CHAPTER THREE: CHANNEL CODING

1. Channel Coding Theorem

A basic block diagram for the channel coding is shown in Fig.1.1

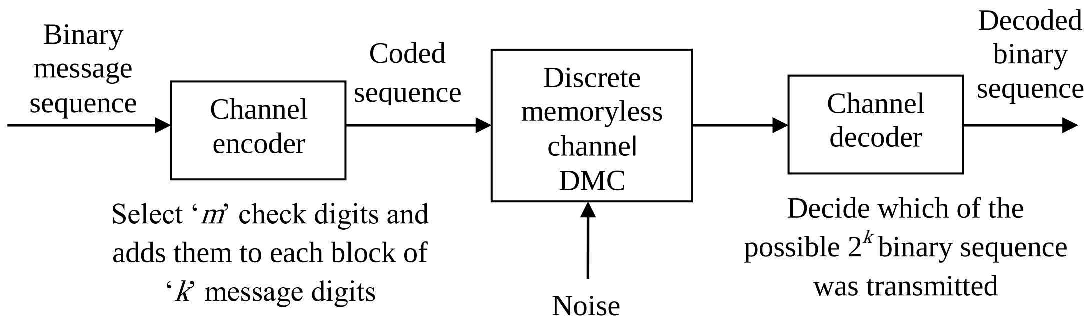

Fig.1.1

The binary message sequence at the input of the channel encoder may be the output of a source encoder or the output of a source directly. The channel encoder introduces systematic redundancy into the data stream by adding bits to the message bits in such a way as to facilitate the detection and/or correction of bit errors in the original binary message sequence at the receiver.

Channel coding theorem for a *DMC* is stated as follows:

Given a DMS X with entropy $H(X)$ bit/symbol and a DMC with capacity C_s bit/symbol, if $H(X) \leq C_s$, there exist a coding scheme for which the source output can be transmitted over the channel with an arbitrary small probability of error.

Conversely, if $H(X) > C_s$, it is not possible to transmit information over the channel with an arbitrarily small probability of error. Note that the channel coding theorem only asserts the existence of codes; it does not tell us how to construct these codes.

2. Block Coding

In block codes, the sequence of the message digits is divided into sequential blocks. Each block contains 'k' binary digits. The encoder add to each block 'm' check digits and so the codeword contains n=k+m coded digits. For any 'n', there are 2^n possible binary sequences. Only 2^k of these are codewords, since for any 'k' sequence of message digits, the 'm' check digits are uniquely determined. This set of 2^k codewords is called an (n,k) block code.

After transmission through the channel, noise may change the codewords, and any of the possible 2^n sequences may arrive at the receiver. A decoder must be provided at the receiver in order to decide which of the possible 2^k codewords was transmitted.

3. Single Parity Check Codes

This is the simplest example of an ***error detection code***. These codes have $m = 1$, and the check digits is taken to be modulo-2 addition of the 'k' message digits (here $k = n - 1$).

In general, the check digits is taken to be '0' or '1' depending on whether the message digits contain an even or odd number of 1's, respectively. Then the total number of 1's in every transmitted codeword is even (for even parity code) and odd (for odd parity code).

Example 3.1: for $k = 2$, the possible codewords are

00**0**, 01**1**, 10**1** and 11**0**

For even party check code, if the received block has odd number of 1's, then an error occurs (1 error, or 3 errors, or 5 errors, …etc.). However, if the number of 1's is even then either no error occurs or an even number of errors.

Suggest that the uncoded bit error probability (prob. of error/bit) $= p = 10^{-3}$

P(Undetected error) $= P_{ue} = \sum_{i=even}^{n} C_i^n (p)^i (1-p)^{n-i} = C_2^n (10^{-3})^2 (1-10^{-3})^{n-2} + C_4^n (10^{-3})^4 (1-10^{-3})^{n-4} + \cdots$

P (Detecting an error) $= P_{de} = \sum_{i=odd}^{n} C_i^n (p)^i (1-p)^{n-i} = C_1^n (10^{-3})^1 (1-10^{-3})^{n-1} + C_3^n (10^{-3})^3 (1-10^{-3})^{n-3} + \cdots$

4. Binary Repetition Codes

The simplest example of an ***error correcting code*** is the binary repetition code. Each single message digit is transmitted along with 'm' check digits, each check digit having the same value as the message digit. Thus k=1 and $n = k + m = 1 + m$

The decoder operates on the following majority decision rule:

No. of 1's received < No. of 0's $\Rightarrow$ 0 was transmitted

No. of 1's received > No. of 0's $\Rightarrow$ 1 was transmitted

No. of 1's received = No. of 0's $\Rightarrow$ no decision

Example 4.1: show that the use of binary repetition code with $n = 3$ reduces the probability of error over BSC with $p = 0.1$.

Solution:

- without channel coding $P_e(uncoded) = p = 0.1$
- with channel coding, each message is repeated three times ($n = 3$). Thus the codewords are (000 and 111)

therefore $P_e(coded) = P(2\ incorrect\ digits\ or\ 3\ incorrect\ digits)$

$$= P(2\ incorrect) + P(3\ incorrect)$$
$$= C_2^3 p^2 (1-p)^{3-2} + C_3^3 p^3 (1-p)^{3-3}$$
$$= \frac{3!}{2!}(0.1)^2(0.9)^1 + (0.1)^3 = 0.028$$

In general, for binary repetition code

$$P_e(coded) = \sum_{i=\frac{n+1}{2}}^{n} C_i^n (p)^i (1-p)^{n-i} \qquad (Bernoulli\ trials)$$

5. Information Rate

The information rate R_c of a code is defined to be:

$$R_c = \frac{k}{n} = \frac{k}{k+m} < 1 \qquad (5.1)$$

The repetition codes have an enormous error correction capability for large values of 'm'. However, there information rate, $R_c = 1/(m+1)$, then becomes very low. i.e., a large number of redundant check digits are transmitted with each message digit. On the other hand, the single parity check code has a very high information rate of, $R_c = k/(k+1)$, but can do nothing more than detect an odd number of errors.

In general, the most useful codes lie between these two extreme cases, and have both moderate information rate and error correction capabilities.

6. Linear Parity-Check Codes

In an (n, k) block code, it is convenient to represent a binary codeword in matrix form as a row vector whose elements are the code symbols. Thus we define a code vector $\boldsymbol{c}$ and a data vector $\boldsymbol{d}$ as follows:

$$\boldsymbol{c} = [c_1\ c_2\ \ldots\ c_n]$$
$$\boldsymbol{d} = [d_1\ d_2\ \ldots\ d_k]$$