1 Basics

Polar codes provide an entirely new ways to think about channel coding. The focus is on a new phenomenon that is known as ‘channel polarization’.

Throughout these lectures, we consider a BMS channel with transition probability \( W \) with \( W(y|x) \) for \( x,y \in \mathcal{Y} \). We assume this channel is used \( N \) times, with channel inputs \( X_1, \ldots, X_n \) and corresponding outputs \( Y_1, \ldots, Y_n \). Given a vector \( v \) of \( n \) bits, we will denote by \( v^a \) the subvector of the first \( a \) bits, \( v^b \) the subvector of the last \( b \) bits. For a subset \( S \subseteq [n] \), we denote by \( v_S \) the subvector indexed by elements in \( S \). We will typically think of these as row vectors.

A linear code can be defined through a generating matrix \( G \) encoding \( m \) uncoded bits \( U_1^m \) into \( n \) coded bits \( X_1^n \): \( X_1^n = U_1^m G \). We can always complete such a matrix to a full rank matrix \( G \).

\[
G = \begin{bmatrix}
G_0 & - \\
- & G_1
\end{bmatrix}
\]  \( (1) \)

This matrix defines an invertible transformation \( \{0,1\}^n \to \{0,1\}^n \).

We can therefore think of the encoder as implementing this transformation \( X_1^n = U_1^n G \). Bits \( U_1, \ldots, U_m \) contain information and are i.i.d. uniform in \( \{0,1\} \). Bits \( U_{m+1}, \ldots, U_n \) are instead deterministically equal to 0 and do not contain information.

Summarizing, a code can be specified by an invertible matrix \( G \), and a subset \( I \subseteq [n] \) (the set of information bits, which needs not to me the first \( m \)). This is used as follows:

1. Generate \( U_1 \) uniformly at random in \( \{0,1\}^I \).
2. Complete this vector with zeros on \( F = [n] \setminus I \), to obtain \( U_1^n \) (these are the so-called frozen bits).
3. Transmit the codeword \( X_1^n = U_1^n G \).
4. Receive the channel output \( Y_1^n \).
5. Decode \( U_I \) from \( Y_1^n \) and \( U_F = 0 \).

Note that by symmetry of the channel, it does not really matter the value to which we fix the frozen bits, as long as these are known to the encoder and decoder. For the purpose of analysis, it is convenient to fix them to random values. We have therefore the following process:

1. Generate \( U_I^n \) uniformly at random in \( \{0,1\}^I \).
2. Transmit the codeword \( X_1^n = U_1^n G \).
3. Receive the channel output \( Y_1^n \).
4. Decode \( U_I \) from \( Y_1^n \) and \( U_F = 0 \).

The rate of such a code is of course

\[
r = \frac{|I|}{n}.
\]  \( (2) \)

The problem of designing a code is therefore reduced to the problem of constructing the invertible matrix \( G \) and the information set \( I \). It turns out that significant progress can be made by focusing on a low-complexity decoder known as the ‘successive cancellations decoder.’
2 Successive cancellations decoder

The decoder orders the uncoded bits from $U_1$ to $U_N$, and sequentially decides on bit $i$ only using knowledge about bits $U_1, \ldots, U_{i-1}$. Explicitly, for each $i \in [n]$ we proceed as follows

1. Compute the log-likelihood ratio

$$L_i(y^n_1, \hat{u}^{i-1}_1) = \log \frac{p_{U_i|Y^n_1, U^{i-1}_1}(0|y^n_1, \hat{u}^{i-1}_1)}{p_{U_i|Y^n_1, U^{i-1}_1}(1|y^n_1, \hat{u}^{i-1}_1)}$$

2. Set

$$\hat{u}_i = \begin{cases} U_i & \text{if } i \in F, \\ 0 & \text{if } i \in I \text{ and } L_i(y^n_1, \hat{u}^{i-1}_1) > 0, \\ 1 & \text{if } i \in I \text{ and } L_i(y^n_1, \hat{u}^{i-1}_1) < 0, \end{cases}$$

with ties decided according to a fair coin.

Notice that this algorithm is suboptimal, in particular because to decide bit $i$, we do not use all the information about frozen bits, but only about bits with index before $i$.

Suppose now we are given the transform $G$. How should we choose the sets $F, I$?

Consider bit $U_i$ and, for the sake of argument, assume that bits $U_1, \ldots, U_{i-1}$ have been decoded correctly. Then the decoding for bit $i$ is equivalent to decoding a bit from the BMS channel that takes as input $U_i$ and has output $\tilde{Y}_i = (Y^n_1, U^{i-1}_1)$, see Fig. 1. We will denote this channel by $W_i$, and the output alphabet by $\tilde{Y}_i = Y^n \times \{0, 1\}^{i-1}$. We want this channel to be 'low noise' for all information bits $i \in I$. It is useful to discuss a few possible metrics for the channel quality:

1. The conditional entropy

$$H(W_i) = H(U_i|Y^n_1, U^{i-1}_1).$$

2. The minimum error probability

$$P_e(W_i) = \sum_{\tilde{y}_i} W_i(\tilde{y}_i|0) \left\{ 1_{W_i(\tilde{y}_i|0) < W_i(\tilde{y}_i|0)} + \frac{1}{2} 1_{W_i(\tilde{y}_i|0) > W_i(\tilde{y}_i|0)} \right\}.$$

3. The Bhatthacharya parameter

$$Z(W_i) = \sum_{\tilde{y}_i} \sqrt{W_i(\tilde{y}_i|0)W_i(\tilde{y}_i|1)}.$$
We want the information set to include all positions corresponding to a good channel. A natural definition is therefore

\[ I(\delta) = \{ i \in [n] : P_e(W_i) \leq \delta \} . \]  

(8)

The good news about this choice is that it allows to bound the block error probability under successive cancellation decoder:

\[ P_B(n) = \sum_{i=1}^{n} P(\hat{U}_i \neq U_i; \hat{U}_i^{-1} = U_i^{-1}) \]
\[ = \sum_{i=1}^{n} P_e(W_i) \leq n\delta . \]  

(9)

(10)

Therefore we will have to choose \( \delta = \delta_n = o(1/n) \). Proceeding analogously for the bit error, we obtain that this vanishes if we choose \( \delta_n = o(1) \).

All of the metric introduced above quantities are small when the channel is good. Informally, we can say that

\[ H(W_i) \approx 0 \iff P_e(W_i) \approx Z(W_i) \approx 0 . \]  

(11)

In particular, we will be interested in the following relations.

**Lemma 1.** Define the binary entropy function \( h(q) = -q \log_2 q - (1 - q) \log_2 (1 - q) \). We then have

\[ P_e(W_i) \leq Z(W_i) , \]  

(12)

\[ 2P_e(W_i) \leq H(W_i) \leq h(P_e(W_i)) . \]  

(13)

**Proof** Assume for simplicity that it never happens that \( W_i(\hat{y}|0) = W_i(\hat{y}|1) \):

\[ P_e(W_i) \leq \sum_{\hat{y} \in Y_i} W_i(\hat{y}|0) 1_{W_i(\hat{y}|0) < W_i(\hat{y}|1)} \]
\[ \leq \sum_{\hat{y} \in Y_i} W_i(\hat{y}|0) \sqrt{\frac{W_i(\hat{y}|0)}{W_i(\hat{y}|1)}} = Z(W_i) . \]  

(14)

(15)

This proves the bound \( (12) \).

In order to prove Eq. \( (13) \), let \( \hat{u}_i(\hat{y}) = 0 \) if \( W_i(\hat{y}|0) > W_i(\hat{y}|1) \), \( \hat{u}_i(\hat{y}) = 1 \) if \( W_i(\hat{y}|0) < W_i(\hat{y}|1) \), (recall that we are assuming that it never happens that \( W_i(\hat{y}|0) = W_i(\hat{y}|1) \)). Then

\[ H(W_i) = H(U_i|U_i^{-1}, Y_i^n) \]
\[ = H(U_i, \hat{u}_i(U_i^{-1}, Y_i^n)|U_i^{-1}, Y_i^n) - H(\hat{u}_i(U_i^{-1}, Y_i^n)|U_i, U_i^{-1}, Y_i^n) \]
\[ = H(U_i|\hat{u}_i(U_i^{-1}, Y_i^n), U_i^{-1}, Y_i^n) + H(\hat{u}_i(U_i^{-1}, Y_i^n)|U_i^{-1}, Y_i^n) \]
\[ = \mathbb{E}(h(\mathbb{P}(\hat{u}_i(U_i^{-1}, Y_i^n) \neq U_i|U_i^{-1}, Y_i^n))) . \]  

(16)

(17)

(18)

(19)

Using concavity of \( h \) to obtain an upper bound, and \( h(q) \geq 2q \) to get a lower bound, we have

\[ 2\mathbb{P}(\hat{u}_i(U_i^{-1}, Y_i^n) \neq U_i) \leq H(W_i) \leq h(\mathbb{P}(\hat{u}_i(U_i^{-1}, Y_i^n) \neq U_i)) . \]  

(20)

\[ \square \]

As a consequence of this lemma, we can as well choose the following information set:

\[ J(\delta) = \{ i \in [n] : Z(W_i) \leq \delta \} . \]  

(21)

As long as we use \( \delta = o(1/n) \), the information set \( J(\delta) \) also yields vanishing block error probability as \( n \to \infty \).

Using entropy has a special advantage: the total entropy of all the effective channels is conserved.
Lemma 2. Denoting by $C$ the capacity of channel $W$, we have
\[
\sum_{i=1}^{n} H(W_i) = H(X^n_1|Y^n) = nH(W) = n(1-C).
\] (22)

Proof This is immediate from the chain rule of entropy, once we notice that
\[
H(W_i) = H(U_i|Y^n_i, U_{i-1}^{i-1}).
\] (23)
and that $H(U^n_i|Y^n) = H(X^n_i|Y^n)$ (since there is a one-to-one correspondence between $U^n_i$ and $X^n_i$).

In a sense, the average quality of channels $W_1, \ldots, W_n$ is conserved and equal to $H(W)$. If the transform $G$ is equal to the identity, then all effective channels will have the same quality $H(W_i) = H(W)$. This is clearly not particularly useful.

However something very interesting happens in the opposite limit, namely when all the channels are either very good or very bad (they cannot be all very good!).

Definition 3. We say that the sequence of transforms $(G_n)_{n \in \mathbb{N}}$ polarizes the channel $W$ if, for any $\delta \in (0, 1/2)$,
\[
\lim_{n \to \infty} \frac{1}{n} \# \{i \in [n] : H(W_i) \in (\delta, 1-\delta)\} = 0.
\] (24)

Lemma 4. If the sequence of transforms $(G_n)_{n \in \mathbb{N}}$ polarizes the channel $W$, then selecting the information set $I = I(\delta_n)$, as per Eq. (3), with $\delta_n \to 0$ sufficiently slowly, allows to achieve vanishing bit error probability

Proof Let $X_n$ be a random variable taking value $H(W_i)$ with probability $1/n$, for $i \in \{1, \ldots, n\}$. We know that $\mathbb{E}(X_n) = 1 - C$, and further $\lim_{n \to \infty} \mathbb{P}(X_n \in (\delta, 1-\delta)) = 0$ for any $\delta > 0$. As a consequence, we can choose $\delta_n \to 0$ so that $\lim_{n \to \infty} \mathbb{P}(X_n \in (2\delta_n, 1-2\delta_n)) = 0$. Note that
\[
C = \mathbb{E}(1-X_n) = \mathbb{P}(1-X_n > 2\delta_n) + \mathbb{P}(1-X_n \leq 2\delta_n)2\delta_n,
\] (25)
which implies $\mathbb{P}(X_n < 1 - 2\delta_n) \geq (C - 2\delta_n)/(1-2\delta_n)$, and therefore
\[
\lim_{n \to \infty} \mathbb{P}(X_n \leq 2\delta_n) = \lim_{n \to \infty} \mathbb{P}(X_n - 1 - 2\delta_n) - \lim_{n \to \infty} \mathbb{P}(X_n \in (2\delta_n, 1-2\delta_n))
\] (26)
\[
\geq \lim_{n \to \infty} \frac{C - 2\delta_n}{1-2\delta_n} = C.
\] (27)

Note that the rate of the resulting code is given by
\[
r_n = \frac{|I(\delta_n)|}{n}
\] (28)
\[
= \frac{1}{n} \# \{i \in [n] : \mathbb{P}_e(W_i) \leq \delta_n\}
\] (29)
\[
\geq \frac{1}{n} \# \{i \in [n] : H(W_i) \leq 2\delta_n\}
\] (30)
\[
= \mathbb{P}(X_n \leq 2\delta_n) \to C.
\] (31)

In other words this sequence of codes has rate asymptotically approaching capacity, and bit error rate $P_e(n) \leq \delta_n \to 0$.

It turns out that a stronger form of polarization holds for polar codes, namely
\[
\lim_{n \to \infty} \frac{1}{n} \# \{i \in [n] : H(W_i) \in (\delta_n, 1-\delta_n)\} = 0.
\] (32)
even if we take $\delta_n = \exp(-n^\alpha)$ for $\alpha > 0$ a suitable exponent. As a consequence, the block error rate vanishes as well. (In the standard construction of polar codes, this holds for any $\alpha \in (0, 1/2)$, but improved constructions exist as well.)
3 Channel polarization

Can we construct a polarizing transform? Recall that this means that the codeword is given in terms of the information and frozen bits through $X_1^n = U_1^n G_n$, where $G_n$ is an $n \times n$ binary matrix. Since $G_n$ is invertible, we can also think in terms of its inverse $U_1^n = X_1^n R_n$.

We construct the matrix $R_n$ recursively. Given the matrix for blocklength $n$, we define implicitly $R_{2n}$ by

$$
\tilde{U}_{1,1}^{2n} = (X_1^n R_n + X_{n+1}^{2n} R_{n+1} ; X_{n+1}^{2n} R_n),
\tag{33}
$$

$$
U_{1,1}^{2n} = (\tilde{U}_1, \tilde{U}_{n+1}, \tilde{U}_2, \tilde{U}_{n+2}, \ldots, \tilde{U}_{2n-1}, \tilde{U}_n).
\tag{34}
$$

Equivalently, we can index the entries of $U_{1,1}^{2n}$ as

$$
U_{1,1}^{2n} = (U_{1+}, U_{1-}, U_{2+}, U_{2-}, \ldots, U_{n+}, U_{n-}).
\tag{35}
$$

The component are then given by

$$
\begin{cases}
U_{1+} = U_{i}^{(1)} \oplus U_{i}^{(2)}, \\
U_{1-} = U_{i}^{(2)},
\end{cases}
\quad U^{(1)} = X_1^n R_n, 
\quad U^{(2)} = X_{n+1}^{2n} R_n.
\tag{36}
$$

The transform $R_n$ gives rise to $n$ effective channels $(W_1, W_2, \ldots, W_n)$, and the transform $R_{2n}$ corresponds to $2n$ channels which we denote by $(W_{1+}, W_{1-}, W_{2+}, W_{2-}, \ldots, W_{n+}, W_{n-})$. How are the new channels defined in terms of the old ones? By definition, $W_{i+}, W_{i-}$ are the channels that define the relationships

$$
W_{i+} : U_{i+} \rightarrow (Y_1^{2n}, U_{i+}^{i-1}, U_{i-}^{i-1}),
\tag{37}
$$

$$
W_{i-} : U_{i-} \rightarrow (Y_1^{2n}, U_{i+}^{i}, U_{i-}^{i+}).
\tag{38}
$$

The transition probabilities of these two channels can be constructed once we are given the channel $W_i$. In order to clarify this point, consider a general channel $Q$ and assume that we transmit independent random
bits $U_1$ and $U_2$ through it, to get outputs $Y_1, Y_2$. We then define
\[
\begin{pmatrix}
U_+ \\
U_-
\end{pmatrix} = \begin{pmatrix} 1 & 1 \\
0 & 1 \end{pmatrix} \begin{pmatrix} U_1 \\
U_2 \end{pmatrix},
\] (39)
and construct the channels
\[
T_+Q : U_+ \rightarrow (Y_1, Y_2),
\] (40)
\[
T_-Q : U_- \rightarrow (Y_1, Y_2, U_+).
\] (41)

Namely, we define
\[
T_+Q(y_1, y_2|u_+)P(Y_1 = y_1, Y_2 = y_2|U_+ = u_+),
\] (42)
\[
T_-Q(y_1, y_2, u_+|u_-)P(Y_1 = y_1, Y_2 = y_2, U_+ = u_+|U_- = u_-).
\] (43)

These transition probabilities are explicitly given by
\[
T_+Q(y_1, y_2|u_+) = \frac{1}{2} \sum_{u_1 \oplus u_2 = u_+} Q(y_1|u_1) Q(y_2|u_2),
\] (44)
\[
T_-Q(y_1, y_2, u_+|u_-) = \frac{1}{2} Q(y_1|u_+ \oplus u_-)Q(y_2|u_-).
\] (45)

Then the effective channels at blocklength $2n$ are given by:
\[
(T_+W_1, T_-W_1, T_+W_2, T_-W_2, \ldots, T_+W_n, T_-W_n).
\] (46)

Polar codes are constructing by iterating the construction given above $\ell$ times, starting with $n = 1, G_1 = 1$. We therefore have $n = 2^\ell$. The resulting effective channels are indexed by $s = (s_1, \ldots, s_n) \in \{+1, -1\}^\ell$. Namely, they are given by $\{T_sW\}_{s \in \{+1, -1\}^\ell}$, where:
\[
T_sW = T_{s_n} \circ T_{s_{n-1}} \circ \cdots \circ T_{s_1} W.
\] (47)

**Theorem 5** (Arikan, 2009). The transform $R_n$ defined above polarizes any BMS channel $W$. Namely, for any $\delta \in (0, 1/2)$:
\[
\lim_{\ell \rightarrow \infty} \frac{1}{2^\ell} \# \{s \in \{+1, -1\}^\ell : H(T_sW) \in (\delta, 1 - \delta)\} = 0.
\] (48)

In fact it turns out that the above construction leads to polarization quite generically:

- The input alphabet $\{0, 1\}$ can be replaced by another finite field.
- Instead of combining 2 channels at a time, we can combine $k$ and replace the $2 \times 2$ matrix in Eq. 39 by other $k \times k$ matrices.

4 Analysis: The erasure channel

As usual, the case $W = \text{BEC}(\varepsilon)$ is a useful warm up to understand the general BMS. Notice that in this case each of the channels $T_+W, T_-W$ are BECs: $T_+W = \text{BEC}(\varepsilon_+), T_-W = \text{BEC}(\varepsilon_-)$. Indeed, it either $U_+, U_-$ are completely determined by the corresponding output, or it their are completely unknown (in the sense that their posterior distribution is uniform). It is easy to compute the functions $\varepsilon_\pm(\varepsilon)$.

To compute $\varepsilon_+$ notice that $U_+$ can be determined from the output only if neither $Y_1$ nor $Y_2$ are erased. We thus have
\[
\varepsilon_+ = 1 - (1 - \varepsilon)^2 = 2\varepsilon - \varepsilon^2.
\] (49)
To compute $\varepsilon_-$, recall that this is the erasure probability of the channel $U_- \rightarrow (Y_1, Y_2, U_+)$. Bit $U_-$ can be decoded either from $Y_2$ or from $Y_1$ and $U_+$. We therefore have

$$\varepsilon_- = \varepsilon^2. \quad (50)$$

Note that $\varepsilon_+ + \varepsilon_- = 2\varepsilon$. This is a consequence of the conservation of entropy:

$$2H(W) = H(T_+ W) + H(T_- W). \quad (51)$$

After we repeat this procedure $\ell$ times, we obtain $n = 2^\ell$ effective channels $\{T_s W : s \in \{+1, -1 \}^\ell\}$. By the above argument, these are all BEC’s, and we will denote their erasure probabilities by $\varepsilon_s$, $s \in \{+1, -1 \}^\ell$. In order to prove channel polarization, we need to control the fraction of channels that is not very good nor very bad:

$$\frac{1}{n} \# \{s \in \{+1, -1 \}^\ell : \varepsilon_s \in (\delta, 1-\delta)\}. \quad (52)$$

We can represent this quantity as a probability. Namely, we let $Z_\ell$ be a random variable taking value $\varepsilon_s$ with probability $1/2^\ell$, for each $s \in \{+1, -1 \}^\ell$. The quantity (52) can then be written as $P(Z_\ell \in (\delta, 1-\delta))$.

It is easy to compute the distribution of $Z_\ell$ for $\ell = 0, 1$:

$$Z_0 = \varepsilon, \quad \text{with probability 1}, \quad (53)$$

$$Z_1 = \begin{cases} 2\varepsilon - \varepsilon^2 & \text{with probability 1/2}, \\ \varepsilon^2 & \text{with probability 1/2}. \end{cases} \quad (54)$$

Further, the distribution of $Z_\ell$ can be constructed by recursion:

$$Z_{\ell+1} = \begin{cases} 2Z_\ell - Z_\ell^2 & \text{with probability 1/2}, \\ Z_\ell^2 & \text{with probability 1/2}. \end{cases} \quad (55)$$

Claim 6. For the sequence of random variables $(Z_\ell)_{\ell \geq 0}$ defined above, and any $\delta \in (0, 1/2)$, we have

$$\lim_{\ell \to \infty} P(Z_\ell \in [\delta, 1-\delta]) = 0. \quad (56)$$

Proof Note that $Z_\ell$ is a bounded martingale. This means that $Z_\ell$ is bounded (indeed, it is at most one) and $E(Z_{\ell+1} | Z_\ell) = Z_\ell$.

Define $\varphi(x) = x(1-x)$. Using Eq. (55), we get

$$E[\varphi(Z_{\ell+1}) | Z_\ell] = Z_\ell - 2Z_\ell^2 + Z_\ell^3 - Z_\ell^4 \quad (57)$$

$$= \varphi(Z_\ell) - \varphi(Z_\ell)^2. \quad (58)$$
Hence, letting \( \varphi_\ell \equiv \mathbb{E}\varphi(Z_\ell) \), we get \( \varphi_{\ell+1} \leq \varphi_\ell - \varphi^2_\ell \), and therefore

\[
\sum_{\ell=0}^{\infty} \varphi^2_\ell \leq \sum_{\ell=0}^{\infty} (\varphi_\ell - \varphi_{\ell+1}) \leq \varphi_0 \leq \frac{1}{4}. \tag{59}
\]

This in turn implies \( \varphi_\ell \leq 1/(4\sqrt{\ell}) \rightarrow 0 \), and therefore

\[
\lim_{\ell \to \infty} \mathbb{P}\{Z_\ell \in [\delta, 1 - \delta]\} \leq \lim_{\ell \to \infty} \frac{\mathbb{E}\{\varphi(Z_\ell)\}}{\varphi(\delta)} = 0. \tag{60}
\]

As mentioned above, polarization happens indeed at a faster rate, namely, for any \( \alpha < 1/2 \), we have

\[
\lim_{\ell \to \infty} \mathbb{P}(Z_\ell \leq 2^{-2^\alpha\ell}) = C_{\text{BEC}}(\varepsilon) = 1 - \varepsilon. \tag{61}
\]

We will not prove this fact formally, but it is instructive to try to understand where the limit \( \alpha = 1/2 \) comes from. Fix some small \( \delta_0 \) and define the event

\[
\mathcal{G}(\delta_0, n_0) \equiv \left\{ Z_n \leq \delta_0 \; \forall n \geq n_0 \right\}. \tag{62}
\]

We claim that, for any \( \delta_0 > 0 \), we can take \( n_0 = n_0(\delta_0, \eta) \) large enough so that \( \mathbb{P}(\mathcal{G}(\delta_0, n_0)) \geq C_{\text{BEC}}(\varepsilon) - \eta \).

Also notice that

\[
Z_{\ell+1} = 2\log Z_{\ell} \quad \text{with probability } 1/2, \\
\leq \log Z_{\ell} + \log 2 \quad \text{with probability } 1/2. \tag{63}
\]

Now on the event \( \mathcal{G}(\delta_0, n_0) \), we have \( -\log Z_{\ell} \geq \log 2 \), and therefore

\[
Z_{\ell+1} = 2\log Z_{\ell} \quad \text{with probability } 1/2, \\
\leq (1 - \delta) \log Z_{\ell} \quad \text{with probability } 1/2. \tag{64}
\]

for some \( \delta \) depending on \( \delta_0 \). By the law of large numbers, we obtain \( \log \log Z_1^{-1} \geq \beta \ell \log 2 + o(\ell) \) for \( \beta \in (0, 1/2) \).

5 Encoding and decoding

As in any linear code, encoding amounts to multiplying by the generating matrix: \( X_1^n = U_1^n G_n \). What is the structure of the generating matrix? Note that, by Eqs. (33), (34), we have

\[
R_{2n} = \begin{pmatrix} R_n & 0 \\ R_n & R_n \end{pmatrix} \Pi_{2n},
\]

\[
= (F \otimes R_n) \Pi_{2n}, \tag{65}
\]

where \( \Pi_{2n} \in \{0, 1\}^{2n \times 2n} \) is the permutation matrix defined by \((v_1, v_2, \ldots, v_{2n}) \Pi_{2n} = (v_1, v_{n+1}, v_2, v_{n+2}, \ldots, v_n, v_{2n})\), and

\[
F = \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix}. \tag{66}
\]
Iterating \( \ell = \log_2 n \) times, we get \( \mathbf{R}_n = \mathbf{F}^{\otimes \ell} \mathbf{B}^{-1}_n \), where \( \mathbf{B}^{-1}_n \) is a permutation matrix. Inverting, we get the following form of the generating matrix:

\[
\mathbf{G}_n = \mathbf{B}_n \cdot \mathbf{F}^{\otimes \ell}.
\]

(This uses the fact that \( (\mathbf{A}_1 \otimes \mathbf{A}_2)(\mathbf{C}_1 \otimes \mathbf{C}_2) = (\mathbf{A}_1 \mathbf{C}_1) \otimes (\mathbf{A}_2 \mathbf{C}_2) \).) It turns out that \( \mathbf{B}_n \) is an easy-to-describe matrix: it is the so-called bit-reversing permutation, which can be described as follows. Any vector \( x \in \{0, 1\}^n \) can be indexed by elements in \( \{0, 1\}^\ell \), i.e. can be viewed as a function \( x: \{0, 1\}^\ell \to \{0, 1\} \). Namely

\[
\chi(b_1, b_2, \ldots, b_\ell) = x_i,
\]

whenever \( b_1, b_2, \ldots, b_\ell \) is the binary expansion of \( i - 1 \), i.e. \( i = 1 + \sum_{j=1}^{\ell} b_j 2^{\ell-j} \). Then \( \mathbf{B}_n \) is defined by

\[
(x \mathbf{B}_n)(b_1, \ldots, b_n) = x(b_n, \ldots, b_1).
\]

Also note that \( \mathbf{F}^{\otimes \ell} \) is an Hadamard matrix.

Encoding can therefore be implemented in two steps:

1. **Bit reversing.** We compute \( \hat{u}_1^n = u_1^n \mathbf{B}_n \). This takes \( O(n) \) operations.

2. **Hadamard transform** \( x_1^n = \hat{u}_1^n(\mathbf{F}^{\otimes \ell}) \). This can be implemented recursively:

\[
x_1^n = (x_1^{n/2}, x_{n/2+1}) = (x_1^{n/2}, x_{n/2+1}) \left( \begin{array}{cc} \mathbf{F}^{\otimes (\ell-1)} & 0 \\ 0 & \mathbf{F}^{\otimes (\ell-1)} \end{array} \right)
\]

\[
= (x_1^{n/2} \mathbf{F}^{\otimes (\ell-1)} \oplus x_{n/2+1}^{n/2} \mathbf{F}^{\otimes (\ell-1)}, x_{n/2+1}^{n/2} \mathbf{F}^{\otimes (\ell-1)}).
\]

Hence, denoting the number of operations for this step \( \chi(n) \), we have \( \chi(n) = 2\chi(n/2) + cn \), and therefore \( \chi(n) = O(n \log n) \).

The successive cancellations decoder can also be implemented with complexity \( O(n \log n) \). Recall that, in order to implement this decision rule, we need to compute the effective channel probabilities

\[
\left\{ (W_1^{(n)}(y_1^n, u_1^{i-1} | u_i = 0), W_1^{(n)}(y_1^n, u_1^{i-1} | u_i = 1)) \right\}_{i \leq n},
\]

where we indicated explicitly the dependence of the effective channels on the blocklength \( n \).

The idea is to derive a recursion for these quantities, using the code structure. Let us assume that we know these probabilities for blocklength \( n \). Recall that

\[
u_1^{2n} = (u_{1+}, u_{1-}, \ldots, u_{n+}, u_{n-})
\]

\[
u_+ = u^{(1)} + u^{(2)},
\]

\[
u_- = u^{(2)},
\]

and \( u^{(1)}, u^{(2)} \) are encoded separately using a code of length \( n \), namely

\[
x_1^n = u^{(1)} \mathbf{G}_n \rightarrow y_1^n,
\]

\[
x_{2n+1}^{2n} = u^{(2)} \mathbf{G}_n \rightarrow y_{2n+1}^{2n}.
\]

Hence we have

\[
W_{2n+1}^{(2n)}(y_{2n+1}^{2n}, u_1^{2n} | x) = \mathbb{P}(Y_1^{2n} = y_2^{2n}, U_1^{(1), i-1} = \hat{u}_+^{i-1}, U_1^{(2), i-1} = \hat{u}_-^{i-1} | U_1^{(1)} \oplus U_1^{(2)} = x)
\]

\[
= \sum_{z=0}^{1} \mathbb{P}(Y_1^{n} = y_1^{n}, U_1^{(1), i-1} = \hat{u}_+^{i-1} \oplus \hat{u}_-^{i-1} | U_1^{(1)} = z) \mathbb{P}(Y_{n+1}^{2n} = y_{n+1}^{2n}, U_1^{(2), i-1} = \hat{u}_-^{i-1} | U_1^{(2)} = z \oplus x),
\]
We therefore derived the recursion
\[
W_{2i+1}^{(2n)}(y_1^n, \hat{u}_1^{2i}|x) = \sum_{z=0}^{1} W_i^{(n)}(y_1^n, \hat{u}_i^{2i-1} \oplus \hat{u}_i^{-1}|z) W_{i+1}^{(n)}(y_1^{2n}, \hat{u}_i^{-1}|z \oplus x).
\]

Proceeding analogously, we have
\[
W_{2i+1}^{(2n)}(y_1^n, \hat{u}_1^{2i+1}|x) = W_i^{(n)}(y_1^n, \hat{u}_i^{2i-1} \oplus \hat{u}_i^{-1}|x \oplus \hat{u}_{i+1}) W_{i}^{(n)}(y_1^{2n}, \hat{u}_i^{-1}|x).
\]  

These recursions allow to implement the decoder in time \(O(n \log n)\) (naively it is \(O(n^2)\), but this is sped up to \(O(n \log n)\) by avoiding the same computation to be repeated twice.

### 6 Construction of polar codes

In order to construct a polar code for channel \(W\) we need to be able to compute the error probabilities \(P_e(W_i)\) of the effective channels \(W_1, \ldots, W_n\). Without loss of generality, we can always redefine these channels so that the channel output is the LLR. The basic recursion when doubling the blocklength is based on the following channel-combining operations
\[
W_1 \boxplus W_2(z_1, z_2|u) = \frac{1}{2} \sum_{x \in \{0,1\}} W_1(z_1|u \oplus x)W_2(z_2|x),
\]
\[
W_1 \oplus W_2(z_1, z_2, u'|u) = W_1(z_1|u \oplus u')W_2(z_2|u).
\]

We can then compute the corresponding LLR’s. A simple calculation yield
\[
L_{\boxplus}(z_1, z_2) \equiv \log \frac{W_1 \boxplus W_2(z_1, z_2|u)}{W_1 \oplus W_2(z_1, z_2|1)}
\]
\[
= 2 \text{atanh} \left( \frac{\text{tanh} (z_1/2) \text{tanh} (z_2/2)}{\text{tanh} (z_1/2) - \text{tanh} (z_2/2)} \right),
\]
\[
L_{\oplus}(z_1, z_2, u') \equiv \log \frac{W_1 \oplus W_2(z_1, z_2, u'|u)}{W_1 \oplus W_2(z_1, z_2, u'|1)}
\]
\[
= \begin{cases} 
  z_1 + z_2 & \text{if } u' = 0, \\
  -z_1 + z_2 & \text{if } u' = 1.
\end{cases}
\]

Note that for the sake of analysis (and for the code construction), we can assume that the transmitted codeword is all zeros and therefore \(u' = 0\). In this case, the above recursion reduces to the one we already encountered when studying LDPC codes. The symmetric channels \(W_1, W_2\) are entirely determined by the probability distribution of the LLRs \(z_1, z_2\) when the channel input is 0. Denote these measures by \(a_1, a_2\), and define

- \(a_1 \boxplus a_2\) is the distribution of \(L_{\boxplus}(z_1, z_2) = 2 \text{atanh} \left( \frac{\text{tanh} (z_1/2) \text{tanh} (z_2/2)}{\text{tanh} (z_1/2) - \text{tanh} (z_2/2)} \right)\) when \(z_1 \sim a_1, z_2 \sim a_2\).
- \(a_1 \oplus a_2\) is the distribution of \(L_{\oplus}(z_1, z_2, 0) = z_1 + z_2\) when \(z_1 \sim a_1, z_2 \sim a_2\) (i.e. it is the standard convolution).

If \(a_i^{(n)}, i \leq n\) are the LLR’s distributions for effective channels \(W_1, \ldots, W_n\) at blocklength \(n\), then we have the compact recursions
\[
a_i^{(2n)} = a_i^{(n)} \boxplus a_i^{(n)}, \quad a_i^{(2n)} = a_i^{(n)} \oplus a_i^{(n)},
\]

\[
to be initialized with \(a_1^{(1)} = a_W\) (the density of channel \(W\)).

This recursion can be approximated numerically using the same techniques developed for density evolution, for instance sampled density evolution.