Lecture 1

Welcome to EE5160: Error-Control Coding @ IIT Madras!

Instructor: Arvind Rameshwar ; Office: ESB-222

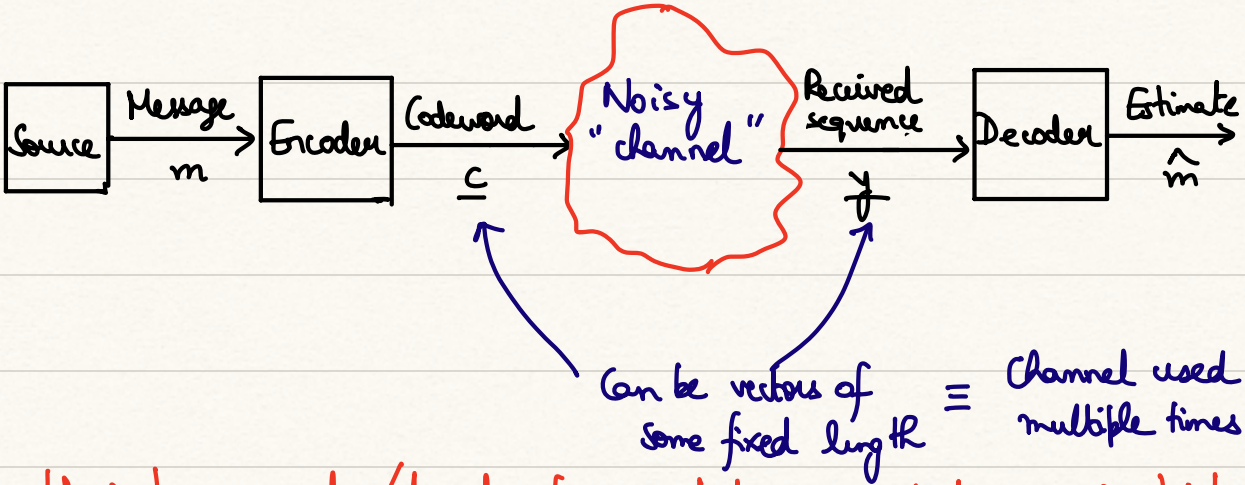
Administrivia, Course Logistics : Discussed in class

Motivation

① Digital communication : Sending information reliably from source to destination.

E.g.: Several layers of TCP/IP stack use error-control coding (ECC), satellite & deep space communication

Oft-used but very succinct picture:



Qn: How to encode/decode for combating noise, but ensuring high communication "rate"?

② Secret sharing : Collection of n high-profile individuals, e.g., board of a company. CEO has a secret (e.g., a will) to be shared with them

such that $\geq t$-sized group can "decode" the secret, but $<(t-1)$-sized group cannot.

Qn: How to share secret so that such a property is obeyed?

③ Group testing: Pandemic infects a neighbourhood; how to test to efficiently identify infected individuals?

Assume: # infected individuals $\ll$ Total # individuals

Qn: How to design "group tests" to isolate infected individuals?

④ Storage in recording media: Data stored on the cloud is booming; how to ensure reliable recovery of stored dog/cat photos when individual nodes in data centres fail at regular intervals?

e.g.: AWS, Azure (+ modern DNA-based data storage)

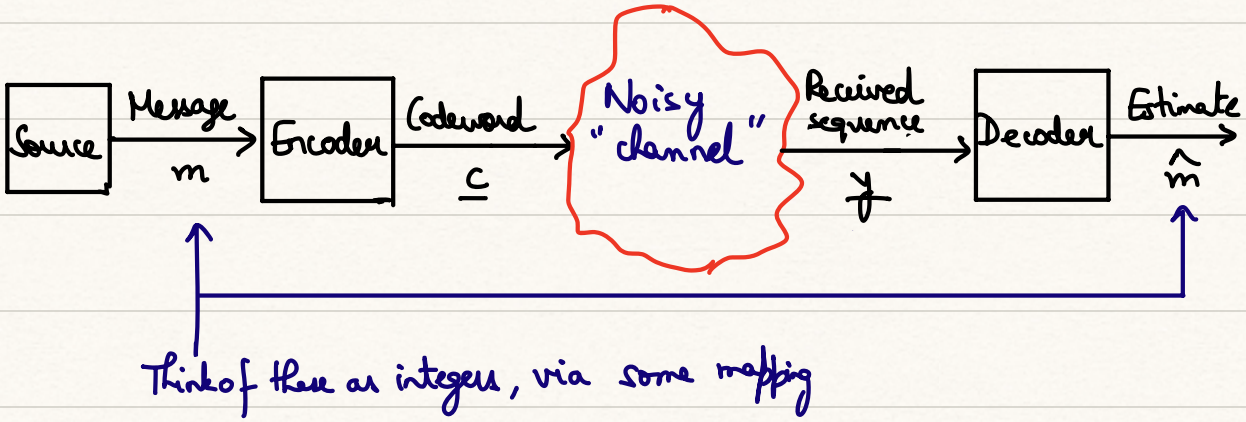
Qn: How to store large quantities of data with reliable recovery?

⑤ Others (too many to name all) applications: Quantum error-correction, correcting errors in streaming data (e.g. Netflix), cryptography, etc.

In this course, we shall mainly work with problem setting ① ≡ ④

[An answer to Problem ② will arise later in the course]

Formalizing Problem ①



① The Encoder

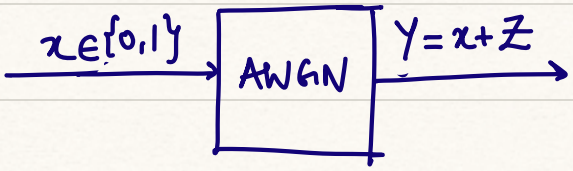
The encoder maps messages $m \in \mathcal{M}$ to a codeword $\underline{c} \in \mathcal{X}^n$, for some finite alphabet $\mathcal{X}$ of size $q := |\mathcal{X}|$. We call this mapping a "code"/"codebook" $\mathcal{C} \subseteq \mathcal{X}^n$.

For simplicity, we often work with $\mathcal{X} = \{0,1\}$ (binary alphabet).

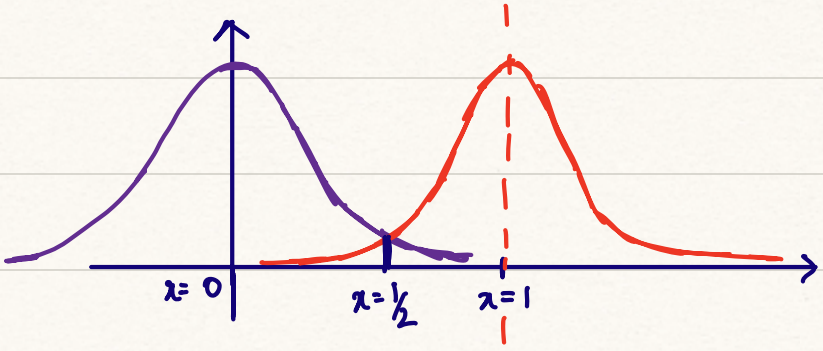
② The channel

A common channel in communications is the binary AWGN channel.

"additive white Gaussian noise"



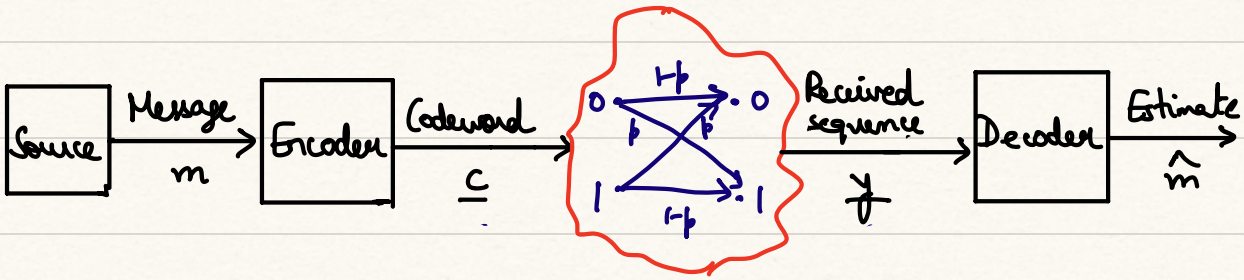
where $Z \sim \mathcal{N}(0, \sigma^2)$.



A "demodulator" declares $\overline{y} = 1$, if $y > 1/2$, and $\overline{y} = 0$, otherwise.

HW ①: Describe the "channel" between X and $\overline{Y}$, i.e., what is $\Pr[\overline{Y} = i \mid x = j]$, for $i, j \in \{0,1\}$?

If you've successfully answered HW ①, you will understand the motivation behind the channel model:



The "binary symmetric channel" (BSC).

$\| 1$

Errors are bit-flips.

Examples of simple codes / decoders: In this course, we shall mainly work in the setting where the # bit-flip errors $\leq t$, for some $1 \leq t \leq n$.

① Repetition code:

$$\underset{\substack{m \\ \| \\ \{0,1\}}}{\text{Message}} \xrightarrow[\text{Encoder}]{\mathcal{E}} \underline{c} = (x, x, x) \xrightarrow{\leq t \text{ bit-flips}} \underline{y}$$

$$\underline{y} \xrightarrow[\text{Decoder}]{\mathcal{D}} \hat{m}$$

Pick $\mathcal{D}$ to be the majority-vote decoder, i.e.,

$$\hat{m} = \begin{cases} 0, & \text{if } \sum_{i=1}^{3} y_i \leq 1, \\ 1, & \text{if } \sum_{i=1}^{3} y_i \geq 2. \end{cases}$$

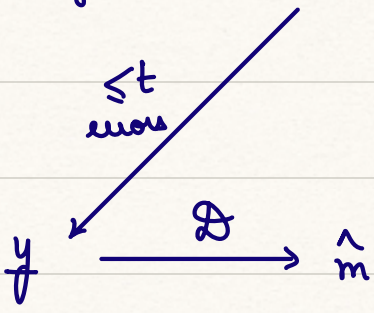
Qn: How many errors (t) can be corrected by this decoder?

Code rate $R := \frac{\log |M|}{n} = \frac{1}{3}$. Error correction capability $\approx \frac{t}{n} = ?$

② Single parity-check code:

Message $m \in \{0,1\}^{n-1} \xrightarrow{\mathcal{C}} \underline{c} = (m, p(m))$, where

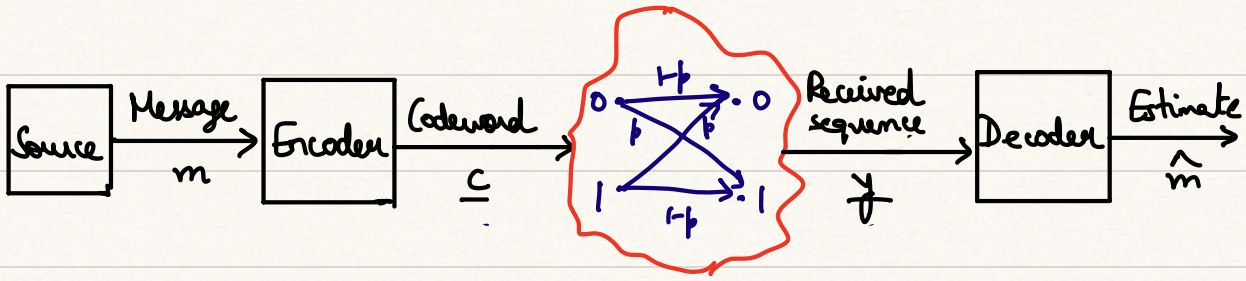
$$p(m) = \text{Parity}(m) = \begin{cases} 1, & \text{if } \sum_{i=1}^{n-1} m_i \text{ is odd}, \\ 0, & \text{o.w.} \end{cases}$$



$$\mathcal{D}(y) = \begin{cases} y, & \text{if } p(y) = 0, \\ \text{Error}, & \text{o.w.} \end{cases}$$

$\mathcal{D}$ can detect odd # errors

Back to



The "binary symmetric channel" (BSC).

Say $m = \underline{m} \in \{0,1\}^K$ and p = "Crossover prob." is given. Suppose $\mathcal{C}$ is a repetition code that repeats each bit m_i, $i \in [K]$, of $\underline{m}$, 3 times

Prob. that a given block is decoded correctly

$$= \sum_{j=0}^{1} \binom{3}{j} \cdot p^j (1-p)^{3-j}.$$

The overall goal of coding theory is to "efficiently" design codes/decoders of "high" rate and "high" error-correction capability.