Nonuniform Codes for Correcting Asymmetric Errors in Data Storage

Hongchao Zhou, Anxiao Jiang, *Senior Member, IEEE*, and Jehoshua Bruck, *Fellow, IEEE*

***Abstract*—The construction of asymmetric error-correcting codes is a topic that was studied extensively, however; the existing approach for code construction assumes that every codeword should tolerate t asymmetric errors. Our main observation is that in contrast to symmetric errors, asymmetric errors are content dependent. For example, in Z-channels, the all-1 codeword is prone to have more errors than the all-0 codeword. This motivates us to develop nonuniform codes whose codewords can tolerate different numbers of asymmetric errors depending on their Hamming weights. The idea in a nonuniform codes' construction is to augment the redundancy in a content-dependent way and guarantee the worst case reliability while maximizing the code size. In this paper, we first study nonuniform codes for Z-channels, namely, they only suffer one type of errors, say $1 \to 0$. Specifically, we derive their upper bounds, analyze their asymptotic performances, and introduce two general constructions. Then, we extend the concept and results of nonuniform codes to general binary asymmetric channels, where the error probability for each bit from 0 to 1 is smaller than that from 1 to 0.**

***Index Terms*—Asymmetric errors, bounds and constructions, coding for data storage, nonuniform codes.**

I. Introduction

Asymmetric errors exist in many storage devices [5]. In optical disks, read-only memories, and quantum memories, the error probability from 1 to 0 is significantly higher than the error probability from 0 to 1, which is modeled by Z-channels where the transmitted sequences only suffer one type of errors, say $1 \to 0$. In some other devices, like flash memories and phase change memories, although the error probability from 0 to 1 is still smaller than that from 1 to 0, it is not ignorable. That means both types of errors say $1 \to 0$ and $0 \to 1$ are possible, modeled by binary asymmetric channels. In contrast to symmetric errors, where the error probability of a codeword is context independent (since the error probability for 1s and 0s is identical), asymmetric errors are context dependent. For example, the all-1 codeword is prone to have more errors than the all-0 codeword in both Z-channels and binary asymmetric channels.

The construction of asymmetric error-correcting codes is a topic that was studied extensively. In [15], Kløve summarized and presented several such codes. In addition, a large amount of efforts are contributed to the design of systematic codes [1], [3], constructing single or multiple error-correcting codes [2], [17], [18], increasing the lower bounds [8]–[10], [28] and applying low-density parity-check (LDPC) codes in the context of asymmetric channels [25]. In particular, Tallini and Bose in [18] and [19] introduced the theory and design of codes capable of simultaneously correcting (or more generally, controlling) t_- $1 \to 0$ errors and t_+ $0 \to 1$ errors. However, the existing approach for code construction is similar to the approach taken in the construction of symmetric error-correcting codes, namely, it assumes that every codeword could tolerate t asymmetric errors (or, in the more general case of t_- $1 \to 0$ errors and t_+ $0 \to 1$ errors) with t (or, t_- and t_+, respectively) independent from the sent codeword. As a result, different codewords might have different reliability. To see this, let us consider errors to be i.i.d., where every bit that is a 1 can change to a 0 by an asymmetric error with crossover probability $p > 0$ and each bit that is a 0 keeps unchanged. For a codeword $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \mathcal{C} \subset \{0,1\}^n$, let $w(\mathbf{x}) = |\{i : 1 \le i \le n, x_i = 1\}|$ denote the Hamming weight of $\mathbf{x}$. Then, the probability for $\mathbf{x}$ to have at most t asymmetric errors is

$$P_t(\mathbf{x}) = \sum_{i=0}^{t} \binom{w(\mathbf{x})}{i} p^i (1-p)^{w(\mathbf{x})-i}.$$

If the code $\mathcal{C}$ can correct t errors, then $P_t(\mathbf{x})$ is the probability of correctly decoding $\mathbf{x}$ (assuming codewords with more than t errors are uncorrectable), and we say that this codeword $\mathbf{x}$ can correct up to t errors. It can be readily observed that the reliability of codewords decreases when their Hamming weights increase, for example, see Fig. 1.

While asymmetric errors are content dependent, in most applications of data storage, the reliability of each codeword should be content independent. So we are interested in the worst case performance rather than the average performance that is commonly considered in telecommunication, and we want to construct error-correcting codes that can guarantee the reliability of every codeword evenly. In this case, it is not desired to let all the codewords tolerate the same number of asymmetric errors, since the codeword with the highest Hamming weight will become a "bottleneck" and limit the code rate. We call the existing codes *uniform codes* while we focus on the notion of *nonuniform codes*, namely, codes whose codewords can tolerate different numbers of asymmetric errors depending on

Manuscript received June 15, 2012; revised October 17, 2012; accepted December 29, 2012. Date of publication January 17, 2013; date of current version April 17, 2013. This work was supported in part by the National Science Foundation (NSF) CAREER Award CCF-0747415, in part by the NSF under Grants ECCS-0802107 and CCF-1217944, and in part by the NSF National Robotics Initiative Award. This paper was presented in part at the 2011 IEEE International Symposium on Information Theory.

H. Zhou is with the Research Laboratory of Electronics, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: hongchao@mit.edu).

A. Jiang is with the Department of Computer Science and Engineering, Texas A&M University, College Station, TX 77843 USA (e-mail: ajiang@cse.tamu.edu).

J. Bruck is with the Department of Electrical Engineering, California Institute of Technology, Pasadena, CA 91125 USA (e-mail: bruck@caltech.edu).

Communicated by O. Milenkovic, Associate Editor for Coding Theory.

Digital Object Identifier 10.1109/TIT.2013.2241175

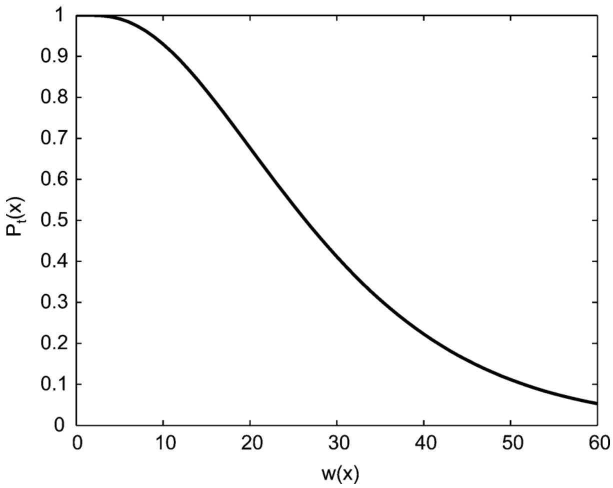

Fig. 1. Relation between $P_t(\mathbf{x})$ and $w(\mathbf{x})$ when $p = 0.1$ and $t = 2$.

their Hamming weights. The goal of introducing nonuniform codes is to maximize the code size while guaranteeing the reliability of each codeword for combating asymmetric errors. Examples of nonuniform codes are error-correcting constant weight codes [4], [13]. In particular, in [21] and [22], some Z-channel capacity achieving feedback coding schemes are given, which are based on constant weight codes. Here, we are interested in forward error correction only (there is no feedback), and we study a general class of nonuniform codes.

In a nonuniform code, given a codeword $\mathbf{x} \in \mathcal{C} \subset \{0,1\}^n$ of weight w, we let $t_\downarrow(w)$ denote the number of $1 \to 0$ errors that $\mathbf{x}$ has to tolerate, and we let $t_\uparrow(w)$ denote the number of $0 \to 1$ errors that $\mathbf{x}$ has to tolerate. Both $t_\downarrow$ and $t_\uparrow$ are step functions on $\{0, 1, \ldots, n\}$ that can be predetermined by the channel, the types of errors, and the required reliability. In this paper, we consider $t_\downarrow$ a nondecreasing function and $t_\uparrow$ a nonincreasing function of codeword weight. As a result, we call such a code as a nonuniform code correcting $[t_\downarrow, t_\uparrow]$ errors. In particular, for Z-channels where $t_\uparrow(w) = 0$ for all $0 \le w \le n$, we call it a nonuniform code correcting $t_\downarrow$ asymmetric errors.

Example 1: In Z-channels, let p be the crossover probability of each bit from 1 to 0 and let $q_e < 1$ be maximal tolerated error probability for each codeword. If we consider the errors to be i.i.d., then we can get

$$t_\downarrow(w) = \min\{s \in N | \sum_{i=0}^{s} \binom{w}{i} p^i (1-p)^{w-i} \ge 1 - q_e\} \quad (1)$$

for $0 \le w \le n$. In this case, every erroneous codeword can be corrected with probability at least $1 - q_e$. □

The following notations will be used throughout of this paper:

q_e	the maximal error probability for each codeword;
$p, p_\downarrow$	the error probability of each bit from 1 to 0;
$p_\uparrow$	the error probability of each bit from 0 to 1;
$t_\downarrow$	a nondecreasing function that indicates the number of $1 \to 0$ errors to tolerate;
$t_\uparrow$	a nonincreasing function that indicates the number of $0 \to 1$ errors to tolerate.

In this paper, we introduce the concept of nonuniform codes and study their basic properties, upper bounds on the rate, asymptotic performance, and code constructions. We first focus on Z-channels and study nonuniform codes correcting $t_\downarrow$ asymmetric errors. The paper is organized as follows: In Section II, we provide some basic properties of nonuniform codes, as the generalizations of those for uniform codes studied in [15]. In Section III, we give an almost explicit upper bound for the size of nonuniform codes. Section IV studies and compares the asymptotic performances of nonuniform codes and uniform codes. Two general constructions, based on multiple layers or bit flips, are proposed in Sections V and VI. Finally, we extend our discussions and results from Z-channels to general binary asymmetric channels in Section VII, where we study nonuniform codes correcting $[t_\downarrow, t_\uparrow]$ errors, namely, $t_\downarrow$ $1 \to 0$ errors and $t_\uparrow$ $0 \to 1$ errors. Concluding remarks are presented in Section VIII.

II. Basic Properties of Nonuniform Codes for Z-Channels

Storage devices such as optical disks, read-only memories, and quantum atomic memories can be modeled by Z-channels, in which the information can suffer a single type of error, namely $1 \to 0$. In this section, we study some properties of nonuniform codes for Z-channels, namely, codes that only correct $t_\downarrow$ asymmetric errors. Typically, $t_\downarrow(w)$ is a nondecreasing function in w, the weight of the codeword. We prove it in the following lemma for the case of i.i.d. errors.

Lemma 1: Assume the errors in a Z-channel are i.i.d.; then, given any $0 < p, q_e < 1$, the function $t_\downarrow$ defined in (1) satisfies $t_\downarrow(w+1) - t_\downarrow(w) \in \{0, 1\}$ for all $0 \le w \le n-1$.

Proof: Let us define

$$P(k, w, p) = \sum_{i=0}^{k} \binom{w}{i} p^i (1-p)^{w-i}.$$

Then

$$P(k, w, p) = (w-k)\binom{w}{k} \int_0^{1-p} t^{w-k-1}(1-t)^k dt$$

which leads us to

$$\begin{aligned} &P(k, w, p) - P(k, w+1, p) \\ &= \frac{k+1}{w+1}[P(k+1, w+1, p) - P(k, w+1, p)]. \end{aligned} \quad (2)$$

First, let us prove that $t_\downarrow(w+1) \ge t_\downarrow(w)$. Since

$$P(k+1, w+1, p) - P(k, w+1, p) > 0$$

we have $P(k, w, p) > P(k, w+1, p)$.

We know that $P(t_\downarrow(w+1), w+1, p) \ge 1 - q_e$, so

$$P(t_\downarrow(w+1), w, p) > 1 - q_e.$$

According to definition of $t_\downarrow(w)$, we can conclude that $t_\downarrow(w+1) \ge t_\downarrow(w)$.

Second, let us prove that $t_\downarrow(w+1) - t_\downarrow(w) \le 1$. Based on (2), we have

$$\begin{aligned} &P(k, w, p) - P(k+1, w+1, p) \\ &= \frac{w-k}{w+1}[P(k, w+1, p) - P(k+1, w+1, p)]. \end{aligned}$$

So $P(k, w, p) < P(k+1, w+1, p)$.

We know that $P(t_\downarrow(w), w, p) \geq 1 - q_e$; therefore

$$P(t_\downarrow(w) + 1, w + 1, p) > 1 - q_e.$$

According to the definition of $t_\downarrow(w+1)$, we have $t_\downarrow(w+1) \leq t_\downarrow(w) + 1$.

This completes the proof. □

Given two binary vectors $\mathbf{x} = (x_1, \ldots, x_n)$ and $\mathbf{y} = (y_1, \ldots, y_n)$, we say $\mathbf{x} \leq \mathbf{y}$ if and only if $x_i \leq y_i$ for all $1 \leq i \leq n$. Let $\mathcal{B}(\mathbf{x})$ be the (asymmetric) "ball" centered at $\mathbf{x}$, namely, it consists of all the vectors obtained by changing at most $t_\downarrow(w(\mathbf{x}))$ 1s in $\mathbf{x}$ into 0s, i.e.,

$$\mathcal{B}(\mathbf{x}) = \{\mathbf{v} \in \{0,1\}^n | \mathbf{v} \leq \mathbf{x} \text{ and } N(\mathbf{x}, \mathbf{v}) \leq t_\downarrow(w(\mathbf{x}))\}$$

where $w(\mathbf{x})$ is the weight of $\mathbf{x}$ and

$$N(\mathbf{x}, \mathbf{y}) \stackrel{\triangle}{=} |\{i : x_i = 1, y_i = 0\}|.$$

We have the following properties of nonuniform codes as the generalizations of those for uniform codes studied in [15].

Lemma 2: Code $\mathcal{C}$ is a nonuniform code correcting $t_\downarrow$ asymmetric errors if and only if $\mathcal{B}(\mathbf{x}) \bigcap \mathcal{B}(\mathbf{y}) = \phi$ for all $\mathbf{x}, \mathbf{y} \in \mathcal{C}$ with $\mathbf{x} \neq \mathbf{y}$.

Proof: According to the definition of nonuniform codes, all the vectors in $\mathcal{B}(\mathbf{x})$ can be decoded as $\mathbf{x}$, and all the vectors in $\mathcal{B}(\mathbf{y})$ can be decoded as $\mathbf{y}$. Hence, $\mathcal{B}(\mathbf{x}) \bigcap \mathcal{B}(\mathbf{y}) = \phi$ for all $\mathbf{x}, \mathbf{y} \in \mathcal{C}$. □

Lemma 3: There always exists a nonuniform code of the maximum size that corrects $t_\downarrow$ asymmetric errors and contains the all-zero codeword.

Proof: Let $\mathcal{C}$ be a nonuniform code correcting $t_\downarrow$ asymmetric errors, and assume that $00\ldots00 \notin \mathcal{C}$. If there exists a codeword $\mathbf{x} \in \mathcal{C}$ such that $00\ldots00 \in \mathcal{B}(\mathbf{x})$, then we can get a new nonuniform code $\mathcal{C}'$ of the same size by replacing $\mathbf{x}$ with $00\ldots00$ in $\mathcal{C}$. If there does not exist a codeword $\mathbf{x} \in \mathcal{C}$ such that $00\ldots00 \in \mathcal{B}(\mathbf{x})$, then we can get a larger nonuniform code $\mathcal{C}'$ by adding $00\ldots00$ to $\mathcal{C}$. □

Given a nonuniform code $\mathcal{C}$, let A_r denote the number of codewords with Hamming weight r in $\mathcal{C}$, i.e.,

$$A_r = |\{\mathbf{x} \in \mathcal{C} | w(\mathbf{x}) = r\}|.$$

Given a nondecreasing function $t_\downarrow$, let R_r denote a set of weights that can reach weight r with at most $t_\downarrow$ asymmetric errors, namely,

$$R_r = \{0 \leq s \leq n | s - t_\downarrow(s) \leq r \leq s\}.$$

Lemma 4: Let $\mathcal{C}$ be a nonuniform code correcting $t_\downarrow$ asymmetric errors. For $0 \leq r \leq n$, we have

$$\sum_{j \in R_r} \binom{j}{r} A_j \leq \binom{n}{r}. \tag{3}$$

Proof: Let $V_r = \{\mathbf{x} \in \{0,1\}^n | w(\mathbf{x}) = r\}$ be the set consisting of all the vectors of length n and weight r. If $\mathbf{x} \in \mathcal{C}$ with $w(\mathbf{x}) = j \in R_r$, according to the properties of $t_\downarrow$, $\mathcal{B}(\mathbf{x})$ contains $\binom{j}{r}$ vectors of weight r, namely

$$|V_r \bigcap \mathcal{B}(\mathbf{x})| = \binom{j}{r}.$$

According to Lemma 2, we know that $\bigcup_{\mathbf{x} \in \mathcal{C}} (V_r \bigcap \mathcal{B}(\mathbf{x}))$ is a disjoint union, in which the number of vectors is

$$\sum_{j \in R_r} \binom{j}{r} A_j.$$

Since $\bigcup_{\mathbf{x} \in \mathcal{C}} (V_r \bigcap \mathcal{B}(\mathbf{x})) \subseteq V_r$ and there are at most $\binom{n}{r}$ vectors in V_r, the lemma follows. □

III. Upper Bounds

Let $B_\alpha(n, t)$ denote the maximum size of a uniform code correcting t asymmetric errors, and let $B_\beta(n, t_\downarrow)$ denote the maximum size of a nonuniform code correcting $t_\downarrow$ asymmetric errors, where t is a constant and $t_\downarrow$ is a nondecreasing function of codeword weight. In this section, we first present some existing results on the upper bounds of $B_\alpha(n, t)$ for uniform codes. Then, we derive an almost explicit upper bound of $B_\beta(n, t_\downarrow)$ for nonuniform codes.

A. Upper Bounds for Uniform Codes

An explicit upper bound to $B_\alpha(n, t)$ was given by Varshamov [23]. In [15], Borden showed that $B_\alpha(n, t)$ is upper bounded by

$$\min\{A(n+t, 2t+1), (t+1)A(n, 2t+1)\}$$

where $A(n, d)$ is the maximal number of vectors in $\{0,1\}^n$ with Hamming distance at least d. Goldbaum [12] pointed out that the upper bounds can be obtained using integer programming. By adding more constraints to the integer programming, the upper bounds were later improved by Delsarte and Piret [7] and Weber *et al.* [26], [27]. Kløve generalized the bounds of Delsarte and Piret, and gave an almost explicit upper bound which is very easy to compute by relaxing some of the constraints[14], in the following way.

Theorem 5: [14] For $n > 2t \geq 2$, let $y_0, y_1, \ldots, y_n$ be defined by

1) $y_0 = 1$
2) $y_r = 0, \quad \forall\, 1 \leq r \leq t$
3) $y_{t+r} = \frac{1}{\binom{t+r}{t}}[\binom{n}{r} - \sum_{j=0}^{t-1} y_{r+j} \binom{r+j}{j}], \forall\, 1 \leq r \leq \frac{n}{2} - t$
4) $y_{n-r} = y_r, \quad \forall\, 0 \leq r < \frac{n}{2}$.

Then, $B_\alpha(n, t) \leq M_\alpha(n, t) \stackrel{\triangle}{=} \sum_{r=0}^{n} y_r$.

This method obtains a good upper bound to $B_\alpha(n, t)$ (although it is not the best known one). Since it is easy to compute, when n and t are large, it is every useful for analyzing the sizes of uniform codes.

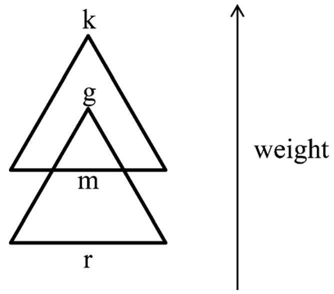

Fig. 2. This diagram demonstrates the relative values of r, g, k, m.

B. Upper Bounds for Nonuniform Codes

We now derive an almost explicit upper bound for the size of nonuniform codes correcting $t_\downarrow$ asymmetric errors, followed the idea of Kløve [14] for uniform codes. According to the lemmas in the previous section, we can get an upper bound of $B_\beta(n, t_\downarrow)$, denoted by $M_\beta(n, t_\downarrow)$, such that

$$M_\beta(n, t_\downarrow) = \max \sum_{i=0}^{n} z_r$$

where the maximum is taken over the following constraints:

$$1)\ z_r \text{ are nonnegative real numbers}$$
$$2)\ z_0 = 1$$
$$3)\ \sum_{j \in R_r} \binom{j}{r} z_j \le \binom{n}{r}, \forall\, 0 \le r \le n.$$

Here, condition 2) is given by Lemma 3, and condition 3) is given by Lemma 4. Our goal is to find an almost explicit way to calculate $M_\beta(n, t_\downarrow)$.

Lemma 6: Assume $\sum_{r=0}^{n} z_r$ is maximized over $z_0, z_1, \ldots, z_n$ in the problem above. If $r = s - t_\downarrow(s)$ for some integer s with $0 \le s, r \le n$, then

$$Z_r = \sum_{j \in R_r} \binom{j}{r} z_j = \binom{n}{r}.$$

Proof: Suppose that $Z_r < \binom{n}{r}$ for some r that satisfies the aforementioned condition. Let $g = \max R_r$ and $k = \min\{w | z_w > 0, w > g\}$, as indicated in Fig. 2, where a triangular denote the ball centered at the top vertex. Furthermore, we let $m = \max\{w | k - t_\downarrow(k) > w\}$. Note that in this case, $r = g - t_\downarrow(g)$ and $m = k - t_\downarrow(k) - 1$.

We first prove that for all $r \le w \le m$, $Z_w < \binom{n}{w}$. In order to prove this, we let $s = w - r$; then, we get

$$\begin{aligned} Z_w &= \sum_{j \in R_w} z_j \binom{j}{w} \\ &= \sum_{j=w}^{g} z_j \binom{j}{w} \\ &= \sum_{j=s}^{g-r} z_{r+j} \binom{r+j}{r+s}. \end{aligned}$$

It is easy to obtain that

$$\binom{r+j}{r+s} = \binom{r+j}{r} \frac{\binom{j}{s}}{\binom{r+s}{s}}$$

So

$$\begin{aligned} Z_w &\le \frac{\binom{g-r}{s}}{\binom{r+s}{s}} \sum_{j=s}^{g-r} z_{r+j} \binom{r+j}{r} \\ &< \frac{\binom{g-r}{s}}{\binom{r+s}{s}} \binom{n}{r} \\ &= \frac{(g-r)(g-r-1)\ldots(g-r-s+1)}{(n-r)(n-r-1)\ldots(n-r-s+1)} \binom{n}{r+s} \\ &\le \binom{n}{w}. \end{aligned}$$

Now, we construct a new group of real numbers $z_0^*, z_1^*, \ldots, z_n^*$ such that

1) $z_g^* = z_g + \Delta$,
2) $z_k^* = z_k - \delta$,
3) $z_r^* = z_r$ for $r \ne h, r \ne k$

with

$$\Delta = \min\left(\left\{\frac{\binom{n}{w} - Z_w}{\binom{g}{w}} \Big| r \le w \le m\right\} \bigcup \left\{\frac{\binom{k}{w}}{\binom{g}{w}} z_k \Big| m < w \le g\right\}\right)$$
$$\delta = \frac{1}{\min\left\{\frac{\binom{k}{w}}{\binom{g}{w}} \Big| m < w \le g\right\}} \Delta.$$

For such Δ, δ, it is not hard to prove that $Z_r^* = \binom{n}{r}$ for $0 \le r \le n$. On the other hand

$$\sum_{r=0}^{n} z_r^* = \sum_{r=0}^{n} z_r + \Delta - \delta > \sum_{r=0}^{n} z_r$$

which contradicts our assumption that $\sum_{r=0}^{n} z_r$ is maximized over the constraints. So the lemma is true. □

Lemma 7: Assume $\sum_{r=0}^{n} z_r$ is maximized over $z_0, z_1, \ldots, z_n$ in the aforementioned problem. If $r = s - t_\downarrow(s)$ for some integer s with $0 \le s, r \le n$, then

$$Z_r = \sum_{j=r}^{h} \binom{j}{r} z_j = \binom{n}{r}$$

where

$$h = \min\{s \in N | s - t_\downarrow(s) = r\}.$$

Proof: Let $g = \max\{s \in N | s - t_\downarrow(s) = r\}$. If $g = h$, then the lemma is true. So we only need to prove it for the case that $g > k$. According to our assumption, for all $0 \le w \le n$

$$Z_w = \sum_{j \in R_w} z_j \binom{j}{w} \le \binom{n}{w}.$$

If $\sum_{j=h+1}^{g} z_j > 0$, then we can construct a new group of real numbers $z_0^*, z_1^*, \dots, z_n^*$ such that

1) $z_h^* = z_h + \Delta$
2) $z_w^* = 0$ for $h < w \le g$
3) $z_w^* = z_w$ if $w \notin [h, g]$

with

$$\Delta = \min\{\frac{\sum_{j=h+1}^{g} \binom{j}{w} z_j}{\binom{h}{w}} | r \le w \le h\}.$$

Now, we show that this new group of real numbers $z_0^*, z_1^*, \dots, z_n^*$ satisfies our constraints, i.e., for all w with $0 \le w \le n$

$$Z_w^* = \sum_{j \in R_r} \binom{j}{r} z_j^* \le \binom{n}{w}.$$

Specifically, we only need to focus on the case that $r \le w \le h$.

If $r \le w \le h$, then

$$\begin{aligned} Z_w^* &= \sum_{j \in R_r} \binom{j}{r} z_j^* \\ &= (\sum_{j=w}^{h} \binom{j}{w} z_j) + \binom{h}{w} \Delta + (\sum_{j=g+1}^{\max R_r} \binom{j}{w} z_j) \\ &\le \sum_{j=w}^{h} \binom{j}{w} z_j + \sum_{j=h+1}^{g} \binom{j}{w} z_j + \sum_{j=g+1}^{\max R_r} \binom{j}{w} z_j \\ &= Z_w \\ &\le \binom{n}{w}. \end{aligned}$$

It is easy to see that $\Delta > \sum_{j=h+1}^{g} z_i$; hence, it implies

$$\sum_{r=0}^{n} z_r^* > \sum_{r=0}^{n} z_r$$

which contradicts with our assumption that $\sum_{r=0}^{n} z_r$ is maximized over the constraints. This completes the proof. □

Now let $y_0, y_1, \dots, y_n$ be a group of optimal solutions to $z_0, z_1, \dots, z_n$ that maximize $\sum_{r=0}^{n} z_r$. Then, $y_0, y_1, \dots, y_n$ satisfy the condition in Lemma 7. We see that $y_0 = 1$. Then, based on Lemma 7, we can get $y_1, \dots, y_n$ uniquely by iteration. Hence, we have the following theorem for calculating the upper bound $M_\beta(n, t_\downarrow)$.

Theorem 8: Let $y_0, y_1, \dots, y_n$ be defined by

1) $y_0 = 1$
2) $y_r = \frac{1}{\binom{r}{t_\downarrow(r)}}[\binom{n}{r - t_\downarrow(r)} - \sum_{j=1}^{t_\downarrow(r)} y_{r-j} \binom{r-j}{t_\downarrow(r) - j}]$
 $\forall\ 1 \le r \le n.$

Then, $B_\beta(n, t_\downarrow) \le M_\beta(n, t_\downarrow) = \sum_{r=0}^{n} y_r$.

This theorem provides an almost explicit expression for the upper bound $M_\beta(n, t_\downarrow)$, which is much easier to calculate than the equivalent expression defined at the beginning of this section. Note that in the theorem, we do not have a constraint like the one (constraint 4) in Theorem 5. It is because that the optimal nonuniform codes do not have symmetric weight distributions due to the fact that $t_\downarrow(w)$ monotonically increases with w (demonstrated in Lemma 1).

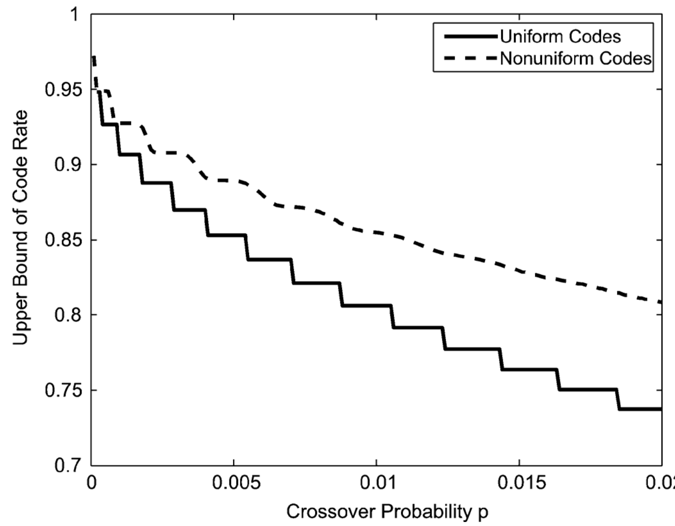

Fig. 3. Upper bounds of the rates for uniform/nonuniform codes when $n = 255$ and $q_e = 10^{-4}$.

C. Comparison of Upper Bounds

Here, we focus on i.i.d. errors, i.e., given the crossover probability p from 0 to 1 and the maximal tolerated error probability q_e, the function $t_\downarrow$ is defined in (1). In this case, we can write the maximum size of a uniform code as $B_\alpha(n, t_\downarrow(n)) = B_\alpha(n, p, q_e)$, and write the maximum size of a nonuniform code as $B_\beta(n, t_\downarrow(w)) = B_\beta(n, p, q_e)$.

Now we let $\eta_\alpha(n, p, q_e)$ denote the maximal code rate defined by

$$\eta_\alpha(n, p, q_e) = \frac{\log B_\alpha(n, p, q_e)}{n}.$$

Similar, we let $\eta_\beta(n, p, q_e)$ denote the maximal code rate defined by

$$\eta_\beta(n, p, q_e) = \frac{\log B_\beta(n, p, q_e)}{n}.$$

By the definition of uniform and nonuniform codes, it is simple to see that $\eta_\beta(n, p, q_e) \ge \eta_\alpha(n, p, q_e)$.

Fig. 3 depicts the upper bounds of $\eta_\alpha(n, p, q_e)$ and $\eta_\beta(n, p, q_e)$ for different values of p when $n = 255$ and $q_e = 10^{-4}$. The upper bound of $\eta_\alpha(n, p, q_e)$ is obtained based on the almost explicit upper bound given by Kløve, and the upper bound of $\eta_\beta(n, p, q_e)$ is obtained based on the almost explicit method proposed in this section. It demonstrates that given the same parameters, the upper bound for nonuniform codes is substantially greater than that for uniform codes.

IV. Asymptotic Performance

In this section, we study and compare the asymptotic rates of uniform codes and nonuniform codes. Note that the performance of nonuniform codes strongly depends on the selection of the function $t_\downarrow$. Here, we focus on i.i.d. errors, so given

$0 < p, q_e < 1$, we study the asymptotic behavior of $\eta_\alpha(n,p,q_e)$ and $\eta_\beta(n,p,q_e)$ as $n \to \infty$. By the definition of nonuniform and uniform codes, the "balls" containing up to $t_\downarrow(\mathbf{x})$ (or $t_\downarrow(n)$) errors that are centered at codewords $\mathbf{x}$ need to be disjoint.

Before giving the asymptotic rates, we first present the following known result. For any $\delta > 0$, when n is large enough, we have

$$2^{n(H(\frac{k}{n})-\delta)} \leq \binom{n}{k} \leq 2^{n(H(\frac{k}{n})+\delta)}$$

where $H(p)$ is the entropy function with

$$H(p) = p\log\frac{1}{p} + (1-p)\log\frac{1}{1-p} \text{for } 0 \leq p \leq 1$$

and

$$H(p) = 0 \text{ for } p > 1 \text{ or } p < 0.$$

Lemma 9: Let $A(n,d,w)$ be the maximum size of a constant-weight binary code of codeword length n with Hamming weight w and minimum distance d. Let $R(n,t,w)$ be the maximum size of a constant weight binary code of codeword length n and Hamming weight w that is capable of correcting t asymmetric errors. Then

$$R(n,t,w) = A(n, 2(t+1), w).$$

Proof: Let $\mathcal{C}$ be a code of length n, constant weight w, and size $R(n,t,w)$ that corrects t asymmetric errors. For all $\mathbf{x} \in \{0,1\}^n$, let us define $S_t(\mathbf{x})$ be the set consisting of all the vectors obtained by changing at most t 1 s in $\mathbf{x}$ into 0s, i.e.,

$$S_t(\mathbf{x}) = \{\mathbf{v} \in \{0,1\}^n | \mathbf{v} < \mathbf{x} \text{ and } N(\mathbf{x},\mathbf{v}) \leq t\}.$$

Then, $\forall \mathbf{x},\mathbf{y} \in \mathcal{C}$, we know that $S_t(\mathbf{x}) \bigcap S_t(\mathbf{y}) = \phi$.

Let $\mathbf{u} = (u_1,\ldots,u_n)$ be a vector such that $u_i = \min\{x_i, y_i\}$ for $1 \leq i \leq n$. Then, $N(\mathbf{x},\mathbf{u}) = N(\mathbf{y},\mathbf{u})$ and $\mathbf{u} \notin S_t(\mathbf{x}) \bigcap S_t(\mathbf{y})$. Without loss of generality (w.l.o.g.), suppose that $\mathbf{u} \notin S_t(\mathbf{x})$. Then, $N(\mathbf{x},\mathbf{u}) > t$, and the Hamming distance between $\mathbf{x}$ and $\mathbf{y}$ is

$$d(\mathbf{x},\mathbf{y}) = N(\mathbf{x},\mathbf{u}) + N(\mathbf{y},\mathbf{u}) \geq 2(t+1).$$

So the minimum distance of $\mathcal{C}$ is at least $2(t+1)$. As a result, $A(n,2(t+1),w) \geq R(n,t,w)$.

On the other hand, if a constant-weight code has minimum distance at least $2(t+1)$, it can correct t asymmetric errors. As a result, $R(n,t,w) \geq A(n,2(t+1),w)$. □

A. Bounds of $\lim_{n\to\infty} \eta_\alpha(n,p,q_e)$

Let us first give the lower bound of $\lim_{n\to\infty} \eta_\alpha(n,p,q_e)$ and then provide the upper bound.

Theorem 10 (Lower Bound): Given $0 < q_e < 1$, if $0 < p \leq \frac{1}{4}$, we have

$$\lim_{n\to\infty} \eta_\alpha(n,p,q_e) \geq 1 - H(2p).$$

Proof: We consider uniform codes that correct t asymmetric errors, where

$$t = \min\{s | \sum_{i=0}^{s} \binom{n}{i} p^i (1-p)^{n-i} \geq 1 - q_e\}$$

i.e.,

$$\sum_{t+1}^{n} \binom{n}{i} p^i (1-p)^{n-i} \leq q_e$$
$$\sum_{t}^{n} \binom{n}{i} p^i (1-p)^{n-i} \geq q_e.$$

According to Hoeffding's inequality, for $\delta > 0$, if $t < (p-\delta)n$ as $n \to \infty$

$$\sum_{t+1}^{n} \binom{n}{i} \to 1$$

which contradicts with the aforementioned inequality. If $t > (p+\delta)n$ as $n \to \infty$

$$\sum_{t}^{n} \binom{n}{i} \to 0$$

which also contradicts with the aforementioned inequality.

So for any $\delta > 0$, as n becomes large enough, we have $(p-\delta)n \leq t \leq (p+\delta)n$. If we write $t = \gamma n$, then $p - \delta \leq \gamma \leq p + \delta$ for n large enough.

Since each codeword tolerates t asymmetric errors, we have

$$B_\alpha(n,p,q_e) = B_\alpha(n,t) \geq R(n,t,w) = A(n,2(t+1),w)$$

for every w with $0 \leq w \leq n$. The Gilbert Bound gives that (see Graham and Sloane[13])

$$A(n,2(t+1),w) \geq \frac{\binom{n}{w}}{\sum_{i=0}^{t} \binom{w}{i}\binom{n-w}{i}}.$$

Hence

$$\begin{aligned} B_\alpha(n,p,q_e) &\geq \max_{w=0}^{n} \frac{\binom{n}{w}}{\sum_{i=0}^{t} \binom{w}{i}\binom{n-w}{i}} \\ &\geq \max_{w=0}^{n} \frac{\binom{n}{w}}{n \max_{i\in[0,t]} \binom{w}{i}\binom{n-w}{i}} \\ &\geq \max_{w:\frac{w(n-w)}{n} > t} \frac{\binom{n}{w}}{n \max_{i\in[0,t]} \binom{w}{i}\binom{n-w}{i}} \\ &\geq \max_{w:\frac{w(n-w)}{n} > t} \frac{\binom{n}{w}}{n\binom{w}{t}\binom{n-w}{t}}. \end{aligned}$$

For a binomial term $\binom{n}{k} = \frac{n!}{k!(n-k)!}$ and $\delta > 0$, when n is large enough

$$2^{n(H(\frac{k}{n})-\delta)} \leq \binom{n}{k} \leq 2^{n(H(\frac{k}{n})+\delta)}.$$

Let $w = \theta n$ and $t = \gamma n$ with $0 \leq \theta, \gamma \leq 1$, as n becomes large enough, we have

$$\begin{aligned}
&\eta_\alpha(n,p,q_e)\\
&= \frac{1}{n}\log_2 B_\alpha(n,p,q_e)\\
&\geq \frac{1}{n}\log_2 \max_{w:\frac{w(n-w)}{n}>t} \frac{\binom{n}{w}}{n\binom{w}{t}\binom{n-w}{t}}\\
&\geq \frac{1}{n}\log_2 \max_{\theta:\theta(1-\theta)>\gamma} \frac{2^{(H(\theta)-\delta)n}}{n2^{(H(\frac{\gamma}{\theta})+\delta)\theta n}2^{(H(\frac{\gamma}{1-\theta})+\delta)(1-\theta)n}}\\
&\geq \max_{\theta:\theta(1-\theta)\geq\gamma} H(\theta) - \theta H(\frac{\gamma}{\theta}) - (1-\theta)H(\frac{\gamma}{1-\theta}) - 2\delta\\
&\qquad + \frac{1}{n}\log\frac{1}{n}.
\end{aligned}$$

From $\theta(1-\theta) \geq \gamma$, we get $\theta > \gamma > 0$; then, $H(\frac{\gamma}{\theta})$ is a continuous function of γ. As n becomes large, we have $p-\delta \leq \gamma \leq p+\delta$, so we can approximate $H(\frac{\gamma}{\theta})$ with $H(\frac{p}{\theta})$. Similarly, we can approximate $H(\frac{\gamma}{1-\theta})$ with $H(\frac{p}{1-\theta})$. Then, we can get as $n \to \infty$

$$\begin{aligned}
&\eta_\alpha(n,p,q_e)\\
&\geq \max_{\theta:\theta(1-\theta)>p} H(\theta) - \theta H(\frac{p}{\theta}) - (1-\theta)H(\frac{p}{1-\theta}).
\end{aligned}$$

If $0 \leq p \leq \frac{1}{4}$, the maximum value can be achieved at $\theta^* = \frac{1}{2}$. Hence, we have

$$\lim_{n\to\infty} \eta_\alpha(n,p,q_e) \geq 1 - H(2p).$$

This completes the proof. □

Theorem 11 (Upper Bound): Given $0 < p, q_e < 1$, we have

$$\lim_{n\to\infty} \eta_\alpha(n,p,q_e) \leq (1+p)[1 - H(\frac{p}{1+p})].$$

Proof: For a uniform code correcting t asymmetric errors, we have the following observations.

1) There is at most one codeword with Hamming weight at most t.
2) For $t+1 \leq w \leq n$, the number of codewords with Hamming weight w is at most $\frac{\binom{n}{w-t}}{\binom{w}{t}}$.

Consequently, the total number of codewords is

$$\begin{aligned}
B_\alpha(n,p,q_e) &\leq 1 + \sum_{w=t+1}^{n} \frac{\binom{n}{w-t}}{\binom{w}{t}}\\
&= 1 + \sum_{w=t+1}^{n} \frac{\binom{n+t}{w}}{\binom{n+t}{t}}\\
&\leq \frac{2^{n+t}}{\binom{n+t}{t}}.
\end{aligned}$$

So as $n \to \infty$, we have

$$\begin{aligned}
\eta_\alpha(n,p,q_e) &\leq \frac{1}{n}\log[\frac{2^{n+t}}{\binom{n+t}{t}}]\\
&\leq \frac{1}{n}\log\frac{2^{(1+\gamma)n}}{2^{H(\frac{\gamma}{1+\gamma})(1+\gamma)n}}\\
&= (1+\gamma) - H(\frac{\gamma}{1+\gamma})(1+\gamma)\\
&= (1+p)[1 - H(\frac{p}{1+p})]
\end{aligned}$$

where the last step is due to the continuousness of $(1+\gamma) - H(\frac{\gamma}{1+\gamma})(1+\gamma)$ over γ.

This completes the proof. □

We see that when $n \to \infty$, $\eta_\alpha(n,p,q_e)$ does not depend on q_e as long as $0 < q_e < 1$. It is because that when $n \to \infty$, we have $t \to pn$, which does not depend on q_e. This property also holds by $\eta_\beta(n,p,q_e)$ when $n \to \infty$.

B. Bounds of $\lim_{n\to\infty} \eta_\beta(n,p,q_e)$

In this section, we study the bounds of the asymptotic rates of nonuniform codes. Here, we use the same idea as that for uniform codes, besides that we need also to prove that the "edge effect" can be ignored, i.e., the number of codewords with Hamming weight $w \ll n$ does not dominate the final result.

Theorem 12 (Lower Bound): Given $0 < p, q_e < 1$, we have

$$\lim_{n\to\infty} \eta_\beta(n,p,q_e) \geq \max_{0\leq\theta\leq 1-p} H(\theta) - \theta H(p) - (1-\theta)H(\frac{p\theta}{1-\theta}).$$

Proof: We consider nonuniform codes that corrects $t_\downarrow$ asymmetric errors, where

$$t_\downarrow(w) = \min\{s | \sum_{i=0}^{s} \binom{w}{i} p^i (1-p)^{w-i} \geq 1 - q_e\}$$

for all $0 \leq w \leq n$.

Based on Hoeffding's inequality, for any $\delta > 0$, as w becomes large enough, we have $(p-\delta)w \leq t_\downarrow(w) \leq (p+\delta)w$. In another word, for any $\epsilon, \delta > 0$, when n is large enough and $w \geq \epsilon n$, we have $(p-\delta)w \leq t_\downarrow(w) \leq (p+\delta)w$.

Let $w = \theta n$ and $t_\downarrow(w) = \gamma w$; then, when n is large enough, if $\theta > \epsilon$, we have

$$(p-\delta) \leq \gamma \leq (p+\delta).$$

If $\theta < \epsilon$, we call it the "edge" effect. In this case, $0 \leq \gamma \leq 1$.

Since each codeword with Hamming weight w can tolerate $t_\downarrow(w)$ errors

$$B_\beta(n,p,q_e) \geq R(n, t_\downarrow(w), w) \geq A(n, 2(t_\downarrow(w)+1), w)$$

for every w with $0 \leq w \leq n$.

Applying the Gilbert Bound, we have

$$B_\beta(n,p,q_e) \geq \max_w \frac{\binom{n}{w}}{\sum_{i=0}^{t_\downarrow(w)} \binom{w}{i}\binom{n-w}{i}}.$$

Then

$$\begin{aligned}
B_\beta(n,p,q_e) &\geq \max_w \frac{\binom{n}{w}}{\max_{i\in[0,t_\downarrow(w)]} n\binom{w}{i}\binom{n-w}{i}}\\
&\geq \max_{w:\frac{w(n-w)}{n}\geq t_\downarrow(w)} \frac{\binom{n}{w}}{n\binom{w}{t_\downarrow(w)}\binom{n-w}{t_\downarrow(w)}}.
\end{aligned}$$

TABLE I
UPPER BOUNDS AND LOWER BOUNDS FOR THE MAXIMUM RATES OF UNIFORM CODES AND NONUNIFORM CODES

	Lower Bound	Upper Bound
$\lim_{n\to\infty}\eta_\alpha(n,p,q_e)$	$[1-H(2p)]I_{0\leq p\leq\frac{1}{4}}$	$(1+p)[1-H(\frac{p}{1+p})]$
$\lim_{n\to\infty}\eta_\beta(n,p,q_e)$	$\max_{0\leq\theta\leq 1-p} H(\theta)-\theta H(p)-(1-\theta)H(\frac{p\theta}{1-\theta})$	$\max_{0\leq\theta\leq 1} H((1-p)\theta)-\theta H(p)$

When $n\to\infty$, we have

$$\begin{aligned}
&\eta_\beta(n,p,q_e)\\
&=\frac{1}{n}\log_2 B_\beta(n,p,q_e)\\
&\geq\frac{1}{n}\log_2\max_{\theta:(1-\theta)\geq\gamma}\frac{2^{(H(\theta)-\delta)n}}{n2^{(H(\gamma)+\delta)\theta n}2^{(H(\frac{\gamma\theta}{1-\theta})+\delta)(1-\theta)n}}\\
&\geq\max_{\theta:(1-\theta)\geq\gamma} H(\theta)-\theta H(\gamma)-(1-\theta)H(\frac{\gamma\theta}{1-\theta})\\
&\qquad-2\delta+\frac{1}{n}\log\frac{1}{n}\\
&=\max_{\theta:(1-\theta)\geq\gamma} H(\theta)-\theta H(\gamma)-(1-\theta)H(\frac{\gamma\theta}{1-\theta}).
\end{aligned}$$

Note that when $\theta<\epsilon$ for small ϵ, we have

$$H(\theta)-\theta H(\gamma)-(1-\theta)H(\frac{\gamma\theta}{1-\theta})\sim 0.$$

So we can ignore this edge effect. That implies that we can write

$$p-\delta\leq\gamma\leq p+\delta$$

for any θ with $0\leq\theta\leq 1$.

Since $1-\theta\geq\gamma>0$ for any fixed θ

$$H(\theta)-\theta H(\gamma)-(1-\theta)H(\frac{\gamma\theta}{1-\theta})$$

is a continuous function of γ. As $n\to\infty$, we have

$$\eta_\beta(n,p,q_e)\geq\max_{\theta:(1-\theta)\geq p} H(\theta)-\theta H(p)-(1-\theta)H(\frac{p\theta}{1-\theta}).$$

This completes the proof. □

Theorem 13 (Upper Bound): Given $0<p,q_e<1$, we have

$$\begin{aligned}
\lim_{n\to\infty}\eta_\beta(n,p,q_e)&\leq\max_{0\leq\theta\leq 1} H((1-p)\theta)-\theta H(p)\\
&=H(\frac{1}{2^{s(p)}+1})+\frac{s(p)}{2^{s(p)}+1}
\end{aligned}$$

with $s(p)=H(p)/(1-p)$.

Proof: The upper bound is the capacity of the Z-channel given in [20]. □

C. Comparison of Asymptotic Performances

Table I summarizes the analytic upper bounds and lower bounds of $\lim_{n\to\infty}\eta_\alpha(n,p,q_e)$ and $\lim_{n\to\infty}\eta_\beta(n,p,q_e)$ obtained in this section. For the convenience of comparison, we plot them in Fig. 4. The dashed curves represent the lower and upper bounds to $\lim_{n\to\infty}\eta_\alpha(n,p,q_e)$, and the solid curves represent the lower and upper bounds to $\lim_{n\to\infty}\eta_\beta(n,p,q_e)$. The gap between the bounds for the two codes indicate the potential improvement in efficiency (code rate) by using the nonuniform codes (compared to using uniform codes) when the codeword length is large. In this figure, the upper bound for nonuniform codes is also the capacity of the Z-channel. It shows that nonuniform codes may be able to achieve the Z-channel capacity as n becomes large, while uniform codes cannot (here we assume that they have codewords of high weights and worst case performance is considered, so the constructions of uniform codes cannot achieve the capacity of Z-channel).

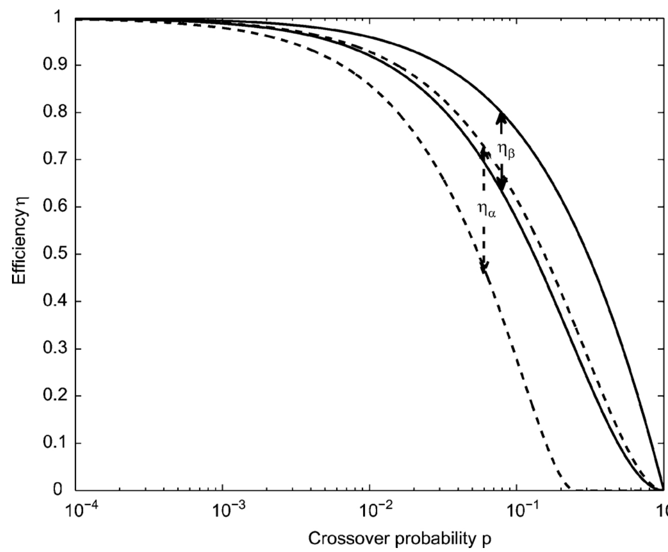

Fig. 4. Bounds of $\lim_{n\to\infty}\eta_\alpha(n,p,q_e)$ and $\lim_{n\to\infty}\eta_\beta(n,p,q_e)$.

V. LAYERED CODES CONSTRUCTION

In [15], Kløve summarized some constructions of uniform codes for correcting asymmetric errors. The code of Kim and Freiman was the first one constructed for correcting multiple asymmetric errors. Varshamov [24] and Constrain and Rao [6] presented some constructions based on group theory. Later, Delsarte and Piret [7] proposed a construction based on "expurgating/puncturing" with some improvements given by Weber *et al.* [27]. It is natural for us to ask whether it is possible to construct nonuniform codes based on existing constructions of uniform codes. In this section, we propose a general construction of nonuniform codes based on multiple layers. It shows that the sizes of the codes can be significantly increased by equalizing the reliability of all the codewords.

A. Layered Codes

Let us start from a simple example: Assume we want to construct a nonuniform code with codeword length $n=10$ and

$$t_\downarrow(w)=\begin{cases}0, & \text{for } w=0\\ 1, & \text{for } 1\leq w\leq 5\\ 2, & \text{for } 6\leq w\leq 10.\end{cases}$$

In this case, how can we construct a nonuniform code efficiently? Intuitively, we can divide all the codewords into two layers such that each layer corresponds to an individual uniform code, namely, we get a nonuniform code

$$\mathcal{C} = \{\mathbf{x} \in \{0,1\}^n | w(\mathbf{x}) \leq 5, \mathbf{x} \in \mathcal{C}_1\} \bigcup \{\mathbf{x} \in \{0,1\}^n | w(\mathbf{x}) \geq 6, \mathbf{x} \in \mathcal{C}_2\}$$

where $\mathcal{C}_1$ is a uniform code correcting one asymmetric error and $\mathcal{C}_2$ is a uniform code correcting two asymmetric errors. So we can obtain a nonuniform code by combining multiple uniform codes, each of which corrects a number of asymmetric errors. We call nonuniform codes constructed in this way as *layered codes*. However, the simple construction above has a problem—due to the interference of neighbor layers, the codewords at the bottom of the higher layer may violate our requirement of reliability, namely, they cannot correct sufficient asymmetric errors. To solve this problem, we can construct a layered code in the following way. Let us first construct a uniform code correcting two asymmetric errors. Then, we add more codewords into the code such that we have the following.

1) The weights of these additional codewords are less than $4 = 6 - t_\downarrow(6)$. This condition can guarantee that in the resulting nonuniform code, all the codewords with weights at least 6 can tolerate two errors.
2) These additional codewords are selected such that the codewords with weights at most 5 can tolerate one error.

B. Construction

Generally, given a nondecreasing function $t_\downarrow$, we can get a nonuniform code with $t_\downarrow(n)$ layers by iterating the aforementioned process. Based on this idea, given $n, t_\downarrow$, we construct layered codes as follows.

Let $k = t_\downarrow(n)$ and let $\mathcal{C}_1, \ldots, \mathcal{C}_k$ be k binary codes of codeword length n, where

$$\mathcal{C}_1 \supset \cdots \supset \mathcal{C}_k$$

and for $1 \leq t \leq k$, the code $\mathcal{C}_t$ can correct t asymmetric errors. Given $t_\downarrow$, we can construct a layered code $\mathcal{C}$ such that

$$\mathcal{C} = \{\mathbf{x} \in \{0,1\}^n | \mathbf{x} \in \mathcal{C}_{t_l(w(\mathbf{x}))}\}$$

where

$$\begin{aligned} t_l(w(\mathbf{x})) &= t_\downarrow(\max R_{w(\mathbf{x})}) \\ &= t_\downarrow(\max\{s | s - t_\downarrow(s) \leq w(\mathbf{x})\}). \end{aligned}$$

We see that there is a shift of the layers (corresponding to the function t_l and the function $t_\downarrow$), see Fig. 5 as a demonstration. The following theorem shows that the aforementioned construction satisfies our requirements of nonuniform codes, i.e., it corrects $t_\downarrow$ asymmetric errors.

Theorem 14: Let $\mathcal{C}$ be a layered code based on the aforementioned construction; then, for all $\mathbf{x} \in \mathcal{C}$, $\mathbf{x}$ can tolerate $t_\downarrow(w(\mathbf{x}))$ asymmetric errors.

Proof: We prove that for all $\mathbf{x}, \mathbf{y} \in \mathcal{C}$ with $\mathbf{x} \neq \mathbf{y}$, $\mathcal{B}(\mathbf{x}) \bigcap \mathcal{B}(\mathbf{y}) = \phi$. W.l.o.g., we assume $w(\mathbf{x}) \geq w(\mathbf{y})$.

If $w(\mathbf{x}) - t_\downarrow(w(\mathbf{x})) > w(\mathbf{y})$, the conclusion is true.

If $w(\mathbf{x}) - t_\downarrow(w(\mathbf{x})) \leq w(\mathbf{y})$ and $w(\mathbf{x}) \geq w(\mathbf{y})$, then $\mathbf{x}, \mathbf{y} \in \mathcal{C}_{t_l(w(\mathbf{y}))}$. That means there does not exist a word $\mathbf{z} \in \{0,1\}^n$

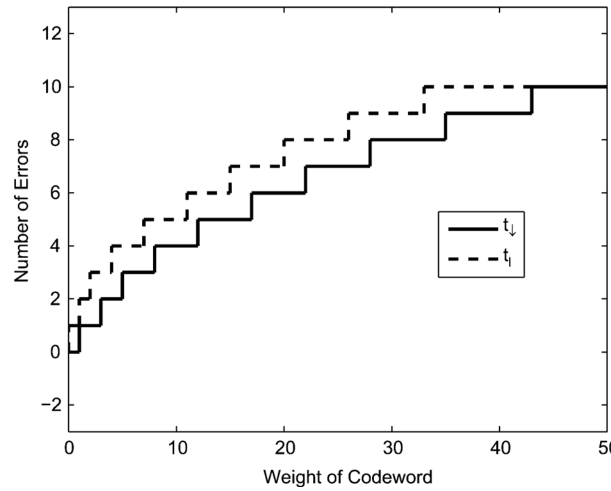

Fig. 5. Demonstration of functions $t_\downarrow$ and t_l.

such that $\mathbf{x}, \mathbf{y} \geq \mathbf{z}$ and $N(\mathbf{x}, \mathbf{z}) \leq t_l(w(\mathbf{y}))$ and $N(\mathbf{y}, \mathbf{z}) \leq t_l(w(\mathbf{y}))$. Since $w(\mathbf{x}) - t_\downarrow(w(\mathbf{x})) \leq w(\mathbf{y})$, according to the definition of t_l, it is easy to get $t_l(w(\mathbf{y})) \geq t_\downarrow(w(\mathbf{x})) \geq t_\downarrow(w(\mathbf{y}))$. So there does not exist a word $\mathbf{z} \in \{0,1\}^n$ such that $\mathbf{x}, \mathbf{y} \geq \mathbf{z}$ and $N(\mathbf{x}, \mathbf{z}) \leq t_\downarrow(w(\mathbf{x}))$ and $N(\mathbf{y}, \mathbf{z}) \leq t_\downarrow(w(\mathbf{y}))$, namely, $\mathcal{B}(\mathbf{x}) \bigcap \mathcal{B}(\mathbf{y}) = \phi$.

This completes the proof. □

We see that the constructions of layered codes are based on the provided group of codes $\mathcal{C}_1, \ldots, \mathcal{C}_k$ such that $\mathcal{C}_1 \supset \mathcal{C}_2 \supset \cdots \supset \mathcal{C}_k$ and for $1 \leq t \leq k$, and the code $\mathcal{C}_t$ corrects t asymmetric errors. Examples of such codes include Varshamov codes [24], Bose–Chaudhuri–Hocquenghem (BCH) codes, etc.

The construction of Varshamov codes can be described as follows. Let $\alpha_1, \alpha_2, \ldots, \alpha_n$ be distinct nonzero elements of F_q, and let $\alpha := (\alpha_1, \alpha_2, \ldots, \alpha_n)$. For $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \{0,1\}^n$, let $\mathbf{x}\alpha = (x_1\alpha_1, x_2\alpha_2, \ldots, x_n\alpha_n)$. For $g_1, g_2, \ldots, g_t \in F_q$ and $1 \leq t \leq k$, let

$$\mathcal{C}_t := \{\mathbf{x} \in \{0,1\}^n | \sigma_l(\mathbf{x}\alpha) = g_l \text{for } 1 \leq l \leq t\}$$

where the elementary symmetric function $\sigma_l(\mathbf{u})$ for $l \geq 0$ are defined by

$$\prod_{i=1}^{r}(z + u_i) = \sum_{l=0}^{\infty} \sigma_l(\mathbf{u}) z^{r-l}.$$

Then, $\mathcal{C}_t$ can correct t asymmetric errors (for $1 \leq t \leq k$), and $\mathcal{C}_1 \supset \mathcal{C}_2 \supset \cdots \supset \mathcal{C}_k$.

Such a group of codes can also be constructed by BCH codes: Let $(\alpha_0, \alpha_1, \ldots, \alpha_{n-1})$ be n distinct nonzero elements of G_{2^m} with $n = 2^m - 1$. For $1 \leq t \leq k$, let

$$\mathcal{C}_t := \{\mathbf{x} \in \{0,1\}^n | \sum_{i=1}^{n} x_i \alpha_i^{(2l-1)} = 0 \text{for } 1 \leq l \leq t\}.$$

C. Decoding Algorithm

Assume $\mathbf{x}$ is a codeword in $\mathcal{C}_t$ and $\mathbf{y} = \mathbf{x} + \mathbf{e}$ is a received erroneous word with error vector e; then, there is an efficient algorithm to decode $\mathbf{y}$ into a codeword, which is denoted by

$D_t(\mathbf{y})$. If $\mathbf{y}$ has at most t asymmetric errors, then $D_t(\mathbf{y}) = \mathbf{x}$. We show that the layered codes proposed above also have an efficient decoding algorithm if $D_t(\cdot)$ (for $1 \leq t \leq k$) are provided and efficient.

Theorem 15: Let $\mathcal{C}$ be a layered code based on the aforementioned construction, and let $\mathbf{y} = \mathbf{x} + \mathbf{e}$ be a received word such that $\mathbf{x} \in \mathcal{C}$ and $|e| \leq t_\downarrow(w(\mathbf{x}))$. To recover $\mathbf{x}$ from $\mathbf{y}$, we enumerate the integers in $[t_l(w(\mathbf{y})), t_l(w(\mathbf{y}) + t_l(w(\mathbf{y})))]$. If we can find an integer t such that $D_t(\mathbf{y}) \in \mathcal{C}$ and $N(D_t(\mathbf{y}), \mathbf{y}) \leq t_\downarrow(w(D_t(\mathbf{y})))$, then $D_t(\mathbf{y}) = \mathbf{x}$.

Proof: If we let $t = t_\downarrow(w(\mathbf{x}))$, then we can get that t satisfies the conditions and $D_t(\mathbf{y}) = \mathbf{x}$. So such t exists.

Now we only need to prove that once there exists t satisfying the conditions in the theorem, we have $D_t(\mathbf{y}) = \mathbf{x}$. We prove this by contradiction. Assume there exists t satisfying the conditions but $\mathbf{z} = D_t(\mathbf{y}) \neq \mathbf{x}$. Then, $N(\mathbf{z}, \mathbf{y}) \leq t_\downarrow(w(\mathbf{z}))$. Since we also have $N(\mathbf{x}, \mathbf{y}) \leq t(w(\mathbf{x}))$, $\mathcal{B}(\mathbf{x}) \bigcap \mathcal{B}(\mathbf{z}) \neq \phi$, which contradicts the property of the layered codes.

This completes the proof. □

In the aforementioned method, to decode an erroneous word $\mathbf{y}$, we can check all the integers between $t_l(w(\mathbf{y}))$ and $t_l(w(\mathbf{y}) + t_l(w(\mathbf{y})))$ to find the value of t. Once we find the integer t satisfying the conditions in the theorem, we can decode $\mathbf{y}$ into $D_t(\mathbf{y})$ directly. (Note that the length of the interval for t, namely $t_l(w(\mathbf{y}) + t_l(w(\mathbf{y}))) - t_l(w(\mathbf{y}))$, is normally much smaller than $w(\mathbf{y})$. It is approximately $\frac{p^2}{(1-p)^2} w(\mathbf{y})$ for i.i.d. errors when $w(\mathbf{y})$ is large.) We see that this decoding process is efficient if $D_t(.)$ is efficient for $1 \leq t \leq k$.

D. Layered Versus Uniform

Typically, nonlinear codes, like Varshamov codes, are superior to BCH codes. But it is still not well known how to estimate the sizes of Varshamov codes and their weight distributions. To compare uniform constructions and nonuniform constructions for correcting asymmetric errors, we focus on BCH codes, namely, we compare normal BCH codes with layered BCH codes. Here, we consider i.i.d. errors, and we assume that the codeword length is $n = 255$, the crossover probability is p, and the maximal tolerated error probability is q_e.

Table II shows the relations between the dimension k and the number of errors t that can be corrected in BCH codes when $n = 255$. According to [16], many BCH codes have approximated binomial weight distribution. So given an $(255, k, t)$ BCH code, the number of codewords of weight i is approximately

$$b_i \sim 2^k \frac{\binom{n}{i}}{2^n}.$$

For a normal BCH code, it has to correct t errors with

$$t = \min\{s \in N | \sum_{i=0}^{s} \binom{n}{i} p^i (1-p)^{n-i} \geq 1 - q_e\}.$$

Then it has 2^k codewords where k can be obtained from Table II based on the value of t.

For a layered BCH code, the codewords with Hamming weight w have to correct $t_\downarrow(w)$ asymmetric errors such that

$$t_\downarrow(w) = \min\{s \in N | \sum_{i=0}^{s} \binom{w}{i} p^i (1-p)^{w-i} \geq 1 - q_e\}$$

TABLE II
BCH CODES WITH CODEWORD LENGTH 255 [11]

n	k	t	n	k	t
255	247	1	255	115	21
255	239	2	255	107	22
255	231	3	255	99	23
255	223	4	255	91	25
255	215	5	255	87	26
255	207	6	255	79	27
255	199	7	255	71	29
255	191	8	255	63	30
255	187	9	255	55	31
255	179	10	255	47	42
255	171	11	255	45	43
255	163	12	255	37	45
255	155	13	255	29	47
255	147	14	255	21	55
255	139	15	255	13	59
255	131	18	255	9	63
255	123	19			

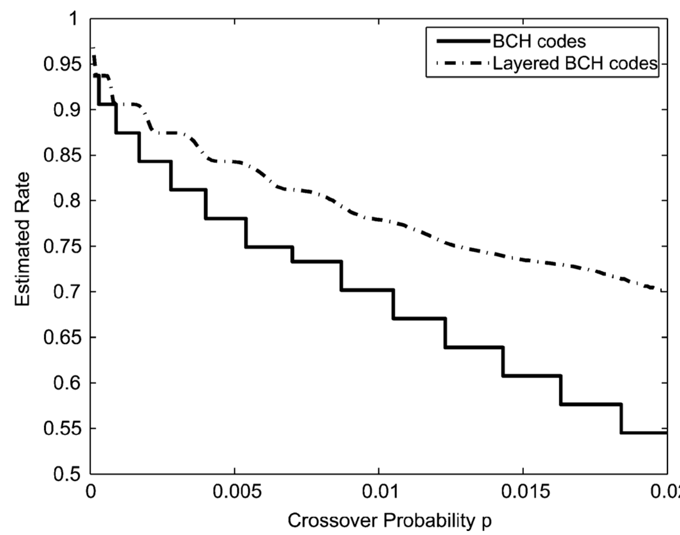

Fig. 6. Estimated rates of BCH codes and layered BCH codes when $n = 255$ and $q_e = 10^{-4}$.

for all $0 \leq w \leq n$. Based on the approximated weight distribution of BCH codes, the number of codewords in a layered BCH codes can be estimated by summing up the numbers of codewords with different weights.

Fig. 6 plots the estimated rates of BCH codes and layered BCH codes for different p when $n = 255$ and $q_e = 10^{-4}$. Here, for a code $\mathcal{C}$, let $\#\mathcal{C}$ be the number of codewords; then, the rate of $\mathcal{C}$ is defined as $\frac{\log_2(\#\mathcal{C})}{n}$. From this figure, we see that under the same parameters (n, p, q_e), the rates of layered BCH codes are much higher than those of BCH codes. By constructing nonuniform codes instead of uniform codes, the code rate can be significantly increased. Comparing Fig. 6 with Fig. 3, it can be seen that the rates of layered BCH codes are very close to the upper bounds of uniform codes. It implies that we can gain more by considering nonuniform codes rather than nonlinear uniform codes.

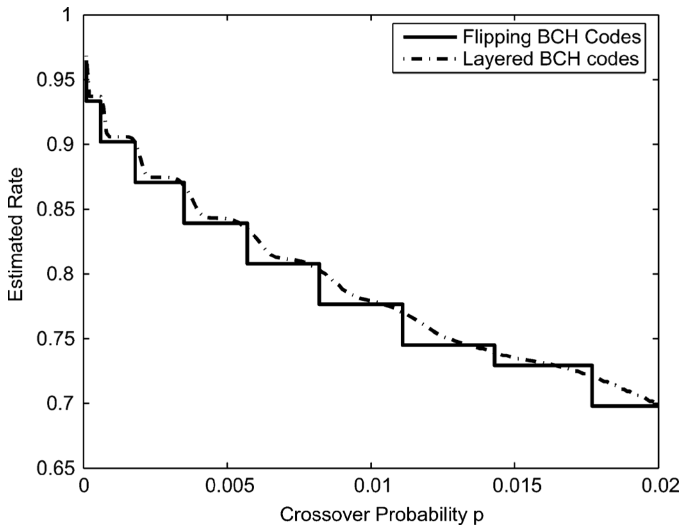

Fig. 7. Estimated rates of flipping/layered BCH codes when $n = 255$ and $q_e = 10^{-4}$.

VI. Flipping Codes Construction

Many nonlinear codes designed to correct asymmetric errors like Varshamov codes are superior to linear codes. However, they do not yet have efficient encoding algorithms, namely, it is not easy to find an efficient encoding function $f : \{0,1\}^k \to \mathcal{C}$ with $k \doteq \lfloor \log |\mathcal{C}| \rfloor$. In this section, we focus on the approach of designing nonuniform codes for asymmetric errors with efficient encoding schemes, by utilizing the well-studied linear codes.

A simple method is that we can use a linear code to correct $t_\downarrow(n)$ asymmetric errors directly, but this method is inefficient not only because the decoding sphere for symmetric errors is greater than the sphere for asymmetric errors (and therefore an overkill), but also because for low-weight codewords, the number of asymmetric errors they need to correct can be much smaller than $t_\downarrow(n)$.

Our idea is to build a *flipping code* that uses only low-weight codewords (specifically, codewords of Hamming weight no more than $\sim \frac{n}{2}$), because they need to correct fewer asymmetric errors and therefore can increase the code's rate. In the rest of this section, we present two different constructions.

A. First Construction

First, we construct a linear code $\mathcal{C}$ (like BCH codes) of length n with generator matrix G that corrects $t_\downarrow(\lfloor \frac{n}{2} \rfloor)$ symmetric errors. Assume the dimension of the code is k. For any binary message $\mathbf{u} \in \{0,1\}^k$, we can map it to a codeword $\mathbf{x}$ in $\mathcal{C}$ such that $\mathbf{x} = \mathbf{u}G$. Next, let $\overline{\mathbf{x}}$ denote a word obtained by flipping all the bits in $\mathbf{x}$ such that if $x_i = 0$, then $\overline{x}_i = 1$ and if $x_i = 1$, then $\overline{x}_i = 0$; and let $\mathbf{y}$ denote the final codeword corresponding to $\mathbf{u}$. We check whether $w(\mathbf{x}) \leq \lfloor \frac{n}{2} \rfloor$ and construct $\mathbf{y}$ in the following way:

$$\mathbf{y} = \begin{cases} \mathbf{x}00\ldots0, & \text{if } w(\mathbf{x}) \leq \lfloor \frac{n}{2} \rfloor \\ \overline{\mathbf{x}}11\ldots1, & \text{otherwise.} \end{cases}$$

Here, the auxiliary bits (0s or 1s) are added to distinguish that whether $\mathbf{x}$ has been flipped or not, and they form a repetition code to tolerate errors.

The corresponding decoding process is straightforward. Assume we received a word $\mathbf{y}'$. If there is at least one 1 in the auxiliary bits, then we "flip" the word by changing all 0s to 1s and all 1s to 0s; otherwise, we keep the word unchanged. Then, we apply the decoding scheme of the code $\mathcal{C}$ to the first n bits of the word. Finally, the message $\mathbf{u}$ can be successfully decoded if $\mathbf{y}'$ has at most $t_\downarrow(\lfloor \frac{n}{2} \rfloor)$ errors in the first n bits.

B. Second Construction

In the previous construction, several auxiliary bits are needed to protect one bit of information, which is not very efficient. Here, we try to move this bit into the information part of the codewords in $\mathcal{C}$. This motivates us to give the following construction.

Let $\mathcal{C}$ be a systematic linear code with length n that corrects t' symmetric errors (we will specify t' later). Assume the dimension of the code is k. Now, for any binary message $\mathbf{u} \in \{0,1\}^{k-1}$ of length $k-1$, we get $\mathbf{u}' = 0\mathbf{u}$ by adding one bit 0 in front of $\mathbf{u}$. Then, we can map $\mathbf{u}'$ to a codeword $\mathbf{x}$ in $\mathcal{C}$ such that

$$\mathbf{x} = (0\mathbf{u})G = 0\mathbf{u}\mathbf{v}$$

where G is the generator matrix of $\mathcal{C}$ in systematic form and the length of $\mathbf{v}$ is $n-k$. Let $\boldsymbol{\alpha}$ be a codeword in $\mathcal{C}$ such that the first bit $\alpha_1 = 1$ and its weight is the maximal one among all the codeword in $\mathcal{C}$, i.e.,

$$\boldsymbol{\alpha} = \arg \max_{\mathbf{x} \in \mathcal{C}, x_1 = 1} w(\mathbf{x}).$$

Generally, $w(\boldsymbol{\alpha})$ is very close to n. For example, in any primitive BCH code of length 255, $\boldsymbol{\alpha}$ is the all-one vector; also we can construct LDPC codes that include the all-one vector as long as their parity-check matrices have even number of ones in each column. In order to reduce the weights of the codewords, we use the following operations. Calculate the relative weight

$$w(\mathbf{x}|\boldsymbol{\alpha}) = |\{1 \leq i \leq n | x_i = 1, \alpha_i = 1\}|.$$

Then, we get the final codeword

$$\mathbf{y} = \begin{cases} \mathbf{x} + \boldsymbol{\alpha}, & \text{if } w\ (\mathbf{x}|\boldsymbol{\alpha}) > \frac{w(\boldsymbol{\alpha})}{2} \\ \mathbf{x}, & \text{otherwise} \end{cases}$$

where $+$ is the binary sum, so $\mathbf{x} + \boldsymbol{\alpha}$ is to flip the bits in $\mathbf{x}$ corresponding the ones in $\boldsymbol{\alpha}$. So far, we see that the maximal weight for $\mathbf{y}$ is $\lfloor n - \frac{w(\boldsymbol{\alpha})}{2} \rfloor$. That means we need to select t' such that

$$t' = t_\downarrow(\lfloor n - \frac{w(\boldsymbol{\alpha})}{2} \rfloor).$$

For many linear codes, α is the all-one vector, so $t' = t_\downarrow(\lfloor \frac{n}{2} \rfloor)$.

In the aforementioned encoding process, for different binary messages, they have different codewords. And for any codeword $\mathbf{y}$, we have $\mathbf{y} \in \mathcal{C}$. That is because either $\mathbf{y} = \mathbf{x}$ or

$\mathbf{y} = \mathbf{x} + \boldsymbol{\alpha}$, where both $\mathbf{x}$ and $\boldsymbol{\alpha}$ are codewords in $\mathcal{C}$ and $\mathcal{C}$, is a linear code. So the resulting flipping code is a subset of code $\mathcal{C}$.

The decoding process is very simple. Given the received word $\mathbf{y}' = \mathbf{y} + \mathbf{e}$, we can always get $\mathbf{y}$ by applying the decoding scheme of the linear code $\mathcal{C}$ if $|\mathbf{e}| \leq t'$. If $y_1 = 1$, that means $\mathbf{x}$ has been flipped based on $\boldsymbol{\alpha}$, so we have $\mathbf{x} = \mathbf{y} + \boldsymbol{\alpha}$; otherwise, $\mathbf{x} = \mathbf{y}$. Then, the initial message $\mathbf{u} = x_2x_3\ldots x_k$.

We see that the second construction is a little more efficient than the first one, by moving the bit that indicates flips from the outside of a codeword (of an error-correcting code) to the inside. Here is an example of the second construction. Let $\mathcal{C}$ be the $(7,4)$ Hamming code, which is able to correct single-bit errors. The generating matrix of the $(7,4)$ Hamming code is

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}.$$

Here, we have $t' = 1$ and $k = 4$. Assume the binary message is $\mathbf{u} = 011$; then, we have $\mathbf{x} = (0\mathbf{u})G = 0011100$. It is easy to see that $\boldsymbol{\alpha}$ is the all-one codeword, i.e., $\boldsymbol{\alpha} = 1111111$. In this case, $w(\mathbf{x}|\boldsymbol{\alpha}) <= \frac{w(\boldsymbol{\alpha})}{2}$, so the final codeword $\mathbf{y} = 0011100$. Assume the binary message is $\mathbf{u} = 110$; then, we have $\mathbf{x} = (0\mathbf{u})G = 0110110$. In this case, $w(\mathbf{x}|\boldsymbol{\alpha}) > \frac{w(\boldsymbol{\alpha})}{2}$, so the final codeword $\mathbf{y} = \mathbf{x} + \boldsymbol{\alpha} = 1001001$.

Assume the received word is $\mathbf{y}' = 0001001$. By applying the decoding algorithm of Hamming codes, we get $\mathbf{y} = 1001001$. Since $y_1 = 1$, we have $\mathbf{x} = \mathbf{y} + \boldsymbol{\alpha}$, and as a result, $\mathbf{u} = 110$.

C. Flipping Versus Layered

When n is sufficiently large, the aforementioned flipping codes become nearly as efficient (in terms of code rate) as a linear codes correcting $t_\downarrow(\lfloor\frac{n}{2}\rfloor)$ symmetric errors. It is much more efficient than designing a linear code correcting $t_\downarrow(n)$ symmetric errors. Note that when n is large and p is small, these codes can have very good performance on code rate. That is because when n is sufficiently large, the rate of an optimal nonuniform code is dominated by the codewords with the same Hamming weight $w_d(\leq \frac{n}{2})$, and w_d approaches $\frac{n}{2}$ as p gets close to 0. We can intuitively understand it based on two facts when n is sufficiently large: 1) there are at most $n2^{n(H(\frac{w_d}{n})+\delta)}$ codewords in this optimal nonuniform code. 2) When p becomes small, we can get a nonuniform code with at least $2^{n(1-\delta)}$ codewords. So when n is sufficiently large and p is small, we have $w_d \to \frac{n}{2}$. Hence, an optimal nonuniform code has almost the same asymptotic performance with an optimal weight-bounded code (Hamming weight is at most *n*/2) that corrects $t_\downarrow(n/2)$ asymmetric errors.

Let us consider a flipping BCH code based on the second construction. Similar as the previous section, we assume that the codeword length is $n = 255$ and the number of codewords with weight i can be approximated by

$$2^k\frac{\binom{n}{i}}{2^n}$$

where k is the dimension of the code. Fig. 7 compares the estimated rates of flipping BCH codes and those of layered BCH codes when $n = 255$ and $q_e = 10^{-4}$. Surprisingly, the flipping BCH codes achieve almost the same rates as layered BCH codes. Note that, for the layered codes, we are able to further improve the efficiency (rates) by replacing BCH codes with Varshamov codes, i.e., based on layered Varshamov codes.

VII. Extension to Binary Asymmetric Channels

In the previous sections, we have introduced and studied nonuniform codes for Z-channels. The concept of nonuniform codes can be extended from Z-channels to general binary asymmetric channels, where the error probability from 0 to 1 is smaller than the error probability from 1 to 0 but it may not be ignorable. In this case, we are able to construct nonuniform codes correcting a big number of $1 \to 0$ errors and a small number of $0 \to 1$ errors. Such codes can be used in flash memories or phase change memories, where the change in data has an asymmetric property. For example, the stored data in flash memories is represented by the voltage levels of transistors, which drift in one direction because of charge leakage. In phase change memories, another class of nonvolatile memories, the stored data is determined by the electrical resistance of the cells, which also drifts due to thermally activated crystallization of the amorphous material. This asymmetric property will introduce more $1 \to 0$ errors than $0 \to 1$ errors after a long duration.

In this section, we first investigate binary asymmetric channels where the probability from 0 to 1 is much smaller than that from 1 to 0, namely, $p_\uparrow \ll p_\downarrow$, but $p_\uparrow$ is not ignorable. In this case, $t_\uparrow$ is very small, we would not gain much by letting $t_\uparrow$ be a variable. Instead, for simplicity, we let $t_\uparrow$ be a constant function. Later, we consider general binary asymmetric channels, where $t_\uparrow$ can be an arbitrary nonincreasing step function.

A. $t_\uparrow$ Is a Constant Function

We show that if $t_\uparrow$ is a constant function, then correcting $[t_\downarrow, t_\uparrow]$ errors is equivalent to correcting $t_\downarrow + t_\uparrow$ asymmetric errors, where $t_\downarrow$ can be an arbitrary step functions on $\{0, 1, \ldots, n\}$.

The following theorem extends Theorem 2 in [18].

Theorem 16: Let $t_\uparrow$ be a constant function; a code $\mathcal{C}$ is a nonuniform code correcting $[t_\downarrow, t_\uparrow]$ errors if and only if it is a nonuniform code correcting $t_\downarrow + t_\uparrow$ asymmetric errors.

Proof: 1) We first show that if $\mathcal{C}$ is a nonuniform code correcting $[t_\downarrow, t_\uparrow]$ errors where $t_\uparrow$ is a constant function, then it can correct $t_\downarrow + t_\uparrow$ asymmetric errors. We need to prove that there does not exist a pair of codewords $\mathbf{x}, \mathbf{y} \in \mathcal{C}$ such that

$$N(\mathbf{x},\mathbf{y}) \leq t_\downarrow(w(\mathbf{x})) + t_\uparrow$$
$$N(\mathbf{y},\mathbf{x}) \leq t_\downarrow(w(\mathbf{y})) + t_\uparrow$$

where

$$N(\mathbf{x},\mathbf{y}) \stackrel{\triangle}{=} |\{i : x_i = 1, y_i = 0\}|.$$

Let us prove it by contradiction. Assume that their exists a pair of codewords $\mathbf{x}, \mathbf{y}$ that satisfy the aforementioned inequalities. By adding at most $t_\uparrow$ $0 \to 1$ errors, we get a vector $\mathbf{x}'$ from $\mathbf{x}$ such that the Hamming distance between $\mathbf{x}'$ and $\mathbf{y}$ is

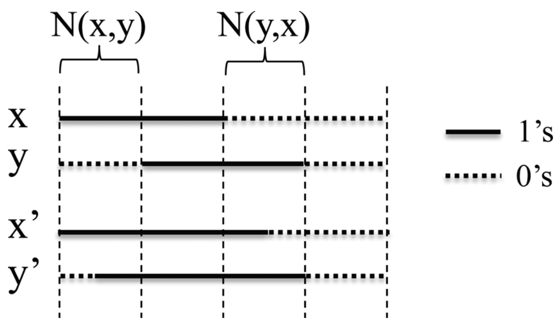

Fig. 8. Demonstration of $\mathbf{x}, \mathbf{y}, \mathbf{x}', \mathbf{y}'$.

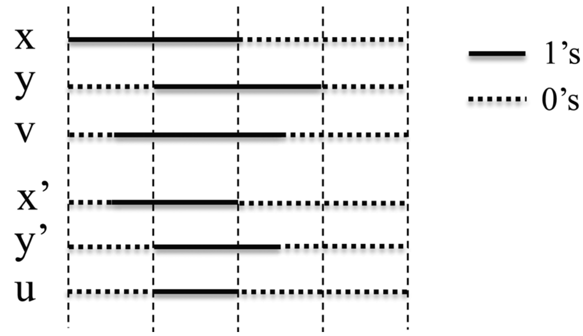

Fig. 9. Demonstration of $\mathbf{x}, \mathbf{y}, \mathbf{x}', \mathbf{y}', \mathbf{v}, \mathbf{u}$.

minimized; also we get a vector $\mathbf{y}'$ from $\mathbf{y}$ such that the Hamming distance between $\mathbf{y}'$ and $\mathbf{x}$ is minimized. In this case, we only need to show that

$$N(\mathbf{x}', \mathbf{y}') \leq t_\downarrow(w(\mathbf{x})), N(\mathbf{y}', \mathbf{x}') \leq t_\downarrow(w(\mathbf{y}))$$

which contradicts with our assumption that $\mathcal{C}$ can correct $[t_\downarrow, t_\uparrow]$ errors. The intuitive way of understanding $\mathbf{x}', \mathbf{y}'$ is shown in Fig. 8. In the figure, we present each vector as a line, in which the solid part is for 1s and the dashed part is for 0s.

If $N(\mathbf{x}', \mathbf{x}) < t_\uparrow$ and $N(\mathbf{y}', \mathbf{y}) < t_\uparrow$, then

$$x'_i = \max(x_i, y_i) = y'_i$$

so $\mathbf{x}' = \mathbf{y}'$. The statement is true.

If $N(\mathbf{x}', \mathbf{x}) < t_\uparrow$ and $N(\mathbf{y}', \mathbf{y}) = t_\uparrow$, then $\mathbf{y}' \leq \mathbf{x}'$. In this case

$$N(\mathbf{x}', \mathbf{y}') \leq N(\mathbf{x}, \mathbf{y}) - t_\uparrow \leq t_\downarrow(w(\mathbf{x})).$$

We get the statement.

Similarly, if $N(\mathbf{y}', \mathbf{y}) < t_\uparrow$ and $N(\mathbf{x}', \mathbf{x}) = t_\uparrow$, we have $\mathbf{x}' \leq \mathbf{y}'$ and

$$N(\mathbf{y}', \mathbf{x}') \leq N(\mathbf{y}, \mathbf{x}) - t_\uparrow \leq t_\downarrow(w(\mathbf{y})).$$

If $N(\mathbf{x}', \mathbf{x}) = t_\uparrow$ and $N(\mathbf{y}', \mathbf{y}) = t_\uparrow$, we can get

$$\begin{aligned} N(\mathbf{x}', \mathbf{y}') &\leq N(\mathbf{x}, \mathbf{y}) - t_\uparrow \leq t_\downarrow(w(\mathbf{x})) \\ N(\mathbf{y}', \mathbf{x}') &\leq N(\mathbf{y}, \mathbf{x}) - t_\uparrow \leq t_\downarrow(w(\mathbf{y})). \end{aligned}$$

Based on the aforementioned discussions, we can conclude that if $\mathcal{C}$ is a nonuniform code correcting $[t_\downarrow, t_\uparrow]$ errors where $t_\uparrow$ is a constant function, then it is also a nonuniform code correcting $t_\downarrow + t_\uparrow$ asymmetric errors.

2) We show that if $\mathcal{C}$ is a nonuniform code correcting $t_\downarrow + t_\uparrow$ asymmetric errors where $t_\uparrow$ is a constant function, then it is also a nonuniform code correcting $[t_\downarrow, t_\uparrow]$ errors. That means for any $\mathbf{x}, \mathbf{y} \in \mathcal{C}$, there does not exist a vector $\mathbf{v}$ such that

$$\begin{aligned} N(\mathbf{v}, \mathbf{x}) &\leq t_\uparrow, \quad N(\mathbf{x}, \mathbf{v}) \leq t_\downarrow(w(\mathbf{x})) \\ N(\mathbf{v}, \mathbf{y}) &\leq t_\uparrow, \quad N(\mathbf{y}, \mathbf{v}) \leq t_\downarrow(w(\mathbf{y})). \end{aligned}$$

Let us prove this by contradiction. We assume there exists a vector $\mathbf{v}$ satisfies the aforementioned conditions. Now, we define a few vectors $\mathbf{x}', \mathbf{y}', \mathbf{u}$ such that

$$\begin{aligned} x'_i &= \min(x_i, v_i) \quad \forall 1 \leq i \leq n \\ y'_i &= \min(y_i, v_i) \quad \forall 1 \leq i \leq n \\ u_i &= \min(x_i, y_i, v_i) \quad \forall 1 \leq i \leq n. \end{aligned}$$

The intuitive way of understanding these vectors is shown in Fig. 9. In the figure, we present each vector as a line, in which the solid part is for 1s and the dashed part is for 0s.

Then

$$\begin{aligned} &\mathbf{x}' \leq \mathbf{x}, \mathbf{x}' \leq \mathbf{v}, N(\mathbf{x}, \mathbf{x}') \leq t_\downarrow(w(\mathbf{x})), N(\mathbf{v}, \mathbf{x}') \leq t_\uparrow \\ &\mathbf{y}' \leq \mathbf{y}, \mathbf{y}' \leq \mathbf{v}, N(\mathbf{y}, \mathbf{y}') \leq t_\downarrow(w(\mathbf{y})), N(\mathbf{v}, \mathbf{y}') \leq t_\uparrow. \end{aligned}$$

Now we want to show that

$$N(\mathbf{x}, \mathbf{u}) \leq t_\downarrow(w(\mathbf{x})) + t_\uparrow.$$

Since

$$N(\mathbf{x}, \mathbf{u}) \leq N(\mathbf{x}, \mathbf{x}') + N(\mathbf{x}\prime, \mathbf{u})$$

we only need to show that

$$N(\mathbf{x}\prime, \mathbf{u}) \leq t_\uparrow.$$

According to the definition of $\mathbf{u}$, it is easy to get that

$$\begin{aligned} N(\mathbf{v}, \mathbf{x}') + N(\mathbf{x}', \mathbf{u}) &= N(\mathbf{v}, \mathbf{y}') + N(\mathbf{y}', \mathbf{u}) \\ &\leq N(\mathbf{v}, \mathbf{x}') + N(\mathbf{v}, \mathbf{y}'). \end{aligned}$$

So $N(\mathbf{x}', \mathbf{u}) \leq t_\uparrow$, which leads us to

$$N(\mathbf{x}, \mathbf{u}) \leq t_\downarrow(w(\mathbf{x})) + t_\uparrow.$$

Similarly, we can also get

$$N(\mathbf{y}, \mathbf{u}) \leq t_\downarrow(w(\mathbf{y})) + t_\uparrow.$$

In this case, $\mathcal{C}$ is not a nonuniform code correcting $t_\downarrow + t_\uparrow$ asymmetric errors, which contradicts with our assumption.

Based on the aforementioned discussions, we can get the conclusion in the theorem. $\square$

According to the aforementioned theorem, if $t_\uparrow$ is a constant function, the upper bound of nonuniform codes correcting $[t_\downarrow, t_\uparrow]$ errors is exactly the upper bound of nonuniform code correcting $t_\downarrow + t_\uparrow$ asymmetric $(1 \to 0)$ errors. To construct a nonuniform code correcting $[t_\downarrow, t_\uparrow]$ errors, it is equivalent to construct a nonuniform code correcting $t_\downarrow + t_\uparrow$ asymmetric $(1 \to 0)$ errors. Hence, our code constructions in Sections V and VI can be applied.

B. $t_\uparrow$ Is a Nonincreasing Function

Another case of binary asymmetric channel is that $p_\uparrow < p_\downarrow$ but $p_\uparrow$ is not much smaller than $p_\downarrow$. In this case, it is not efficient to write $t_\uparrow$ as a constant function. Instead, we consider it as a nonincreasing step function.

Theorem 17: Let $t_\downarrow$ be a nondecreasing function and $t_\uparrow$ be a nonincreasing function. A code $\mathcal{C}$ is a nonuniform code correcting $[t_\downarrow, t_\uparrow]$ errors if it is a nonuniform code correcting $t_\downarrow + \overline{t_\uparrow}$ asymmetric errors. Here, for all $0 \le w \le n$

$$\overline{t_\uparrow}(w) = t_\uparrow(\max\{s | t_\uparrow(s) + s \le w - t_\downarrow(w)\}).$$

Proof: Let $\mathcal{C}$ be a nonuniform code correcting $t_\downarrow + \overline{t_\uparrow}$ errors. For any $\mathbf{x}, \mathbf{y} \in \mathcal{C}$, w.l.o.g, we assume $w(\mathbf{x}) \le w(\mathbf{y})$. If $w(\mathbf{x}) + t_\uparrow(w(\mathbf{x})) < w(\mathbf{y}) - t_\downarrow(w(\mathbf{y}))$, then there does not exist a vector $\mathbf{v}$ such that

$$N(\mathbf{v}, \mathbf{x}) \le t_\uparrow, \quad N(\mathbf{x}, \mathbf{v}) \le t_\downarrow(w(\mathbf{x}))$$
$$N(\mathbf{v}, \mathbf{y}) \le t_\uparrow, \quad N(\mathbf{y}, \mathbf{v}) \le t_\downarrow(w(\mathbf{y})).$$

If $w(\mathbf{x}) + t_\uparrow(w(\mathbf{x})) \ge w(\mathbf{y}) - t_\downarrow(w(\mathbf{y}))$, according to the proof in Theorem 16, we can get that there does not exist a vector $\mathbf{v}$ such that

$$N(\mathbf{v}, \mathbf{x}) \le t_\uparrow(w(\mathbf{x}))$$
$$N(\mathbf{x}, \mathbf{v}) \le t_\downarrow(w(\mathbf{x})) + \overline{t_\uparrow}(w(\mathbf{x})) - t_\uparrow(w(\mathbf{x}))$$
$$N(\mathbf{v}, \mathbf{y}) \le t_\uparrow(w(\mathbf{x}))$$
$$N(\mathbf{y}, \mathbf{v}) \le t_\downarrow(w(\mathbf{y})) + \overline{t_\uparrow}(w(\mathbf{y})) - t_\uparrow(w(\mathbf{x})).$$

Since

$$\overline{t_\uparrow}(w(\mathbf{x})) - t_\uparrow(w(\mathbf{x})) \ge 0$$
$$t_\uparrow(w(\mathbf{x})) \ge t_\uparrow(w(\mathbf{y}))$$
$$\overline{t_\uparrow}(w(\mathbf{y})) \ge t_\uparrow(w(\mathbf{x}))$$

we can get that there does not exist a vector $\mathbf{v}$ such that

$$N(\mathbf{v}, \mathbf{x}) \le t_\uparrow, \quad N(\mathbf{x}, \mathbf{v}) \le t_\downarrow(w(\mathbf{x}))$$
$$N(\mathbf{v}, \mathbf{y}) \le t_\uparrow, \quad N(\mathbf{y}, \mathbf{v}) \le t_\downarrow(w(\mathbf{y})).$$

Finally, we conclude that $\mathcal{C}$ is a nonuniform code correcting $[t_\downarrow, t_\uparrow]$ errors. □

According to the aforementioned theorem, to construct a nonuniform code correcting $[t_\downarrow, t_\uparrow]$ errors, instead, we can construct a nonuniform code correcting $t_\downarrow + \overline{t_\uparrow}$ asymmetric errors. So the problem of constructing a nonuniform code for an arbitrary binary asymmetric channel is converted to the problem of constructing a nonuniform correcting for a Z-channel. Note that this conversion results in a little loss of code efficiency, but typically it is very small. Both layered codes and flipping codes can be applied for correcting errors in binary asymmetric channels. A little point to notice is that $t_\downarrow + \overline{t_\uparrow}$ might not be a strict nondecreasing function of codeword weight. In this case, we can find a nondecreasing function t_h which is slightly larger than $t_\downarrow + \overline{t_\uparrow}$, and construct a nonuniform code correcting t_h asymmetric errors.

When we apply flipping codes for correcting errors in binary asymmetric channels, we do not have to specify $t_\downarrow$ and $t_\uparrow$ separately. For example, assume that i.i.d. errors are considered. If the maximal tolerated error probability is q_e, then given a codeword of weight w, it has to tolerate total $t_f(w)$ errors. For $0 \le w \le n$, $t_f(w)$ can be obtained by calculating the minimal integer t such that

$$\sum_{i=0}^{t} \sum_{j=0}^{t-i} \binom{w}{i} \binom{n-w}{j} p_\downarrow^i (1-p_\downarrow)^{w-i} p_\uparrow^j (1-p_\uparrow)^{(n-w-j)} \ge 1 - q_e.$$

To construct a flipping code, we only need to find a linear code such that it corrects $t_f(\lfloor n - \frac{\alpha}{2} \rfloor)$ symmetric errors, where α is the codeword with the maximum weight in the linear code.

Theorem 18: Let $t_\downarrow$ be a nondecreasing function and $t_\uparrow$ be a nonincreasing function. If a code $\mathcal{C}$ is a nonuniform code correcting $[t_\downarrow, t_\uparrow]$ errors, then it corrects $t_\downarrow + \underline{t_\uparrow}$ asymmetric errors. Here

$$\underline{t_\uparrow}(w) = t_\uparrow(\min\{s | s - t_\uparrow(s) - t_\downarrow(s) \le w\}).$$

Proof: The proof of this theorem is very similar as the proof for the previous theorem. It follows the conclusion in Theorem 16. □

According to the aforementioned theorem, to calculate the upper bound of nonuniform codes correcting $[t_\downarrow, t_\uparrow]$ errors, we can first calculate the upper bound of nonuniform codes correcting $t_\downarrow + \underline{t_\uparrow}$ asymmetric errors. Generally speaking, nonuniform codes correcting $[t_\downarrow, t_\uparrow]$ errors (considering the optimal case) are more efficient than nonuniform codes correcting $t_\downarrow + \overline{t_\uparrow}$ asymmetric errors, but less efficient than those correcting $t_\downarrow + \underline{t_\uparrow}$ asymmetric errors. According to the definitions of $\underline{t_\uparrow}$ and $\overline{t_\uparrow}(w)$, it is easy to get that

$$\underline{t_\uparrow}(w) \le t_\uparrow(w) \le \overline{t_\uparrow}(w)$$

for $0 \le w \le n$. Typically, if $p_\downarrow, p_\uparrow \ll 1$, then $\overline{t_\uparrow}(w) - \underline{t_\uparrow}(w) \ll t_\uparrow(w)$. It implies that nonuniform codes correcting $[t_\downarrow, t_\uparrow]$ errors are roughly as efficient as those correcting $t_\downarrow + t_\uparrow$ asymmetric errors. If we consider i.i.d. errors and long codewords, it is equally difficult to correct errors introduced by a binary asymmetric channel with crossover probabilities $p_\downarrow$ and $p_\uparrow$ or a Z-channel with a crossover probability $p_\downarrow + p_\uparrow$.

VIII. Concluding Remarks

In storage systems with asymmetric errors, it is desirable to design error-correcting codes such that the reliability of each codeword is guaranteed in the worst case, and the size of the code is maximized. This motivated us to propose the concept of nonuniform codes, whose codewords can tolerate a number of asymmetric errors that depends on their Hamming weights. We derived an almost explicit upper bound on the size of nonuniform codes and compared the asymptotic performances of nonuniform codes and uniform codes—it is evident that there is a potential performance gain by using nonuniform codes. In addition, we presented two general constructions of nonuniform codes, including *layered codes* and *flipping codes*. Open problems include efficient encoding for *layered codes* and the construction of *flipping codes* when p is not small. In general, the construction of simple and efficient nonuniform codes is still an open problem.

Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve the quality of the paper.

References

[1] K. A. S. Abdel-Ghaffar and H. C. Ferreira, "Systematic encoding of the Varshamov–Tenengol'ts codes and the Constantin–Rao codes," *IEEE Trans. Inf. Theory*, vol. 44, no. 1, pp. 340–345, Jan. 1998.

[2] S. Al-Bassam, R. Venkatesan, and S. Al-Muhammadi, "New single asymmetric error-correcting codes," *IEEE Trans. Inf. Theory*, vol. 43, no. 5, pp. 1619–1623, Sep. 1997.

[3] B. Bose and S. Al-Bassam, "On systematic single asymmetric error-correcting codes," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 669–672, Mar. 2000.

[4] A. E. Brouwer, J. B. Shearer, N. J. A. Sloane, and W. D. Smith, "A new table of constant weight codes," *IEEE Trans. Inf. theory*, vol. 36, no. 6, pp. 1334–1380, Nov. 1990.

[5] Y. Cassuto, M. Schwartz, V. Bohossian, and J. Bruck, "Codes for asymmetric limited-magnitude errors with application to multi-level flash memories," *IEEE Trans. Inf. Theory*, vol. 56, no. 4, pp. 1582–1595, Apr. 2010.

[6] S. D. Constantin and T. R. N. Rao, "On the theory of binary asymmetric error-correcting codes," *Inf. Control*, vol. 40, pp. 20–36, 1979.

[7] P. Delsarte and P. Piret, "Bounds and constructions for binary asymmetric error-correcting codes," *IEEE Trans. Inf. Theory*, vol. 27, no. 1, pp. 125–128, Jan. 1981.

[8] T. Etzion, "New lower bounds for asymmetric and unidirectional codes," *IEEE Trans. Inf. Theory*, vol. 37, no. 6, pp. 1696–1704, Nov. 1991.

[9] G. Fang and H. C. A. van Tilborg, "Bounds and constructions of asymmetric or unidirectional error-correcting codes," in *Applicable Algebra in Engineering, Communication and Computing*. Berlin, Germany: Springer, 1992, vol. 3, pp. 269–300.

[10] F. Fu, S. Ling, and C. Xing, "New lower bounds and constructions for binary codes correcting asymmetric errors," *IEEE Trans. Inf. Theory*, vol. 49, no. 12, pp. 3294–3299, Dec. 2003.

[11] E. Fujiwara, *Code Design for Dependable Systems: Theory and Practical Applications*. New York, NY, USA: Wiley, 2006.

[12] I. Y. Goldbaum, "Bounds on the number of signals in codes with asymmetrical error correction," *Automat. Tekmekh.*, vol. 32, pp. 94–97, 1971.

[13] R. L. Graham and N. J. A. Sloane, "Lower bounds for constant weight codes," *IEEE Trans. Inf. Theory*, vol. 26, no. 1, pp. 37–43, Jan. 1980.

[14] T. Kløve, "Upper bounds on codes correcting asymmetric errors," *IEEE Trans. Inf. Theory*, vol. 27, no. 1, pp. 128–131, Jan. 1981.

[15] T. Kløve, Error correcting codes for the asymmetric channel Univ. Bergen, Dept. Informatics, Bergen, Norway, 1995.

[16] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam, The Netherlands: North Holland, 1977.

[17] Y. Saitoh, K. Yamaguchi, and H. Imai, "Some new binary codes correcting asymmetric/unidirectional errors," *IEEE Trans. Inf. Theory*, vol. 36, no. 3, pp. 645–647, May 1990.

[18] L. G. Tallini and B. Bose, "On a new class of error control codes and symmetric functions," in *Proc. IEEE Int. Symp. Inf. Theory*, 2008, pp. 980–984.

[19] L. G. Tallini and B. Bose, "On L1-distance error control codes," in *Proc. IEEE Int. Symp. Inf. Theory*, 2011, pp. 1026–1030.

[20] L. G. Tallini, S. Al-Bassam, and B. Bose, "On the capacity and codes for the Z-channel," in *Proc. IEEE Int. Symp. Inf. Theory*, 2002, p. 442.

[21] L. G. Tallini, S. Al-Bassam, and B. Bose, "Feedback codes achieving the capacity of the Z-channel," *IEEE Trans. Inf. Theory*, vol. 54, no. 3, pp. 1357–1362, Mar. 2008.

[22] L. G. Tallini, S. Elmougy, and B. Bose, "Analysis of plain and diversity combining hybrid ARQ protocols over the $\mathrm{m}(\geq 2)$–ary asymmetric channel," *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5550–5558, Dec. 2006.

[23] R. R. Varshamov, "Some features of linear codes that correct asymmetric errors (in Russian)," *Doklady Akad. Nauk. SSSR*, vol. 157, no. 3, pp. 546–548, 1964.

[24] R. R. Varshamov, "A class of codes for asymmetric channels and a problem from the additive theory of numbers," *IEEE Trans. Inf. Theory*, vol. 19, no. 1, pp. 92–95, Jan. 1973.

[25] C. Wang, S. R. Kulkarni, and H. V. Poor, "Density evolution for asymmetric memoryless channels," *IEEE Trans. Inf. Theory*, vol. 51, no. 12, pp. 4216–4236, Dec. 2005.

[26] J. H. Weber, C. de Vroedt, and D. E. Beokee, "New upper bounds on the size of codes correcting asymmetric errors," *IEEE Trans. Inf. Theory*, vol. 33, no. 3, pp. 434–437, May 1987.

[27] J. H. Weber, C. de Vroedt, and D. E. Boekee, "Bounds and constructions for binary codes of length less than 24 and asymmetric distance less than 6," *IEEE Trans. Inf. Theory*, vol. 34, no. 5, pp. 1321–1331, Sep. 1988.

[28] Z. Zhang and X. Xia, "New lower bounds for binary codes of asymmetric distance two," *IEEE Trans. Inf. Theory*, vol. 38, no. 5, pp. 1592–1597, Sep. 1992.

Hongchao Zhou received the B.Sc. degree in physics and mathematics and M.Sc. degree in control science and engineering from Tsinghua University, Beijing, China, in 2006 and 2008, respectively, and the M.Sc. degree and Ph.D. degree in electrical engineering from the California Institute of Technology, Pasadena, CA, in 2009 and 2012, respectively.

He is a postdoctoral researcher in the Research Laboratory of Electronics at the Massachusetts Institute of Technology, Cambridge. His current interests include information theory and randomness, quantum communication, information storage, and stochastic biological networks.

Anxiao Jiang (S'00–M'05–SM'12) received the B.Sc. degree in electronic engineering from Tsinghua University, Beijing, China in 1999, and the M.Sc. and Ph.D. degrees in electrical engineering from the California Institute of Technology, Pasadena, California in 2000 and 2004, respectively.

He is currently an Associate Professor in the Computer Science and Engineering Department at Texas A&M University in College Station, Texas. His research interests include information theory, data storage, networks and algorithm design.

Prof. Jiang is a recipient of the NSF CAREER Award in 2008 for his research on information theory for flash memories and a recipient of the 2009 IEEE Communications Society Best Paper Award in Signal Processing and Coding for Data Storage.

Jehoshua Bruck (S'86–M'89–SM'93–F'01) received the B.Sc. and M.Sc. degrees in electrical engineering from the Technion-Israel Institute of Technology, Haifa, Israel, in 1982 and 1985, respectively, and the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA, in 1989.

He is the Gordon and Betty Moore Professor of computation and neural systems and electrical engineering at the California Institute of Technology, Pasadena, CA. His extensive industrial experience includes working with IBM Almaden Research Center, as well as cofounding and serving as Chairman of Rainfinity, acquired by EMC in 2005; and XtremIO, acquired by EMC in 2012. His current research interests include information theory and systems and the theory of computation in biological networks.

Dr. Bruck is a recipient of the Feynman Prize for Excellence in Teaching, the Sloan Research Fellowship, the National Science Foundation Young Investigator Award, the IBM Outstanding Innovation Award, and the IBM Outstanding Technical Achievement Award. His papers were recognized in journals and conferences, including winning the 2010 IEEE Communications Society Best Student Paper Award in Signal Processing and Coding for Data Storage for his paper on codes for limited-magnitude errors in flash memories, the 2009 IEEE Communications Society Best Paper Award in Signal Processing and Coding for Data Storage for his paper on rank modulation for flash memories, the 2005 A. Schelkunoff Transactions Prize Paper Award from the IEEE Antennas and Propagation Society for his paper on signal propagation in wireless networks, and the 2003 Best Paper Award in the Design Automation Conference for his paper on cyclic combinational circuits.