Coding on Barrier Channels beyond Guaranteed Correction

Yuval Ben-Hur and Yuval Cassuto
The Andrew and Erna Viterbi Department of Electrical and Computer Engineering
Technion - Israel Institute of Technology, Technion City, Haifa 3200003, Israel
Email: {yuvalbh@campus, ycassuto@ee}.technion.ac.il

***Abstract*—This paper studies coding on channels with the barrier property: only errors to and from a special barrier state are possible. Our contributions include derivation of the channel capacity, efficient maximum-likelihood (ML) and list decoding algorithms, and finite-block-length analysis using random codes. Emerging non-volatile memory technologies may exhibit controlled unreliability as their representation power is increased, and thus may benefit from the high capacity and improved coding of barrier channels.**

I. Introduction

Given the exponential growth in data volumes world-wide, data-storage technologies are constantly required to support higher information densities. One of the keys toward denser data-storage devices is scaling the number of representation levels per memory cell. The most common scaling principle is the *multi-bit cell*, in which an integer power of 2 number of levels are uniformly spaced across the dynamic range. In this paper we study a different scaling principle called the *barrier multi-level cell*. This principle requires a media technology (e.g., NextGen EEPROM [2]) in which one of the Q levels is designated as the *barrier level*, and the dominant error transitions are from non-barrier levels to the barrier, and from the barrier to non-barrier levels.

A sample motivation for this model is the ternary-input additive white Gaussian channel, shown in Fig. 1a (note the negligible transition probability between the two outer levels). We define the corresponding dual-parameter channel, $W_Q(p,q)$, as in Fig. 1b for $Q = 3$. Two single-parameter special cases of the barrier channel have already been studied: [1] addresses the one-directional case that only allows error transitions to the barrier level; and [2] studies another case in which the transition probabilities to and from the barrier level are equal.

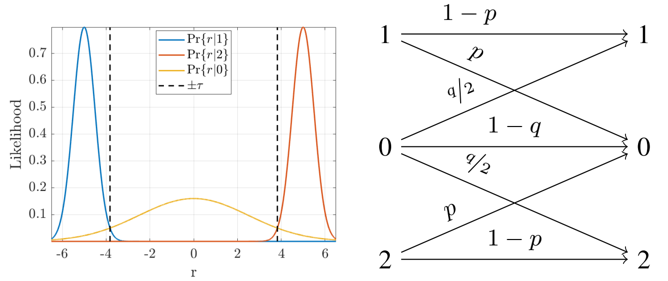
II. Channel Capacity of $W_3(p,q)$

We derive an analytic expression for the dual-parameter barrier channel capacity. Although given for the special case $Q = 3$, it can be extended to the general case. To simplify the following expressions, we define the functions $\beta_{p,q}(\varphi) \triangleq q + (1-p-q)\,\varphi$ and $\gamma(a) \triangleq h_2(a) + a$, where $h_2(\cdot)$ is the binary entropy function.

Theorem 1: For $p + q < 1$, the capacity of $W_3(p,q)$ equals

$$\gamma\left(\beta_{p,q}(\varphi^*)\right) - \varphi^* h_2(p) - (1-\varphi^*)\gamma\left(q\right), \tag{1}$$

This paper is based on a paper that was published at the 2021 IEEE Global Communications Conference [3].



(a) Cell-level distribution of a ternary-input AWGN channel.
(b) The ternary dual-parameter barrier channel ($Q = 3$).

where $\varphi^* \triangleq \min\left\{ \frac{1-q-\left(1+2^{-\frac{\gamma(p)-1-h_2(q)}{1-p-q}}\right)^{-1}}{1-p-q}, 1 \right\}$.

Fig. 2 depicts the channel capacity as a function of $p + q$, for several relations between p and q. The plots motivate the study of the *dual*-parameter version of the channel, showing examples for interesting parameter regimes not covered by the known single-parameter special cases $q = 0$ and $p = q/2$.

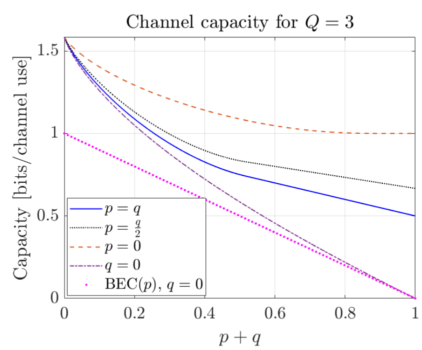


Fig. 2: Capacity of $W_3(p,q)$ for several p, q relations.

III. Decoding Algorithms over $W_Q(p,q)$

A construction method for ternary codes correcting t barrier errors using a pair of binary Hamming-metric constituent codes was suggested in [2]. Extending to Q-ary codes, each codeword $\mathbf{c} \in \mathcal{C}$ is composed of two constituent codewords: (1) $\boldsymbol{\theta} \in \Theta$ where Θ is a binary code with length n and minimum *Hamming* distance $2t + 1$; and (2) $\boldsymbol{\lambda} \in \Lambda_{w_H(\boldsymbol{\theta})}$ where Λ_w is a $(Q-1)$-ary code with length w and minimum *Hamming* distance $t + 1$. The locations of zeros in $\mathbf{c}$ are determined by the locations of zeros in $\boldsymbol{\theta}$, and the remaining positions are set to the symbols of $\boldsymbol{\lambda}$ offset by $+1$.

A. Decomposed MLD using simpler constituent MLDs

In [2], an ML decoder (for the special case $p = q/2$) is defined through a distance metric on the ternary alphabet $\mathbb{Z}_3$. We show that ML decoding for $\mathcal{C}$ can be performed more efficiently by sequentially invoking the simpler ML decoders of the constituent codes Θ and Λ_w. This reduction is advantageous since the constituent codes are traditional Hamming-metric codes over lower-order alphabets.

Let $\mathbf{r} = (r_1, \ldots, r_n) \in \mathbb{Z}_Q^n$ be the word output from the channel $W(p,q)$. We define the *indicator mapping* from the alphabet of $\boldsymbol{r}$ to the binary alphabet of Θ, and then define the *ML indicator decoder*.

Definition 1: Let $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{Z}_Q^n$. The **indicator mapping** $\imath(\mathbf{x}) = (\imath(x_1), \ldots, \imath(x_n))$ is defined as

$$\imath(x_j) = \begin{cases} 1, & x_j \in \mathbb{Z}_Q \setminus \{0\} \\ 0, & x_j = 0 \end{cases} . \tag{2}$$

Definition 2: The **ML indicator decoder** for Θ outputs

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta} \in \Theta} \{\mu_1 \imath(\mathbf{r}) \boldsymbol{\theta}^T - \mu_2 w_H(\boldsymbol{\theta})\}, \tag{3}$$

where $\mu_1 \triangleq \log\left(\frac{(Q-1)(1-p)(1-q)}{pq}\right)$ and $\mu_2 \triangleq \log\left(\frac{1-q}{p}\right)$. It can be proved that $\hat{\boldsymbol{\theta}}$ in (3) maximizes the indicator vector's likelihood function [3]. Again, before moving to the residual ML decoder, we first define a mapping from the channel output to the decoder input.

Definition 3: Given $\boldsymbol{\theta} \in \{0,1\}^n$, the **residual mapping** maps a vector $\mathbf{r} \in \mathbb{Z}_Q^n$ to a vector $\rho^{(\boldsymbol{\theta})}(\mathbf{r}) = \left(\rho_1^{(\boldsymbol{\theta})}(\mathbf{r}), \ldots, \rho_{w_H(\boldsymbol{\theta})}^{(\boldsymbol{\theta})}(\mathbf{r})\right)$ such that for every $1 \le j \le n$ with $\theta_j = 1$,

$$\rho_{\sigma_j(\boldsymbol{\theta})}^{(\boldsymbol{\theta})}(\mathbf{r}) = \begin{cases} ?, & r_j = 0 \\ r_j - 1, & otherwise \end{cases} \tag{4}$$

Definition 4: Given $\boldsymbol{\theta} \in \{0,1\}^n$, the **ML residual decoder** for $\Lambda_{w_H(\boldsymbol{\theta})}$ first finds all the codewords $\boldsymbol{\lambda}$ such that $\lambda_i = \rho_i^{(\boldsymbol{\theta})}(\mathbf{r})$ for every i with $\rho_i^{(\boldsymbol{\theta})}(\mathbf{r}) \neq ?$. Then the decoder outputs $\hat{\boldsymbol{\lambda}}$ if unique, "fail" if multiple codewords were found, and "reject" if none were found.

As described in Algorithm 1, our decoder is formulated using a combination of the two constituent MLDs. It can be proved that our decoder is ML equivalent [3].

Algorithm 1 MLD for $\mathcal{C} = \Theta \otimes \{\Lambda_w\}_w$:

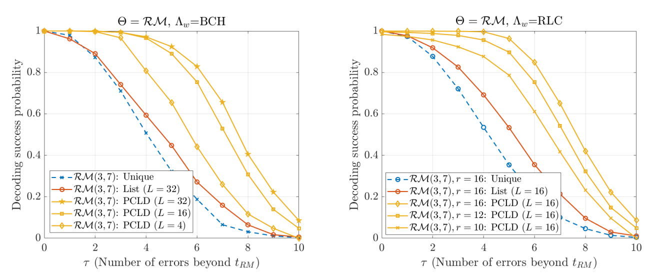
<u>*Input*</u> : $\mathbf{r} \in \mathbb{Z}_Q^n$ - channel output
<u>*Output*</u> : $\hat{\mathbf{c}} \in \mathbb{Z}_Q^n$ - decoded codeword
Initialize: $\Theta' \leftarrow \Theta$
while not returned **do**
  set $\hat{\boldsymbol{\theta}}$ to the output of indicator MLD with input $\imath(\mathbf{r})$ and code Θ'
  invoke residual MLD with input $\rho^{(\hat{\boldsymbol{\theta}})}(\mathbf{r})$ and code $\Lambda_{w_H(\hat{\boldsymbol{\theta}})}$
  if "reject" **then**
    $\Theta' \leftarrow \Theta' \setminus \hat{\boldsymbol{\theta}}$
  else if "fail" **then**
    return *decoding failure*
  else
    return $\hat{\mathbf{c}}$ reconstruced by $\hat{\boldsymbol{\theta}}$ and $\hat{\boldsymbol{\lambda}}$
  end if
end while

B. (P)CLD: (Persistent) Cooperative List Decoding

Since practical ML decoding is hard even for the simpler constituent codes, we propose a simplification of Algorithm 1 using *list decoding*, which is a more tractable computational task. We define two variants of the decoder, inspired by Algorithm 1.

1) The ***Cooperative list decoder (CLD)*** uses a list decoder to obtain a list of likely codewords $\{\hat{\boldsymbol{\theta}}_l\}_{l=1}^L$, where L is the list size. Then, cooperative list decoding is identical to Algorithm 1, besides the initialization $\Theta' \leftarrow \{\hat{\boldsymbol{\theta}}_l\}_{l=1}^L$ (instead of $\Theta' \leftarrow \Theta$).
2) The ***Persistent Cooperative list decoder (PCLD)*** is identical to CLD, but instead of returning "decoding failure" when residual MLD fails, it continues to the next codeword in the list (this amounts to merging the 'else if' of Algorithm 1 into the 'if' statement, which will now be: if "reject" or "fail").

We evaluate the PCLD performance for the ternary channel $W_3(p,q)$ using widely adopted binary codes for Θ and Λ_w. In Fig. 3a, the successful block decoding probability of PCLD is presented for Reed-Muller ($\mathcal{RM}$) codes as the indicator code and modified (shortened/lengthened) BCH codes as residual codes. Fig. 3b presents similar results for residual random linear code (RLC) with a prescribed redundancy r. With the latter construction, the failure/reject probabilities of the residual decoder are calculated analytically, saving the need to simulate the residual decoder. For both constructions, PCLD extends the code's correction capability beyond its designed guarantees. Note also that the proposed PCLD significantly outperforms the standard (non-CLD) list decoder with the same list size.



(a) $\mathcal{RM}$ and BCH (b) $\mathcal{RM}$ and RLC analysis

Fig. 3: Probability of successful decoding

References

[1] R. Ahlswede and H. Aydinian, "Error control codes for parallel asymmetric channels," IEEE Trans. Inf. Theory, vol. 54, no. 2, pp. 831–836, 2008.
[2] N. Bitouze, A. Graell i Amat and E. Rosnes, "Error correcting coding for a nonsymmetric ternary channel," IEEE Trans. Comm., vol. 56, no. 11, pp. 5715–5729, 2010.
[3] Y. Ben-Hur and Y. Cassuto, "Coding on dual-parameter barrier channels beyond worst-case correction," IEEE Global Comm. Conf., 2021.