Construction and Decoding of Codes over the Dual-Parameter Barrier Error Model

Yuval Ben-Hur and Yuval Cassuto
The Viterbi Faculty of Electrical and Computer Engineering,
Technion–Israel Institute of Technology, Haifa, Israel
Emails: {yuvalbh@campus, ycassuto@ee}.technion.ac.il

***Abstract*—Barrier-error channels have been suggested as a model for non-binary channels that are milder than symmetric-error channels. Such channels are motivated by practical applications in data storage and communications. The barrier-error model allows errors only to (downward) and from (upward) a specific symbol within the alphabet. We study a generalization of prior barrier-error models that considers different numbers of downward and upward errors. The results of this paper include a sufficient condition for error correction, code-size upper and lower bounds, a code construction decomposing to just two constituent symmetric-error codes (prior ones used many), and a decoding algorithm that decodes the two codes jointly. The decoder is shown empirically to achieve better block error rates when compared to previous algorithms.**

I. Introduction

Despite the soaring demand for data rate and efficiency, many information systems still store, process and communicate data represented by the binary alphabet. Moving to larger representation alphabets – spanning $Q > 2$ symbols per channel use – is tempting but often challenging. Recent such attempts in practice range from IoT biosensors [1] to novel memory technologies such as resistive processing memories [2] and magnetic storage media [3].

Easier transitions from binary to non-binary can occur when the resulting error models are "milder" than the canonical model of Q-ary symmetric errors. With such error models, the coding rates can be higher, and the decoding complexities can be lower. Such a ternary ($Q = 3$) channel was proposed in [4], prescribing that errors can happen between the center symbol and one of the outer symbols, but not between the outer symbols themselves. The motivation for this channel comes from non-volatile memories and wide-band communication devices [5] whose dominant errors are only between neighboring symbols. For general Q, this error type was termed *barrier errors* [6], where the generalization is motivated by channels where all transitions from the $Q - 1$ non-barrier symbols are attracted into the barrier symbol, while transitions from the barrier symbol are symmetric. This prior work indeed showed that designing codes for the barrier channel is beneficial for both rate and complexity.

In a Q-ary *barrier channel*, we designate one alphabet symbol (here assumed to be 0) as the *barrier symbol*, and allow errors only to the barrier symbol (downward errors) and from the barrier symbol (upward errors). Downward-only barrier errors were considered in [7] that contributed code-size bounds and a single-error correcting construction. A channel allowing barrier errors in both directions, under a single parameter, was studied in [8] (and its extension [4]). This work addressed both the probabilistic version of the channel (under a transition parameter p), and its guaranteed-correction version (under an error-count parameter t). The work of [9] derived code-size bounds on a "proxy" error model defined in [4]. The work of [6] generalized the model to allow two transition parameters in the probabilistic model: p for downward errors and q for upward errors. In this present paper, we generalize the guaranteed-decoding error model to two parameters: up to t_d downward errors and up to t_u upward errors.

The dual parameter (t_d, t_u) error model is an important variant of the barrier-error model, because downward and upward errors are different. In particular, the results of [6] on the probabilistic channel show that increasing p (downward probability) is much more detrimental to the channel capacity than q (upward probability). That means that we can potentially gain a lot by differentiating between the two error directions. Toward realizing this potential, the paper presents various results for the (t_d, t_u) error model: sufficient conditions for correction (Section II), code-size upper and lower bounds (Section III), a new code construction based on a pair of constituent codes for symmetric errors (Section IV), and a decoding algorithm that decodes the two codes jointly (Section V), whose correction performance beyond the guaranteed-decoding bound is demonstrated empirically.

The main improvements offered by this paper's contributions relative to prior work are as follows: 1) The code-size lower bounds prove the existence of codes that (for small t_d) are significantly larger than the Gilbert-Varshamov (GV) bound for symmetric errors, 2) the code construction improves over that in [4] by using only a single erasure-correcting code, instead of one code per each codeword weight, and 3) the proposed joint-decoding algorithm (enabled by the new code construction) significantly improves the correction performance beyond all prior decoders.

II. Error Correction over the Dual-Parameter Barrier Error Model

The Q-ary dual-parameter barrier error model, denoted $W_Q(t_d, t_u)$, is defined using two (non-negative integer) parameters, t_d and t_u, describing the maximal number of downward and upward barrier errors, respectively. We denote the set of integers by $\mathbb{Z}$ and the subset of non-negative integers smaller than Q by $\mathbb{Z}_Q$. The notation $-\mathbb{Z}_Q$ denotes the set of corresponding negative integers, including 0. We further denote by $[n]$ the set of integers $\{1, 2, \ldots, n\}$. Formally, barrier errors are defined as follows.

Definition 1: Given $n \in \mathbb{N}$ and a codeword $\mathbf{c} = (c_1, \ldots, c_n) \in \mathbb{Z}_Q^n$, a vector $\mathbf{e} = (e_1, \ldots, e_n) \in \{-\mathbb{Z}_Q \cup \mathbb{Z}_Q\}^n$ is called a barrier error vector of $\mathbf{c}$ if for every index $i \in [n]$ such that $c_i \neq 0$ either $e_i = -c_i$ or $e_i = 0$. If $c_i = 0$, e_i can be an arbitrary element of $\mathbb{Z}_Q$.

Given a codeword $\mathbf{c} \in \mathbb{Z}_Q^n$ transmitted through $W_Q(t_d, t_u)$, the output $\mathbf{r}$ equals $\mathbf{c} + \mathbf{e}$ (addition over the integers), where $\mathbf{e}$ is a (t_d, t_u)-barrier error, defined as follows.

Definition 2: Given a codeword $\mathbf{c} \in \mathbb{Z}_Q^n$, a barrier error vector $\mathbf{e}$ (as defined in Definition 1) is a (t_d, t_u) barrier error if $|\{i \in [n] : e_i < 0\}| \leq t_d$ and $|\{i \in [n] : e_i > 0\}| \leq t_u$.

A code $\mathcal{C} \subseteq \mathbb{Z}_Q^n$ is said to be a (t_d, t_u) barrier error correcting code if it corrects any (t_d, t_u) barrier error.

Throughout the paper we assume the standard definitions of Hamming weight $w_H(\mathbf{x}) \triangleq |\{i \in [n] : x_i \neq 0\}|$ and Hamming distance $d_H(\mathbf{x}, \mathbf{z}) = |\{i \in [n] : x_i \neq z_i\}|$ for $\mathbf{x} = (x_1, \ldots, x_n), \mathbf{z} = (z_1, \ldots, z_n) \in \mathbb{Z}_Q^n$. We also use the following indicator function.

Definition 3: Let $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{Z}_Q^n$. The **indicator mapping** of $\mathbf{x}$ is defined as $\imath(\mathbf{x}) \triangleq (\imath(x_1), \ldots, \imath(x_n))$ where

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

Toward characterizing the correction capability of a given code, we split the calculation of the Hamming distance into two complementary additive elements using the indicator distance defined next.

Definition 4: For any $\mathbf{x} = (x_1, \ldots, x_n), \mathbf{z} = (z_1, \ldots, z_n) \in \mathbb{Z}_Q^n$, define the indicator distance as

$$d_\imath(\mathbf{x}, \mathbf{z}) = d_H(\imath(\mathbf{x}), \imath(\mathbf{z})). \tag{2}$$

Correspondingly, we define the residual distance as its complement to the full Hamming distance:

Definition 5: For any $\mathbf{x} = (x_1, \ldots, x_n), \mathbf{z} = (z_1, \ldots, z_n) \in \mathbb{Z}_Q^n$, define the residual distance as

$$d_{\bar{\imath}}(\mathbf{x}, \mathbf{z}) = d_H(\mathbf{x}, \mathbf{z}) - d_\imath(\mathbf{x}, \mathbf{z}). \tag{3}$$

We now prove the following sufficient condition for the guaranteed correction capability of (t_d, t_u) barrier errors.

Proposition 1: Let $\mathcal{C} \subseteq \mathbb{Z}_Q^n$ be a code such that for any two codewords $\mathbf{x}, \mathbf{z} \in \mathcal{C}$, either $d_H(\mathbf{x}, \mathbf{z}) \geq 2(t_u + t_d) + 1$ or $d_{\bar{\imath}}(\mathbf{x}, \mathbf{z}) \geq t_d + 1$ (or both). Then, $\mathcal{C}$ is a (t_d, t_u) barrier error correcting code.

Proof: Let $\mathbf{c}$ be a codeword in a code $\mathcal{C}$ that satisfies the aforementioned condition. Let $\mathbf{r} \in \mathbb{Z}_Q^n$ be the output of $W_Q(t_d, t_u)$ for the input codeword $\mathbf{c} \in \mathcal{C}$, i.e., there exists a (t_d, t_u) barrier error $\mathbf{e}$ such that $\mathbf{r} = \mathbf{c} + \mathbf{e}$. Assume there exists another codeword, $\mathbf{c}' \in \mathcal{C}$, such that $\mathbf{r} = \mathbf{c}' + \mathbf{e}'$ for some (t_d, t_u) barrier error $\mathbf{e}'$, and $d_H(\mathbf{c}', \mathbf{r}) \leq d_H(\mathbf{c}, \mathbf{r})$. We now consider two cases. If $d_H(\mathbf{c}, \mathbf{c}') \geq 2(t_u + t_d) + 1$, we immediately get a contradiction, since

$$d_H(\mathbf{c}, \mathbf{c}') \leq d_H(\mathbf{c}, \mathbf{r}) + d_H(\mathbf{c}', \mathbf{r}) \leq 2 d_H(\mathbf{c}, \mathbf{r}) \leq 2(t_u + t_d).$$

Therefore, $d_{\bar{\imath}}(\mathbf{c}, \mathbf{c}') \geq t_d + 1$. To show that this also implies a contradiction, define the following index sets sizes

$$\begin{aligned} M_1 &= |\{j \in [n] : r_j = c_j = 0 \text{ and } c'_j \neq 0\}| \\ M_2 &= |\{j \in [n] : r_j = 0 \text{ and } 0 \neq c_j \neq c'_j \neq 0\}|. \end{aligned} \tag{4}$$

Observe that by the error-model definition, $M_1 + M_2 \leq t_d$. Consequently, $d_{\bar{\imath}}(\mathbf{c}, \mathbf{c}') = M_2 \leq M_1 + M_2 \leq t_d$, where the second inequality follows from its preceding observation. This leads to a contradiction. ■

Remark 1: The trivial sufficient condition for (t_d, t_u) barrier error correction, stating that $d_H(\mathbf{c}, \mathbf{c}') \geq 2(t_u + t_d) + 1$ for any $\mathbf{c}, \mathbf{c}' \in \mathcal{C}$, is improved by Proposition 1 which guarantees correction with a weaker condition on the code, namely allowing $\mathcal{C}$ to contain codewords $\mathbf{c}, \mathbf{c}' \in \mathcal{C}$ with $d_H(\mathbf{c}, \mathbf{c}') \leq 2(t_u + t_d)$, as long as $d_{\bar{\imath}}(\mathbf{c}, \mathbf{c}') \geq t_d + 1$.

III. Bounds on Maximum Code Sizes

We now derive upper and lower bounds on the size of the largest (t_d, t_u) barrier-error-correcting code. The analogue of a Hamming ball for the barrier channel is the (t_d, t_u) ball defined next.

Definition 6: Define the (t_d, t_u) ball around a word $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{Z}_Q^n$ as

$$\mathcal{E}_{(t_d, t_u)}(\mathbf{x}) = \{\mathbf{y} \in \mathbb{Z}_Q^n : \mathbf{y} = \mathbf{x} + \mathbf{e}\}, \tag{5}$$

where $\mathbf{e} = (e_1, \ldots, e_n)$ is a (t_d, t_u)-barrier error.

Note that $\mathcal{E}_{(t_d, t_u)}(\mathbf{x})$ defines the set of possible channel outputs when $\mathbf{x}$ is transmitted through $W_Q(t_d, t_u)$. The number of words in a (t_d, t_u) ball is a key ingredient in the derivation of code-size bounds.

Proposition 2: Let $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{Z}_Q^n$ and $t_d, t_u \in \mathbb{N}$. The size of the set $\mathcal{E}_{(t_d, t_u)}(\mathbf{x})$ is given by

$$|\mathcal{E}_{(t_d, t_u)}(\mathbf{x})| = \sum_{\tau_d = 0}^{\alpha_{\mathbf{x}}} \binom{\omega_{\mathbf{x}}}{\tau_d} \times \sum_{\tau_u = 0}^{\beta_{\mathbf{x}}} \binom{n - \omega_{\mathbf{x}}}{\tau_u} (Q - 1)^{\tau_u}, \tag{6}$$

where $\omega_{\mathbf{x}} \triangleq w_H(\mathbf{x})$, $\alpha_{\mathbf{x}} \triangleq \min\{\omega_{\mathbf{x}}, t_d\}$ and $\beta_{\mathbf{x}} \triangleq \min\{n - \omega_{\mathbf{x}}, t_u\}$.

Proof: Let $\omega_{\mathbf{x}} \triangleq w_H(\mathbf{x})$ be the number of non-zero elements in $\mathbf{x}$. Counting the size of the set can be partitioned to the product of two subsets of indices:

1) $\{i : x_i = 0\}$: Each zero element has $Q - 1$ possible non-zero assignments, and given $\tau_u \leq \min\{t_u, n - \omega_{\mathbf{x}}\}$ such transitions, there are $\binom{n - \omega_{\mathbf{x}}}{\tau_u}$ possible index subsets. Summing over all possible words with τ_u transitions yields the second sum in the product in (6).
2) $\{i : x_i \neq 0\}$: Each non-zero element has one possible erroneous assignment (transition to 0), and given $\tau_d \leq \min\{t_d, \omega_{\mathbf{x}}\}$ such transitions, there are $\binom{\omega_{\mathbf{x}}}{\tau_d}$ possible index subsets. Summing over all possible τ_d assignments yields the first sum in the product in (6).

Since the two sets of indices are disjoint, the product of sizes yields the size of $\mathcal{E}_{(t_d,t_u)}(\mathbf{x})$. ∎

Note that the size of the set $\mathcal{E}_{(t_d,t_u)}(\mathbf{x})$ depends on $\mathbf{x}$, or more precisely, on $w_H(\mathbf{x})$. We therefore rename it $|\mathcal{E}_{(t_d,t_u)}(\omega)|$, where $0 \le \omega \le n$. An upper bound using these ball volumes will depend on the code's *weight support*, defined next.

Definition 7: Let $\mathcal{C} \subseteq \mathbb{Z}_Q^n$ be a code of length n. Define the weight support of $\mathcal{C}$, $\Omega(\mathcal{C}) \subseteq \{0, 1 \ldots, n\}$, as the set of Hamming weights of codewords in $\mathcal{C}$.

The following proposition gives an upper bound on the size of any (t_d, t_u) barrier-error-correcting code.

Proposition 3: Let $t_d, t_u \in \mathbb{N}$ and let $\mathcal{C} \subseteq \mathbb{Z}_Q^n$ be a (t_d, t_u) barrier error correcting code with weight support Ω. Then,

$$|\mathcal{C}| \le \frac{Q^n}{\min_{\omega\in\Omega} |\mathcal{E}_{(t_d,t_u)}(\omega)|}. \tag{7}$$

Proof: Let $\mathcal{C}$ be a (t_d, t_u) barrier error correcting code with a given weight support Ω. For every two codewords $\mathbf{c}, \mathbf{c}' \in \mathcal{C}$, the corresponding (t_d, t_u) balls, $\mathcal{E}_{(t_d,t_u)}(\mathbf{c})$ and $\mathcal{E}_{(t_d,t_u)}(\mathbf{c}')$, do not intersect (otherwise, there exists a (t_d, t_u) barrier error that cannot be corrected). Consequently,

$$\begin{aligned} Q^n = |\mathbb{Z}_Q^n| &\ge | \cup_{\mathbf{c}\in\mathcal{C}} \mathcal{E}_{(t_d,t_u)}(\mathbf{c})| = \sum_{\mathbf{c}\in\mathcal{C}} |\mathcal{E}_{(t_d,t_u)}(\mathbf{c})| \\ &\ge |\mathcal{C}| \min_{\mathbf{c}\in\mathcal{C}} |\mathcal{E}_{(t_d,t_u)}(\mathbf{c})| = |\mathcal{C}| \min_{\omega\in\Omega} |\mathcal{E}_{(t_d,t_u)}(\omega)|. \end{aligned} \tag{8}$$

∎

Toward deriving a *lower bound* on the maximal size of a (t_d, t_u) barrier error correcting code, we now define another type of "ball" in $\mathbb{Z}_Q^n$ and calculate its volume. In symmetric error correction, the balls used for lower bounds (Gilbert Varshamov (GV) [10], [11]) are the same as the balls for upper bounds, only with the radius doubled. In the barrier error model there is no such simple doubling that we can use. Instead, we define the ball using the sufficient condition of Proposition 1.

Definition 8: For a word $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{Z}_Q^n$, define $\mathcal{B}_{(t_d,t_u)}(\mathbf{x})$ to be the set consisting of all words $\mathbf{y} \in \mathbb{Z}_Q^n$ such that $d_H(\mathbf{x}, \mathbf{y}) \le 2(t_u + t_d)$ *and* $d_{\bar{1}}(\mathbf{x}, \mathbf{y}) \le t_d$.

Proposition 4: Let $\mathbf{c} = (c_1, \ldots, c_n) \in \mathbb{Z}_Q^n$, $t_d, t_u \in \mathbb{N}$ and denote $\omega_\mathbf{c} \triangleq w_H(\mathbf{c})$. The size of the set $\mathcal{B}_{(t_d,t_u)}(\mathbf{c})$ is given by

$$\begin{aligned} |\mathcal{B}_{(t_d,t_u)}(\mathbf{c})| &= \sum_{\tau_u=0}^{\alpha'_\mathbf{c}} \binom{n-\omega_\mathbf{c}}{\tau_u} (Q-1)^{\tau_u} \\ &\times \sum_{\tau_d=0}^{\beta'_\mathbf{c}} \binom{\omega_\mathbf{c}}{\tau_d} \times \sum_{\tau_r=0}^{\gamma'_\mathbf{c}} \binom{\omega_\mathbf{c}-\tau_d}{\tau_r} (Q-2)^{\tau_r}, \end{aligned} \tag{9}$$

where $\alpha'_\mathbf{c} \triangleq \min\{n - \omega_\mathbf{c}, 2(t_u + t_d)\}$, $\beta_{\mathbf{c}'} \triangleq \min\{\omega_\mathbf{c}, 2(t_u + t_d) - \tau_u\}$ and $\gamma'_\mathbf{c} \triangleq \min\{\omega_\mathbf{c} - \tau_d, t_d, 2(t_d + t_u) - \tau_d - \tau_u\}$.

Proof: To satisfy the first condition, $\tau_u \le 2(t_u+t_d)$ zero indices change to non-zeros, and $\tau_d \le 2(t_u+t_d)-\tau_u$ non-zero indices change to zeros. These give the first two sums in the product (9), which also consider the respective bounds $n-\omega_\mathbf{c}$ and $\omega_\mathbf{c}$ on τ_u and τ_d. To satisfy the second condition, $\tau_r \le t_d$ non-zero indices (not selected by the τ_d above) change to a different non-zero symbol. This adds the third sum in (9). ∎

Note that $|\mathcal{B}_{(t_d,t_u)}(\mathbf{c})|$ depends only on the weight $\omega_\mathbf{c}$, hence, we denote it $|\mathcal{B}_{(t_d,t_u)}(\omega_\mathbf{c})|$. Since $|\mathcal{B}_{(t_d,t_u)}(\mathbf{c})|$ contains all the words violating the sufficient condition of Proposition 1 with respect to $\mathbf{c}$, one can greedily select $\mathbf{c}$ to be in the code, and removing $\mathcal{B}_{(t_d,t_u)}(\mathbf{c})$ from the list of candidate codewords guarantees ending up with a (t_d, t_u) barrier code. This also gives a lower bound on the maximal code size, similar to the GV bound.

Proposition 5: Let $t_d, t_u \in \mathbb{N}$ and let $\mathcal{C} \subseteq \mathbb{Z}_Q^n$ be the largest possible (t_d, t_u) barrier-error-correcting code with weight support Ω. Then,

$$|\mathcal{C}| \ge \frac{\sum_{\omega\in\Omega} \binom{n}{\omega}(Q-1)^\omega}{\max_{\omega\in\Omega} |\mathcal{B}_{(t_d,t_u)}(\omega)|}. \tag{10}$$

Proof: To prove the inequality, we use the following chain of equalities and inequalities

$$\begin{aligned} \sum_{\omega\in\Omega} \binom{n}{\omega}(Q-1)^\omega &= |\{\mathbf{x} \in \mathbb{Z}_Q^n : w_H(\mathbf{x}) \in \Omega(\mathcal{C})| \\ &= | \cup_{\mathbf{c}\in\mathcal{C}} \mathcal{B}_{(t_d,t_u)}(\mathbf{c})| \le \sum_{\mathbf{c}\in\mathcal{C}} |\mathcal{B}_{(t_d,t_u)}(\mathbf{c})| \\ &\le |\mathcal{C}| \max_{\mathbf{c}\in\mathcal{C}} |\mathcal{B}_{(t_d,t_u)}(\mathbf{c})| = |\mathcal{C}| \max_{\omega\in\Omega} |\mathcal{B}_{(t_d,t_u)}(\omega)|, \end{aligned} \tag{11}$$

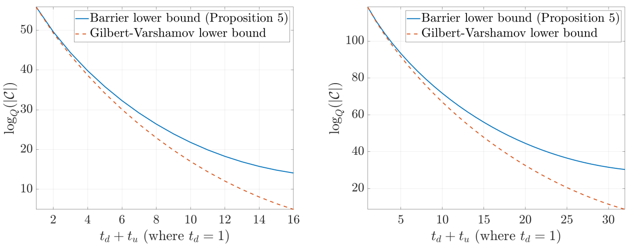
where the first equality is the standard counting of weight-ω vectors, the second equality is from the fact that the greedy codeword selection stops only when all words within the weight support are contained in a codeword ball, the first inequality is because sum of sizes is never smaller than the union size, and the second inequality bounds the sum of sizes by a product with the largest size. ∎

To effectively use the bounds of Propositions 3 and 5, we need to determine Ω wisely. We focus on the lower bound, because in that case guessing a "good" Ω can prove the existence of a large code with that weight support. We observe that the size of $\mathcal{B}_{(t_d,t_u)}(0)$ is $\sum_{\tau_u=0}^{2(t_u+t_d)} \binom{n}{\tau_u}(Q-1)^{\tau_u}$, which is exactly the size of a Q-ary GV ball for correcting $t_d + t_u$ *symmetric* errors, which is a stronger error model than ours. Therefore, Proposition 5 can potentially improve over known lower bounds (for symmetric errors) only if $0 \notin \Omega$. Luckily, it is possible to get strictly better lower bounds when we bound ω from below, as we show in the following example.

Example 1: In the following, for $Q = 3$ we fix $t_d = 1$ and set $\Omega = \{\omega_{\min}, \ldots, n\}$, where $\omega_{\min}$ is optimized to achieve the largest lower bound for each $t_u \in \{0, \ldots, n/4\}$ separately. The results are shown in Figure 1 for $n = 64$ (left) and $n = 128$ (right). Each plot compares the value of (10) (solid) to the GV bound for t_d+t_u symmetric errors (dashed). It can be observed that the proposed lower bound consistently improves over the GV bound, especially for $t_u \gg t_d$.

IV. A Construction for Correcting (t_d, t_u)-Barrier Errors

In [4], a code construction is proposed for the single-parameter (t) barrier-error model, which allows any combi-



(a) $n = 64$ (b) $n = 128$

Figure 1. Comparison of the lower bound in (10) with the GV lower bound, for $Q = 3$ and $t_d = 1$, as a function of the total number of correctable errors ($t_d + t_u$). Each point is obtained by setting the minimum weight in Ω to maximize the value of the bound.

nation $t_d + t_u \leq t$. A code of this construction is decomposed to one binary code of length n and minimum Hamming distance $d_1 = 2t + 1$, and $n + 1$ additional $(Q-1)$-ary codes (one for each codeword weight of the first code), each with minimum Hamming distance $d_2 = t + 1$. Firstly, this construction does not consider the separate bounds t_d, t_u, and thus it introduces unessential redundancy. Using the sufficient condition of Proposition 1, one can change d_2 to $t_d + 1$, and guarantee (t_d, t_u) correction with lower redundancy cost. Secondly, the need for a multitude of codes, each with a different length, is cumbersome for the construction, and limiting for decoding the codes, since it may not be clear to the decoder which of the $n + 1$ codes it needs to decode, and on which index subset. We solve these two issues with our new construction, and in the next section show its benefits with respect to decoding.

For the specified parameters (t_d, t_u), the construction decomposes to the following *two* codes, both with length n.

Definition 9 (Indicator code): Given $t_d, t_u \in \mathbb{N}$ and $n \in \mathbb{N}$, define the indicator code as a binary code with length n and minimum Hamming distance $2(t_d + t_u) + 1$.

Definition 10 (Residual code): Given $t_d \in \mathbb{N}$ and $n \in \mathbb{N}$, define the residual code as a code over $\in \mathbb{Z}_{Q-1}^n$ with length n and minimum Hamming distance $t_d + 1$.

Construction 1: Let Θ be an indicator code for parameters t_d, t_u and length n; let Λ be a residual code for parameter t_d and length n. A word $\mathbf{c} = (c_1, \ldots, c_n) \in \mathbb{Z}_Q^n$ is a codeword in the composed code $\mathcal{C} = \Theta \otimes \Lambda$ if the following conditions hold:

1) There exists $\theta = (\theta_1, \ldots, \theta_n) \in \Theta$ such that $\imath(\mathbf{c}) = \theta$.
2) For this $\theta \in \Theta$, there exists $\lambda = (\lambda_1, \ldots, \lambda_n) \in \Lambda$ such that for every $j \in [n]$: $c_j = \lambda_j + 1$ if $\theta_j \neq 0$ and $c_j = \lambda_j = 0$ if $\theta_j = 0$.

To prove that $\mathcal{C}$ is a (t_d, t_u) barrier-error-correcting code, we show that it satisfies the sufficient condition of Proposition 1.

Theorem 1: Given $t_d, t_u \in \mathbb{N}$ and $n \in \mathbb{N}$, let $\mathcal{C} \subseteq \mathbb{Z}_Q^n$ be a code constructed by Construction 1. Then, $\mathcal{C}$ is a (t_d, t_u) barrier-error-correcting code.

Proof: Let $\mathbf{c}, \mathbf{c}' \in \mathcal{C}$ be a pair of arbitrary different codewords ($\mathbf{c} \neq \mathbf{c}'$). Let $\theta = \imath(\mathbf{c})$ and $\theta' = \imath(\mathbf{c}')$. We distinguish two cases. If $\theta \neq \theta'$, then

$$\begin{aligned} d_H(\mathbf{c}, \mathbf{c}') \geq d_\imath(\mathbf{c}, \mathbf{c}') &= d_H(\imath(\mathbf{c}), \imath(\mathbf{c}')) \\ &= d_H(\theta, \theta') \geq 2(t_d + t_u) + 1, \end{aligned}$$

where the last inequality is from the specification of Θ using Definition 9.

Otherwise $\theta = \theta'$, in which case $d_\imath(\mathbf{c}, \mathbf{c}') = d_H(\theta, \theta') = 0$, which gives

$$d_{\bar{\imath}}(\mathbf{c}, \mathbf{c}') \triangleq d_H(\mathbf{c}, \mathbf{c}') - d_\imath(\mathbf{c}, \mathbf{c}') = d_H(\mathbf{c}, \mathbf{c}'). \tag{12}$$

We prove that $d_{\bar{\imath}}(\mathbf{c}, \mathbf{c}') \geq t_d + 1$ by chaining the above with the inequalities

$$d_H(\mathbf{c}, \mathbf{c}') \geq d_H(\lambda, \lambda') \geq t_d + 1, \tag{13}$$

where the first follows from condition 2 in the construction, and the second from the fact that $\lambda \neq \lambda'$ (otherwise, $\mathbf{c} = \mathbf{c}'$) and from the specification of Λ using Definition 10. In both cases we proved that the sufficient condition of Proposition 1 is satisfied, which completes the proof. ■

When Λ is a linear code, encoding of Construction 1 can be done by first encoding Θ, then taking from the parity-check matrix of Λ only the columns corresponding to non-zero indices, denoted $\mathbf{H}_\Lambda(\{j : \theta_j \neq 0\})$, and outputing a λ that has zeros in $\{j : \theta_j = 0\}$, and whose sub-vector with indices in $\{j : \theta_j \neq 0\}$ is in the null space of $\mathbf{H}_\Lambda(\{j : \theta_j \neq 0\})$. For decoding, the use of a single residual code (instead of a multitude as in [4]) allows to decode Λ independent of the success of Θ's decoder to find the correct θ.

V. Joint decoding of the constituent codes

To decode Construction 1, we now propose an algorithm that decodes the indicator and residual codes *jointly*, such that the decoders of Θ and Λ help each other, and not just Θ helping Λ as in prior work. In [4] a bounded-distance decoder for Θ is invoked once, and this output is used to decode Λ, with no option to succeed if the initial output was not correct. An improved decoder was proposed in [6], in which a *list decoder* for Θ is employed, and the decoder then selects from the list the most likely codeword consistent with Λ. In both cases the operation of the decoders is sequential $\Theta \to \Lambda$; in the first the cost of that is poor performance beyond the code guarantees, and in the second the cost is high computational complexity of list decoding. The main idea of the proposed decoder is to use information from Λ's decoder to improve the decoding outcome of Θ's decoder. In particular, this information is chosen to be the *inconsistency score* of each index in $[n]$, which is the number of violated parity-check equations of $\mathbf{H}_\Lambda$ that the index appears in.

The decoding algorithm is given formally in Algorithms 1 and 2 (which is specified here for $Q = 3$). The vector operation $\cdot$ in the algorithms represents element-wise multiplication over the integers. Given a channel output $\mathbf{r}$, we first decode Θ to get an indicator candidate codeword $\bar{\theta}$. We then use Algorithm 2 to order the indices of $[n]$ based on their Λ-inconsistency with respect to $\bar{\theta}$. For an iteration-count limit of $M = 2^m$, we take a set S of the m indices with the highest inconsistencies, and use this set to generate *bit-flipping subsets*. Each subset is used to obtain a codeword $\hat{\theta}$ by the decoder of Θ, this codeword is then decoded by Λ, and the combined Q-ary codeword is returned if consistent. This iteration starts with the empty subset (equivalent to decoding Λ with $\bar{\theta}$), and proceeds in increasing subset sizes; within each size, subsets are ordered by mapping each to a vector of increasing *positions* in S, and ordering these vectors lexicographically. This decoding approach mimics *generalized minimum-distance decoding (GMD)* [12] and *ordered statistics decoding (OSD)* [13], only that the index ordering is done based on information from Λ and not from the channel. Note that Algorithm 1 actually generalizes the decoder from [4] when $M = 1$ and bounded-distance decoders are used for both constituent codes.

Algorithm 1 Joint decoding of $\mathcal{C} = \Theta \otimes \Lambda$

Input: $\mathbf{r} \in \mathbb{Z}_Q^n$: channel output, $M = 2^m$: max # iterations
Output: $\hat{\mathbf{c}} \in \mathbb{Z}_Q^n$: decoded codeword
$\bar{\theta} \leftarrow$ decode $\imath(\mathbf{r})$ over Θ
$\pi \leftarrow$ Inconsistency-ordering$(\mathbf{r}, \bar{\theta}, \mathbf{H}_\Lambda)$
$S \leftarrow \pi(1:m)$ ▷ highest-inconsistency indices
for $\sigma = \{\}, \ldots, S$ **do** ▷ order subsets $\sigma \subseteq S$ by size, higher inconsistencies first in each size
  set $\mathbf{f}$ to be a vector with ones in the indices of σ, and zeros elsewhere
  $\hat{\theta} \leftarrow$ decode $\imath(\mathbf{r}) \oplus \mathbf{f}$ over Θ
  $\hat{\lambda} \leftarrow$ decode $\hat{\theta} \cdot (\mathbf{r} - \mathbf{1})$ over Λ
  if $\mathbf{H}_\Lambda \hat{\lambda}^\top = \mathbf{0}^\top$ **then** ▷ $\hat{\lambda}$ is consistent
    return $\hat{\mathbf{c}} = \hat{\theta} \cdot (\mathbf{1} + \hat{\lambda})$
  end if
end for
return *decoding failure*

Algorithm 2 Inconsistency-ordering of $[n]$

Input: $\mathbf{r} \in \mathbb{Z}_Q^n$: channel output, $\hat{\theta}$: candidate indicator codeword, $\mathbf{H}$: $\rho \times n$ residual-code parity-check matrix
Output: π: sorting of $[n]$ in non-increasing inconsistency
Extract residual parity-check equations with unknown consistency

$$\mathcal{E} \leftarrow \{i \in [\rho] : \text{exists } j \in [n], r_j = 0 \wedge \hat{\theta}_j = 1 \wedge H_{i,j} = 1\}$$

Extract inconsistent residual parity-check equations

$$\mathcal{S} \leftarrow \left\{ i \in [\rho] \setminus \mathcal{E} : \mathbf{H}_{i,:}(\hat{\theta} \cdot \mathbf{r})^\top \neq 0 \right\}$$

Calculate "inconsistency score" for each index $j \in [n]$ as

$$w_j = \sum_{i \in \mathcal{S}} H_{i,j}$$

$\pi \leftarrow$ sort $[n]$ in non-increasing w_j
return π

To demonstrate the new decoder's advantage, we simulated a stochastic ternary ($Q = 3$) barrier channel, parameterized by p: the probability of downward error ($\neq 0 \to 0$ transition), and q: the probability of upward error ($0 \to\neq 0$ transition; $q/2$ to each of the symbols 1 and 2). We used binary $(2, 7)$ Reed–Muller (RM) as the indicator code and decoded it using the recursive projection-aggregation decoder [14]. For the residual codes, we used rate $1/2$ binary Low-Density Parity-Check (LDPC) codes that were decoded using a standard peeling erasure decoder [15]. This configuration resulted in a combined code rate of 0.16 symbols per channel use. The channel was simulated with asymmetric transition probabilities: $p = 0.04$ for downward errors and $q = 0.4$ for upward errors.

Fig. 2 plots the decoding block-error probability as a function of the number of errors beyond the indicator code's guaranteed correction parameter $t_{RM} = 15$. It clearly shows the advantage of the proposed decoder over the previously proposed decoders of [4] and [6]. The decoder outperforms [6], even though the latter uses a computationally-heavy list decoder with list-size 16.

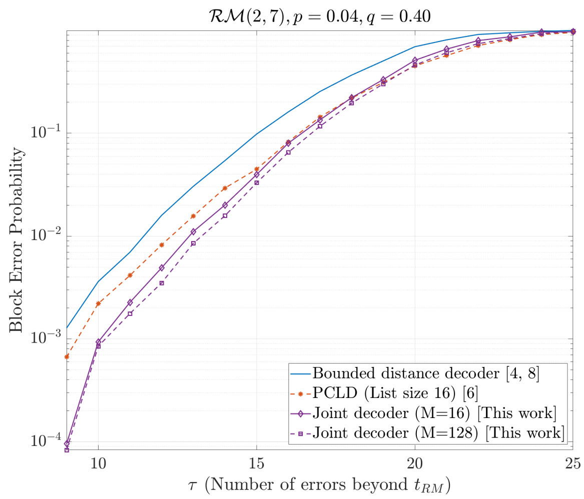


Figure 2. Block error rates of $\mathcal{RM}(2, 7)$ indicator codes in conjunction with rate $\sim 1/2$ LDPC residual codes simulated over the ternary Barrier channel with transition probabilities $p = 0.04, q = 0.4$.

VI. Conclusion

This paper presents coding-theoretic results for the previously unaddressed model of barrier errors with separate parameters for downward and upward errors. The results come in three different levels: code-size existential bounds, a code construction with simpler decomposition to symmetric error-correcting codes, and a joint decoding algorithm. For future work, one may explore the gap between the size lower bound and practical-code sizes, as well as devising new decoding algorithms using other modes of cooperation between Θ and Λ.

VII. Acknowledgement

This work was supported in part by the Israel Science Foundation under grant number 2525/19.

REFERENCES

[1] X. Tang and W. Tang, "A 151nw second-order ternary delta modulator for ecg slope variation measurement with baseline wandering resilience," in *2020 IEEE Custom Integrated Circuits Conference (CICC)*. IEEE, 2020, pp. 1–4.

[2] L. Luo, Z. Dong, X. Hu, L. Wang, and S. Duan, "Mtl: Memristor ternary logic design," *International Journal of Bifurcation and Chaos*, vol. 30, no. 15, pp. 205–222, 2020.

[3] Y. Telepinsky, V. Mor, M. Schultz, Y.-M. Hung, A. D. Kent, and L. Klein, "Towards a six-state magnetic memory element," *Applied Physics Letters*, vol. 108, no. 18, p. 182401, 2016.

[4] N. Bitouzé, A. G. i Amat, and E. Rosnes, "Error correcting coding for a nonsymmetric ternary channel," *IEEE Transactions on Information Theory*, vol. 56, no. 11, pp. 5715–5729, 2010.

[5] S. Vladimirov and O. Kognovitsky, "Wideband data signals with direct ternary maximum length sequence spread spectrum and their characteristics," *Proceedings of Telecommunication Universities*, vol. 3, pp. 28–36, 2017.

[6] Y. Ben-Hur and Y. Cassuto, "Coding on dual-parameter barrier channels beyond worst-case correction," in *2021 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2021, pp. 01–06.

[7] R. Ahlswede and H. Aydinian, "Error control codes for parallel asymmetric channels," *IEEE Transactions on Information Theory*, vol. 54, no. 2, pp. 831–836, 2008.

[8] N. Bitouzé and A. G. i Amat, "Coding for a non-symmetric ternary channel," in *2009 Information Theory and Applications Workshop*. IEEE, 2009, pp. 113–118.

[9] L. Tolhuizen, "Bounds for codes for a non-symmetric ternary channel," *arXiv preprint arXiv:1004.1511*, 2010.

[10] R. R. Varshamov, "Estimate of the number of signals in error correcting codes," *Dokl. Acad. Nauk SSSR*, vol. 117, pp. 739–754, 1957.

[11] E. N. Gilbert, "A comparison of signalling alphabets," *The Bell System Technical Journal*, vol. 31, no. 3, pp. 504–522, 1952.

[12] G. Forney, "Generalized minimum distance decoding," *IEEE Transactions on Information Theory*, vol. 12, no. 2, pp. 125–131, 1966.

[13] M. P. Fossorier and S. Lin, "Soft-decision decoding of linear block codes based on ordered statistics," *IEEE Transactions on Information Theory*, vol. 41, no. 5, pp. 1379–1396, 1995.

[14] M. Ye and E. Abbe, "Recursive projection-aggregation decoding of reed-muller codes," *IEEE Transactions on Information Theory*, vol. 66, no. 8, pp. 4948–4965, 2020.

[15] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, "Efficient erasure correcting codes," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 569–584, 2001.