EE 121: Coding for Digital Communication & Beyond **Fall 2013**

Lecture 11 — October 14

Lecturer: Prof. Anant Sahai *Scribe: Sharmishtha Pal*

11.1 Recap

In previous lectures, we looked at the Binary Symmetric Channel (BSC), where each bit(0 or 1) was flipped with probability p. We did an error analysis of this channel with a random codeword and found the following:

$$\text{If rate } (\frac{k}{n}) < 1 - H(p) \text{ then we could drive } P(\varepsilon) \to 0$$

In this lecture, we'll trace similar steps to analyze the Binary Erasure Channel(BEC).

11.2 Introducing the Erasure Channel

11.2.1 Definition

The BEC is represented by the following figure:

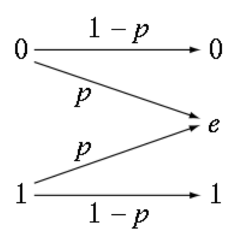

Figure 11.1. Schematic representation of the Binary Erasure Channel

For instance,

$$10110 \longrightarrow \boxed{\text{Binary Erasure Channel}} \longrightarrow 1\phi1\phi0$$

where ϕ represents the erasures.

11.2.2 Why is the Erasure Channel important

In terms of number of parity bits we need to add, it is easier to tolerate erasures than errors. If we cam recognize errors and convert them to erasures, we can make efficient use of the channel. Suppose communicating across space, erased bits can be used as a metric. In orthogonal signaling, if we have two signals crossing the threshold, we can treat them as erasures instead of errors.

11.2.3 Decoding strategy

We cannot use the correlation strategy with codewords as we did in orthogonal signaling. We can instead use a 'matching strategy'. We compare the unerased bits with all the codewords in the codebook. In an extremely lucky situation, we might get only once match, in which case we are done. Else, we need to do some analysis to see how such a 'compatibility test' might help

11.3 Analysis

11.3.1 Analyzing the error probability

Our decoding strategy is to match the non-erased bits of the received codeword with the source codewords in the codebook. Let us make use of a trick to help ourselves with the analysis. We assume that the fraction of the number of erased bits on the number of sent bits has a threshold.

$$\text{Let that threshold be } = p + \epsilon \text{ where } p \text{ is the probability of erasure}$$

When the length of the sent codeword is high enough, the chance of having more than $p + \epsilon$ erasures is very low.

$$\therefore \text{ if length of sent codeword bit string =n}$$

$$\text{no. of erasures will be } \approx np < n(p + \epsilon)$$

Using our trick, we decide to decode only when less than $p + \epsilon$ of the original no. of bits is erased. Therefore, there are $[1 - (p + \epsilon)]n$ non-erased bits when trying to decode.
$\therefore$ Probability that a codeword matches the received string $= (\frac{1}{2})^{(1-(p+\epsilon))n}$

Our codebook here is the same as that for the Binary Symmetric Channel. Therefore, it has 2^k rows of n-bit strings each bit of which is generated using a Bernoulli($\frac{1}{2}$) distribution.

$$\begin{aligned}
\therefore\ P(\varepsilon) &\equiv \text{there exists a false codeword such that it matches the received string} \\
&\leq 2^k (\frac{1}{2})^{(1-(p+\epsilon))n} \\
&= 2^k 2^{(p+\epsilon)n-n} \\
&= 2^{-k(-1-(p+\epsilon)\frac{n}{k}+\frac{n}{k})}
\end{aligned}$$

$$\begin{aligned}
\therefore\ \text{we need } \frac{n}{k} - (p+\epsilon)\frac{n}{k} - 1 &> 0 \\
\therefore\ \frac{n}{k}(1-p) &> 1 \text{ (ignoring } \epsilon) \\
\text{or, } \frac{k}{n} &< 1-p
\end{aligned}$$

Thus, the limitation on the rate($\frac{k}{n}$) is much greater (since rate is bounded by $1-p$) for the erasure channel. Due to this rate efficiency, it is very useful to convert errors to erasures.

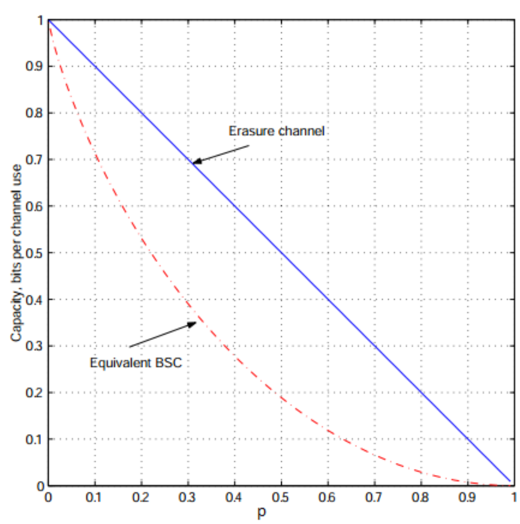

Figure 11.2. Comparison between the erasure channel capacity and an equivalent BSC

11.3.2 Why the trick works!!!

We have assumed that probability of a bit being erased is p. If the bit is not erased, probability that a bit of the received codeword matches with a bit of the sent codeword is 0.5 (since the bits are generated randomly with $\beta(\frac{1}{2})$ distribution).

$$\therefore \text{ Total probability of a bit matching (either erased or non-erased and matches)} = p + (1-p)\frac{1}{2} = \frac{p+1}{2}$$

So if there is no threshold on the number of erasures, then we can potentially have n erasures.
$\therefore$ Probability of a false codeword matching the received string

$$= (\frac{p+1}{2})^n \quad [\because \text{ each bit has probability } \tfrac{p+1}{2} \text{ of matching}]$$

$$\begin{aligned} \therefore\ P(\varepsilon) &\leq 2^k(\frac{p+1}{2})^n \\ &= 2^k 2^{n\log_2 \frac{p+1}{2}} \\ &= 2^{k+n\log_2 \frac{p+1}{2}} \\ &= 2^{-k(-1-\frac{n}{k}\log_2 \frac{p+1}{2})} \end{aligned}$$

Thus in order for the $P(\varepsilon) \to 0$ as $k \to \infty$

$$\begin{aligned} -1 - \frac{n}{k}\log_2\frac{p+1}{2} &> 0 \\ -1 - \frac{n}{k}\log_2(p+1) + \frac{n}{k} &> 0 \\ \therefore\ \frac{n}{k}(1-\log_2(1+p)) &> 1 \\ \therefore\ \frac{k}{n} &< [1 - log_2(1+p)] \end{aligned}$$

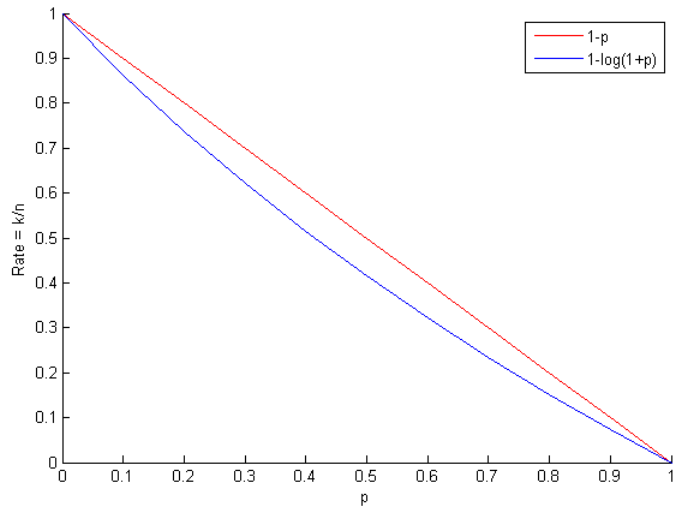

Figure 11.3. Rate efficiency with or without erasure threshold

11.4 Transmission Strategy

If we knew why there are erasures, we could retransmit until there are no erasures. This strategy can be modeled by a geometric random variable with parameter p. Therefore, for a single bit, we need $\frac{1}{1-p}$ transmissions on average. For k bits, we need $\frac{k}{1-p}$ transmissions.

$$\therefore \frac{k}{1-p} = n$$
$$\therefore \frac{k}{n} = (1-p)$$

The rate therefore satisfies the bound calculated previously.
The codewords can be generated using a generator matrix G and an affine code: $G\overline{d} + \overline{b} = \overline{c}$ where $\overline{d}$ is the message string and $\overline{c}$ the generated codeword. We can actually drop the $\overline{b}$ since we are only employing XOR operations at the decoder (we are only matching non-erased bits); so we do not need the affine part. $\therefore\ \overline{c} = G\overline{d}$

11.5 Decoding Strategy

Decoding involves solving a system of linear equations of the form

$$\underbrace{A}_{kXk}\underbrace{\bar{d}}_{kX1} = \underbrace{\bar{b}}_{kX1}$$

This is a $O(k^3)$ operation if solved by Gaussian elimination. To reduce computational inefficiency we can make an observation: we only need to compare with those rows of G (the generator matrix) corresponding to the non-erased bits. Thus if k bits were sent, roughly kp bits are erased in the received codeword. We can therefore compare with the remaining $k - kp$ rows.