Now we will focus on random error model

\[
X + e \xrightarrow{W} Y \xrightarrow{W} X
\]

We will focus on discrete memoryless channel: \( W(y_1, \ldots, y_N | x_1, \ldots, x_N) = \prod_{i=1}^{N} W(y_i | x_i) \)

We encode a message in an \( m \in \{0, \ldots, 2^m-1\} \), send it through the channel and decode it.

When we want \( m^2 + m \in \lambda \cdot \frac{\log |C|}{1 - p} \)

The rate of a code: \( r = \frac{\log |C|}{1 - p} \)

The Shannon capacity of a channel \( W \) is defined as \( I(W) = \log |C| \)

At rate \( I(W) - \epsilon \) reliable communication is possible (error probability \( \leq 2^{-n\epsilon} \))

At rate \( I(W) + \epsilon \) the error probability is large \( (2^n)^{\epsilon} \)

Specific Channels:

BSC: \( X = Y \in \{0,1\} \), \( W(0|1) = 1 - p \), \( W(1|0) = p \)

BEC: \( X \in \{0,1\}, Y \in \{0,1, e\} \), \( W(0|1) = 1 - \alpha \), \( W(1|0) = W(1|1) = \alpha \)

The capacity of:

(a) BSC: \( 1 - H(p) \)

(b) BEC: \( 1 - \alpha \)

The calculation is in the notes; the general idea is using the fact that the typical error will be in a ball of radius \( \epsilon \cdot n \), which has volume \( \frac{e^{n\epsilon} \cdot n}{n!} \).

We can pair codewords in their typical regions are almost disjoint.

Achieving Capacity for BSC

We can have two codes, \( C_{mn} \subseteq \{0,1\}^m \), that can correct \( \epsilon \) fraction of worst case errors, with rate \( (C_{mn}) = 1 - H(\epsilon) \), when \( \epsilon(\epsilon) \to 0 \) as \( n \to \infty \)
with $G_n: (0, y^n) \rightarrow (0, y^n)$ with $\frac{k}{n} = 1 - h(p) - \epsilon$

When $G_n$ is capacity achieving code, $k, B$ are constant so we can decode efficiently by going over all options.

For $B = O(\frac{\log 1}{\log \frac{1}{\epsilon}})$, $b = \mathcal{O}(\log(n))$, $G_n$ code succeeds up to $\epsilon^2 10^n$, so for our $k$, this is lower than $\epsilon$.

The drawback of this coding is the decoding time $O(n^2)$.

It's also not very explicit, as $G_n$ is random, this can be fixed by choosing for each block a different $G_n$ in some order.

Goal: achieve capacity for BSCp with overall complexity bounded by $\text{poly}(\frac{1}{\epsilon})$, (which also implies block length $N = \text{poly}(\frac{1}{\epsilon})$) with rate $1 - h(p) - \epsilon$.

Step 1: reduction to compression (source coding)

We get $m$ independent samples of $X \sim \text{Bernoulli}(\frac{1}{2})$, $X_1, \ldots, X_m$, and we want to compress it to $H(x_1, \ldots, x_m) \in \{(x_1, \ldots, x_m)\}$, so we can recover $x_1, \ldots, x_m$ with high probability.

It's possible as $(x_1, \ldots, x_m)$ are in a ball with radius $\epsilon$, which has volume $\mathcal{O}(\epsilon^{2m})$, so we can enumerate this ball.

But we want linear compression scheme, a linear $H: \mathbb{F}_2^m \rightarrow \mathbb{F}_p^m$, denote $m = (h(p) + \epsilon_0)m$, then $H: \mathbb{F}_2^m \rightarrow \mathbb{F}_p^m$.

A random $H$ will be good. Compression $x \mapsto Hx$, the decomposition is finding the smallest $E$ s.t. $Hx = y$. $\text{Pr}[\text{Decompression}(Hx) \neq y | Hx = y] < \frac{1}{n}$

Claim: Such $H$ gives a capacity achieving binary linear code for the BSCp.

Proof idea: we take $C = \ker(H)$, the encoding is multiplying by a matrix.

The decoder: 1) Compute $Hy$

2) Compute $e = \text{Decompression}(Hy)$

3) Output $y - e$

So, instead of looking for a code we look at linear compression matrix.

We will see that compression is polarizing.
We take $H$ and add rows to get a full rank matrix $P$:

$$
P = \begin{bmatrix}
H \\
A
\end{bmatrix}
$$

Denote:

$$
P: \begin{pmatrix}
x_1 \\
x_2 \\
\vdots \\
x_N
\end{pmatrix} = \begin{pmatrix}
u_1 \\
u_2 \\
\vdots \\
u_N
\end{pmatrix}
$$

$H(x_1)$ is high, so $x_i$ are independent, $H(x_1, \ldots, x_N) = NH(x_1)$.

$P$ is invertible, so $H(u_1, \ldots, u_N) = H(x_1, \ldots, x_N) = NH(x_1)$.

[reminder: conditional entropy $H(X|Y) = \sum_{y \in Y} P(y) H(X|Y=y) \in [0, H(X)]$]

The conditional entropy $H(x_1, x_1, \ldots, x_1) = \log N$, so $x_i$'s are independent.

[reminder: chain rule $H(X,Y) = H(X) + H(Y|X) = H(Y) + H(X|Y)$]

The entropy of $U_i$; using the chain rule:

$$
H(u_1, \ldots, u_N) = H(u_1) + H(u_2|u_1) + \cdots + H(u_N|u_1, \ldots, u_{N-1})
$$

We can also write:

$$
H(u_1, \ldots, u_N) = H(u_1, \ldots, u_m) + H(u_{m+1}, \ldots, u_N)
$$

We choose the matrix $H$ such that we can recover $U_{m+1}, \ldots, U_N$ from $U_1, \ldots, U_m$.

In particular, this means that $H(u_1, \ldots, u_m) \ll \frac{1}{N}$.

It also means that $H(u_1, \ldots, u_m) = H(u_2) + H(u_3|u_2) + \cdots + H(u_m|u_1, \ldots, u_{m-1})$ is almost $mH(u_2)$, as $m = N(\log p + \epsilon)$, we get that all $H$'s entropy are almost $1$.

A polarization matrix $P$ is a matrix such that $(u_1, \ldots, u_N) \rightarrow (x_1, \ldots, x_N)$, if $N \rightarrow \infty$,

$$
\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{i=1}^N H(u_i) = \frac{1}{2}
$$

(all $u_i$'s are either near 0 or near 1.)

Note: a random matrix $P$ is also polarizing, simply not in order.
Insights:
1. Polarizing matrix also implies compression.
2. There is an explicit construction of polarizing matrix with efficient method for \( u_i \) if \( H(U; U_1 \ldots U_{i-1} U_i U_{i+1}) \) is small.

Weight Vector

If we get that \( H(U; U_1 \ldots U_{i-1}) \) is small except for \( (\log \log N) \log 2 \) indices, we must have:
- Low \( H(U; U_1 \ldots U_{i-1}) \leq 1/3 \)
- High \( H(U; U_1 \ldots U_{i-1}) \geq 1/3 \)

\( H \) will be projected to rows in high \( H \) for successful cancellation.

If \( \frac{1}{2} < \delta \), there is a decompression algorithm that succeeds with \( 1 - 2^{-k} \).

(we will have to see that it's efficient)
When $P$ is invertible.

We want the size of $H(y) = \frac{1}{N} H(u_1 u_2 \ldots u_N)$ to satisfy

$$\lim_{N \to \infty} \frac{1}{N} H(y) = H(x)$$

If finite $N$, we would like $\frac{1}{N} H(y) \leq H(x) + \epsilon$ for $N = \text{poly}(\frac{1}{\epsilon})$

In fact, $\frac{1}{N} H(y)$ can be replaced by $2^{-H(y)}$.

The map $P$:

$$N^{-k} : x_0 \rightarrow u_0 \quad x_1 \rightarrow u_1 \quad \cdots \quad x_k \rightarrow u_k$$

$H(u_k) = h(p)$

If we take $P : G = \left( \begin{array}{c} 0 \\ 1 \end{array} \right)$, we get:

$$P(u_0 u_1) = P(x_0 x_1) + P(D x_0 u_1) = \epsilon(p) p$$

$H(u_0) = h(p) + \epsilon(p) p$ for $p \in \text{poly}(\frac{1}{\epsilon})$.

$H(u_0) = \frac{1}{2} H(p) - \epsilon(p) p$

We need to use the second step that are used, so we take $V_0 T_0$ and $V_1 T_1$.

Note: the order of $U_i$'s is important, since we look at conditional entropy in the previous. There is more than one order that works.

We can also look at it as:

Note: it’s not clear what is larger from $H(u_0 u_1)$ and $H(u_0 u_1 u_2)$. 
General $M$ recursive step: we go from $M$ to $2M$

$\begin{align*}
\begin{bmatrix}
X_0 & \cdots & X_M
\end{bmatrix}
\end{align*}$

We can also write:

$\begin{align*}
\mathcal{B}(X_0, \ldots, X_M) &= \left( G^m_X(X_0, \ldots, X_M), G^m_Y(X_0, \ldots, X_M) \right)
\end{align*}$

But this isn't the right order. This order is equivalent to $G^m_X$, we apply bit reversal permutation $\pi_M$ before applying $G^m_X$ (we can move all the permutation to the end).

Summary: the permutation will be $U_0^{-1} = \pi_M G^m_X U_0^{-1}$

A basic step $\mathcal{O}_n$:

$\begin{align*}
W_0 &\rightarrow W_i \\
T_i &\rightarrow U_0
\end{align*}$

$H(W_i, U_0) = H(W_i, U_0, U_0)$

$H(V_i, T_i) = H(V_i, T_i, U_0)$

$W_i$'s and $T_i$'s are independent and have the same distribution.

The process can be looked at as a tree:

- Theorem: the set $\{(X_0, \ldots, X_M) : H(U_0, U_0, \ldots, U_0) \geq \frac{1}{2} \}$ satisfies $|\text{High}| < (2^M)^{1/2}$

- Claim: $H(W_i, U_0) = H(W_i, U_0, U_0)$ for the binary representation of $i$ be $s_0, \ldots, s_M$, and we treat $s_0 = 0$ as (-) and $s_0 = 1$ as (+)
Summer school Vazquez lecture 4: polar codes - page 2

We call $W^*$'s channels $\mathcal{A}_n \mathcal{B}_n$.

Let $H_m$ be the entropy of a random channel in the $m$th level.

$E[H_m] = E[H_0] = H(X)$, so the mean entropy is preserved on all levels of the tree.

We can treat $H_m$ as a martingale, it is also bounded but we will do another calculation.

$E[H_m^2] = E[H_m^2]$, because we have $H_m^2 = H_m^2$ and $0$.

The second moment forms a monotonically bounded sequence, so it converges.

But, we also need to show that it convergence to the largest possible value: $H(X)$.

Entropy increase lemma: if $W^n(A_1B_1)$ is a channel $A_1 \to B_1$ and $H(W^n(A_1B_1))$.

Then $H(W^n) - H(W) \geq \delta(n) > 0$.

This lemma proves the limit convergence. We already know $H_m$ convergence into $H(X)$, meaning $\forall \epsilon > 0$, $\lim_{n \to \infty} \frac{1}{n} \sum_{m=0}^{n-1} E[H_m] > H(X)$. To prove the statement for finite $N$ we need some more work.

The entropy increase lemma follows from a more refined statement:

$H(A_1A_2) - H(A_1B_1)$ is minimized when $B_1 \in \text{supp}(A_1)$, $H(A_1B_1) = H(A_1B_1) = H(A_1) = h(p_1, 1-p_1) - H(p_1)$.

For $\theta = h^{-1}(H(A_1B_1))$.

Claim: $h(p_1, 1-p_1) - h(p) \geq \frac{1}{2} h(p_1, 1-p_1) - h(p_1)$, $p \in (0, 1)$.

(Analytic claim, given without proof.)

From above we get $h(0) = 0$, $h(1) = \frac{1}{2} h(0, 1)$, so that we get Kullback-Leibler convergence with linear step.

The claim helps on Mr. Gerb's lemma. $h(a, b) = h(a, h^{-1}(x))$ is convex in $x$, when $a > 0$, $a = b = a(1-b) + b(1-a)$.
The proof will use the statement \( H(W) - H(w) \geq \frac{1}{n} H(X) (1-H(X)) \).

We will define a potential function \( T_n = \sum X_n H(X_n) \) and prove that:  

**Theorem:** \( \exists c \text{ st } \mathbb{E}[T_n] \leq c \cdot \mathbb{E}[T_1] \).

**Proof idea:** we have \( \mathbb{E}[T_n] \) when we know that \( n \geq \frac{1}{n} H(X) \).

We want to prove that \( \mathbb{E}[T_n] \) is more than \( \frac{1}{n} H(X) \).

It's done by opening the brackets and doing Taylor expansion, and getting:

\[
\frac{1}{n} (\sum X_n H(X_n) + \sum X_n (1-H(X_n))) \leq H(V_{1:n}) - \frac{(1-H)^{n-1}}{8(t-U)^2}
\]

Which can be shown \( \leq \mathbb{E}[T_n] \).

**Remark:** The above theorem also proves some quadratic speed of convergence.

We have proved that \( \exists c \text{ st } \mathbb{E}[H(U | U^n)] > c^n \leq H(X) + c^n \).

And also that \( \mathbb{E}[H(U | U^n)] \leq c^n \).

This bound isn't strong enough, because \( c^n \) can be \( \frac{1}{N^2} \), and we want \( 1 \).

We have proved "weak polarization" - the fraction of high entropy is good, but the bound on the entropy isn't good enough.

---

In order to continue, we need the *Bäcklund parameter* (Hellerer, Fornari, Gallager).

For p.q. distributions on the same set, it's \( \mathcal{Z} \) (finite set).

**Definition:** for a channel \( W(A,B) \), \( A \in \mathcal{A}, B \in \mathcal{B} \), finite set, with joint probability distribution \( p(a,b) \), \( Z(W) = \sum_{b \in \mathcal{B}} p(b | \mathcal{B}) \).

\[
Z(W) = \sum_{b \in \mathcal{B}} \left( p(b | \mathcal{B}) \cdot p(1, b) \right) = \mathbb{E}_{\mathcal{A}} \left[ (p(0, b) \cdot p(b | \mathcal{B}) \right]
\]

\[
\begin{align*}
\mathcal{Z}(W) &= \mathbb{E}_{\mathcal{A}} \left[ (p(0, b) \cdot p(b | \mathcal{B}) \right] \\
\mathcal{Z}(W) &= \mathbb{E}_{\mathcal{A}} \left[ (p(0, b) \cdot p(b | \mathcal{B}) \right]
\end{align*}
\]

For uniform \( a \), an equivalent definition is \( \mathcal{Z}(W) = \mathbb{E}_{\mathcal{A}} \left[ (p(0, b) \cdot p(b | \mathcal{B}) \right] \), which is more similar to the case above.
note: \( Z(W) \) can be used to bound the probability of error
\[
P_{\text{err}} = \sum_{b \in \text{supp}(B)} \min \{ p(0|b), p(1|b) \}
\]

because the less likely thing happened, and we always choose the less likely as the guess.

\[
\min \{ p(0|b), p(1|b) \} \leq \frac{1}{2} Z(W) \quad \text{(also smaller than the entropy)}
\]

**Lemma:** For all channels \( W(A, B) \), \( A \in \{0, 1\} \)
\( Z(W)^* \leq H(W) \leq Z(W) \)

**Proof:**
\[
Z(W) = \mathbb{E}_{b \in B} \left( \frac{1}{2} \left( p(0|b) + p(1|b) \right) \right) \geq \mathbb{E}_{b \in B} \left( \frac{1}{2} \left( h(p(0|b)) + h(p(1|b)) \right) / h(p(0|b)) \right) = H(A|B) = H(W)
\]

\[
Z(W)^* = \mathbb{E}_{b \in B} \left( \frac{1}{2} \left( p(0|b) \cdot p(1|b) \right) \right) \leq \mathbb{E}_{b \in B} \left( \frac{1}{2} \left( h(p(0|b)) \right) \right) = H(W)
\]

**Lemma:** set \( W = (A, B) \), \( A \in \{0, 1\} \) a channel, then:
\( Z(W)^* = Z(W) \) and
\( Z(W) \leq \alpha Z(W) - Z(W)^* \)

(on exercise \( Z(W) \leq \alpha Z(W) \))
Polar codes Page 20

Summar school Verhulst lecture 6.9 - polar codes - page 1

Note: instead of Batacharyya, it's also possible to use entropy, by the entropy lemma we know that \( H(W) = \sum_{b_1} h(p_0) \log_2 \left( \frac{h(b_0)}{h(b_1)} \right) \) with \( p \) calculated by \( W \)'s entropy. We get that \( H(W) = p^* = \frac{h(W)}{\log_2(h(W))} \).

Claim: \( Z(W) = (Z(W))^\alpha \)

Proof: \( Z(W) = \sum_{u,b_2} \left[ p(0,u,b_2) \cdot p(1,u,b_2) \right] \). \( p \) is the channel probability of \( W \).

\[
= \sum_{u,b_2} \left[ p(0,u,b_2) \cdot p(0,0,b_2) \cdot p(0,1,b_2) \cdot p(1,0,b_2) \right] \cdot p \text{ is the channel probability of } W
\]

Explanation of the notation: \( p(0,u,b_2) = \frac{\Pr[A_{2} = 0, A_{1} = u, A_{0} = b_2]}{\Pr[A_{0} = b_2]} \)

\[
= \frac{\Pr[A_{2} = 0, A_{1} = u, A_{0} = b_2]}{\Pr[A_{2} = 0, A_{0} = b_2]} = \frac{\Pr[A_{2} = 0, A_{1} = u, A_{0} = b_2]}{\Pr[A_{0} = b_2]} \cdot \Pr[A_{2} = 0, A_{0} = b_2]
\]

So we removed the sum over \( b_2 \).

\[
= Z(W) \cdot p(0,0) \cdot p(0,1)
\]

We will look at a subtree starting at a good vertices.

and by Chernoff we can claim that we have with exponential high probability, enough + + steps: \( \Pr[ \text{less than } \frac{m}{2} \text{ + + steps} ] < 2^{-m} \).

Now we look only on vertices with at one least 1 + + on the way. The worse case is when all the (-) are to start. So,

In any odd case the output \( Z \) is at most:

\[
\frac{3^m}{2^{\frac{m}{2}}} \cdot (q^{-n})^2 = (2^{-\frac{m}{2}})^2 \leq \left( \frac{1}{2} \right)^2 = \frac{1}{4}
\]

Moreover we need to take \( m \) s.t. \( \frac{3^m}{2^{\frac{m}{2}}} \cdot (q^{-n})^2 < \frac{1}{2} \).

Therefore, if this step is done in stages, the constant in "(5)" can vanish.
This finishes the polarizing theorem.

In order to use it as a code, we need to have an algorithm that efficiently finds which outputs have good high entropy.

Finding the high indices

Exact calculation is too expensive - the alphabet size is squared at each step, so after n steps it's $2^n$ - too large in $N=2^n$.

We can identify the good vertices with in the n steps easily - we count the # of (-) and (+), and take all those with at least $\frac{N}{2}$ (+).

So, we need to approximate the entropy in the next n levels.

We would approximate the entropy of $W(W(W(\ldots W(x,y))))$, we do it by approximating it by $W(x,y)$ when support $\leq N/\text{poly}(N)$ (instead of $2^n$), then we can calculate $H(W)$ directly.

We do it by binning algorithm - we group y's that have similar effects:

$$H(W) = \sum_{y \in \text{supp}(x)} p(y) h(p_x^y)$$

where y's that have same similar value.

We take a range of size $\frac{1}{b}$, we group all y with $h(p_x^y)$ in same range, to a specific $\theta_i$.

Then $p_i(\theta_i) = \sum_{y \in \text{supp}(x)} p(y), p_i(\theta_i) = \sum_{y \in \text{supp}(x)} p(y)$

The variance will be

In the end of the process we get $H(W) = H(\Theta) \leq \log(1 + \frac{\text{variance}}{b})$.

We do this process in each step, and reduce the support to $\Delta$, the approximation error will become less as we get an $\text{supp}(x)$.

Using this, we can get polynomial in N algorithm that computes a superset of $H_{\Delta}$, not much larger than $H_{\Delta}$. 
Decoding: Successive Cancellation Decoder

SC decoder: 

1. For \( i \in \{0, \ldots, N-1\} \)
   a. If \( \hat{v}_i \) is high, we know \( \hat{u}_i \)
   b. Else, \( \mathbb{E}[\hat{u}_i] \) can be calculated as 

\[
\hat{u}_i = \arg\max \{ P[\hat{u}_i = 0 | \hat{v}_0, \ldots, \hat{v}_{i-1}] , P[\hat{u}_i = 1 | \hat{v}_0, \ldots, \hat{v}_{i-1}] \}
\]

Note: we can calculate the conditional probability for a specific value of \( u_0, \ldots, u_{i-1} \) that we've good found in the previous ones.

How do we calculate the probability?

\[
P[\hat{u}_i = 0 | \hat{v}_0, \ldots, \hat{v}_{i-1}] = P[\hat{v}_i = 0, \hat{u}_i = 0 | \hat{v}_0, \ldots, \hat{v}_{i-1}] + 
\]

\[
P[\hat{v}_i = 0, \hat{u}_i = 0 | \hat{v}_0, \ldots, \hat{v}_{i-1}] = P[\hat{v}_i = 0, \hat{u}_i = 0 | \hat{v}_0, \ldots, \hat{v}_{i-1}, u_i = 0] 
\]

Note that \( \hat{v}_i \) and \( \hat{u}_i \) are independent, so we have two problems of half the size. This can be computed recursively in \( O(N^2) \), carefully \( \mathbb{E}[\hat{u}_i] \).

Expander codes

-low density parity check codes.

We will work on binary codes with worst case errors.

\[
\begin{align*}
\text{We can define a code also by parity check matrix } & H, C \subseteq \mathbb{F}_2^n \\
H & = (1, \ldots, 1) \ \text{constant degree } \end{align*}
\]

We can look at \( H \) as a bipartite graph, with \( n \) degree \( d \) (constant).

Definition: \( G \) is a \((d, \delta, \epsilon)\) expander (as small constant, \( d > 0 \) if \( \epsilon \) small enough, \( \epsilon > 0 \) for any \( \epsilon > 0 \)).

\[
|S| < n, \ \ |N(S)| \geq (1-\epsilon)|S| \quad (\text{unbalanced expander})
\]

There are explicit constructions for such \( G \)'s, with \( d(1+\epsilon) = O(1) \), \( m \leq n \) for any \( \epsilon > 0 \).