Coding Theory

11

Key Topics

Groups, Rings and Fields
Block Codes
Error Detection and Correction
Generation Matrix
Hamming Codes

11.1 Introduction

Coding theory is a practical branch of mathematics concerned with the reliable transmission of information over communication channels. It allows errors to be detected and corrected, which is essential when messages are transmitted through a noisy communication channel. The channel could be a telephone line, radio link or satellite link, and coding theory is applicable to mobile communications, and satellite communications. It is also applicable to storing information on storage systems such as the compact disc.

It includes theory and practical algorithms for error detection and correction, and it plays an important role in modern communication systems that require reliable and efficient transmission of information.

An error correcting code encodes the data by adding a certain amount of redundancy to the message. This enables the original message to be recovered if a small number of errors have occurred. The extra symbols added are also subject to errors, as accurate transmission cannot be guaranteed in a noisy channel.

G. O'Regan, *Guide to Discrete Mathematics*, Texts in Computer Science,
DOI 10.1007/978-3-319-44561-8_11

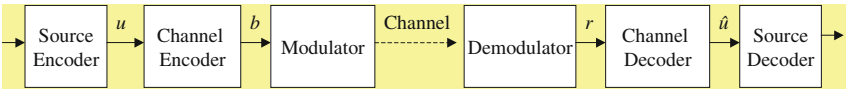


Fig. 11.1 Basic digital communication

The basic structure of a digital communication system is shown in Fig. 11.1. It includes transmission tasks such as source encoding, channel encoding and modulation; and receiving tasks such as demodulation, channel decoding and source decoding.

The modulator generates the signal that is used to transmit the sequence of symbols b across the channel. The transmitted signal may be altered due to the fact that there is noise in the channel, and the signal received is demodulated to yield the sequence of received symbols r.

The received symbol sequence r may differ from the transmitted symbol sequence b due to the noise in the channel, and therefore a channel code is employed to enable errors to be detected and corrected. The channel encoder introduces redundancy into the information sequence u, and the channel decoder uses the redundancy for error detection and correction. This enables the transmitted symbol sequence $\hat{u}$ to be estimated.

Shannon [1] showed that it is theoretically possible to produce an information transmission system with an error probability as small as required provided that the information rate is smaller than the channel capacity.

Coding theory uses several results from pure mathematics, and so first we briefly discuss the mathematical foundations of coding theory.

11.2 Mathematical Foundations

Coding theory is built from the results of modern algebra, and it uses abstract algebraic structures such as groups, rings, fields and vector spaces. These abstract structures provide a solid foundation for the discipline, and the main structures used include vector spaces and fields. A *group* is a non-empty set with a single binary operation, whereas *rings* and *fields* are algebraic structures with two binary operations satisfying various laws. A *vector space* consists of vectors over a field.

We discussed these abstract mathematical structures in Chap. 6, and presented examples of each structure. The representation of codewords is by n-dimensional vectors over the finite field F_q. A codeword vector v is represented as the n-tuple:

$$v = (a_0, a_1, \ldots . a_{n-1})$$

where each $a_i \in F_q$. The set of all n-dimensional vectors is the n-dimensional vector space $\boldsymbol{F}_q^n$ with q^n elements. The addition of two vectors v and w, where $v = (a_0, a_1, \ldots . a_{n-1})$ and $w = (b_0, b_1, \ldots . b_{n-1})$ is given by:

$$v + w = (a_0 + b_0, a_1 + b_1, \ldots a_{n-1} + b_{n-1})$$

The scalar multiplication of a vector $v = (a_0, a_1, \ldots a_{n-1}) \in \boldsymbol{F}^n{}_q$ by a scalar $\beta \in F_q$ is given by:

$$\beta v = (\beta a_0, \beta a_1, \ldots \beta a_{n-1})$$

The set $\boldsymbol{F}^n{}_q$ is called the vector space over the finite field F_q. If the vector space properties above hold. A finite set of vectors v_1, v_2,… v_k is said to be *linearly independent* if

$$\beta_1 v_1 + \beta_2 v_2 + \cdots + \beta_k v_k = 0 \;\Rightarrow\; \beta_1 = \beta_2 = \ldots \beta_k = 0$$

Otherwise, the set of vectors v_1, v_2,…v_k is said to be *linearly dependent*.

The *dimension* (dim *W*) of a subspace $W \subseteq V$ is k if there are k linearly independent vectors in *W* but every $k + 1$ vectors are linearly dependent. A subset of a vector space is a *basis* for *V* if it consists of linearly independent vectors, and its linear span is *V* (i.e., the basis generates *V*). We shall employ the basis of the vector space of codewords to create the generator matrix to simplify the encoding of the information words. The linear span of a set of vectors v_1, v_2, …, v_k is defined as $\beta_1 v_1 + \beta_2 v_2 + \cdots + \beta_k v_k$.

11.3 Simple Channel Code

This section presents a simple example to illustrate the concept of an error correcting code. The example code presented is able to correct a single transmitted error only.

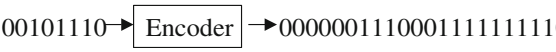
We consider the transmission of binary information over a noisy channel that leads to differences between the transmitted sequence and the received sequence. The differences between the transmitted and received sequence are illustrated by underlining the relevant digits in the example.

Sent	00$\underline{1}$0$\underline{1}$1110
Received	00$\underline{0}$0$\underline{0}$0110

Initially, it is assumed that the transmission is done without channel codes as follows:

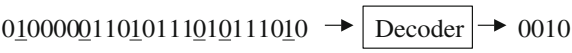
$$00101110 \xrightarrow{\text{Channel}} 00000110$$

Next, the use of an encoder is considered and a triple repetition-encoding scheme is employed. That is, the binary symbol 0 is represented by the code word 000, and the binary symbol 1 is represented by the code word 111.



Another words, if the symbol 0 is to be transmitted then the encoder emits the codeword 000, and similarly the encoder emits 111 if the symbol 1 is to be transmitted. Assuming that on average one symbol in four is incorrectly transmitted, then transmission with binary triple repetition may result in a received sequence such as:

The decoder tries to estimate the original sequence by using a *majority decision* on each 3-bit word. Any 3-bit word that contains more zeros than ones is decoded to 0, and similarly if it contains more ones than zero it is decoded to 1. The decoding algorithm yields:



In this example, the binary triple repetition code is able to correct a single error within a code word (as the majority decision is two to one). This helps to reduce the number of errors transmitted compared to unprotected transmission. In the first case where an encoder is not employed there are two errors, whereas there is just one error when the encoder is used.

However, there are disadvantages with this approach in that the transmission bandwidth has been significantly reduced. It now takes three times as long to transmit an information symbol with the triple replication code than with standard transmission. Therefore, it is desirable to find more efficient coding schemes.

11.4 Block Codes

There were two code words employed in the simple example above: namely 000 and 111. This is an example of a (n, k) code where the code words are of length $n = 3$, and the information words are of length $k = 1$ (as we were just encoding a single symbol 0 or 1). This is an example of a (3, 1) block code, and the objective of this section is to generalize the simple coding scheme to more efficient and powerful channel codes.

The fundamentals of the q-nary (n, k) block codes (where q is the number of elements in the finite field F_q) involve converting an information block of length k to a codeword of length n. Consider an information sequence $u_0, u_1, u_2, \ldots$ of discrete information symbols where $u_i \in \{0, 1, \ldots q - 1\} = F_q$. The normal class of channel codes is when we are dealing with binary codes: i.e., $q = 2$. The information sequence is then grouped into blocks of length k as follows:

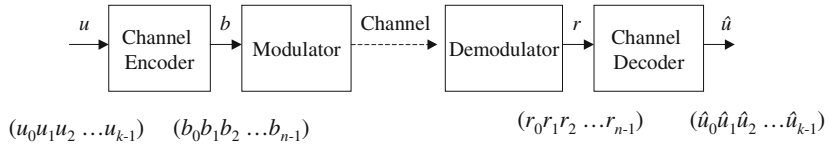


Fig. 11.2 Encoding and decoding of an (n, k) block

$$\underbrace{u_0u_1u_2 \ldots u_{k-1}}\ \underbrace{u_ku_{k+1}u_{k+2} \ldots u_{2k-1}}\ \underbrace{u_{2k}u_{2k+1}u_{2k+2} \ldots u_{3k-1}}\ldots\ldots$$

Each block is of length k (i.e., the information words are of length k), and it is then encoded separately into codewords of length n. For example, the block u_ku_{k+1} $u_{k+2} \ldots u_{2k-1}$ is encoded to the code word $b_nb_{n+1}b_{n+2} \ldots b_{2n-1}$ of length n where $b_i \in F_q$. Similarly, the information word $u_0u_1u_2 \ldots u_{k-1}$ is uniquely mapped to a code word $b_0b_1b_2 \ldots b_{n-1}$ of length n as follows:

$$(u_0u_1u_2 \ldots u_{k-1}) \rightarrow \boxed{\text{Encoder}} \rightarrow (b_0b_1b_2 \ldots b_{n-1})$$

These code words are then transmitted across the communication channel and the received words are then decoded. The received word $r = (r_0r_1r_2 \ldots r_{n-1})$ is decoded into the information word $\hat{u} = (\hat{u}_0\hat{u}_1\hat{u}_2 \ldots \hat{u}_{k-1})$.

$$(r_0r_1r_2 \ldots r_{n-1}) \rightarrow \boxed{\text{Decoder}} \rightarrow (\hat{u}_0\hat{u}_1\hat{u}_2 \ldots \hat{u}_{k-1})$$

Strictly speaking the decoding is done in two steps with the received n-block word r first decoded to the n-block codeword b^*. This is then decoded into the k-block information word $\hat{u}$. The encoding, transmission and decoding of an (n, k) block may be summarized as follows (Fig. 11.2):

A lookup table may be employed for the encoding to determine the code word b for each information word u. However, the size of the table grows exponentially with increasing information word length k, and so this is inefficient due to the large memory size required. We shall discuss later how a generator matrix provides an efficient encoding and decoding mechanism.

Notes

(i) The codeword is of length n.
(ii) The information word is of length k.
(iii) The codeword length n is larger than the information word length k.
(iv) A block (n, k) code is a code in which all codewords are of length n and all information words are of length k.
(v) The number of possible information words is given by $M = q^k$ (where each information symbol can take one of q possible values and the length of the information word is k).

(vi) The code rate R in which information is transmitted across the channel is given by:

$$R = \frac{k}{n}$$

(vii) The weight of a codeword is $b = (b_0 b_1 b_2 \ldots b_{n-1})$ is given by the number of non-zero components of b. That is,

$$\mathrm{wt}(b) = |\{i : b_i \neq 0,\ 0 \leq i < n\}|$$

(viii) The distance between two codewords $b = (b_0 b_1 b_2 \ldots b_{n-1})$ and $b' = (b_0' b_1' b_2' \ldots b_{n-1}')$ measures how close the codewords b and b' are to each other. It is given by the Hamming distance:

$$\mathrm{dist}(b, b') = |\{i : b_i \neq b_i',\ 0 \leq i < n\}|$$

(ix) The minimum Hamming distance for a code $\boldsymbol{B}$ consisting of M codewords $b_1, \ldots, b_M$ is given by:

$$d = \min\{\mathrm{dist}(b, b') : \text{where } b \neq b'\}$$

(x) The (n, k) block code $B = \{b_1, \ldots, b_M\}$ with M $(=q^k)$ codewords of length n and minimum Hamming distance d is denoted by $\boldsymbol{B}(n, k, d)$.

11.4.1 Error Detection and Correction

The minimum Hamming distance offers a way to assess the error detection and correction capability of a channel code. Consider two codewords b and b' of an (n, k) block code $\boldsymbol{B}(n, k, d)$.

Then, the distance between these two codewords is greater than or equal to the minimum Hamming distance d, and so errors can be detected as long as the erroneously received word is not equal to a codeword different from the transmitted code word.

That is, the *error detection capability* is guaranteed as long as the number of errors is less than the minimum Hamming distance d, and so the number of detectable errors is $d - 1$.

Any two codewords are of distance at least d and so if the number of errors is less than $d/2$ then the received word can be properly decoded to the codeword b. That is, the *error correction capability* is given by:

$$E_{\mathrm{cor}} = \frac{d-1}{2}$$

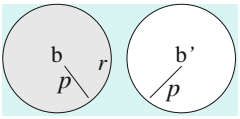


Fig. 11.3 Error correcting capability sphere

An error-correcting sphere (Fig. 11.3) may be employed to illustrate the error correction of a received word to the correct codeword b. This may be done when all received words are within the error-correcting sphere with radius p ($<d/2$).

If the received word r is different from b in less than $d/2$ positions, then it is decoded to b as it is more than $d/2$ positions from the next closest codeword. That is, b is the closest codeword to the received word r (provided that the error-correcting radius is less than $d/2$).

11.5 Linear Block Codes

Linear block codes have nice algebraic properties, and the codewords in a linear block code are considered to be vectors in the finite vector space $\boldsymbol{F}^n_q$. The representation of codewords by vectors allows the nice algebraic properties of vector spaces to be used, and this simplifies the encoding of information words as a generator matrix may be employed to create the codewords.

An (n, k) block code $\boldsymbol{B}(n, k, d)$ with minimum Hamming distance d over the finite field F_q is called *linear* if $\boldsymbol{B}(n, k, d)$ is a subspace of the vector space $\boldsymbol{F}^n_q$ of dimension k. The number of codewords is then given by:

$$M = q^k$$

The rate of information (R) through the channel is given by:

$$R = \frac{k}{n}$$

Clearly, since $\boldsymbol{B}(n, k, d)$ is a subspace of $\boldsymbol{F}^n_q$ any linear combination of the codewords (vectors) will be a codeword. That is, for the codewords $b_1, b_2, \ldots, b_r$ we have that:

$$\alpha_1 b_1 + \alpha_2 b_2 + \cdots + \alpha_r b_r \in \boldsymbol{B}(n, k, d)$$

where $\alpha_1, \alpha_2, \ldots, \alpha_r \in F_q$ and $b_1, b_2, \ldots, b_r \in \boldsymbol{B}(n, k, d)$.

Clearly, the n-dimensional zero row vector (0,0, …, 0) is always a codeword, and so (0, 0, …, 0) $\in \boldsymbol{B}(n, k, d)$. The minimum Hamming distance of a linear block code $\boldsymbol{B}(n, k, d)$ is equal to the minimum weight of the non-zero codewords: That is,

$$d = \min_{\forall b \neq b'} \{\text{dist}(b, b')\} = \min_{\forall b \neq 0} \text{wt}(b)$$

In summary, an (n, k) linear block code $\mathbf{B}(n, k, d)$ is:

1. A subspace of $\boldsymbol{F}_q^n$.
2. The number of codewords is $M = q^k$.
3. The minimum Hamming distance d is the minimum weigh of the non-zero codewords.

The encoding of a specific k-dimensional information word $u = (u_0, u_1, \ldots. u_{k-1})$ to a n-dimensional codeword $b = (b_0, b_1, \ldots., b_{n-1})$ may be done efficiently with a generator matrix. First, a basis $\{g_0, g_1, \ldots. g_{k-1}\}$ of the k-dimensional subspace spanned by the linear block code is chosen, and this consists of k linearly independent n-dimensional vectors. Each basis element g_i (where $0 \leq i \leq k - 1$) is a n-dimensional vector:

$$\mathrm{g}_i = (g_{i,0}, g_{i,1}, \ldots., g_{i,n-1})$$

The corresponding codeword $b = (b_0, b_1, \ldots, b_{n-1})$ is then a linear combination of the information word with the basis elements. That is,

$$\mathrm{b} = u_0\mathrm{g}_0 + u_1\mathrm{g}_1 + \cdots + u_{k-1}\mathrm{g}_{k-1}$$

where each information symbol $u_i \in F_q$. The generator matrix G is then constructed from the k linearly independent basis vectors as follows (Fig. 11.4):

The encoding of the k-dimensional information word u to the n-dimensional codeword b involves matrix multiplication (Fig. 11.5):

This may also be written as:

$$b = uG$$

$$G = \begin{pmatrix} \mathrm{g}_{0,} \\ \mathrm{g}_{1,} \\ \mathrm{g}_{2,} \\ \ldots. \\ \ldots. \\ \ldots. \\ \mathrm{g}_{k\text{-}1} \end{pmatrix} = \begin{pmatrix} g_{0,0} & g_{0,1} & g_{0,2} & \ldots. & g_{0,n\text{-}1} \\ g_{1,0} & g_{1,1} & g_{1,2} & \ldots. & g_{1,n\text{-}1} \\ g_{2,0} & g_{2,1} & g_{2,2} & \ldots. & g_{2,n\text{-}1} \\ \ldots & \ldots. & \ldots. & \ldots. & \ldots. \\ \ldots & \ldots. & \ldots. & \ldots. & \ldots. \\ \ldots & \ldots. & \ldots. & \ldots. & \ldots. \\ g_{k\text{-}1,0} & g_{k\text{-}1,1} & g_{k\text{-}1,2} & \ldots. & g_{k\text{-}1,n\text{-}1} \end{pmatrix}$$

Fig. 11.4 Generator matrix

$$(u_0, u_1, \ldots, u_{k-1}) \begin{pmatrix} g_{0,0} & g_{0,1} & g_{0,2} & \cdots & g_{0,n-1} \\ g_{1,0} & g_{1,1} & g_{1,2} & \cdots & g_{1,n-1} \\ g_{2,0} & g_{2,1} & g_{2,2} & \cdots & g_{2,n-1} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ g_{k-1,0} & g_{k-1,1} & g_{k-1,2} & \cdots & g_{k-1,n-1} \end{pmatrix} = (b_0, b_1, \ldots, b_{n-1})$$

Fig. 11.5 Generation of codewords

$$\mathrm{I}_k = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 1 \end{pmatrix}$$

Fig. 11.6 Identity matrix ($k \times k$)

Clearly, all $M = q^k$ codewords $b \in \boldsymbol{B}(n, k, d)$ can be generated according to this rule, and so the matrix G is called the generator matrix. The generator matrix defines the linear block code $\boldsymbol{B}(n, k, d)$.

There is an equivalent $k \times n$ generator matrix for $\boldsymbol{B}(n, k, d)$ defined as:

$$\mathrm{G} = \mathrm{I}_k | \ \mathrm{A}_{k,n-k}$$

where I_k is the $k \times k$ identity matrix (Fig. 11.6):

The encoding of the information word u yields the codeword b such that the first k symbols b_i of b are the same as the information symbols u_i $0 \leq i \leq k$.

$$b = uG = \left(u \,|\, u A_{k,n-k}\right)$$

The remaining $m = n - k$ symbols are generated from $uA_{k,n-k}$ and the last m symbols are the m parity check symbols. These are attached to the information vector u for the purpose of error detection and correction.

11.5.1 Parity Check Matrix

The linear block code $\boldsymbol{B}(n, k, d)$ with generator matrix $G = (I_k, |A_{k,n-k})$ may be defined equivalently by the $(n - k) \times n$ parity check matrix H, where this matrix is defined as:

$$H = \left(-A^T_{k,n-k} \,|\, I_{n-k}\right).$$

The generator matrix G and the parity check matrix H are orthogonal: i.e.,

$$HG^T = 0_{n-k,k}$$

The parity check orthogonality property holds if and only if the vector belongs to the linear block code. That is, for each code vector in $b \in \boldsymbol{B}(n, k, d)$ we have

$$Hb^T = 0_{n-k,1}$$

and vice verse whenever the property holds for a vector r, then r is a valid codeword in $\boldsymbol{B}(n, k, d)$. We present an example of a parity check matrix in Example 9.5 below.

11.5.2 Binary Hamming Code

The Hamming code is a linear code that has been employed in dynamic random access memory to detect and correct deteriorated data in memory. The generator matrix for the $\boldsymbol{B}(7, 4, 3)$ binary Hamming code is given by (Fig. 11.7):

The information words are of length $k = 4$ and the codewords are of length $n = 7$. For example, it can be verified by matrix multiplication that the information word (0, 0, 1, 1) is encoded into the codeword (0, 0, 1, 1, 0, 0, 1).

That is, three parity bits 001 have been added to the information word (0, 0, 1, 1) to yield the codeword (0, 0, 1, 1, 0, 0, 1).

The minimum Hamming distance is $d = 3$, and the Hamming code can detect up to two errors, and it can correct one error.

Example 9.5 (**Parity Check Matrix—Hamming Code**) The objective of this example is to construct the Parity Check Matrix of the Binary Hamming Code (7, 4, 3), and to show an example of the parity check orthogonality property.

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

Fig. 11.7 Hamming code $B(7, 4, 3)$ generator matrix

First, we construct the parity check matrix H which is given by $H = \left(-A^T_{k,n-k} \,|\, I_{n-k}\right)$ or another words $H = \left(-A^T_{4,3} \,|\, I_3\right)$. We first note that

$$A_{4,3} = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix} \quad A^T_{4,3} = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \end{pmatrix}$$

Therefore, H is given by:

$$H = \begin{pmatrix} 0 & -1 & -1 & -1 & 1 & 0 & 0 \\ -1 & 0 & -1 & -1 & 0 & 1 & 0 \\ -1 & -1 & 0 & -1 & 0 & 0 & 1 \end{pmatrix}$$

We noted that the encoding of the information word u = (0011) yields the codeword b = (0011001). Therefore, the calculation of Hb^T yields (recalling that addition is modulo two):

$$Hb^T = \begin{pmatrix} 0 & -1 & -1 & -1 & 1 & 0 & 0 \\ -1 & 0 & -1 & -1 & 0 & 1 & 0 \\ -1 & -1 & 0 & -1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

11.5.3 Binary Parity-Check Code

The binary parity-check code is a linear block code over the finite field F_2. The code takes a k-dimensional information word $u = (u_0, u_1, \ldots. u_{k-1})$ and generates the codeword $b = (b_0, b_1, \ldots., b_{k-1}, b_k)$ where $u_i = b_i$ $(0 \leq i \leq k - 1)$ and b_k is the parity bit chosen so that the resulting codeword is of even parity. That is,

$$b_k = u_0 + u_1 + \ldots. + u_{k-1} = \sum_{i=0}^{k-1} u_i$$

11.6 Miscellaneous Codes in Use

There are many examples of codes in use such as repetition codes (such as the triple replication code considered earlier in Sect. 11.3); parity check codes where a parity symbol is attached such as the binary parity-check code; Hamming codes such as the (7, 4) code that was discussed in Sect. 11.5.2, and which has been applied for error correction of faulty memory.

The Reed-Muller codes form a class of error correcting codes that can correct more than one error. Cyclic codes are special linear block codes with efficient algebraic decoding algorithms. The BCH codes are an important class of cyclic codes, and the Reed Solomon codes are an example of a BCH code.

Convolution codes have been applied in the telecommunications field, for example, in GSM, UMTS and in satellite communications. They belong to the class of linear codes, but also employ a memory so that the output depends on the current input symbols and previous input. For more detailed information on coding theory see [2].

11.7 Review Questions

1. Describe the basic structure of a digital communication system.
2. Describe the mathematical structure known as the field. Give examples of fields.
3. Describe the mathematical structure known as the ring and give examples of rings. Give examples of zero divisors in rings.
4. Describe the mathematical structure known as the vector space and give examples.
5. Explain the terms linear independence and linear dependence and a basis.
6. Describe the encoding and decoding of an (n, k) block code where an information word of length k is converted to a codeword of length n.
7. Show how the minimum Hamming distance may be employed for error detection and error correction.
8. Describe linear block codes and show how a generator matrix may be employed to generate the codewords from the information words.

11.8 Summary

Coding theory is the branch of mathematics that is concerned with the reliable transmission of information over communication channels. It allows errors to be detected and corrected, and this is extremely useful when messages are transmitted through a noisy communication channel. This branch of mathematics includes theory and practical algorithms for error detection and correction.

The theoretical foundations of coding theory were considered, and its foundations lie in abstract algebra including group theory, ring theory, fields and vector spaces. The codewords are represented by n-dimensional vectors over a finite field F_q.

An error correcting code encodes the data by adding a certain amount of redundancy to the message so that the original message can be recovered if a small number of errors have occurred.

The fundamentals of block codes were discussed where an information word is of length k and a codeword is of length n. This led to the linear block codes $\boldsymbol{B}(n, k, d)$ and a discussion on error detection and error correction capabilities of the codes.

The goal of this chapter was to give a flavour of coding theory, and the reader is referred to more specialised texts (e.g., [2]) for more detailed information.

References

1. A Mathematical Theory of Communication. Claude Shannon. Bell System Technical Journal, vol. 27, pp. 379–423. 1948.
2. Coding Theory. Algorithms, Architectures and Applications. André Neubauer, Jürgen Freunderberger and Volker Kühn. John Wiley & Sons. 2007.