IITM-CS6845: Theory Toolkit Jan 6 & 11, 2012

Lecture B01-B02 : Introduction to Coding Theory

Lecturer: Jayalal Sarma *Scribe: No Scribe*

Coding theory had its inception in the late 1940's with the theory of reliable communication over a channel in the presence of noise - an area that started with the pioneering work of Claude Shannon and Richard Hamming. The former addressed answered the fundamental questions about the possibility of the use of codes for reliable communication and the later develped some basic combinatorial constructions of error correcting codes that laid the foundations for the work later.

Theoretical computer scientists have a major role to play in the algorithmic aspects of coding theory research, and coding theory has proved to be instrumental in several interesting results in theoretical computer science as well. This quarter of the course will aim to discuss some of the important aspects of the theory as well as some of the applications to research in theoretical computer science. However, we do not intend to be exhaustive.

This lecture will address some of the classical ideas in the theory. A typical situation that we are interested in is as follows: Alice has some information to be sent out to Bob. The data is represented as a string over some alphabet. She sends it through a channel which could introduce errors in the transmission. This has two incarnations one can immediately see.

Communication over time : Alice stores some information on a magnetic disk and tries to retrieve it at a future point in time. In this case Alice could be thought of as communicating to herself through the disk and the parties are at two points in time.
Communication over space: Alice and Bob are physically apart, at two points in space, and uses the physical channel for communication. The model and the theory we will be describing applies to both of them.

For our purpose we will be interested in understanding the give and take that this independently developed branch of study has with theoretical computer science. There has been several surprising applications of the theory and the associated mathematical objects, in areas like algorithms, complexity theory and cryptography. Before getting into the applications side we will set the stage in which these objects are being looked at in a fundamental way.

The area was initiated by two path-breaking papers in the late 40s. One by Shannon which gave a mathematical treatise of the model of communication channel and the parameter values that can be achieved and the second by Hamming which gave the theory of error correcting codes which he developed with a clear engineering motivation too (and hence

is more constructive) : storing information on a magnetic disk and then retrieving it later even when there has been a few errors.

1 Modeling the Channel

The object of study is the channel. Messages are passed through the channel. Some messages may get lost, some may get corrupted.

But simple questions first. Suppose the channel is harmlesss. That is, whatever bit is sent by Alice is received by Bob precisely without any error. How do we encode the message in such a way that the minimal information is what we need to send. This task is familar to us by the name *compression* and *decompression*. But, how much can we compress?

Example : We want to send a message whose contents is what is written in a piece of paper. But in advance we know that the paper is almost empty (say only 1 out of the 100 characters(or bits) is non-trivial(1) everything else is blank(0)). How do we describe this knowledge? One way is to look at it as a probability distribution. That is, we say, choose a character randomly from this paper, it is blank with probability 0.99.

One trivial strategy is to send 100 characters. But this is far from efficient because there is only 1 blank symbol. Here is a better scheme. Let us send the information in the paper as blocks of length 10 characters each. For each block of size 10, if there is a non-blank symbol, we send the block as it, adding a 1 on the left (as a delimiter). Otherwise, we send it across as a single 0.

Viewing the paper to be sent, as a random source of characters, the expected size of the block can be calculated as follows. $1.Pr[$ Block is all-zero $] + 11.Pr[$ Block is no all-zero$] = 11 - (10 * (0.99)^{10}) = 2$ bits !

Further, if we us fix the size of the paper to be 100 characters. Since only one of the blocks will be sent as it is, and all the others will be just one bit. So the total length of the message that will be sent is really $9 + 11 = 20 bits$ (assuming each character is encoded using only one bit).

One intuition that the above example is giving us that we want to send the high probability event with a lower length sequence to reach optimality. But what is really the optimal one? Shannon asked and answered this question in the Noisless coding theorem which we will only state. To answer the question, he associated a non-negative real number which captures the structural information about the object in a more precise way. This became backbone of the theory he developed. This was the notion of *entropy* of a source which measures the amount of randomness in the object to be sent across.

A simple example first, suppose that the number of 0s and 1s (blanks and non-blanks) in

the page was exactly 50 each. In this case out strategy of encoding the higher probability event spending lower number of bits does not imply an improvement. Shannon formalised this intuition using the notion of entropy.

Let X be a 0-1 random variable that takes value 1 with a probability p and 0 with probability $1-p$. The entry number given in this case is : $H(p) = p \log \frac{1}{p} + (1-p)\frac{1}{p}$. One can verify that this function (plot it !) achieves the maximum when $p = \frac{1}{2}$; and matches the above intuition.

Let $D : U \to \{0,1\}$ be a probability distribution on the domain U. Let X be a random variable with distribution D. The entropy of the distribution is:

$$H(D) = \sum_{x \in U} D(x) \log \left(\frac{1}{D(x)} \right)$$

.

Now we are ready to state the noiseless coding theorem due to Shannon which states the existence of the best "compression" that we can achieve.

Theorem 1 (Noiseless Coding Theorem)**.** *For a finite set U, for every distribution $D : U \to [0,1]$, there exist functions $Enc : U \to \{0,1\}^*$ and $Enc : \{0,1\}^* \to U$ such that : $\forall\ x : Dec(Enc(x)) = x$ and,*

$$H(D) \leq E_{x \in_D U} |Enc(x)| \leq H(D+1)$$

Moreover, no other pair of functions that achieves the first condition can achieve the second condition.

We will not be doing the proof here since it is beyond the scope of this lecture and in general the aim of the course. The upper bound follows roughly the intuition that we stated while describing the example of the sparse message on paper. The main idea is to use lower number of bits to encode symbols with high probability (that are too frequent). In essence, we will round down each probability value to the 2^{-i} values and then use i bits to encode them. Just to do a sanity check here; why can we even hope to assign unique strings to each rounded message (remember, this is a requirement if we need Dec to be a function). There cannot be more than 2^i message symbols which gets rounded to 2^{-i} since the total probability will add up to more than 1.

The argument then estimates precisely he non-optimality that we have incurred by the rounding process and proves that expected length of the encoding is at most $H(D) + 1$. The lower bound is more involved and we will skip completely. The reference for reading it up is the text-book by Cover and Thomas.

2 Noisy Channel

The channel is not harmless in the real world. It introduces errors in the transmission. Depending on the application the error may be in the physical storage media (communication over time) or in the physical channel (communication over space). Some of the 0s gets flipped to 1s and vice versa, and some bits may get dropped too. For the purposes of this course we will study only model (Shannon studied several interesting variants), namely what are called Binary Symmetric Channels. In this model, each bit gets flipped with a probability p. That is, a 1 gets flipped to a 0 with probability and 0 gets flipped to 1 with probability p.

What is the natural strategy to cope up with errors in transmission? Create redundancy. For example, if Alice wants to send a bit 0 to Bob, she will do it five times, and send 11111 and ask Bob to take the majority of the bits as the bit that was sent. In this simple looking example we have all the essence. The string that was sent will be called the *codeword* and the original bit to be sent is called the *message*. There are only two codewords 00000 and 11111 in the above example. If we define the notion of distance as the hamming distance, then the majority decoding mechanism described above can also be seen as choosing the codeword that is closest to the received word. This natural strategy of decoding is called *nearest neighbor decoding* or *maximum likelyhood decoding*.

Now let us observe facts about guarantees. Clearly if the channel is such that it will not corrupt more than 2 bits in a sequence of 5 bits, then Bob will be able to decode the message bit correctly. But the channel may actually flip more number of bits but with relatively lower probability. Thus if we increase the number of copies we make of the original message, with high probability (over the errors) introduced by the channel we are going to be able to decode the bit correctly.

To fix some notations, we denote $E : \{0,1\}^k \to \{0,1\}^n$ as the encoding function where k is the message length (in general) and n is the length of the codeword (which we will call the *block length*. Let $m \in \{0,1\}^k$ be a message, and $E(m) \in \{0,1\}^n$ is the transmitted word. The channel corrupts the message and let $y \in \{0,1\}^n$ is the received word. The error introduced by the channel could also be thought of as a string $\eta \in \{0,1\}^n$ where the η_i determines whether $y_i = (E(m))_i$ or not.

We want the following guarantee for any $m \in \{0,1\}^k$ as translating the above intuition:

$$Pr_\eta(D(E(m) + \eta) = m) \geq 1 - o(1)$$

where the $o(1)$ term is exponentially small depending on n and hence on k (since c is a constant). To relax even further we want the guarantee,

$$Pr_{m \in \{0,1\}^n} Pr_\eta(D(E(m) + \eta) = m) \geq 1 - o(1)$$

Shannons theorem essentially states that there is a pair of encoding-decoding functions that can achieve this high confidence decoding of the original message. Now we are ready to state the theorem formally.

Theorem 2 (Noisy Coding Theorem)**.** *For every $0 \leq p < \frac{1}{2}$, and $c > \frac{1}{1-H(p)}$, there exists $\delta > 0$ such that for large enough n, there exists an encoding function $E : \{0,1\}^k \to \{0,1\}^n$ and a decoding function $D : \{0,1\}^n \to \{0,1\}^k$, for $n = ck$, such that for every $m \in \{0,1\}^k$,*

$$Pr_\eta \left(D(E(m) + \eta) = m\right) \geq 1 - \frac{1}{2^{\delta n}} \tag{1}$$

where η is the error bit-vector introduced by the channel. Moreover, if $c \leq \frac{1}{1-H(p)+\epsilon}$, then the decoding error probability will be close to 1 for large n and k.

Proof. We need some notations first. For $x, y \in \Sigma^*$, $\Delta(x, y)$ is the set of indices in which the symbol differs in the two strings x and y. We also need the notion of a Hamming Ball in the space of $\{0,1\}^n$ around a string $y \in \{0,1\}^n$ of radius $r \in \mathbb{N}$.

For the first part, we prove a relaxed version. That is we show equation 1 holds over a random choice of $m \in \{0,1\}^n$ with high probability. We will then outline how to prove equation 1 for all m.

We need to show the existence of an encoding and decoding algorithm which achieves the error bound. Notice that we are not worried about computation of these functions, but only existence In fact we prove a stronger theorem. We show that there exists an encoding function even for a fixed decoding function that we choose. . The decoding function we fix is the nearest neighbor decoding described with the above example. To formally state this, the function $D : \{0,1\}^n \to \{0,1\}^k$ works as follows: given a string $y \in \{0,1\}^n$, the decoded string $D(y)$ is $m \in \{0,1\}^k$ such that $\Delta(y, E(m))$ is minimised.

With this D, now we will argue that over a randomly chosen encoding function, we can satisfy the equation 1, with probability greater than 0. This will then show that there exists an encoding function such that equation 1 is satisfied.

To execute the plan, we first describe the random process of choosing E. For every $m \in \{0,1\}^k$, choose a string $E(m)$ uniformly at random from the set $\{0,1\}^n$. (Notice that this may not give us a injective map sometimes, but that will get estimated in the decoding error probability finally.)

Let us analyse the event $D(E(m) + \eta) \neq m$. We also keep in mind the property that the number of bits that gets flipped is pn on the average. By applying Chernoff Bound, we see that it is between $pn + \epsilon$ and $pn - \epsilon$ with probability less than $e^{-\frac{\epsilon^2 n}{2}}$. In particular,

$$Pr_\eta(\Delta(E(m), y) \geq pn + \epsilon) \leq e^{-\frac{\epsilon^2 n}{2}}$$

$$
\begin{aligned}
Pr_\eta(D(E(m)+\eta) \neq m) &= \sum_{y \in \{0,1\}^n \,:\, D(y) \neq m} Pr_\eta(E(m)+\eta = y) \\
= \sum_{\left\{y:\ \substack{\Delta(E(m),\,y) \leq np+\epsilon \\ D(y) \neq m}\right\}} Pr_\eta(E(m)+\eta = y) &+ \sum_{\left\{y:\ \substack{\Delta(E(m),\,y) > np+\epsilon \\ D(y) \neq m}\right\}} Pr_\eta(E(m)+\eta = y) \\
\leq \sum_{\left\{y:\ \substack{\Delta(E(m),\,y) \leq np+\epsilon \\ D(y) \neq m}\right\}} Pr_\eta(E(m)+\eta = y) &+ \sum_{\{y:\ \Delta(E(m),\,y) > np+\epsilon\ \}} Pr_\eta(E(m)+\eta = y) \\
\leq \sum_{\left\{y:\ \substack{\Delta(E(m),\,y) \leq np+\epsilon \\ D(y) \neq m}\right\}} Pr_\eta(E(m)+\eta = y) &+ e^{-\frac{\epsilon^2 n}{2}}
\end{aligned}
$$

The first term is summing over those strings y such that the decoded message is not m. By our decoding algorithm this can happen only if another message m' has its encoding $E(m')$ closer to y than $E(m)$. But for this, $E(m')$ has to be within a ball of radius $np+\epsilon$ with y as the center. Counting the number of strings in the ball gives us a handle on the probability.

To do this let us define volume first. For $y \in \{0,1\}^n$, $r \in n$, the volume, $\mathsf{Vol}(y,r)$ as the volume of the ball centred at y of radius r, that is the number of strings within a distance of r from$y$. This is precisely, $\sum_{i=0}^{i=r} \binom{n}{i}$. But due to the symmetry in$\{0,1\}^n$ this number is the same for any y. Hence we will denote it as $\mathsf{Vol}(n,r) = \sum_{i=0}^{i=r} \binom{n}{i}$.

We will also estimate the volume in our case by the following lemma.

Lemma 3. *For $p < \frac{1}{2}$, $\mathsf{Vol}(p,n) \leq 2^{n.H(p)}$*

Proof. $H(p) = p\log\frac{1}{p} + (1-p)\log\frac{1}{1-p}$. Hence we have, $2^{-H(p).n} = p^{np}(1-p)^{np}$.

$$\begin{aligned}
1 &= (p+(1-p))^n \\
&\geq \sum_{i=0}^{i=pn} \binom{n}{i} p^i (1-p)^{n-i} \\
&= \sum_{i=0}^{i=pn} \binom{n}{i} \left(\frac{p}{1-p}\right)^i (1-p)^n \\
&\geq \sum_{i=0}^{i=pn} \binom{n}{i} \left(\frac{p}{1-p}\right)^{pn} (1-p)^n \\
&= \sum_{i=0}^{i=pn} \binom{n}{i} p^{pn}(1-p)^{(1-p)n} \\
&= \sum_{i=0}^{i=pn} \binom{n}{i} 2^{-n.H(p)}
\end{aligned}$$

□

Now we use this bound to complete our estimation of decoding error probability.

$$\begin{aligned}
\sum_{y:\Delta(E(m),y)\leq np+\epsilon, D(y)\neq m} Pr_\eta(E(m)+\eta = y) &\leq Pr_\eta(\exists m' : \Delta(E(m'),y) \leq np+\epsilon) && (2)\\
&= (2^k-1)\frac{\mathsf{Vol}(y, np+\epsilon)}{2^n} && (3)\\
&\leq 2^k \frac{2^{-n.H(p)}}{2^n} && (4)\\
&\leq 2^{k-(1-H(p))n} && (5)
\end{aligned}$$

We want to choose a c such that this error probability is less than 1, and hence our randomly chosen E will work correctly with non-zero probability. A choice of n such that $k > (1-H(p))n$ will work here. That is, $\frac{k}{n} = \frac{1}{c} < 1 - H(p)$.

The above analysis was for a fixed m and $E(m)$. The analysis works for a random m too. Hence, there is a δ such that $2^{-\delta n} = 2^{-\frac{\epsilon^2 n}{2}} + 2^{k+(1-H(p))n}$ such that,

$$E_{\eta,m,E}(D(E(m)+\eta \neq m) \leq 2^{-\delta n}$$

Hence there exists E such that,

$$Pr_{\eta,m}(D(E(m)+\eta \neq m) \leq 2^{-\delta n} \quad (6)$$

That completes the proof of the relaxed case.

Remark 4. We remark about the strengthening of the argument to the case of all m in the domain. We want equation 6 to hold for every m. Taking union bound will introduce a factor of 2^k in our error proability and makes our c larger. But here is a better way: we know that there is an E that works well on the average. We will tinker with E to get an E' that works well for all m. Sort the messages in the increasing order of their decoding error probabilities. The message a which appears in the middle ($(2^{k-1})^{th}$ message has decoding error probability at most $2^{-\delta n+1}$ (otherwise it contradicts equation 6). Since the list is sorted *all* messages below a have at most $2^{-\delta n+1}$ decoding error probability. We just discard the top 2^{k-1} messages (these had higher decoding error probability) from the domain and work with the smaller domain (call it R_k). We have $|R_k| = 2^{k-1}$[1] How do we describe the function E' in terms of E? We know that $|R_k| = 2^{k-1}$. Let f_k be a bijection from $\{0,1\}^{k-1} \to R_k$. Now $E'(m) = E(f_{k+1}(m))$. Similarly the decoding function is $D'(y) = f_{k+1}^{-1}(D(y))$. Note that the bijection does not introduce error and is a part of the encoding and the decoding functions. Hence there exists encoding and decoding functions such that for all messages the decoding error probability is exponentially small.

Capacity of the channel: Given that we wanted to optimize the redundancy that we create in our example, which translates to optimizing the value of c, a natural question that arises out of the above discussion is if c that we chose was optimal. The fact that $\frac{1}{c} > 1 - H(p)$ arose out of the analysis that we had for the probabilities. For sending information across how much is the capacity that we have. Is this really just that this c works? or is there a better c that can work?

Shannon asked this question and argued that the choice is c is optimal. This is the second part of the theorem that we stated.

The parameter is also a property of the channel, can also be thought of as the limiting value $\frac{k}{n}$. Or rather, the limiting value of how many bits of messages do we communicate by sending one bit of the codeword.

Optimality: We will not formally prove the second part of the theorem in the course, but gave an intuitive argument related to it.

Alice and Bob has a noiseless channel available to them. Two unknown thieves A and B hijacked the channel. Their plan is to send information across using the error part in the coding scheme. They have an object to send (say the content of the piece of paper) which is a probability distribution with probability p that each bit is 1 is 1. They inform Alice abd Bob that the cross-over probability of the binary symmetric channel is p and hence with entropy $H(p)$ (normalized to per bit).

Alice and Bob, believing the false information designed and agreed on a coding scheme

[1] This repair work will affect the value of k that we need to choose to send our information. It also affects the rate $\frac{n}{k}$ at which we are sending the messages but at most additively by $\frac{1}{n}$ which is less than ϵ for large enough n.

which will have some decoding guarantees with respect to the binary symmetric channel presented to them.

The process goes on likes this: A wishes to send out a message η (we used the same notation as the error bit vector). When alice sends out her message m as the encoded codeword $E(m)$, since A hijacked the channel he gets it and adds the bit-vector η to the message to get $E(m) + \eta$. Since the channel is actually noiseless, this is the only "error" and the message $y = E(m) + \eta$ is what we will receive at the other end. B who receives the message passes it on to Bob. Bob will decode the message and retrieve m (with high probability). B will also do the same to get m and then find out $E(m)$. The message η that B was supposed to get can be recovered by $y - E(m) = (E(m) + \eta) - E(m) = \eta$.

The rate of the noiseless channel expressed as the limit of $1 - H(p)$ as $p \to 0$, is 1. Let us calculate the rate at which information is sent out by the parties. For every bit that sent out from one side to another - alice can send $\frac{1}{c} = \frac{k}{n}$ bits of her message and A can send atmost $\frac{1}{H(p)}$ bits of his message.

□