Lecture 20*

Agenda for the lecture

- Formalization of the channel coding problem
- Coding rates for the binary symmetry channel

20.1 Channels, channel codes, and capacity

Summary of what we saw in Part I of the course. We encountered the channel coding problem in the first part of the course. We saw that the problem of channel coding is mainly that of designing a decoder which can distinguish between exponentially many (in $n$) input sequences for repeated independent uses of the channel. This is a twist on the pre-Shannon digital communication view, where the an absolute error bottleneck is usually associated with a fixed modulation scheme. Therefore, if one uses a fixed modulation scheme repeatedly, the combined probability of error approaches one. Shannon suggested an alternative view where a trade-off between error and the “rate” of transmission is established. One of the key insights is that longer codes must be designed separately, and not by repeating shorter codes. In this part of the course, we shall formalize this problem and provide formal proofs of some of the heuristics mentioned in the first part. In particular, we shall establish Shannon’s Channel Coding Theorem.

*©Himanshu Tyagi. Feel free to use with acknowledgement.
20.1.1 Channels

A channel consists of a tuple \((\mathcal{X}, W, \mathcal{Y})\) where \(\mathcal{X}\) is the input alphabet, namely the set of possible “signals” you can transmit, \(\mathcal{Y}\) is the output alphabet, the set of possible received signals at the channel output, and a channel transformation \(W : \mathcal{X} \to \mathcal{Y}\) which is a family of probability distributions \(\{W_x : x \in \mathcal{X}\}\) on \(\mathcal{Y}\), indexed by the elements of input alphabet. Specifically, \(W_x \equiv W(\cdot | x)\) denotes the probability distribution of the random output of the channel when the input to the channel is \(x\). When both \(\mathcal{X}\) and \(\mathcal{Y}\) are discrete, the channel can be described by an \(|\mathcal{X}| \times |\mathcal{Y}|\) matrix, termed the channel matrix, with the \(x\)th row denoting the conditional pmf of \(Y\) given \(X = x\); such matrices are called stochastic matrices.

Our goal in channel coding is to use this channel repeatedly, \(n\) times, to enable reliable transmission over this channel. We assume that each use of channel produces independent output, namely the noise of the channel is independent in each use. Such channels are called memoryless channels. For the most part, we shall focus on Discrete Memoryless Channels (DMC). Alternatively, a DMC can be viewed as a discrete channel \(W^n : \mathcal{X}^n \to \mathcal{Y}^n\) with the channel matrix given by

\[
W^n(y|x) = \prod_{i=1}^{n} W(y_i|x_i), \quad \forall x \in \mathcal{X}^n, y \in \mathcal{Y}^n.
\]

20.1.2 Channel codes

Although at each instance we shall transmit only one message over the channel, we need to formulate the problem for transmitting a set of messages, since otherwise there is no uncertainty about the message at the decoder. Specifically, we consider the problem of sending a message out of the set \(\mathcal{M} = \{1, ..., M\}\). Each message is mapped using an encoder to a sequence of inputs \(x = (x_1, ..., x_n)\) for the channel. This sequence is transmitted by using the channel \(n\) times. The receiver observes the corresponding sequence \(Y_1, ..., Y_n\) at the output, which is now random due to noise introduced by the channel. Finally, the
receiver forms an estimate $\hat{m}$ of the transmitted message. Formally, we have the following definition.

**Definition 20.1.** An $(n,M)$ code for the channel $(\mathcal{X},W,\mathcal{Y})$ consists of an encoder $e : \mathcal{M} \to \mathcal{X}^n$ which maps a message $m$ to an $n$-length input for the channel; and a decoder $d : \mathcal{Y}^n \to \mathcal{M}$ which observes an $n$-length output sequence and makes an estimate $\hat{m}$ of the message that was sent. The code set (or simply the code) is given by the set of codewords $\{e(m), m \in \mathcal{M}\}$. The rate of this code is given by $R = \frac{1}{n} \log M$.

There are two possible measures for the error of a code. For simplicity, we restrict to DMCs.

1. **Maximum probability of error.** The max. prob. of error $\epsilon_{\text{max}}(e,d)$ is given by

   \[ \epsilon_{\text{max}}(e,d) = \max_{m \in \mathcal{M}} \mathbb{P}(d(Y^n) \neq m | e(m) \text{ was sent} ) \]

   \[ = \max_{m \in \mathcal{M}} W^n(\{y : d(y) \neq m\}|e(m)) \]

   where $W^n$ denotes the channel corresponding to $n$ uses of the DMC $W$.

2. **Average probability of error.** The avg. prob. of error $\epsilon_{\text{avg}}(e,d)$ is given by

   \[ \epsilon_{\text{avg}}(e,d) = \frac{1}{M} \sum_{m \in \mathcal{M}} W^n(\{y : d(y) \neq m\}|e(m)) . \]

Note that the error of the code when a message $m$ is sent is given by

\[ W^n(\{y : d(y) \neq m\}|e(m)) ; \]

the max. and the avg. prob. of error, respectively, are the max. and the avg. of these quantities. The important object here is the set

\[ D_m = \{y \in \mathcal{Y}^n : d(y) = m\} , \]
the set of output sequences decoded to a message \( m \). The sets \( D_m, m \in \mathcal{M} \), form a partition of the space of output sequences \( Y^n \). The problem of code design with max. probability of error less than \( \epsilon \) is to identify such a partition with the additional property that there exist \( x_1, \ldots, x_M \) in \( X^n \) such that

\[
W^n(D_m|x_m) \geq 1 - \epsilon, \quad m \in \mathcal{M}.
\]

(1)

Note that there is no additional constraint on \( x \)'s, they can all be the same in principle. The crux of the issue is designing the partition \( \{D_m, m \in \mathcal{M}\} \). Indeed, there is no limit to what you can transmit over the channel, the limit is on what you can receive over the channel!

### 20.1.3 Achievable rates and channel capacity

**Definition 20.2.** Given a memoryless channel \((X, W, Y)\), a rate \( R > 0 \) is an \( \epsilon \)-achievable rate if for all \( n \) sufficiently large there exists an \((n, M_n)\) code of rate \( 1/n \log M > R \) with maximum probability of error less than \( \epsilon \). The supremum of all \( \epsilon \)-achievable rates \( R \) is called the \( \epsilon \)-channel capacity, denoted \( C_\epsilon(W) \). The channel capacity \( C(W) \) is defined as the limit

\[
C(W) = \lim_{\epsilon \to 0} C_\epsilon(W).
\]

Some remarks are in order. First, note that several other variants of the definition of capacity above can be considered. For instance, instead of requiring codes of rate \( R \) “for all \( n \) sufficiently large” we might require that for all \( n \) there exists a code of rate greater than \( R \) for an \( n' > n \). Or, instead of fixing an \( \epsilon \), taking limits in \( n \), and then taking limit in \( \epsilon \), we can simply require that there exists a sequence \( \epsilon_n \) converging to 0 and the probability of error of the required codes is less than \( \epsilon_n \). For a DMC, all these definitions lead to the same answer. This equivalence, however, does not hold in general.

Also, the definition above can also be expressed in a lim sup form. Denote by \( M(n, \epsilon) \)
the least \( M \) such that an \((n, M)\) code with maximum probability of error less than \( \epsilon \) exists. The \( \epsilon \)-capacity \( C_\epsilon(W) \) is defined as

\[
C_\epsilon(W) = \limsup_n \frac{1}{n} M(n, \epsilon).
\]

Finally, we have used max. probability of error in our definitions. It will be seen that \( C(W) \) does not change if we replace max. prob. of error with avg. prob. of error.

## 20.2 Coding rates for binary symmetric channel

### 20.2.1 Converse: The sphere packing upper bound

An important channel model is the so-called binary symmetric channel (BSC), which is a channel with \( \mathcal{X} = \mathcal{Y} = \{0, 1\} \). The channel matrix \( W \) is given by \( W(1|0) = W(0|1) = \delta \); we say that \( W \) is a BSC with crossover probability \( \delta \). For this channel,

\[
W^n(y|x) = \prod_{i=1}^{n} W(y_i|x_i) = \prod_{i=1}^{n} \left[ \delta 1(y_i \neq x_i) \cdot (1 - \delta) 1(y_i = x_i) \right] = \prod_{i=1}^{n} P_{Z}(x_i \oplus y_i),
\]

where \( Z \sim \text{Bernoulli}(\delta) \) and \( \oplus \) denotes the XOR function. Therefore, denoting \( z_i = x_i \oplus y_i \in \{0, 1\} \),

\[
W^n(y|x) = P_{Z^n}(z) = P_{Z^n}(x \oplus y).
\]

Thus, for an \((n, M)\) code with probability of error less than \( \epsilon \), the set \( D_m \) of sequences decoded to a message \( m \) satisfies

\[
P_{Z^n}(D_m \oplus e(m)) \geq 1 - \epsilon.
\]
Recall that the Little-Big Lemma applied to $Z^n$ together with Chebyshev inequality shows that the cardinality of any set $B$ with $P_{Z^n}(B) \geq 1 - \epsilon$ satisfies
\[
\log |B| \geq nh(\delta) - \sqrt{\frac{n\text{Var}(h(Z))}{\eta}} + \log(1 - \epsilon - \eta),
\]
for every $\eta < 1 - \epsilon$. Hence, $|D_m \oplus e(m)| = |D_m|$
\[
\log |D_m| \geq nh(\delta) - \sqrt{\frac{n\text{Var}(h(Z))}{\eta}} + \log(1 - \epsilon - \eta).
\]
Since $\{D_m, m \in \mathcal{M}\}$ constitutes a partition of $2^n$ sequences,
\[
|\mathcal{M}| \leq \frac{2^n}{2^{nh(\delta)} - \sqrt{\frac{n\text{Var}(h(Z))}{\eta}} + \log(1 - \epsilon - \eta)} = 2^{n \left(1 - h(\delta) + \sqrt{\frac{\text{Var}(h(Z))}{n\eta}} + \frac{1}{n} \log \frac{1}{1 - \epsilon - \eta}\right)}.
\]
Choosing $\eta = (1 - \epsilon)/2$, we get the following converse bound (known as the sphere packing bound) for the $\epsilon$-capacity of $W$.

**Theorem 20.3** (Sphere Packing Bound). For a BSC with crossover probability $\delta$ and every $0 < \epsilon < 1$,
\[
\frac{1}{n} \log M(n, \epsilon) \leq 1 - h(\delta) + \sqrt{\frac{2\text{Var}(Z)}{n(1 - \epsilon)}} + \frac{1}{n} \log \frac{2}{1 - \epsilon}.
\]
In particular,
\[
C_\epsilon(W) \leq 1 - h(\delta).
\]

20.2.2 Achievability: The Gilbert-Varshamov bound

For achievability, we first relate sets of large conditional probability under $W_x^n$ to balls centered at $x$ w.r.t. an appropriately defined distance metric. Specifically, let
\[
d(x, y) = \sum_{i=1}^{n} \mathbb{1}(x_i \neq y_i).
\]
This distance is called the Hamming distance and denotes the number of places at which two binary sequences differ. Denote by $B_{\rho}(x)$ the Hamming ball of radius $\rho$ centered at $x$, i.e., the set of sequences $y \in \{0,1\}^n$ with $d(x,y) \leq \rho$. The next lemma gives us a handle over the typical Hamming distance between the inputs and the outputs of a BSC. Specifically, it shows that with probability greater than $1 - \epsilon$, the output lies within a Hamming distance $n\delta + o(\sqrt{n})$ of the input sequence.

**Lemma 20.4.** Given a BSC $W$ with crossover probability $\delta$, for every sequence $x$ and $
abla = n\delta + \sqrt{n\delta(1-\delta)}$, we have

$$W^n(B_{\rho_n}(x)|x) \geq 1 - \epsilon.$$

**Proof.** The proof is simply by Chebyshev inequality since, as we noted earlier, $W^n(y|x) = P_{Z^n}(z)$ where $z_i = 1(x_i \oplus y_i)$. Thus, the probabilities are computed by a Binomial($n, \delta$) distribution, with 1 corresponding to an index where $x_i$ and $y_i$ differ and 0 to an index where they are the same. (Note that the mean and the variance of this experiment are given by $n\delta$ and $n\delta(1 - \delta)$ respectively.)

Thus, the picture of BSC is very simple. Every input $x$ sent over this channel is replaced at the output by a Hamming ball of radius no more than $n(\delta + \eta_n)$ with center at $x$, where $\eta_n = \sqrt{\delta(1-\delta)/n\epsilon}$. In fact, the lemma above directly reveals a good choice for $D_m$ — it is simply a Hamming ball of radius $\rho_n(\epsilon) = n(\delta + \eta_n)$. Thus, the problem of code design is reduced to that of finding the largest collection of disjoint Hamming balls of radius $\rho_n(\epsilon)$ in $\{0,1\}^n$; such a collection is called a “packing of Hamming balls.”

Denote by $B_{\rho}(x)$ the Hamming ball of radius $\rho$ with center at $x$. Note that $|B_{\rho}(x)| = |B_{\rho}(0)| =: C_{\rho}$. Furthermore, $C_{\rho}$ equals the number of binary sequences with less than $\rho$ 1s. Thus,

$$C_{\rho} = \sum_{k=1}^{\rho} \binom{n}{k}.$$
Using \( \binom{n}{k} \geq 2^{n \log \left( \frac{k}{n} \right)} \) (we proved this bound using method of types), we get

\[ C_\rho \geq 2^{n \log(\rho/n) - \log(n+1)}. \]

Since the total number of binary sequences is \( 2^n \), the maximum size of a packing of Hamming balls of radius \( \rho \), denoted \( N(n, \rho) \), is bounded as

\[ N(n, \rho) \leq \frac{2^n}{C_\rho} \leq \frac{(n + 1)2^n}{2^{n \log(n+1)}} = 2^n \left( 1 - h(\rho/n) + \frac{1}{n} \log(n+1) \right). \]

Next, we construct packing of Hamming balls. Once we choose a point \( x \) as center of a ball in the packing, we can’t select any other point within a radius \( 2\rho \) of \( x \) as the center of another ball in the packing. We construct a packing by selecting the centers of the balls in a greedy fashion. The process is very simple: Once a point is selected, we remove the Hamming ball of radius \( 2\rho \) around it. The next point is selected arbitrarily from the remaining points. The process stops when you can find no more points. We make two observations about this construction:

**Observation 1.** Let \( x_1, \ldots, x_M \) be the sequences selected as centers in the process above. Then, for every \( x_i, x_j \),

\[ B_\rho(x_i) \cap B_\rho(x_j) = \emptyset. \]

*Proof.\footnote{\textcopyright 2023, MIT}{Proof.} Suppose not. Then, there exists a \( y \) such that \( d(x_i, y) \leq \rho \) and \( d(x_j, y) \leq \rho \). But then \( d(x_i, x_j) \leq 2\rho \), which is a contradiction since once a point is selected no point within \( 2\rho \) of it is selected in the future. \qed

Note that while \( B_\rho(x_i) \)s are disjoint, \( B_{2\rho}(x_i) \)s in the construction above may not be disjoint.

**Observation 2.** The number of balls selected in the process above is greater than \( 2^n / C_{2\rho} \).
Proof. Note that after selecting $L$ balls, we can select one more point if

$$|\bigcup_{i=1}^{L} B_{2\rho}(x_i)| < 2^n.$$ 

Thus, when the process stops,

$$|\bigcup_{i=1}^{M} B_{2\rho}(x_i)| \geq 2^n,$$

which implies

$$\sum_{i=1}^{M} |B_{2\rho}(x_i)| \geq 2^n.$$ 

The claim follows since $B_{2\rho}(x_i) = C_{2\rho}$. \(\square\)

Note that for $\rho \leq n/2$,

$$\binom{n}{k} \leq 2^{nh(k/n)} \leq 2^{nh(\rho/n)}, \quad \forall k \leq \rho.$$ 

Thus, for $\rho \leq n/4$

$$C_{2\rho} = \sum_{k=1}^{2\rho} \binom{n}{k} \leq n2^{nh(2\rho/n)}.$$ 

Thus, the packing formed above have at least $2^n(1-h(2\rho/n)-\frac{1}{n} \log n)$ balls.

In summary, for $\rho \leq n/4$ we have established

$$1 - h(2\rho/n) - \frac{1}{n} \log n \leq \frac{1}{n} \log N(n, \rho) \leq 1 - h(\rho/n) + \frac{1}{n} \log(n + 1).$$

The lower bound for $N(n, \rho)$, together with Lemma 20.4, immediately leads the following bound for $M(n, \epsilon)$.

Lemma 20.5 (Gilbert Varshamov Bound). Given a BSC $W$ with crossover probability $\delta < 1/4$, for every $0 < \epsilon < 1$, it holds that

$$\frac{1}{n} \log M(n, \epsilon) \geq 1 - h \left( 2\delta + 2\sqrt{\frac{\delta(1-\delta)}{n\epsilon}} \right) - \frac{1}{n} \log n.$$
In particular,
\[ C_e(W) \geq 1 - h(2\delta). \]

Note that the Hamming balls based construction of code above describes both encoder and the decoder. The encoder simply maps each message to the center of the balls. The decoder simply returns the codeword closest to the received binary vector in Hamming distance. This decoder is termed the \textit{minimum distance decoder}. Another decoder which serves the same purpose with this construction is the \textit{distance threshold decoder} which returns the codeword within a Hamming distance \( \rho_n \) of the received vector.

Unfortunately, the achievability bound above does not match our sphere packing converse. Indeed, we used a greedy approach and were hoping to be lucky. In the next lecture, we will present two different approaches which yield achievability bounds matching our converse bound: First, Feinstein’s \textit{maximal code construction} and the second, Shannon’s \textit{random code construction}. 