runs over the codewords of C.

Proof Let $A = \{\mathbf{x} \in C \mid \mathbf{x} \cdot \mathbf{y} = 0\}$
and $B = \{\mathbf{x} \in C \mid \mathbf{x} \cdot \mathbf{y} = 1\}$.

Let $\mathbf{u}$ be a codeword of C such that $\mathbf{u} \cdot \mathbf{y} = 1$ ($\mathbf{u}$ exists since $\mathbf{y} \notin C^\perp$). Let $\mathbf{u} + A$ denote the set $\{\mathbf{u} + \mathbf{x} \mid \mathbf{x} \in A\}$. Then

$$\mathbf{u} + A \subseteq B,$$

for if $\mathbf{x} \in A$, then $(\mathbf{u} + \mathbf{x}) \cdot \mathbf{y} = \mathbf{u} \cdot \mathbf{y} + \mathbf{x} \cdot \mathbf{y} = 1 + 0 = 1$.

Similarly

$$\mathbf{u} + B \subseteq A.$$

Hence

$$|A| = |\mathbf{u} + A| \leq |B| = |\mathbf{u} + B| \leq |A|.$$

Hence $|A| = |B|$ and the lemma is proved.

Lemma 13.3 Let C be a binary $[n, k]$-code and let $\mathbf{y}$ be any element of $V(n, 2)$. Then

$$\sum_{\mathbf{x} \in C} (-1)^{\mathbf{x} \cdot \mathbf{y}} = \begin{cases} 2^k & \text{if } \mathbf{y} \in C^\perp \\ 0 & \text{if } \mathbf{y} \notin C^\perp \end{cases}.$$

Proof If $\mathbf{y} \in C^\perp$, then $\mathbf{x} \cdot \mathbf{y} = 0$ for all $\mathbf{x} \in C$, and so

$$\sum_{\mathbf{x} \in C} (-1)^{\mathbf{x} \cdot \mathbf{y}} = |C| \cdot 1 = 2^k.$$

If $\mathbf{y} \notin C^\perp$, then by Lemma 13.2, as $\mathbf{x}$ runs over the elements of C, $(-1)^{\mathbf{x} \cdot \mathbf{y}}$ is equal to 1 and -1 equally often, giving

$$\sum_{\mathbf{x} \in C} (-1)^{\mathbf{x} \cdot \mathbf{y}} = 0.$$

Lemma 13.4 Let $\mathbf{x}$ be a fixed vector in $V(n, 2)$ and let z be an indeterminate. Then the following polynomial identity holds:

$$\sum_{\mathbf{y} \in V(n,2)} z^{w(\mathbf{y})} (-1)^{\mathbf{x} \cdot \mathbf{y}} = (1 - z)^{w(\mathbf{x})} (1 + z)^{n - w(\mathbf{x})}.$$

Proof

$$\begin{aligned} \sum_{\mathbf{y} \in V(n,2)} z^{w(\mathbf{y})} (-1)^{\mathbf{x} \cdot \mathbf{y}} &= \sum_{y_1=0}^{1} \sum_{y_2=0}^{1} \cdots \sum_{y_n=0}^{1} z^{y_1 + y_2 + \cdots + y_n} (-1)^{x_1 y_1 + \cdots + x_n y_n} \\ &= \sum_{y_1=0}^{1} \cdots \sum_{y_n=0}^{1} \left(\prod_{i=1}^{n} z^{y_i} (-1)^{x_i y_i} \right) \\ &= \prod_{i=1}^{n} \left(\sum_{j=0}^{1} z^j (-1)^{j x_i} \right) \\ &= (1 - z)^{w(\mathbf{x})} (1 + z)^{n - w(\mathbf{x})}, \end{aligned}$$

since $$\sum_{j=0}^{1} z^j (-1)^{j x_i} = \begin{cases} 1 + z & \text{if } x_i = 0 \\ 1 - z & \text{if } x_i = 1 \end{cases}.$$

Theorem 13.5 (*The MacWilliams identity for binary linear codes*) If C is a binary $[n, k]$-code with dual code $C^\perp$, then

$$W_{C^\perp}(z) = \frac{1}{2^k} (1 + z)^n W_C\left(\frac{1 - z}{1 + z} \right).$$

Proof We express the polynomial

$$f(z) = \sum_{\mathbf{x} \in C} \left(\sum_{\mathbf{y} \in V(n,2)} (-1)^{\mathbf{x} \cdot \mathbf{y}} z^{w(\mathbf{y})} \right)$$

in two ways.

On the one hand, using Lemma 13.4,

$$\begin{aligned} f(z) &= \sum_{\mathbf{x} \in C} (1 - z)^{w(\mathbf{x})} (1 + z)^{n - w(\mathbf{x})} \\ &= (1 + z)^n \sum_{\mathbf{x} \in C} \left(\frac{1 - z}{1 + z} \right)^{w(\mathbf{x})} \\ &= (1 + z)^n W_C\left(\frac{1 - z}{1 + z} \right). \end{aligned}$$

On the other hand, reversing the order of summation, we have

$$f(z) = \sum_{\mathbf{y}\in V(n,2)} z^{w(\mathbf{y})}\Big(\sum_{\mathbf{x}\in C}(-1)^{\mathbf{x}\cdot\mathbf{y}}\Big)$$
$$= \sum_{\mathbf{y}\in C^\perp} z^{w(\mathbf{y})}2^k \quad \text{(by Lemma 13.3)}$$
$$= 2^k W_{C^\perp}(z).$$

Equating the two expressions for $f(z)$ establishes the result.

The proof of the following more general result is similar to that of Theorem 13.5, using generalized versions of the preceding lemmas, but we omit the details.

Theorem 13.6 (*The MacWilliams identity for general linear codes*) If C is a linear $[n, k]$-code over $GF(q)$ with dual code $C^\perp$, then

$$W_{C^\perp}(z) = \frac{1}{q^k}[1+(q-1)z]^n W_C\Big(\frac{1-z}{1+(q-1)z}\Big).$$

Remark If C is a binary $[n, k]$-code, then, since the dual code of $C^\perp$ is just C, we can write the MacWilliams identity in the (often more useful) form:

$$W_C(z) = \frac{1}{2^{n-k}}(1+z)^n W_{C^\perp}\Big(\frac{1-z}{1+z}\Big). \tag{13.7}$$

Examples 13.8 We apply Theorem 13.5 to the codes of Examples 13.1.

(i) We have $W_C(z) = 1+3z^2$. Hence, by Theorem 13.5,

$$W_{C^\perp}(z) = \tfrac{1}{4}(1+z)^3 W_C\Big(\frac{1-z}{1+z}\Big)$$
$$= \tfrac{1}{4}[(1+z)^3 + 3(1-z)^2(1+z)]$$
$$= 1+z^3,$$

as already found directly from $C^\perp$.

Let us interchange the roles of C and $C^\perp$ in order to check the

formula (13.7). We have

$$\tfrac{1}{2}(1+z)^3 W_{C^\perp}\Big(\frac{1-z}{1+z}\Big) = \tfrac{1}{2}[(1+z)^3 + (1-z)^3]$$
$$= 1+3z^2,$$

which is indeed $W_C(z)$.

(ii) We have $W_C(z) = 1+z^2$. Hence

$$W_{C^\perp}(z) = \tfrac{1}{2}(1+z)^2 W_C\Big(\frac{1-z}{1+z}\Big)$$
$$= \tfrac{1}{2}[(1+z)^2 + (1-z)^2]$$
$$= 1+z^2.$$

Thus $W_{C^\perp}(z) = W_C(z)$, as we expect, since C is self-dual.

For the very small codes just considered, the use of the MacWilliams identity is an inefficient way of calculating their weight enumerators, which can be written down directly from the lists of codewords. But suppose we are required to calculate the weight enumerator of an $[n, k]$-code C over $GF(q)$ where k is large. To enumerate all q^k codewords by weight may be a formidable task. However, if k is so large that $n-k$ is small, then the dual code $C^\perp$ may be small enough to find *its* weight enumerator, and then the MacWilliams identity can be used to find the weight enumerator of C.

For example, the binary Hamming code Ham $(r, 2)$ has dimension $2^r - 1 - r$, and so the number of codewords in Ham $(r, 2)$ is 2^{2^r-1-r}, a large number even for moderately small values of r. But the dual code has only 2^r codewords and, as we shall soon see, it has a particularly simple weight enumerator. From this, the weight enumerator of Ham $(r, 2)$ itself is easily determined. First we look at a particular case.

Example 13.9 Let C be the binary $[7, 4]$-Hamming code. Then the dual code $C^\perp$ has generator matrix

$$\begin{bmatrix} 0&0&0&1&1&1&1 \\ 0&1&1&0&0&1&1 \\ 1&0&1&0&1&0&1 \end{bmatrix}.$$

When we compute $W_{C^\perp}(z)$ directly, by listing the codewords, we find, surprisingly, that each of the non-zero codewords has weight 4 (the next theorem shows this to be no isolated phenomenon, as far as the Hamming codes are concerned). Thus

$$W_{C^\perp}(z) = 1 + 7z^4,$$

and so the weight enumerator of C itself is, by equation (13.7),

$$\tfrac{1}{8}[(1+z)^7 + 7(1-z)^4(1+z)^3] = 1 + 7z^3 + 7z^4 + z^7.$$

Theorem 13.10 Let C be the binary Hamming code Ham $(r, 2)$. Then every non-zero codeword of $C^\perp$ has weight 2^{r-1}.

Proof Let

$$H = \begin{bmatrix} \mathbf{h}_1 \\ \mathbf{h}_2 \\ \vdots \\ \mathbf{h}_r \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1n} \\ h_{21} & h_{22} & \cdots & h_{2n} \\ \vdots & \vdots & & \vdots \\ h_{r1} & h_{r2} & \cdots & h_{rn} \end{bmatrix}$$

be a parity-check matrix of C where the rows of H are denoted by $\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_r$. Then a non-zero codeword $\mathbf{c}$ of $C^\perp$ is a vector of the form $\mathbf{c} = \sum_{i=1}^r \lambda_i \mathbf{h}_i$ for some scalars $\lambda_1, \lambda_2, \ldots, \lambda_r$, not all zero. We will find the weight of $\mathbf{c}$ by finding the number $n_0(\mathbf{c})$ of *zero* entries of $\mathbf{c}$ and then subtracting $n_0(\mathbf{c})$ from the length n. Now $\mathbf{c}$ has a zero in its jth position if and only if $\sum_{i=1}^r \lambda_i h_{ij} = 0$, i.e. if and only if $\sum_{i=1}^r \lambda_i x_i = 0$, where $(x_1 x_2 \cdots x_r)^T$ is the jth column of H. Since C is a Hamming code, the columns of H are precisely the non-zero vectors of $V(r, 2)$ and so $n_0(\mathbf{c})$ is equal to the number of *non-zero* vectors in the set

$$X = \left\{ x_1 x_2 \cdots x_r \in V(r,2) \;\middle|\; \sum_{i=1}^r \lambda_i x_i = 0 \right\},$$

i.e. $n_0(\mathbf{c}) = |X| - 1$.

It is easy to see that X is an $(r-1)$-dimensional subspace of $V(r, 2)$ (e.g. view X as the dual code of the $[r, 1]$-code which has generator matrix $[\lambda_1 \lambda_2 \cdots \lambda_r]$, so that $\dim(X) = r - 1$, by Theorem 7.3). Hence

$$|X| = 2^{r-1} \quad \text{and so} \quad n_0(\mathbf{c}) = 2^{r-1} - 1.$$

(Note that $n_0(\mathbf{c})$ is independent of the choice of non-zero codeword $\mathbf{c}$ in $C^\perp$). Thus

$$\begin{aligned} w(\mathbf{c}) &= n - n_0(\mathbf{c}) = 2^r - 1 - (2^{r-1} - 1) \\ &= 2^{r-1}. \end{aligned}$$

Corollary 13.11 The weight enumerator of the binary Hamming code Ham $(r, 2)$, of length $n = 2^r - 1$, is

$$\frac{1}{2^r}[(1+z)^n + n(1-z^2)^{(n-1)/2}(1-z)].$$

Proof This is a straightforward application of the MacWilliams identity which is left to Exercise 13.5.

Probability of undetected errors

Suppose we wish to find $P_{\text{undetec}}(C)$ for a binary $[n, k]$-code C. By Theorem 6.14, we have

$$\begin{aligned} P_{\text{undetec}}(C) &= \sum_{i=1}^n A_i p^i (1-p)^{n-i} \\ &= (1-p)^n \sum_{i=1}^n A_i \left(\frac{p}{1-p}\right)^i. \end{aligned}$$

Since

$$W_C\left(\frac{p}{1-p}\right) = \sum_{i=0}^n A_i \left(\frac{p}{1-p}\right)^i,$$

and since $A_0 = 1$, we have

$$P_{\text{undetec}}(C) = (1-p)^n \left[W_C\left(\frac{p}{1-p}\right) - 1 \right]. \tag{13.12}$$

If we know $W_C(z)$, then we can find $P_{\text{undetec}}(C)$ by means of equation (13.12). If we know only $W_{C^\perp}(z)$ to start with, then we could use the MacWilliams identity (13.7) to calculate $W_C(z)$ and then use equation (13.12). Alternatively, we could use the formula derived in Exercise 13.9, which gives $P_{\text{undetec}}(C)$ directly in terms of $W_{C^\perp}(z)$, and thereby avoid the intermediate calculation of $W_C(z)$.

Exercises 13

13.1 Suppose C is a binary linear code of length n which contains the vector $11\cdots1$ consisting of all 1s. Show that

$$A_i = A_{n-i},$$

for $i = 0, 1, \ldots, n$.

13.2 Find the weight enumerator of the code whose generator matrix is

$$\begin{bmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

(a) directly,
(b) by using the MacWilliams identity.

13.3 Let C be the binary $[9, 7]$-code having the generator matrix

$$\left[\begin{array}{c|c} & 01 \\ & 01 \\ & 10 \\ I_7 & 10 \\ & 11 \\ & 11 \\ & 11 \end{array}\right].$$

Let $\sum_{i=0}^{9} A_i z^i$ denote the weight enumerator of C. Use the MacWilliams identity to find the values of A_0, A_1, A_2 and A_3. Show that C contains the vector consisting of all 1s and hence, or otherwise, determine the full weight enumerator of C.

13.4 Using the result of Example 13.9, write down the weight enumerator of the extended binary Hamming code of length 8.

13.5 Prove Corollary 13.11.

13.6 Find the number of codewords of each of the weights 0, 1, 2, 3 and 4 in the binary Hamming code of length 15.

13.7 Let C be a binary linear code and let C_0 denote the subcode of C consisting of all codewords of C of even weight. Show that

$$W_{C_0}(z) = \tfrac{1}{2}[W_C(z) + W_C(-z)].$$

13.8 Let C be a binary linear code and let $\hat{C}$ be the extended code obtained from C by adding an overall parity check. Show that

$$W_{\hat{C}}(z) = \tfrac{1}{2}[(1+z)W_C(z) + (1-z)W_C(-z)].$$

13.9 Suppose C is a binary $[n, k]$-code. Prove that

$$P_{\text{undetec}}(C) = \frac{1}{2^{n-k}} W_{C^\perp}(1-2p) - (1-p)^n.$$

13.10 Let G_{24} be the extended binary Golay code defined in Theorem 9.3. Notice that the vector consisting of all 1s belongs to G_{24} (add all the rows of G together). Using properties of G_{24} found during the proof of Theorem 9.3, show that

$$W_{G_{24}}(z) = 1 + 759z^8 + 2576z^{12} + 759z^{16} + z^{24}.$$

13.11 Let G_{23} be the cyclic binary code defined in Theorem 12.20, and let G_{24} be its extended code. Using results from Chapter 12, including Exercise 12.18, determine the weight enumerator of G_{24}.

13.12 Use either Exercise 13.10 or 13.11, together with Exercise 9.4(a), to determine the weight enumerator of the binary Golay code G_{23}.

14 The main linear coding theory problem

In Chapter 2 we discussed the 'main coding theory problem'. This was the problem of finding $A_q(n, d)$, the largest value of M for which there exists a q-ary (n, M, d)-code. In the present chapter we shall consider the same problem restricted to linear codes. If q is a prime power, we denote by $B_q(n, d)$ the largest value of M for which there exists a *linear* (n, M, d)-code over $GF(q)$. (The function $B_q(n, d)$ was briefly introduced in Exercises 5.8 and 5.9). Clearly $B_q(n, d)$ is always a power of q, and $B_q(n, d) \leq A_q(n, d)$. We shall refer to the problem of finding $B_q(n, d)$ as the *main linear coding theory problem*, or MLCT problem for short.

If we regard the values of q and d as fixed, we may state the problem as follows.

MLCT problem (Version 1) For given length n, find the maximum dimension k such that there exists an $[n, k, d]$-code over $GF(q)$. (Then, for this k, $B_q(n, d) = q^k$).

Recall that the *redundancy* r of an $[n, k, d]$-code is just $n - k$ (the number of check symbols in a codeword). An alternative version of the MLCT problem is:

MLCT problem (Version 2) For given redundancy r, find the maximum length n such that there exists an $[n, n-r, d]$-code over $GF(q)$.

Solving Version 1 for all n is equivalent to solving Version 2 for all r, because in either case we then know exactly those values of n and k for which an $[n, k, d]$-code exists. The equivalence of the two versions will be made explicit in Theorem 14.3.

It turns out that Version 2 provides the more natural approach. The key to this approach, which was touched upon in

Concluding Remark 3 of Chapter 8, is given in the next theorem. But first we make some definitions.

Definitions An (n, s)-*set* in $V(r, q)$ is a set of n vectors in $V(r, q)$ with the property that any s of them are linearly independent.

We denote by $\max_s(r, q)$ the largest value of n for which there exists an (n, s)-set in $V(r, q)$. An (n, s)-set in $V(r, q)$ which has $n = \max_s(r, q)$ is called *optimal*. The *packing problem* for $V(r, q)$ is that of determining the values of $\max_s(r, q)$ and the optimal (n, s)-sets.

The packing problem was first considered by Bose (1947) for its statistical interest and later (1961) for its connection with coding theory, which is given by the following theorem.

Theorem 14.1 There exists an $[n, n-r, d]$-code over $GF(q)$ if and only if there exists an $(n, d-1)$-set in $V(r, q)$.

Proof Suppose C is an $[n, n-r, d]$-code over $GF(q)$ with parity-check matrix H. Then, by Theorem 8.4, the columns of H form an $(n, d-1)$-set in $V(r, q)$. On the other hand, suppose K is an $(n, d-1)$-set in $V(r, q)$. If we form an $r \times n$ matrix H with the vectors of K as its columns, then, again by Theorem 8.4, H is the parity-check matrix of an $[n, n-r]$-code whose minimum distance is at least d.

Corollary 14.2 For given values of q, d and r, the largest value of n for which there exists an $[n, n-r, d]$-code over $GF(q)$ is $\max_{d-1}(r, q)$.

So the MLCT problem (Version 2) is the same as the packing problem of finding $\max_{d-1}(r, q)$. We now show that the values of $B_q(n, d)$ are also given by the solutions to this problem.

Theorem 14.3 Suppose $\max_{d-1}(r-1, q) < n \leqslant \max_{d-1}(r, q)$. Then $B_q(n, d) = q^{n-r}$.

Proof Since $n \leqslant \max_{d-1}(r, q)$, there exists an $[n, n-r, d]$-code over $GF(q)$, and so $B_q(n, d) \geqslant q^{n-r}$. If $B_q(n, d)$ were strictly greater than q^{n-r}, then there would exist an $[n, n-r+1, d]$-code, implying that $n \leqslant \max_{d-1}(r-1, q)$, contrary to hypothesis.

Let us pause to outline our plan of campaign for the remainder of this and the next chapter. We shall consider the MLCT problem for increasing values of the minimum distance d. The cases $d = 1$ and $d = 2$ are easily dealt with in Exercise 14.2. We will therefore consider first the problem for $d = 3$ and will solve it for all values of q and r. We will then consider the case $d = 4$, solving the MLCT problem for $q = 2$ and giving the known results for $q > 2$. For cases of d greater than 4, very little is known in the way of general results, at least not until d reaches its maximum value for given redundancy r, which is $d = r + 1$. We will consider this very interesting case in Chapter 15.

The MLCT problem for $d = 3$ (or Hamming codes revisited)

Theorem 14.4 For given redundancy r, the maximum length n of an $[n, n-r, 3]$-code over $GF(q)$ is $(q^r - 1)/(q-1)$; i.e. $\max_2(r, q) = (q^r - 1)/(q-1)$.

Proof By Corollary 14.2, the required value of n is $\max_2(r, q)$, the largest size of an $(n, 2)$-set in $V(r, q)$. Now a set S of vectors in $V(r, q)$ is an $(n, 2)$-set if and only if no vector in S is a scalar multiple of any other vector in S. As we saw in the construction of q-ary Hamming codes in Chapter 8, the $q^r - 1$ non-zero vectors of $V(r, q)$ are partitioned into $(q^r - 1)/(q-1)$ classes, each class consisting of $q - 1$ vectors which are scalar multiples of each other. Thus an $(n, 2)$-set of largest size is just a set of $(q^r - 1)/(q-1)$ vectors, one from each of these classes.

The optimal $[n, n-r, 3]$-codes with $n = (q^r - 1)/(q-1)$ are just the Hamming codes $\mathrm{Ham}(r, q)$ defined in Chapter 8. The solution to MLCT problem (Version 1) follows immediately from Theorems 14.3 and 14.4:

Theorem 14.5 $B_q(n, 3) = q^{n-r}$, where r is the unique integer such that $(q^{r-1} - 1)/(q-1) < n \leqslant (q^r - 1)/(q-1)$.

Remarks (1) It is easy to express $B_q(n, 3)$ as an explicit function of q and n (see Exercise 14.3).

(2) To construct a linear $(n, M, 3)$-code with $M = B_q(n, 3)$, one simply finds the least integer r such that $n \leqslant (q^r - 1)/(q-1)$ and writes down, as a parity-check matrix, n column vectors of

$V(r, q)$ such that no column is a scalar multiple of another. Such a parity-check matrix can always be obtained by deleting columns from the parity-check matrix of a Hamming code Ham (r, q). Thus the best linear single-error-correcting codes of given length are either Hamming or shortened Hamming codes.

Before proceeding to the case $d = 4$, we remark that it will be advantageous to view an (n, s)-set not only as a set of vectors in the vector space $V(r, q)$, but also as a set of points in the associated projective geometry $PG(r-1, q)$, which we now define.

The projective geometry $PG(r-1, q)$

With the vector space $V(r, q) = \{(a_1, a_2, \ldots, a_r) \mid a_i \in GF(q)\}$, we associate a combinatorial structure $PG(r-1, q)$ consisting of points and lines defined as follows.

The *points* of $PG(r-1, q)$ are the one-dimensional subspaces of $V(r, q)$. The *lines* of $PG(r-1, q)$ are the two-dimensional subspaces of $V(r, q)$. The point P is said to belong to (or lie on) the line L if and only if P is a subspace of L. $PG(r-1, q)$ is called the *projective geometry of dimension* $r-1$ *over* $GF(q)$.

Each point P of $PG(r-1, q)$, as a subspace of $V(r, q)$ of dimension 1, is generated by a single non-zero vector. So, if $\mathbf{a} = (a_1, a_2, \ldots, a_r) \in P$, then

$$P = \{\lambda \mathbf{a} \mid \lambda \in GF(q)\}.$$

In practice, we identify the point P with any non-zero vector it contains. In other words, we take the points of $PG(r-1, q)$ to be the non-zero vectors of $V(r, q)$ with the rule that if $\mathbf{a} = (a_1, a_2, \ldots, a_r)$ and $\mathbf{b} = (b_1, b_2, \ldots, b_r)$ are two such vectors, then

$\mathbf{a} = \mathbf{b}$ in $PG(r-1, q)$ if and only if $\mathbf{a} = \lambda \mathbf{b}$ in $V(r, q)$,

for some non-zero scalar λ.

We now list some elementary properties of $PG(r-1, q)$.

Lemma 14.6 In $PG(r-1, q)$,

(i) the number of points is $(q^r - 1)/(q-1)$,
(ii) any two points lie on exactly one line,
(iii) each line contains exactly $q+1$ points,
(iv) each point lies on $(q^{r-1} - 1)/(q-1)$ lines.

Proof (i) Since each of the $q^r - 1$ non-zero vectors in $V(r, q)$ has $q-1$ non-zero scalar multiples, the number of points of $PG(r-1, q)$ is $(q^r - 1)/(q-1)$.

(ii) If $\mathbf{a}$ and $\mathbf{b}$ are distinct points of $PG(r-1, q)$, then the unique line through them consists of the points $\lambda \mathbf{a} + \mu \mathbf{b}$, where λ and μ are scalars not both zero.

(iii) In (ii), there are $q^2 - 1$ choices for the pair (λ, μ), but since we are identifying scalar multiples, the number of distinct points on the line is $(q^2 - 1)/(q-1) = q + 1$.

(iv) Let t be the number of lines on which a given point P lies. Let X denote the set $\{(Q, L) \mid Q$ is a point $\neq P$, L is a line containing both P and $Q\}$. We count the members of X in two ways. For each of the $(q^r - 1)/(q-1) - 1$ choices for Q, there is a unique line L containing P and Q. Thus

$$|X| = (q^r - 1)/(q-1) - 1 = (q^r - q)/(q-1).$$

On the other hand for each of the t lines through P, there are, by part (iii), q points Q other than P lying on L. Thus

$$|X| = tq.$$

Equating the two expressions for $|X|$ gives $t = (q^{r-1} - 1)/(q-1)$.

Definition The projective geometry $PG(2, q)$ is called the *projective plane over* $GF(q)$. It follows from Lemma 14.6 that $PG(2, q)$ is a symmetric $(q^2 + q + 1, q + 1, 1)$-design, so that it is a projective plane as defined in Chapter 2.

Examples 14.7 (i) The simplest projective plane is $PG(2, 2)$. This contains 7 points labelled 001, 010, 100, 011, 101, 110, 111,

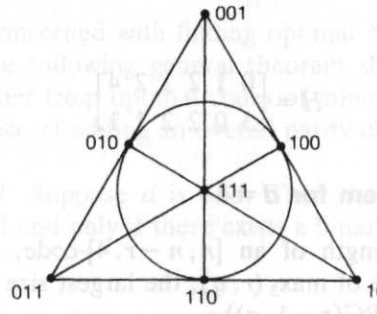


Fig. 14.8. The projective plane $PG(2, 2)$.

and 7 lines as shown in Fig. 14.8. This shows that $PG(2, 2)$ is the same as the 7-point plane of Example 2.19.

(ii) The 6 points of $PG(1, 5)$ are 01, 10, 11, 12, 13 and 14, and there is just one line consisting of all 6 points. The points could equally well be labelled 03, 10, 22, 12, 21, and 41, say, because in $PG(1, 5)$, $01 = 03$, $11 = 22$, $13 = 21$ and $14 = 41$.

Remarks (1) The points of $PG(r-1, q)$ can be uniquely labelled by making the left-most non-zero coordinate equal to 1.

(2) If $q = 2$, the points of $PG(r-1, 2)$ are labelled by the non-zero vectors of $V(r, 2)$.

Definition A set K of n points in $PG(r-1, q)$ is called an (n, s)-*set* if the vectors representing the points of K form an (n, s)-set in the underlying vector space $V(r, q)$.

Remarks (1) Two advantages of working in $PG(r-1, q)$ are that (a) some neat counting arguments may then be used to obtain upper bounds on $\max_s(r, q)$ and (b) many optimal (n, s)-sets turn out to be natural geometric configurations.

(2) An $(n, 2)$-set in $PG(r-1, q)$ is just a set of n distinct points of $PG(r-1, q)$. So we may describe a Hamming code $\mathrm{Ham}(r, q)$ as a code having a parity-check matrix H whose columns are the distinct points of $PG(r-1, q)$. Of course, different representations of these points as vectors will give rise to different, but equivalent, codes. For example (cf. Example 14.7(ii)), $\mathrm{Ham}(1, 5)$ may be defined to have parity-check matrix

$$H = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 2 & 3 & 4 \end{bmatrix}$$

or, equally well,

$$H = \begin{bmatrix} 0 & 1 & 2 & 1 & 2 & 4 \\ 3 & 0 & 2 & 2 & 1 & 1 \end{bmatrix}.$$

The MLCT problem for $d = 4$

The maximum length of an $[n, n-r, 4]$-code, for given r, is equal to the value of $\max_3(r, q)$, the largest size of an $(n, 3)$-set in $V(r, q)$ (or in $PG(r-1, q)$).

An $(n, 3)$-set in the plane $PG(2, q)$ is usually called an n-*arc*, while an $(n, 3)$-set in $PG(r-1, q)$, for $r > 3$, is called an n-*cap*.

Since three points of $PG(r-1, q)$ are linearly dependent if and only if they are collinear (i.e. they lie on the same line), we may describe an n-arc/n-cap as a set of n points, no three of which are collinear.

The problem of determining the values of $\max_3(r, q)$, first considered by Bose (1947), was quickly solved for $q = 2$, for all r, and for $r \leqslant 4$, for all q. But, despite having received much attention since, the problem has been solved only for the additional pairs $(r, q) = (4, 3)$ and $(5, 3)$. The known values of $\max_3(r, q)$ are listed in Fig. 14.9.

$\max_3(r, 2) = 2^{r-1}$		(Bose 1947)
$\max_3(3, q) = \begin{cases} q+1, \\ q+2, \end{cases}$	q odd q even	(Bose 1947)
$\max_3(4, q) = \begin{cases} q^2+1, \\ q^2+1, \end{cases}$	q odd q even	(Bose 1947) (Qvist 1952)
$\max_3(5, 3) = 20$		(Pellegrino 1970)
$\max_3(6, 3) = 56$		(Hill 1973)

Fig. 14.9. The known values of $\max_3(r, q)$.

We now prove the more straightforward of these results.

The determination of $\max_3(r, 2)$

Here we are concerned with finding optimal *binary* linear codes with $d = 4$. The following general theorem shows that we may obtain such codes from optimal codes of minimum distance 3 by the simple device of adding an overall parity-check.

Theorem 14.10 Suppose d is odd. Then there exists a binary $[n, k, d]$-code if and only if there exists a binary $[n+1, k, d+1]$-code.

Proof The proof of Theorem 2.7 is valid in the restriction to

linear codes. This is because an 'extended' linear code (i.e. the code obtained from a linear code by adding an overall parity-check) is also linear (see Exercise 5.4).

Corollary 14.11 Suppose d is even. Then

(i) $B_2(n, d) = B_2(n-1, d-1)$

(ii) $\max_{d-1}(r, 2) = \max_{d-2}(r-1, 2) + 1$.

Proof

(i) is immediate from Theorem 14.10.

(ii) $n \leqslant \max_{d-1}(r, 2) \Leftrightarrow$ there exists a binary $[n, n-r, d]$-code

$\Leftrightarrow$ there exists a binary $[n-1, n-r, d-1]$-code

$\Leftrightarrow n-1 \leqslant \max_{d-2}(r-1, 2)$

$\Leftrightarrow n \leqslant \max_{d-2}(r-1, 2) + 1$.

Corollary 14.12 $\max_3(r, 2) = 2^{r-1}$.

Proof By Theorem 14.4, $\max_2(r, 2) = 2^r - 1$. Hence

$$\max_3(r, 2) = (2^{r-1} - 1) + 1 = 2^{r-1}.$$

The optimal binary code with $d = 4$ and redundancy r is the extended Hamming code Hâm $(r-1, 2)$. As we saw in Chapter 8, a parity-check matrix for this code is

$$\bar{H} = \begin{bmatrix} & & 0 \\ & H & \vdots \\ & & 0 \\ 1 & 1 \cdots & 1 \end{bmatrix},$$

where H is a parity-check matrix for Ham $(r-1, 2)$, so that the columns of H are just the points of $PG(r-2, 2)$ (i.e. the non-zero vectors of $V(r-1, 2)$).

The columns of $\bar{H}$ form an optimal 2^{r-1}-cap in $PG(r-1, 2)$. It consists of the points of $PG(r-1, 2)$ *not* lying in the subspace $\{(x_1, \ldots, x_r) \mid x_r = 0\}$. Geometrically, it may be described as the complement of a hyperplane.

The determination of $\max_3(3, q)$

First we give some examples of good linear codes with $d = 4$ and redundancy 3. We then prove that these codes are optimal by showing that there cannot exist such codes of greater length.

Theorem 14.13 Let $a_1, a_2, \ldots, a_{q-1}$ be the non-zero elements of $GF(q)$.

(i) The matrix $H = \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 & 0 \\ a_1 & a_2 & \cdots & a_{q-1} & 0 & 0 \\ a_1^2 & a_2^2 & \cdots & a_{q-1}^2 & 0 & 1 \end{bmatrix}$

is the parity-check matrix of a $[q+1, q-2, 4]$-code. Equivalently, the columns of H form a $(q+1)$-arc in $PG(2, q)$.

(ii) If q is even, then the matrix

$$H^* = \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 & 0 & 0 \\ a_1 & a_2 & \cdots & a_{q-1} & 0 & 1 & 0 \\ a_1^2 & a_2^2 & \cdots & a_{q-1}^2 & 0 & 0 & 1 \end{bmatrix}$$

is the parity-check matrix of a $[q+2, q-1, 4]$-code. Equivalently, the columns of H^* form a $(q+2)$-arc in $PG(2, q)$.

Proof (i) It is enough to show that any three columns of H are linearly independent. Any three of the first $q-1$ columns of H form a Vandermonde matrix, and so are linearly independent by Theorems 11.1 and 11.2. For any three columns which include one or both of the last two columns, the determinant may be expanded about these last columns to get again the determinant of a Vandermonde matrix.

(ii) We have shown in the proof of part (i) that any three columns of H^* are linearly independent, with the possible exception of three of the form

$$\begin{bmatrix} 1 \\ a_i \\ a_i^2 \end{bmatrix}, \begin{bmatrix} 1 \\ a_j \\ a_j^2 \end{bmatrix}, \text{ and } \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}.$$

The determinant of the matrix A formed by these three columns

is equal to $a_i^2 - a_j^2$. Since q is even, $GF(q)$ has characteristic 2 (cf. Exercise 4.6). Hence, by Exercise 3.12, $a_i^2 - a_j^2 = (a_i - a_j)^2$. Since $a_i \neq a_j$, $\det A$ is non-zero.

Corollary 14.14 $\max_3(3, q) \geqslant \begin{cases} q+1, & \text{if } q \text{ is odd} \\ q+2, & \text{if } q \text{ is even.} \end{cases}$

Remark The $(q+1)$-arc formed by the columns of H in Theorem 14.13 is the *conic* $\{(x, y, z) \in PG(2, q) \mid yz = x^2\}$.

We now show that the codes/arcs given in Theorem 14.13 are optimal.

Theorem 14.15
(i) For any prime power q, $\max_3(3, q) \leqslant q + 2$.
(ii) If q is odd, then $\max_3(3, q) \leqslant q + 1$.

First proof (i) Let H be a standard form parity-check matrix for an $[n, n-3, 4]$-code C over $GF(q)$, with $n = \max_3(3, q)$:

$$H = \begin{bmatrix} a_1 & a_2 & \cdots & a_{n-3} & 1 & 0 & 0 \\ b_1 & b_2 & \cdots & b_{n-3} & 0 & 1 & 0 \\ c_1 & c_2 & \cdots & c_{n-3} & 0 & 0 & 1 \end{bmatrix}.$$

Since any three columns of H are linearly independent, the determinant formed by any three columns must be non-zero. From the non-vanishing of the determinant formed by any two of the last three columns and one of the first $n-3$ columns, we find that the a_is, b_is and c_is are all non-zero. Multiplying the ith column by a_i^{-1} for $i = 1, 2, \ldots, n-3$, we have that C is equivalent to a code in which the a_is are all 1. Thus we may assume that

$$A = \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 & 0 & 0 \\ b_1 & b_2 & \cdots & b_{n-3} & 0 & 1 & 0 \\ c_1 & c_2 & \cdots & c_{n-3} & 0 & 0 & 1 \end{bmatrix},$$

where the b_is and c_is are non-zero. As the determinant formed by the last column and two of the first $n-3$ columns is non-zero, the b_is must be distinct non-zero elements of $GF(q)$. Hence $n - 3 \leqslant q - 1$ and so $n \leqslant q + 2$.

(ii) (Adapted from Fenton and Vámos, 1982). Now suppose q is odd. Suppose, for a contradiction, that a $[q+2, q-1, 4]$-code C exists. Then, as in (i), we may assume that C has a parity-check matrix

$$H = \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 & 0 & 0 \\ b_1 & b_2 & \cdots & b_{q-1} & 0 & 1 & 0 \\ c_1 & c_2 & \cdots & c_{q-1} & 0 & 0 & 1 \end{bmatrix}$$

where $b_1, b_2, \ldots, b_{q-1}$ are the distinct non-zero elements of $GF(q)$ and similarly $c_1, c_2, \ldots, c_{q-1}$ are also the distinct non-zero elements of $GF(q)$ in some order. The non-vanishing of determinants of the form

$$\det \begin{bmatrix} 1 & 1 & 1 \\ b_i & b_j & 0 \\ c_i & c_j & 0 \end{bmatrix}$$

implies that the elements $b_1c_1^{-1}, b_2c_2^{-1}, \ldots, b_{q-1}c_{q-1}^{-1}$ are distinct and so they too are the non-zero elements of $GF(q)$ in some order. Hence, by Exercise 3.13, all three of the products $\prod_{i=1}^{q-1} b_i$, $\prod_{i=1}^{q-1} c_i$, and $\prod_{i=1}^{q-1} (b_ic_i^{-1})$ are equal to -1. But then

$$\prod_{i=1}^{q-1} (b_ic_i^{-1}) = \left(\prod_{i=1}^{q-1} b_i\right)\left(\prod_{i=1}^{q-1} c_i\right)^{-1} = (-1)(-1)^{-1} = 1.$$

Since $1 \neq -1$ if q is odd, this gives the desired contradiction.

Second proof (geometric) (i) Suppose K is an n-arc in $PG(2, q)$ of maximum size $n = \max_3(3, q)$. Let P be a point of K. By Lemma 14.6(iv), there are $q+1$ lines through P and every point of K lies on one or other of them. But on none of these lines can there be more than one point of K besides P (by definition of an n-arc, no three points of K are collinear). Thus $n \leqslant 1 + (q+1) = q + 2$.

(ii) Now suppose q is odd. Suppose, for a contradiction, that K is a $(q+2)$-arc in $PG(2, q)$. Then if P is any point of K, each of the $q+1$ lines through P must contain exactly one further point of K. This means that every line in $PG(2, q)$ meets K in either 2 or 0 points (but never in 1). Now let Q be any point of $PG(2, q)$ lying outside K. Through Q there pass $q+1$ lines and each point of K lies on one (and only one) of them. So if t of

these lines meet K in two points, then $|K| = 2t$, contradicting $|K| = q + 2$ being odd.

Remark The author feels that the attractiveness of the above proofs merits the inclusion of both. The geometric proof has two important advantages: (1) it generalizes to give upper bounds on $\max_3(r, q)$ for larger values of r; (2) it does not assume specific properties of the field $GF(q)$, and so gives the same upper bound on the size of n-arcs in any projective plane of order q.

Corollary 14.14 and Theorem 14.15 give

Theorem 14.16 (Bose 1947)

$$\max_3(3, q) = \begin{cases} q+1 & \text{if } q \text{ is odd} \\ q+2 & \text{if } q \text{ is even.} \end{cases}$$

Remark It has been shown by Segre (1954) that, for q odd, every $(q+1)$-arc in $PG(2, q)$ is a conic. This implies that the optimal $[q+1, q-2, 4]$-code is unique, up to equivalence. For q even, optimal $(q+2)$-arcs in $PG(2, q)$ are not in general unique, and a classification is unknown.

The determination of $\max_3(4, q)$, for q odd

As we shall adopt a geometric approach here, we introduce a little more terminology concerning the projective geometry $PG(r-1, q)$. In defining $PG(r-1, q)$ from the vector space $V(r, q)$, recall that the points and lines in $PG(r-1, q)$ are the 1- and 2-dimensional subspaces respectively of $V(r, q)$. More generally we define a *t-space* in $PG(r-1, q)$ to be a $(t+1)$-dimensional subspace of $V(r, q)$. Thus a 0-space is a point and a 1-space is a line. A 2-space is called a *plane* and an $(r-2)$-space in $PG(r-1, q)$ is called a *hyperplane*. Note that the *dimension* t of a t-space in $PG(r-1, q)$ is always one less than the corresponding vector space dimension.

We usually identify a t-space in $PG(r-1, q)$ with the set of points it contains. The number of points in a t-space is $(q^{t+1}-1)/(q-1)$, since a $(t+1)$-dimensional subspace of $V(r, q)$ contains $q^{t+1}-1$ non-zero vectors, each of which has $q-1$ non-zero scalar multiples. A t-space is just a copy of $PG(t, q)$ in so far as the incidence properties of its subspaces are concerned. In particular, a cap in $PG(r-1, q)$ must meet a $(t-1)$-space in at most $\max_3(t, q)$ points, bearing in mind that any subset of a cap is also a cap.

We may now derive an upper bound on $\max_3(4, q)$, for q odd.

Theorem 14.17 If q is odd, then $\max_3(4, q) \leq q^2 + 1$.

Proof Suppose K is an n-cap in $PG(3, q)$ of maximum size. Let P_1 and P_2 be any two points of K and let L be the line on which P_1 and P_2 lie. Since no three points of K are collinear, L contains no other point of K. Through the line L there pass $q+1$ planes (Exercise 14.4), and each point of K, other than P_1 and P_2, lies on one and only one of these planes. Since q is odd, it follows from Theorem 14.15(ii) that no plane can contain more than $q+1$ points of K. In particular, a plane through L can contain at most $q-1$ points in addition to P_1 and P_2. Hence

$$n \leq 2 + (q+1)(q-1) = q^2 + 1.$$

We next show that (q^2+1)-caps exist in $PG(3, q)$, when q is odd.

Theorem 14.18 Suppose q is odd and let b be a non-square in $GF(q)$. Then the set

$$Q = \{(x, y, z, w) \in PG(3, q) \mid zw = x^2 - by^2\}$$

is a (q^2+1)-cap in $PG(3, q)$.

Proof Since b is a non-square, the only point of Q having $z = 0$ is $(0, 0, 0, 1)$. Each of the remaining points may be represented by a vector having $z = 1$, and so we may write

$$Q = \{(0, 0, 0, 1), (x, y, 1, x^2 - by^2) \mid (x, y) \in V(2, q)\}. \quad (14.19)$$

This shows that $|Q| = q^2 + 1$. We must show that no three points of Q are collinear. Clearly $(0, 0, 0, 1)$ cannot be collinear with two other points of Q because there is only one point of Q of the form $(x, y, 1, *)$ for any given pair (x, y). Now let $\mathbf{a}_1 = (x_1, y_1, 1, x_1^2 - by_1^2)$ and $\mathbf{a}_2 = (x_2, y_2, 1, x_2^2 - by_2^2)$ be any two points of Q, other than $(0, 0, 0, 1)$. Suppose, for a contradiction, that the line

joining $\mathbf{a}_1$ and $\mathbf{a}_2$ contains a third point of Q. Then, for some non-zero scalar λ, $\mathbf{a}_1 + \lambda\mathbf{a}_2 \in Q$, i.e. the point $(x, y, z, w) = (x_1 + \lambda x_2, y_1 + \lambda y_2, 1 + \lambda, x_1^2 - by_1^2 + \lambda x_2^2 - \lambda by_2^2)$ satisfies $zw = x^2 - by^2$. This condition implies, after some cancellation, that

$$\lambda x_1^2 + \lambda x_2^2 - \lambda by_1^2 - \lambda by_2^2 = 2\lambda x_1 x_2 - 2\lambda by_1 y_2.$$

Since $\lambda \neq 0$, it follows that

$$(x_1 - x_2)^2 = b(y_1 - y_2)^2,$$

which is impossible since b is a non-square.

Putting Theorems 14.17 and 14.18 together gives

Theorem 14.20 If q is odd, then $\max_3(4, q) = q^2 + 1$.

Example 14.21 Take $q = 3$ and $b = -1$ in Theorem 14.18. By (14.19), a 10-cap in $PG(3, 3)$ is formed by the columns of the matrix

$$H = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 2 & 2 & 1 & 2 & 2 \end{bmatrix}.$$

Thus H is the parity-check matrix of a ternary $[10, 6, 4]$-code which is of greatest length for $d = 4$ and $r = 4$.

Remarks (1) The set Q of Theorem 14.18 is an example of an *elliptic quadric*. For q odd, any elliptic quadric is a $(q^2 + 1)$-cap, and conversely (Barlotti 1955) any $(q^2 + 1)$-cap is an elliptic quadric. This implies that the optimal $[q^2 + 1, q^2 - 3, 4]$-code is unique, up to equivalence.

(2) For $q = 2^h$, with $h > 1$, it is also true that $\max_3(4, q) = q^2 + 1$, but the proof is a little trickier and is omitted here.

The values of $B_q(n, 4)$, for $n \leq q^2 + 1$

By means of Theorem 14.3, we can instantly translate our results concerning $\max_3(r, q)$ for $r = 2$ and 3 into results about $B_q(n, 4)$.

Theorem 14.22 If q is odd, then

$$B_q(n, 4) = \begin{cases} q^{n-3} & \text{for } 4 \leq n \leq q + 1 \\ q^{n-4} & \text{for } q + 2 \leq n \leq q^2 + 1. \end{cases}$$

If q is even, then

$$B_q(n, 4) = \begin{cases} q^{n-3} & \text{for } 4 \leq n \leq q + 2 \\ q^{n-4} & \text{for } q + 3 \leq n \leq q^2 + 1. \end{cases}$$

Remarks on $\max_3(r, q)$ for $r \geq 5$

For $r = 3$ and $r = 4$ the packing problem for caps in $PG(r - 1, q)$ was fairly easy to solve because of the existence of natural geometric configurations (conics in $PG(2, q)$ and elliptic quadrics in $PG(3, q)$) which are optimal caps. But in $PG(r - 1, q)$ for $r \geq 5$, large caps do not appear to arise in such a natural way and so the packing problem is much more difficult. As we see from Table 14.9, the only known values of $\max_3(r, q)$ for $q \neq 2$ and $r \geq 5$ are $\max_3(5, 3) = 20$ and $\max_3(6, 3) = 56$. (For a coding-theoretic proof of the second result, wherein the uniqueness of the optimal ternary $[56, 50, 4]$-code is also demonstrated, see Hill (1978).)

It is easy to construct 20-caps in $PG(4, 3)$ (Exercise 14.9) but hard to show that 20 is the largest size possible. By contrast, it is rather difficult to describe a 56-cap in $PG(5, 3)$, but a short proof of the maximality of 56 has been given by Bruen and Hirschfeld (1978) (cf. Exercise 14.11). In the next dimension up for $q = 3$, the best known bounds are

$$112 \leq \max_3(7, 3) \leq 163,$$

suggesting that the problem of finding optimal caps in $PG(6, 3)$ is far from solution.

Concluding remarks on Chapter 14

(1) We have mentioned that the problem of determining $\max_s(r, q)$ was first considered by Bose (1947). Much of the subsequent work has been carried out by the Italian school of geometers led by Segre, Barlotti and Tallini.

For a survey of the known results concerning $\max_s(r, q)$ and similar functions, see Hirschfeld (1983). For a comprehensive coverage of the theory of projective geometries over finite fields, see Hirschfeld (1979 and Volume 2, to appear).

(2) For recent results concerning $\max_s(r, q)$ for $q = 3$ and $s \leq r \leq 15$, see Games (1983).

(3) There seems to be little pattern to results concerning $\max_{d-1}(r, q)$ for fixed values of d greater than 4. However, when d takes its maximum value for given r, that is $d = r + 1$, an interesting pattern once again emerges. This case is the subject of the next chapter.

(4) Another version of the MLCT problem is to find, for given q, n and k, the maximum value of d for which there exists an $[n, k, d]$-code over $GF(q)$. In the case of binary linear codes, Helgert and Stinaff (1973) give a table of such values (or bounds when the values are not known) for $k \leq n \leq 127$. For a comprehensive update of this table, incorporating many improvements by various authors, see Verhoeff (1985).

Exercises 14

14.1 Is it true that $B_2(n, d)$ is always equal to the highest power of 2 less than or equal to $A_2(n, d)$?

14.2 Show that (i) $B_q(n, 1) = q^n$, (ii) $B_q(n, 2) = q^{n-1}$.

14.3 Show that $B_q(n, 3) = q^{\lfloor n - \log_q(nq-n+1) \rfloor}$.

14.4 Show that in $PG(3, q)$ the number of planes containing a given line is $q + 1$.

14.5 Which code is the optimal $[n, n-5, 5]$-code having $n = \max_4(5, 3)$?

14.6 Specify a $[26, 22, 4]$-code over $GF(5)$.

14.7 Pinpoint where the proofs of Theorems 14.17 and 14.18 fail when q is even.

14.8 Devise a syndrome-decoding algorithm for a $[q^2 + 1, q^2 - 3, 4]$-code over $GF(q)$ (q odd), which will correct any single error and detect any double error.

14.9 Given the 10-cap of Example 14.21, construct a 20-cap in $PG(4, 3)$.

14.10 Show that, in $PG(m, q)$, the number of $(t+1)$-spaces containing a given t-space is $(q^{m-t} - 1)/(q - 1)$. In $PG(5, 3)$, state (i) how many planes contain a given line, (ii) how many 3-spaces contain a given plane, (iii) how many 4-spaces contain a given 3-space.

14.11 Given that $\max_3(5, 3) = 20$, show that $\max_3(6, 3) \leq 56$. [*Hint:* Use parts (i), (ii) and (iii) of Exercise 14.10.]

14.12 State the values of $B_3(n, 4)$ for $4 \leq n \leq 112$.

15 MDS codes

In the previous chapter we considered the problem of finding linear codes of maximum length for given redundancy r and given minimum distance d. Particular attention was paid to the cases $d \leq 4$. In this chapter we consider the problem when d is as large as possible for given redundancy r. The following theorem shows that this is the case $d = r + 1$.

Theorem 15.1 An $[n, n-r, d]$-code satisfies $d \leq r + 1$.

Proof 1 This is just the Singleton bound applied to linear codes. Theorem 10.17 states that any q-ary (n, M, d)-code satisfies $M \leq q^{n-d+1}$. So, in particular, an $[n, n-r, d]$-code over $GF(q)$ satisfies $q^{n-r} \leq q^{n-d+1}$, whence $d \leq r + 1$.

Proof 2 Suppose C is an $[n, n-r, d]$-code and let $G = [I_{n-r} \mid A]$ be a standard form generator matrix of C. Since A has r columns, those codewords which are rows of G have weight $\leq r + 1$. The result follows by Theorem 5.2.

Definition An $[n, n-r, r+1]$-code (i.e. a linear code of redundancy r whose minimum distance is equal to $r + 1$) is called a *maximum distance separable code*, or *MDS code* for short.

By Theorem 14.1, the maximum length of an $[n, n-r, r+1]$-code over $GF(q)$ is equal to the value of $\max_r(r, q)$, the largest size of an (n, r)-set in $V(r, q)$. We recall that an (n, r)-set in $V(r, q)$ is a set of n vectors such that any r of them are linearly independent. Equivalently, an (n, r)-set in $V(r, q)$ is a set of n vectors such that any r of them form a basis for $V(r, q)$.

MDS codes were first studied explicitly by Singleton (1964), although the problem of finding $\max_r(r, q)$ had already been studied as a problem in statistics (Bush 1952) and as a problem in geometry (Segre 1955, 1961). (In the geometrical context, an

(n, r)-set, regarded as a subset of $PG(r-1, q)$, is called an *n-arc*. This agrees with the usage of the term *n-arc* for an $(n, 3)$-set in $PG(2, q)$ already met in Chapter 14.)

MacWilliams and Sloane (1977) introduce their chapter on MDS codes as 'one of the most fascinating in all of coding theory'. The problem of determining the values of $\max_r(r, q)$ is a particularly attractive one for two reasons. Firstly, the problem is equivalent to a surprising list of combinatorial problems; no fewer than six different interpretations are given in MacWilliams' and Sloane's book, while yet another is given in Fenton and Vámos (1982). Secondly, although a complete solution to the problem seems inaccessible at present, the known results suggest a tantalizingly simply stated conjecture:

Conjecture 15.2 If $2 \leqslant r \leqslant q$, then

$$\max_r(r, q) = q + 1$$

(except that $\max_3(3, q) = \max_{q-1}(q-1, q) = q + 2$ if $q = 2^h$).

Note that the conjecture has already been proved for $r = 2$ (Theorem 14.4) and for $r = 3$ (Theorem 14.16). Before considering the conjecture further let us dispose of the rather uninteresting cases outside the range to which it applies. For redundancies 0 and 1, MDS codes exist of any length n over any field $GF(q)$ (for $r = 0$, $V(n, q)$ is an $[n, n, 1]$-code, while for $r = 1$, the matrix

$$\left[\begin{array}{c|c} I_{n-1} & \begin{matrix}1\\ \vdots\\ 1\end{matrix}\end{array}\right]$$

generates an $[n, n-1, 2]$-code). Cases $r > q$ are covered by the following theorem.

Theorem 15.3 If $r \geqslant q$, then $\max_r(r, q) = r + 1$. Any MDS code of redundancy $r \geqslant q$ is equivalent to a repetition code of length $r + 1$.

Proof The repetition code of length $r + 1$ is an $[r+1, 1, r+1]$-code with generator matrix $[1\,1 \cdots 1]$. Hence

$$\max_r(r, q) \geqslant r + 1.$$

Also, it is clear that any $[r+1, 1, r+1]$-code is equivalent to a repetition code.

Now suppose $r \geqslant q$ and suppose for a contradiction that $\max_r(r, q) \geqslant r + 2$. Then there exists an $[r+2, 2, r+1]$-code C over $GF(q)$. This code C must be equivalent to a code having generator matrix

$$G = \begin{bmatrix} 1 & 0 & 1 & 1 & \cdots & 1 \\ 0 & 1 & a_1 & a_2 & \cdots & a_r \end{bmatrix}.$$

In order that any linear combination of the rows of G has weight at least $r + 1$, the a_is must be distinct non-zero elements of $GF(q)$. This implies that $r \leqslant q - 1$, contrary to hypothesis.

Remark It follows from Theorem 15.3 and the preceding remarks that the only *binary* MDS codes are $V(n, 2)$, the even weight codes E_n, and repetition codes. So this chapter is really of interest only for codes over $GF(q)$ with $q > 2$.

From now on we assume that r lies in the range $2 \leqslant r \leqslant q$ and return to our consideration of Conjecture 15.2. Our first task will be to show that there exist MDS codes which meet the conjectured values of $\max_r(r, q)$ in all cases.

Theorem 15.4 Suppose $2 \leqslant r \leqslant q$. Let $a_1, a_2, \ldots, a_{q-1}$ be the non-zero elements of $GF(q)$. Then the matrix

$$H = \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 & 0 \\ a_1 & a_2 & \cdots & a_{q-1} & 0 & 0 \\ a_1^2 & a_2^2 & \cdots & a_{q-1}^2 & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ & & & & 0 & 0 \\ a_1^{r-1} & a_2^{r-1} & \cdots & a_{q-1}^{r-1} & 0 & 1 \end{bmatrix}$$

is the parity-check matrix of an MDS $[q+1, q+1-r, r+1]$-code. Equivalently, the columns of H form a $(q+1)$-arc in $PG(r-1, q)$.

Proof This is exactly the same as the proof of Theorem 14.13(i), for the determinant of a matrix formed by any r columns of H is equal to the determinant of a Vandermonde

matrix and so is non-zero. Thus any r columns of H are linearly independent.

Corollary 15.5 If $2 \leq r \leq q$, then $\max_r (r, q) \geq q + 1$.

As we saw in Theorem 14.13(ii), in the case where q is even and $r = 3$, we may add the further column

$$\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

to the matrix H of Theorem 15.4 to get an MDS code of length $q+2$. Such a trick will not work for $r > 3$. However, we see from Conjecture 15.2 that the case q even and $r = q - 1$ also seems to be special. Indeed there exists an MDS code of length $q+2$ in this case too. This fact will follow from the very useful result that the dual code of an MDS code is also MDS, thus implying that the roles of dimension and redundancy are interchangeable in so far as the existence of MDS codes is concerned. In order to show this duality, we first reformulate our problem in terms of matrices having every square submatrix non-singular.

Definitions A square matrix is called *non-singular* if its columns are linearly independent, or equivalently, if it has a non-zero determinant (cf. Theorem 11.2).

Given any matrix A, a $t \times t$ *square submatrix* of A is a $t \times t$ matrix consisting of the entries of A lying in some t rows and some t columns of A.

For example, if

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \end{bmatrix},$$

then

$$\begin{bmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} a_{11} & a_{14} \\ a_{31} & a_{34} \end{bmatrix}$$

are examples of 2×2 square submatrices of A.

Theorem 15.6 Suppose C is an $[n, n-r]$-code with parity-check

matrix $H = [A^T \mid I_r]$. Then C is an MDS code (i.e. $d(C) = r + 1$) if and only if every square submatrix of A is non-singular.

Proof By Theorem 8.4, C is an MDS code if and only if any r columns of H are linearly independent, i.e. if and only if any $r \times r$ submatrix of H is non-singular. Let us interpret this last condition on H as a condition on A^T. Suppose B is an $r \times r$ submatrix of H obtained by choosing some r columns of H. Suppose t of the chosen columns are from A^T and $r - t$ of them from I_r. If we expand $\det B$ about the last $r - t$ columns, we end up with

$$\det B = \pm \det B',$$

where B' is the $t \times t$ matrix obtained by taking the $r \times t$ matrix consisting of the t chosen columns of A^T and then deleting the $r - t$ rows corresponding to where the chosen columns of I_r have 1s. To illustrate this point suppose

$$H = \begin{bmatrix} a_{11} & a_{21} & a_{31} & 1 & 0 & 0 & 0 \\ a_{12} & a_{22} & a_{32} & 0 & 1 & 0 & 0 \\ a_{13} & a_{23} & a_{33} & 0 & 0 & 1 & 0 \\ a_{14} & a_{24} & a_{34} & 0 & 0 & 0 & 1 \end{bmatrix}.$$

If B is the 4×4 submatrix of H consisting of columns 1, 3, 5 and 6, then

$$\det B = \det \begin{bmatrix} a_{11} & a_{31} & 0 & 0 \\ a_{12} & a_{32} & 1 & 0 \\ a_{13} & a_{33} & 0 & 1 \\ a_{14} & a_{34} & 0 & 0 \end{bmatrix} = \det \begin{bmatrix} a_{11} & a_{31} \\ a_{14} & a_{34} \end{bmatrix} = \det B'.$$

Returning to the general case, it follows that B is non-singular if and only if the corresponding square submatrix B' is non-singular. It is clear that *any* $t \times t$ square submatrix B' of A^T (for any t with $1 \leq t \leq r$) arises from some $r \times r$ submatrix B of H in this way, and so the result follows.

Corollary 15.7 The dual code of an MDS code is also MDS.

Proof The code C with parity-check matrix $[A^T \mid I_r]$ is MDS $\Leftrightarrow A^T$ has the property that every square submatrix is non-singular

$\Leftrightarrow A$ has the same property (since the determinant of any square matrix is equal to the determinant of its transpose)
$\Leftrightarrow$ the code $C^{\perp}$ with parity-check matrix $[I_{n-r} \mid -A]$ is MDS.

It follows from Corollary 15.7 that generator matrices and parity-check matrices of MDS $[n, k]$-codes serve also as parity-check matrices and generator matrices respectively of MDS $[n, n-k]$-codes.

Corollary 15.8 There exists an MDS $[n, k]$-code over $GF(q)$ if and only if there exists an MDS $[n, n-k]$-code over $GF(q)$.

Corollary 15.9 Suppose $q = 2^h$, $h > 1$. Then there exists a $[q+2, 3, q]$-code over $GF(q)$. Equivalently, there exists a $(q+2)$-arc in $PG(q-2, q)$.

Proof By Theorem 14.13(ii), there exists a $[q+2, q-1, 4]$-code over $GF(q)$. By Corollary 15.7, its dual code is a $[q+2, 3, q]$-code.

Combining the results of Corollaries 14.14, 15.5 and 15.9, we have

Theorem 15.10 If $2 \leqslant r \leqslant q$, then $\max_r(r, q) \geqslant q+1$. If also $q = 2^h$ and $r = 3$ or $q-1$, then $\max_r(r, q) \geqslant q+2$.

The known results concerning Conjecture 15.2

Theorem 15.10 shows that the conjectured values of $\max_r(r, q)$ are all lower bounds. The conjecture was shown to be true for $r = 2$ and $r = 3$ in Theorems 14.4 and 14.16. We mention without proof that, by geometric methods, the conjecture has also been proved for $r = 4$ and $r = 5$, for all q (Segre 1955 and Casse 1969). Using the duality result of Corollary 15.8, the truth of the conjecture for $r \leqslant 5$ implies its truth also for r in the range $q-3 \leqslant r \leqslant q$ (see Exercises 15.2 and 15.3). [This last result was first proved in a different way by Thas (1968), who also showed (1968, 1969) that the conjecture is true for q odd in the ranges $q > (4r-9)^2$ and $q-3 > r > q - \frac{1}{4}\sqrt{q} - 5/4$].

Following MacWilliams and Sloane (1977), we show the results

graphically in Fig. 15.11, which neatly illustrates the symmetry between dimension k and redundancy r.

The broken line $n = k + r = q + 1$ in Fig. 15.11 is the conjectured bound above which no MDS code is known to exist. The heavy line represents an upper bound given by repeated application of the recursive bound

$$\max_{r+1}(r+1, q) \leqslant \max_r(r, q) + 1$$

(see Exercise 15.5), starting at $\max_5(5, q) = q+1$ (thus $\max_6(6, q) \leqslant q+2$, $\max_7(7, q) \leqslant q+3, \ldots, \max_r(r, q) \leqslant q + r - 4$ for $r \geqslant 6$). The region marked with a question mark is therefore the 'grey' area where the existence of MDS codes is undecided.

Finally we mention that the conjecture has been verified by exhaustive search for $q \leqslant 11$, for all r (Maneri and Silverman

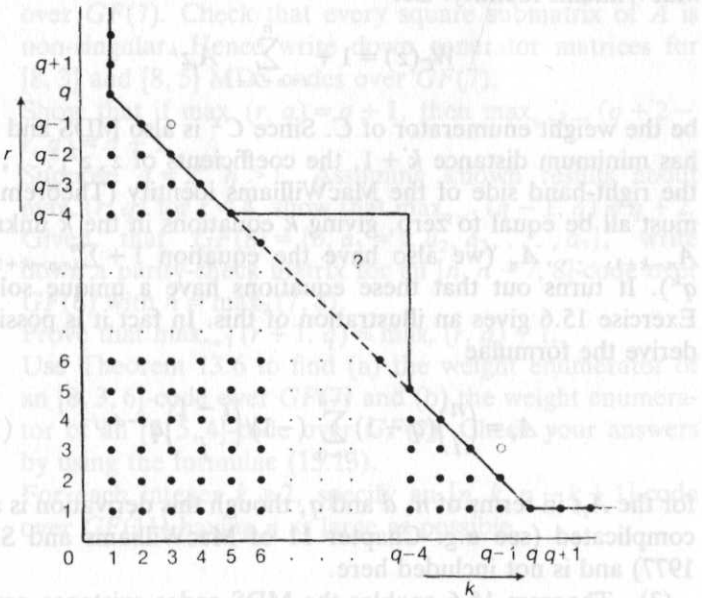


Fig. 15.11. Values of k, r for which a $[k+r, k]$-MDS code exists. ● means MDS code exists for all q. ○ means MDS code exists if and only if $q = 2^h$.

1966 and Jurick 1968) and so the smallest undecided case is $\max_6(6, 13) = 14$ or 15.

Concluding remarks on Chapter 15

(1) The Reed–Solomon codes described in Chapter 11 are MDS codes; they are shortened versions of the codes defined in Theorem 15.4. Since MDS codes meet the Singleton bound, Theorem 15.4 enables Theorem 11.4 to be improved to

Theorem 15.12 If q is a prime power and if $d \leq n \leq q+1$, then

$$A_q(n, d) = B_q(n, d) = q^{n-d+1}.$$

(2) One remarkable property of an MDS $[n, k]$-code C over $GF(q)$ is that its weight enumerator is completely determined by the values of n, k and q and does not depend on the code C itself. This fact is a little less surprising when one considers the MacWilliams identity. Let

$$W_C(z) = 1 + \sum_{i=n-k+1}^{n} A_i z^i$$

be the weight enumerator of C. Since $C^\perp$ is also MDS and hence has minimum distance $k+1$, the coefficients of $z, z^2, \ldots, z^k$ on the right-hand side of the MacWilliams identity (Theorem 13.6) must all be equal to zero, giving k equations in the k unknowns $A_{n-k+1}, \ldots, A_n$ (we also have the equation $1 + \sum_{i=n-k+1}^{n} A_i = q^k$). It turns out that these equations have a unique solution. Exercise 15.6 gives an illustration of this. In fact it is possible to derive the formulae

$$A_i = \binom{n}{i}(q-1)\sum_{j=0}^{i-d}(-1)^j\binom{i-1}{j}q^{i-d-j} \qquad (15.13)$$

for the A_is in terms of n, d and q, though this derivation is a little complicated (see e.g. Chapter 11 of MacWilliams and Sloane, 1977) and is not included here.

(3) Theorem 15.6 enables the MDS codes existence problem to be posed in elementary terms, independently of any terminology from coding theory or geometry. In view of Theorem 15.10, Conjecture 15.2 may be simply stated as follows.

Conjecture 15.14 Any $r \times k$ matrix over $GF(q)$ with $2 \leq r$, $k \leq q$ and having the property that any square sub-matrix is non-singular satisfies

$$r + k \leq q + 1$$

(except for case $q = 2^h$ and r or $k = 3$).

From an earlier remark, the smallest possible counter-example is a 6×9 or 7×8 matrix over $GF(13)$.

Exercises 15

15.1 Consider the matrix

$$A = \begin{bmatrix} 1 & 6 & 2 & 5 & 1 \\ 1 & 4 & 3 & 3 & 6 \\ 1 & 5 & 5 & 1 & 5 \end{bmatrix}$$

over $GF(7)$. Check that every square submatrix of A is non-singular. Hence write down generator matrices for $[8, 3]$ and $[8, 5]$ MDS codes over $GF(7)$.

15.2 Show that if $\max_r(r, q) = q+1$, then $\max_{q+2-r}(q+2-r, q) = q+1$.

15.3 Suppose $q = 2^h$, $h > 1$. Assuming known results about $\max_r(r, q)$ for $r \leq 4$, show that $\max_{q-1}(q-1, q) = q+2$.

15.4 Given that $GF(8) = \{0, a_1 = 1, a_2, a_3, \ldots, a_7\}$, write down a parity-check matrix for an $[n, n-7, 8]$-code over $GF(8)$ with $n = \max_7(7, 8)$.

15.5 Prove that $\max_{r+1}(r+1, q) \leq \max_r(r, q) + 1$.

15.6 Use Theorem 13.6 to find (a) the weight enumerator of an $[8, 3, 6]$-code over $GF(7)$ and (b) the weight enumerator of an $[8, 5, 4]$-code over $GF(7)$. Check your answers by using the formulae (15.13).

15.7 For each integer $k \geq 2$, specify an $[n, k, n-k+1]$-code over $GF(11)$ having n as large as possible.

16 Concluding remarks, related topics, and further reading

The main aims of this final chapter are to review the progress made in the earlier chapters and to mention some related topics, with suggestions for further reading.

The treatment presented in the book has been motivated mainly by two recurring themes:

(1) the problem of finding codes which are optimal in some sense;

(2) the problem of decoding such codes efficiently.

This has led to a rich interplay with several well-established branches of mathematics, notably algebra, combinatorics, and geometry.

With regard to optimal codes, the main emphasis has been on finding values of $A_q(n, d)$, the largest size M of an (n, M, d)-code over an alphabet of q letters. In the case of binary codes, we gave in Table 2.4 the state of knowledge regarding values of $A_2(n, d)$ for small n and d. We now consider this table again (Table 16.1), for $d \leq 5$, in order to indicate those places in the text where results have been proved.

Remarks 16.2 (1) All of the bounds in Table 16.1 have been proved in the text or exercises with the exceptions of (i) the upper bounds obtained by linear programming methods and (ii) the lower bounds for $d = 3$ and $n = 9$, 10, and 11. A rather complicated construction of an $(11, 144, 3)$-code was given by Golay (1954). Successive shortenings of this code give codes with parameters $(10, 72, 3)$, $(9, 38, 3)$ and $(8, 20, 3)$. For a long time it was believed that the $(9, 38, 3)$-code was optimal, but recently Best (1980) found a $(9, 40, 3)$-code (despite a publication of 1959 which claimed that 39 was an upper bound on $A_2(9, 3)$!).

(2) It is conjectured that the Plotkin bound is always attained in the range $d \leq n \leq 2d + 1$. Indeed it has been shown by Levenshtein (1964) that there exist codes which meet the Plotkin

Table 16.1

Values of $A_2(n, d)$

n	$d=3$			$d=5$		
3	R	2	P		—	
4	R^+	2	P		—	
5	SH	4	P	R	2	P
6	SH	8	P	R^+	2	P
7	H	16	S	R^{++}	2	P
8	G	20	L_1	E_2	4	P
9	B	40	E_3	SD	6	P
10	G	72–79	L_2	SD	12	P
11	G	144–158	E_3	D	24	P
12	SH	256	L_1	SNR	32	L_1
13	SH	512	E_3	SNR	64	E_3
14	SH	1024	E_3	SNR	128	E_3
15	H	2048	S	NR	256	E_3
16	E_1	2560–3276	L_1	NR^+	256–340	L_1

Key to Table 16.1

Lower Bounds

If C is a given code, then:

C^+ denotes the code obtained from C by adding an extra zero coordinate,

SC denotes a code obtained by shortening C, possibly more than once, i.e. use E_3 (below) in the form $A_2(n-1, d) \geqslant \frac{1}{2}A_2(n, d)$.

R: repetition code (Example 1.11).

H: Hamming code (Theorem 8.2).

B: Best (1980).

G: Golay (1954); for alternative constructions see MacWilliams and Sloane (1977, Chapter 2, §7). A $(20, 8, 3)$-code is also constructed in Exercise 2.16.

E_1: a $(\mathbf{u} \mid \mathbf{u}+\mathbf{v})$-construction (see Exercise 2.18).

E_2: see Exercise 2.8.

D: constructed from a Hadamard design (Exercise 2.12).

NR: Nordstrom–Robinson code (Exercise 9.9).

Upper Bounds

P: Plotkin bound (Exercise 2.22).

Key to Table 16.1 (Contd.)

S: sphere-packing bound (Theorem 2.16).

L: linear programming bound (L_1: see Best *et al.* (1978) or MacWilliams and Sloane (1977); L_2: see Best (1980)).

E_3: $A_2(n, d) \leqslant 2A_2(n-1, d)$ (Exercise 2.2).

---

bound provided certain Hadamard matrices of order $m \leqslant n$ exist, for $m \equiv 0 \pmod 4$. [A *Hadamard matrix* of order m is an $m \times m$ matrix of $+1$s and -1s such that $HH^T = mI$ (over the field of real numbers). It is easy to associate a Hadamard design with a Hadamard matrix and we have already seen how such designs give rise to optimal codes (see Exercises 2.15 and 2.24)]. An introduction to Hadamard configurations may be found in Anderson (1974). A proof of Levenshtein's theorem may be found in Chapter 2 of MacWilliams and Sloane (1977). It is also a well-known conjecture that Hadamard matrices of order m exist for all positive integers $m \equiv 0 \pmod 4$. This conjecture is known to be true for $m \leqslant 264$ and so the Plotkin bound is indeed tight for $n \leqslant 264$ (in the range $2d+1 \geqslant n$).

(3) Values of $A_2(n, d)$ found in the text but outside the range of Table 16.1 include:

$A_2(23, 7) = 4096$ (Theorem 11.3 or 12.20)

$A_2(n, 3) = 2^{n-r}$, whenever $n = 2^r - 1$ (Corollary 8.7).

As well as considering optimal binary codes, much attention has also been given in this text to optimal q-ary codes for general q. For example: in Chapter 8 we showed that, for a prime power q, $A_q(n, 3) = q^{n-r}$ for any n of the form $(q^r - 1)/(q-1)$; in Chapter 9 we showed that $A_3(11, 5) = 3^6$; in Chapter 10 we found the values of $A_q(4, 3)$ for general q; and in Chapter 15 we showed that $A_q(n, d) = q^{n-d+1}$ if q is a prime power and $d \leqslant n \leqslant q+1$.

Finally, the problem of finding optimal *linear* codes over $GF(q)$ was considered in Chapters 14 and 15.

A topic not covered in this text is that of *asymptotic bounds*, applicable when n is large. However, much research has been devoted to closing the gap between the best-known asymptotic lower and upper bounds, which are currently an asymptotic

version of the Gilbert–Varshamov lower bound (cf. Theorem 8.10) and an upper bound, obtained by linear programming methods, due to McEliece *et al.* (1977). Good accounts of this topic may be found in MacWilliams and Sloane (1977) and van Lint (1982).

We now give brief descriptions of some types of code not previously discussed in this text.

Burst-error correcting codes

The codes we have considered to date are designed to correct *random* errors (e.g. for a binary symmetric channel). It often happens that we need a code for a channel which does not have random errors but which has errors in *bursts*, i.e. several errors close together. There are some linear cyclic codes which are well adapted for burst-error correcting, two important families being *Reed–Solomon codes* and *Fire codes*. An alternative procedure is to scramble the order in which the digits are transmitted, the scrambling occurring over a length of several blocks. Then at the receiving end the order is changed back to the original sequence. This change-back will break up any bursts of errors, leaving errors scattered in a pseudo-random way over several blocks, so that they fall within the capacity of random-error correcting codes. The *interleaving of codes* is one way of carrying out this procedure.

For a good account of burst-error correcting codes, see Peterson and Weldon (1972) or Dornhoff and Hohn (1978).

Convolutional codes

Convolutional codes are powerful error-correcting codes which were introduced by Elias in 1955. They are unlike the codes we have already considered in that message symbols are not broken up into blocks for encoding. Instead check digits are interleaved within a long stream of information digits. For example, for rate $\frac{1}{2}$, one might have the information input $x_1x_2x_3\cdots$ encoded as $x_1x_1'x_2x_2'x_3x_3'\ldots$, where each check digit x_i' is a function of $x_1, x_2, \ldots, x_i$ which is found by means of a feed-back shift register. The decoding is done one digit at a time using the previously received and corrected digits.

Mathematicians tend to be less interested in convolutional codes because the mathematical theory is nothing like as well developed as for block codes. Convolutional codes are also intrinsically more difficult. Despite this, such codes have been extensively used in practice. For example, NASA has been using convolutional codes in deep-space applications since 1977 (from 1969 to 1976, NASA's Mariner-class spacecraft had used a Reed–Muller [32, 6]-block code, as mentioned in Chapter 1).

Chapters on convolutional codes are included in the books by Blahut (1983), McEliece (1977), Peterson and Weldon (1972), and van Lint (1982).

Cryptographic codes

Cryptographic codes have little in common with error-correcting codes, for their aim is the *concealment of information*. The last decade has seen an explosion of interest in such codes following the invention of the concept of the *public-key cipher system* by Diffie and Hellman (1976). Such a system makes use of a *one-way trapdoor function*. This is an encrypting function which has an inverse decrypting function; but if only the encrypting function is known, it is computationally infeasible to discover the decrypting function. This means that a person R can publish his encrypting algorithm (e.g. in a directory) so that *any* member of the public can send messages to R in complete secrecy, for only R knows his own decrypting algorithm. Such a public-key system thus overcomes the weakness of a traditional cipher system which requires the secret delivery of a 'key' in advance of sending a secret message.

Rivest *et al.* (1978) found an elegant way to implement the Diffie–Hellman system by using prime numbers and a simple consequence of Fermat's theorem (Exercise 16.1). Their method relies on the facts that

(a) there are computer algorithms for testing primality which are extremely fast (e.g. a few seconds for a 100-digit number), while

(b) all known algorithms for factorizing composite numbers are extremely slow (e.g. if n is a 200-digit number obtained by multiplying two 100-digit prime numbers, the fastest of today's

computers, using the best-known algorithm, would take millions of years to find the prime factors of n).

THE RIVEST–SHAMIR–ADLEMAN (R–S–A) CRYPTOSYSTEM

Let us assume that all messages are encoded as large decimal numbers (e.g. via $A = 01$, $B = 02$, ..., $Z = 26$). The purpose here is not to encrypt the message but merely to get it in the numeric form necessary for encryption.

A subscriber R chooses two large prime numbers p and q, each about 100 digits long, and calculates $n = pq$. He then finds two numbers s and t such that

$$st \equiv 1 \pmod{(p-1)(q-1)},$$

i.e. $st = r(p-1)(q-1) + 1$, for some integer r.

R publishes the numbers n and s but keeps the numbers p, q, and t secret. He also publishes the encryption algorithm, which is simply:

'encipher a message number x as $y = x^s \pmod{n}$'.

To decipher the received message y, R simply calculates $y^t \pmod{n}$. This gives the original message x because, using Exercise 16.1, we have

$$y^t = x^{st} = x^{r(p-1)(q-1)+1} \equiv x \pmod{n}.$$

Remarks (i) A long message number must be broken into blocks, so that each block represents a number smaller than n. The blocks are then enciphered separately.

(ii) Even if n is an enormous number, say 200 digits, a message can be enciphered or deciphered very efficiently, using less than one second of computing time.

(iii) A subscriber R can construct (privately) his key numbers p, q, n, s and t very quickly with a computer. It takes a few seconds to generate a pair of random prime numbers p and q, each having about 100 digits. Then, for a random choice of s, the Euclidean algorithm provides a very fast method of calculating t such that $st \equiv 1 \pmod{(p-1)(q-1)}$.

(iv) The deciphering procedure is secret because t is known only to R. To find t from n and s requires knowledge of p and q.

This in turn requires factorizing n, which we have already remarked to be computationally infeasible (by known methods).

An illustration of an R–S–A cryptosystem in which p and q are small prime numbers, so that the code may easily be broken, is given in Exercise 16.2.

Interesting expository articles on cryptographic codes are Gardner (1977) and Sloane (1981). For a comprehensive treatment of cipher systems in general, Beker and Piper (1982) is recommended.

Variable-length source codes

In order to illustrate the ideas here, let us consider the problem of transmitting English text over a binary symmetric channel as quickly and as reliably as possible. This can be carried out by applying two codes in series. First a *source code* encodes the text into a long string of binary digits. For reliability, this binary data is then broken into blocks of length k and each block encoded into a codeword of length n by means of an error-correcting $[n, k]$-code. Decoding of the two codes is, of course, done in reverse order.

In choosing the source code we are not concerned with the error-correcting aspects. Our main aim is to encode the source alphabet as economically as possible. If letters in the source alphabet occur with differing frequencies, we can best do this by using a *variable-length source code*.

We now give three examples of source codes for our alphabet of 27 letters ('A' to 'Z' and 'space').

ASCII CODE (AMERICAN STANDARD CODE FOR INFORMATION INTERCHANGE)

Computers are usually constructed internally to handle only 0s and 1s. A source code is therefore required to translate each typed character into a binary vector. A common such code is the ASCII code. This has $128 = 2^7$ codewords representing letters of the alphabet (upper and lower case), digits 0 to 9, and assorted other symbols and instructions. Each codeword is a binary vector of length 7 together with an overall parity check (so that any

Character	Probability	ASCII code	Morse code	Huffman code
space	0.185 9	01000001	space	000
A	0.064 2	10000010	01	0100
B	0.012 7	10000100	1000	0111111
C	0.021 8	10000111	1010	11111
D	0.031 7	10001000	100	01011
E	0.103 1	10001011	0	101
F	0.020 8	10001101	0010	001100
G	0.015 2	10001110	110	011101
H	0.046 7	10010000	0000	1110
I	0.057 5	10010011	00	1000
J	0.000 8	10010101	0111	0111001110
K	0.004 9	10010110	101	01110010
L	0.032 1	10011001	0100	01010
M	0.019 8	10011010	11	001101
N	0.057 4	10011100	10	1001
O	0.063 2	10011111	111	0110
P	0.015 2	10100000	0110	011110
Q	0.000 8	10100011	1101	0111001101
R	0.048 4	10100101	010	1101
S	0.051 4	10100110	000	1100
T	0.079 6	10101001	1	0010
U	0.022 8	10101010	001	11110
V	0.008 3	10101100	0001	0111000
W	0.017 5	10101111	011	001110
X	0.001 3	10110001	1001	0111001100
Y	0.016 4	10110010	1011	001111
Z	0.000 5	10110100	1100	0111001111

Fig. 16.3. Codes for the English alphabet.

single error may be detected). In other words, the ASCII code is the binary even-weight code of length 8. Those codewords representing upper case letters are shown in Fig. 16.3.

For other applications, a fixed-length code such as the ASCII code may be uneconomical.

MORSE CODE

This is a variable-length code which takes advantage of the high frequency of occurrence of some letters, such as 'E', by making their codewords short, while very infrequent letters, such as 'Q',

are represented by longer codewords. The Morse code is given in Fig. 16.3, where the 0s may be read as dots and the 1s as dashes. Although the Morse code may appear to be a binary code, it is in fact a ternary code, having the symbols dot, dash, and space. A space has to be left between letters (and at least two spaces between words), for otherwise the code cannot be uniquely decoded; for example, the message 01000110 can mean either LEG or RUN unless spaces are inserted between letters. This drawback means that the Morse code is rarely used nowadays.

HUFFMAN CODES

Suppose a source alphabet has N letters $a_1, a_2, \ldots, a_N$ and that the probability of occurrence of a_i is p_i. Then if each a_i is encoded into a word of length l_i, the average word-length of the code is $\sum_{i=1}^{N} p_i l_i$.

Huffman coding is an ingenious way of matching codewords to source symbols so that

(a) the code is uniquely decodable, i.e. when any string of source symbols has been encoded into a string of binary digits, it is always clear where one codeword ends and the next one begins, and
(b) the average word-length is as small as possible.

While omitting the details of how Huffman codes may be constructed, we give an example of such a code for the English alphabet in Fig. 16.3. From the given probabilities, it may be calculated that the average word length is 4.1195. This gives a saving of nearly 18% on the best fixed-length code we could have used, in which all codewords have length 5 (any fixed-length code is clearly uniquely decodable). The reason why a Huffman code is uniquely decodable is that no codeword is a *prefix* of any other codeword, i.e. if $x_1x_2\cdots x_n$ is any codeword, then there is no codeword of the form $x_1x_2\cdots x_nx_{n+1}\cdots x_m$ for any $m > n$.

For a good account of Huffman source coding, the reader is referred to McEliece (1977), Jones (1979), or Hamming (1980).

Exercises 16

16.1 Suppose p and q are distinct prime numbers. Prove that for any integers x and r,

$$x^{r(p-1)(q-1)+1} \equiv x \pmod{pq}$$

[*Hint:* Use Fermat's theorem: 'if $x \not\equiv 0 \pmod{p}$, then $x^{p-1} \equiv 1 \pmod{p}$' (cf. Exercise 3.8).]

16.2 Suppose a person's published encryption algorithm reads: 'Convert your message to a large decimal number via the code $A = 01$, $B = 02, \ldots, Z = 26$, space $= 00$. Break this number into blocks of length 4. Encipher each block x into the 4-digit block $y = x^{283} \pmod{2813}$'.

Find the decryption algorithm for the above code and hence (with the aid of a pocket calculator) decipher the following intercepted message:

2385 0593 0736 0209 1671 2595 2026 2418.

16.3 In the R–S–A cryptosystem, explain how messages can be 'signed' to prevent forgeries.

16.4 Consider a source alphabet a_1, a_2, a_3, a_4 with probabilities of occurrence $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{8}$ respectively. Which of the following source codes are (a) uniquely decodable, (b) prefix-free?

Source letter	p_i	Code A	Code B	Code C	Code D
a_1	0.5	0	00	0	0
a_2	0.25	1	01	10	01
a_3	0.125	00	10	110	011
a_4	0.125	11	11	111	0111

For those codes which are uniquely decodable, calculate the average word-length.

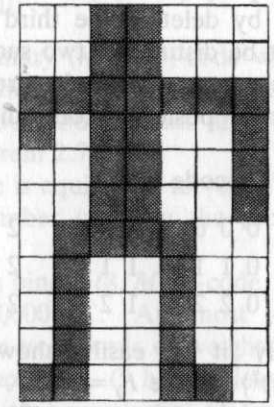
16.5 Use the Huffman code of Fig. 16.3 to decode the message

00101110101000101100101011.

Solutions to exercises

Chapter 1

1.1



[*Remark*: Pictures have actually been transmitted from Earth into outer space in this way. Two large prime numbers were used so that a much more detailed picture could be sent. It is reasonable to expect that a civilized recipient of such a message would be able to work out how to reconstruct the picture, since factorization of a number into prime factors is a property independent of language or notation.]

1.2 If either 00000 or 11111 is sent, then the received vector will be decoded as the codeword sent if and only if two or fewer errors occur. So the probability that the received vector is corrected to the codeword sent is

$$(1-p)^5 + 5(1-p)^4 p + \binom{5}{2}(1-p)^3 p^2$$
$$= 1 - 10p^3 + 15p^4 - 6p^5,$$

whence the word error probability is $10p^3 - 15p^4 + 6p^5$.

1.3 Suppose $d(C) = 4$. If a received vector $\mathbf{y}$ has distance $\leqslant 1$ from some codeword, we decode as that codeword. If $\mathbf{y}$ has distance at least 2 from every codeword, we seek re-transmission. This scheme guarantees the simultaneous correction of single errors and detection of double errors. Note that C could also be used either as a single-error-correcting code or as a *triple*-error-detecting code, but not both *simultaneously* (why not?).

1.4 $\lfloor (16-1)/2 \rfloor = 7$.

1.5 Suppose C is a q-ary $(3, M, 2)$-code. Then the M ordered pairs obtained by deleting the third coordinate of each codeword must be distinct (if two such pairs were identical, then the corresponding codewords of C would differ only in the third position, contradicting $d(C) = 2$). So $M \leqslant q^2$.

A 3-ary $(3, 9, 2)$-code is

0 0 0	1 0 1	2 0 2
0 1 1	1 1 2	2 1 0.
0 2 2	1 2 0	2 2 1

More generally it is easily shown that $\{(a, b, a+b) \mid (a, b) \in (F_q)^2\}$, where $F_q = \{0, 1, \ldots, q-1\}$ and $a+b$ is calculated modulo q, is a q-ary $(3, q^2, 2)$-code.

Chapter 2

2.1 (i) $\{000000, 111111\}$. (ii) $(F_2)^3$. (iii) Add overall parity-check to $(F_2)^3$. (iv) Not possible. Suppose C were a $(5, 3, 4)$-code. There is no loss in assuming 00000 is a codeword. But then the other two codewords each have at least four 1s, which implies that they differ in at most 2 places. (v) Not possible. A binary $(8, M, 3)$-code satisfies the sphere-packing bound, $M(1+8) \leqslant 2^8$, which implies that $M \leqslant 28$.

2.2 Suppose C is a binary (n, M, d)-code. Partition the codewords of C into two disjoint sets, those ending with a 0 and those ending with a 1. One or other of these sets contains at least $M/2$ of the codewords. Take this set and delete the last coordinate to get an $(n-1, \geqslant M/2, d)$-code

(this is called a *shortened code* of C). Taking $M = A_2(n, d)$ gives $A_2(n-1, d) \geqslant \frac{1}{2}A_2(n, d)$.

2.3 Immediate from Exercise 1.5.

2.4 Let C be the code obtained from $(F_2)^{n-1}$ by adding an overall parity check. Every codeword of C has even weight and so $C \subseteq E_n$. Since every vector of E_n may be obtained from one in $(F_2)^{n-1}$ in this way, we have $C = E_n$. Thus $|E_n| = |(F_2)^{n-1}| = 2^{n-1}$. $(F_2)^{n-1}$ has minimum distance 1, and so E_n has minimum distance 2.

2.5 $\binom{50}{8} \Big/ \binom{10}{8} > 10^7$.

2.6 Let C be a binary (n, M, d)-code with d even. Delete a suitable coordinate from all codewords to get an $(n-1, M, d-1)$-code and then add an overall parity check (cf. proof of Theorem 2.7).

2.7 Any such code is equivalent to $\{00\cdots0, 11\cdots100\cdots0\}$, where the number of 1s in the second word is one of $1, 2, \ldots, n$.

2.8 Suppose C is a binary $(8, M, 5)$-code, with $M \geqslant 4$. We may assume $00000000 \in C$. At most one codeword has weight $\geqslant 6$, for two words of weight $\geqslant 6$ could differ in at most four places. So C has at least two codewords of weight 5. Up to equivalence, we may assume these are 11111000 and 11000111. It is now easy to show that the only further codeword possible is 00111111.

2.9 Let C be an (n, q, n)-code over $F_q = \{1, 2, \ldots, q\}$ and let A be a matrix whose rows are the codewords of C. Since $d(C) = n$, the q elements of any column of A must be distinct and so must be precisely the symbols $1, 2, \ldots, q$ in some order. For each column of A a suitable permutation of the symbols may be performed to give

$$A = \begin{matrix} 1 & 1 & \cdots & 1 \\ 2 & 2 & \cdots & 2 \\ \vdots & \vdots & & \vdots \\ q & q & \cdots & q. \end{matrix}$$

2.10 Apply either the sphere-packing bound or an argument similar to that of Exercise 1.5 (i.e. the words formed by deleting the last two coordinates must be distinct).

2.11 By Corollary 2.8 and Example 2.23, we have

$$A_2(8,4) = A_2(7,3) = 16.$$

2.12 Take as codewords the 11 rows of an incidence matrix of the design, the 11 vectors obtained by interchanging all 0s and 1s, the all-0 vector, and the all-1 vector. The minimum distance may be shown to be 5 by an argument similar to that used in Example 2.23. A binary $(11, M, 5)$-code satisfies

$$M\left[1 + 11 + \binom{11}{2}\right] \leqslant 2^{11},$$

and so $M \leqslant 2^{11}/67$, which implies $M \leqslant 30$.

2.13 (i) Following the hint: for each of the v choices of x there are r choices of B: for each of the b choices of B there are k choices of x. So the number of pairs in the set is $vr = bk$.

(ii) Let y be a fixed point. Count in two ways the number of ordered pairs in the set

$$\{(x, B)\colon x \text{ is a point, } B \text{ is a block, } x \neq y \text{ and both } x \text{ and } y \in B\}.$$

2.14 (i) Condition (ii) of the previous exercise is not satisfied.
(ii) Immediate from Theorem 2.27(i).

2.15 Easy generalization of the argument of Example 2.23, Exercise 2.12.

2.16 Straightforward check (just 34 comparisons of codewords are required: 11010000 with 19 others, then 11100100 with 11 others, then 10101010 with 3 others, and finally **0** with **1**).

2.17 Since $(\mathbf{u}_1 \mid \mathbf{u}_1 + \mathbf{v}_1) = (\mathbf{u}_2 \mid \mathbf{u}_2 + \mathbf{v}_2)$ if and only if $(\mathbf{u}_1, \mathbf{v}_1) = (\mathbf{u}_2, \mathbf{v}_2)$, the number of codewords in C_3 is M_1M_2.

Let $\mathbf{a} = (\mathbf{u} \mid \mathbf{u} + \mathbf{v})$ and $\mathbf{b} = (\mathbf{u}' \mid \mathbf{u}' + \mathbf{v}')$ be distinct codewords of C_3.

If $\mathbf{v} = \mathbf{v}'$, then $d(\mathbf{a}, \mathbf{b}) = 2d(\mathbf{u}, \mathbf{u}') \geqslant 2d_1$.

If $\mathbf{v} \neq \mathbf{v}'$, then

$$\begin{aligned} d(\mathbf{a}, \mathbf{b}) &= d(\mathbf{u}, \mathbf{u}') + d(\mathbf{u} + \mathbf{v}, \mathbf{u}' + \mathbf{v}') \\ &= w(\mathbf{u} + \mathbf{u}') + w(\mathbf{u} + \mathbf{v} + \mathbf{u}' + \mathbf{v}') \\ &= d(\mathbf{u} + \mathbf{u}', \mathbf{0}) + d(\mathbf{u} + \mathbf{u}', \mathbf{v} + \mathbf{v}') \\ &\geqslant d(\mathbf{0}, \mathbf{v} + \mathbf{v}') \quad \text{(by the triangle inequality)} \\ &= d(\mathbf{v}, \mathbf{v}') \geqslant d_2. \end{aligned}$$

2.18 Let C_1 be the $(8, 128, 2)$-code E_8 (see Exercise 2.4) and let C_2 be the $(8, 20, 3)$-code of Exercise 2.16. Apply Exercise 2.17 to get a $(16, 2560, 3)$-code.

2.19 $C_1 = (4, 8, 2)$-code, $C_2 = (4, 2, 4)$-code $\Rightarrow$ $C_3 = (8, 16, 4)$-code.

$C_1 = (8, 16, 4)$-code, $C_2 = (8, 2, 8)$-code $\Rightarrow$ $C_3 = (16, 32, 8)$-code.

$C_1 = (16, 32, 8)$-code, $C_2 = (16, 2, 16)$-code $\Rightarrow$ $C_3 = (32, 64, 16)$-code.

2.20 Since $w(\mathbf{x}_i + \mathbf{x}_j) = d(\mathbf{x}_i, \mathbf{x}_j) \geqslant d$, we have

$$w(T) \geqslant \tfrac{1}{2}M(M-1)d \tag{1}$$

Suppose $\frac{1}{2}M - t_j$ codewords have 1 in the jth position, so that $\frac{1}{2}M + t_j$ codewords have 0 in the jth position. Then the number of 1s in the jth column of T is

$$\begin{aligned} (\tfrac{1}{2}M - t_j)(\tfrac{1}{2}M + t_j) &= (\tfrac{1}{2}M)^2 - t_j^2 \\ &\leqslant \begin{cases} (\frac{1}{2}M)^2 & \text{if } M \text{ is even} \\ (\frac{1}{2}M)^2 - \frac{1}{4} & \text{if } M \text{ is odd,} \end{cases} \end{aligned}$$

since $t_j^2 \geqslant (\frac{1}{2})^2$ if M is odd. Hence

$$w(T) \leqslant \begin{cases} \frac{1}{4}M^2n & \text{if } M \text{ is even} \\ \frac{1}{4}(M^2-1)n & \text{if } M \text{ is odd} \end{cases} \tag{2}$$

(1) and (2) give the required result.

2.21 If $A_2(n, d)$ is even, the result is immediate. If $A_2(n, d)$ is odd, use $\lfloor 2x \rfloor \leqslant 2\lfloor x \rfloor + 1$.

The result gives $A_2(9, 5) \leqslant 10$ and $A_2(10, 6) \leqslant 6$. The former bound can be improved via Corollary 2.8 and the latter bound; thus $A_2(9, 5) = A_2(10, 6) \leqslant 6$.

2.22 (i) was shown in Exercise 2.21. (ii) follows from (i) and Corollary 2.8. (iii) By (i), $A_2(2d-1, d) \leqslant 2d$. Hence $A_2(2d, d) \leqslant 4d$ by Exercise 2.2. (iv) follows from (iii) and Corollary 2.8.

2.23 The $(32, 64, 16)$-code is optimal by Exercise 2.22 (iii). The generalization follows from the Remark in Exercise 2.19 and Exercise 2.22 (iii).

2.24 Immediate from Exercises 2.15 and 2.22(iii).

Chapter 3

3.1 $2^{20} \equiv (2^3)^6 2^2 \equiv 1^6 2^2 \equiv 4 \pmod 7$.
$3^{100} \equiv (3^4)^{25} \equiv 1^{25} \equiv 1 \pmod{10}$.

3.2 $x \equiv 0, 1, 2$ or $3 \pmod 4 \Rightarrow x^2 \equiv 0, 1, 0$ or $1 \pmod 4$ respectively. Hence $x^2 + y^2 \equiv 0, 1$ or $2 \pmod 4$, but $1839 \equiv 3 \pmod 4$.

3.3

x:	1	2	3	4	5	6	1	2	3	4	5	6	7	8	9	10	11	12
x^{-1}:	1	4	5	2	3	6	1	7	9	10	8	11	2	5	3	4	6	12

3.4 (i) 2, (ii) $\frac{1}{11}$.

3.5 Yes, No, No.

3.6 (i) $1 \cdot 0 + 2 \cdot 1 + 3 \cdot 3 + 4 \cdot 1 + 5x + 6 \cdot 9 + 7 \cdot 1 + 8 \cdot 3 + 9 \cdot 9 + 10 \cdot 9 \equiv 0 \pmod{11} \Rightarrow 5x + 7 \equiv 0 \Rightarrow 5x \equiv 4 \Rightarrow x \equiv 4 \cdot 5^{-1} \equiv 4 \cdot 9 \equiv 3$.

(ii) The number is 00232xy800, where we see that each of x, y is 0, 8 or 9. For the number to be an ISBN, we require $6x + 7y = 7$, i.e. $y = 1 + 7x$. Now $x = 0 \Rightarrow y = 1$; $x = 8 \Rightarrow y = 2$; $x = 9 \Rightarrow y = 9$. So $x = y = 9$.

3.7 Suppose $x_1 \cdots x_{10}$ is the codeword sent and $y_1 \cdots y_{10}$ the vector received. If a single error has occurred of magnitude a, then $\sum_{i=1}^{10} y_i = (\sum_{i=1}^{10} x_i) + a \equiv a \pmod{11}$. So the error is detected. Unlike the ISBN code, any transposition of two digits will go undetected, for then $\sum y_i = \sum x_i \equiv 0$.

3.8 $1a, 2a, \ldots, (p-1)a$ are distinct $(\mathrm{mod}\, p)$, for $ia \equiv ja \pmod p \Rightarrow i \equiv j \pmod p$ (multiplying both sides by a^{-1}). So $1a, 2a, \ldots, (p-1)a$ are congruent to the elements $1, 2, \ldots, p-1$ in some order. Hence $1a \cdot 2a \cdot \cdots \cdot (p-1)a \equiv 1 \cdot 2 \cdot \cdots \cdot (p-1) (\mathrm{mod}\, p)$ and so $(p-1)!\, a^{p-1} \equiv (p-1)! \pmod p$. Multiplying through by the inverse of $(p-1)!$ gives $a^{p-1} \equiv 1 \pmod p$.

3.9 $a \neq 0 \Rightarrow \gcd(a, p) = 1$. By the Euclidean algorithm, $1 = ax + py$ for some integers x and y. Hence $ax \equiv 1 \pmod p$ and so $x = a^{-1}$. $31 = 1 \cdot 23 + 8$; $23 = 2 \cdot 8 + 7$; $8 = 1 \cdot 7 + 1$. So $1 = 8 - 1 \cdot 7 = 3 \cdot 8 - 23 = 3 \cdot 31 - 4 \cdot 23$. Hence $-4 \cdot 23 \equiv 1 \pmod{31}$ and so $23^{-1} = -4 \equiv 27$.

3.10 2, 3, 2 (other answers possible).

3.11 Let 1 be the multiplicative identity element of F. The field elements $n1$ for $n = 1, 2, 3, \ldots$ cannot all be distinct, since F is finite. So $l1 = m1$ for some $0 < m < l$, whence $(l-m)1 = 0$. This implies that $n1 = 0$ for some integer n. Let p be the smallest positive integer such that $p1 = 0$. Then p is prime because $p = rs$, with $1 < r, s < p$, $\Rightarrow p1 = (r1)(s1) = 0 \Rightarrow r1 = 0$ or $s1 = 0$ (by Lemma 3.1 (ii)), contradicting the minimality of p. Finally, if $\alpha \in F$, then $p\alpha = \alpha + \alpha + \cdots + \alpha = \alpha(1 + 1 + \cdots + 1) = \alpha(p1) = \alpha 0 = 0$.

3.12 $\binom{p}{i} = p!/i!\,(p-i)!$. If $i \in \{1, 2, \ldots, p-1\}$, then the numerator $p!$ is divisible by p, whereas the denominator $i!\,(p-i)!$ is not. Hence $\binom{p}{i} \equiv 0 \pmod p$. By the binomial theorem,

$$(a+b)^p \equiv \sum_{i=0}^{p} \binom{p}{i} a^i b^{p-i} \equiv a^p + b^p \pmod p$$

For the last part, use induction on a.

3.13 In the product, each element x will cancel with its inverse, except when $x = x^{-1}$. Now $x = x^{-1} \Leftrightarrow x^2 = 1 \Leftrightarrow (x-1)(x+1) = 0 \Leftrightarrow x = 1$ or $x = -1$.

3.14 (i) $x^2 = y^2 \Rightarrow (x-y)^2 = x^2 - y^2 = 0 \Rightarrow x - y = 0 \Rightarrow x = y$.
So the squares of the non-zero elements are precisely the distinct non-zero elements (in some order).

(ii) *Hint:* show that if $a \neq 0$, then $x^2 = a$ has either 2 or 0 solutions.

Chapter 4

4.1 Show that this single condition holds if and only if both conditions (1) and (2) of Theorem 4.1 hold.

4.2 Suppose $\mathbf{x}, \mathbf{y} \in E_n$, so that $w(\mathbf{x})$ and $w(\mathbf{y})$ are even numbers. By Lemma 2.6, $w(\mathbf{x}+\mathbf{y}) = w(\mathbf{x}) + w(\mathbf{y}) - 2w(\mathbf{x} \cap \mathbf{y}) =$ an even number. So $\mathbf{x} + \mathbf{y} \in E_n$ and hence E_n is a subspace. By Exercise 2.4, $|E_n| = 2^{n-1}$ and so $\dim(E_n) = n - 1$.

The rows of

$$\begin{matrix} 1\,0\,0\,\cdots\,0\,1 \\ 0\,1\,0\,\cdots\,0\,1 \\ \vdots \\ 0\,0\,\cdots\,0\,1\,1 \end{matrix}$$

form a basis (other answers are possible).

4.3 $(1,2,0,1) = 2(0,1,2,1) + (1,0,2,2)$. So $\{(0,1,2,1), (1,0,2,2)\}$ is a basis and $\dim(C) = 2$.

4.4 Show that $\{\mathbf{u}, \mathbf{v}\}$ is linearly dependent if and only if either $\mathbf{u}$ or $\mathbf{v}$ is zero or $\mathbf{v}$ is a scalar multiple of $\mathbf{u}$.

4.5 In each case show that the new set is still both a spanning set and a linearly independent set.

4.6 In F, let n denote the element $1+1+\cdots+1$ (n1s). Then the subset $\{0, 1, \ldots, p-1\}$ of F may be regarded as the field $GF(p)$, since addition and multiplication are carried out modulo p. It follows at once that F is a vector space over $GF(p)$, all the axioms following immediately from the field properties of F and $GF(p)$. If the vector space F over $GF(p)$ has dimension h, then it follows, as in the proof of Theorem 4.3, that $|F| = p^h$.

4.7 We omit the proof of the general result here, as it will be given in Chapter 14. The points of P_2 are $\{000, 100\}$, $\{000, 010\}$, $\{000, 001\}$, $\{000, 110\}$, $\{000, 101\}$, $\{000, 011\}$, and $\{000, 111\}$. The lines are $\{000, 100, 010, 110\}$, $\{000, 100, 001, 101\}$, etc. That this 7-point plane is the same as that of Example 2.19 may be seen from Fig. 14.8, wherein a vector $\mathbf{x}$ stands for the point $\{\mathbf{0}, \mathbf{x}\}$.

Chapter 5

5.1 No; 24 is not a power of 2.

5.2 $[n, n-1, 2]$, $\left[\begin{array}{c|c} I_{n-1} & \begin{matrix}1\\1\\\vdots\\1\end{matrix}\end{array}\right]$.

5.3 We use Theorem 4.1.

$$\mathbf{x}, \mathbf{y} \in C \Rightarrow (\mathbf{x}+\mathbf{y})H^T = \mathbf{x}H^T + \mathbf{y}H^T = \mathbf{0}+\mathbf{0} = \mathbf{0}$$
$$\Rightarrow \mathbf{x}+\mathbf{y} \in C.$$

$$\mathbf{x} \in C \text{ and } a \in GF(q) \Rightarrow (a\mathbf{x})H^T = a(\mathbf{x}H^T) = a\mathbf{0} = \mathbf{0}$$
$$\Rightarrow a\mathbf{x} \in C.$$

5.4 If $\mathbf{x} = (x_1, \ldots, x_n) \in C$, let $\hat{\mathbf{x}} = (x_1, \ldots, x_n, \sum_{i=1}^n x_i)$, where $\sum x_i$ is calculated modulo 2. Then $\hat{C} = \{\hat{\mathbf{x}} \mid \mathbf{x} \in C\}$. Suppose

C is linear, so that $\mathbf{x}, \mathbf{y} \in C \Rightarrow \mathbf{x}+\mathbf{y} \in C$. Then

$$\hat{\mathbf{x}}, \hat{\mathbf{y}} \in \hat{C} \Rightarrow \hat{\mathbf{x}}+\hat{\mathbf{y}} = \left(x_1+y_1, \ldots, x_n+y_n, \sum x_i + \sum y_i\right)$$
$$= \left(x_1+y_1, \ldots, x_n+y_n, \sum (x_i+y_i)\right)$$
$$= \widehat{(\mathbf{x}+\mathbf{y})} \in \hat{C}.$$

So $\hat{C}$ is linear.

Adding an overall parity check to the code of Example 5.6(ii) gives an $[8, 4, 4]$-code with generator matrix

$$\begin{bmatrix} 1&0&0&0&1&0&1&1 \\ 0&1&0&0&1&1&1&0 \\ 0&0&1&0&1&1&0&1 \\ 0&0&0&1&0&1&1&1 \end{bmatrix}.$$

5.5 Let Ev and Od denote the subsets of C consisting of words of even and odd weights respectively. Suppose $Ev \neq C$. Then there exists a codeword, $\mathbf{y}$ say, of odd weight. Now the set $Ev + \mathbf{y} = \{\mathbf{x}+\mathbf{y} \mid \mathbf{x} \in Ev\}$ is contained in C (since C is linear). But all words in $Ev + \mathbf{y}$ are odd (via $w(\mathbf{x}+\mathbf{y}) = w(\mathbf{x}) + w(\mathbf{y}) - 2w(\mathbf{x} \cap \mathbf{y})$, cf. Lemma 2.6), and so we have $Ev + \mathbf{y} \subseteq Od$. Hence $|Ev| = |Ev + \mathbf{y}| \leq |Od|$. Also $Od + \mathbf{y} \subseteq Ev$ and so $|Od| \leq |Ev|$. Hence $|Ev| = |Od| = \frac{1}{2}|C|$.

5.6
$$C_1 = \begin{matrix} 0\,0\,0\,0\,0 \\ 1\,1\,1\,1\,0 \\ 0\,0\,1\,1\,1 \\ 1\,1\,0\,0\,1 \end{matrix} \qquad d(C_1) = \text{minimum non-zero weight} = 3$$

$$C_2 = \begin{matrix} 0\,0\,0\,0\,0\,0\,0\,0 \\ 1\,0\,0\,1\,1\,0\,1 = \mathbf{x}_1 \\ 0\,1\,0\,1\,0\,1\,1 = \mathbf{x}_2 \\ 0\,0\,1\,0\,1\,1\,1 = \mathbf{x}_3 \\ 1\,1\,0\,0\,1\,1\,0 = \mathbf{x}_1+\mathbf{x}_2 \\ 1\,0\,1\,1\,0\,1\,0 = \mathbf{x}_1+\mathbf{x}_3 \\ 0\,1\,1\,1\,1\,0\,0 = \mathbf{x}_2+\mathbf{x}_3 \\ 1\,1\,1\,0\,0\,0\,1 = \mathbf{x}_1+\mathbf{x}_2+\mathbf{x}_3 \end{matrix} \qquad d(C_2) = 4.$$

5.7 0 0 0 0 $d(C)$ = minimum non-zero weight = 3.
1 0 1 1 = $\mathbf{x}_1$
0 1 1 2 = $\mathbf{x}_2$ Since $9[1 + 2 \cdot 4] = 3^4$, the sphere-packing bound is attained and so C is perfect.
2 0 2 2 = $2\mathbf{x}_1$
0 2 2 1 = $2\mathbf{x}_2$
1 1 2 0 = $\mathbf{x}_1 + \mathbf{x}_2$
2 2 1 0 = $2\mathbf{x}_1 + 2\mathbf{x}_2$
1 2 0 2 = $\mathbf{x}_1 + 2\mathbf{x}_2$
2 1 0 1 = $2\mathbf{x}_1 + \mathbf{x}_2$

5.8 By Table 2.4, $A_2(8, 3) = 20$, $A_2(8, 4) = 16$, and $A_2(8, 5) = 4$. By Exercise 5.4(ii), there exists a linear $[8, 4, 4]$-code and so $B_2(8, 4) = 16$. There certainly exists also an $[8, 4, 3]$-code and so, since $B_2(8, 3)$ is a power of 2 and is ≤ 20, we have $B_2(8, 3) = 16$. The code constructed in Exercise 2.8 is linear and so $B_2(8, 5) = 4$.

5.9 $\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$ generates a $[3, 2, 2]$-code over $GF(q)$.

5.10 First get the required permutation of the rows of A by permuting the rows of G. The I_k part will have been disturbed but can be restored by a suitable permutation of the first k columns.

5.11 $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$.
No, Yes (by Exercise 5.10).

5.12 $(\mathbf{u} \mid \mathbf{u} + \mathbf{v}) + (\mathbf{u}' \mid \mathbf{u}' + \mathbf{v}') = (\mathbf{u} + \mathbf{u}' \mid \mathbf{u} + \mathbf{u}' + \mathbf{v} + \mathbf{v}') \in C_3$. Thus C_3 is linear. So $B_2(2d, d) = 4d$ by Exercises 2.19 and 2.23 since, at each step, C_1 and C_2 are linear.

Chapter 6

6.1 C_1: 00 01 10 11 C_2: 000 101 011 110
100 001 111 010

C_3:

00000	10110	01011	11101
10000	00110	11011	01101
01000	11110	00011	10101
00100	10010	01111	11001
00010	10100	01001	11111
00001	10111	01010	11100
11000	01110	10011	00101
10001	00111	11010	01100

(i) 11101, 01011
(ii) e.g. (a) 00000 received as 11000, (b) 00000 received as 10100.

6.2 $P_{\text{corr}}(C_1) = (1-p)^2 = 0.9801$
$P_{\text{corr}}(C_2) = (1-p)^3 + p(1-p)^2 = (1-p)^2 = 0.9801$
$P_{\text{corr}}(C_3) = (1-p)^3(1 - 2p^2 + 3p) \simeq 0.9992$
There is no point in using C_2 for error correction since P_{corr} is the same as for C_1, while C_2 takes 50% longer than C_1 to transmit messages. C_3 reduces the word error rate considerably.
$P_{\text{undetec}}(C_1) = 2p(1-p) + p^2 = 0.0199$
$P_{\text{undetec}}(C_2) = 3p^2(1-p) \simeq 0.000297$
$P_{\text{undetec}}(C_3) = 2p^3(1-p)^2 + (1-p)p^4 \simeq 0.00000197$.

6.3 (i) No, communication is impossible.
(ii) Yes, interchange all 0s and 1s in the received vector before decoding.

6.4 The coset leaders include all vectors of weight $\leq t$ and α_{t+1} vectors of weight $t+1$. So the probability that the error vector is *not* a coset leader is

$$\left[\binom{n}{t+1} - \alpha_{t+1}\right] p^{t+1}(1-p)^{n-t-1} + \text{terms involving } p^{t+2}$$

and higher powers. Hence

$$P_{\text{err}} \simeq \left[\binom{n}{t+1} - \alpha_{t+1}\right] p^{t+1} \text{ for small } p.$$

6.5 Straightforward calculation, with $A_3 = A_4 = 7$, $A_7 = 1$.

6.6 Since the code is perfect 3-error-correcting, we have

$$\alpha_0 = 1, \qquad \alpha_1 = 23, \qquad \alpha_2 = \binom{23}{2}, \qquad \alpha_3 = \binom{23}{3},$$

and

$$\alpha_i = 0 \text{ for } i \geqslant 4.$$

$$P_{\text{corr}} = (1-p)^{20}(1540p^3 + 210p^2 + 20p + 1) \simeq 0.99992$$

if $p = 0.01$.
So $P_{\text{err}} \simeq 0.000\,08$ [*Remark:* A fair approximation is obtained by using Exercise 6.4; namely $\binom{23}{4}10^{-8}$.]

6.7 Suppose $\mathbf{x} = x_1x_2\cdots x_n$ is sent and that the received vector is decoded as $\mathbf{x}' = x_1'x_2'\cdots x_n'$. Then

$$P_{\text{symb}} = \frac{1}{k}\sum_{j=1}^{k} \text{Prob}\,(x_j' \neq x_j)$$

$$= \frac{1}{k}\sum_{\mathbf{e}\in V(n,2)} f(\mathbf{e})\,\text{Prob}\,(\mathbf{e} \text{ is error vector}),$$

where $f(\mathbf{e})$ = number of incorrect information symbols after decoding if the error vector is $\mathbf{e}$, and so

$$P_{\text{symb}} = \frac{1}{k}\sum_{i=1}^{2^k} F_iP_i.$$

6.8
$$P_{\text{symb}} = \tfrac{1}{2}[P_2 + P_3 + 2P_4]$$
$$= \tfrac{1}{2}[\{2(1-p)^2p^2 + (1-p)p^3 + p^4\}$$
$$+ \{(1-p)^3p + (1-p)^2p^2 + 2(1-p)p^3\}$$
$$+ 2\{3(1-p)^2p^2 + (1-p)p^3\}].$$

6.9 Note that $P_{\text{err}} = \sum_{i=2}^{2^k} P_i$. Since $F_1 = 0$ and $1 \leqslant F_i \leqslant k$ for all $i \geqslant 2$, we have

$$\frac{1}{k}\sum_{i=2}^{2^k} P_i \leqslant \frac{1}{k}\sum_{i=1}^{2^k} F_iP_i \leqslant \sum_{i=2}^{2^k} P_i$$

and hence the result.

Chapter 7

7.1
$$\mathbf{u}\cdot\mathbf{v} = \sum_{i=1}^{n} u_iv_i = \sum_{i=1}^{n} v_iu_i = \mathbf{v}\cdot\mathbf{u}.$$

$$(\lambda\mathbf{u} + \mu\mathbf{v})\cdot\mathbf{w} = \sum_{i=1}^{n} (\lambda u_i + \mu v_i)w_i = \sum_{i=1}^{n} (\lambda u_iw_i + \mu v_iw_i)$$

$$= \lambda\sum_{i=1}^{n} u_iw_i + \mu\sum_{i=1}^{n} v_iw_i = \lambda\mathbf{u}\cdot\mathbf{w} + \mu\mathbf{v}\cdot\mathbf{w}.$$

7.2 The standard form generator matrix of E_n was found in Exercise 5.2. It follows from this and Theorem 7.6 that a generator matrix for $E_n^{\perp}$ is $[11\cdots1]$. So $E_n^{\perp} = \{00\cdots0, 11\cdots1\}$, which is the repetition code of length n.

7.3 Find the syndrome $S(\mathbf{y})$ of the received vector $\mathbf{y}$. If $S(\mathbf{y}) = \mathbf{0}$, then $\mathbf{y}$ is a codeword. If $S(\mathbf{y}) \neq \mathbf{0}$, then $\mathbf{y}$ is not a codeword and we have detected errors.

7.4 Suppose $\mathbf{x}$ is the codeword sent and $\mathbf{y} = \mathbf{x} + \mathbf{e}$ is received, where $\mathbf{e} = e_1e_2\cdots e_n$ is the error vector. Then $S(\mathbf{y}) = (\mathbf{x} + \mathbf{e})H^T = \mathbf{x}H^T + \mathbf{e}H^T = \mathbf{e}H^T$. So $S(\mathbf{y})^T = H\mathbf{e}^T = \sum_{j=1}^{n} e_j\mathbf{H}_j$, where $\mathbf{H}_j$ is the jth column of H.

7.5 Since the code is perfect, the coset leaders are precisely those vectors of weight $\leqslant 1$. G is in the form $[I_4 \mid A]$ and so

$$H = [-A^T \mid I_3] = \begin{bmatrix} 1110100 \\ 1101010 \\ 1011001 \end{bmatrix}.$$

We use this to construct the syndrome look-up table:

Syndrome	coset leader
000	0000000
111	1000000
110	0100000
101	0010000
011	0001000
100	0000100
010	0000010
001	0000001

$S(0000011) = 011$; decode as

$$0000011 - 0001000 = 0001011.$$

The other three vectors are decoded as 1111111, 0100110, 0010101.

7.6 (a) $\begin{bmatrix}1022\\0121\end{bmatrix}$ (b) $\begin{bmatrix}1110\\1201\end{bmatrix}$.

(c) A listing of the codewords reveals that $d(C) = 3$. So the 9 vectors of weight ≤ 1 are all coset leaders. Since the total number of coset leaders $= 3^4/3^2 = 9$, the vectors of weight ≤ 1 are *precisely* the coset leaders (in fact the code is perfect). The look-up table is now easily constructed, and the given vectors decoded as

0121, 1201, 2220.

7.7 0612960587.

7.8 Let C be a q-ary $(10, M, 3)$-code. Consider the M vectors of length 8 obtained by deleting the last two coordinates. These vectors must be distinct (or the corresponding vectors of C would be distance ≤ 2 apart). So $M \leq q^8$ (this is a particular case of the Singleton bound, Theorem 10.17). In particular, $A_{10}(10, 3) \leq 10^8$, $A_{11}(10, 3) \leq 11^8$. [*Remark:* The sphere-packing bound is not as good in these cases.] We have $A_{11}(10, 3) = 11^8$ because the linear $[10, 8]$-code over $GF(11)$ having

$$H = \begin{bmatrix}111\cdots1\\123\cdots10\end{bmatrix}$$

is an 11-ary $(10, 11^8, 3)$-code.

7.9 For example, $\mathbf{0}$ and 0505000000 are codewords only distance 2 apart.

7.10 Let $\mathbf{e}_j = 0\cdots01\cdots1$ (j1s). We require a code such that $\mathbf{e}_0, \mathbf{e}_1, \ldots, \mathbf{e}_7$ are all in different cosets (we could then decode via syndrome decoding with the $\mathbf{e}_i$s as coset leaders). This requires that $2^7/2^k \geq 8$, i.e. $k \leq 4$, and so the rate cannot be greater than $\frac{4}{7}$. To achieve rate $\frac{4}{7}$ we would need a 3×7 parity-check matrix H such that $\mathbf{e}_i H^T \neq \mathbf{e}_j H^T$ if $i \neq j$, i.e. such that $(\mathbf{e}_i - \mathbf{e}_j)H^T \neq \mathbf{0}$ for all $i \neq j$. Note that each $\mathbf{e}_i - \mathbf{e}_j$ is a vector of the form $0\cdots01\cdots10\cdots0$. A suitable H is

$$\begin{bmatrix}0001000\\0100010\\1010101\end{bmatrix}.$$

If $\mathbf{e}_i - \mathbf{e}_j$ is orthogonal to the first row of H, then all its 1s are to the left or to the right of centre. If also $\mathbf{e}_i - \mathbf{e}_j$ is orthogonal to the second row of H, then there can only be one 1, in one of the 1st, 3rd, 5th or 7th positions. But then $\mathbf{e}_i - \mathbf{e}_j$ is not orthogonal to the third row of H. (*Note:* a similar code of maximum possible rate may be constructed of *any* given length.)

7.11 If C is an $[n, k]$-code, then $\hat{C}$ is an $[n+1, k]$-code and so a parity-check matrix of $\hat{C}$ is an $(n+1-k) \times (n+1)$ matrix whose rows form a linearly independent set of codewords in $\hat{C}^\perp$. It is easily seen that $\bar{H}$ is such a matrix.

Chapter 8

8.1 $H = \begin{bmatrix}0000000011111111\\0001111000011111\\0110011001100111\\1010101010101010\end{bmatrix}$.

When $\mathbf{y}$ is received, calculate $\mathbf{y}H^T$; this gives the binary representation of the assumed error position. If two or more errors have occurred, then $\mathbf{y}$ will be decoded as a codeword different from that sent.

8.2 11100001, 01111000, at least two errors, 00110011.

8.3 From the standard form generator matrix (see Example 5.6(ii)), write down a parity-check matrix (via Theorem 7.6) and observe that its columns are the non-zero vectors of $V(3, 2)$.

8.4 For C, $\alpha_0 = 1$ and $\alpha_1 = n$, giving $P_{\text{corr}}(C) = (1-p)^{n-1}(1-p+np)$. Because every vector in $V(n, 2)$ has distance ≤ 1 from a codeword of C, it follows that every vector in $V(n+1, 2)$ has distance ≤ 2 from a codeword of $\hat{C}$. Consequently, the coset leaders for $\hat{C}$ all have weight ≤ 2

and so $\alpha_0 = 1$, $\alpha_1 = n+1$, $\alpha_2 = n$, which leads to $P_{\text{corr}}(\hat{C}) = (1-p)^{n-1}(1-p+np)$. [*Remark:* This result will be generalized to any perfect binary code in Exercise 9.1.]

8.5 (i) $\begin{bmatrix} 0&1&1&1&1&1&1&1 \\ 1&0&1&2&3&4&5&6 \end{bmatrix}$, 35234106, 10561360.

(ii) $\begin{bmatrix} 0&0&0&0&0&0&1&1&1&1&1&1&1&1&1&1 \\ 0&1&1&1&1&1&0&0&0&0&0&1&1&1&1 \\ 1&0&1&2&3&4&0&1&2&3&4&0&1&2&3 \end{bmatrix}$

$$\begin{bmatrix} 1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1 \\ 1&2&2&2&2&2&3&3&3&3&3&4&4&4&4&4 \\ 4&0&1&2&3&4&0&1&2&3&4&0&1&2&3&4 \end{bmatrix}$$

8.6 3.

8.7 $\begin{bmatrix} 0&4&3&1&0 \\ 3&2&0&0&1 \end{bmatrix}$, $\begin{bmatrix} 1&1&1&0&0&0 \\ 1&0&0&2&1&0 \\ 0&2&0&1&0&1 \end{bmatrix}$

[For the code C_2, a column operation (e.g. interchange of columns 3 and 4) is necessary during the reduction of G to a standard form of G'. So, after applying Theorem 7.6 to get a parity check matrix H' corresponding to G', the above column operation must be reversed in H' in order to get a parity-check matrix for the original code C_2.]

$d(C_1) = 2$, $d(C_2) = 3$.

(other answers possible)

8.8 For example,

$$H = \begin{bmatrix} 1&0&0&1&1&1 \\ 0&1&0&1&2&3 \\ 0&0&1&1&3&4 \end{bmatrix}$$

has the property that any three columns form a linearly independent subset of $V(3,5)$, and so H is the parity-check matrix of a $[6,3,4]$-code.

8.9 $R_r = \dfrac{k}{n} = (2^r - 1 - r)/(2^r - 1) = 1 - \dfrac{r}{2^r - 1} \to 1$ as $r \to \infty$.

8.10 As in solution to Exercise 7.8, $A_q(n,3) \leqslant q^{n-2}$. Now suppose q is a prime power. Then the bound is achieved for $n = q+1$ by Ham $(2,q)$ and for $n < q+1$ by shortenings of Ham $(2,q)$.

8.11 $f(t)$ = least value of M for which there exists a ternary code of length t with M codewords such that any vector in $V(t,3)$ has distance $\leqslant 1$ from at least one codeword. For such a code the spheres of radius 1 about codewords must 'cover' the whole space $V(t,3)$ and so a lower bound on M is given by

$$M(1+2t) \geqslant 3^t \qquad (1)$$

(This is the sphere-packing bound, but with the inequality reversed.)

(a) (i) If $t = (3^r - 1)/2$ then (1) gives $f(t) \geqslant 3^{t-r}$. The bound is achieved by a perfect $[t, t-r, 3]$-Hamming code over $GF(3)$. So, for $t = (3^r - 1)/2$, we have $f(t) = 3^{t-r}$.

(ii) Generating Ham $(2,3)$ by $\begin{bmatrix} 1&0&1&1 \\ 0&1&1&2 \end{bmatrix}$ and replacing '0' by 'X', we get the entry

X 1 X 2 X 1 2 1 2
X X 1 X 2 1 2 2 1
X 1 1 2 2 2 1 X X
X 1 2 2 1 X X 2 1

(b) The lower bound $f(5) \geqslant 23$ is given by (1). A crude upper bound is $f(5) \leqslant 27$. This is obtained by combining each of the 9 bets for $t = 4$ with each of the forecasts 1, 2, X for the 5th match. The surprising result proved by Kamps and van Lint is that one cannot do better than this.

8.12 Let C be an (n, M, d)-code with $M = A_q(n,d)$. Then there is no vector in $V(n,q)$ with distance $\geqslant d$ from all codewords in C. Thus the spheres of radius $d-1$ about codewords cover $V(n,q)$, whence the result. (The proof shows that a code meeting the lower bound may be constructed simply by starting with any word and then successively adding new words which have distance at least d from the words already chosen).

Chapter 9

9.1 Suppose C is a perfect t-error-correcting $[n,k]$-code, so that

$$\sum_{i=0}^{t} \binom{n}{i} = 2^{n-k}.$$

As in Exercise 8.4, for $\hat{C}$,

$$\alpha_i = \binom{n+1}{i} \text{ for } 0 \leqslant i \leqslant t,$$

and

$$\alpha_{t+1} = 2^{n+1-k} - \sum_{i=0}^{t} \binom{n+1}{i}$$

$$= 2 \cdot \sum_{i=0}^{t} \binom{n}{i} - \sum_{i=0}^{t} \binom{n}{i} - \sum_{i=1}^{t} \binom{n}{i-1} = \binom{n}{t}.$$

Hence

$$P_{\text{corr}}(\hat{C}) = \sum_{i=0}^{t} \binom{n+1}{i} p^i (1-p)^{n+1-i} + \binom{n}{t} p^{t+1} (1-p)^{n-t}$$

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

$$+ \sum_{i=1}^{t} \binom{n}{i-1} p^i (1-p)^{n+1-i} + \binom{n}{t} p^{t+1} (1-p)^{n-t}$$

$$= (1-p) P_{\text{corr}}(C)$$

$$+ \left(p\, P_{\text{corr}}(C) - \binom{n}{t} p^{t+1} (1-p)^{n-t} \right)$$

$$+ \binom{n}{t} p^{t+1} (1-p)^{n-t}$$

$$= P_{\text{corr}}(C).$$

9.2 It is easily checked that $\mathbf{u} \cdot \mathbf{v} = 0$ for any rows $\mathbf{u}$ and $\mathbf{v}$ of G. It follows that $G_{12}^{\perp} = G_{12}$. Now show that G_{12} has no codeword of weight $\leqslant 5$ by imitating the proof of Lemma 3 in the proof of Theorem 9.3.

9.3 If $H = [I_5 \mid A]$ has no 4 columns linearly dependent, then each column of A has at most one zero, and no two columns of A can have a zero entry in common (or their sum or difference would be a linear combination of two of the columns of I_5). The hint now follows easily. It then follows that in each of the undecided columns of A, two of the *s are 2s and the other * is a 1. The remaining columns may now be completed, one at a time, in a unique way (up to equivalence).

9.4 (a) Suppose $\mathbf{y}$ has weight 4. Since G_{23} is perfect, there is a unique codeword $\mathbf{x}$ such that $d(\mathbf{x}, \mathbf{y}) \leqslant 3$, and so $1 \leqslant w(\mathbf{x}) \leqslant 7$. But every non-zero codeword has weight $\geqslant 7$ and so $w(\mathbf{x}) = 7$, which implies that $\mathbf{x}$ covers $\mathbf{y}$. The uniqueness of $\mathbf{x}$ as a codeword having distance $\leqslant 3$ from $\mathbf{y}$ ensures that $\mathbf{x}$ is the only codeword of weight 7 which covers $\mathbf{y}$. Counting in two ways the number of pairs in the set $\{(\mathbf{x}, \mathbf{y}) \mid \mathbf{x}$ is a codeword of weight 7, $\mathbf{y}$ is a vector of weight 4, $\mathbf{x}$ covers $\mathbf{y}\}$ gives $A_7 \cdot \binom{7}{4} = \binom{23}{4} \cdot 1$, whence $A_7 = 253$.

(b) Let $P_1, \ldots, P_{23}$ be points and $B_1, \ldots, B_{253}$ be blocks, and define $P_i \in B_j$ if and only if the (i, j)th entry of M is 1.

9.5 (a) Straightforward generalization of the argument of Exercise 9.4.

(b) Let X be the set of codewords of weight $2t+1$ beginning with i 1s. Let Y be the set of vectors in $V(n, 2)$ of weight $t+1$ beginning with i 1s. As in the proof of Theorem 9.7, counting in two ways the number of pairs in the set $\{(\mathbf{x}, \mathbf{y}) \mid \mathbf{x} \in X, \mathbf{y} \in Y, \mathbf{x}$ covers $\mathbf{y}\}$ gives $\binom{n-i}{t+1-i} = \binom{2t+1-i}{t+1-i} \cdot |X|$, whence the result.

9.6 We must show that an arbitrary vector $\mathbf{y} = y_1 y_2 \cdots y_{24}$ of weight 5 in $V(24, 2)$ is covered by a unique codeword of weight 8 in G_{24}. Certainly there cannot be two such codewords or their distance apart would be $\leqslant 6$, a contradiction. If $y_{24} = 0$, then since G_{23} is perfect, G_{23} contains a codeword $\mathbf{x}$ having distance at most 3 from $y_1 y_2 \cdots y_{23}$. So $\mathbf{x}$ has weight 7 or 8; in either case $w(\hat{\mathbf{x}}) = 8$ and $\hat{\mathbf{x}}$ covers $\mathbf{y}$. If $y_{24} = 1$, then $y_1 \cdots y_{23}$ is covered by a unique codeword $\mathbf{x}$ of weight 7 in G_{23} and then $\hat{\mathbf{x}}$ covers $\mathbf{y}$.

9.7 By a now familiar argument, $A_3 \cdot \binom{3}{2} = \binom{2^r - 1}{2}$.

9.8 $A_5 \cdot \binom{5}{3} = \binom{11}{3} \cdot 2^3$.

9.9 (i) Assume $111111100 \cdots 0 \in G_{24}$. Let G be a generator matrix of G_{24}. Since $d(G_{24}) = 8$ and since G_{24} is self-dual, it follows by Theorem 8.4 that any 7

columns of G are linearly independent. In particular, the first 7 columns are linearly independent and so by elementary row operations, G may be transformed to a matrix having its first 7 columns as shown. Since $1111111100\cdots0$ is orthogonal to every row of G, the eighth column of G must also be as shown.

(ii) Let the rows of G be $\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_{12}$. The set of codewords with one of the given starts is given by adding to $\mathbf{0}$, or to one of $\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_7$, all vectors of the form $\sum_{i=8}^{12} \lambda_i \mathbf{r}_i$, $\lambda_i \in GF(2)$. So for each of the 8 starts, there are 2^5 codewords.

(iii) Immediate, since $d(G_{24}) = 8$, and any two of the chosen 256 codewords differ in at most 2 of the first 8 positions.

(iv) Immediate.

9.10 Shorten N_{15} thrice (cf. Exercise 2.2) to get a $(12, \geqslant 32, 5)$-code.

9.11 (i) Let the rows of G be $\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_6$. To show that G_2 generates an $[8, 5, 3]$-code, it is enough to show that if $\mathbf{x}$ is any non-zero codeword of C generated by $\mathbf{r}_2, \mathbf{r}_3, \ldots, \mathbf{r}_6$, then $\mathbf{x}$ has at least three 1s in the last 8 positions. If $\mathbf{x}$ had at most two 1s in the last 8 positions, then either $\mathbf{x}$ or $\mathbf{x}+\mathbf{r}_1$ would be a codeword of C having weight $\leqslant 4$, a contradiction.

(ii) If there existed a $[15, 8, 5]$-code, then it could be twice shortened to give a $[13, 6, 5]$-code, contrary to the result of part (i).

(iii) Not immediately, for in this case G_2 would generate a $[7, 4, 3]$-code, and a code with these parameters *does* exist. However, further considerations do lead to a contradiction; see, e.g., van Lint (1982), §4.4.

Chapter 10

10.1 Use Theorem 10.8 with $\mu = 1$, $\nu = 2$.

10.2 In Theorem 10.10, take

$$A_1 = B_1 = \begin{bmatrix} 0 & 1 & 2 \\ 1 & 2 & 0 \\ 2 & 0 & 1 \end{bmatrix}, \qquad A_2 = B_2 = \begin{bmatrix} 0 & 1 & 2 \\ 2 & 0 & 1 \\ 1 & 2 & 0 \end{bmatrix}.$$

10.3 Using Theorem 10.19, a set of three MOLS of order 4 is

$$A_1 = \begin{matrix} 0 & 1 & a & b \\ 1 & 0 & b & a \\ a & b & 0 & 1 \\ b & a & 1 & 0 \end{matrix} \qquad A_2 = \begin{matrix} 0 & a & b & 1 \\ 1 & b & a & 0 \\ a & 0 & 1 & b \\ b & 1 & 0 & a \end{matrix} \qquad A_3 = \begin{matrix} 0 & b & 1 & a \\ 1 & a & 0 & b \\ a & 1 & b & 0 \\ b & 0 & a & 1 \end{matrix}$$

10.4 Ham $(2, q)^\perp$ has generator matrix

$$\begin{bmatrix} 0 & 1 & 1 & 1 & \cdots & 1 \\ 1 & \lambda_0 & \lambda_1 & \lambda_2 & \cdots & \lambda_{q-1} \end{bmatrix},$$

where $GF(q) = \{\lambda_0, \lambda_1, \ldots, \lambda_{q-1}\}$. Clearly no non-zero linear combination of these two rows can have more than one zero and so Ham $(2, q)^\perp$ has minimum distance q. If we list the codewords generated by

$$\begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 2 & 3 & 4 \end{bmatrix}$$

and then apply Theorem 10.20, we get

$$A_1 = \begin{matrix} 0 & 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 & 0 \\ 2 & 3 & 4 & 0 & 1 \\ 3 & 4 & 0 & 1 & 2 \\ 4 & 0 & 1 & 2 & 3 \end{matrix} \qquad A_2 = \begin{matrix} 0 & 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 0 & 1 \\ 4 & 0 & 1 & 2 & 3 \\ 1 & 2 & 3 & 4 & 0 \\ 3 & 4 & 0 & 1 & 2 \end{matrix}$$

$$A_3 = \begin{matrix} 0 & 1 & 2 & 3 & 4 \\ 3 & 4 & 0 & 1 & 2 \\ 1 & 2 & 3 & 4 & 0 \\ 4 & 0 & 1 & 2 & 3 \\ 2 & 3 & 4 & 0 & 1 \end{matrix} \qquad A_4 = \begin{matrix} 0 & 1 & 2 & 3 & 4 \\ 4 & 0 & 1 & 2 & 3 \\ 3 & 4 & 0 & 1 & 2 \\ 2 & 3 & 4 & 0 & 1 \\ 1 & 2 & 3 & 4 & 0 \end{matrix}$$

10.5

n:	3	4	5	6	7	8	9	10	11	12
$f(n)$:	2	3	4	1	6	7	8	2–9	10	2–11

n:	13	14	15	16	17	18	19	20
$f(n)$:	12	2–13	2–14	15	16	2–17	18	3*–19

* Take three MOLS of order 4 and three MOLS of order 5 and generalize the construction of Theorem 10.10 to get 3 MOLS of order 20.

10.6 The existence of 3 MOLS of order 20 (see previous exercise) gives the existence of a $(5, 400, 4)$-code, by Theorem 10.20. Since this code achieves the Singleton bound, we have $A_{20}(5, 4) = 400$.

Chapter 11

11.1 0204006910.

11.2
$$G = \left[\begin{array}{c|cccc} & 4 & 7 & 9 & 1 \\ & 10 & 8 & 1 & 2 \\ I_6 & 9 & 7 & 7 & 9 \\ & 2 & 1 & 8 & 10 \\ & 1 & 9 & 7 & 4 \\ & 7 & 6 & 7 & 1 \end{array}\right]$$

11.3 0000001000, 1005000003.

11.4 Identify the letters A, B, ..., Z with the field elements $0, 1, \ldots, 25$ of GF(29). Let H be the parity-check matrix
$$\begin{bmatrix} 1 & 1 & 1 & & 1 \\ 1 & 2 & 3 & & 8 \\ 1 & 2^2 & 3^2 & \cdots & 8^2 \\ 1 & 2^3 & 3^3 & & 8^3 \end{bmatrix}$$
for an $(8, 29^4, 5)$-code over $GF(29)$. Let C be the 26-ary code obtained by taking only those codewords consisting of symbols $0, 1, \ldots, 25$, i.e.
$$C = \Big\{ x_1 x_2 \cdots x_8 \,|\, x_i \in \{0, 1, \ldots, 25\}, \sum_{i=1}^{8} i^j x_i \equiv 0 \pmod{29}, j = 0, 1, 2, 3 \Big\}.$$
A probabilistic estimate for the number of codewords in C is $29^4 \times \left(\frac{26}{29}\right)^8 \simeq 295,253$ (it happens that this is a remarkably good estimate).

Alternatively we could base our code on 26 of the elements of $GF(27)$. This would give us more codewords, but the arithmetic involved in the decoding would be less straightforward.

11.5
$$\sigma(\theta) = \prod_{i=1}^{e} (1 - X_i\theta) \Rightarrow \sigma'(\theta) = -\sum_{t=1}^{e} X_t \prod_{\substack{i=1 \\ i \neq t}}^{e} (1 - X_i\theta)$$
$$\Rightarrow \sigma'(X_j^{-1}) = -X_j \prod_{\substack{i=1 \\ i \neq j}}^{e} (1 - X_i X_j^{-1}).$$
The result now follows from equation (11.10).

11.6 $\mathrm{H} = \begin{bmatrix} 3 & 7 & 6 & 1 & 8 & 9 & 4 & 5 & 2 & 1 & 0 \\ 5 & 4 & 3 & 2 & 7 & 6 & 5 & 4 & 3 & 2 & 1 \end{bmatrix}$.
There exists a codeword $(x_1, \ldots, x_{11})$ of weight 2 with non-zero entries x_i and x_j if and only if $\mathbf{H}_i = -(x_j/x_i)\mathbf{H}_j$, where $\mathbf{H}_i$ denotes the ith column of H. In order to determine which columns of H are scalar multiples of others, calculate the ratios h_1/h_2 for each column
$$\begin{bmatrix} h_1 \\ h_2 \end{bmatrix}.$$
They are 5, 10, 2, 6, 9, 7, 3, 4, 8, 6, 0. It follows that a double-error vector will go undetected if and only if it is of the form $(0, 0, 0, \lambda, 0, 0, 0, 0, 0, -\lambda, 0)$ for some $\lambda \in \{1, 2, \ldots, 10\}$.

Chapter 12

12.1 (i) No, No (not linear), (ii) No, No, (iii) No, Yes, (iv) Yes, provided the alphabet is a field, (v) Yes, (vi) No, No, (vii) Yes.

12.2

$\cdot$	0	1	x	$1+x$
0	0	0	0	0
1	0	1	x	$1+x$
x	0	x	1	$1+x$
$1+x$	0	$1+x$	$1+x$	0

$1 + x$ has no inverse

12.3 Just imitate the proof of Theorem 3.5.

12.4 If $f(x)$ had an even number of non-zero coefficients, then we would have $f(1) = 0$ and so $x - 1$ would be a factor of $f(x)$.

12.5 Because $p(x) = f(x)g(x) \Rightarrow \deg p(x) = \deg f(x) + \deg g(x)$
$\Rightarrow$ either $\deg f(x) \leqslant \frac{1}{2}\deg p(x)$ or
$\deg g(x) \leqslant \frac{1}{2}\deg p(x)$.

12.6 x, $1+x$, $1+x+x^2$, $1+x+x^3$, $1+x^2+x^3$, $1+x+x^4$, $1+x^3+x^4$, $1+x+x^2+x^3+x^4$. (Using Lemma 12.3 and Exercise 12.4, it easily follows that the irreducible polynomials of degrees 2, 3 and 4 are precisely those with constant coefficient 1 and with an odd number of non-zero coefficients, with the exception of $(1+x+x^2)^2 = 1+x^2+x^4$). For example, $F[x]/(1+x+x^3)$ is a field of order 8.

12.7 (i) By Exercise 3.12, $(x^p - 1) = (x-1)^p$.
(ii) From Fermat's theorem (Exercise 3.8) and Lemma 12.3(i), it follows that $x^{p-1} = (x-1)(x-2)\cdots(x-(p-1))$.

12.8 By Lemma 12.3(i), $x^5 - 1 = (x-1)(x^4+x^3+x^2+x+1)$, and the second factor is irreducible by Exercise 12.6. So the only cyclic codes are $\{\mathbf{0}\}$, $\langle x-1\rangle$ (the even weight code), $\langle x^4+x^3+x^2+x+1\rangle$ (the repetition code), and the whole of $V(5,2)$.

12.9 Yes, $(x-1)g(x)$.

12.10 2^t. (In a factor of $x^n - 1$, each of the t distinct irreducible factors may or may not be present).

12.11 $\langle 1\rangle$ = whole space
$\langle x-1\rangle$ = even weight code E_7
$\langle x^3+x+1\rangle$, $\langle x^3+x^2+1\rangle$ } both are Hamming codes Ham (3, 2)
$\langle (x-1)(x^3+x+1)\rangle$, $\langle (x-1)(x^3+x^2+1)\rangle$ } both are even weight subcodes of Ham (3, 2) (alternatively, both are duals of Ham (3, 2))
$\langle (x^3+x+1)(x^3+x^2+1)\rangle$ = repetition code of length 7
$\langle x^7-1\rangle = \{\mathbf{0}\}$.

12.12 $x^8 - 1 = (x^4-1)(x^4+1) = (x-1)(x+1)(x^2+1)(x^2+x+2)(x^2+2x+2)$, 32.

12.13 Straightforward application of Theorem 12.15.

12.14 Not in general; Yes, $C^\perp$ is obtained from $\langle h(x)\rangle$ by writing the codewords backwards.

12.15 Let $g(x)$ be the generator polynomial of C. Then $g(x)$ is a divisor of $(x-1)(x^{n-1}+\cdots+x+1)$. If $g(x)$ is a multiple of $x-1$, then so is every codeword, and so every codeword has even weight. So if there exists a codeword of odd weight, then $x^{n-1}+\cdots+x+1$ must be a multiple of $g(x)$, i.e. $\mathbf{1} \in C$. The reverse implication is immediate since $w(\mathbf{1})$ is odd.

12.16 Let $\mathbf{g}_1, \ldots, \mathbf{g}_k$ denote the rows of G. Let $\vec{\mathbf{x}}$ denote a cyclic shift of $\mathbf{x}$. If $\mathbf{x} = \sum \lambda_i \mathbf{g}_i \in C$, then $\vec{\mathbf{x}} = \sum \lambda_i \vec{\mathbf{g}}_i \in C$.

12.17 Check that $2^0, 2^1, \ldots, 2^9$ are precisely the distinct non-zero elements of $GF(11)$. Hence the code of Example 7.12 is equivalent to the code C with parity-check matrix

$$\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 2^0 & 2^1 & 2^2 & \cdots & 2^9 \end{bmatrix}.$$

Now

$$(2^9, 2^0, 2^1, \ldots, 2^8) = 2^9(2^0, 2^1, \ldots, 2^9)$$

and so $C^\perp$ is cyclic by Exercise 12.16. Therefore C is cyclic by Theorem 12.15(ii). The result for Example 11.3 follows similarly.

12.18 The subcode D of G_{23} consisting of codewords of even weight is $\langle (x-1)g_1(x)\rangle$. Thus $D^\perp = \langle \bar{g}_2(x)\rangle = \langle g_1(x)\rangle$ and so $D \subseteq D^\perp$. Hence $\mathbf{u}\cdot\mathbf{v} = 0$ if $\mathbf{u}$ and $\mathbf{v}$ are codewords of even weight. Since $\mathbf{1} \in G_{23}$, any codeword of odd weight is of the form $\mathbf{u}+\mathbf{1}$ for some codeword $\mathbf{u}$ of even weight. If $\mathbf{u}+\mathbf{1}, \mathbf{v}+\mathbf{1}$ are codewords of odd weight, then $(\mathbf{u}+\mathbf{1})\cdot(\mathbf{v}+\mathbf{1}) = \mathbf{u}\cdot\mathbf{v} + \mathbf{1}\cdot\mathbf{v} + \mathbf{u}\cdot\mathbf{1} + \mathbf{1}\cdot\mathbf{1} = 0+0+0+1 = 1$. Also if $\mathbf{u}+\mathbf{1}$ has odd weight and $\mathbf{v}$ has even weight, then $(\mathbf{u}+\mathbf{1})\cdot\mathbf{v} = \mathbf{u}\cdot\mathbf{v} + \mathbf{1}\cdot\mathbf{v} = 0+0 = 0$. Now let $\mathbf{x}, \mathbf{y}$ be any codewords of G_{23} and let $\hat{\mathbf{x}}, \hat{\mathbf{y}}$ be the corresponding codewords of G_{24}. Then $\hat{\mathbf{x}}\cdot\hat{\mathbf{y}} = \mathbf{x}\cdot\mathbf{y} + x_{24}y_{24} = 0$, since $\mathbf{x}\cdot\mathbf{y} = 1 \Leftrightarrow \mathbf{x}, \mathbf{y}$ both have odd weight $\Leftrightarrow x_{24} = y_{24} = 1$. So $G_{24} \subseteq G_{24}^\perp$ and since $\dim(G_{24}) = \dim(G_{24}^\perp) = 12$, it follows that $G_{24} = G_{24}^\perp$.

12.19 x^4+x+1 is a generator polynomial for Ham (4, 2). Dividing $x^{15}-1$ by x^4+x+1 (e.g. by long division) gives $h(x) = x^{11}+x^8+x^7+x^5+x^3+x^2+x+1$.

12.20 Ham $(r, 2)$ is a $[2^r-1, 2^r-r-1, 3]$-code. By Exercise 12.9, $\langle (x-1)g(x)\rangle$ is the subcode of codewords of even

weight. This subcode must have dimension $2^r - r - 2$ and minimum distance 4.

12.21 It is enough to show that no vector of the form

$$(x^i + x^{i+1}) + (x^j + x^{j+1}) = (x+1)(x^i + x^j)$$

is a codeword of $\langle (x+1)g(x) \rangle$ (then all vectors of the form 0, x^i, and $x^i + x^{i+1}$ will be coset leaders). But $(x+1)(x^i + x^j)$ is a codeword $\Rightarrow (x+1)(x^i + x^j)$ is a multiple of $(x+1)g(x) \Rightarrow x^i + x^j$ is a multiple of $g(x) \Rightarrow x^i + x^j \in \langle g(x) \rangle$, contradicting $d(\langle g(x) \rangle) = 3$.

12.22 (van Lint 1982, solution to Exercise 6.11.7). Show that every non-zero codeword of C has exactly one zero entry. Show also that there is exactly one codeword $\mathbf{c} = c_0 c_1 \cdots c_q$ such that $c_0 = c_{(q+1)/2} = 1$ [Consider the q^2 ordered pairs $(c_0, c_{(q+1)/2})$ as $\mathbf{c}$ runs over all codewords of C]. If C were cyclic, then a cyclic shift of $\mathbf{c}$ through $(q+1)/2$ positions would yield the same codeword $\mathbf{c}$, but this is not possible if $\mathbf{c}$ contains only one zero entry. Thus C is not cyclic and so Ham $(2, q)$, being the dual code of C, is not cyclic by Theorem 12.15(ii).

Chapter 13

13.1 The mapping $\mathbf{x} \to \mathbf{x} + \mathbf{1}$ gives a one-to-one correspondence between the set of codewords of weight i and the set of codewords of weight $n - i$.

13.2 (b) $C^\perp$ is generated by

$$\begin{bmatrix} 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 \end{bmatrix},$$

and so $C^\perp = \{00000, 10010, 11101, 01111\}$. Hence $W_{C^\perp}(z) = 1 + z^2 + 2z^4$. So

$$W_C(z) = \tfrac{1}{4}(1+z)^5 W_{C^\perp}\left(\frac{1-z}{1+z}\right)$$
$$= \tfrac{1}{4}[(1+z)^5 + (1+z)^3(1-z)^2 + 2(1+z)(1-z)^4]$$
$$= 1 + 3z^2 + 3z^3 + z^5.$$

13.3 $C^\perp$ is generated by

$$\begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix},$$

and so $W_{C^\perp}(z) = 1 + 3z^6$. Hence $W_C(z) = \frac{1}{4}[(1+z)^9 + 3(1+z)^3(1-z)^6]$, whence $A_0 = 1$, $A_1 = 0$, $A_2 = \frac{1}{4}[36 + 3(3 + 15 - 18)] = 9$, $A_3 = 27$. The sum of all the rows of the generator matrix of C is $\mathbf{1}$. By Exercise 13.1, $A_i = A_{9-i}$, and so $2(A_0 + A_1 + A_2 + A_3 + A_4) = 2^7$, which gives $A_4 = 27$. Hence

$$W_C(z) = 1 + 9z^2 + 27z^3 + 27z^4 + 27z^5 + 27z^6 + 9z^7 + z^9.$$

13.4 Adding an overall parity check increases each odd weight by 1 and leaves each even weight unchanged. So $W_{\hat{C}}(z) = 1 + 14z^4 + z^8$.

13.5 Let C be Ham $(r, 2)$. Then by Theorem 13.10, $W_{C^\perp}(z) = 1 + (2^r - 1)z^{2^{r-1}} = 1 + nz^{(n+1)/2}$. So

$$W_C(z) = \frac{1}{2^r}[(1+z)^n + n(1-z)^{(n+1)/2}(1+z)^{(n-1)/2}]$$
$$= \frac{1}{2^r}[(1+z)^n + n(1-z^2)^{(n-1)/2}(1-z)].$$

13.6 $W_C(z) = \frac{1}{16}[(1+z)^{15} + 15(1-z^2)^7(1-z)]$. (1)
$A_0 = 1$, $A_1 = A_2 = 0$ (either from (1) or because we know $d(C) = 3$), $A_3 = 35$, $A_4 = 105$.

13.7 The coefficient of z^i in the right-hand side is $\frac{1}{2}A_i(1 + (-1)^i) = A_i$ if i is even, 0 if i is odd.

13.8 If $W_C(z) = \sum A_i z^i$, then

$$W_{\hat{C}}(z) = \sum_{i \text{ even}} (A_i + A_{i-1}) z^i$$
$$= \sum_{i \text{ even}} A_i z^i + z \sum_{j \text{ odd}} A_j z^j$$
$$= \tfrac{1}{2}[W_C(z) + W_C(-z)] + \tfrac{1}{2}z[W_C(z) - W_C(-z)].$$

13.9 From equation (13.12),

$$P_{\text{undetec}}(C) = (1-p)^n \frac{1}{2^{n-k}} \left(1 + \frac{p}{1-p}\right)^n$$
$$\times W_{C^\perp}\left(\left(1 - \frac{p}{1-p}\right) \Big/ \left(1 + \frac{p}{1-p}\right)\right)$$
$$- (1-p)^n$$
$$= \frac{1}{2^{n-k}} W_{C^\perp}(1-2p) - (1-p)^n.$$

13.10 By Lemmas 1, 3, and 4 in the proof of Theorem 9.3, G_{24} is self-dual, $A_i \neq 0$ only if i is divisible by 4, and $A_4 = 0$. Since $\mathbf{1} \in G_{24}$, it follows from Exercise 13.1 that $A_{20} = 0$ and that $A_{16} = A_8$. So $W_C(z) = 1 + A_8 z^8 + A_{12} z^{12} + A_8 z^{16} + z^{24}$. Applying the MacWilliams identity and equating coefficients of $W_C(z)$ and $W_{C^\perp}(z)$ (since C is self-dual) gives: $2 + 2A_8 + A_{12} = 2^{12}$ (constant coefficients) $0 = 0$ (coefficients of z) and $138 + 10A_8 - 3A_{12} = 0$ (coefficients of z^2). Solving these gives $A_8 = 759$, $A_{12} = 2576$.

13.11 G_{24} is self-dual by Exercise 12.18. By Lemma 12.19, codewords of G_{23} of even weight have weight divisible by 4. Since $\mathbf{1}\,(= g_1(x)g_2(x)) \in G_{23}$, it follows by Exercise 13.1 that any odd weight of a codeword of G_{23} is congruent to 3 (mod 4). Consequently, all codewords of G_{24} have weight divisible by 4. Also $A_4 = 0$, since $d(G_{24}) = 8$. The result now follows exactly as in Exercise 13.10.

13.12 By Exercise 13.10 (or 13.11) the only A_is in $W_{G_{23}}(z)$ which can be non-zero are A_0, A_7, A_8, A_{11}, A_{12}, A_{15}, A_{16} and A_{23}. Also $A_7 + A_8 = 759$ and $A_{11} + A_{12} = 2576$. By Exercise 9.4(a), $A_7 = 253$ and so $A_8 = 506$. Since $\mathbf{1} \in G_{23}$, we have $A_{11} = A_{12} = 1288$, $A_{15} = 506$, and $A_{16} = 253$. So

$$W_{G_{23}}(z) = 1 + 253z^7 + 506z^8 + 1288z^{11} + 1288z^{12} + 506z^{15} + 253z^{16} + z^{23}.$$

Chapter 14

14.1 No; Exercises 9.9 and 9.11 show that $A_2(15, 5) \geqslant 256$, $B_2(15, 5) = 128$.

14.2 (i) $V(n, q)$ is an $[n, n, 1]$-code.

(ii) $C = \{x_1 x_2 \cdots x_n \mid x_1 + x_2 + \cdots + x_n = 0\}$ is an $[n, n-1, 2]$-code. Since there cannot exist an $[n, n, 2]$-code, we have $B_q(n, 2) = q^{n-1}$.

14.3 By Theorem 14.4, there exists an $[n, n-r, 3]$-code over

$$GF(q) \Leftrightarrow n \leqslant (q^r - 1)/(q-1)$$
$$\Leftrightarrow r \geqslant \log_q \{n(q-1) + 1\}$$
$$\Leftrightarrow n - r \leqslant n - \log_q \{n(q-1) + 1\}.$$

So $B_q(n, 3) = q^{\lfloor n - \log_q \{n(q-1)+1\} \rfloor}$.

14.4 Let t be the number of planes in which a given line L lies. Counting in two ways the number of members of the set $\{(P, \pi) \mid P$ is a point not on L, π is a plane containing both L and $P\}$ gives $q^3 + q^2 + q + 1 - (q+1) = t[q^2 + q + 1 - (q+1)]$, whence $t = q + 1$.

14.5 The Golay code G_{11} is a ternary $[11, 6, 5]$-code, showing that $\max_4(5, 3) \geqslant 11$. If $\max_4(5, 3)$ were $\geqslant 12$, then there would exist a ternary $[12, 7, 5]$-code, contradicting the sphere-packing bound.

14.6 Use Theorem 14.18. Since 2 is a non-square in $GF(5)$, the 4×26 matrix whose columns are $(0, 0, 0, 1)^T$ and $(x, y, 1, x^2 - 2y^2)^T$, for $(x, y) \in V(2, 5)$, is the parity-check matrix of a $[26, 22, 4]$-code.

14.7 (i) By Theorem 14.16, a plane can contain $q + 2$ points of a cap.

(ii) By Exercise 3.14, if q is even, then every element of $GF(q)$ is a square. [*Remark:* a version of Theorem 14.18 does hold for q even, with an elliptic quadric specified in a different way].

14.8 Let H be the parity-check matrix whose columns form the $(q^2 + 1)$-cap defined in (14.19). Label the column $(0001)^T$ by ∞ and each column $(x, y, 1, x^2 - by^2)^T$ by (x, y). A decoding algorithm is the following. Calculate the syndrome $\mathbf{s} = \mathbf{y}H^T = s_1 s_2 s_3 s_4$. If $\mathbf{s} = \mathbf{0}$, assume no errors. If $\mathbf{s} \neq \mathbf{0}$, calculate $\theta = s_3 s_4 - s_1^2 + b s_2^2$. If $\theta = 0$ and $s_3 = 0$, assume an error of magnitude s_4 in position ∞. If $\theta = 0$ and $s_3 \neq 0$, assume an error of magnitude s_3 in position $(s_1/s_3, s_2/s_3)$. If $\theta \neq 0$, then there are $\geqslant 2$ errors.

14.9 If $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{10}\}$ is a 10-cap in $PG(3, 3)$ then the set

$$\{(\mathbf{x}_1 \mid 0), (\mathbf{x}_2 \mid 0), \ldots, (\mathbf{x}_{10} \mid 0), (\mathbf{x}_1 \mid 1), (\mathbf{x}_2 \mid 1), \ldots, (\mathbf{x}_{10} \mid 1)\}$$

is a 20-cap in $PG(4, 3)$.

14.10 For a given t-space, the number of ways of choosing an extra point of $PG(m,q)$ to generate a $(t+1)$-space is

$$\frac{q^{m+1}-1}{q-1}-\frac{q^{t+1}-1}{q-1}.$$

Many of these extra points generate the same $(t+1)$-space, and so we must divide by

$$\frac{q^{t+2}-1}{q-1}-\frac{q^{t+1}-1}{q-1},$$

the number of points in such a $(t+1)$-space not lying in the given t-space.

(i) 40, (ii) 13, (iii) 4.

14.11 (Bruen and Hirschfeld 1978). Suppose K is a cap in $PG(5,3)$. We shall show that $|K| \leqslant 56$. We may assume some plane π meets K in four points, for otherwise $|K| \leqslant 42$ (two points on some line L plus at most one further point on each of the 40 planes through L). Similarly, we may assume some 3-space contains at least 8 points of K, for otherwise $|K| \leqslant 4+3\cdot 13 = 43$. Finally, since $\max_3(5,3) = 20$, we have $|K| \leqslant 8+4(20-8) = 56$.

14.12 $B_3(n,4) = 3^{n-4}$ for $5 \leqslant n \leqslant 10$, 3^{n-5} for $11 \leqslant n \leqslant 20$, 3^{n-6} for $21 \leqslant n \leqslant 56$, 3^{n-7} for $57 \leqslant n \leqslant 112$. (It is not known whether $B_3(113,4) = 3^{106}$ or 3^{105}.)

Chapter 15

15.1 $[I_3 \mid A]$, $[I_5 \mid A^T]$.

15.2 Suppose, for a contradiction, that $\max_{q+2-r}(q+2-r,q) \geqslant q+2$. Then there exists a $[q+2, r, q+3-r]$-code whose dual is a $[q+2, q+2-r, r+1]$-code, contradicting $\max_r(r,q) = q+1$.

15.3 $\max_{q-1}(q-1,q) \geqslant q+2$ by Corollary 15.9. If there existed a $[q+3, 4, q]$-code over $GF(q)$, then its dual would be a $[q+3, q-1, 5]$-code, contrary to $\max_4(4,q) = q+1$.

15.4
$$H = \left[\begin{array}{c|ccc} & 1 & 1 & 1 \\ & 1 & a_2 & a_2^2 \\ I_7 & 1 & a_3 & a_3^2 \\ & \vdots & \vdots & \vdots \\ & 1 & a_6 & a_6^2 \end{array}\right].$$

15.5 Let $H = [A \mid I]$ be a standard form parity-check matrix for an $[n, n-r, r+1]$-code with $n = \max_r(r,q)$. Deleting the last row and last column of H gives a matrix whose columns form an $(n-1, r-1)$-set in $V(r-1,q)$ and so $n-1 \leqslant \max_{r-1}(r-1,q)$.

15.6 Let C be an $[8,3,6]$-code over $GF(7)$. By Corollary 15.7, $C^\perp$ is an $[8,5,4]$-code. Let $W_C(z) = \sum A_i z^i$ and $W_{C^\perp}(z) = \sum B_i z^i$. By Theorem 13.6,

$$7^3\Big(1+\sum_{i=4}^{8} B_i z^i\Big) = (1+6z)^8 + A_6(1-z)^6(1+6z)^2 + A_7(1-z)^7(1+6z) + A_8(1-z)^8 \quad (1)$$

Equating coefficients of 1, z and z^2 and solving for A_6, A_7 and A_8 gives $W_C(z) = 1 + 168z^6 + 48z^7 + 126z^8$. $W_{C^\perp}(z)$ is now easily obtained directly from (1).

15.7 For $2 \leqslant k \leqslant 11$,

$$\begin{bmatrix} 1 & 1 & \cdots & 1 & 1 & 0 \\ 1 & 2 & \cdots & 10 & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ 1 & 2^{k-1} & \cdots & 10^{k-1} & 0 & 1 \end{bmatrix}$$

generates a $[12, k, 13-k]$-code.

For $k \geqslant 11$, $\left[\begin{array}{c|c} I_k & \begin{matrix}1\\1\\\vdots\\1\end{matrix} \end{array}\right]$ generates a $[k+1, k, 2]$-code.

Chapter 16

16.1 If $x \not\equiv 0 \pmod p$, then $x^{r(p-1)(q-1)+1} = (x^{p-1})^{r(q-1)}x \equiv x \pmod p$ by Fermat's theorem. If $x \equiv 0 \pmod p$, then $x^{r(p-1)(q-1)+1} \equiv x \pmod p$ holds trivially. So p is a factor

of $x^{r(p-1)(q-1)+1} - x$ for any integer x. Similarly q is also a factor for any integer x. Since p and q are distinct prime numbers, pq is a factor of $x^{r(p-1)(q-1)+1} - x$ for any x.

16.2 LEAVING TOMORROW.

16.3 When the subscriber R (of the text) has encrypted a message he is to send to S (using S's encryption algorithm) he signs it with a further message z which he sends in the form $z^t \pmod n$ (i.e. via R's own *de*crypting algorithm). The receiver S verifies the signature by calculating $(z^t)^s \equiv z \pmod n$. Only R could have sent the message, since only R knows t.

16.4 B, C and D are uniquely decodable. B and C are prefix-free. Average word-lengths of B, C and D are 2, 1.75 and 1.875, respectively. [*Remark:* It is a consequence of Shannon's 'source coding theorem' (see, e.g., Jones 1979) that the 'source entropy', $-\sum_{i=1}^{4} p_i \log_2 p_i$ (=1.75 here), gives the smallest possible average word length. So the above code C here is best possible.]

16.5 THE END.

Bibliography

At the end of each entry the number in square brackets gives the chapter which refers to this entry.

Anderson, I. (1974). *A first course in combinatorial mathematics*. Clarendon Press, Oxford. [2, 16]

Assmus, E. F. and Mattson, H. F. (1967). On tactical configurations and error-correcting codes. *J. Comb. Theory* **2,** 243–57. [9]

—— (1969). New 5-designs. *J. Comb. Theory* **6,** 122–51. [9]

—— (1974). Coding and combinatorics. *SIAM Review* **16,** 349–88. [9]

Barlotti, A. (1955). Un estensione del teorema di Segre-Kustaanheimo. *Boll. Un. Mat. Ital.* **10,** 96–8. [14]

Beker, H. and Piper, F. (1982). *Cipher Systems*. Van Nostrand Reinhold, London. [16]

Berlekamp, E. R. (1968). *Algebraic coding theory*. McGraw-Hill, New York. [Pref., 11]

—— (1972). Decoding the Golay code. *JPL Technical Report* 32–1526, Vol. IX, 81–5. [9]

Best, M. R. (1980). Binary codes with a minimum distance of four. *IEEE Trans. Info. Theory* **26,** 738–42. [16]

—— (1982). A contribution to the nonexistence of perfect codes. Ph.D. dissertation, University of Amsterdam. [9]

—— (1983). Perfect codes hardly exist. *IEEE Trans. Info. Theory* **29,** 349–51. [9]

—— and Brouwer, A. E. (1977). The triply shortened Hamming code is optimal. *Discrete Math.* **17,** 235–45. [8]

Best, M. R., Brouwer, A. E., MacWilliams, F. J., Odlyzko, A. M., and Sloane, N. J. A. (1978). Bounds for binary codes of length less than 25. *IEEE Trans. Info. Theory* **24,** 81–92. [16]

Blahut, R. E. (1983). *Theory and practice of error control codes*. Addison-Wesley, Reading, Mass. [Pref., 11, 12, 16]

Blake, I. F. and Mullin, R. C. (1976). *An introduction to algebraic and combinatorial coding theory*. Academic Press, New York. [Pref.]

Blokhuis, A. and Lam, C. W. H. (1984). More coverings by rook domains. *J. Comb. Theory*, Ser. A **36,** 240–4. [8]

Bose, R. C. (1947). Mathematical theory of the symmetrical factorial design. *Sankhyā* **8,** 107–166. [14]

—— and Ray-Chaudhuri, D. K. (1960). On a class of error-correcting binary group codes. *Info. and Control* **3,** 68–79. [11, 12]
——, Shrikhande, S. S., and Parker, E. T. (1960). Further results on the construction of mutually orthogonal Latin squares and the falsity of Euler's conjecture. *Canad. J. Math.* **12,** 189–203. [10]
Brinn, L. W. (1984). Algebraic coding theory in the undergraduate curriculum. *American Math. Monthly,* 509–13. [Pref.]
Brown, D. A. H. (1974). Some error correcting codes for certain transposition and transcription errors in decimal integers. *Computer Journal* **17,** 9–12. [11]
Bruen, A. A. and Hirschfeld, J. W. P. (1978). Application of line geometry over finite fields. II. The Hermitian surface. *Geom. Dedicata* **7,** 333–53. [14]
Bush, K. A. (1952). Orthogonal arrays of index unity. *Ann. Math. Stat.* **23,** 426–34. [15]
Cameron, P. J. and van Lint, J. H. (1980). *Graphs, codes and designs.* London Math. Soc. Lecture Note Series, Vol. 43. Cambridge Univ. Press, Cambridge. [Pref.]
Casse, L. R. A. (1969). A solution to Beniamino Segre's 'Problem $I_{r,q}$' for q even. *Atti. Accad. Naz. Lincei Rend.* **46,** 13–20. [15]
Delsarte, P. and Goethals, J.-M. (1975). Unrestricted codes with the Golay parameters are unique. *Discrete Math.* **12,** 211–24. [9]
Dénes, J. and Keedwell, A. D. (1974). *Latin squares and their applications.* Academic Press, New York. [10]
Diffie, W. and Hellman, M. E. (1976). New directions in cryptography. *IEEE Trans. Info. Theory* **22,** 644–54. [16]
Dornhoff, L. L. and Hohn, F. E. (1978). *Applied Modern Algebra.* Macmillan, New York. [16]
Fenton, N. E. and Vámos, P. (1982). Matroid interpretation of maximal k-arcs in projective spaces. *Rend. Mat.* (7) **2,** 573–80. [14, 15]
Fernandes, H. and Rechtschaffen, E. (1983). The football pool problem for 7 and 8 matches. *J. Comb. Theory,* Series A **35,** 109–14. [8]
Games, R. A. (1983). The packing problem for projective geometries over $GF(3)$ with dimension greater than five. *J. Comb. Theory,* Series A **35,** 126–44. [14]
Gardner, M. (1977). Mathematical games. *Scientific American,* August, 120–4. [16]
Gibson, I. B. and Blake, I. F. (1978). Decoding the binary Golay code with miracle octad generators. *IEEE Trans. Info. Theory* **24,** 261–4. [9]
Gilbert, E. N. (1952). A comparison of signalling alphabets. *Bell Syst. Tech. J.* **31,** 504–22. [8]
Goethals, J.-M. (1971). On the Golay perfect binary code. *J. Comb. Theory* **11,** 178–86. [9]
—— (1977). The extended Nadler code is unique. *IEEE Trans. Info. Theory* **23,** 132–5. [9]

Golay, M. J. E. (1949). Notes on digital coding. *Proc. IEEE* **37,** 657. [8, 9]
—— (1954). Binary coding. *Trans IRE PGIT* **4,** 23–8. [16]
—— (1958). Notes on the penny-weighing problem, lossless symbol coding with nonprimes, etc. *IEEE Trans. Info. Theory* **4,** 103–9. [9]
Golomb, S. W. and Posner, E. C. (1964). Rook domains, Latin squares, affine planes, and error-distribution codes. *IEEE Trans. Info. Theory* **10,** 196–208. [9]
Goppa, V. D. (1970). A new class of linear error-correcting codes. *Problems of Info. Transmission* **6** (3), 207–12. [11]
Hall, M. (1980). *Combinatorial theory.* Wiley, New York. [2]
Hamming, R. W. (1950). Error detecting and error correcting codes. *Bell Syst. Tech. J.* **29,** 147–60. [8]
—— (1980). *Coding and information theory.* Prentice-Hall, New Jersey. [16]
Hardy, G. H. (1940). *A mathematician's apology.* Cambridge University Press. [11]
Helgert, H. J. and Stinaff, R. D. (1973). Minimum distance bounds for binary linear codes. *IEEE Trans. Info. Theory* **19,** 344–56. [14]
Hill, R. (1973). On the largest size of cap in $S_{5,3}$. *Atti Accad. Naz. Lincei Rend.* **54,** 378–84. [14]
—— (1978). Caps and codes. *Discrete Math.* **22,** 111–37. [14]
Hirschfeld, J. W. P. (1979). *Projective geometries over finite fields.* Oxford University Press. [14]
—— (1983). Maximum sets in finite projective spaces. In *Surveys in combinatorics,* LMS Lecture Note Series 82, edited by E. K. Lloyd. Cambridge University Press, 55–76. [14]
Hocquenghem, A. (1959). Codes correcteurs d'erreurs. *Chiffres* (Paris) **2,** 147–58. [11]
Jones, D. S. (1979). *Elementary information theory.* Clarendon Press, Oxford. [16]
Jurick, R. R. (1968). An algorithm for determining the largest maximally independent set of vectors from an r-dimensional space over a Galois field of n elements. Tech. Rep. ASD–TR–68–40, Air Force Systems Command, Wright–Patterson Air Force Base, Ohio. [15]
Kamps, H. J. L. and van Lint, J. H. (1967). The football pool problem for 5 matches. *J. Comb. Theory* **3,** 315–25. [8]
Levenshtein, V. I. (1964). The application of Hadamard matrices to a problem in coding. *Problems of Cybernetics* **5,** 166–84. [16]
Levinson, N. (1970). Coding theory: a counterexample to G. H. Hardy's conception of applied mathematics. *Amer. Math. Monthly* **77,** 249–58. [11]
Lidl, R. and Niederreiter, H. (1983). *Finite fields.* Addison-Wesley, and (1984) Cambridge University Press. [3]
Lin, S. and Costello, D. J. (1983). *Error control coding: fundamentals and applications.* Prentice-Hall, New Jersey. [Pref., 12]

Lindstrom, B. (1969). On group and nongroup perfect codes in q symbols. *Math. Scand.* **25,** 149–58. [9]
van Lint, J. H. (1975). A survey of perfect codes. *Rocky Mountain J. of Mathematics* **5,** 199–224. [9]
—— (1982). *Introduction to coding theory*. Springer-Verlag, New York. [Pref., 6, 9, 16]
Lloyd, S. P. (1957). Binary block coding. *Bell Syst. Tech. J.* **36,** 517–35. [9]
McEliece, R. J. (1977). *The theory of information and coding*. Addison-Wesley, Reading, Mass. [Pref., 6, 16]
McEliece, R. J., Rodemich, E. R., Rumsey, H. and Welch, L. R. (1977). New upper bounds on the rate of a code via the Delsarte–MacWilliams inequalities. *IEEE Trans. Info. Theory* **23,** 157–66. [16]
Mackenzie, C. and Seberry, J. (1984). Maximal ternary codes and Plotkin's bound. *Ars Combinatoria* **17A,** 251–70. [2]
MacWilliams, F. J. (1963). A theorem on the distribution of weights in a systematic code. *Bell Syst. Tech. J.* **42,** 79–94. [13]
—— and Sloane, N. J. A. (1977). *The theory of error-correcting codes*. North-Holland, Amsterdam. [Pref., 2, 8, 9, 11–15, 16]
Magliveras, S. S. and Leavitt, D. W. (1983). Simple six-designs exist. *Congressus Numerantium* **40,** 195–205. [9]
Maneri, C. and Silverman, R. (1966). A vector space packing problem. *J. of Algebra* **4,** 321–30. [15]
Massey, J. L. (1969). Shift-register synthesis and BCH decoding. *IEEE Trans. Info. Theory* **15,** 122–27. [11]
Nadler, M. (1962). A 32-point $n = 12$, $d = 5$ code. *IEEE Trans. Info. Theory* **8,** 58. [9]
Nordstrom, A. W. and Robinson, J. P. (1967). An optimum non-linear code. *Info. and Control* **11,** 613–16. [2]
Pellegrino, G. (1970). Sul massimo ordine delle calotte in $S_{4,3}$. *Matematiche* (Catania) **25,** 1–9. [14]
Peterson, W. W. and Weldon, E. J. (1972). *Error-correcting codes*, 2nd ed. MIT Press, Cambridge, Mass. [Pref., 11, 16]
Phelps, K. T. (1983). A combinatorial construction of perfect codes. *SIAM J. Alg. Disc. Math.* **4,** 398–403. [9]
Pless, V. (1968). On the uniqueness of the Golay codes. *J. Comb. Theory* **5,** 215–28. [9]
—— (1982). *Introduction to the theory of error-correcting codes*. Wiley, New York. [Pref., 9]
Plotkin, M. (1960). Binary codes with specified minimum distance. *IEEE Trans. Info. Theory* **6,** 445–450. [2]
Prange, E. (1957). Cyclic error-correcting codes in two symbols. Technical Note TN-57-103, Air Force Cambridge Research Labs., Bedford, Mass. [12]
Qvist, B. (1952). Some remarks concerning curves of the second degree in a finite plane. *Ann. Acad. Sci. Fenn.* Ser. A, no. 134. [14]

Ramanujan, S. (1912). Note on a set of simultaneous equations. *J. Indian Math. Soc.* **4,** 94–6. [11]
Rivest, R. L., Shamir, A., and Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. *Comm. ACM* **21,** 120–6. [16]
Ryser, H. J. (1963). *Combinatorial mathematics*. Carus Monograph 14, Math. Assoc. America. [10]
Schönheim, J. (1968). On linear and non-linear single-error-correcting q-nary perfect codes. *Info. and Control* **12,** 23–6. [9]
Segre, B. (1954). Sulle ovali nei piani lineari finiti. *Atti Accad. Naz. Lincei Rend.* **17,** 1–2. [14]
—— (1955). Curve razionali normali e k-archi negli spazi finiti. *Ann. Mat. Pura Appl.* **39,** 357–79. [15]
—— (1961). *Lectures on modern geometry*. Cremonese, Rome. [15]
Selmer, E. S. (1967). Registration numbers in Norway: some applied number theory and psychology. *Journal of the Royal Statistical Society*, Ser. A **130,** 225–31. [11]
Shannon, C. E. (1948). A mathematical theory of communication. *Bell Syst. Tech. J.* **27,** 379–423 and 623–56. [6]
Singleton, R. C. (1964). Maximum distance q-nary codes. *IEEE Trans. Info. Theory* **10,** 116–18. [10, 15]
Slepian, D. (1960). Some further theory of group codes. *Bell Syst. Tech. J.* **39,** 1219–52. [6]
Sloane, N. J. A. (1981). Error-correcting codes and cryptography. In Klarner, D. A., *The mathematical Gardner*, Wadsworth, Belmont, Calif., pp. 346–382. [16]
—— (1982). Recent bounds for codes, sphere packings and related problems obtained by linear programming and other methods. *Contemporary Mathematics* **9,** 153–85. [2]
Snover, S. L. (1973). The uniqueness of the Nordstrom–Robinson and the Golay binary codes. Ph.D. Thesis, Department of Mathematics, Michigan State Univ. [9]
Stinson, D. R. (1984). A short proof of the nonexistence of a pair of orthogonal Latin squares of order six. *J. Comb. Theory*, Ser. A **36,** 373–76. [9]
Tarry, G. (1901). Le problème des 36 officiers. *C. R. Acad. Sci. Paris* **2,** 170–203. [9]
Thas, J. A. (1968). Normal rational curves and k-arcs in Galois spaces. *Rend. Mat.* **1,** 331–4. [15]
—— (1969). Connection between the Grassmannian $G_{k-1;n}$ and the set of the k-arcs of the Galois space $S_{n,q}$. *Rend. Mat.* **2,** 121–34. [15]
Tietäväinen, A. (1973). On the nonexistence of perfect codes over finite fields. *SIAM J. Appl. Math.* **24,** 88–96. [9]
—— (1980). Bounds for binary codes just outside the Plotkin range. *Info. and Control* **47,** 85–93. [2]
Varshamov, R. R. (1957). Estimate of the number of signals in error

correcting codes. *Dokl. Akad. Nauk SSSR* **117,** 739–41. [8]
Vasil'ev, J. L. (1962). On nongroup closepacked codes. *Probl. Kibernat.* **8,** 337–39. (In Russian), translated in *Probleme der Kybernetik* **8** (1965), 375–78. [9]
Verhoeff, J. (1969). *Error detecting decimal codes.* Mathematical Centre Tracts 29, Mathematisch Centrum, Amsterdam. [11]
Verhoeff, T. (1985). Updating a table of bounds on the minimum distance of binary linear codes. Eindhoven University of Technology Report 85-WSK-01. [14]
Weber, E. W. (1983). On the football pool problem for 6 matches: a new upper bound. *J. Comb. Theory,* Ser. A **35,** 106–8. [8]
Zinov'ev, V. A. and Leont'ev, V. K. (1973). The nonexistence of perfect codes over Galois fields. *Problems of Control and Info. Theory* **2,** 123–32. [9]

Index

[*Note:* the bibliography on pages 243–8 serves as a comprehensive index of authors' names since each entry is followed by the numbers of those chapters which refer to that entry.]