11 GROUPS AND CODING

Prerequisites: Chapter 9

In today's modern world of communication, data items are constantly being transmitted from point to point. This transmission may range from the simple task of a computer terminal interacting with the mainframe computer located 200 feet away, to the more complicated task of sending a signal thousands of miles away via a satellite that is parked in an orbit 20,000 miles from the earth, or to a telephone call or letter to another part of the country. The basic problem in transmission of data is that of receiving the data as sent and not receiving a distorted piece of data. Distortion can be caused by a number of factors.

Coding theory has developed techniques for introducing redundant information in transmitted data that help in detecting, and sometimes in correcting, errors. Some of these techniques make use of group theory. We present a brief introduction to these ideas in this chapter.

11.1 CODING OF BINARY INFORMATION AND ERROR DETECTION

The basic unit of information, called a **message**, is a finite sequence of characters from a finite alphabet. We shall choose as our alphabet the set $B = \{0, 1\}$. Every character or symbol that we want to transmit is now represented as a sequence of m elements from B. That is, every character or symbol is represented in binary form. Our basic unit of information, called a **word**, is a sequence of m 0's and 1's.

The set B is a group under the binary operation $+$ whose table is shown in Table 11.1. (See Example 5 of Section 9.4.)

Table 11.1

$+$	0	1
0	0	1
1	1	0

If we think of B as the group Z_2, then $+$ is merely mod 2 addition. It follows from Theorem 1 of Section 9.5 that $B^m = B \times B \times \cdots \times B$ (m factors) is a group under

the operation $\oplus$ defined by

$$(x_1, x_2, \ldots, x_m) \oplus (y_1, y_2, \ldots, y_m) = (x_1 + y_1, x_2 + y_2, \ldots, x_m + y_m).$$

This group has been introduced in Example 2 of Section 9.5. Its identity is $\overline{0} = (0, 0, \ldots, 0)$ and every element is its own inverse. An element in B^m will be written as $(b_1, b_2, \ldots, b_m)$ or more simply as $b_1 b_2 \cdots b_m$. Observe that B^m has 2^m elements. That is, the order of the group B^m is 2^m.

Figure 11.1 shows the basic process of sending a word from one point to another point over a transmission channel. An element $x \in B^m$ is sent through the transmission channel and is received as an element $x_t \in B^m$. In actual practice, the transmission channel may suffer disturbances, which are generally called **noise**, due to weather interference, electrical problems, and so on, that may cause a 0 to be received as a 1, or vice versa. This erroneous transmission of digits in a word being sent may give rise to the situation where the word received is different from the word that was sent; that is, $x \neq x_t$. If an error does occur, then x_t could be any element of B^m.

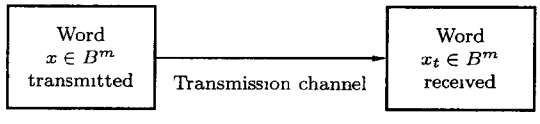

Figure 11.1

The basic task in the transmission of information is to reduce the likelihood of receiving a word that differs from the word that was sent. This is done as follows. We first choose an integer $n > m$ and a one-to-one function $e\colon B^m \to B^n$. The function e is called an (m, n) **encoding function**, and we view it as a means of representing every word in B^m as a word in B^n. If $b \in B^m$, then $e(b)$ is called the **code word** representing b. The additional 0's and 1's can provide the means to detect or correct errors produced in the transmission channel.

We now transmit the code words by means of a transmission channel. Then each code word $x = e(b)$ is received as the word x_t in B^n. This situation is illustrated in Figure 11.2.

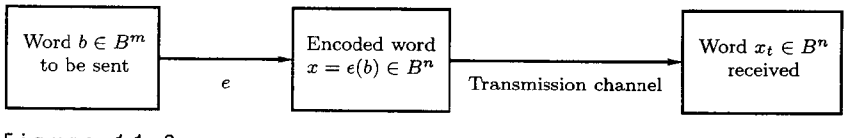

Figure 11.2

Observe that we want an encoding function e to be one to one so that different words in B^m will be assigned different code words.

If the transmission channel is noiseless, then $x_t = x$ for all x in B^n. In this case $x = e(b)$ is received for each $b \in B^m$, and since e is a known function, b may be identified.

In general, errors in transmission do occur. We will say that the code word $x = e(b)$ has been transmitted with k **or fewer errors** if x and x_t differ in at least 1 but no more than k positions.

Let $e\colon B^m \to B^n$ be an (m, n) encoding function. We say that e **detects k or fewer errors** if whenever $x = e(b)$ is transmitted with k or fewer errors, then x_t is not a code word (thus x_t could not be x and therefore could not have been correctly transmitted). For $x \in B^n$, the number of 1's in x is called the **weight** of x and is denoted by $|x|$.

EXAMPLE 1 Find the weight of each of the following words in B^5:

(a) $x = 01000$; (b) $x = 11100$; (c) $x = 00000$; (d) $x = 11111$.

Solution

(a) $|x| = 1$. (b) $|x| = 3$. (c) $|x| = 0$. (d) $|x| = 5$. ■

EXAMPLE 2 *Parity Check Code*

The following encoding function $e\colon B^m \to B^{m+1}$ is called the **parity $(m, m+1)$ check code**: If $b = b_1 b_2 \cdots b_m \in B^m$, define

$$e(b) = b_1 b_2 \cdots b_m b_{m+1},$$

where

$$b_{m+1} = \begin{cases} 0 & \text{if } |b| \text{ is even} \\ 1 & \text{if } |b| \text{ is odd.} \end{cases}$$

Observe that b_{m+1} is zero if and only if the number of 1's in b is an even number. It then follows that every code word $e(b)$ has even weight. A single error in the transmission of a code word will change the received word to a word of odd weight and therefore can be detected. In the same way we see that any odd number of errors can be detected.

For a concrete illustration of this encoding function, let $m = 3$. Then

$$\left.\begin{array}{l} e(000) = 0000 \\ e(001) = 0011 \\ e(010) = 0101 \\ e(011) = 0110 \\ e(100) = 1001 \\ e(101) = 1010 \\ e(110) = 1100 \\ e(111) = 1111 \end{array}\right\} \text{ code words.}$$

Suppose now that $b = 111$. Then $x = e(b) = 1111$. If the transmission channel transmits x as $x_t = 1101$, then $|x_t| = 3$, and we know that an odd number of errors (at least one) has occurred. ■

It should be noted that if the received word has even weight, then we cannot conclude that the code word was transmitted correctly, since this encoding function does not detect an even number of errors. Despite this limitation, the parity check code is widely used.

EXAMPLE 3 Consider the following $(m, 3m)$ encoding function $e\colon B^m \to B^{3m}$. If

$$b = b_1 b_2 \cdots b_m \in B^m,$$

define

$$e(b) = e(b_1b_2\cdots b_m) = b_1b_2\cdots b_mb_1b_2\cdots b_mb_1b_2\cdots b_m.$$

That is, the encoding function e repeats each word of B^m three times. For a concrete example, let $m = 3$. Then

$$\left.\begin{array}{l} e(000) = 000000000 \\ e(001) = 001001001 \\ e(010) = 010010010 \\ e(011) = 011011011 \\ e(100) = 100100100 \\ e(101) = 101101101 \\ e(110) = 110110110 \\ e(111) = 111111111 \end{array}\right\} \text{ code words.}$$

Suppose now that $b = 011$. Then $e(011) = 011\underline{0}11011$. Assume now that the transmission channel makes an error in the underlined digit and that we receive the word 011111011. This is not a code word, so we have detected the error. It is not hard to see that any single error and any two errors can be detected. ■

Let x and y be words in B^m. The **Hamming distance** $\delta(x, y)$ between x and y is the weight, $|x \oplus y|$, of $x \oplus y$. Thus the distance between $x = x_1x_2\cdots x_m$ and $y = y_1y_2\cdots y_m$ is the number of values of i such that $x_i \neq y_i$, that is, the number of positions in which x and y differ. Using the weight of $x \oplus y$ is a convenient way to count the number of different positions.

EXAMPLE 4 Find the distance between x and y:

(a) $x = 110110$, $y = 000101$.
(b) $x = 001100$, $y = 010110$.

Solution

(a) $x \oplus y = 110011$, so $|x \oplus y| = 4$.
(b) $x \oplus y = 011010$, so $|x \oplus y| = 3$. ■

Theorem 1 Properties of the Distance Function

Let x, y, and z be elements of B^m. Then
(a) $\delta(x, y) = \delta(y, x)$.
(b) $\delta(x, y) \geq 0$.
(c) $\delta(x, y) = 0$ if and only if $x = y$.
(d) $\delta(x, y) \leq \delta(x, z) + \delta(z, y)$. ●

Proof Properties (a), (b), and (c) are simple to prove and are left as exercises.
(d) For a and b in B^m,

$$|a \oplus b| \leq |a| + |b|,$$

since at any position where a and b differ one of them must contain a 1. Also, if $a \in B^m$, then $a \oplus a = \bar{0}$, the identity element in B^m. Then

$$\begin{aligned}\delta(x, y) &= |x \oplus y| = |x \oplus \bar{0} \oplus y| = |x \oplus z \oplus z \oplus y| \\ &\leq |x \oplus z| + |z \oplus y| \\ &= \delta(x, z) + \delta(z, y).\end{aligned}$$ ▼

The **minimum distance** of an encoding function $e: B^m \to B^n$ is the minimum of the distances between all distinct pairs of code words; that is,

$$\min\{\delta(e(x), e(y)) \mid x, y \in B^m\}.$$

EXAMPLE 5 Consider the following (2, 5) encoding function e:

$$\left.\begin{aligned} e(00) &= 00000 \\ e(10) &= 00111 \\ e(01) &= 01110 \\ e(11) &= 11111 \end{aligned}\right\} \text{ code words.}$$

The minimum distance is 2, as can be checked by computing the minimum of the distances between all six distinct pairs of code words. ■

Theorem 2 An (m, n) encoding function $e: B^m \to B^n$ can detect k or fewer errors if and only if its minimum distance is at least $k + 1$. ●

Proof Suppose that the minimum distance between any two code words is at least $k + 1$. Let $b \in B^m$, and let $x = e(b) \in B^n$ be the code word representing b. Then x is transmitted and is received as x_t. If x_t were a code word different from x, then $\delta(x, x_t) \geq k + 1$, so x would be transmitted with $k + 1$ or more errors. Thus, if x is transmitted with k or fewer errors, then x_t cannot be a code word. This means that e can detect k or fewer errors.

Conversely, suppose that the minimum distance between code words is $r \leq k$, and let x and y be code words with $\delta(x, y) = r$. If $x_t = y$, that is, if x is transmitted and is mistakenly received as y, then $r \leq k$ errors have been committed and have not been detected. Thus it is not true that e can detect k or fewer errors. ▼

EXAMPLE 6 Consider the (3, 8) encoding function $e: B^3 \to B^8$ defined by

$$\left.\begin{aligned} e(000) &= 00000000 \\ e(001) &= 10111000 \\ e(010) &= 00101101 \\ e(011) &= 10010101 \\ e(100) &= 10100100 \\ e(101) &= 10001001 \\ e(110) &= 00011100 \\ e(111) &= 00110001 \end{aligned}\right\} \text{ code words.}$$

How many errors will e detect?

Solution The minimum distance of e is 3, as can be checked by computing the minimum of the distances between all 28 distinct pairs of code words. By Theorem 2, the code will detect k or fewer errors if and only if its minimum distance is at least $k + 1$. Since the minimum distance is 3, we have $3 \geq k + 1$ or $k \leq 2$. Thus the code will detect two or fewer errors. ■

Group Codes

So far, we have not made use of the fact that $(B^n, \oplus)$ is a group. We shall now consider an encoding function that makes use of this property of B^n.

An (m, n) encoding function $e: B^m \to B^n$ is called a **group code** if

$$e(B^m) = \{e(b) \mid b \in B^m\} = \text{Ran}(e)$$

is a subgroup of B^n.

Recall from the definition of subgroup given in Section 9.4 that N is a subgroup of B^n if (a) the identity of B^n is in N, (b) if x and y belong to N, then $x \oplus y \in N$, and (c) if x is in N, then its inverse is in N. Property (c) need not be checked here, since every element in B^n is its own inverse. Moreover, since B^n is Abelian, every subgroup of B^n is a normal subgroup.

EXAMPLE 7 Consider the (3, 6) encoding function $e: B^3 \to B^6$ defined by

$$\left.\begin{array}{l} e(000) = 000000 \\ e(001) = 001100 \\ e(010) = 010011 \\ e(011) = 011111 \\ e(100) = 100101 \\ e(101) = 101001 \\ e(110) = 110110 \\ e(111) = 111010 \end{array}\right\} \text{ code words.}$$

Show that this encoding function is a group code.

Solution We must show that the set of all code words

$$N = \{000000, 001100, 010011, 011111, 100101, 101001, 110110, 111010\}$$

is a subgroup of B^6. This is done by first noting that the identity of B^6 belongs to N. Next we verify, by trying all possibilities, that if x and y are elements in N, then $x \oplus y$ is in N. Hence N is a subgroup of B^6, and the given encoding function is a group code. ■

The strategy of the next proof is similar to the way we often show two sets A and B are the same by showing that $A \subseteq B$ and $B \subseteq A$. Here we show that $\delta = \eta$ by proving $\delta \leq \eta$ and $\eta \leq \delta$.

Theorem 3 Let $e: B^m \to B^n$ be a group code. The minimum distance of e is the minimum weight of a nonzero code word. ●

Proof Let δ be the minimum distance of the group code, and suppose that $\delta = \delta(x, y)$, where x and y are distinct code words. Also, let η be the minimum weight of a nonzero code word and suppose that $\eta = |z|$ for a code word z. Since e is a group code, $x \oplus y$ is a nonzero code word. Thus

$$\delta = \delta(x, y) = |x \oplus y| \geq \eta.$$

On the other hand, since 0 and z are distinct code words,

$$\eta = |z| = |z \oplus 0| = \delta(z, 0) \geq \delta.$$

Hence $\eta = \delta$. ▼

One advantage of a group code is given in the following example.

EXAMPLE 8 The minimum distance of the group code in Example 7 is 2, since by Theorem 3 this distance is equal to the smallest number of 1's in any of the seven nonzero code words. To check this directly would require 28 different calculations. ■

We shall now take a brief look at a procedure for generating group codes. First, we need several additional results on Boolean matrices. Consider the set B with operation $+$ defined in Table 11.1. Now let $\mathbf{D} = [\, d_{ij} \,]$ and $\mathbf{E} = [\, e_{ij} \,]$ be $m \times n$ Boolean matrices. We define the **mod-2 sum** $\mathbf{D} \oplus \mathbf{E}$ as the $m \times n$ Boolean matrix $\mathbf{F} = [\, f_{ij} \,]$, where

$$f_{ij} = d_{ij} + e_{ij}, \qquad 1 \leq i \leq m, \quad 1 \leq j \leq n. \qquad \text{(Here } + \text{ is addition in } B.\text{)}$$

EXAMPLE 9 We have

$$\begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} \oplus \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 1+1 & 0+1 & 1+0 & 1+1 \\ 0+1 & 1+1 & 1+0 & 0+1 \\ 1+0 & 0+1 & 0+1 & 1+1 \end{bmatrix}$$

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

Observe that if $\mathbf{F} = \mathbf{D} \oplus \mathbf{E}$, then f_{ij} is zero when *both* d_{ij} and e_{ij} are zero or both are one. ■

Table 11.2

$\cdot$	0	1
0	0	0
1	0	1

Next, consider the set $B = \{0, 1\}$ with the binary operation given in Table 11.2. This operation has been seen earlier in a different setting and with a different symbol. In Chapter 6 it was shown that B is the unique Boolean algebra with two elements. In particular, B is a lattice with partial order $\leq$ defined by $0 \leq 0$, $0 \leq 1$, $1 \leq 1$. Then the reader may easily check that if a and b are any two elements of B,

$$a \cdot b = a \wedge b \qquad \text{(the greatest lower bound of } a \text{ and } b\text{)}.$$

Thus Table 11.2 for $\cdot$ is just a copy of the table for $\wedge$.

Let $\mathbf{D} = [\, d_{ij} \,]$ be an $m \times p$ Boolean matrix, and let $\mathbf{E} = [\, e_{ij} \,]$ be a $p \times n$ Boolean matrix. We define the **mod-2 Boolean product** $\mathbf{D} * \mathbf{E}$ as the $m \times n$ matrix $\mathbf{F} = [\, f_{ij} \,]$, where

$$f_{ij} = d_{i1} \cdot e_{1j} + d_{i2} \cdot e_{2j} + \cdots + d_{ip} \cdot e_{pj}, \qquad 1 \le i \le m, \quad 1 \le j \le n.$$

This type of multiplication is illustrated in Figure 11.3. Compare this with similar figures in Section 1.5.

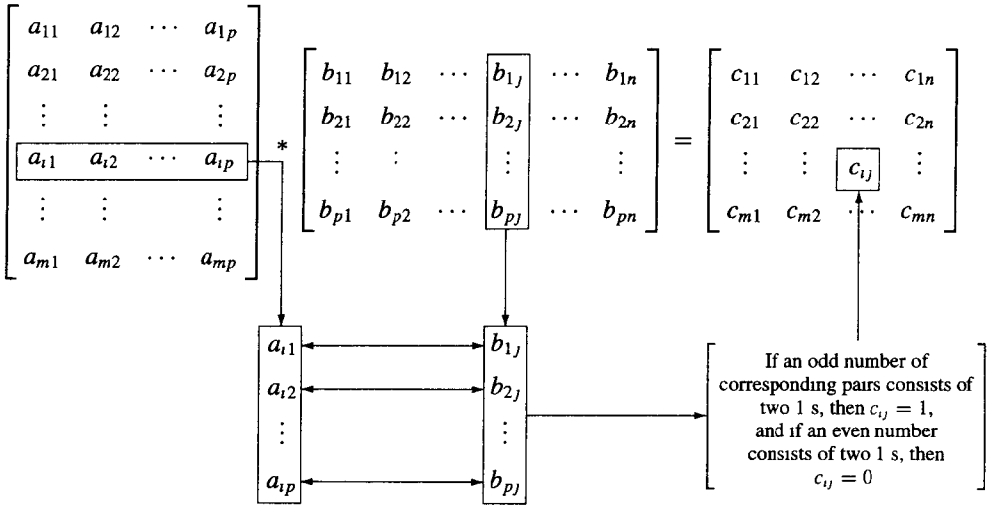

Figure 11.3

EXAMPLE 10 We have

$$\begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1\cdot 1 + 1\cdot 1 + 0\cdot 0 & 1\cdot 0 + 1\cdot 1 + 0\cdot 1 \\ 0\cdot 1 + 1\cdot 1 + 1\cdot 0 & 0\cdot 0 + 1\cdot 1 + 1\cdot 1 \end{bmatrix}$$

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

The proof of the following theorem is left as an exercise.

Theorem 4 Let $\mathbf{D}$ and $\mathbf{E}$ be $m \times p$ Boolean matrices, and let $\mathbf{F}$ be a $p \times n$ Boolean matrix. Then

$$(\mathbf{D} \oplus \mathbf{E}) * \mathbf{F} = (\mathbf{D} * \mathbf{F}) \oplus (\mathbf{E} * \mathbf{F}).$$

That is, a distributive property holds for $\oplus$ and $*$. ●

We shall now consider the element $x = x_1 x_2 \cdots x_n \in B^n$ as the $1 \times n$ matrix $[\, x_1 \;\; x_2 \;\; \cdots \;\; x_n \,]$.

Theorem 5 Let m and n be nonnegative integers with $m < n$, $r = n - m$, and let $\mathbf{H}$ be an $n \times r$ Boolean matrix. Then the function $f_H: B^n \to B^r$ defined by

$$f_H(x) = x * \mathbf{H}, \qquad x \in B^n$$

is a homomorphism from the group B^n to the group B^r. ●

Proof If x and y are elements in B^n, then

$$\begin{aligned} f_H(x \oplus y) &= (x \oplus y) * \mathbf{H} \\ &= (x * \mathbf{H}) \oplus (y * \mathbf{H}) \qquad \text{by Theorem 4} \\ &= f_H(x) \oplus f_H(y). \end{aligned}$$

Hence f_H is a homomorphism from B^n to B^r. ▼

Corollary 1 Let m, n, r, $\mathbf{H}$, and f_H be as in Theorem 5. Then

$$N = \{x \in B^n \mid x * \mathbf{H} = \bar{0}\}$$

is a normal subgroup of B^n. ●

Proof It follows from the results in Section 9.5 that N is the kernel of the homomorphism f_H, so it is a normal subgroup of B^n. ■

Let $m < n$ and $r = n - m$. An $n \times r$ Boolean matrix

$$\mathbf{H} = \begin{cases} \left[\begin{array}{cccc} h_{11} & h_{12} & \cdots & h_{1r} \\ h_{21} & h_{22} & \cdots & h_{2r} \\ \vdots & \vdots & & \vdots \\ h_{m1} & h_{m2} & \cdots & h_{mr} \end{array}\right] \\ n - m = r \text{ rows} \quad \left\{ \left[\begin{array}{cccc} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 1 \end{array}\right] \right. \end{cases},$$

whose last r rows form the $r \times r$ identity matrix, is called a **parity check matrix**. We use $\mathbf{H}$ to define an encoding function $e_H: B^m \to B^n$. If $b = b_1 b_2 \cdots b_m$, let $x = e_H(b) = b_1 b_2 \cdots b_m x_1 x_2 \cdots x_r$, where

$$\begin{aligned} x_1 &= b_1 \cdot h_{11} + b_2 \cdot h_{21} + \cdots + b_m \cdot h_{m1} \\ x_2 &= b_1 \cdot h_{12} + b_2 \cdot h_{22} + \cdots + b_m \cdot h_{m2} \\ &\vdots \\ x_r &= b_1 \cdot h_{1r} + b_2 \cdot h_{2r} + \cdots + b_m \cdot h_{mr}. \end{aligned} \tag{1}$$

Theorem 6 Let $x = y_1 y_2 \cdots y_m x_1 \cdots x_r \in B^n$. Then $x * \mathbf{H} = \bar{0}$ if and only if $x = e_H(b)$ for some $b \in B^m$. ●

Proof Suppose that $x * \mathbf{H} = \overline{0}$. Then

$$\begin{aligned} y_1 \cdot h_{11} + y_2 \cdot h_{21} + \cdots + y_m \cdot h_{m1} + x_1 &= 0 \\ y_1 \cdot h_{12} + y_2 \cdot h_{22} + \cdots + y_m \cdot h_{m2} + x_2 &= 0 \\ &\vdots \\ y_1 \ h_{1r} + y_2 \cdot h_{2r} + \cdots + y_m \cdot h_{mr} + x_r &= 0. \end{aligned}$$

The first equation is of the form

$$a + x_1 = 0, \quad \text{where } a = y_1 \cdot h_{11} + y_2 \cdot h_{21} + \cdots + y_m \cdot h_{m1}.$$

Adding a to both sides, we obtain

$$\begin{aligned} a + (a + x_1) &= a + 0 = a \\ (a + a) + x_1 &= a \\ 0 + x_1 &= a \quad \text{since } a + a = 0 \\ x_1 &= a. \end{aligned}$$

This can be done for each row; therefore,

$$x_i = y_1 \cdot h_{1i} + y_2 \cdot h_{2i} + \cdots + y_m \cdot h_{mi}, \qquad 1 \le i \le r.$$

Letting $b_1 = y_1$, $b_2 = y_2$, ..., $b_m = y_m$, we see that $x_1, x_2, \ldots, x_r$ satisfy the equations in (1). Thus $b = b_1 b_2 \cdots b_m \in B^m$ and $x = e_H(b)$.

Conversely, if $x = e_H(b)$, the equations in (1) can be rewritten by adding x_i to both sides of the ith equation, $i = 1, 2, \ldots, n$, as

$$\begin{aligned} b_1 \cdot h_{11} + b_2 \cdot h_{21} + \cdots + b_m \cdot h_{m1} + x_1 &= 0 \\ b_1 \cdot h_{12} + b_2 \cdot h_{22} + \cdots + b_m \cdot h_{m2} + x_2 &= 0 \\ &\vdots \\ b_1 \cdot h_{1r} + b_2 \cdot h_{2r} + \cdots + b_m \cdot h_{mr} + x_r &= 0, \end{aligned}$$

which shows $x * \mathbf{H} = \overline{0}$. ▼

Corollary 2 $e_H(B^m) = \{e_H(b) \mid b \in B^m\}$ is a subgroup of B^n. ●

Proof The result follows from the observation that

$$e_H(B^m) = \ker(f_H)$$

and from Corollary 1. Thus e_H is a group code. ▼

EXAMPLE 11 Let $m = 2$, $n = 5$, and

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

Determine the group code $e_H : B^2 \to B^5$.

Solution We have $B^2 = \{00, 10, 01, 11\}$. Then

$$e(00) = 00x_1x_2x_3,$$

where x_1, x_2, and x_3 are determined by the equations in (1). Thus

$$x_1 = x_2 = x_3 = 0$$

and

$$e(00) = 00000.$$

Next

$$e(10) = 10x_1x_2x_3.$$

Using the equations in (1) with $b_1 = 1$ and $b_2 = 0$, we obtain

$$x_1 = 1 \cdot 1 + 0 \cdot 0 = 1$$
$$x_2 = 1 \cdot 1 + 0 \cdot 1 = 1$$
$$x_3 = 1 \cdot 0 + 0 \cdot 1 = 0.$$

Thus $x_1 = 1$, $x_2 = 1$, and $x_3 = 0$, so

$$e(10) = 10110.$$

Similarly (verify),

$$e(01) = 01011$$
$$e(11) = 11101.$$ ■

11.1 Exercises

Find the weights of the given words.

1. (a) 1011 (b) 0110 (c) 1110

2. (a) 011101 (b) 11111 (c) 010101

3. Consider the (3, 4) parity check code. For each of the received words, determine whether an error will be detected.

(a) 0100 (b) 1100

4. Consider the (3, 4) parity check code. For each of the received words, determine whether an error will be detected.

(a) 0010 (b) 1001

5. Consider the (6, 7) parity check code. For each of the received words, determine whether an error will be detected.

(a) 1101010 (b) 1010011

(c) 0011111 (d) 1001101

6. Consider the $(m, 3m)$ encoding function of Example 3, where $m = 4$. For each of the received words, determine whether an error will be detected.

(a) 011010011111 (b) 110110010110

7. Consider the $(m, 3m)$ encoding function of Example 3, where $m = 4$. For each of the received words, determine whether an error will be detected.

(a) 010010110010 (b) 001001111001

8. Explain how $|x \oplus y|$ counts the number of positions in which x and y differ.

9. Find the distance between x and y.

(a) $x = 1100010$, $y = 1010001$

(b) $x = 0100110$, $y = 0110010$

10. Find the distance between x and y.

(a) $x = 00111001$, $y = 10101001$

(b) $x = 11010010$, $y = 00100111$

11. (a) Prove Theorem 1(a).
(b) Prove Theorem 1(b).

12. Prove Theorem 1(c).

13. Find the minimum distance of the (2, 4) encoding function e.

$$e(00) = 0000 \qquad e(10) = 0110$$
$$e(01) = 1011 \qquad e(11) = 1100$$

14. Find the minimum distance of the (3, 8) encoding function e.

$$e(000) = 00000000 \qquad e(100) = 01100101$$
$$e(001) = 01110010 \qquad e(101) = 10110000$$
$$e(010) = 10011100 \qquad e(110) = 11110000$$
$$e(011) = 01110001 \qquad e(111) = 00001111$$

15. Consider the (2, 6) encoding function e.

$$e(00) = 000000 \qquad e(10) = 101010$$
$$e(01) = 011110 \qquad e(11) = 111000$$

(a) Find the minimum distance of e.
(b) How many errors will e detect?

16. Consider the (3, 9) encoding function e.

$$e(000) = 000000000 \qquad e(100) = 010011010$$
$$e(001) = 011100101 \qquad e(101) = 111101011$$
$$e(010) = 010101000 \qquad e(110) = 001011000$$
$$e(011) = 110010001 \qquad e(111) = 110000111$$

(a) Find the minimum distance of e.
(b) How many errors will e detect?

17. Show that the (2, 5) encoding function $e: B^2 \to B^5$ defined by

$$e(00) = 00000 \qquad e(10) = 10101$$
$$e(01) = 01110 \qquad e(11) = 11011$$

is a group code.

18. Show that the (3, 7) encoding function $e: B^3 \to B^7$ defined by

$$e(000) = 0000000 \qquad e(100) = 1000101$$
$$e(001) = 0010110 \qquad e(101) = 1010011$$
$$e(010) = 0101000 \qquad e(110) = 1101101$$
$$e(011) = 0111110 \qquad e(111) = 1111011$$

is a group code.

19. Find the minimum distance of the group code defined in Exercise 17.

20. Find the minimum distance of the group code defined in Exercise 18.

21. Compute

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

22. Compute

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

23. Compute

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

24. Compute

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

25. Let

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

be a parity check matrix. Determine the (2, 5) group code function $e_H: B^2 \to B^5$.

26. Let

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

be a parity check matrix. Determine the (3, 6) group code $e_H: B^3 \to B^6$.

27. Prove Theorem 4.

11.2 DECODING AND ERROR CORRECTION

Consider an (m, n) encoding function $e\colon B^m \to B^n$. Once the encoded word $x = e(b) \in B^n$, for $b \in B^m$, is received as the word x_t, we are faced with the problem of identifying the word b that was the original message.

An onto function $d\colon B^n \to B^m$ is called an (n, m) **decoding function associated with** e if $d(x_t) = b' \in B^m$ is such that when the transmission channel has no noise, then $b' = b$, that is,

$$d \circ e = 1_{B^m},$$

where 1_{B^m} is the identity function on B^m. The decoding function d is required to be onto so that every received word can be decoded to give a word in B^m. It decodes properly received words correctly, but the decoding of improperly received words may or may not be correct.

EXAMPLE 1 Consider the parity check code that is defined in Example 2 of Section 11.1. We now define the decoding function $d\colon B^{m+1} \to B^m$. If $y = y_1 y_2 \cdots y_m y_{m+1} \in B^{m+1}$, then

$$d(y) = y_1 y_2 \cdots y_m.$$

Observe that if $b = b_1 b_2 \cdots b_m \in B^m$, then

$$(d \circ e)(b) = d(e(b)) = b,$$

so $d \circ e = 1_{B^m}$.

For a concrete example, let $m = 4$. Then we obtain $d(10010) = 1001$ and $d(11001) = 1100$. ■

Let e be an (m, n) encoding function and let d be an (n, m) decoding function associated with e. We say that the pair (e, d) **corrects k or fewer errors** if whenever $x = e(b)$ is transmitted correctly or with k or fewer errors and x_t is received, then $d(x_t) = b$. Thus x_t is decoded as the correct message b.

EXAMPLE 2 Consider the $(m, 3m)$ encoding function defined in Example 3 of Section 11.1. We now define the decoding function $d\colon B^{3m} \to B^m$. Let

$$y = y_1 y_2 \cdots y_m y_{m+1} \cdots y_{2m} y_{2m+1} \cdots y_{3m}.$$

Then

$$d(y) = z_1 z_2 \cdots z_m,$$

where

$$z_i = \begin{cases} 1 & \text{if } \{y_i, y_{i+m}, y_{i+2m}\} \text{ has at least two 1's} \\ 0 & \text{if } \{y_i, y_{i+m}, y_{i+2m}\} \text{ has less than two 1's.} \end{cases}$$

That is, the decoding function d examines the ith digit in each of the three blocks transmitted. The digit that occurs at least twice in these three blocks is chosen as the decoded ith digit. For a concrete example, let $m = 3$. Then

$$e(100) = 100100100$$
$$e(011) = 011011011$$
$$e(001) = 001001001.$$

Suppose now that $b = 011$. Then $e(011) = 011011011$. Assume now that the transmission channel makes an error in the underlined digit and that we receive the word $x_t = 011111011$. Then, since the first digits in two out of the three blocks are 0, the first digit is decoded as 0. Similarly, the second digit is decoded as 1, since all three second digits in the three blocks are 1. Finally, the third digit is also decoded as 1, for the analogous reason. Hence $d(x_t) = 011$; that is, the decoded word is 011, which is the word that was sent. Therefore, the single error has been corrected. A similar analysis shows that, if e is this $(m, 3m)$ code for any value of m and d is as defined, then (e, d) corrects any single error. ■

Given an (m, n) encoding function $e\colon B^m \to B^n$, we often need to determine an (n, m) decoding function $d\colon B^n \to B^m$ associated with e. We now discuss a method, called the **maximum likelihood technique**, for determining a decoding function d for a given e.

Since B^m has 2^m elements, there are 2^m code words in B^n. We first list the code words in a fixed order:

$$x^{(1)}, x^{(2)}, \ldots, x^{(2^m)}.$$

If the received word is x_t, we compute $\delta(x^{(i)}, x_t)$ for $1 \leq i \leq 2^m$ and choose the first code word, say it is $x^{(s)}$, such that

$$\min_{1 \leq i \leq 2^m} \{\delta(x^{(i)}, x_t)\} = \delta(x^{(s)}, x_t).$$

That is, $x^{(s)}$ is a code word that is closest to x_t and the first in the list. If $x^{(s)} = e(b)$, we define the **maximum likelihood decoding function** d associated with e by

$$d(x_t) = b.$$

Observe that d depends on the particular order in which the code words in $e(B^n)$ are listed. If the code words are listed in a different order, we may obtain a different maximum likelihood decoding function d associated with e.

Theorem 1 Suppose that e is an (m, n) encoding function and d is a maximum likelihood decoding function associated with e. Then (e, d) can correct k or fewer errors if and only if the minimum distance of e is at least $2k + 1$. ●

Proof Assume that the minimum distance of e is at least $2k + 1$. Let $b \in B^m$ and $x = e(b) \in B^n$. Suppose that x is transmitted with k or fewer errors, and x_t is received. This means that $\delta(x, x_t) \leq k$. If z is any other code word, then

$$2k + 1 \leq \delta(x, z) \leq \delta(x, x_t) + \delta(x_t, z) \leq k + d(x_t, z).$$

Thus $\delta(x_t, z) \geq 2k + 1 - k = k + 1$. This means that x is the unique code word that is closest to x_t, so $d(x_t) = b$. Hence (e, d) corrects k or fewer errors.

Conversely, assume that the minimum distance between code words is $r \leq 2k$, and let $x = e(b)$ and $x' = e(b')$ be code words with $\delta(x, x') = r$. Suppose that x' precedes x in the list of code words used to define d. Write $x = b_1 b_2 \cdots b_n$, $x' = b'_1 b'_2 \cdots b'_n$. Then $b_i \neq b'_i$ for exactly r integers i between 1 and n. Assume, for simplicity, that $b_1 \neq b'_1$, $b_2 \neq b'_2$, $\ldots$, $b_r \neq b'_r$, but $b_i = b'_i$ when $i > r$. Any other case is handled in the same way.

(a) Suppose that $r \leq k$. If x is transmitted as $x_t = x'$, then $r \leq k$ errors have been committed, but $d(x_t) = b'$; so (e, d) has not corrected the r errors.

(b) Suppose that $k + 1 \leq r \leq 2k$, and let

$$y = b_1' b_2' \cdots b_k' b_{k+1} \cdots b_n.$$

If x is transmitted as $x_t = y$, then $\delta(x_t, x') = r - k \leq k$ and $\delta(x_t, x) \geq k$. Thus x' is at least as close to x_t as x is, and x' precedes x in the list of code words; so $d(x_t) \neq b$. Then we have committed k errors, which (e, d) has not corrected. ▼

EXAMPLE 3 Let e be the $(3, 8)$ encoding function defined in Example 6 of Section 11.1, and let d be an $(8, 3)$ maximum likelihood decoding function associated with e. How many errors can (e, d) correct?

Solution Since the minimum distance of e is 3, we have $3 \geq 2k + 1$, so $k \leq 1$. Thus (e, d) can correct one error. ■

We now discuss a simple and effective technique for determining a maximum likelihood decoding function associated with a given group code. First, we prove the following result.

Theorem 2 If K is a finite subgroup of a group G, then every left coset of K in G has exactly as many elements as K. ●

Proof Let aK be a left coset of K in G, where $a \in G$. Consider the function $f: K \to aK$ defined by

$$f(k) = ak, \qquad \text{for } k \in K.$$

We show that f is one to one and onto.

To show that f is one to one, we assume that

$$f(k_1) = f(k_2), \qquad k_1, k_2 \in K.$$

Then

$$ak_1 = ak_2.$$

By Theorem 2 of Section 9.4, $k_1 = k_2$. Hence f is one to one.

To show that f is onto, let b be an arbitrary element in aK. Then $b = ak$ for some $k \in K$. We now have

$$f(k) = ak = b,$$

so f is onto. Since f is one to one and onto, K and aK have the same number of elements. ▼

Let $e: B^m \to B^n$ be an (m, n) encoding function that is a group code. Thus the set N of code words in B^n is a subgroup of B^n whose order is 2^m, say $N = \{x^{(1)}, x^{(2)}, \ldots, x^{(2^m)}\}$.

Suppose that the code word $x = e(b)$ is transmitted and that the word x_t is received. The left coset of x_t is

$$x_t \oplus N = \{\epsilon_1, \epsilon_2, \ldots, \epsilon_{2^m}\},$$

where $\epsilon_t = x_t \oplus x^{(i)}$. The distance from x_t to code word $x^{(i)}$ is just $|\epsilon_t|$, the weight of ϵ_t. Thus, if ϵ_j is a coset member with smallest weight, then $x^{(j)}$ must be a code word that is closest to x_t. In this case, $x^{(j)} = \overline{0} \oplus x^{(j)} = x_t \oplus x_t \oplus x^{(j)} = x_t \oplus \epsilon_j$. An element ϵ_j, having smallest weight, is called a **coset leader**. Note that a coset leader need not be unique.

If $e\colon B^m \to B^n$ is a group code, we now state the following procedure for obtaining a maximum likelihood decoding function associated with e.

***STEP 1*:** Determine all the left cosets of $N = e(B^m)$ in B^n.

***STEP 2*:** For each coset, find a coset leader (a word of least weight). Steps 1 and 2 can be carried out in a systematic tabular manner that will be described later.

***STEP 3*:** If the word x_t is received, determine the coset of N to which x_t belongs. Since N is a normal subgroup of B^n, it follows from Theorems 3 and 4 of Section 9.5 that the cosets of N form a partition of B^n, so each element of B^n belongs to one and only one coset of N in B^n. Moreover, there are $2^n/2^m$ or 2^r distinct cosets of N in B^n.

***STEP 4*:** Let ϵ be a coset leader for the coset determined in Step 3 Compute $x = x_t \oplus \epsilon$. If $x = e(b)$, we let $d(x_t) = b$. That is, we decode x_t as b.

To implement the foregoing procedure, we must keep a complete list of all the cosets of N in B^n, usually in tabular form, with each row of the table containing one coset. We identify a coset leader in each row. Then, when a word x_t is received, we locate the row that contains it, find the coset leader for that row, and add it to x_t. This gives us the code word closest to x_t. We can eliminate the need for these additions if we construct a more systematic table.

Before illustrating with an example, we make several observations. Let

$$N = \{x^{(1)}, x^{(2)}, \ldots, x^{(2^m)}\},$$

where $x^{(1)}$ is $\overline{0}$, the identity of B^n.

Steps 1 and 2 in the preceding decoding algorithm are carried out as follows. First, list all the elements of N in a row, starting with the identity $\overline{0}$ at the left. Thus we have

$$\overline{0} \quad x^{(2)} \quad x^{(3)} \quad \cdots \quad x^{(2^m)}.$$

This row is the coset $[\overline{0}]$, and it has $\overline{0}$ as its coset leader. For this reason we will also refer to $\overline{0}$ as ϵ_1. Now choose any word y in B^n that has not been listed in the first row. List the elements of the coset $y \oplus N$ as the second row. This coset also has 2^m elements. Thus we have the two rows

$$\begin{array}{cccccc} \overline{0} & x^{(2)} & x^{(3)} & \cdots & x^{(2^m)} \\ y \oplus \overline{0} & y \oplus x^{(2)} & y \oplus x^{(3)} & \cdots & y \oplus x^{(2^m)}. \end{array}$$

In the coset $y \oplus N$, pick an element of least weight, a coset leader, which we denote by $\epsilon^{(2)}$. In case of ties, choose any element of least weight. Recall from Section 9.5 that, since $\epsilon^{(2)} \in y \oplus N$, we have $y \oplus N = \epsilon^{(2)} \oplus N$. This means that every word

in the second row can be written as $\epsilon^{(2)} \oplus v$, $v \in N$. Now rewrite the second row as follows:

$$\epsilon^{(2)} \quad \epsilon^{(2)} \oplus x^{(2)} \quad \epsilon^{(2)} \oplus x^{(3)} \quad \cdots \quad \epsilon^{(2)} \oplus x^{(2^m)}$$

with $\epsilon^{(2)}$ in the leftmost position.

Next, choose another element z in B^n that has not yet been listed in either of the first two rows and form the third row $(z \oplus x^{(j)})$, $1 \leq j \leq 2^m$ (another coset of N in B^n). This row can be rewritten in the form

$$\epsilon^{(3)} \quad \epsilon^{(3)} \oplus x^{(2)} \quad \epsilon^{(3)} \oplus x^{(3)} \quad \cdots \quad \epsilon^{(3)} \oplus x^{(2^m)},$$

where $\epsilon^{(3)}$ is a coset leader for the row.

Continue this process until all elements of B^n have been listed. The resulting Table 11.3 is called a **decoding table**. Notice that it contains 2^r rows, one for each coset of N. If we receive the word x_t, we locate it in the table. If x is the element of N that is at the top of the column containing x_t, then x is the code word closest to x_t. Thus, if $x = e(b)$, we let $d(x_t) = b$.

Table 11.3

$\overline{0}$	$x^{(2)}$	$x^{(3)}$	$\cdots$	$x^{(2^m-1)}$
$\epsilon^{(2)}$	$\epsilon^{(2)} \oplus x^{(2)}$	$\epsilon^{(2)} \oplus x^{(3)}$	$\cdots$	$\epsilon^{(2)} \oplus x^{(2^m-1)}$
$\vdots$	$\vdots$	$\vdots$		$\vdots$
$\epsilon^{(2^r)}$	$\epsilon^{(2^r)} \oplus x^{(2)}$	$\epsilon^{(2^r)} \oplus x^{(3)}$	$\cdots$	$\epsilon^{(2^r)} \oplus x^{(2^m-1)}$

EXAMPLE 4 Consider the (3, 6) group code defined in Example 7 of Section 11.1. Here

$$N = \{000000, 001100, 010011, 011111, 100101, 101001, 110110, 111010\}$$
$$= \{x^{(1)}, x^{(2)}, \ldots, x^{(8)}\}$$

as defined in Example 1. We now implement the decoding procedure for e as follows.

***STEPS 1 and 2*:** Determine all the left cosets of N in B^6, as rows of a table. For each row i, locate the coset leader ϵ_i, and rewrite the row in the order

$$\epsilon_i, \quad \epsilon_i \oplus 001100, \quad \epsilon_i \oplus 010011, \quad \ldots, \quad \epsilon_i \oplus 111010.$$

The result is shown in Table 11.4.

Table 11.4

000000	001100	010011	011111	100101	101001	110110	111010
000001	001101	010010	011110	100100	101000	110111	111011
000010	001110	010001	011101	100111	101011	110100	111000
000100	001000	010111	011011	100001	101101	110010	111110
010000	011100	000011	001111	110101	111001	100110	101010
100000	101100	110011	111111	000101	001001	010110	011010
000110	001010	010101	011001	100011	101111	110000	111100
010100	011000	000111	001011	110001	111101	100010	101110

STEPS 3 and 4: If we receive the word 000101, we decode it by first locating it in the decoding table: it appears in the fifth column, where it is underlined. The word at the top of the fifth column is 100101. Since $e(100) = 100101$, we decode 000101 as 100. Similarly, if we receive the word 010101, we first locate it in the third column of the decoding table, where it is underlined twice. The word at the top of the third column is 010011. Since $e(010) = 010011$, we decode 010101 as 010.

We make the following observations for this example. In determining the decoding table in Steps 1 and 2, there was more than one candidate for coset leader of the last two cosets. In row 7 we chose 00110 as coset leader. If we had chosen 001010 instead, row 7 would have appeared in the rearranged form

$$001010 \quad 001010 \oplus 001100 \quad \cdots \quad 001010 \oplus 111010$$

or

$$001010 \quad 000110 \quad 011001 \quad 010101 \quad 101111 \quad 100011 \quad 111100 \quad 110000.$$

The new decoding table is shown in Table 11.5.

Table 11.5

000000	001100	010011	011111	100101	101001	110110	111010
000001	001101	010010	011110	100100	101000	110111	111011
000010	001110	010001	011101	100111	101011	110100	111000
000100	001000	010111	011011	100001	101101	110010	111110
010000	011100	000011	001111	110101	111001	100110	101010
100000	101100	110011	111111	000101	001001	010110	011010
001010	000110	011001	<u>010101</u>	101111	100011	111100	110000
010100	011000	000111	001011	110001	111101	100010	101110

Now, if we receive the word 010101, we first locate it in the *fourth* column of Table 11.5. The word at the top of the fourth column is 011111. Since $e(011) = 011111$, we decode 010101 as 011. ■

Suppose that the (m, n) group code is $e_H: B^m \to B^n$, where $\mathbf{H}$ is a given parity check matrix. In this case, the decoding technique above can be simplified. We now turn to a discussion of this situation.

Recall from Section 11.1 that $r = n - m$,

$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1r} \\ h_{21} & h_{22} & \cdots & h_{2r} \\ \vdots & \vdots & & \vdots \\ h_{m1} & h_{m2} & \cdots & h_{mr} \\ 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$$

and the function $f_H: B^n \to B^r$ defined by

$$f_H(x) = x * \mathbf{H}$$

is a homomorphism from the group B^n to the group B^r.

Theorem 3 If m, n, r, $\mathbf{H}$, and f_H are as defined, then f_H is onto. ●

Proof Let $b = b_1 b_2 \cdots b_r$ be any element in B^r. Letting

$$x = \underbrace{00\cdots0}_{m\ 0\text{'s}} b_1 b_2 \cdot \cdot b_r$$

we obtain $x * \mathbf{H} = b$. Thus $f_H(x) = b$, so f_H is onto. ▼

It follows from Corollary 1 of Section 9.5 that B^r and B^n/N are isomorphic, where $N = \ker(f_H) = e_H(B^m)$, under the isomorphism $g: B^n/N \to B^r$ defined by

$$g(xN) = f_H(x) = x * \mathbf{H}.$$

The element $x * \mathbf{H}$ is called the **syndrome** of x. We now have the following result.

Theorem 4 Let x and y be elements in B^n. Then x and y lie in the same left coset of N in B^n if and only if $f_H(x) = f_H(y)$, that is, if and only if they have the same syndrome. ●

Proof It follows from Theorem 4 of Section 9.5 that x and y lie in the same left coset of N in B^n if and only if $x \oplus y = (-x) \oplus y \in N$. Since $N = \ker(f_H)$, $x \oplus y \in N$ if and only if

$$\begin{aligned} f_H(x \oplus y) &= \overline{0}_{B^r} \\ f_H(x) \oplus f_H(y) &= \overline{0}_{B^r} \\ f_H(x) &= f_H(y). \end{aligned}$$

▼

In this case, the decoding procedure given previously can be modified as follows. Suppose that we compute the syndrome of each coset leader. If the word x_t is received, we also compute $f_H(x_t)$, the syndrome of x_t. By comparing $f_H(x_t)$ and the syndromes of the coset leaders, we find the coset in which x_t lies. Suppose that a coset leader of this coset is ϵ. We now compute $x = x_t \oplus \epsilon$. If $x = e(b)$, we then decode x_t as b. Thus we need only the coset leaders and their syndromes in order to decode. We state the new procedure in detail.

***STEP 1*:** Determine all left cosets of $N = e_H(B^m)$ in B^n.

***STEP 2*:** For each coset, find a coset leader, and compute the syndrome of each leader.

***STEP 3*:** If x_t is received, compute the syndrome of x_t and find the coset leader ϵ having the same syndrome. Then $x_t \oplus \epsilon = x$ is a code word $e_H(b)$, and $d(x_t) = b$.

For this procedure, we do not need to keep a table of cosets, and we can avoid the work of computing a decoding table. Simply list all cosets once, in any order, and select a coset leader from each coset. Then keep a table of these coset leaders and their syndromes. The foregoing procedure is easily implemented with such a table.

EXAMPLE 5 Consider the parity check matrix

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

and the (3, 6) group $e_H\colon B^3 \to B^6$. Then

$$\left.\begin{array}{l} e(000) = 000000 \\ e(001) = 001011 \\ e(010) = 010101 \\ e(011) = 011110 \\ e(100) = 100110 \\ e(101) = 101101 \\ e(110) = 110011 \\ e(111) = 111000 \end{array}\right\} \text{ code words.}$$

Table 11.6

Syndrome of Coset Leader	Coset Leader
000	000000
001	000001
010	000010
011	001000
100	000100
101	010000
110	100000
111	001100

Thus

$$N = \{000000, 001011, 010101, 011110, 100110, 101101, 110011, 111000\}.$$

We now implement the decoding procedure as follows.

In Table 11.6 we give only the coset leaders together with their syndromes. Suppose now that we receive the word 001110. We compute the syndrome of $x_t = 001110$, obtaining $f_H(x_t) = x_t * \mathbf{H} = 101$, which is the sixth entry in the first column of Table 11.6. This means that x_t lies in the coset whose leader is $\epsilon = 010000$. We compute $x = x_t \oplus \epsilon = 001110 \oplus 010000 = 011110$. Since $e(011) = 011110$, we decode 001110 as 011. ■

11.2 Exercises

1. Let d be the (4, 3) decoding function defined by letting m be 3 in Example 1. Determine $d(y)$ for the word y in B^4.
   (a) $y = 0110$ (b) $y = 1011$

2. Let d be the (6, 5) decoding function defined by letting m be 5 in Example 1. Determine $d(y)$ for the word y in B^6.
   (a) $y = 001101$ (b) $y = 110100$

3. Let d be the (6, 2) decoding function defined in Example 2. Determine $d(y)$ for the word y in B^6.
   (a) $y = 111011$ (b) $y = 010100$

4. Let d be the (9, 3) decoding function defined in the same way as the decoding function in Example 2. Determine $d(y)$ for the word y in B^9.
   (a) $y = 101111101$ (b) $y = 100111100$

In Exercises 5 *through* 10, *let* e *be the indicated encoding function and let* d *be an associated maximum likelihood decoding function. Determine the number of errors that* (e, d) *will correct.*

5. e is the encoding function in Exercise 13 of Section 11.1.

6. e is the encoding function in Exercise 14 of Section 11.1.

7. e is the encoding function in Exercise 15 of Section 11.1.

8. e is the encoding function in Exercise 16 of Section 11.1.

9. e is the encoding function in Exercise 17 of Section 11.1.

10. e is the encoding function in Exercise 18 of Section 11.1.

11. Consider the group code defined in Exercise 17 of Section 11.1 Decode the following words relative to a maximum likelihood decoding function.

(a) 11110 (b) 10011 (c) 10100

12. Consider the (2, 4) group encoding function $e: B^2 \to B^4$ defined by

$$e(00) = 0000 \qquad e(10) = 1001$$
$$e(01) = 0111 \qquad e(11) = 1111.$$

Decode the following words relative to a maximum likelihood decoding function.

(a) 0011 (b) 1011 (c) 1111

13. Consider the (3, 5) group encoding function $e: B^3 \to B^5$ defined by

$$e(000) = 00000 \qquad e(100) = 10011$$
$$e(001) = 00110 \qquad e(101) = 10101$$
$$e(010) = 01001 \qquad e(110) = 11010$$
$$e(011) = 01111 \qquad e(111) = 11100.$$

Decode the following words relative to a maximum likelihood decoding function.

(a) 11001 (b) 01010 (c) 00111

14. Consider the (3, 6) group encoding function $e: B^3 \to B^6$ defined by

$$e(000) = 000000 \qquad e(100) = 100101$$
$$e(001) = 000110 \qquad e(101) = 100011$$
$$e(010) = 010010 \qquad e(110) = 110111$$
$$e(011) = 010100 \qquad e(111) = 110001.$$

Decode the following words relative to a maximum likelihood decoding function.

(a) 011110 (b) 101011 (c) 110010

15. Let G be a group and H a subgroup of G.

(a) Prove that if g_1 and g_2 are elements of G, then either $g_1H = g_2H$ or $g_1H \cap g_2H = \{\ \}$.

(b) Use the result of part (a) to show that the left cosets of H form a partition of G

In Exercises 16 *through* 18, *determine the coset leaders for* $N = e_H(B^m)$ *for the given parity check matrix* $\mathbf{H}$.

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

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

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

In Exercises 19 *through* 21, *compute the syndrome for each coset leader found in the specified exercise.*

19. Exercise 16.

20. Exercise 17.

21. Exercise 18.

22. Let

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

be a parity check matrix. Decode the following words relative to a maximum likelihood decoding function.

(a) 0101 (b) 1010 (c) 1101

23. Let

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

be a parity check matrix. Decode the following words relative to a maximum likelihood decoding function associated with e_H.

(a) 10100 (b) 01101 (c) 11011

24. Let

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

be a parity check matrix. Decode the following words relative to a maximum likelihood decoding function associated with e_H.

(a) 011001 (b) 101011 (c) 111010

TIPS FOR PROOFS

The proofs in this chapter rely heavily on earlier results. Many of the concepts developed throughout the book are applied here to the problems of coding and decoding. In Section 11.1, we pointed out the similarity of proving two numbers equal to proving two sets are the same. Analogous proofs could be developed for any relation that has the antisymmetric property as "is less than" and "is a subset of" do.

In Section 11.2, Theorem 2 we use a one-to-one, onto function to "match" the elements of two sets in order to show that they have the same number of elements. This is also a technique that can be used in solving counting problems if the cardinality of one of the sets used is known.

KEY IDEAS FOR REVIEW

- Message: finite sequence of characters from a finite alphabet
- Word: sequence of 0's and 1's
- (m, n) encoding function: one-to-one function $e: B^m \to B^n$, $m < n$
- Code word: element in Ran(e)
- Weight of x, $|x|$: number of 1's in x
- Parity check code: see page 403
- Hamming distance between x and y, $\delta(x, y)$: $|x \oplus y|$
- Theorem (Properties of the Distance Function): Let x, y, and z be elements of B^m. Then
  (a) $\delta(x, y) = \delta(y, x)$.
  (b) $\delta(x, y) \geq 0$.
  (c) $\delta(x, y) = 0$ if and only if $x = y$.
  (d) $\delta(x, y) \leq \delta(x, z) + \delta(z, y)$.
- Minimum distance of an (m, n) encoding function: minimum of the distances between all distinct pairs of code words
- Theorem: An (m, n) encoding function $e: B^m \to B^n$ can detect k or fewer errors if and only if its minimum distance is at least $k + 1$.
- Group code: (m, n) encoding function $e: B^m \to B^n$ such that $e(B^m) = \{e(b) \mid b \in B^m\}$ is a subgroup of B^n
- Theorem: The minimum distance of a group code is the minimum weight of a nonzero code word.
- Mod-2 sum of Boolean matrices $\mathbf{D}$ and $\mathbf{E}$, $\mathbf{D} \oplus \mathbf{E}$: see page 407
- Mod-2 Boolean product of Boolean matrices $\mathbf{D}$ and $\mathbf{E}$, $\mathbf{D} * \mathbf{E}$: see page 408
- Theorem: Let m and n be nonnegative integers with $m < n$, $r = n - m$, and let $\mathbf{H}$ be an $n \times r$ Boolean matrix. Then the function $f_H: B^n \to B^r$ defined by

$$f_H(x) = x * \mathbf{H}, \qquad x \in B^n$$

  is a homomorphism from the group B^n to the group B^r.
- Group code e_H corresponding to parity check matrix $\mathbf{H}$: see page 409
- (n, m) decoding function: see page 413
- Maximum likelihood decoding function associated with e: see page 414
- Theorem: Suppose that e is an (m, n) encoding function and d is a maximum likelihood decoding function associated with e. Then (e, d) can correct k or fewer errors if and only if the minimum distance is at least $2k + 1$.
- Decoding procedure for a group code: see page 416
- Decoding procedure for a group code given by a parity check matrix: see page 419

CODING EXERCISES

For each of the following, write the requested program or subroutine in pseudocode (as described in Appendix A) or in a programming language that you know. Test your code either with a paper-and-pencil trace or with a computer run.

1. Write a function that finds the weight of a word in B^n.
2. Write a subroutine that computes the Hamming distance between two words in B^n.
3. Let M and N be Boolean matrices of size $n \times n$. Write a program that given M and N returns their mod-2 Boolean product.
4. Write a subroutine to simulate the $(m, 3m)$-encoding function $e\colon B^m \to B^{3m}$ described in Example 3, Section 11.1.
5. Write a subroutine to simulate the decoding function d for the encoding function of Exercise 4 as described in Example 2, Section 11.2.

CHAPTER 11 SELF-TEST

1. Consider the (3, 4) parity check code. For each of the received words, determine whether an error will be detected.
   (a) 1101 (b) 1010 (c) 1111 (d) 0011
2. Consider the $(m, 3m)$ encoding function with $m = 4$. For each of the received words, determine whether an error will be detected.
   (a) 001100100011 (b) 110111001101
   (c) 010111010011
3. Let e be the (3, 5) encoding function defined by

$$\begin{aligned} e(000) &= 0000 & e(100) &= 01010 \\ e(001) &= 11110 & e(101) &= 10100 \\ e(010) &= 01101 & e(110) &= 00111 \\ e(011) &= 10011 & e(111) &= 11001. \end{aligned}$$

   How many errors will e detect?
4. Show that the (3, 5) encoding function in Problem 3 is a group code.
5. Let e be the encoding function defined in Problem 3 and let d be an associated maximum likelihood decoding function. Determine the number of errors that (e, d) will correct.
6. Let

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

   be a parity check matrix. Decode 0110 relative to a maximum likelihood decoding function associated with e_H.