Shannon's Noisy Channel Theorem assures us the existence of capacity-achieving codes. However, exhaustive search for the code has double-exponential complexity: Search over all codebook of size 2^{nR} over all possible $|\mathcal{X}|^n$ codewords.

Plan for today: Constructive version of Shannon's Noisy Channel Theorem. The goal is to show that for BSC, it is possible to achieve capacity in polynomial time. Note that we need to consider three aspects of complexity

- Encoding
- Decoding
- Construction of the codes

22.1 Error exponents

Recall we have defined the fundamental limit

$$M^*(n, \epsilon) = \max\{M : \exists(n, M, \epsilon)\text{-code}\}$$

For notational convenience, let us define its functional inverse

$$\epsilon^*(n, M) = \inf\{\epsilon : \exists(n, M, \epsilon)\text{-code}\}$$

Shannon's theorem shows that for stationary memoryless channels, $\epsilon_n \triangleq \epsilon^*(n, \exp(nR)) \to 0$ for any $R < C = \sup_X I(X;Y)$. Now we want to know how fast it goes to zero as $n \to \infty$. It turns out the speed is exponential, i.e., $\epsilon_n \approx \exp(-nE(R))$ for some error exponent $E(R)$ as a function R, which is also known as the reliability function of the channel. Determining $E(R)$ is one of the most long-standing open problems in information theory. What we know are

- Lower bound on $E(R)$ (achievability): Gallager's random coding bound (which analyzes the ML decoder, instead of the suboptimal decoder as in Shannon's random coding bound or DT bound).
- Upper bound on $E(R)$ (converse): Sphere-packing bound (Shannon-Gallager-Berlekamp), etc.

It turns out there exists a number $R_{\text{crit}} \in (0, C)$, called the critical rate, such that the lower and upper bounds meet for all $R \in (R_{\text{crit}}, C)$, where we obtain the value of $E(R)$. For $R \in (0, R_{\text{crit}})$, we do not even know the existence of the exponent!

Deriving these bounds is outside the scope of this lecture. Instead, we only need the *positivity* of error exponent, i.e., for any $R < C$, $E(R) > 0$. On the other hand, it is easy to see that $E(C-) = 0$ as a consequence of weak converse. Since as the rate approaches capacity from below, the communication becomes less reliable. The next theorem is a simple application of large deviation.

Theorem 22.1. *For any DMC, for any $R < C = \sup_X I(X;Y)$,*

$$\epsilon^*(n, \exp(nR)) \le \exp(-nE(R)), \quad \textit{for some } E(R) > 0.$$

Proof. Fix $R < C$ so that $C - R > 0$. Let P_X^* be the capacity-achieving input distribution, i.e., $C = I(X^*;Y^*)$. Recall Shannon's random coding bound (DT/Feinstein work as well):

$$\epsilon \le P(i(X;Y) \le \log M + \tau) + \exp(-\tau).$$

As usual, we apply this bound with iid $P_{X^n} = (P_X^*)^n$, $\log M = nR$ and $\tau = \frac{n(C-R)}{2}$, to conclude the achievability of

$$\epsilon_n \le P\left(\frac{1}{n} i(X^n;Y^n) \le \frac{C+R}{2}\right) + \exp\left(-\frac{n(C-R)}{2}\right).$$

Since $i(X^n;Y^n) = \sum i(X_k;Y_k)$ is an iid sum, and $\mathbb{E}i(X;Y) = C > (C+R)/2$, the first term is upper bounded by $\exp(-n\psi_T^*(\frac{R+C}{2}))$ where $T = i(X;Y)$. The proof is complete since ϵ_n is smaller than the sum of two exponentially small terms. □

Note: Better bound can be obtained using DT bound. But to get the best lower bound on $E(R)$ we know (Gallager's random coding bound), we have to analyze the ML decoder.

22.2 Achieving polynomially small error probability

In the sequel we focus on BSC channel with cross-over probability δ, which is an additive-noise DMC. Fix $R < C = 1 - h(\delta)$ `bits`. Let the block length be n. Our goal is to achieve error probability $\epsilon_n \le n^{-\alpha}$ for arbitrarily large $\alpha > 0$ in polynomial time.

To this end, fix some $b > 1$ to be specified later and pick $m = b \log n$ and divide the block into $\frac{n}{m}$ sub-blocks of m bits. Applying Theorem 22.1, we can find [later on how to find] an $(m, \exp(Rm), \epsilon_m)$-code such that

$$\epsilon_m \le \exp(-mE(R)) = n^{-bE(R)}$$

where $E(R) > 0$. Apply this code to each m-bit sub-block and apply ML decoding to each block. The encoding/decoding complexity is at most $\frac{n}{m}\exp(O(m)) = n^{O(1)}$. To analyze the probability of error, use union bound:

$$P_e \le \frac{n}{m}\epsilon_m \le n^{-bE(R)+1} \le n^{-\alpha},$$

if we choose $b \ge \frac{\alpha+1}{E(R)}$.

Remark 22.1. The final question boils down to how to find the shorter code of blocklength m in poly(n)-time. This will be done if we can show that we can find good code (satisfying the Shannon random coding bound) for BSC of blocklenth m in exponential time. To this end, let us go through the following strategies:

1. Exhaustive search: A codebook is a subset of cardinality 2^{Rm} out of 2^m possible codewords. Total number of codebooks: $\binom{2^m}{2^{Rm}} = \exp(\Omega(m2^{Rm})) = \exp(\Omega(n^c \log n))$. The search space is too big.

2. Linear codes: In Lecture 16 we have shown that for additive-noise channels on finite fields we can focus on linear codes. For BSC, each linear code is parameterized by a generator matrix, with Rm^2 entries. Then there are a total of $2^{Rm^2} = n^{\Omega(\log n)}$ – still superpolynomial and we cannot afford the search over all linear codes.

3. Toeplitz generator matrices: In Homework 8 we see that it does not lose generality to focus on linear codes with **Toeplitz** generator matrices, i.e., G such that $G_{ij} = G_{i-1,j-1}$ for all $i, j > 1$. Toeplitz matrices are determined by diagonals. So there are at most $2^{2m} = n^{O(1)}$ and we can find the optimal one in poly(n)-time.

Since the channel is additive-noise, linear codes + syndrome decoder leads to the same maximal probability of error as average (Lecture 16).

Remark 22.2. Remark on de-randomization; randomness as a resource, coin flips and cooking (brown both sides of onions)...

22.3 Concatenated codes

Forney introduced the idea of concatenated codes in 1965 to build longer codes from shorter codes with manageable complexity. It consists of an inner code and an outer code:

1. $C_{\text{in}} : \{0,1\}^k \to \{0,1\}^n$, with rate $\frac{k}{n}$
2. $C_{\text{out}} : B^K \to B^N$ for some alphabet B of cardinality 2^k, with rate $\frac{K}{N}$.

The concatenated code $C : \{0,1\}^{kK} \to \{0,1\}^{nN}$ works as follows (Fig. 22.1):

1. Collect the kK message bits into K symbols in the alphabet B, apply C_{out} componentwise to get a vector in B^N
2. Map each symbol in B into k bits and apply C_{in} componentwise to get a nN-bit codeword.

The rate of the concatenated code is the product of the rates of the inner and outer codes: $R = \frac{k}{n}\frac{K}{N}$.

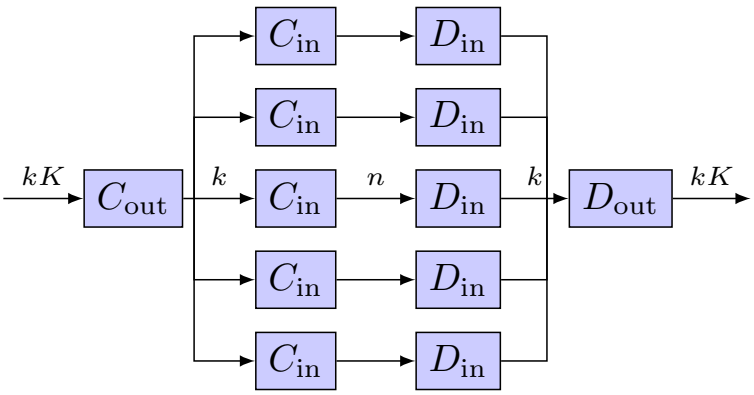

Figure 22.1: Concatenated code, where there are N inner encoder-decoder pairs.

22.4 Achieving exponentially small error probability

Forney proposed the following idea:

- Use an optimal code as the inner code
- Use a Reed-Solomon code as the outer code which can correct a constant fraction of errors.

Reed-Solomon (RS) codes are linear codes from $\mathbb{F}_q^K \to \mathbb{F}_q^N$ where the block length $N = q-1$ and the message length is K. Similar to the Reed-Muller code, the RS code treats the input $(a_0, a_1, \ldots, a_{K-1})$ as a polynomial $p(x) = \sum_{i=0}^{K-1} a_i z^i$ over $\mathbb{F}_q$ of degree at most $K-1$, and encodes it by its values at all non-zero elements. Therefore the RS codeword is a vector $(p(\alpha) : \alpha \in \mathbb{F}_q \backslash \{0\}) \in \mathbb{F}_q^N$. Therefore the generator matrix of RS code is a Vandermonde matrix.

The RS code has the following advantages:

1. The minimum distance of RS code $N - K + 1$. So if we choose $K = (1-\epsilon)N$, then RS code can correct $\frac{\epsilon N}{2}$ errors.
2. The encoding and decoding (e.g., Berlekamp-Massey decoding algorithm) can be implemented in $\mathrm{poly}(N)$ time.

In fact, as we will see later, any efficient code which can correct a constant fraction of errors will suffice as the outer code for our purpose.

Now we show that we can achieve any rate below capacity and exponentially small probability of error in polynomial time: Fix $\eta, \epsilon > 0$ arbitrary.

- Inner code: Let $k = (1 - h(\delta) - \eta)n$. By Theorem 22.1, there exists a $C_{\text{in}} : \{0,1\}^k \to \{0,1\}^n$, which is a linear $(n, 2^k, \epsilon_n)$-code and *maximal* error probability $\epsilon_n \le 2^{-nE(\eta)}$. By Remark 22.1, C_{in} can be chosen to be a linear code with Toeplitz generator matrix, which can be found in 2^n time. The inner decoder is ML, which we can afford since n is small.
- Outer code: We pick the RS code with field size $q = 2^k$ with blocklength $N = 2^k - 1$. Pick the number of message bits to be $K = (1-\epsilon)N$. Then we have $C_{\text{out}} : \mathbb{F}_{2^k}^K \to \mathbb{F}_{2^k}^N$.

Then we obtain a concatenated code $C : \{0,1\}^{kK} \to \{0,1\}^{nN}$ with blocklength $L = nN = n2^{Cn}$ for some constant C and rate $R = (1-\epsilon)(1-h(\delta)-\eta)$. It is clear that the code can be constructed in $2^n = \mathrm{poly}(L)$ time and all encoding/decoding operations are $\mathrm{poly}(L)$ time.

Now we analyze the probability of error: Let us conditioned on the message bits (input to C_{out}). Since the outer code can correct $\frac{\epsilon N}{2}$ errors, an error happens only if the number of erroneous inner encoder-decoder pairs exceeds $\frac{\epsilon N}{2}$. Since the channel is memoryless, each of the N pairs makes an error independently[1] with probability at most ϵ_n. Therefore the number of errors is stochastically smaller than $\mathrm{Binom}(N, \epsilon_n)$, and we can upper bound the total probability of error using Chernoff bound:

$$P_e \le \mathbb{P}\left[\mathrm{Binom}(N,\epsilon_n) \ge \frac{\epsilon N}{2}\right] \le \exp\left(-N d(\epsilon/2 \| \epsilon_n)\right) = \exp\left(-\Omega(N \log N)\right) = \exp(-\Omega(L)).$$

where we have used $\epsilon_n \le \exp(-\Omega(n))$ and $d(\epsilon/2\|\epsilon_n) \ge \frac{\epsilon}{2}\log\frac{\epsilon}{2\epsilon_n} = \Omega(n) = \Omega(\log N)$.

[1] Here controlling the *maximal* error probability of inner code is the key. If we only have average error probability, then given a uniform distributed input to the RS code, the output symbols (which are the inputs to the inner encoders) need *not* be independent, and Chernoff bound is not necessarily applicable.

Note: For more details see the excellent exposition by Spielman [Spi97]. For modern constructions using sparse graph codes which achieve the same goal in *linear* time, see, e.g., [Spi96].

6.441 Information Theory

Spring 2016