Constructing Polar Codes for Non-Binary Alphabets and MACs

Ido Tal
University of California San Diego,
La Jolla, CA 92093, USA
idotal@ieee.org

Artyom Sharov
Technion,
Haifa, 32000, Israel
sharov@cs.technion.ac.il

Alexander Vardy
University of California San Diego,
La Jolla, CA 92093, USA
avardy@ucsd.edu

***Abstract*—Consider a channel with an input alphabet that is finite but not necessarily binary. A method for approximating such a channel having a large output alphabet size by a degraded version of it having a smaller output alphabet size is presented and analyzed. The approximation method is used to construct polar codes for both single-user and multiple-access channels with prime input alphabet sizes.**

I. Introduction

Polar codes [1] have recently been invented by Arıkan. In his seminal paper, Arıkan introduced polar codes in the context of a binary-input, memoryless, output-symmetric channel over which information is to be sent. The definition of polar codes was soon generalized to a channel with prime input alphabet size in [2]. A further generalization to a polar coding scheme for a multiple-access channel (MAC) with prime input alphabet size is presented in [3] and [4].

The merits of the polar coding schemes presented in [1], [2], [3], and [4] are as follows. Firstly, the schemes are both explicit and symmetric-capacity achieving (sum-rate symmetric-capacity achieving in the MAC setting). Secondly, they have corresponding encoding and decoding algorithms that are efficient. To date, no other family of codes attains all of these properties. However, a major shortcoming of the above schemes is that, although the constructions are *explicit*, the corresponding papers do not suggest an *efficient* method of carrying them out. In fact, in a naive implementation, the construction complexity is exponential in n, the codeword length. Recently, an efficient construction algorithm for the single-user binary-input channel was presented in [5] and analyzed in [6]. Our aim in this paper is to generalize the construction algorithm presented in [5] to single-user channels and MACs with input alphabets having size p, where p is a prime.

The main building block that our construction algorithm will make use of, is a method to approximate a MAC by another MAC. The utility of the approximation is that the new MAC will have a reduced output alphabet size. As was the case in [5], our method will have a tunable fidelity parameter, allowing for an arbitrarily close approximation (in a sense which will shortly be defined). Also, in contrast to the naive implementation which requires running time *exponential* in n, our construction algorithms will run in time *linear* in n. Let the underlying MAC have an input alphabet of size p and t users ($t = 1$ if we are in fact considering a single-user channel). We would like to mention up-front that the running time of our building-block method and thus of our whole algorithm grows very fast in $q = p^t$. Thus, our algorithm can only be argued to be practical for small values of q.

The structure of our paper is as follows. In Section II we setup the basic concepts and notation that will be used later on. Section III introduces the sum-rate of a MAC, and explains why we choose it as the figure of merit of our approximation. Section IV contains our approximation algorithm and Section V outlines how it is used as a building block in constructing a polar code. In Section VI we introduce a more refined analysis of our approximating algorithm.

II. Preliminaries

A. Multiple-access channel

Since a single-user channel is a special case of a MAC, we find it more general to consider MACs. Let $W : \mathcal{X}^t \to \mathcal{Y}$ designate a generic t-user MAC, where $\mathcal{X} = \{0, 1, \ldots, p-1\}$ is the input alphabet, p is prime, and $\mathcal{Y}$ is the finite output alphabet. Denote a vector of user inputs by $\mathbf{u} \in \mathcal{X}^t$, where $\mathbf{u} = (u^{(i)})_{i=1}^t$. Our MAC is specified through the probability function W, where $W(y|\mathbf{u})$ is the probability of observing the output y given that the user input was $\mathbf{u}$.

B. Arıkan transforms

For input vectors $\mathbf{u}_0 = (u_0^{(i)})_{i=1}^t$ and $\mathbf{u}_1 = (u_1^{(i)})_{i=1}^t$, denote by $\mathbf{u}_0 \oplus_p \mathbf{u}_1$ the component-wise sum modulo p of $\mathbf{u}_0$ and $\mathbf{u}_1$. That is,

$$\mathbf{u}_0 \oplus_p \mathbf{u}_1 = (u_0^{(i)} \oplus_p u_1^{(i)})_{i=1}^t \, .$$

Next, denote the Arıkan transforms of P as

$$W^- = W \boxplus W \quad \text{and} \quad W^+ = W \circledast W \, ,$$

where

$$W^- : \mathcal{X}^t \to \mathcal{Y}^2 \quad \text{and} \quad W^+ : \mathcal{X}^t \to \mathcal{Y}^2 \times \mathcal{X}^t$$

are defined as follows:

$$W^-(y_0, y_1|\mathbf{u}_0) = \sum_{\mathbf{u}_1 \in \mathcal{X}^t} \frac{1}{p^t} W(y_0|\mathbf{u}_0 \oplus_p \mathbf{u}_1) \cdot W(y_1|\mathbf{u}_1) \, , \quad (1)$$

$$W^+(y_0, y_1, \mathbf{u}_0|\mathbf{u}_1) = \frac{1}{p^t} W(y_0|\mathbf{u}_0 \oplus_p \mathbf{u}_1) \cdot W(y_1|\mathbf{u}_1) \ . \quad (2)$$

C. Underlying and evolved MACs

Let the underlying MAC through which information is to be transmitted be denoted by $\mathbb{W} : \mathcal{X}^t \to \mathcal{Y}$. Denote the length of the codewords to be used by $n = 2^m$. Given an index $0 \leq i < n$, we recursively define $\mathcal{W}_i^{(m)}$, termed the ith MAC to have evolved from m transforms, as follows. The base of the recursion is the underlying channel:

$$\mathcal{W}_0^{(0)} = \mathbb{W} \ . \quad (3)$$

The recursion step is given by:

$$\mathcal{W}_{2i}^{(m+1)} = (\mathcal{W}_i^{(m)} \circledast \mathcal{W}_i^{(m)}) \ , \quad (4)$$

$$\mathcal{W}_{2i+1}^{(m+1)} = (\mathcal{W}_i^{(m)} \circledast \mathcal{W}_i^{(m)}) \ . \quad (5)$$

As explained in [3] and [4], polar coding is achieved through the evolved MACs. Thus, they are the objects we will be interested in.

D. Degradation

We define the notion of a degraded MAC in an analogous way to that of a degraded single-user channel. Specifically, let two t-user MACs $W : \mathcal{X}^t \to \mathcal{Y}$ and $Q : \mathcal{X}^t \to \mathcal{Y}'$ be given. We say that Q is degraded with respect to W and denote this as $Q \preceq W$ if there exists a function $P : \mathcal{Y} \to \mathcal{Y}'$ such that the following holds. First, P must be a probability function in the following sense: for all $y \in \mathcal{Y}$ and $y' \in \mathcal{Y}'$ we must have that $P(y'|y) \geq 0$. More so, we must have for all $y \in Y$ that

$$\sum_{y' \in \mathcal{Y}'} P(y'|y) = 1 \ .$$

Secondly, the concatenation of P to W must result in Q. That is, for all $y' \in \mathcal{Y}'$ and $\mathbf{u} \in \mathcal{X}^t$ we require that

$$Q(y'|\mathbf{u}) = \sum_{y \in \mathcal{Y}} W(y|\mathbf{u}) \cdot P(y'|y) \ .$$

The following lemma is essentially a restatement of [7, Lemma 4.7].

Lemma 1: Let two t-user MACs $W : \mathcal{X}^t \to \mathcal{Y}$ and $Q : \mathcal{X}^t \to \mathcal{Y}'$ be such that $Q \preceq W$. Denote by Q^- and Q^+ the result of applying the Arıkan transforms to Q and let W^- and W^+ be the result of applying the same transforms on W. Then,

$$Q^- \preceq W^- \quad \text{and} \quad Q^+ \preceq W^+ \ .$$

Namely, degradation is preserved by both Arıkan transforms.

III. The sum-rate criterion

Let a t-user MAC W be given. Next, let $\mathbf{U} = (U^{(i)})_{i=1}^t$ be a random variable uniformly distributed over $\mathcal{X}^t$. Let Y be the random variable one gets as the output of W when the input is $\mathbf{U}$. The sum-rate of W is defined as the mutual information

$$R(W) = I(\mathbf{U}; Y) \ .$$

We first mention the following simplification. As shown in [3, Appendix A], there is a conceptually simple coding scheme in which the sum of the rates of all users[1] can be made to approach $R(W)$. In short: code for user i assuming that the previous $i-1$ users are known to the receiver and the next $t-i$ users are treated as noise. In this scheme, the coding problem essentially reduces to coding for a single-user channel, and we can effectively consider only the case $t = 1$ (which is solved in [5] for $p = 2$). Note that in this scheme, we do not code using the evolved MACs. However, the problem of coding directly via the evolved MACs seems to also have merit, and so we chose to consider the more general case of $t \geq 1$.

In the next section, we show a method by which to approximate one MAC by another MAC with a smaller output alphabet size. The figure of merit we chose to measure our approximation by is the sum-rate. Namely, we give bounds on how much the sum-rate might decrease due to the approximation. At this point, we would like to explain why we chose the sum-rate as the figure of merit. In fact, there are two complementary explanations. First, as was shown in [3] and [4], although the mean symmetric-capacity region is generally reduced after each polarization step, the sum-rate is not. Namely, for every MAC W we have that

$$2R(W) = R(W^-) + R(W^+) \ . \quad (6)$$

This, together with the fact that [3] and [4] show that the sum-rate can be approached by polar coding make $R(W)$ a natural figure of merit. Secondly, the following lemma shows that a given fidelity of $R(Q)$ with respect to $R(W)$ essentially implies the same fidelity with respect to the other mutual informations associated with the MACs $Q \preceq W$. Note that these mutual informations do indeed play a role when constructing a polar coding scheme for a MAC [3][4].

Lemma 2: Let $W : \mathcal{X}^t \to \mathcal{Y}$ and $Q : \mathcal{X}^t \to \mathcal{Y}'$ be a pair of t-user MACs such that $Q \preceq W$ and

$$R(Q) \geq R(W) - \varepsilon \ , \quad (7)$$

where $\varepsilon \geq 0$. Let $\mathbf{U}$ be uniformly distributed over $\mathcal{X}^t$. Denote by Y and Y' the random variables one gets as the output of W and Q, respectively, when the input is $\mathbf{U}$. Let the sets A, B, and C form a partition of the user index set $\{1, 2, \ldots, t\}$. Denote $\mathbf{U}_A = (U^{(i)})_{i \in A}$ and $\mathbf{U}_B = (U^{(i)})_{i \in B}$. Then,

$$I(\mathbf{U}_A; \mathbf{U}_B, Y') \geq I(\mathbf{U}_A; \mathbf{U}_B, Y) - \varepsilon \ . \quad (8)$$

Proof: Let $\mathbf{U}_C = (U^{(i)})_{i \in C}$, and note that both

$$I(\mathbf{U}_A, \mathbf{U}_B, \mathbf{U}_C; Y) = I(\mathbf{U}_B; Y) + I(\mathbf{U}_A; \mathbf{U}_B, Y) + I(\mathbf{U}_C; \mathbf{U}_A, \mathbf{U}_B, Y) \ , \quad (9)$$

and

$$I(\mathbf{U}_A, \mathbf{U}_B, \mathbf{U}_C; Y') = I(\mathbf{U}_B; Y') + I(\mathbf{U}_A; \mathbf{U}_B, Y') + I(\mathbf{U}_C; \mathbf{U}_A, \mathbf{U}_B, Y') \ . \quad (10)$$

[1] In fact, the whole symmetric rate region can be attained using the method in [3, Appendix A], either by time sharing or rate splitting.

By definition, since Q is degraded with respect to W, there exists a corresponding intermediate channel $P : \mathcal{Y} \to \mathcal{Y}'$. Thus, by the data processing inequality applied to Y and Y' through P, we get that each term on the RHS of (9) is an upper bound on the corresponding term in (10). Since

$$\begin{aligned}\varepsilon \geq I(\mathbf{U}_A, \mathbf{U}_B, \mathbf{U}_C; Y) - I(\mathbf{U}_A, \mathbf{U}_B, \mathbf{U}_C; Y') = \\ [I(\mathbf{U}_B; Y) - I(\mathbf{U}_B; Y')] + [I(\mathbf{U}_A; \mathbf{U}_B, Y) - I(\mathbf{U}_A; \mathbf{U}_B, Y')] \\ + [I(\mathbf{U}_C; \mathbf{U}_A, \mathbf{U}_B, Y) - I(\mathbf{U}_C; \mathbf{U}_A, \mathbf{U}_B, Y')] \ , \quad (11)\end{aligned}$$

and each parenthesized difference term of the RHS is positive, we deduce that

$$I(\mathbf{U}_A; \mathbf{U}_B, Y) - I(\mathbf{U}_A; \mathbf{U}_B, Y') \leq \varepsilon \ .$$

■

IV. Degraded approximation

We now start the exposition of our MAC approximation algorithm. As before, consider a t-user MAC $W : \mathcal{X}^t \to \mathcal{Y}$, where $\mathcal{X} = \{0, 1, \ldots, p-1\}$. Let the random variables $\mathbf{U}$ and Y be as before, and define the function $\varphi_W = \varphi : \mathcal{X}^t \times \mathcal{Y} \to [0, 1]$ as follows: for $\mathbf{u} \in \mathcal{X}^t$ and $y \in \mathcal{Y}$,

$$\varphi(\mathbf{u}|y) = \mathbb{P}(\mathbf{U} = \mathbf{u}|Y = y) = \frac{\mathbb{P}(\mathbf{U} = \mathbf{u}, Y = y)}{\mathbb{P}(Y = y)} \ . \quad (12)$$

Note that in our definition of φ, we make the implicit assumption that the output alphabet $\mathcal{Y}$ has been purged of all letters y for which the denominator in (12) is zero, since these outputs will never occur. Next, for $y \in \mathcal{Y}$ let us abuse notation and denote $\varphi_W(y) = \varphi(y)$ as shorthand for

$$\varphi(y) = \mathbb{P}(Y = y) \ .$$

In the interest of brevity, denote

$$\eta(x) = -x \log_2 x \ .$$

Using the above notation, we have that

$$R(W) = t \log_2 p - \sum_{y \in \mathcal{Y}} \varphi(y) \sum_{\mathbf{u} \in \mathcal{X}^t} \eta(\varphi(\mathbf{u}|y)) \ . \quad (13)$$

In a nutshell, our algorithm will merge output letters $y_1, y_2 \in \mathcal{Y}$ if they both fall into the same "bin". We now show how to place output letters into bins.

It is easy to prove that the function $\eta(x)$ is $\cap$-concave on $0 \leq x \leq 1$ and attains its maximum when x is equal to

$$\alpha = \frac{1}{e} \approx 0.3679 \ .$$

For $x = \alpha$ the value of $\eta(x)$ is denoted as

$$\beta = \frac{1}{e \cdot \ln 2} \approx 0.5307$$

(see Figure 1).

Let the positive real μ be a given fidelity parameter and define

$$\widehat{\mu} = \lceil \beta \cdot \mu \rceil \ .$$

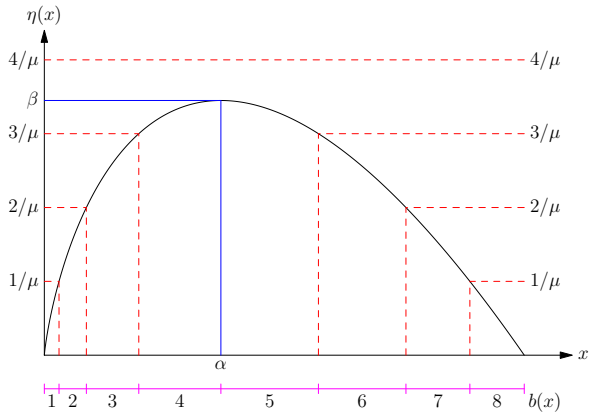

Fig. 1. Functions $\eta(x) = -x \log_2(x)$ and $b(x)$ with $\mu = 6.5$ and $\widehat{\mu} = \lceil \beta \cdot \mu \rceil = 4$.

Next, define the following function $b : [0, 1] \to \{1, 2, \ldots, 2\widehat{\mu}\}$

$$b(x) = \begin{cases} j & \text{if } x < \alpha \text{ and } \frac{j-1}{\mu} \leq \eta(x) < \frac{j}{\mu} \\ \widehat{\mu} & \text{if } x = \alpha \ , \\ 2\widehat{\mu} + 1 - j & \text{if } x > \alpha \text{ and } \frac{j-1}{\mu} \leq \eta(x) < \frac{j}{\mu} \end{cases} \ . \quad (14)$$

The next lemma is a simple consequence of the definition of $b(x)$.

Lemma 3: Let $0 \leq x \leq 1$ and $0 \leq x' \leq 1$ be such that $b(x) = b(x')$. Then,

$$|\eta(x) - \eta(x')| \leq \frac{1}{\mu} \ .$$

We say that two output letters are in the same bin if for all $\mathbf{u} \in \mathcal{X}^t$ we have that $b(\varphi(\mathbf{u}|y_1)) = b(\varphi(\mathbf{u}|y_2))$. The degrading process consists of the following procedure.

- We count the number of non-empty bins, and then construct an output alphabet $\mathcal{Y}'$, the size of which is the number of non-empty bins.
- To each non-empty bin we associate a distinct letter in $\mathcal{Y}'$. For a $y' \in \mathcal{Y}'$, denote by $\mathcal{B}(y')$ all the $y \in \mathcal{Y}$ in the bin associated with y'.
- The degraded MAC is obtained from the original MAC through an intermediate channel $P : \mathcal{Y} \to \mathcal{Y}'$. The channel maps (with probability 1) each letter $y \in \mathcal{Y}$ to the letter $y' \in \mathcal{Y}'$ associated with the bin y is in. That is, each member of $\mathcal{B}(y')$ is mapped to y'.

Denote the MAC one gets by the above degrading procedure applied to a MAC W by Q. By definition, Q is a t-user MAC which is degraded with respect to W. Let Y' be the random variable one gets as the output of Q when the input is $\mathbf{U}$. Clearly, for all $y' \in \mathcal{Y}'$ and $u \in \mathbf{U}$,

$$\mathbb{P}(Y' = y'|\mathbf{U} = \mathbf{u}) = \sum_{y \in \mathcal{B}(y')} \mathbb{P}(Y = y|\mathbf{U} = \mathbf{u}) \ .$$

Lemma 4: Let μ be specified and $y' \in \mathcal{Y}'$ be given. Then, for all $y \in \mathcal{B}(y')$ and for all $\mathbf{u} \in \mathcal{X}^t$,

$$b(\varphi_W(\mathbf{u}|y)) = b(\varphi_Q(\mathbf{u}|y')) \ .$$

That is, loosely speaking, if y' were to be binned, it would occupy the same bin as (any) $y \in \mathcal{B}(y')$.

Proof: The main point to notice is that for a fixed $1 \leq j \leq 2\widehat{\mu}$, the set of x for which $b(x) = j$ is contiguous (as is exemplified in Figure 1). To see this, recall the definition of $b(x)$ in (14) and note that $\eta(x)$ is strictly increasing for $0 \leq x < \alpha$ and strictly decreasing for $\alpha < x \leq 1$.

To finish the proof, it suffices to show that $\varphi_Q(\mathbf{u}|y')$ is a convex combination of $\varphi_W(\mathbf{u}|y)$, $y \in \mathcal{B}(y')$, since this implies that $\varphi_Q(\mathbf{u}|y')$ is contained in the contiguous set all the $\varphi_W(\mathbf{u}|y)$ are members of. Indeed, it can be easily seen that

$$\varphi_Q(\mathbf{u}|y') = \sum_{y \in \mathcal{B}(y')} \frac{\varphi_W(y)}{\varphi_Q(y')} \cdot \varphi_W(\mathbf{u}|y) \tag{15}$$

and

$$\varphi_Q(y') = \sum_{y \in \mathcal{B}(y')} \varphi_W(y) \ . \tag{16}$$

■

Theorem 5: Let W be a t-user MAC and let Q be the MAC one gets by applying the degrading operation described above with fidelity criterion μ. Then,

$$R(Q) \geq R(W) - \frac{p^t}{\mu} \ .$$

Proof: By (13) and (16), we can write the difference $R(W) - R(Q)$ as

$$R(W) - R(Q) = \sum_{y' \in \mathcal{Y}'} \sum_{y \in \mathcal{B}(y')} \varphi(y) \sum_{\mathbf{u} \in \mathcal{X}^t} \left[\eta(\varphi_Q(\mathbf{u}|y')) - \eta(\varphi_W(\mathbf{u}|y))\right] \ . \tag{17}$$

Considering the innermost sum, we have by Lemma 4 that

$$b(\varphi_W(\mathbf{u}|y)) = b(\varphi_Q(\mathbf{u}|y')) \ .$$

Thus, by Lemma 3 we conclude that each innermost term has absolute value at most $1/\mu$. Plugging this in yields

$$R(W) - R(Q) \leq \sum_{y' \in \mathcal{Y}'} \sum_{y \in \mathcal{B}(y')} \varphi(y) \sum_{\mathbf{u} \in \mathcal{X}^t} \frac{1}{\mu} = \sum_{y' \in \mathcal{Y}'} \sum_{y \in \mathcal{B}(y')} \varphi(y) \cdot \frac{p^t}{\mu} = \frac{p^t}{\mu} \ .$$

■

Recall that the point of running the degrading approximation was to reduce the size of the output alphabet. The following lemma gives an upper bound on its size.

Lemma 6: Let $W : \mathcal{X}^t \to \mathcal{Y}$ be a t-user MAC and let $Q : \mathcal{X} \to \mathcal{Y}'$ be the MAC one gets by applying the degrading operation described above with fidelity criterion μ. Recall that $q = p^t$. Then,

$$|\mathcal{Y}'| \leq (2\widehat{\mu})^q \leq (2\mu)^q \ .$$

Proof: Recall that two letters $y_1, y_2 \in \mathcal{Y}$ are in the same bin if and only if $b(\varphi(\mathbf{u}|y_1))$ and $b(\varphi(\mathbf{u}|y_2))$ are equal for all $\mathbf{u} \in \mathcal{X}^t$. The proof follows by recalling that the number of values $\mathbf{u}$ can take is q and the number of values $b(\cdot)$ can take is $2\widehat{\mu}$. ■

V. Construction algorithm

In this section we outline a construction method for polar codes. Due to space limitations, we only show and analyze the algorithm used to approximate $\mathcal{W}_i^{(m)}$. That is, we show here the analog of [5, Algorithm A]. In order to actually construct polar codes, one needs an analog of [5, Algorithm C]. However, getting from one algorithm to the other, with [3], [4], and [5] as references, should be rather straightforward.

For a given channel index $0 \leq i < n$, denote by $i = \langle b_1, b_2, \ldots, b_m \rangle_2$ the binary representation of i, where b_1 is the most significant bit. Also, let $\texttt{degrading\_merge}(W, \mu)$ be the result of applying the approximation method outlined in Section IV to a MAC W using the fidelity parameter μ.

Algorithm A: A high level description of the degrading procedure

input : An underlying MAC $\mathbb{W}$, a fidelity parameter μ, an index $i = \langle b_1, b_2, \ldots, b_m \rangle_2$.
output: A MAC that is degraded with respect to $\mathcal{W}_i^{(m)}$.

```
Q ← degrading_merge(𝕎, μ);
for j = 1, 2, ..., m do
    if b_j = 0 then
        W ← Q ⊞ Q
    else
        W ← Q ⊛ Q
    Q ← degrading_merge(W, μ);
return Q;
```

Note that since we trim the output alphabet size after each iteration, it does not grow out of control. Specifically, it is easy to show that the output alphabet size of each MAC encountered during the run of Algorithm A, except for possibly $\mathbb{W}$, is at most $q \cdot (2\mu)^{2q}$. Thus, if p, t, and μ are regarded as constants, and i ranges from 0 to $n-1$, the running time of the algorithm (assuming calculations are shared) is a constant times $n = 2^m$, the codeword length.

The following theorem gives an upper bound on the average loss in sum-rate due to running Algorithm A. It is a direct consequence of Lemma 1 and Theorem 5

Theorem 7: Let an underlying t-user MAC $\mathbb{W} : \mathcal{X}^t \to \mathcal{Y}$ be given, where $\mathcal{X} = \{0, 1, \ldots, p-1\}$ and p is prime. Denote by $\mathcal{Q}_i^{(m)}$ the channel returned by running Algorithm A with parameters i and μ. Then,

$$\frac{1}{n} \sum_{0 \leq i < n} \left(R(\mathcal{W}_i^{(m)}) - R(\mathcal{Q}_i^{(m)}) \right) \leq \frac{m \cdot p^t}{\mu} \ .$$

VI. Improvements

Recall that the bound derived in Theorem 5 came about by bounding the difference in the innermost sum in (17) by $1/\mu$. We now show how to rewrite the difference $R(W) - R(Q)$ in a slightly different manner, which will lead to a tighter bound.

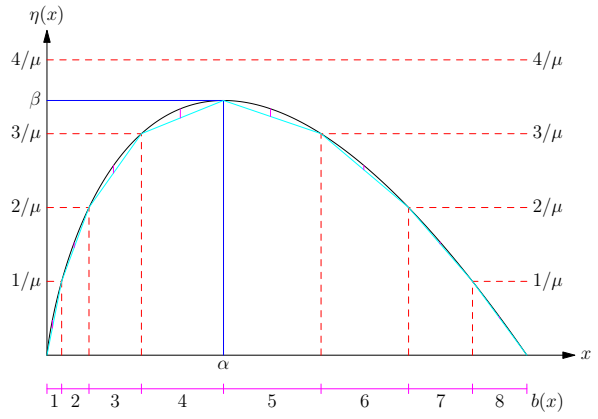

Fig. 2. Function $\eta(x) = -x\log_2(x)$. The horizontal magenta lines are a depiction of the improved bound one gets by the application of Lemma 8.

Rearranging the sums in (17) and recalling (16), we have that

$$R(W) - R(Q) = \sum_{y' \in \mathcal{Y}'} \sum_{\mathbf{u} \in \mathcal{X}^t} \varphi_Q(y') \eta(\varphi_Q(\mathbf{u}|y')) - \sum_{y \in \mathcal{B}(y')} \varphi_W(y) \eta(\varphi_W(\mathbf{u}|y)) \ .$$

Next, recalling the definition of $\varphi_Q(\mathbf{u}|y')$ given in (15), we have that

$$R(W) - R(Q) = \sum_{y' \in \mathcal{Y}'} \varphi_Q(y') \sum_{\mathbf{u} \in \mathcal{X}^t} \left(\eta \left[\sum_{y \in \mathcal{B}(y')} \frac{\varphi_W(y)}{\varphi_Q(y')} \cdot \varphi_W(\mathbf{u}|y) \right] - \left[\sum_{y \in \mathcal{B}(y')} \frac{\varphi_W(y)}{\varphi_Q(y')} \eta(\varphi_W(\mathbf{u}|y)) \right] \right) \ .$$

We now focus on bounding the difference

$$\eta \left[\sum_{y \in \mathcal{B}(y')} \frac{\varphi_W(y)}{\varphi_Q(y')} \cdot \varphi_W(\mathbf{u}|y) \right] - \left[\sum_{y \in \mathcal{B}(y')} \frac{\varphi_W(y)}{\varphi_Q(y')} \eta(\varphi_W(\mathbf{u}|y)) \right] \ . \quad (18)$$

Recall that by the binning operation, we have that $b(\varphi_W(\mathbf{u}|y))$ has the same value for all $y \in \mathcal{B}(y')$. This observation implies that the following is well-defined. Pick some $y \in \mathcal{B}(y')$ and denote by $I_{y'}$ the interval that is the pre-image of $b(\varphi_W(\mathbf{u}|y))$ with respect to the binning function b:

$$I_{y'} = \{x \ : \ b(x) = b(\varphi_W(\mathbf{u}|y)\} \ , \quad \text{where } y \in \mathcal{B}(y') \ .$$

The following lemma implies a simple upper bound on (18). For a graphical representation, see Figure 2.

Lemma 8: Let W, u, and y' be given. Denote by a and b the infimum and supremum of $I_{y'}$, respectively. Then, the difference in (18) is at most

$$\max_{0 \le \theta \le 1} \left\{ \eta\left[\theta \cdot a + (1-\theta) \cdot b\right] - \left[\theta \cdot \eta(a) + (1-\theta) \cdot \eta(b)\right] \right\} \ . \quad (19)$$

More so, the θ maximizing the above expression is given by

$$\theta_{\max} = \frac{b - \frac{1}{e} \cdot 2^{\frac{-(\eta(b)-\eta(a))}{b-a}}}{b-a} \ .$$

Proof: Obtaining the expression for $\theta_{\max}$ is an easy application of calculus. We now focus on the first part of the theorem, and begin by setting up some notation. Let,

$$\gamma_y = \frac{\varphi_W(y)}{\varphi_Q(y')} \ , \quad \text{and} \quad \delta_y = \varphi_W(\mathbf{u}|y) \ .$$

Recall that

$$\sum_{y \in \mathcal{B}(y')} \gamma_y = 1 \ ,$$

and that for each δ_y there is a corresponding $0 \le \theta_y \le 1$ such that

$$\delta_y = \theta_y \cdot a + (1-\theta_y) \cdot b \ .$$

Next, we set

$$\theta = \sum_{y \in \mathcal{B}(y')} \gamma_y \cdot \theta_y \ .$$

and rewrite (18) as,

$$\eta\left[\theta \cdot a + (1-\theta) \cdot b\right] - \left[\sum_{y \in \mathcal{B}(y')} \gamma_y \cdot \eta(\delta_y) \right] \le \eta\left[\theta \cdot a + (1-\theta) \cdot b\right] - \left[\sum_{y \in \mathcal{B}(y')} \gamma_y \cdot (\theta_y \cdot \eta(a) + (1-\theta_y) \cdot \eta(b)) \right] = \eta\left[\theta \cdot a + (1-\theta) \cdot b\right] - \left[\theta \cdot \eta(a) + (1-\theta) \cdot \eta(b)\right] \ ,$$

where the first step follows from Jensen's inequality. Recall that here we have set θ to a specific value, whereas in (19) θ is optimized. Thus, the last displayed equation is a lower bound on the expression given in (19). ■

Acknowledgments

We would like to thank Eren Şaşoğlu for very productive discussions.

References

[1] E. Arıkan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. Inform. Theory*, vol. 55, pp. 3051–3073, 2009.

[2] E. Şaşoğlu, E. Telatar, and E. Arıkan, "Polarization for arbitrary discrete memoryless channels," `arXiv:0908.0302v1`.

[3] E. Şaşoğlu, E. Telatar, and E. Yeh, "Polar codes for the two-user multiple-access channel," `arXiv:1006.4255v1`.

[4] E. Abbe and E. Telatar, "Polar codes for the m-user MAC and matroids," `arXiv:1002.0777v2`.

[5] I. Tal and A. Vardy, "How to construct polar codes," *submitted to IEEE Trans. Inform. Theory, available online as* `arXiv:1105.6164v2`, 2011.

[6] R. Pedarsani, S. H. Hassani, I. Tal, and E. Telatar, "On the construction of polar codes," in *Proc. IEEE Int'l Symp. Inform. Theory (ISIT'2011)*, Saint Petersburg, Russia, 2011, pp. 11–15.

[7] S. B. Korada, "Polar codes for channel and source coding," Ph.D. dissertation, Ecole Polytechnique Fédérale de Lausanne, 2009.