

## Lecture 13

The purpose of this lecture is to give a brief overview about error control coding for point to point channels, since some of the surveys are on this topic. This lecture is by no means exhaustive. We only present some basic bounds on the sizes of codes. For more detailed information, please refer to standard texts such as *Error Control Coding, 2<sup>nd</sup> Ed., Prentice Hall, 2004* by authors S. Lin and D. J. Costello.

### 1 Error Control Coding

We model a point to point link as a blackbox that takes in a block of symbols  $\{x_i\}_{i=1}^n$  and outputs a block of symbols  $\{y_i\}_{i=1}^n$  where  $n$  is the length of the block. We shall assume that the  $x_i$ 's and the  $y_i$ 's belong to a discrete set. We shall also implicitly assume that they are from a finite field  $GF(q)$ . There are many different channel models that are possible such as discrete memoryless channels (DMC's) where each symbol is affected independent of others or models with memory where the effect of the channel on adjacent input symbols is similar.

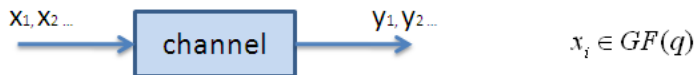


Figure 1: A point to point link model.

Typically the channel introduces some noise so that  $y_i \neq x_i$  at least for some indices  $i \in \{1, \dots, n\}$ . The objective of error control coding is to introduce some redundancy in the input to the channel so that we can recover the message error-free (or almost error-free) at the channel output after some processing.

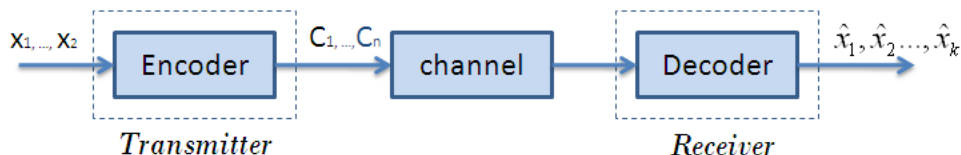


Figure 2: A point to point communication system with coding

Towards this end we introduce an encoder and a decoder into our system as shown in Fig. 2. The encoder is a mapping from  $k$  input symbols  $\{x_i\}_{i=1}^k$  to  $n$  coded symbols  $\{c_i\}_{i=1}^n$  where  $n \geq k$  (i.e.  $(n - k)$  symbols of redundancy are added) and the decoder is a mapping from the  $n$  coded symbols to  $k$  decoded symbols  $\{\hat{x}_i\}_{i=1}^k$ . The vector  $[c_1 \ c_2 \ \dots \ c_n]$  is called a codeword. The hope is to design encoders and decoders so that  $x_i = \hat{x}_i$  for all  $i = 1 \dots k$ .

Our channel model for this lecture will be somewhat simple. We assume that the channel is such that it introduces at most  $t$  errors over a block of length  $n$  i.e.  $y_i \neq c_i$  for at most  $t$  locations.

## 1.1 Bounds on the size of codes

For vectors  $v$  and  $w$ , the Hamming distance between them, denoted by  $d_H(v, w)$  is the number of positions where they differ. The encoder maps the space of all possible  $q^k$  messages to  $q^k$  codewords that lie in a space of size  $q^n$ . We shall denote the set of codewords by  $\mathcal{C}$ .

The minimum distance of a code denoted  $d_{\min}$  is the minimum Hamming distance between any two codewords.

$$d_{\min} = \min_{c_i, c_j \in \mathcal{C}} d_H(c_i, c_j)$$

It turns out that the maximum error correction capability depends critically on the minimum distance of the code. If  $d_{\min} = 2t + 1 \implies$  then we are guaranteed that we can correct  $t$  errors. To see this note that if a codeword  $c \in \mathcal{C}$  is transmitted and the channel introduces at most errors so that at the receiver a errored vector  $y$  is obtained, then

$$d_H(c, y) < d_H(c', y) \text{ for all } c' \in \mathcal{C}, c' \neq c.$$

Therefore there at the receiver we can declare the transmitted codeword to be  $c$  without any ambiguity. In general even if there are more than  $t$  errors the above condition might hold, but this is not guaranteed.

One can also detect errors i.e. detect if the channel introduced any errors instead of correcting them. This can be done by simply checking whether the received word is a valid codeword. By an argument similar to the one made above if  $d_{\min} = 2t + 1$ , then we can detect up to  $2t$  errors.

Clearly there is a trade-off between the number of messages  $q^k$  and the number of errors that can be corrected for a fixed  $n$ . We now present some bounds on the sizes of codes.

**Definition 1.** *Hamming sphere.* The Hamming sphere centered at  $c$  (where  $c$  belongs to the  $n$ -dimensional space over  $GF(q)$ ) of radius  $t$  is the set of points that lies within Hamming distance  $t$  of  $c$ . Let the sphere be denoted  $S_c^{(t)}$ . Then the volume of  $S_c^{(t)}$  is given by

$$S_c^{(t)} = \{\vec{x} \mid d_H(c, x) \leq t\} \mid S_c^{(t)} \mid = \sum_{j=0}^t \binom{n}{j} (q-1)^j. \quad (1)$$

### 1.1.1 Hamming Bound

A necessary condition for a code to be  $t$ -error correcting is that there should be no other codeword within the radius- $t$  Hamming sphere of a given codeword i.e. the number of codewords is at most  $\frac{q^n}{\text{vol}(S_c^{(t)})}$ . i.e

$$\begin{aligned} q^k &\leq \frac{q^n}{\text{vol}(S_c^{(t)})} \\ \Rightarrow \text{vol}(S_c^{(t)}) &\leq q^{n-k} \\ \Rightarrow \log_q \text{vol}(S_c^{(t)}) &\leq n - k \end{aligned}$$

i.e. we obtain a bound on the minimum redundancy required in the code for it to be  $t$ -error correcting. This bound is known as the Hamming bound and is a necessary condition on the code to have a certain error correction capability.

### 1.1.2 Gilbert Bound

We now present a sufficient condition on the code to be  $t$ -error correcting. This will be done by presenting a constructive procedure for finding a code that is  $t$ -error correcting. Consider the following algorithm.

1. Pick  $\vec{c}_1$  from  $q^n$ .
2. Choose  $\vec{c}_2$  from  $\{S_{c_1}^{(2t)}\}^c$ .
3. While it is possible to do, given  $\vec{c}_1, \vec{c}_2, \dots, \vec{c}_{j-1}$  choose  $\vec{c}_j$  from

$$\{S_{c_1}^{(2t)} \cup S_{c_2}^{(2t)} \cup \dots \cup S_{c_{j-1}}^{(2t)}\}^c.$$

In order to analyze the performance of this algorithm, we first need to figure out, how many codewords we can find. Clearly, each codeword  $c_i$  rules out at most  $S_{c_i}^{(2t)}$  other choices for future codewords. Therefore we can find at least  $\frac{q^n}{\text{vol}(S_{c_i}^{(2t)})}$  codewords. i.e.

$$q^k \geq \frac{q^n}{\text{vol}(S_{c_i}^{(2t)})} \tag{2}$$

$$\implies \log_q \text{vol}(S_{c_i}^{(2t)}) \geq n - k. \tag{3}$$

i.e. we obtain an upper bound on the redundancy required in a code for it to be  $t$ -error correcting. This is called the Gilbert bound. It is important to note that though the Gilbert bound gives an extremely simple construction of codes that have good error correction capability, this does not mean that the problem of error control code design is solved. One of the main problems in coding theory is the design of efficient encoding and decoding algorithms for codes with good error correction power. The decoding complexity of the codes constructed above will in general be very high.

### 1.2 Singleton Bound

The Singleton bound is also a necessary condition on the redundancy required in the code for a given error correction power. Here we state it in terms of the minimum distance of the code.

**Lemma 1.** *For any code  $d_{\min} \leq n - k + 1$ .*

*Proof:* Suppose that the number of codewords in the code is  $s = q^k$ . consider a codeword  $C_1 = [C_{11} \ C_{12} \ \dots \ C_{1n}]$ . Suppose we project  $C_1$  onto its first  $(k - 1)$  coordinates. Note that the total number of vectors of length  $k - 1$  is  $q^{k-1}$ . Therefore, at least two codewords have the same projection onto the first  $(k - 1)$  coordinates (also known as the *pigeon hole principle*), which implies that the number of positions where these two codewords differ is at most  $n - (k - 1) = n - k + 1$ . This means that  $d_{\min} \leq n - k + 1$ . Codes that meet this bound are called maximum-distance separable (MDS) codes. We have already seen an instance of these codes previously, namely Reed-Solomon codes.