

Introduction to Communications

Lecture 29: *Channel Coding*

This lecture:

1. Mutual Information.
2. Channel Capacity.
3. The Hamming Code.

Ref: CCR pp. 713–719, 549–550 (& 560–567), *A Mathematical Theory of Communication*.



Conditional Entropy & Mutual Information

The aim is to see that channels have a *capacity* and that capacity is obtained through a quantity called *mutual information*.

Notation

Consider r.v.s X and Y .

- To discriminate between the somewhat cumbersome $P(X = x)$ and $P(Y = y)$ we simply write $P(x)$, $P(y)$.
- Similarly, we write $P(x, y)$ for the joint probability $P(X = x, Y = y)$.
- We write $P(x | y)$ for the conditional probability $P(X = x | Y = y)$.
 - Conversely, $P(y | x)$ for $P(Y = y | X = x)$



Conditional Entropy

We define the *conditional entropy* or *equivocation* of X given Y as

$$H(X | Y) = E \left[\log \frac{1}{P(x | y)} \right] = \sum_{x,y} P(x, y) \log \frac{1}{P(x | y)}.$$

- This is a measure of the amount of uncertainty or information about X that remains if we are first told Y .
- If X is a function of Y then $H(X | Y) = 0$.
- If X is independent of Y then $H(X | Y) = H(X)$.
- Indeed, it can be shown that

$$0 \leq H(X | Y) \leq H(X).$$



Mutual Information

The *mutual information* between X and Y is defined as

$$I(X; Y) = H(X) - H(X | Y).$$

- The mutual information is therefore a measure of how much the uncertainty about X is *reduced* when we are told Y .
 - Or how much information we *learn* about X given Y .
- It turns out that it is also true that

$$I(X; Y) = I(Y; X) = H(Y) - H(Y | X).$$

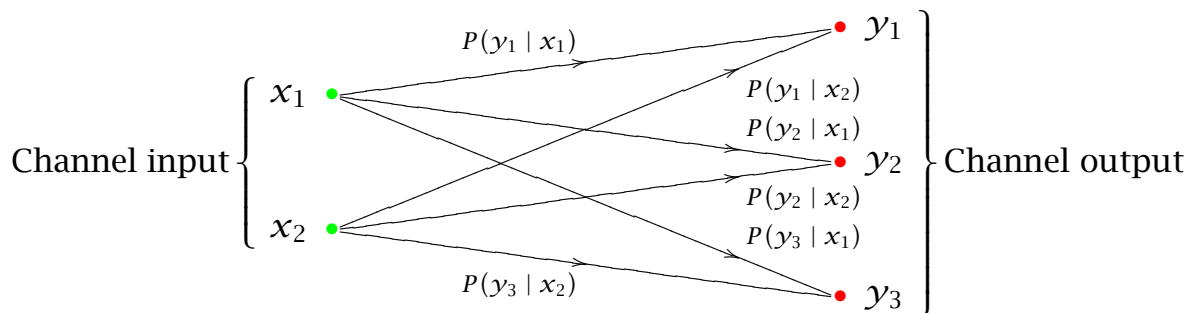
- That is, whatever information we learn about X given Y is also the amount we learn about Y given X .
 - \Rightarrow The information is *mutual*.



The Discrete, Memoryless Channel

Let's return again to our model of the channel.

- Here, we'll assume a *discrete, memoryless channel (DMC)*.
- It is *discrete* in the sense that the input and the output of the channel belong to discrete alphabets (not necessarily the same).
- It is *memoryless* in the sense that the channel output on the current use is independent of previous uses.
- The output Y is a r.v. that is conditioned on the input.





Channel Capacity

Suppose the input to the channel is itself a r.v. X .

- From observing the output Y , we will obtain $I_2(X; Y)$ bits of information about X .
- Thus, this is the amount of information being transmitted through the channel at each use.
- The information transferred is affected by the input probability distribution $P(x_i)$.
- The *channel capacity* is the maximum possible value of mutual information, *i.e.*,

$$C = \max_{P(x_i)} I(X; Y).$$

- Shannon showed that, by using a channel coder and decoder, a bit rate arbitrarily close to C_2 can be achieved per channel use with arbitrarily low BER.



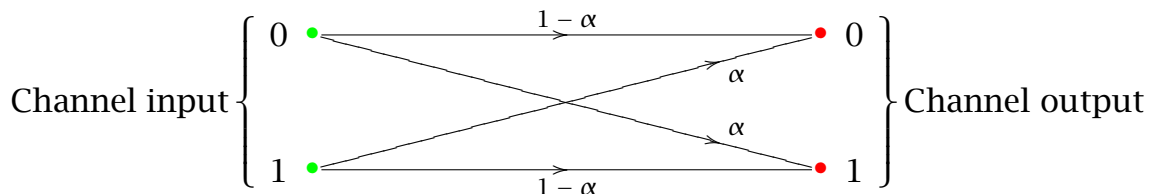
- Conversely, any scheme which attempts to transmit more than C_2 bits per channel use will have a BER bounded away from zero.
- This is Shannon's *channel coding theorem* (and converse).
- If we use the channel s times per second, the channel capacity is clearly sC_2 bits per second.

The Binary Symmetric Channel

The *binary symmetric channel (BSC)* is a DMC with binary input, binary output and a symmetric probability of error.

- By symmetric, we mean that

$$P(Y = 1 | X = 0) = P(Y = 0 | X = 1) = \alpha.$$



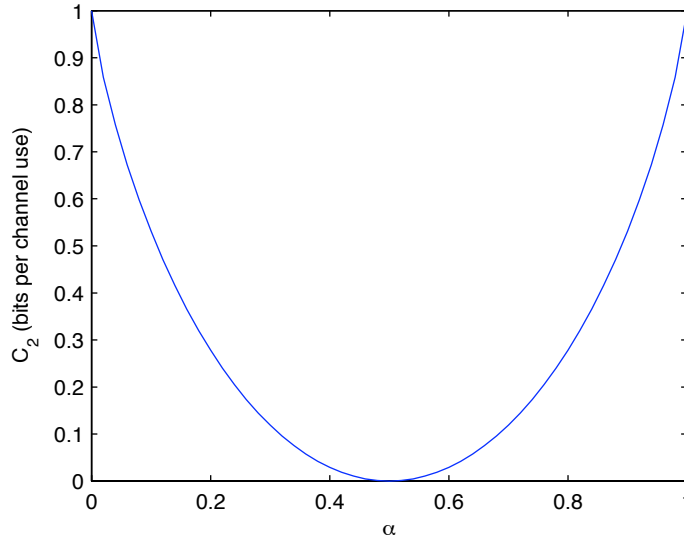
- This is a good model for the NRZ-modulated channel with additive Gaussian noise that we studied in Lecture 25.





It turns out that the capacity is

$$C_2 = 1 + \alpha \log_2 \alpha + (1 - \alpha) \log_2(1 - \alpha) \text{ bits/channel use.}$$





Shannon's Channel Coding Procedure

In proving the channel coding theorem, Shannon devised a simple but impractical channel coding scheme.

- As for source coding, we use blocking—we will use the channel n times in a block.
- Consider all bit sequences of length nC_2 (round up).
- To each bit sequence, *randomly* assign a sequence of n channel input symbols (using a suitable probability distribution).
 - This constitutes the *codebook*.
- To perform channel coding, divide the input bitstream into blocks, consult the codebook & transmit the corresponding symbols.
- To decode, find the most likely transmitted symbols in the codebook for those received & output the corresponding bit sequence.
- With this procedure, $\text{BER} \rightarrow 0$ as $n \rightarrow \infty$.
- As for source coding, the codebook lacks structure, quickly becomes huge and is prohibitively slow to search.



Practical Channel Coding

From Shannon's paper, there followed 50 and more years of effort to find practical channel codes.

- For the coding theorem to work, n must go to infinity.
- For finite n , the BER will be non-zero.
- The larger n is made, the smaller the BER can be made.
- We need to wait for a block of n symbols to be received before we can start decoding.
- This exposes another design parameter: the amount of *latency* or *delay* we are prepared to accept in our communication system.

Channel Coding on the BSC

On the BSC, a very popular method for channel coding is to divide the n -bit blocks into two 'sub-blocks'.





- The size of one sub-block is k bits and the other $n - k$.
- The first k *message* bits are copied directly (unencoded) from the input bitstream.
- The last $n - k$ bits are calculated from the first k and are called *check* or *parity* bits.
- At the receiver, the check bits are used to determine whether any bits have been received in error and, if so, to correct them.
- For this reason, channel codes, especially on the BSC, are also known by the name *forward error correction (FEC) codes*.
 - *Forward* because the error correction is done without feedback.

Repetition Codes

If n is odd, a very simple coding scheme is the *repetition code*.

- In *repetition coding*, $k = 1$, and the $n - 1$ check bits are just repetitions of the single message bit.





- At the receiver, the numbers of marks and spaces are counted and, whichever is in the majority, that is the output bit.
- This scheme is able to detect and correct up to $\frac{1}{2}(n - 1)$ errors.
- However, clearly, it's a very slow way to transmit information!

Simple Parity

Another very simple coding scheme is to set $k = n - 1$.

- The single *parity bit* is the summation of the $n - 1$ message bits modulo two.
- At the receiver, the modulo two equation is checked.
- If the received parity bit is not the sum, modulo two, of the received message bits, then an error is declared.
- This scheme detects, but cannot correct, any odd number of bit errors.
- This is still a useful scheme if, *e.g.*, the receiver has a *feedback channel* to request re-transmission.



The Hamming Code

A more sophisticated FEC code was discovered by Richard Hamming in 1948.

- In the simplest version, $n = 7$ and $k = 4$.
- If we label the message bits x_1, x_2, x_3, x_4 then the check bits are computed so that

$$x_5 \equiv x_2 + x_3 + x_4 \pmod{2}$$

$$x_6 \equiv x_1 + x_3 + x_4 \pmod{2}$$

$$x_7 \equiv x_1 + x_2 + x_4 \pmod{2}$$

- At the receive we calculate three bits s_1, s_2, s_3 which are called the *syndrome* so that

$$s_1 \equiv y_2 + y_3 + y_4 + y_5 \pmod{2}$$

$$s_2 \equiv y_1 + y_3 + y_4 + y_6 \pmod{2}$$

$$s_3 \equiv y_1 + y_2 + y_4 + y_7 \pmod{2}$$

where y_1, \dots, y_7 are the received bits.





- The pattern of bits in the syndrome is used to further process the received bits.
- If $s_1s_2s_3 = 000$ then the receiver declares no error and outputs the message bits unchanged.
- Otherwise, the other seven possible patterns in the syndrome determine which of the bits y_1, \dots, y_7 is in error.
- The appropriate bit is inverted and the message bits are then output.
- The Hamming code can detect and correct up to one error in every seven transmitted bits.





Further Developments of Channel Coding

The Hamming code was just the first of a series of discoveries, each increasing the sophistication of practical channel codes.

- In the last decade or so, *low-density parity-check codes* and *turbo codes* have become practical and are near-optimal.
- Shannon also considered sources and channels with ‘continuous alphabets’.
- A well-known result is that the capacity of the *bandlimited, additive white Gaussian noise (AWGN) channel* is

$$C = B \log(1 + \text{SNR})$$

where B is the bandwidth.

- Information theory is applicable well beyond point-to-point communications.
- For instance, results have been obtained for *broadcast* (one to many), *multiple access* (many to one), *MIMO* and *interference* (many to many) channels.

