# Syndrome-Coupled Rate-Compatible Error-Correcting Codes: Theory and Application

Pengfei Huang, *Member, IEEE*, Yi Liu, *Student Member, IEEE*, Xiaojie Zhang, Paul H. Siegel, *Life Fellow, IEEE*, and Erich F. Haratsch, *Senior Member, IEEE*

***Abstract*—Rate-compatible error-correcting codes (ECCs), which consist of a set of extended codes, are of practical interest in both wireless communications and data storage.**

**In this work, we first study the lower bounds for rate-compatible ECCs, thus proving the existence of good rate-compatible codes. Then, we propose a general framework for constructing rate-compatible ECCs based on cosets and syndromes of a set of nested linear codes. We evaluate our construction from two points of view. From a combinatorial perspective, we show that we can construct rate-compatible codes with increasing minimum distances, and we discuss decoding algorithms and correctable patterns of errors and erasures. From a probabilistic point of view, we prove that we are able to construct capacity-achieving rate-compatible codes, generalizing a recent construction of capacity-achieving rate-compatible polar codes. Applications of rate-compatible codes to data storage are considered. We design two-level rate-compatible codes based on Bose-Chaudhuri-Hocquenghem (BCH) and low-density parity-check (LDPC) codes which are two popular codes widely used in the data storage industry, and then we evaluate the performance of these codes in multi-level cell (MLC) flash memories. We also examine code performance on binary and $q$-ary symmetric channels. Finally, we briefly discuss two variations of our main construction and their relative performance.**

***Index Terms*—Rate-compatible codes, nested linear codes, capacity-achieving, polar codes, flash memory.**

## I. Introduction

Rate-compatible error-correcting codes (ECCs) consist of a set of codes with the same dimension but various code lengths, where all symbols of the higher rate code are part of the lower rate code. This allows to match the code rate of the sent data to the channel conditions by retransmitting incremental redundancy to the receiver. Such a scheme is known as hybrid automatic repeat request (HARQ) in wireless communications [7], [10], [19], [23], [24].

The idea of rate-compatible codes dates back to Davida and Reddy [8]. The most commonly used way to construct such codes is to use puncturing; that is, to start with a good low-rate code and then successively discard some of the coded symbols (parity-check symbols) to produce higher-rate codes. This approach has been used for algebraic codes [8], [38], convolutional codes [14], [20], turbo codes [25], [29], and low-density parity-check (LDPC) codes [10], [13]. The performance of punctured codes depends on the selected puncturing pattern. However, in general, determining good puncturing patterns is nontrivial, usually done with the aid of computer simulations.

The second approach is to use extending; that is, to start with a good high-rate code and then successively add more parity-check symbols to generate lower-rate codes. A two-level extending method called Construction X was introduced in [26], and later was generalized to Construction XX [1]. Both constructions were utilized to find new codes with good minimum distance. In [19], codes from the extending scheme were used for HARQ systems. Extension-based rate-compatible LDPC codes were designed in [6], [22], [37]. More recently, the extending approach was employed to construct capacity-achieving rate-compatible polar codes [15], [21], [27].

Besides their successful applications in wireless communications, rate-compatible ECCs are also potentially useful for emerging data storage, in particular, non-volatile solid-state memories, such as NAND flash memory. In fact, this particular application actually serves as a motivation of this research. NAND flash memory is a versatile non-volatile data storage medium, and has been widely used in consumer electronics as well as enterprise data centers [3], [4]. For flash memories, it is well known that the raw bit error rate (BER) increases with repeated program/erase (P/E) operations; see [5], [36], [40] for multi-level cell (MLC) flash memories and [39] for three-level cell (TLC) flash memories. Hence, at a higher P/E cycle count, a stronger ECC is needed to maintain the data integrity. Rate-compatible codes are a promising approach for satisfying this requirement, since they are capable of adapting different error-correcting capabilities to the corresponding bit error rates.

The goal of this paper is to provide a systematic approach for constructing rate-compatible codes with theoretically guaranteed properties. We use the extending approach and propose a new algebraic construction for rate-compatible codes;

This work was supported in part by Seagate Technology, and in part by NSF Grants CCF-1405119 and CCF-1619053. This paper was presented at the IEEE Information Theory Workshop, Kaohsiung, Taiwan, November, 2017, and the 9th Annual Non-Volatile Memories Workshop, San Diego, CA, USA, March, 2018.

P. Huang was with the Center for Memory and Recording Research, Department of Electrical and Computer Engineering, University of California San Diego, La Jolla, CA 92093, USA. He is now with Western Digital Corporation, Milpitas, CA 95035, USA (e-mail: pehuangucsd@gmail.com).

Y. Liu and P. H. Siegel are with the Center for Memory and Recording Research, Department of Electrical and Computer Engineering, University of California San Diego, La Jolla, CA 92093, USA (e-mail: yil333@ucsd.edu; psiegel@ucsd.edu).

X. Zhang was with CNEX Labs, San Jose, CA 95128, USA. He is now with Uber Advanced Technologies Group, San Francisco, CA 94107, USA (e-mail: snuzxj@gmail.com).

E. F. Haratsch is with Seagate Technology, Fremont, CA 94538, USA.

the properties of the constructed codes are then analyzed from both combinatorial and probabilistic perspectives. Our contributions are as follows:

1) lower bounds for rate-compatible codes, which have not been fully explored before, are derived;

2) a simple and general construction of rate-compatible codes based on cosets and syndromes (Construction 1) is proposed, and some examples are given;

3) minimum distances of the constructed codes are determined, decoding algorithms are presented, and correctable error-erasure patterns are studied;

4) capacity-achieving rate-compatible codes are constructed, and a connection to recent capacity-achieving rate-compatible polar codes is made;

5) although the focus of this paper is on the theoretical aspect of rate-compatible codes, applications of the codes in MLC flash memories, binary symmetric channels, and $q$-ary symmetric channels are investigated;

6) finally, some variations of Construction 1 are briefly discussed.

Portions of this work were presented in [16], [17].

The remainder of the paper is organized as follows. In Section II, we give the formal definition of rate-compatible codes and introduce notation used in the paper. In Section III, we study lower bounds for rate-compatible codes. In Section IV, we present a general construction for $M$-level rate-compatible codes, whose minimum distances are studied. Correctable patterns of errors and erasures are also investigated. In Section V, we show our construction can generate capacity-achieving rate-compatible codes by choosing the component codes properly and establish the connection to recent constructions of capacity-achieving rate-compatible polar codes. In Section VI, we investigate an application of two-level rate-compatible codes to MLC flash memories, and also evaluate the performance of rate-compatible codes over binary and $q$-ary symmetric channels. In Section VII, we briefly discuss two variations of our construction of rate-compatible codes. We conclude the paper in Section VIII.

## II. Definitions and Preliminaries

In this section, we give the basic definitions and preliminaries that will be used in the paper.

We use the notation $[n]$ to denote the set $\{1,\ldots,n\}$. Let $\mathbb{F}_q$ denote the finite field of size $q$, where $q$ is a prime power. For a length-$n$ vector $\boldsymbol{v}$ over $\mathbb{F}_q$ and a set $\mathcal{I} \subseteq [n]$, the operation $\pi_{\mathcal{I}}(\boldsymbol{v})$ denotes the restriction of the vector $\boldsymbol{v}$ to the coordinates in the set $\mathcal{I}$, and $w_q(\boldsymbol{v})$ represents the Hamming weight of the vector over $\mathbb{F}_q$. The transpose of a matrix $H$ is written as $H^T$, and the inverse of $H$ is denoted by $H^{-1}$. The kernel of a matrix $H$ is written as $\ker(H)$ and the row space of $H$ is the span of the row vectors of $H$, denoted by $\mathcal{R}(H)$. A linear code $\mathcal{C}$ over $\mathbb{F}_q$ of length $n$, dimension $k$, and minimum distance $d$ will be denoted by $\mathcal{C} = [n,k,d]_q$ or by $[n,k,d]_q$ for simplicity; in some cases, we will use notation $[n,k]_q$ to indicate only length and dimension. For any integers $a > b$, the summation in the form of $\sum_{i=a}^{b} X_i$ is defined to be 0. A binomial coefficient $\binom{a}{b}$ is defined to be 0 if $a < b$. For a set $\mathcal{C}$, $|\mathcal{C}|$ represents its cardinality. The $q$-ary entropy function $H_q\colon [0,1] \to [0,1]$, is defined by $H_q(x) = -x\log_q x - (1-x)\log_q(1-x) + x\log_q(q-1)$.

Now, we present the definition of rate-compatible codes.

**Definition 1.** *For $1 \leqslant i \leqslant M$, let $\mathcal{C}_i$ be an $[N_i,k,d_i]_q$ linear code, where $N_1 < N_2 < \cdots < N_M$. The encoder of $\mathcal{C}_i$ is denoted by $\mathcal{E}_{\mathcal{C}_i} : \mathbb{F}_q^k \to \mathcal{C}_i$. These $M$ linear codes are said to be $M$-level rate-compatible, if for each $i$, $1 \leqslant i \leqslant M-1$, the following condition is satisfied for every possible input $\boldsymbol{u} \in \mathbb{F}_q^k$,*

$$\mathcal{E}_{\mathcal{C}_i}(\boldsymbol{u}) = \pi_{[N_i]}\Big(\mathcal{E}_{\mathcal{C}_{i+1}}(\boldsymbol{u})\Big). \tag{1}$$

*We denote this $M$-level rate-compatible relation among these codes by $\mathcal{C}_1 \prec \mathcal{C}_2 \prec \cdots \prec \mathcal{C}_M$.*

**Remark 1.** For $1 \leqslant i \leqslant M-1$, the rates satisfy $R_i = \frac{k}{N_i} > R_{i+1} = \frac{k}{N_{i+1}}$, but the minimum distances obey $d_i \leqslant d_{i+1}$. For systematic codes, the condition in (1) indicates that the set of parity-check symbols of a higher-rate code is a subset of the parity-check symbols of a lower-rate code.

In this paper, we will consider $q$-ary memoryless symmetric (QMS) channels $W\colon \mathcal{X} \to \mathcal{Y}$, with input alphabet $\mathcal{X}$, output alphabet $\mathcal{Y}$, and transition probabilities $W(y|x)$, $x \in \mathcal{X}$, $y \in \mathcal{Y}$. The input alphabet $\mathcal{X}$ will always be $\mathbb{F}_q$.

For a QMS channel $W$, it is known that its capacity $C(W)$ is attained by the uniform input distribution, i.e.,

$$\begin{aligned} C(W) &= \max_{p(x)} I(X;Y) \\ &= \sum_{x\in\mathcal{X}} \sum_{y\in\mathcal{Y}} \frac{1}{q} W(y|x) \log_q \frac{W(y|x)}{\sum_{x'\in\mathcal{X}} \frac{1}{q} W(y|x')}. \end{aligned}$$

Note that the base of the logarithm is $q$.

Of particular interest are the $q$-ary symmetric channel (QSC) and the $q$-ary erasure channel (QEC), defined as follows [28].

The QSC with crossover probability $p$ has output alphabet $\mathbb{F}_q$, and for every pair of a sent symbol $x \in \mathbb{F}_q$ and a received symbol $y \in \mathbb{F}_q$, the conditional probability is:

$$W\{y|x\} = \begin{cases} 1-p & \text{if } y = x \\ p/(q-1) & \text{if } y \neq x. \end{cases}$$

The capacity of this channel is $C(W) = 1 - H_q(p)$ [28].

The QEC with erasure probability $p$ has output alphabet $\mathbb{F}_q \cup \{?\}$, where ? represents an erasure symbol. For every pair of a sent symbol $x \in \mathbb{F}_q$ and a received symbol $y \in \mathbb{F}_q \cup \{?\}$, the transition probability $W(y|x)$ is:

$$W(y|x) = \begin{cases} 1-p & \text{if } y = x \\ p & \text{if } y = ? \\ 0 & \text{otherwise.} \end{cases}$$

Its capacity $C(W)$ is $1-p$ [28].

## III. Lower Bounds for Rate-Compatible Codes

In this section, we derive lower bounds for rate-compatible codes.

### A. A General Lower Bound for M-Level Rate-Compatible Codes

Based on the technique used in the derivation of the Gilbert-Varshamov (GV) bound, we derive a GV-like lower bound for $M$-level rate-compatible codes.

**Theorem 2.** *There exist $M$-level rate-compatible codes $\mathcal{C}_1 \prec \mathcal{C}_2 \prec \cdots \prec \mathcal{C}_M$, where $\mathcal{C}_i = [N_i = \sum_{j=1}^{i} n_j,\ k,\ \geqslant d_i]_q$ for $1 \leqslant i \leqslant M$, if the following inequalities are satisfied for all $1 \leqslant i \leqslant M$,*

$$d_i = \max\left\{ d : \sum_{m=0}^{d-2} \binom{\sum_{j=1}^{i} n_j - 1}{m} (q-1)^m < \frac{q^{\sum_{j=1}^{i} n_j - k}}{M} \right\}. \tag{2}$$

*Proof:* The proof is based on a combinatorial argument. See Appendix A. ∎

The following corollary follows from Theorem 2, which shows that there exist good rate-compatible codes in the sense that each code can meet the corresponding asymptotic GV bound.

**Corollary 3.** *There exist $M$-level rate-compatible codes $\mathcal{C}_1 \prec \mathcal{C}_2 \prec \cdots \prec \mathcal{C}_M$, where $\mathcal{C}_i = [N_i,\ k = R_i N_i,\ \geqslant \delta_i N_i]_q$ for $1 \leqslant i \leqslant M$ and $0 < \delta_i \leqslant 1 - (1/q)$. As the code length $N_i$ tends to infinity, the rate $R_i$ meets the asymptotic GV bound:*

$$R_i \geqslant 1 - H_q(\delta_i).$$

*Proof:* Let $V_q(n,t) = \sum_{m=0}^{t} \binom{n}{m} (q-1)^m$. From Theorem 2, there exist $M$-level rate-compatible codes $\mathcal{C}_i = [N_i,\ k = R_i N_i,\ \geqslant \delta_i N_i]_q$ for $1 \leqslant i \leqslant M$ such that

$$V_q(N_i - 1, \delta_i N_i - 1) \geqslant \frac{q^{N_i - k}}{M}.$$

Since $V_q(n,t) \leqslant q^{nH_q(t/n)}$ for $0 \leqslant t/n \leqslant 1 - (1/q)$ [28], we have

$$q^{N_i H_q(\delta_i)} \geqslant V_q(N_i, \delta_i N_i) \geqslant V_q(N_i - 1, \delta_i N_i - 1) \geqslant \frac{q^{N_i - k}}{M},$$

which gives $R_i \geqslant 1 - H_q(\delta_i) - \frac{\log_q M}{N_i}$. As $N_i$ goes to infinity, we obtain the result. ∎

**Remark 2.** The minimum distance $d_i$ in (2) grows at least linearly in the code length $N_i$ for a fixed rate $\frac{k}{N_i}$. From the proof of Corollary 3, we can see that there exist $M$-level rate-compatible codes $\mathcal{C}_i = [N_i,\ k,\ \geqslant d_i]_q$ for $1 \leqslant i \leqslant M$ and $\frac{d_i}{N_i} \leqslant 1 - (1/q)$ such that

$$H_q\left(\frac{d_i}{N_i}\right) \geqslant 1 - \frac{k}{N_i} - \frac{\log_q M}{N_i}.$$

The function $H_q(x)$ is monotonically increasing for $x$ in $[0, 1 - (1/q)]$. For this range of interest, let us define the corresponding inverse function $H_q^{-1} : [0,1] \to [0, 1 - (1/q)]$ which is also a monotonically increasing function. Now, a lower bound on $d_i$ can be expressed as

$$d_i \geqslant N_i H_q^{-1}\left(1 - \frac{k}{N_i} - \frac{\log_q M}{N_i}\right). \tag{3}$$

The lower bound (3) shows that the minimum distance $d_i$ grows at least linearly in the code length $N_i$.

### B. A Lower Bound for Two-Level Rate-Compatible Codes with Known Weight Enumerator

For two-level rate-compatible codes, if the weight enumerator of the higher-rate code is known, we have the following lower bound.

**Theorem 4.** *Let $\mathcal{C}_1$ be an $[n_1, k, d_1]_q$ code with weight enumerator $A(s) = \sum_{w=0}^{n_1} A_w s^w$, where $A_w$ is the number of codewords of Hamming weight $w$. There exist two-level rate-compatible codes $\mathcal{C}_1 \prec \mathcal{C}_2 = [N_2 = n_1 + n_2,\ k,\ \geqslant d_2]_q$, if*

$$\sum_{w=1}^{d_2 - 1} B_w < q^{n_2},$$

*where $B_w = \frac{1}{q-1} \sum_{m=1}^{w} A_m \binom{n_2}{w-m} (q-1)^{w-m}$, for $1 \leqslant w \leqslant N_2$.*

*Proof:* The proof is based on a probabilistic argument. See Appendix B. ∎

## IV. A General Construction for $M$-Level Rate-Compatible Codes

In this section, we present a general algebraic construction for $M$-level rate-compatible codes $\mathcal{C}_1 \prec \mathcal{C}_2 \prec \cdots \prec \mathcal{C}_M$. We then derive their minimum distances. The decoding algorithm and correctable error-erasure patterns are studied. We focus on combinatorial properties here and will leave the discussion of the capacity-achieving property of our construction to the next section.

The heuristic idea underlying our rate-compatible code construction is based upon an elementary structural property of linear codes. It is known that a linear code $\mathcal{C}$ can be partitioned into cosets of a proper subcode $\mathcal{C}'$ of $\mathcal{C}$. Each coset contains fewer codewords than $\mathcal{C}$ and hence has a potentially stronger error-correcting capability than the code $\mathcal{C}$, and every codeword $\boldsymbol{c} \in \mathcal{C}$ belongs to a unique coset. If one knows the coset to which the codeword $\boldsymbol{c}$ belongs, then $\boldsymbol{c}$ can be decoded in that coset instead of in the entire code $\mathcal{C}$; as a result, more errors can be corrected. This coset identifier can be represented conveniently by the syndrome of $\boldsymbol{c}$ with respect to some appropriate subcode parity-check matrix, thanks to the fact that all the codewords in the same coset have the same syndrome and disjoint cosets have distinct syndromes. In other words, the syndrome of the codeword $\boldsymbol{c}$ can be used to identify the coset that includes $\boldsymbol{c}$, and can therefore help improve the overall error-correcting capability. Our rate-compatible code construction exploits this fact and includes the syndrome information among the extra

parities appended to the codeword $\boldsymbol{c}$. To protect the syndrome information, an auxiliary code is used. If the auxiliary code in turn contains a subcode, an auxiliary syndrome can be generated for the protected syndrome codeword. This idea can be generalized by using a set of nested subcodes at each level, and by progressively generating and encoding syndromes of component codes in a coupled manner across levels. For this reason, we refer to the resulting codes as *syndrome-coupled rate-compatible codes*. The precise details of the construction are presented in the following subsection.

### A. Code Construction

Our construction of $M$-level rate-compatible codes requires several sets of nested component codes defined as follows.

1) For level 1, choose a set of nested codes $\mathcal{C}_1^M \subset \mathcal{C}_1^{M-1} \subset \cdots \subset \mathcal{C}_1^1 = \mathcal{C}_1 = [n_1, k, d_1]_q$, where $\mathcal{C}_1^i = [n_1, n_1 - \sum_{m=1}^{i} v_m, d_i]_q$ for $1 \leqslant i \leqslant M$. We have $k = n_1 - v_1$ and $d_1 \leqslant d_2 \leqslant \cdots \leqslant d_M$. Define $\mathcal{C}_1^0 = \emptyset$ and for $1 \leqslant \ell \leqslant i$, let $H_{\mathcal{C}_1^\ell | \mathcal{C}_1^{\ell-1}}$ represent a $v_\ell \times n_1$ matrix over $\mathbb{F}_q$ such that $\mathcal{C}_1^i$ has the following parity-check matrix:

$$H_{\mathcal{C}_1^i} = \begin{bmatrix} H_{\mathcal{C}_1^1} \\ H_{\mathcal{C}_1^2 | \mathcal{C}_1^1} \\ \vdots \\ H_{\mathcal{C}_1^i | \mathcal{C}_1^{i-1}} \end{bmatrix}. \tag{4}$$

The encoder of the code $\mathcal{C}_1$ is denoted by $\mathcal{E}_{\mathcal{C}_1} : \mathbb{F}_q^k \to \mathcal{C}_1$. We also use $\mathcal{E}_{\mathcal{C}_1}^{-1}$ as the inverse of the encoding mapping.

2) For level $i$, $2 \leqslant i \leqslant M$, consider an auxiliary set of nested codes $\mathcal{A}_i^M \subset \mathcal{A}_i^{M-1} \subset \cdots \subset \mathcal{A}_i^{i+1} \subset \mathcal{A}_i^i$, where $\mathcal{A}_i^j = [n_i, v_i + \sum_{m=2}^{i-1} \lambda_m^i - \sum_{\ell=i+1}^{j} \lambda_i^\ell, \delta_i^j]_q$ for $i \leqslant j \leqslant M$. Let $H_{\mathcal{A}_i^i}$ represent an $(n_i - v_i - \sum_{m=2}^{i-1} \lambda_m^i) \times n_i$ matrix over $\mathbb{F}_q$ and $H_{\mathcal{A}_i^\ell | \mathcal{A}_i^{\ell-1}}$, $i+1 \leqslant \ell \leqslant j$, represent a $\lambda_i^\ell \times n_i$ matrix over $\mathbb{F}_q$, such that $\mathcal{A}_i^j$ has the following parity-check matrix:

$$H_{\mathcal{A}_i^j} = \begin{bmatrix} H_{\mathcal{A}_i^i} \\ H_{\mathcal{A}_i^{i+1} | \mathcal{A}_i^i} \\ \vdots \\ H_{\mathcal{A}_i^j | \mathcal{A}_i^{j-1}} \end{bmatrix}. \tag{5}$$

For each $2 \leqslant i \leqslant M$, the encoder of the code $\mathcal{A}_i^i$ is denoted by $\mathcal{E}_{\mathcal{A}_i^i} : \mathbb{F}_q^{v_i + \sum_{m=2}^{i-1} \lambda_m^i} \to \mathcal{A}_i^i$. We also use $\mathcal{E}_{\mathcal{A}_i^i}^{-1}$ as the inverse of the encoding mapping.

We also define $\mathcal{C}_1^{M+1} = \emptyset$ and $\mathcal{A}_i^{M+1} = \emptyset$ for $2 \leqslant i \leqslant M$.

The general construction of $M$-level syndrome-coupled rate-compatible codes $\mathcal{C}_1 \prec \mathcal{C}_2 \prec \cdots \prec \mathcal{C}_M$, based upon the nested component codes introduced above, is given in Construction 1.

**Construction 1: Encoding Procedure**

**Input:** A length-$k$ vector $\boldsymbol{u}$ of information symbols over $\mathbb{F}_q$.
**Output:** A codeword $\boldsymbol{c}_i \in \mathcal{C}_i$ over $\mathbb{F}_q$, for $i = 1, \ldots, M$.

1: $\boldsymbol{c}_1 = \mathcal{E}_{\mathcal{C}_1}(\boldsymbol{u})$.
2: $\boldsymbol{s}_i = \boldsymbol{c}_1 H_{\mathcal{C}_1^i | \mathcal{C}_1^{i-1}}^T$ for $i = 2, 3, \ldots, M$.
3: **for** $i = 2, \ldots, M$ **do**
4: $\quad \boldsymbol{a}_i^i = \mathcal{E}_{\mathcal{A}_i^i}\Big((\boldsymbol{s}_i, \Lambda_2^i, \ldots, \Lambda_{i-1}^i)\Big)$. // For $i = 2$, we define $(\boldsymbol{s}_i, \Lambda_2^i, \ldots, \Lambda_{i-1}^i) = \boldsymbol{s}_2$. //
5: $\quad \boldsymbol{c}_i = (\boldsymbol{c}_1, \boldsymbol{a}_2^2, \ldots, \boldsymbol{a}_i^i)$.
6: $\quad$ **for** $j = i+1, \ldots, M$ **do**
7: $\quad\quad \Lambda_i^j = \boldsymbol{a}_i^i H_{\mathcal{A}_i^j | \mathcal{A}_i^{j-1}}^T$.
8: $\quad$ **end for**
9: **end for**

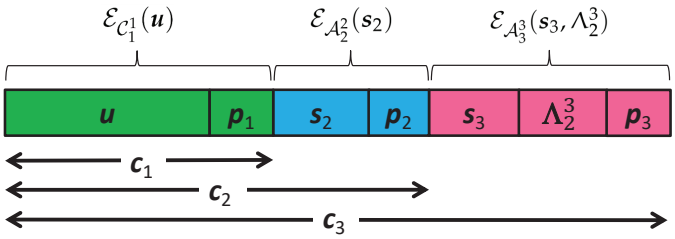



Fig. 1. Structure of systematic three-level rate-compatible codes obtained from Construction 1.

**Remark 3.** As an example of Construction 1, we consider the case $M = 3$. Fig. 1 illustrates the extended structure of the systematic three-level rate-compatible codes produced by the construction. First, we encode $\boldsymbol{u}$ using $\mathcal{C}_1^1$ to obtain $\boldsymbol{c}_1 = \mathcal{E}_{\mathcal{C}_1^1}(\boldsymbol{u}) = (\boldsymbol{u}, \boldsymbol{p}_1)$, where $\boldsymbol{p}_1$ represents the parity. With $\boldsymbol{c}_1$, we generate syndromes $\boldsymbol{s}_2$ and $\boldsymbol{s}_3$. For the second level, we encode $\boldsymbol{s}_2$ using $\mathcal{A}_2^2$ to obtain $\boldsymbol{a}_2^2 = \mathcal{E}_{\mathcal{A}_2^2}(\boldsymbol{s}_2) = (\boldsymbol{s}_2, \boldsymbol{p}_2)$, where $\boldsymbol{p}_2$ represents the parity. The extended codeword $\boldsymbol{c}_2 \in \mathcal{C}_2$ thus has the form $\boldsymbol{c}_2 = \Big(\mathcal{E}_{\mathcal{C}_1^1}(\boldsymbol{u}), \mathcal{E}_{\mathcal{A}_2^2}(\boldsymbol{s}_2)\Big)$. The syndrome $\Lambda_2^3$ is calculated from $\boldsymbol{a}_2^2$ with respect to $\mathcal{A}_2^3$ . For the third level, we encode $\boldsymbol{s}_3$ and $\Lambda_2^3$ together using $\mathcal{A}_3^3$ to obtain $\boldsymbol{a}_3^3 = \mathcal{E}_{\mathcal{A}_3^3}(\boldsymbol{s}_3, \Lambda_2^3) = (\boldsymbol{s}_3, \Lambda_2^3, \boldsymbol{p}_3)$, where $\boldsymbol{p}_3$ represents the parity. The extended codeword $\boldsymbol{c}_3 \in \mathcal{C}_3$ thus has the form $\boldsymbol{c}_3 = \Big(\mathcal{E}_{\mathcal{C}_1^1}(\boldsymbol{u}), \mathcal{E}_{\mathcal{A}_2^2}(\boldsymbol{s}_2), \mathcal{E}_{\mathcal{A}_3^3}(\boldsymbol{s}_3, \Lambda_2^3)\Big)$.

We have the following theorem on the code parameters of the constructed rate-compatible codes $\mathcal{C}_1 \prec \mathcal{C}_2 \prec \cdots \prec \mathcal{C}_M$.

**Theorem 5.** *From Construction 1, the code $\mathcal{C}_i$, $1 \leqslant i \leqslant M$, has length $N_i = \sum_{j=1}^{i} n_j$ and dimension $K_i = k$. Moreover, assume that $\mathcal{A}_i^j$, $2 \leqslant i \leqslant M$ and $i \leqslant j \leqslant M$, has minimum distance $\delta_i^j \geqslant d_j - d_{i-1}$. Then $\mathcal{C}_i$ has minimum distance $D_i = d_i$.*

*Proof:* The code length and dimension are obvious. In the following, we prove the minimum distance. Since the proofs for all $\mathcal{C}_i$, $1 \leqslant i \leqslant M$, are similar, we only give a proof for the code $\mathcal{C}_M$.

We first prove $D_M \geqslant d_M$ by showing that any nonzero codeword $\boldsymbol{c}_M \in \mathcal{C}_M$ has weight at least $d_M$. To see this, for any nonzero codeword $\boldsymbol{c}_1 \in \mathcal{C}_1$, there exists an integer $\gamma_1$,

$1 \leqslant \gamma_1 \leqslant M$, such that $\boldsymbol{c}_1 \in \mathcal{C}_1^{\gamma_1}$ and $\boldsymbol{c}_1 \notin \mathcal{C}_1^{\gamma_1+1}$. Let $\boldsymbol{c}_M \in \mathcal{C}_M$ be the codeword derived from $\boldsymbol{c}_1$. Then, we have $w_q(\boldsymbol{c}_M) \geqslant w_q(\boldsymbol{c}_1) \geqslant d_{\gamma_1}$. If $\gamma_1 = M$, we are done; otherwise if $1 \leqslant \gamma_1 \leqslant M-1$ we have $\boldsymbol{s}_{\gamma_1+1} \neq \boldsymbol{0}$ and $\boldsymbol{a}_{\gamma_1+1}^{\gamma_1+1} \neq \boldsymbol{0}$.

Now, for $\boldsymbol{a}_{\gamma_1+1}^{\gamma_1+1}$, there exists an integer $\gamma_2$, $\gamma_1+1 \leqslant \gamma_2 \leqslant M$, such that $\boldsymbol{a}_{\gamma_1+1}^{\gamma_1+1} \in \mathcal{A}_{\gamma_1+1}^{\gamma_2}$ and $\boldsymbol{a}_{\gamma_1+1}^{\gamma_1+1} \notin \mathcal{A}_{\gamma_1+1}^{\gamma_2+1}$. Then, we have $w_q(\boldsymbol{c}_M) \geqslant w_q(\boldsymbol{c}_1) + w_q(\boldsymbol{a}_{\gamma_1+1}^{\gamma_1+1}) \geqslant d_{\gamma_1} + d_{\gamma_2} - d_{\gamma_1} = d_{\gamma_2}$. If $\gamma_2 = M$, done; otherwise for $\gamma_1 + 1 \leqslant \gamma_2 \leqslant M-1$, we have $\Lambda_{\gamma_1+1}^{\gamma_2+1} \neq \boldsymbol{0}$ and $\boldsymbol{a}_{\gamma_2+1}^{\gamma_2+1} \neq \boldsymbol{0}$.

Using the same argument as above, it is clear that we can find a sequence of $\gamma_1 < \gamma_2 < \cdots < \gamma_i$, where $i$ is a certain integer $1 \leqslant i \leqslant M$ and $\gamma_i = M$, such that $w_q(\boldsymbol{c}_1) \geqslant d_{\gamma_1}$, $w_q(\boldsymbol{a}_{\gamma_1+1}^{\gamma_1+1}) \geqslant d_{\gamma_2} - d_{\gamma_1}$, $w_q(\boldsymbol{a}_{\gamma_2+1}^{\gamma_2+1}) \geqslant d_{\gamma_3} - d_{\gamma_2}$, $\cdots$, $w_q(\boldsymbol{a}_{\gamma_{i-1}+1}^{\gamma_{i-1}+1}) \geqslant d_{\gamma_i} - d_{\gamma_{i-1}} = d_M - d_{\gamma_{i-1}}$. Then, we have $w_q(\boldsymbol{c}_M) \geqslant w_q(\boldsymbol{c}_1) + \sum_{j=1}^{i-1} w_q(\boldsymbol{a}_{\gamma_j+1}^{\gamma_j+1}) \geqslant d_M$. Thus, we have $D_M \geqslant d_M$.

There exists a codeword $\boldsymbol{c}_1 \in \mathcal{C}_1^M$ such that $w_q(\boldsymbol{c}_1) = d_M$, so we have $w_q(\boldsymbol{c}_M) = d_M$, implying $D_M \leqslant d_M$. ■

Next, we provide an example of three-level rate-compatible codes produced by Construction 1.

**Example 1.** Consider a set of nested binary BCH codes $\mathcal{C}_1^3 = [15,5,7]_2 \subset \mathcal{C}_1^2 = [15,7,5]_2 \subset \mathcal{C}_1^1 = [15,11,3]_2$. Choose a set of auxiliary codes $\mathcal{A}_2^3 = [5,1,4]_2 \subset \mathcal{A}_2^2 = [5,4,2]_2$, and $\mathcal{A}_3^3 = [6,5,2]_2$, where the code $\mathcal{A}_2^3$ is obtained by shortening an $[8,4,4]_2$ extended Hamming code by three information bits.

Then, from Construction 1 and Theorem 5, we obtain three-level rate-compatible codes $\mathcal{C}_1 = [15,11,3]_2 \prec \mathcal{C}_2 = [20,11,5]_2 \prec \mathcal{C}_3 = [26,11,7]_2$. Note that $\mathcal{C}_1$ and $\mathcal{C}_2$ are optimal, achieving the maximum possible dimensions with the given code length and minimum distance. The dimension of $\mathcal{C}_3$ is close to the upper bound 13 according to the online Table [33]. □

The following example demonstrates some optimal two-level rate-compatible codes from Construction 1.

**Example 2.** Consider two nested maximum distance separable (MDS) codes $\mathcal{C}_1^2 = [n_1, k-2, d_1+2]_q \subset \mathcal{C}_1^1 = [n_1, k, d_1]_q$. Choose an auxiliary code $\mathcal{A}_2^2 = [3,2,2]_q$. Then, from Construction 1 and Theorem 5, we obtain the code $\mathcal{C}_2 = [n_1+3, k, d_1+2]_q$, which is an almost MDS (AMDS) code [9], [28]. Let $L_q(k)$ denote the largest length of any linear MDS code of dimension $k$ over $\mathbb{F}_q$, e.g., $L_q(2) = q+1$ [28]. If $L_q(k) < n_1 + 3$, then $\mathcal{C}_2$ is optimal with respect to the minimum distance for the given code length and dimension. Thus, we have optimal rate-compatible codes $\mathcal{C}_1 = [n_1, k, d_1]_q \prec \mathcal{C}_2 = [n_1+3, k, d_1+2]_q$.

Now, consider two nested binary Reed-Muller (RM) codes $\mathcal{C}_1^2 = RM(0,m) = [2^m, 1, 2^m]_2 \subset \mathcal{C}_1^1 = RM(1,m) = [2^m, m+1, 2^{m-1}]_2$. We choose the auxiliary code $\mathcal{A}_2^2 = [2^m - 1, m, 2^{m-1}]_2$ to be a simplex code. From Construction 1 and Theorem 5, we obtain the code $\mathcal{C}_2 = [2^{m+1}-1, m+1, 2^m]_2$, which is also a simplex code, so it is optimal. Therefore, we have optimal rate-compatible codes $\mathcal{C}_1 = [2^m, m+1, 2^{m-1}]_2 \prec \mathcal{C}_2 = [2^{m+1}-1, m+1, 2^m]_2$. □

### *B. Decoding Algorithm and Correctable Error-Erasure Patterns*

In the following, we study decoding algorithms and correctable patterns of errors and erasures for rate-compatible codes obtained from Construction 1. For simple notation and concise analysis, we focus on the code $\mathcal{C}_M$. Any results obtained for $\mathcal{C}_M$ can be easily modified for other codes $\mathcal{C}_i$, $1 \leqslant i \leqslant M-1$, so details are omitted.

Assume a codeword $\boldsymbol{c}_M \in \mathcal{C}_M$, $\boldsymbol{c}_M = (\boldsymbol{c}_1, \boldsymbol{a}_2^2, \ldots, \boldsymbol{a}_M^M)$, is transmitted. Let the corresponding received word be $\boldsymbol{y} = (\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_M)$ with errors and erasures, i.e., $\boldsymbol{y} \in (\mathbb{F}_q \cup \{?\})^{N_M}$, where the symbol ? represents an erasure. For $1 \leqslant i \leqslant M$, let $t_i$ and $\tau_i$ denote the number of errors and erasures in the sub-block $\boldsymbol{y}_i$ of the received word $\boldsymbol{y}$.

The code $\mathcal{C}_M$ can correct any combined error and erasure pattern that satisfies the following condition:

$$\begin{aligned} &2t_1 + \tau_1 \leqslant d_M - 1, \\ &2t_i + \tau_i \leqslant \delta_i^M - 1, \ \forall\ 2 \leqslant i \leqslant M. \end{aligned} \tag{6}$$

To see this, we present a decoding algorithm, referred to as Algorithm 1, for $\mathcal{C}_M$. It uses the following component error-erasure decoders:

a) The error-erasure decoder $\mathcal{D}_{\mathcal{C}_1^i}$ for a coset of the code $\mathcal{C}_1^i$, for $1 \leqslant i \leqslant M$, is defined by

$$\mathcal{D}_{\mathcal{C}_1^i} : (\mathbb{F}_q \cup \{?\})^{n_1} \times (\mathbb{F}_q \cup \{?\})^{\sum_{j=1}^{i} v_j} \to \mathcal{C}_1^i + \boldsymbol{e} \cup \{\text{“e”}\}.$$

The decoder $\mathcal{D}_{\mathcal{C}_1^i}$ either produces a codeword in the coset $\mathcal{C}_1^i + \boldsymbol{e}$ or a decoding failure “e”. For our purpose, we require that $\mathcal{D}_{\mathcal{C}_1^i}$ have the following error-erasure correcting capability. For a sent codeword $\boldsymbol{c}$ in the coset $\mathcal{C}_1^i + \boldsymbol{e}$, where the vector $\boldsymbol{e}$ is a coset leader, if the inputs of $\mathcal{D}_{\mathcal{C}_1^i}$ are a length-$n_1$ received word $\boldsymbol{y}$ having $t$ errors and $\tau$ erasures, where $2t + \tau \leqslant d_i - 1$, and a correct length-$\sum_{j=1}^{i} v_j$ syndrome vector $\boldsymbol{s}$, $\boldsymbol{s} = \boldsymbol{e} H_{\mathcal{C}_1^i}^T$, then $\mathcal{D}_{\mathcal{C}_1^i}$ can correct all these errors and erasures. It is well known that such a decoder exists [28].

b) The error-erasure decoder $\mathcal{D}_{\mathcal{A}_i^j}$ for a coset of the code $\mathcal{A}_i^j$, for $2 \leqslant i \leqslant M$ and $i \leqslant j \leqslant M$, is defined by

$$\begin{aligned} \mathcal{D}_{\mathcal{A}_i^j} : (\mathbb{F}_q \cup \{?\})^{n_i} \times (\mathbb{F}_q \cup \{?\})^{n_i - v_i - \sum_{m=2}^{i-1} \lambda_m^i + \sum_{\ell=i+1}^{j} \lambda_i^\ell} \\ \to \mathcal{A}_i^j + \boldsymbol{e} \cup \{\text{“e”}\}. \end{aligned}$$

The decoder $\mathcal{D}_{\mathcal{A}_i^j}$ either produces a codeword in the coset $\mathcal{A}_i^j + \boldsymbol{e}$ or a decoding failure “e”. Similar to $\mathcal{D}_{\mathcal{C}_1^i}$, we assume that $\mathcal{D}_{\mathcal{A}_i^j}$ has the following error-erasure correcting capability. For a sent codeword $\boldsymbol{c}$ in the coset $\mathcal{A}_i^j + \boldsymbol{e}$, where $\boldsymbol{e}$ is a coset leader, if the inputs of $\mathcal{D}_{\mathcal{A}_i^j}$ are a length-$n_i$ received word $\boldsymbol{y}$ having $t$ errors and $\tau$ erasures, where $2t + \tau \leqslant \delta_i^j - 1$, and a correct

length-$(n_i - v_i - \sum_{m=2}^{i-1} \lambda_m^i + \sum_{\ell=i+1}^{j} \lambda_i^\ell)$ syndrome vector $\boldsymbol{s}$, $\boldsymbol{s} = \boldsymbol{e} H_{\mathcal{A}_i^j}^T$, then $\mathcal{D}_{\mathcal{A}_i^j}$ can correct all these errors and erasures.

The decoding algorithm is presented in Algorithm 1.

**Algorithm 1: Decoding Procedure for $\mathcal{C}_M$**

**Input:** Received word $\boldsymbol{y} = (\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_M)$.
**Output:** A length-$k$ vector $\boldsymbol{u}$ of information symbols over $\mathbb{F}_q$ or a decoding failure "e".

1: **for** $i = M, M-1, \ldots, 2$ **do**
2: Let the syndrome $\Lambda_i^i = \boldsymbol{0}$.
3: $\hat{\boldsymbol{a}}_i^i = \mathcal{D}_{\mathcal{A}_i^M}\Big(\boldsymbol{y}_i, (\Lambda_i^i, \Lambda_i^{i+1}, \ldots, \Lambda_i^M)\Big)$.
4: $(\boldsymbol{s}_i, \Lambda_2^i, \ldots, \Lambda_{i-1}^i) = \mathcal{E}_{\mathcal{A}_i^i}^{-1}(\hat{\boldsymbol{a}}_i^i)$. // For $i = 2$, we define $(\boldsymbol{s}_i, \Lambda_2^i, \ldots, \Lambda_{i-1}^i) = \boldsymbol{s}_2$. //
5: **end for**
6: Let the syndrome $\boldsymbol{s}_1 = \boldsymbol{0}$.
7: $\boldsymbol{c}_1 = \mathcal{D}_{\mathcal{C}_1^M}\big(\boldsymbol{y}_1, (\boldsymbol{s}_1, \boldsymbol{s}_2, \ldots, \boldsymbol{s}_M)\big)$.
8: Output $\boldsymbol{u} = \mathcal{E}_{\mathcal{C}_1}^{-1}(\boldsymbol{c}_1)$ if all above steps are successful; otherwise, return "e".

**Theorem 6.** *The code $\mathcal{C}_M$ can correct any pattern of errors and erasures satisfying the conditions in (6) when decoded using Algorithm 1.*

*Proof:* The proof follows from Algorithm 1 that decodes the last sub-block $\boldsymbol{y}_M$ to the first sub-block $\boldsymbol{y}_1$ progressively. First, since the code $\mathcal{A}_M^M$ has minimum distance $\delta_M^M$, it can correct $\boldsymbol{y}_M$ under the condition $2t_M + \tau_M \leqslant \delta_M^M - 1$. Thus, we obtain correct syndromes $\boldsymbol{s}_M, \Lambda_2^M, \ldots, \Lambda_{M-1}^M$.

Next, with the correct syndrome $\Lambda_{M-1}^M$, the coset decoder $\mathcal{D}_{\mathcal{A}_{M-1}^M}$ can correct $\boldsymbol{y}_{M-1}$ under the condition $2t_{M-1} + \tau_{M-1} \leqslant \delta_{M-1}^M - 1$. Thus, we obtain correct syndromes $\boldsymbol{s}_{M-1}, \Lambda_2^{M-1}, \ldots, \Lambda_{M-2}^{M-1}$.

We now conduct the above decoding procedure progressively. For any $i$, $2 \leqslant i \leqslant M-2$, using the correct syndromes $\Lambda_i^{i+1}, \ldots, \Lambda_i^M$ for coset decoding, the sub-block $\boldsymbol{y}_i$ can be corrected under the condition $2t_i + \tau_i \leqslant \delta_i^M - 1$.

At the last step, we have obtained correct syndromes $\boldsymbol{s}_2, \ldots, \boldsymbol{s}_M$. Therefore, the sub-block $\boldsymbol{y}_1$ is corrected. ∎

Using nested MDS codes as component codes, Construction 1 can generate an *optimal* code $\mathcal{C}_M$ with respect to the capability of correcting certain error-erasure patterns. For simplicity, we start with the case of $M = 3$ as an example.

**Example 3.** Consider a set of nested MDS codes, e.g., Reed-Solomon codes, $\mathcal{C}_1^3 = [n_1, n_1 - d_3 + 1, d_3]_q \subset \mathcal{C}_1^2 = [n_1, n_1 - d_2 + 1, d_2]_q \subset \mathcal{C}_1^1 = [n_1, n_1 - d_1 + 1, d_1]_q$. Choose a set of auxiliary MDS codes $\mathcal{A}_2^3 = [2(d_2 - d_1) - 1, 2d_2 - d_3 - d_1, d_3 - d_1]_q \subset \mathcal{A}_2^2 = [2(d_2 - d_1) - 1, d_2 - d_1, d_2 - d_1]_q$, and $\mathcal{A}_3^3 = [3(d_3 - d_2) - 1, 2(d_3 - d_2), d_3 - d_2]_q$.

Then, from Construction 1 and Theorem 5, we obtain three-level rate-compatible codes $\mathcal{C}_1 = [n_1, n_1 - d_1 + 1, d_1]_q \prec \mathcal{C}_2 = [n_1 + 2(d_2 - d_1) - 1, n_1 - d_1 + 1, d_2]_q \prec \mathcal{C}_3 = [n_1 + 2(d_2 - d_1) + 3(d_3 - d_2) - 2, n_1 - d_1 + 1, d_3]_q$.

From the condition in (6) and Theorem 6, the code $\mathcal{C}_3$ can correct any pattern of errors and erasures satisfying

$$2t_i + \tau_i \leqslant d_3 - d_{i-1} - 1, \ \forall\ 1 \leqslant i \leqslant 3, \tag{7}$$

where $d_0$ is defined to be 0. □

In general, the dimension of $\mathcal{C}_3$ in Example 3 cannot achieve the upper bounds given by traditional bounds (e.g., Singleton and Hamming bounds). However, $\mathcal{C}_3$ is optimal in the sense of having the largest possible dimension among all codes with the three-level structure and the same error-erasure correcting capability; that is, we have the following lemma.

**Lemma 7.** *Let $\mathcal{C}_3$ be a code of length $n_1 + 2(d_2 - d_1) + 3(d_3 - d_2) - 2$ and dimension $k_3$ over $\mathbb{F}_q$. Each codeword $\boldsymbol{c}_3 \in \mathcal{C}_3$ has three sub-blocks $(\boldsymbol{c}_1, \boldsymbol{a}_2^2, \boldsymbol{a}_3^3)$ : 1) $\boldsymbol{c}_1$ of length $n_1$, 2) $\boldsymbol{a}_2^2$ of length $2(d_2 - d_1) - 1$, and 3) $\boldsymbol{a}_3^3$ of length $3(d_3 - d_2) - 1$. Assume that each sub-block of $\mathcal{C}_3$ can correct all error and erasure patterns satisfying the condition in (7). Then, we must have $k_3 \leqslant n_1 - d_1 + 1$.*

*Proof:* We prove Lemma 7 by contradiction.

Let $\mathcal{I}_1$ be the set of any $d_3 - 1$ coordinates of $\boldsymbol{c}_1$, $\mathcal{I}_2$ be the set of any $d_3 - d_1 - 1$ coordinates of $\boldsymbol{a}_2^2$, and $\mathcal{I}_3$ be the set of any $d_3 - d_2 - 1$ coordinates of $\boldsymbol{a}_3^3$. Let $\mathcal{I}$ be the set of all the coordinates of $\boldsymbol{c}_3$.

We have $|\mathcal{I} \backslash (\mathcal{I}_1 \cup \mathcal{I}_2 \cup \mathcal{I}_3)| = n_1 - d_1 + 1$. Now, assume that $k_3 > n_1 - d_1 + 1$. Then, there exist at least two distinct codewords $\boldsymbol{c}_3'$ and $\boldsymbol{c}_3''$ in $\mathcal{C}_3$ that agree on the coordinates in the set $\mathcal{I} \backslash (\mathcal{I}_1 \cup \mathcal{I}_2 \cup \mathcal{I}_3)$. We erase the values on the coordinates in the set $\mathcal{I}_1 \cup \mathcal{I}_2 \cup \mathcal{I}_3$ of both $\boldsymbol{c}_3'$ and $\boldsymbol{c}_3''$. This erasure pattern satisfies the condition in (7). Since $\boldsymbol{c}_3'$ and $\boldsymbol{c}_3''$ are distinct, this erasure pattern is uncorrectable. Thus, our assumption that $k_3 > n_1 - d_1 + 1$ is violated. ∎

The following example and lemma generalize Example 3 and Lemma 7, respectively, from the three-level case to the $M$-level case.

**Example 4.** Consider a set of $M$ nested MDS codes $\mathcal{C}_1^M \subset \mathcal{C}_1^{M-1} \subset \cdots \subset \mathcal{C}_1^1 = \mathcal{C}_1 = [n_1, n_1 - d_1 + 1, d_1]_q$, where $\mathcal{C}_1^i = [n_1, n_1 - d_i + 1, d_i]_q$ for $1 \leqslant i \leqslant M$, and $d_1 < d_2 < \cdots < d_M$.

For level $i$, $2 \leqslant i \leqslant M$, choose a set of auxiliary nested MDS codes $\mathcal{A}_i^M \subset \mathcal{A}_i^{M-1} \subset \cdots \subset \mathcal{A}_i^{i+1} \subset \mathcal{A}_i^i$, where $\mathcal{A}_i^j = [i(d_i - d_{i-1}) - 1, id_i - (i-1)d_{i-1} - d_j, d_j - d_{i-1}]_q$ for $i \leqslant j \leqslant M$.

Assume that $n_1 - d_M + 1 > 0$ and $id_i - (i-1)d_{i-1} - d_M > 0$ for all $2 \leqslant i \leqslant M$. From Construction 1 and Theorem 5, we obtain $M$-level rate-compatible codes $\mathcal{C}_1 = [n_1, n_1 - d_1 + 1, d_1]_q \prec \mathcal{C}_2 = [n_1 + 2(d_2 - d_1) - 1, n_1 - d_1 + 1, d_2]_q \prec \cdots \prec \mathcal{C}_M = [n_1 + \sum_{i=2}^M \Big(i(d_i - d_{i-1}) - 1\Big), n_1 - d_1 + 1, d_M]_q$.

From the condition in (6) and Theorem 6, the code $\mathcal{C}_M$ can correct any pattern of errors and erasures satisfying

$$2t_i + \tau_i \leqslant d_M - d_{i-1} - 1, \ \forall\ 1 \leqslant i \leqslant M, \tag{8}$$

where $d_0$ is defined to be 0. □

**Lemma 8.** *Let $\mathcal{C}_M$ be a code of length $n_1 + \sum_{i=2}^{M}\Big(i(d_i - d_{i-1}) - 1\Big)$ and dimension $k_M$ over $\mathbb{F}_q$. Each codeword $\boldsymbol{c}_M \in \mathcal{C}_M$ has $M$ sub-blocks $(\boldsymbol{c}_1, \boldsymbol{a}_2^2, \ldots, \boldsymbol{a}_M^M)$: 1) $\boldsymbol{c}_1$ of length $n_1$, and 2) $\boldsymbol{a}_i^i$ of length $i(d_i - d_{i-1}) - 1$, $2 \leqslant i \leqslant M$. Assume that each sub-block of $\mathcal{C}_M$ can correct all error and erasure patterns satisfying the condition in (8). Then, we must have $k_M \leqslant n_1 - d_1 + 1$; that is, the code $\mathcal{C}_M$ constructed in Example 4 is optimal.*

*Proof:* The proof follows the same argument as that of Lemma 7, so it is omitted. ■

### C. Decoding Error Probability Analysis

For a linear code $\mathcal{C} = [n, k, d]_q$ over a QSC, let $P_e^{(n)}(\boldsymbol{x})$ denote the conditional block probability of error, assuming that $\boldsymbol{x}$ was sent, $\boldsymbol{x} \in \mathcal{C}$. Let $P_e^{(n)}(\mathcal{C})$ denote the average probability of error of this code. Due to symmetry, assuming equiprobable codewords, it is clear that,

$$P_e^{(n)}(\mathcal{C}) = \frac{1}{|\mathcal{C}|}\sum_{\boldsymbol{x}\in\mathcal{C}} P_e^{(n)}(\boldsymbol{x}) = P_e^{(n)}(\boldsymbol{x}).$$

In Algorithm 1, the code $\mathcal{C}_M$ is decoded by $M$ steps, so the decoding will be successful if each decoding step is correct. Therefore, the probability of successful decoding $P_s^{(N_M)}(\mathcal{C}_M)$ of $\mathcal{C}_M$ can be bounded as

$$P_s^{(N_M)}(\mathcal{C}_M) \geqslant \Big(1 - P_e^{(n_1)}(\mathcal{C}_1^M)\Big)\prod_{i=2}^{M}\Big(1 - P_e^{(n_i)}(\mathcal{A}_i^M)\Big).$$

Correspondingly, we can bound the decoding error probability $P_e^{(N_M)}(\mathcal{C}_M)$ of $\mathcal{C}_M$ by the decoding error probability of each step as

$$\begin{aligned} P_e^{(N_M)}(\mathcal{C}_M) &= 1 - P_s^{(N_M)}(\mathcal{C}_M) \\ &\leqslant 1 - \Big(1 - P_e^{(n_1)}(\mathcal{C}_1^M)\Big)\prod_{i=2}^{M}\Big(1 - P_e^{(n_i)}(\mathcal{A}_i^M)\Big), \end{aligned}$$

which provides a fast way to predict the performance of $\mathcal{C}_M$. In particular, if each component code is a (shortened) BCH code, then $P_e^{(N_M)}(\mathcal{C}_M)$ can be easily bounded, as illustrated in the following example.

**Example 5.** Consider two nested binary BCH codes $\mathcal{C}_1^2 = [8191, 7411]_2 \subset \mathcal{C}_1^1 = [8191, 7671]_2$. The codes $\mathcal{C}_1^1$ and $\mathcal{C}_1^2$ can correct 40 and 60 errors, respectively. Choose an auxiliary shortened BCH code $\mathcal{A}_2^2 = [359, 260]_2$, which can correct 11 errors. Then, from Construction 1, we obtain two-level rate-compatible codes $\mathcal{C}_1 = [8191, 7671]_2 \prec \mathcal{C}_2 = [8550, 7671]_2$. Now, send $\mathcal{C}_2$ over a binary symmetric channel (BSC) with crossover probability $p$. The error probability of $\mathcal{C}_2$ satisfies

$$\begin{aligned} P_e^{(N_2)}(\mathcal{C}_2) &\leqslant 1 - \Big(1 - P_e^{(n_1)}(\mathcal{C}_1^2)\Big)\Big(1 - P_e^{(n_2)}(\mathcal{A}_2^2)\Big) \\ &\leqslant 1 - \Big(\sum_{i=0}^{t_1}\binom{n_1}{i}p^i(1-p)^{n_1-i}\Big)\Big(\sum_{i=0}^{t_2}\binom{n_2}{i}p^i(1-p)^{n_2-i}\Big), \end{aligned}$$

where $N_2 = 8550$, $n_1 = 8191$, $n_2 = 359$, $t_1 = 60$, and $t_2 = 11$. For instance, for $p = 0.0035$, we compute $P_e^{(N_2)}(\mathcal{C}_2) \leqslant 1.049 \times 10^{-7}$; for $p = 0.004$, we have $P_e^{(N_2)}(\mathcal{C}_2) \leqslant 6.374 \times 10^{-6}$. For $p \geqslant 0.0035$, the performance of $\mathcal{C}_2$ (rate 0.8972) is comparable to, although still worse than, that of a $[8553, 7671]_2$ shortened BCH code $\mathcal{C}_2'$ which has rate 0.8969 and can correct 63 errors. For instance, for $p = 0.0035$ and 0.004, $\mathcal{C}_2'$ has error probabilities $4.035 \times 10^{-8}$ and $3.315 \times 10^{-6}$. □

**Remark 4.** The discussion above is readily extended to the QEC.

## V. Capacity-Achieving Rate-Compatible Codes

In this section, we show that if we choose component codes properly, Construction 1 can generate capacity-achieving rate-compatible codes for a set of QMS channels with decreasing channel capacities. This implies the optimality of our construction from a probabilistic point of view.

### A. Capacity-Achieving Rate-Compatible Codes for QMS Channels

Given a family of $M$ QMS channels $W_1, W_2, \ldots, W_M$ with capacities $C(W_1) > C(W_2) > \cdots > C(W_M)$. For any rates $R_1 > R_2 > \cdots > R_M$ such that $R_i < C(W_i)$ for all $1 \leqslant i \leqslant M$, our goal is to construct capacity-achieving rate-compatible codes $\mathcal{C}_1 \prec \mathcal{C}_2 \prec \cdots \prec \mathcal{C}_M$ where $\mathcal{C}_i = [N_i, R_i N_i]_q$ such that the decoding error probability of $\mathcal{C}_i$ over the QMS channel $W_i$ satisfies $P_e^{(N_i)}(\mathcal{C}_i) \to 0$, as $N_i$ goes to infinity.

An example of such a family is a set of $M$ degraded QSCs $W_1 \succ W_2 \succ \cdots \succ W_M$ with crossover probabilities $p_1 < p_2 < \cdots < p_M$ respectively, where $p_1 > 0$ and $p_M < 1 - (1/q)$. Let $C(W_i)$ denote the capacity of the QSC $W_i$, i.e., $C(W_i) = 1 - H_q(p_i)$. It is clear that $C(W_1) > C(W_2) > \cdots > C(W_M)$.

We first present the following proposition showing the existence of nested capacity-achieving linear codes over a set of QMS channels with decreasing channel capacities.

**Proposition 9.** *Consider a family of $M$ QMS channels $W_1, W_2, \ldots, W_M$ with capacities $C(W_1) > C(W_2) > \cdots > C(W_M)$. For any rates $R_1 > R_2 > \cdots > R_M$ such that $R_i < C(W_i)$, there exists a sequence of nested linear codes $\mathcal{C}_1^M = [n, k_M = R_M n]_q \subset \mathcal{C}_1^{M-1} = [n, k_{M-1} = R_{M-1} n]_q \subset \cdots \subset \mathcal{C}_1^1 = [n, k_1 = R_1 n]_q$ such that the decoding error probability of each $\mathcal{C}_1^i$ over the QMS channel $W_i$, under maximum-likelihood decoding, satisfies $P_e^{(n)}(\mathcal{C}_1^i) \to 0$, as $n$ goes to infinity.*

To prove the proposition, we use the following result, which is a direct consequence of Theorem 6.2.1 of Gallager [11, p. 206].

**Lemma 10.** *For a QMS channel $W$, let $n$ and $nR$ be integers such that $R < C(W)$. Let $\overline{P_e^{(n)}(\mathcal{C})}$ denote the average of $P_e^{(n)}(\mathcal{C})$ over*

*all linear $[n, nR]_q$ codes $\mathcal{C}$ under maximum-likelihood decoding. Then,*

$$\overline{P_e^{(n)}(\mathcal{C})} \leqslant q^{-nE_W(R)},$$

*where $E_W(R)$ is called Gallager's random coding error exponent and $E_W(R) > 0$ for all $R$ satisfying $0 \leqslant R < C(W)$.*

The following lemma is a direct consequence of Lemma 10.

**Lemma 11.** *For every $\rho \in (0, 1]$, at least a fraction $1 - \rho$ (i.e., $\geqslant 1 - \rho$) of all linear $[n, nR]_q$ codes $\mathcal{C}$ satisfy*

$$P_e^{(n)}(\mathcal{C}) \leqslant (1/\rho)q^{-nE_W(R)}.$$

*Proof:* Consider the set $\mathcal{S}$ of codes $\mathcal{C}$ for which $P_e^{(n)}(\mathcal{C}) > (1/\rho)q^{-nE_W(R)}$. Assume to the contrary that $\mathcal{S}$ forms more than a fraction $\rho$ of all linear $[n, nR]_q$ codes $\mathcal{C}$. Then, we have

$$\overline{P_e^{(n)}(\mathcal{C})} > \frac{\rho}{|\mathcal{S}|} \sum_{\mathcal{C} \in \mathcal{S}} P_e^{(n)}(\mathcal{C}) > q^{-nE_W(R)},$$

contradicting Lemma 10. Therefore, $\mathcal{S}$ only forms at most a fraction $\rho$ of all linear $[n, nR]_q$ codes $\mathcal{C}$. ∎

Using Lemmas 10 and 11, we now prove Proposition 9.

*Proof:* Consider an ensemble $\mathcal{G}_1$ of all $k_1 \times n$ full rank matrices over $\mathbb{F}_q$. The size of $\mathcal{G}_1$ is $|\mathcal{G}_1| = (q^n - 1)(q^n - q) \cdots (q^n - q^{k_1 - 1})$. Now, for each matrix $G_i^1 \in \mathcal{G}_1$, $1 \leqslant i \leqslant |\mathcal{G}_1|$, take the lowest $k_2$ rows to form a new matrix $G_i^2$. All these new matrices form a new ensemble $\mathcal{G}_2$, including possible repetitions. It is clear that $|\mathcal{G}_2| = |\mathcal{G}_1|$ and in $\mathcal{G}_2$, each $k_2 \times n$ full rank matrix over $\mathbb{F}_q$ appears $(q^n - q^{k_2})(q^n - q^{k_2+1}) \cdots (q^n - q^{k_1-1})$ times. Similarly, for each matrix $G_i^1 \in \mathcal{G}_1$, $1 \leqslant i \leqslant |\mathcal{G}_1|$, take the lowest $k_j$, $3 \leqslant j \leqslant M$, rows to form a new matrix $G_i^j$. All these new matrices form a new ensemble $\mathcal{G}_j$. It is clear that $|\mathcal{G}_j| = |\mathcal{G}_1|$ and in $\mathcal{G}_j$, each $k_j \times n$ full rank matrix over $\mathbb{F}_q$ appears $(q^n - q^{k_j})(q^n - q^{k_j+1}) \cdots (q^n - q^{k_1-1})$ times.

Note that the number of generator matrices of a linear $[n, k]_q$ code is the same for all such codes. Therefore, from Lemma 11, in each ensemble $\mathcal{G}_j$ for $1 \leqslant j \leqslant M$, at least a fraction $x$ of all matrices in this ensemble will generate linear codes $\mathcal{C}$ such that the error probability $P_e^{(n)}(\mathcal{C}) \leqslant (\frac{1}{1-x})q^{-nE_{W_j}(R_j)}$.

Now, choose $x$ to be a certain value satisfying $\frac{1}{2} < x < 1$. Let $\mathcal{S}_1$ be the subset of the ensemble $\mathcal{G}_1$ such that $\frac{|\mathcal{S}_1|}{|\mathcal{G}_1|} \geqslant x$ and each matrix in $\mathcal{S}_1$ generates a linear code $\mathcal{C}_1^1$ with the error probability $P_e^{(n)}(\mathcal{C}_1^1) \leqslant (\frac{1}{1-x})q^{-nE_{W_1}(R_1)}$. Let $\mathcal{S}_2$ be the subset of the ensemble $\mathcal{G}_1$ such that $\frac{|\mathcal{S}_2|}{|\mathcal{G}_1|} \geqslant x$ and for each matrix in $\mathcal{S}_2$, its lowest $k_2$ rows generate a linear code $\mathcal{C}_1^2$ with the error probability $P_e^{(n)}(\mathcal{C}_1^2) \leqslant (\frac{1}{1-x})q^{-nE_{W_2}(R_2)}$. Then, with the basic property on set operations, we have

$$\begin{aligned}\frac{|\mathcal{S}_1 \cap \mathcal{S}_2|}{|\mathcal{G}_1|} &= \frac{|\mathcal{S}_1|}{|\mathcal{G}_1|} + \frac{|\mathcal{S}_2|}{|\mathcal{G}_1|} - \frac{|\mathcal{S}_1 \cup \mathcal{S}_2|}{|\mathcal{G}_1|} \\ &\geqslant \frac{|\mathcal{S}_1|}{|\mathcal{G}_1|} + \frac{|\mathcal{S}_2|}{|\mathcal{G}_1|} - 1 \\ &\geqslant 2x - 1 > 0.\end{aligned}$$

Thus, we find a non-empty subset $\mathcal{S}_{12} = \mathcal{S}_1 \cap \mathcal{S}_2$ in the ensemble $\mathcal{G}_1$ such that: 1) $\mathcal{S}_{12}$ has at least a fraction $2x - 1 > 0$ of all the matrices in $\mathcal{G}_1$, and 2) for each matrix in $\mathcal{S}_{12}$, it generates a linear code $\mathcal{C}_1^1$ with the error probability $P_e^{(n)}(\mathcal{C}_1^1) \leqslant (\frac{1}{1-x})q^{-nE_{W_1}(R_1)}$ and its lowest $k_2$ rows generate a linear code $\mathcal{C}_1^2$ with the error probability $P_e^{(n)}(\mathcal{C}_1^2) \leqslant (\frac{1}{1-x})q^{-nE_{W_2}(R_2)}$.

Similarly, following from the above argument, it is not hard to see that for any $x$ satisfying $\frac{M-1}{M} < x < 1$, in the ensemble $\mathcal{G}_1$, we can find a non-empty subset $\overline{\mathcal{G}}_1 \subseteq \mathcal{G}_1$ such that: 1) $\overline{\mathcal{G}}_1$ has at least a fraction $Mx - (M-1) > 0$ of all the matrices in $\mathcal{G}_1$, and 2) for each matrix $\overline{G}_1$ in $\overline{\mathcal{G}}_1$, for each $j$, $1 \leqslant j \leqslant M$, the lowest $k_j$ rows of $\overline{G}_1$ will generate a linear code $\mathcal{C}_1^j$ with the error probability $P_e^{(n)}(\mathcal{C}_1^j) \leqslant (\frac{1}{1-x})q^{-nE_{W_j}(R_j)}$.

Thus, there exists a sequence of nested linear codes $\mathcal{C}_1^M = [n, k_M = R_M n]_q \subset \mathcal{C}_1^{M-1} = [n, k_{M-1} = R_{M-1} n]_q \subset \cdots \subset \mathcal{C}_1^1 = [n, k_1 = R_1 n]_q$ such that for all $1 \leqslant i \leqslant M$, the error probability $P_e^{(n)}(\mathcal{C}_1^i) \to 0$, as $n$ goes to infinity. ∎

Now, we are ready to construct capacity-achieving rate-compatible codes from Construction 1. To do so, we choose a set of nested capacity-achieving codes, whose existence is guaranteed by Proposition 9, to be the component codes, as follows.

1) Choose a set of nested capacity-achieving codes $\mathcal{C}_1^M \subset \mathcal{C}_1^{M-1} \subset \cdots \subset \mathcal{C}_1^1 = \mathcal{C}_1 = [n_1, k]_q$, where $\mathcal{C}_1^i = [n_1, n_1 - \sum_{m=1}^{i} v_m]_q$ for $1 \leqslant i \leqslant M$. Let $\mathcal{C}_1^i$ have the required rate $R_i < C(W_i)$, such that the decoding error probability for $\mathcal{C}_1^i$ over the QMS channel $W_i$ satisfies $P_e^{(n_1)}(\mathcal{C}_1^i) \to 0$, as $n_1$ goes to infinity.

2) For $i$th level, $2 \leqslant i \leqslant M$, choose a set of auxiliary nested capacity-achieving codes $\mathcal{A}_i^M \subset \mathcal{A}_i^{M-1} \subset \cdots \subset \mathcal{A}_i^{i+1} \subset \mathcal{A}_i^i$, where $\mathcal{A}_i^j = [n_i, v_i + \sum_{m=2}^{i-1} \lambda_m^i - \sum_{\ell=i+1}^{j} \lambda_i^\ell]_q$ for $i \leqslant j \leqslant M$. Let $\mathcal{A}_i^j$ have the required rate $R_j < C(W_j)$, such that the decoding error probability for $\mathcal{A}_i^j$ over the QMS channel $W_j$ satisfies $P_e^{(n_i)}(\mathcal{A}_i^j) \to 0$, as $n_i$ goes to infinity.

Note that, in contrast to Section IV, here we care about the rate and capacity-achieving properties, rather than the minimum distance properties, of the component codes.

**Theorem 12.** *With the above component codes, from Construction 1, we obtain a sequence of capacity-achieving rate-compatible codes $\mathcal{C}_1 \prec \mathcal{C}_2 \prec \cdots \prec \mathcal{C}_M$, where $\mathcal{C}_i$, $1 \leqslant i \leqslant M$, has length $N_i = \sum_{j=1}^{i} n_j$, dimension $K_i = k$, and rate $R_i$. Moreover, for $\mathcal{C}_i$ over the QMS channel $W_i$, the error probability $P_e^{(N_i)}(\mathcal{C}_i) \to 0$, as $N_i$ goes to infinity.*

*Proof:* The code length and dimension of $\mathcal{C}_i$ are obvious. We iteratively show that the rate of $\mathcal{C}_i$ is given by $R_i = \frac{k}{N_i} = \frac{k}{\sum_{j=1}^{i} n_j}$. For $i = 1$, the result is trivial, since the rate of $\mathcal{C}_1^1$ is $R_1$. For $i = 2$, observe that the rate of $\mathcal{C}_1^2$ is $R_2 = \frac{k - v_2}{n_1}$ and the rate of $\mathcal{A}_2^2$ is $R_2 = \frac{v_2}{n_2}$, so we have $(n_1 + n_2)R_2 = k$, or $R_2 = \frac{k}{N_i} = \frac{k}{\sum_{j=1}^{2} n_j}$. Similarly, for $3 \leqslant i \leqslant M$, we use the rates of the

codes $\mathcal{C}_1^i$, $\mathcal{A}_2^i$, …, $\mathcal{A}_i^i$ to show that $(n_1+n_2+\cdots+n_i)R_i = k$, or $R_i = \frac{k}{N_i} = \frac{k}{\sum_{j=1}^{i} n_j}$, as desired.

We now confirm that the error probabilities go to zero. We prove the result for $\mathcal{C}_M$; the same proof will work for any $\mathcal{C}_i$, $1 \leqslant i \leqslant M-1$.

For code $\mathcal{C}_M$ over the channel $W_M$, we use Algorithm 1 for decoding, where each component decoder is chosen to be a maximum-likelihood decoder. The decoding for $\mathcal{C}_M$ consists of $M$ steps, so it will succeed if each step is successful. Thus, we can bound the decoding error probability $P_e^{(N_M)}(\mathcal{C}_M)$ by the decoding error probability of each step as

$$\begin{aligned} & P_e^{(N_M)}(\mathcal{C}_M) \\ \leqslant & 1-\left(1-P_e^{(n_1)}(\mathcal{C}_1^M)\right)\prod_{i=2}^{M}\left(1-P_e^{(n_i)}(\mathcal{A}_i^M)\right) \\ = & 1-\left(1-P_e^{(\phi_1 N_M)}(\mathcal{C}_1^M)\right)\prod_{i=2}^{M}\left(1-P_e^{(\phi_i N_M)}(\mathcal{A}_i^M)\right) \end{aligned} \tag{9}$$

where the constants $\phi_1 = \frac{R_M}{R_1}$ and $\phi_i = \frac{(R_{i-1}-R_i)R_M}{R_i R_{i-1}}$ for $2 \leqslant i \leqslant M$. From the chosen capacity-achieving component codes, we already have $P_e^{(\phi_1 N_M)}(\mathcal{C}_1^M) \to 0$ and $P_e^{(\phi_i N_M)}(\mathcal{A}_i^M) \to 0$ as $N_M$ goes to infinity, so in (9), $P_e^{(N_M)}(\mathcal{C}_M) \to 0$ as $N_M$ goes to infinity. Thus, we conclude that $\mathcal{C}_M$ can achieve the capacity of the channel $W_M$. ∎

**Remark 5.** In the proof of Theorem 12, when the channels are QSCs, we can use a nearest-codeword decoder, defined as follows.

1) The nearest-codeword decoder $\mathcal{D}_{\mathcal{C}_1^i}$ for a coset of the code $\mathcal{C}_1^i$, for $1 \leqslant i \leqslant M$, is defined by

$$\mathcal{D}_{\mathcal{C}_1^i} : \mathbb{F}_q^{n_1} \times \mathbb{F}_q^{\sum_{j=1}^{i} v_j} \to \mathcal{C}_1^i + \boldsymbol{e}$$

according to the following decoding rules: for a length-$n_1$ input vector $\boldsymbol{y}$, and a length-$\sum_{j=1}^{i} v_j$ syndrome vector $\boldsymbol{s}$, if $\boldsymbol{c}$ is a closest codeword to $\boldsymbol{y}$ in the coset $\mathcal{C}_1^i + \boldsymbol{e}$, where the vector $\boldsymbol{e}$ is a coset leader determined by both the code $\mathcal{C}_1^i$ and the syndrome vector $\boldsymbol{s}$, i.e., $\boldsymbol{s} = \boldsymbol{e}H_{\mathcal{C}_1^i}^T$, then $\mathcal{D}_{\mathcal{C}_1^i}(\boldsymbol{y}, \boldsymbol{s}) = \boldsymbol{c}$.

2) The nearest-codeword decoder $\mathcal{D}_{\mathcal{A}_i^j}$ for a coset of the code $\mathcal{A}_i^j$, for $2 \leqslant i \leqslant M$ and $i \leqslant j \leqslant M$, is defined by

$$\mathcal{D}_{\mathcal{A}_i^j} : \mathbb{F}_q^{n_i} \times \mathbb{F}_q^{n_i - v_i - \sum_{m=2}^{i-1} \lambda_m^i + \sum_{\ell=i+1}^{j} \lambda_i^\ell} \to \mathcal{A}_i^j + \boldsymbol{e}$$

according to the following decoding rules: for a length-$n_i$ input vector $\boldsymbol{y}$, and a length-$(n_i - v_i - \sum_{m=2}^{i-1} \lambda_m^i + \sum_{\ell=i+1}^{j} \lambda_i^\ell)$ syndrome vector $\boldsymbol{s}$, if $\boldsymbol{c}$ is a closest codeword to $\boldsymbol{y}$ in the coset $\mathcal{A}_i^j + \boldsymbol{e}$, where the vector $\boldsymbol{e}$ is a coset leader determined by both the code $\mathcal{A}_i^j$ and the syndrome vector $\boldsymbol{s}$, i.e., $\boldsymbol{s} = \boldsymbol{e}H_{\mathcal{A}_i^j}^T$, then $\mathcal{D}_{\mathcal{A}_i^j}(\boldsymbol{y}, \boldsymbol{s}) = \boldsymbol{c}$.

**Remark 6.** Polar codes are a family of linear codes that provably achieve the capacity of memoryless symmetric channels using low-complexity encoding and decoding algorithms [2], [31], [32]. Moreover, binary polar codes were proved to have the nested capacity-achieving property for a set of degraded binary memoryless symmetric (BMS) channels [18, Lemma 4.7]. Thus, they can be used as the component codes in Construction 1 to construct capacity-achieving rate-compatible codes for a set of degraded BSCs. This can be extended to a set of BMS channels with ordered capacities, and more generally to such a set of QMS channels using results and remarks in [27].

### *B. Equivalence to HHM Construction of Polar Codes*

There are several recent independent works on capacity-achieving rateless and rate-compatible codes based on polar codes [15], [21], [27]. An analysis of the construction in [15], which we refer to as the *HHM construction*, shows that it is equivalent to our syndrome-coupled code construction with polar codes as component codes through a mapping of the *syndrome* in our construction to the *information bits* in that construction. In the following, we show this equivalence for the case of three-level rate-compatible codes. Extension to the $M$-level case is similar, so the details are omitted.

We first introduce the HHM construction [15] for generating three-level rate-compatible codes, based on Arikan's construction of polar codes [2]. It has the following three steps:

**Step 1**: Begin with three nested binary polar codes $\mathcal{C}_1^3 \subset \mathcal{C}_1^2 \subset \mathcal{C}_1^1$, where $\mathcal{C}_1^i = [n_1, n_1 - \sum_{m=1}^{i} v_m]_2$ for $1 \leqslant i \leqslant 3$.

The set of information bit indices of $\mathcal{C}_1^i$ is denoted by $\mathcal{I}_1^i$, for $1 \leqslant i \leqslant 3$. The nested property of polar codes gives $\mathcal{I}_1^3 \subset \mathcal{I}_1^2 \subset \mathcal{I}_1^1$ [18]. Let $\mathcal{N}_1^2 = \mathcal{I}_1^1 \backslash \mathcal{I}_1^2$ and $\mathcal{N}_1^3 = \mathcal{I}_1^2 \backslash \mathcal{I}_1^3$, and it is clear that $|\mathcal{N}_1^2| = v_2$ and $|\mathcal{N}_1^3| = v_3$.

Let a length-$n_1$ vector $\bar{\boldsymbol{u}}_1$ have $k = n_1 - v_1$ information bits $\boldsymbol{u}$ on the coordinates in $\mathcal{I}_1^1$ and value 0 on the coordinates in $[n_1] \backslash \mathcal{I}_1^1$. A codeword $\boldsymbol{c}_1 \in \mathcal{C}_1$ is obtained by $\boldsymbol{c}_1 = \bar{\boldsymbol{u}}_1 G_{n_1}$. Here, we assume that the code length $n_1$ is $n_1 = 2^{m_1}$ and the matrix $G_{n_1}$ is $G_{n_1} = \mathrm{B}_{n_1} G_2^{\otimes m_1}$, where $G_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$ and $\mathrm{B}_{n_1}$ is a bit-reversal permutation matrix defined in [2]. It is known that $G_{n_1} = G_{n_1}^{-1}$, i.e., $G_{n_1} G_{n_1} = I$ [12].

**Step 2**: Now take two nested binary polar codes $\mathcal{A}_2^3 = [n_2, v_2 - \lambda_2^3]_2 \subset \mathcal{A}_2^2 = [n_2, v_2]_2$.

The set of information bit indices of $\mathcal{A}_2^i$ is denoted by $\mathcal{I}_2^i$, $i = 2, 3$. The sets $\mathcal{I}_2^2$ and $\mathcal{I}_2^3$ satisfy $\mathcal{I}_2^3 \subset \mathcal{I}_2^2$. Let the index set $\mathcal{N}_2^3 = \mathcal{I}_2^2 \backslash \mathcal{I}_2^3$ and we have $|\mathcal{N}_2^3| = \lambda_2^3$.

Let a length-$n_2$ vector $\bar{\boldsymbol{u}}_2$ have $\pi_{\mathcal{N}_1^2}(\bar{\boldsymbol{u}}_1)$ on the coordinates in $\mathcal{I}_2^2$ and value 0 on the coordinates in $[n_2] \backslash \mathcal{I}_2^2$. A codeword $\boldsymbol{a}_2^2 \in \mathcal{A}_2^2$ is obtained by $\boldsymbol{a}_2^2 = \bar{\boldsymbol{u}}_2 G_{n_2}$. Here, we assume that the code length $n_2$ is $n_2 = 2^{m_2}$ and the matrix $G_{n_2}$ is $G_{n_2} = \mathrm{B}_{n_2} G_2^{\otimes m_2}$, where $\mathrm{B}_{n_2}$ is a bit-reversal permutation matrix defined in [2] and it is known that $G_{n_2} G_{n_2} = I$ [12].

**Step 3**: Use a binary polar code $\mathcal{A}_3^3 = [n_3, v_3 + \lambda_2^3]_2$ to encode $\pi_{\mathcal{N}_1^3}(\bar{\boldsymbol{u}}_1)$ and $\pi_{\mathcal{N}_2^3}(\bar{\boldsymbol{u}}_2)$. We denote the set of information bit indices of $\mathcal{A}_3^3$ by $\mathcal{I}_3^3$. Let a length-$n_3$ vector

$\bar{\boldsymbol{u}}_3$ have $\pi_{\mathcal{N}_1^3}(\bar{\boldsymbol{u}}_1)$ and $\pi_{\mathcal{N}_2^3}(\bar{\boldsymbol{u}}_2)$ on the coordinates in $\mathcal{I}_3^3$ and value 0 on the coordinates in $[n_3]\backslash\mathcal{I}_3^3$. A codeword $\boldsymbol{a}_3^3 \in \mathcal{A}_3^3$ is obtained by $\boldsymbol{a}_3^3 = \bar{\boldsymbol{u}}_3 G_{n_3}$, where $G_{n_3} = \mathrm{B}_{n_3} G_2^{\otimes m_3}$ and $\mathrm{B}_{n_3}$ is a bit-reversal permutation matrix. □

In summary, the preceding encoding procedure of the HHM construction generates

$$\begin{aligned}&\left(\boldsymbol{c}_1, \boldsymbol{a}_2^2, \boldsymbol{a}_3^3\right)\\ =&\left(\mathcal{E}_{\mathcal{C}_1^1}(\boldsymbol{u}), \mathcal{E}_{\mathcal{A}_2^2}\left(\pi_{\mathcal{N}_1^2}(\bar{\boldsymbol{u}}_1)\right), \mathcal{E}_{\mathcal{A}_3^3}\left(\pi_{\mathcal{N}_1^3}(\bar{\boldsymbol{u}}_1), \pi_{\mathcal{N}_2^3}(\bar{\boldsymbol{u}}_2)\right)\right),\end{aligned} \tag{10}$$

where $\mathcal{E}_{\mathcal{C}_1^1}$, $\mathcal{E}_{\mathcal{A}_2^2}$, and $\mathcal{E}_{\mathcal{A}_3^3}$ denote the encoders of $\mathcal{C}_1^1$, $\mathcal{A}_2^2$, and $\mathcal{A}_3^3$, respectively.

Now, consider our syndrome-coupled construction, i.e., Construction 1. We use the same component polar codes as those in the HHM construction. According to Lemma 1 in [12], the rows of the canonical parity-check matrix of a polar code are the columns of its generator matrix corresponding to the frozen indices. Therefore, for the nested binary polar codes $\mathcal{C}_1^3 \subset \mathcal{C}_1^2 \subset \mathcal{C}_1^1$, the canonical parity-check matrices are in the form of (4), where the matrix $H_{\mathcal{C}_1^1}$ is formed by the columns of $G_{n_1}$ with indices in $[n_1]\backslash\mathcal{I}_1^1$, the matrix $H_{\mathcal{C}_1^2|\mathcal{C}_1^1}$ is formed by the columns of $G_{n_1}$ with indices in $\mathcal{N}_1^2$, and the matrix $H_{\mathcal{C}_1^3|\mathcal{C}_1^2}$ is formed by the columns of $G_{n_1}$ with indices in $\mathcal{N}_1^3$. Similarly, for the nested binary polar codes $\mathcal{A}_2^3 \subset \mathcal{A}_2^2$, the canonical parity-check matrices are in the form of (5), where the matrix $H_{\mathcal{A}_2^2}$ is formed by the columns of $G_{n_2}$ with indices in $[n_2]\backslash\mathcal{I}_2^2$, and the matrix $H_{\mathcal{A}_2^3|\mathcal{A}_2^2}$ is formed by the columns of $G_{n_2}$ with indices in $\mathcal{N}_2^3$.

Using the polar codes $\mathcal{C}_1^3 \subset \mathcal{C}_1^2 \subset \mathcal{C}_1^1$, $\mathcal{A}_2^3 \subset \mathcal{A}_2^2$, and $\mathcal{A}_3^3$, according to Construction 1, our syndrome-coupled construction produces:

$$\left(\boldsymbol{c}_1, \boldsymbol{a}_2^2, \boldsymbol{a}_3^3\right) = \left(\mathcal{E}_{\mathcal{C}_1^1}(\boldsymbol{u}), \mathcal{E}_{\mathcal{A}_2^2}(\boldsymbol{s}_2), \mathcal{E}_{\mathcal{A}_3^3}(\boldsymbol{s}_3, \Lambda_2^3)\right), \tag{11}$$

where $\boldsymbol{s}_2 = \bar{\boldsymbol{u}}_1 G_{n_1} H_{\mathcal{C}_1^2|\mathcal{C}_1^1}^T$, $\boldsymbol{s}_3 = \bar{\boldsymbol{u}}_1 G_{n_1} H_{\mathcal{C}_1^3|\mathcal{C}_1^2}^T$, and $\Lambda_2^3 = \bar{\boldsymbol{u}}_2 G_{n_2} H_{\mathcal{A}_2^3|\mathcal{A}_2^2}^T$.

By comparing (10) and (11), we have the following equivalence result on the information bits of the HHM construction and the syndrome of our construction.

**Theorem 13.** *The HHM construction is equivalent to our syndrome-coupled construction (i.e., Construction 1); that is, we have* $\pi_{\mathcal{N}_1^2}(\bar{\boldsymbol{u}}_1) = \boldsymbol{s}_2$, $\pi_{\mathcal{N}_1^3}(\bar{\boldsymbol{u}}_1) = \boldsymbol{s}_3$, *and* $\pi_{\mathcal{N}_2^3}(\bar{\boldsymbol{u}}_2) = \Lambda_2^3$.

*Proof:* We prove $\pi_{\mathcal{N}_1^2}(\bar{\boldsymbol{u}}_1) = \boldsymbol{s}_2$ and $\pi_{\mathcal{N}_1^3}(\bar{\boldsymbol{u}}_1) = \boldsymbol{s}_3$. A similar proof shows that $\pi_{\mathcal{N}_2^3}(\bar{\boldsymbol{u}}_2) = \Lambda_2^3$, so details are omitted.

First, we have the following equations,

$$\begin{aligned}\boldsymbol{s}_2 &= \boldsymbol{c}_1 H_{\mathcal{C}_1^2|\mathcal{C}_1^1}^T\\ &= \bar{\boldsymbol{u}}_1 G_{n_1} H_{\mathcal{C}_1^2|\mathcal{C}_1^1}^T\\ &\overset{(a)}{=} \pi_{\mathcal{N}_1^2}(\bar{\boldsymbol{u}}_1),\end{aligned}$$

where step (a) follows from the fact that: 1) the matrix $H_{\mathcal{C}_1^2|\mathcal{C}_1^1}$ is formed by the columns of $G_{n_1}$ with indices in $\mathcal{N}_1^2$, and 2) $G_{n_1} G_{n_1} = I$ [12].

Second, we have the following equations,

$$\begin{aligned}\boldsymbol{s}_3 &= \boldsymbol{c}_1 H_{\mathcal{C}_1^3|\mathcal{C}_1^2}^T\\ &= \bar{\boldsymbol{u}}_1 G_{n_1} H_{\mathcal{C}_1^3|\mathcal{C}_1^2}^T\\ &\overset{(a)}{=} \pi_{\mathcal{N}_1^3}(\bar{\boldsymbol{u}}_1),\end{aligned}$$

where step (a) follows from the fact that : 1) the matrix $H_{\mathcal{C}_1^3|\mathcal{C}_1^2}$ is formed by the columns of $G_{n_1}$ with indices in $\mathcal{N}_1^3$, and 2) $G_{n_1} G_{n_1} = I$ [12]. ■

Since the HHM construction [15] is based on the generator matrix, our construction can be seen as another interpretation of the HHM construction from a parity-check matrix perspective.

### *C. Equivalence between Syndrome and Information Symbols of Linear Codes*

Note that in the preceding proof of Theorem 13, we use the special relation between the generator matrix and the canonical parity-check matrix of polar codes [12]. In [15], the authors also used nested punctured polar codes as component codes in the HHM construction to relax the constraint that the code lengths be a power of two. However, for a punctured polar code, a simple description of a canonical parity-check matrix in terms of the generator matrix is not known. Therefore, to prove the equivalence between the HHM construction and our syndrome-coupled construction using punctured polar codes, we prove the more general result that for a set of nested linear codes (including nested punctured polar codes) with given generator matrices, we can always find a set of parity-check matrices such that the information symbols are identical to the corresponding syndrome. To describe this relationship more precisely, we first introduce some notation.

Consider $M$-level nested codes $\mathcal{C}_M \subset \mathcal{C}_{M-1} \subset \cdots \subset \mathcal{C}_1$, where $\mathcal{C}_i = [n, n - \sum_{m=1}^{i} v_m]_q$ for $1 \leqslant i \leqslant M$. Assume that $\mathcal{C}_i$ has the following generator matrix:

$$G(\mathcal{C}_i) = \begin{bmatrix} G_i \\ G_{i+1} \\ \vdots \\ G_M \end{bmatrix}, \tag{12}$$

where for $i \leqslant \ell \leqslant M-1$, the submatrix $G_\ell$ is a $v_{\ell+1} \times n$ matrix over $\mathbb{F}_q$ and $G_M$ is an $(n - \sum_{m=1}^{M} v_m) \times n$ matrix

over $\mathbb{F}_q$. The corresponding parity-check matrix of $\mathcal{C}_i$ has the following form:

$$H(\mathcal{C}_i) = \begin{bmatrix} H_1 \\ H_2 \\ \vdots \\ H_i \end{bmatrix}, \tag{13}$$

where for $1 \leqslant \ell \leqslant i$, the submatrix $H_\ell$ is a $v_\ell \times n$ matrix over $\mathbb{F}_q$.

Let $\boldsymbol{u}$ be an information vector of length $n - v_1$ over $\mathbb{F}_q$. Let the index set $\mathcal{N}_2 = \{1, 2, \ldots, v_2\}$ and for $3 \leqslant i \leqslant M$, $\mathcal{N}_i = \{v_{i-1}+1, \ldots, v_{i-1}+v_i\}$. The following theorem relates the information symbols to corresponding syndromes.

**Theorem 14.** *For a set of $M$-level nested codes $\mathcal{C}_M \subset \mathcal{C}_{M-1} \subset \cdots \subset \mathcal{C}_1$ with given nested generator matrices in (12), there exist nested parity-check matrices in the form of (13) such that for all $2 \leqslant i \leqslant M$, $\pi_{\mathcal{N}_i}(\boldsymbol{u}) = \boldsymbol{s}_i$, where the syndrome $\boldsymbol{s}_i = \boldsymbol{u}G(\mathcal{C}_1)H_i^T$.*

Note that the $M$ matrices $G_1, G_2, \ldots, G_M$ define all the generator matrices of $\mathcal{C}_M \subset \mathcal{C}_{M-1} \subset \cdots \subset \mathcal{C}_1$. Correspondingly, the $M$ matrices $H_1, H_2, \ldots, H_M$ define all the parity-check matrices of these codes.

For a given set of $G_1, G_2, \ldots, G_M$, there exist many choices on $H_1, H_2, \ldots, H_M$ such that the duality between $G(\mathcal{C}_i)$ in (12) and $H(\mathcal{C}_i)$ in (13) is satisfied, for $1 \leqslant i \leqslant M$. However, not all of them can also satisfy the constraint $\pi_{\mathcal{N}_i}(\boldsymbol{u}) = \boldsymbol{s}_i$, for $2 \leqslant i \leqslant M$.

To prove Theorem 14, we make use of the following algorithm, referred to as Algorithm 2, whose inputs are a set of matrices $G_1, G_2, \ldots, G_M$ and a set of matrices $H_1, H_2, \ldots, H_M$ that satisfy the duality between $G(\mathcal{C}_i)$ in (12) and $H(\mathcal{C}_i)$ in (13), for $1 \leqslant i \leqslant M$. Algorithm 2 transforms the matrices $H_1, H_2, \ldots, H_M$ into a new set of matrices $H'_1, H'_2, \ldots, H'_M$.

**Algorithm 2: Parity-Check Matrix Transform**

**Input:** 1) $G_1, G_2, \ldots, G_M$; 2) $H_1, H_2, \ldots, H_M$.

**Output:** $H'_1, H'_2, \ldots, H'_M$.

**for** $i = 1, 2, \ldots, M$ **do**
  Let $H_i(1) = H_i$.
**end for**
**for** $i = 2, 3, \ldots, M$ **do**
  Let $B_i = \left(H_i(1)G_{i-1}^T\right)^{-1}$.
  Let $H_i(2) = B_i H_i(1)$.
**end for**
**for** $i = 3, 4, \ldots, M$ **do**
  $H_i(i) = H_i(2)$.
  **for** $j = 2, 3, \ldots, i-1$ **do**
    Let $T_{j,i} = -H_i(2)G_{j-1}^T$.
    Let $H_i(i) = H_i(i) + T_{j,i}H_j(j)$.
  **end for**
**end for**
**for** $i = 1, 2, \ldots, M$ **do**
  Let $H'_i = H_i(i)$.
**end for**

The proof of Theorem 14 relies on the following sequence of lemmas. Their proofs, consisting of linear algebraic derivations, are given in Appendix C.

**Lemma 15.** *In Algorithm 2, for $2 \leqslant i \leqslant M$, the matrix $G_{i-1}H_i^T$ is a $v_i \times v_i$ full rank matrix; that is, the matrix $B_i$ exists and is also a full rank matrix.*

*Proof:* See Appendix C. ∎

Now, using the outputs $H'_1, H'_2, \ldots, H'_M$ from Algorithm 2, let $H'(\mathcal{C}_i)$, $1 \leqslant i \leqslant M$, have the following form,

$$H'(\mathcal{C}_i) = \begin{bmatrix} H'_1 \\ H'_2 \\ \vdots \\ H'_i \end{bmatrix}. \tag{14}$$

Using Lemma 15, the next lemma confirms that the matrix $H'(\mathcal{C}_i)$ is indeed a parity-check matrix for the code $\mathcal{C}_i$.

**Lemma 16.** *For $1 \leqslant i \leqslant M$, using the matrices $H'_1, H'_2, \ldots, H'_i$ from Algorithm 2, the matrix $H'(\mathcal{C}_i)$ in (14) is a parity-check matrix of $\mathcal{C}_i$.*

*Proof:* See Appendix C. ∎

The next lemma establishes the relationship between information symbols and syndromes.

**Lemma 17.** *For $2 \leqslant i \leqslant M$, we have $\pi_{\mathcal{N}_i}(\boldsymbol{u}) = \boldsymbol{s}_i = \boldsymbol{u}G(\mathcal{C}_1)H_i'^T$.*

*Proof:* See Appendix C. ∎

We now have the ingredients for the proof of Theorem 14.

*Proof of Theorem 14:* Theorem 14 follows immediately from the combination of Lemma 16 and Lemma 17. ∎

## VI. Application and Performance of Rate-Compatible Codes

In this section, we investigate an application of rate-compatible codes to flash memories. Specifically, we first construct two-level rate-compatible codes based on BCH and LDPC codes, respectively, and then evaluate the performance of these codes on MLC flash memories. We also study the performance of rate-compatible codes with various code rates over binary and $q$-ary symmetric channels.

### A. Flash Memory and Adaptive Error Mitigation

We begin with a brief review of the structure of NAND flash memories [3], [4]. The basic storage unit in a flash memory is a floating-gate transistor referred to as a cell. The voltage levels of a cell can be adjusted by a program operation and are used to represent the stored data. The cells typically have 2, 4, and 8 voltage levels (1, 2, and 3 bits/cell, respectively) and are referred to as single-level cell (SLC), multi-level cell (MLC), and three-level cell (TLC), respectively. Cells are organized into a rectangular array, interconnected by horizontal *wordlines* and vertical *bitlines*, that constitute a *block*. A flash memory chip comprises a collection of such blocks.

During program (i.e., write) operations, the voltage level of a cell cannot be decreased. In order to do so, the entire containing block must be erased and reprogrammed. Repeated program/erase (P/E) operations induce wear on the cells, with a detrimental effect on the lifetime of the flash memory. It has been observed that the raw bit error rate increases as the P/E cycle count grows [5], [36], [39], [40]; as a result, at the stage where the P/E cycle count is higher, the capability to tolerate more errors is needed. It was suggested in [4] to use several distinct ECCs with decreasing code rates throughout the lifetime of the flash memory. Here, we take a different approach and explore the use of syndrome-coupled rate-compatible ECCs.

Although we do not address here system implementation and performance issues arising from the use of rate-compatible codes in solid-state drives, some potential advantages can be seen by contrasting a two-level rate-compatible code with a pair of distinct codes. First, if one uses a pair of separately optimized codes, their encoders (and decoders) usually require distinct implementations. In contrast, in the rate-compatible case, the first code is embedded as a part of the second code, so the encoder and decoder implementations of the second code can incorporate those of the first code, resulting in reduced overall complexity. Second, the use of two distinct codes with different rates and block lengths may require a change in the layout of codewords in a page. With a rate-compatible code, when the second level is invoked, the same number of first-level codewords can be stored in a page, while the second-level parities can be stored in a special reserved area, to be read only when needed. Such an ECC architecture was recently proposed in [34], [35] in the context of 3D NAND flash subject to errors induced by fast-drift.

In our experiments, we focus on MLC flash memories, where a block has 128 wordlines, each of which consists of upper and lower pages. Each page has a size of 17KB. ECCs are employed in the page to protect data. Two typical ECCs that have been widely used in flash memory industry are BCH and LDPC codes. In the following, we consider both of them, and will construct two-level rate-compatible codes based on BCH and LDPC codes, respectively. The two code rates of practical interest lie in the ranges: $0.93-0.94$ and $0.89-0.90$. We will show the lifetime improvement of the MLC flash memory by utilizing the second level redundancy in rate-compatible codes.

### B. Application of Rate-Compatible Codes on MLC Flash Memories

*1) Rate-Compatible Codes Based on BCH Codes:* We first construct two-level rate-compatible codes based on binary BCH codes. We choose two nested binary BCH codes $\mathcal{C}_1^2 = [8191, 7398]_2 \subset \mathcal{C}_1^1 = [8191, 7697]_2$ as our component codes; the codes $\mathcal{C}_1^1$ and $\mathcal{C}_1^2$ can correct 38 and 61 errors, respectively. We also choose an auxiliary shortened BCH code $\mathcal{A}_2^2 = [398, 299]_2$, which can correct 11 errors. Then, from Construction 1, we obtain two-level rate-compatible codes $\mathcal{C}_1 = [8191, 7697]_2 \prec \mathcal{C}_2 = [8589, 7697]_2$, whose code rates are 0.9397 and 0.8961, respectively. We apply $\mathcal{C}_1$ and $\mathcal{C}_2$ to an MLC flash memory and evaluate their performance. In addition, we evaluate a shortened BCH code $\mathcal{C}_2' = [8593, 7697]_2$ with rate 0.8957, whose code length and rate are similar to those of the code $\mathcal{C}_2$. The code $\mathcal{C}_2'$ can correct 64 errors.

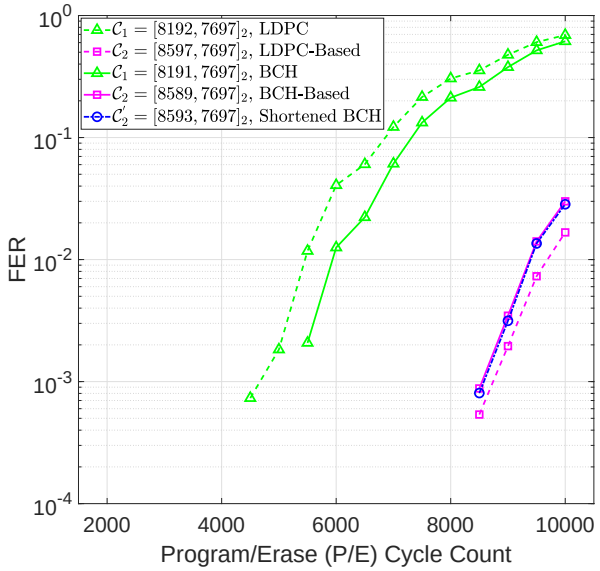
For the performance evaluation of BCH-based two-level rate-compatible codes for an MLC flash memory, we assume that the all-zero codeword is stored and that the memory introduces errors in the locations indicated by our empirical measurements. For the BCH decoder, we assume that if the BCH code (or its coset) could correct $t$ errors, then it would correct any error vector with at most $t$ errors. If the number of errors exceeds $t$, we assume that the BCH decoder would fail. The constructed rate-compatible codes $\mathcal{C}_1 \prec \mathcal{C}_2$ are evaluated over a total of 20 blocks, i.e., 40960 codewords.

The frame error rate (FER) performance of the constructed codes $\mathcal{C}_1$ and $\mathcal{C}_2$ for the lower page and upper page of an MLC flash memory is shown in Fig. 2(a) and Fig. 2(b), respectively. Compared to $\mathcal{C}_1$, the code $\mathcal{C}_2$ extends the lifetime around 3500 program/erase (P/E) cycles for the lower page and around 2000 P/E cycles for the upper page.

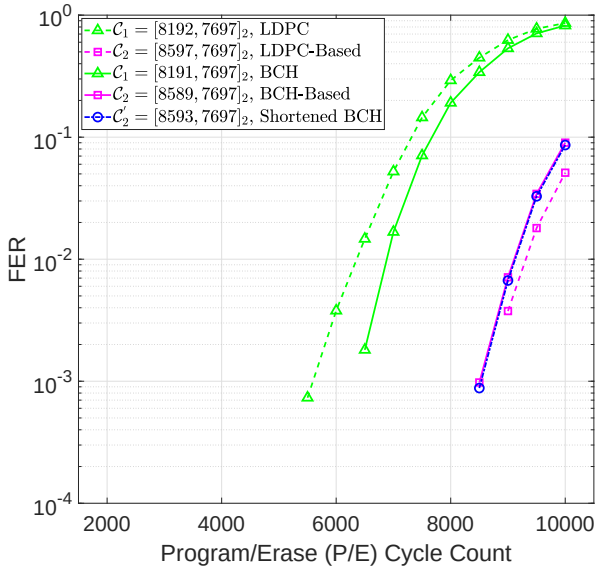
In addition, we evaluate the shortened BCH code $\mathcal{C}_2'$. The FER performance results for the lower page and upper page are shown in Fig. 2(a) and Fig. 2(b), respectively. It can be seen that the FER of $\mathcal{C}_2$ is comparable to that of $\mathcal{C}_2'$, which indicates the effectiveness of our construction.

*2) Rate-Compatible Codes Based on LDPC Codes:* We next construct two-level rate-compatible codes based on binary LDPC codes. We use a Reed-Solomon (RS) codes based construction for regular LDPC codes, since this construction provides a nested and 4-cycle free structure [30]. We can obtain two nested binary LDPC codes $\mathcal{C}_1^2 \subset \mathcal{C}_1^1$, where $\mathcal{C}_1^1$ is a $(4, 64)$-regular $[8192, 7697]_2$ LDPC code with rate 0.9396 and $\mathcal{C}_1^2$ is a $(7, 64)$-regular $[8192, 7400]_2$ LDPC code with rate 0.9033. We also choose an auxiliary (4,15)-regular $[405, 300]_2$ LDPC code $\mathcal{A}_2^2$. Then, in the second step of Construction 1, we obtain the syndrome $\boldsymbol{s}_2$ of length 297. We add three zeros to the end of $\boldsymbol{s}_2$ to form a new vector which is encoded by $\mathcal{A}_2^2$ to generate the vector $\boldsymbol{a}_2^2$. Thus, from Construction 1, we obtain two-level rate-compatible codes $\mathcal{C}_1 = [8192, 7697]_2 \prec \mathcal{C}_2 = [8597, 7697]_2$, whose code rates are 0.9396 and 0.8953, respectively.

For the performance evaluation of LDPC-based two-level rate-compatible codes for an MLC flash memory, we assume that the all-zero codeword is stored and that the memory




(a)




(b)

Fig. 2. FER performance of two-level rate-compatible codes based on BCH and LDPC codes for an MLC flash memory: (a) lower page and (b) upper page.

introduces errors in the locations indicated by our empirical measurements. We treat the channel as a binary symmetric channel with crossover error probability $p$ equal to the average probability of error reflected in the measured error data. The decoder is based upon belief-propagation (BP) decoding, implemented in software as the floating-point sum-product algorithm (SPA). The maximum number of iterations is set to be 100 and early termination is used. The constructed rate-compatible codes $\mathcal{C}_1 \prec \mathcal{C}_2$ are evaluated over a total of 20 blocks, i.e., 40960 codewords.

The FER performance of the constructed codes $\mathcal{C}_1$ and $\mathcal{C}_2$ for the lower page and upper page of an MLC flash memory is shown in Fig. 2(a) and Fig. 2(b), respectively. Compared to $\mathcal{C}_1$, the code $\mathcal{C}_2$ extends the lifetime around 4000 P/E cycles for the lower page and around 3000 P/E cycles for the upper page.

From Fig. 2(a) and Fig. 2(b), it is shown that at a higher rate, i.e., around 0.94, the $[8191, 7697]_2$ BCH code outperforms the $[8192, 7697]_2$ LDPC code. However, at a lower rate of around 0.90, the $[8597, 7697]_2$ LDPC-based code is better than the $[8589, 7697]_2$ BCH-based code.

### *C. Performance of Rate-Compatible Codes over Binary and Q-ary Symmetric Channels*

In this subsection, we provide several additional examples of rate-compatible codes obtained from Construction 1. In particular, we consider binary two-level rate-compatible codes based on BCH and LDPC codes with rates in the range $0.6 - 0.8$ and in the range $0.3 - 0.55$. We present results of computer simulations that demonstrate their FER performance on the BSC. We also describe a binary BCH-based three-level rate-compatible code and show its FER performance on the BSC. Finally, we consider a non-binary two-level rate-compatible code based on Reed-Solomon codes over $\mathbb{F}_{64}$ and examine its FER performance on the 64-ary symmetric channel.

The rate-compatible codes constructed for MLC flash memories in the previous subsection have relatively high rates, i.e., in the range $0.85 - 0.95$. With the similar steps, in the next two examples, we construct two-level rate-compatible binary codes with lower code rates and also present their performances over the BSC by simulations.

**Example 6.** In this example, we construct two-level rate-compatible binary codes with code rates in the range $0.6 - 0.8$.

We first construct two-level rate-compatible codes based on binary BCH codes. Consider two nested binary BCH codes $\mathcal{C}_1^2 = [255, 171]_2 \subset \mathcal{C}_1^1 = [255, 199]_2$ that can correct 11 and 7 errors, respectively. Choose an auxiliary shortened BCH code $\mathcal{A}_2^2 = [61, 28]_2$ which corrects 6 errors. From Construction 1, we obtain two-level rate-compatible codes $\mathcal{C}_1 = [255, 199]_2 \prec \mathcal{C}_2 = [316, 199]_2$ whose code rates are 0.7804 and 0.6298, respectively. In addition, we evaluate a shortened BCH code $\mathcal{C}_2' = [316, 199]_2$ which can correct 13 errors. The code lengths and dimensions of $\mathcal{C}_2$ and $\mathcal{C}_2'$ are identical.

We next construct two-level rate-compatible codes based on binary LDPC codes. Using a Reed-Solomon codes based construction for regular LDPC codes [30, Ch. 11.9], we obtain two nested binary LDPC codes $\mathcal{C}_1^2 \subset \mathcal{C}_1^1$, where $\mathcal{C}_1^1$ is a $(4, 16)$-regular $[256, 203]_2$ LDPC code and $\mathcal{C}_1^2$ is a $(16, 16)$-regular $[256, 175]_2$ LDPC code. We also choose an auxiliary shortened BCH code $\mathcal{A}_2^2 = [61, 28]_2$ which corrects 6 errors. From Construction 1, we obtain two-level rate-compatible codes $\mathcal{C}_1 = [256, 203]_2 \prec \mathcal{C}_2 = [317, 203]_2$, whose code rates are 0.793 and 0.6404, respectively.

The FER performances of these codes over the BSC are shown in Fig. 3. The $[255, 199]_2$ BCH code outperforms the

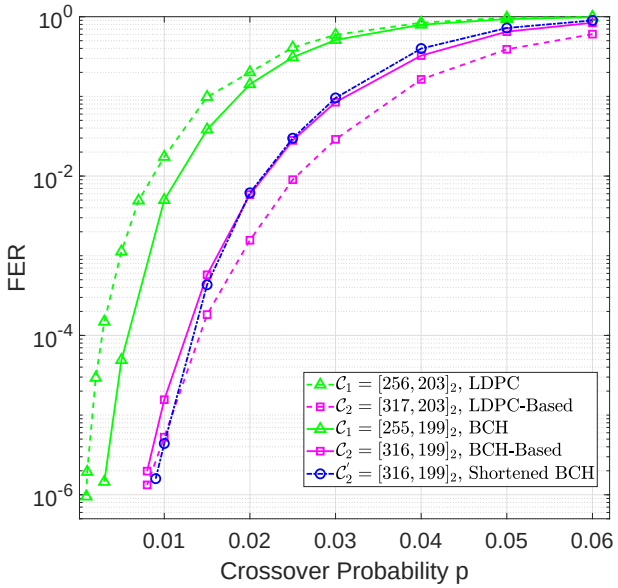



Fig. 3. FER performance of two-level rate-compatible codes with code rates in the range $0.6 - 0.8$ over the BSC.

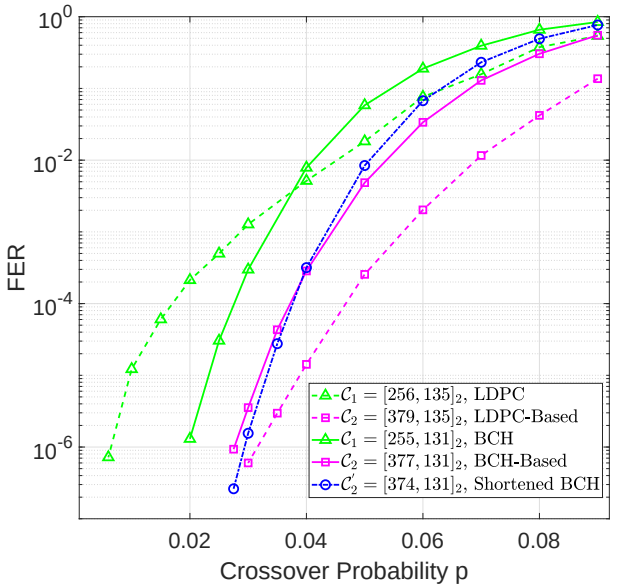



Fig. 4. FER performance of two-level rate-compatible codes with code rates in the range $0.3 - 0.55$ over the BSC.

$[256, 203]_2$ LDPC code. In contrast, for the second level, the $[317, 203]_2$ LDPC-based code is better than the $[316, 199]_2$ BCH-based code which performs very close to the $[316, 199]_2$ shortened BCH code. For the crossover probability $p$ of the BSC around 0.01 (i.e., low noise region), the LDPC-based code $\mathcal{C}_2$ has a similar performance to $\mathcal{C}_2'$. At FER around $10^{-6}$, no error floor is found for these constructed rate-compatible codes. □

**Example 7.** In this example, we present two-level rate-compatible binary codes with code rates in the range $0.3 - 0.55$.

We again begin by constructing two-level rate-compatible codes based on binary BCH codes. Consider two nested binary BCH codes $\mathcal{C}_1^2 = [255, 107]_2 \subset \mathcal{C}_1^1 = [255, 131]_2$ that can correct 22 and 18 errors, respectively. Choose an auxiliary shortened BCH code $\mathcal{A}_2^2 = [122, 24]_2$ which corrects 21 errors. From Construction 1, we obtain two-level rate-compatible codes $\mathcal{C}_1 = [255, 131]_2 \prec \mathcal{C}_2 = [377, 131]_2$ whose code rates are 0.5137 and 0.3475, respectively. Moreover, we evaluate a shortened BCH code $\mathcal{C}_2' = [374, 131]_2$ which can correct 29 errors. The codes $\mathcal{C}_2$ and $\mathcal{C}_2'$ have the similar code length and the same dimension.

We also construct two-level rate-compatible codes based on binary LDPC codes. Using a Reed-Solomon codes based construction for regular LDPC codes [30, Ch. 11.9], we obtain two nested binary LDPC codes $\mathcal{C}_1^2 \subset \mathcal{C}_1^1$, where $\mathcal{C}_1^1$ is a $(5, 8)$-regular $[256, 135]_2$ LDPC code and $\mathcal{C}_1^2$ is an $(8, 8)$-regular $[256, 110]_2$ LDPC code. We also choose an auxiliary shortened BCH code $\mathcal{A}_2^2 = [123, 25]_2$ which corrects 21 errors. From Construction 1, we obtain two-level rate-compatible codes $\mathcal{C}_1 = [256, 135]_2 \prec \mathcal{C}_2 = [379, 135]_2$, whose code rates are 0.5273 and 0.3562, respectively.

The FER performances of these codes over the BSC are plotted in Fig. 4. For the crossover probability $p$ of the BSC below 0.04, the $[255, 131]_2$ BCH code surpasses the $[256, 135]_2$ LDPC code, while for the crossover probability $p$ above 0.04, the opposite result holds. For the second level, in general, the $[379, 135]_2$ LDPC-based code performs much better than the $[377, 131]_2$ BCH-based code which is comparable to the $[374, 131]_2$ shortened BCH code. For the crossover probability $p$ around 0.03, the LDPC-based code $\mathcal{C}_2$ only slightly outperforms $\mathcal{C}_2'$. At FER around $10^{-6}$, for the constructed rate-compatible codes, no error floor occurs. □

The following example demonstrates three-level rate-compatible binary codes based on BCH codes.

**Example 8.** Consider a set of nested binary BCH codes $\mathcal{C}_1^3 = [511, 241]_2 \subset \mathcal{C}_1^2 = [511, 304]_2 \subset \mathcal{C}_1^1 = [511, 376]_2$ that can correct 36, 25, and 15 errors, respectively. Choose two auxiliary shortened BCH codes $\mathcal{A}_2^3 = [172, 48]_2 \subset \mathcal{A}_2^2 = [172, 72]_2$ that can correct 18 and 13 errors, respectively, and also choose $\mathcal{A}_3^3 = [227, 87]_2$ which corrects 21 errors. Then, from Construction 1, we obtain three-level rate-compatible codes $\mathcal{C}_1 = [511, 376]_2 \prec \mathcal{C}_2 = [683, 376]_2 \prec \mathcal{C}_3 = [910, 376]_2$. Their simulation results over the BSC are shown in Fig. 5.

Moreover, for comparison, we evaluate two shortened BCH codes $\mathcal{C}_2' = [681, 376]_2$ and $\mathcal{C}_3' = [906, 376]_2$ that can correct 31 and 59 errors, respectively. Fig. 5 illustrates that for the second level, the code $\mathcal{C}_2$ performs very close to $\mathcal{C}_2'$. For the third level, the gap between $\mathcal{C}_3$ and $\mathcal{C}_3'$ becomes more evident. At a high FER around $10^{-1}$, $\mathcal{C}_3$ performs slightly better than $\mathcal{C}_3'$, while at a low FER around $10^{-5}$, $\mathcal{C}_3$ is worse than $\mathcal{C}_3'$. Note that at FER around $10^{-5}$, the gap between $\mathcal{C}_3$ and $\mathcal{C}_3'$ is larger

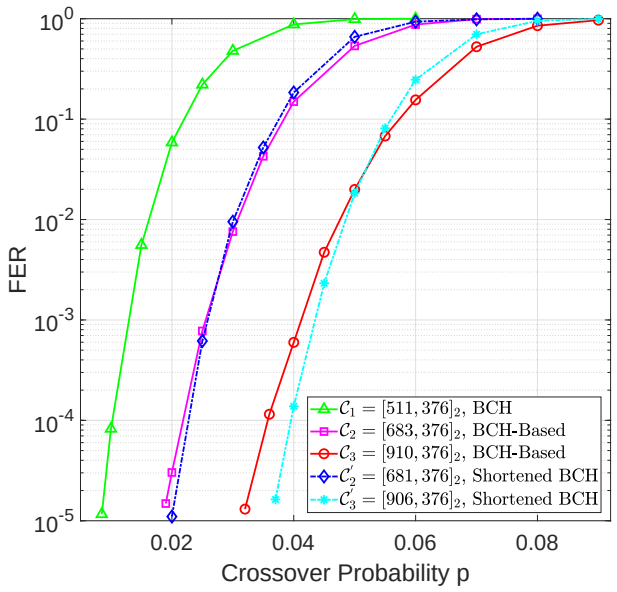



Fig. 5. FER performance of three-level rate-compatible codes based on BCH codes over the BSC.

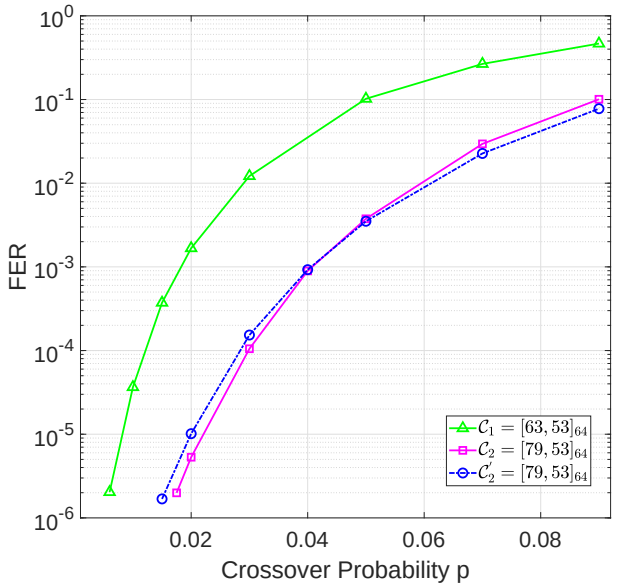



Fig. 6. FER performance of two-level rate-compatible non-binary codes based on Reed-Solomon codes over the 64-ary symmetric channel.

than the one between $\mathcal{C}_2$ and $\mathcal{C}_2'$, which implies that at finite lengths the performance degrades with the growing number of levels. □

The examples above are all binary codes. Here we provide an example of rate-compatible non-binary codes based on Reed-Solomon codes.

**Example 9.** We construct two-level rate-compatible non-binary codes over $\mathbb{F}_{64}$. Consider two nested Reed-Solomon codes $\mathcal{C}_1^2 = [63, 47]_{64} \subset \mathcal{C}_1^1 = [63, 53]_{64}$. Choose a shortened Reed-Solomon code as the auxiliary code $\mathcal{A}_2^2 = [16, 6]_{64}$. From Construction 1, we obtain two-level rate-compatible non-binary codes $\mathcal{C}_1 = [63, 53]_{64} \prec \mathcal{C}_2 = [79, 53]_{64}$. Since there does not exist a Reed-Solomon code (or a shortened one) of length 79 and dimension 53 over $\mathbb{F}_{64}$, for simplicity, we use a code $\mathcal{C}_2' = [79, 53]_{64}$ formed by concatenating two shortened Reed-Solomon codes $[41, 27]_{64}$ and $[38, 26]_{64}$ for comparison. The simulation results of these codes over the 64-ary symmetric channel are presented in Fig. 6, which shows that $\mathcal{C}_2$ is comparable to $\mathcal{C}_2'$. At FER around $2 \times 10^{-6}$, $\mathcal{C}_2$ is slightly better than $\mathcal{C}_2'$, and no error floor is observed. □

We conclude this section by commenting on some of the performance and implementation considerations associated with designing rate-compatible codes, including the proposed syndrome-coupled codes. As mentioned in Section I, there is a substantial literature in which the rate-compatible codes are obtained by puncturing a low-rate base code, as well as a large body of work in which the code is obtained by extending a high-rate base code, as in our syndrome-coupled construction. In fact, a number of papers, e.g., [10], [22], [41], describe constructions that make use of both puncturing and extending of LDPC codes in order to exploit the benefits and avoid the drawbacks inherent in the two approaches.

Puncturing is attractive in its use of a single underlying code structure, with higher rate codes obtained by not using all parity bits. However, it may require generation of all of the parity bits corresponding to the low-rate base code, whether or not they will ultimately be used. Moreover, the decoder effectively operates on the full length of the base code, with untransmitted parities treated as erasures. In the case of LDPC codes, this means that the decoding algorithm will operate on a Tanner graph whose size is dictated by the lowest-rate (base) code [10], [22], [37]. Moreover, it has been found that puncturing a low-rate code to a very high rate leads to performance degradation relative to an optimized code.

Extension, on the other hand, can begin with an optimized high-rate code. However, the extending to produce lower rate codes tends to introduce complexity in the code design in one form or another to generate the additional parities. The potential advantage is that additional parities are generated only when needed. On the other hand, the decoder must be able to decode each of the lower-rate codes produced by extending. For example, for $M$-level syndrome-coupled rate-compatible codes, the decoding for the lowest-rate code (i.e., $M$-level code) requires $M$ decoders. However, each of these additional decoders is only used to decode an auxiliary code which will often be of smaller length and hence have relatively lower complexity. Moreover, if channel conditions are such that the decoder only invokes a small number of code levels to succeed, the actual decoding complexity is proportionately reduced by requiring only a corresponding small number of decoders.

There are two additional features of the proposed syndrome-coupled rate-compatible coding scheme that may offer further practical benefits. First, it is a universal design method, in

the sense that it can be easily applied to a wide range of codes, including BCH, RS, LDPC and polar codes. Thus, it is compatible with systems in which there may be a preference for one of these classes of codes. Second, the construction provides flexibility in choosing the class of each individual component code. For example, in constructing two-level codes, one can use LDPC component codes for both levels, or an LDPC code for the first level and a BCH code for the second level. Under certain channel conditions and constraints on code lengths and rates, the latter choice of code classes may be preferable. Of course, there are many engineering considerations that enter into the design of a practical and effective rate-compatible coding scheme, and our aim here has been simply to highlight some of them.

## VII. Variations of Construction 1

In this section, we briefly discuss two possible variations of our main construction, i.e., Construction 1. The following two constructions are similar to Construction 1, so some proofs are omitted. In contrast to Construction 1, one distinct feature of these two constructions is that they do not require the auxiliary codes have the nested structure.

### A. Construction 2

In this construction for $M$-level rate-compatible codes, we use a set of component codes described below.

1) Choose a set of nested codes $\mathcal{C}_1^M \subset \mathcal{C}_1^{M-1} \subset \cdots \subset \mathcal{C}_1^1 = \mathcal{C}_1 = [n_1, k, d_1]_q$, where $\mathcal{C}_1^i = [n_1, n_1 - \sum_{m=1}^{i} v_m, d_i]_q$ for $1 \leqslant i \leqslant M$. The encoder of $\mathcal{C}_1$ is denoted by $\mathcal{E}_{\mathcal{C}_1} : \mathbb{F}_q^k \to \mathcal{C}_1$. These nested codes are the same as those used in Construction 1.

2) Choose a set of $[n_i, \sum_{j=2}^{i} v_j, \delta_i]_q$ auxiliary codes $\mathcal{A}_i$, $2 \leqslant i \leqslant M$. The corresponding encoder is denoted by $\mathcal{E}_{\mathcal{A}_i} : \mathbb{F}_q^{\sum_{j=2}^{i} v_j} \to \mathcal{A}_i$.

**Construction 2: Encoding Procedure**

**Input:** A length-$k$ vector $\boldsymbol{u}$ of information symbols over $\mathbb{F}_q$.
**Output:** A codeword $\boldsymbol{c}_i \in \mathcal{C}_i$ over $\mathbb{F}_q$, for $i = 1, \ldots, M$.

1: $\boldsymbol{c}_1 = \mathcal{E}_{\mathcal{C}_1}(\boldsymbol{u})$.
2: $\boldsymbol{s}_i = \boldsymbol{c}_1 H^T_{\mathcal{C}_1^i | \mathcal{C}_1^{i-1}}$ for $i = 2, 3, \ldots, M$.
3: **for** $i = 2, \ldots, M$ **do**
4: $\quad \boldsymbol{a}_i = \mathcal{E}_{\mathcal{A}_i}\Big((\boldsymbol{s}_2, \boldsymbol{s}_3, \ldots, \boldsymbol{s}_i)\Big)$.
5: $\quad \boldsymbol{c}_i = (\boldsymbol{c}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_i)$.
6: **end for**

The following theorem describes the code parameters of $\mathcal{C}_i$ obtained from Construction 2.

**Theorem 18.** *From Construction 2, the code $\mathcal{C}_i$, $1 \leqslant i \leqslant M$, has length $N_i = \sum_{j=1}^{i} n_j$, dimension $K_i = k$, and minimum distance $D_i \geqslant \min\{d_1 + \sum_{j=2}^{i} \delta_j, d_2 + \sum_{j=3}^{i} \delta_j, \ldots, d_{i-1} + \delta_i, d_i\}$.*

*Proof:* The code length and dimension are obvious. In the following, we prove the minimum distance $D_i$.

From the construction, any nonzero codeword $\boldsymbol{c}_i \in \mathcal{C}_i$ is associated with a certain nonzero codeword $\boldsymbol{c}_1$.

If $\boldsymbol{c}_1 \in \mathcal{C}_1^1$ and $\boldsymbol{c}_1 \notin \mathcal{C}_1^2$, we have $w_q(\boldsymbol{c}_1) \geqslant d_1$ and $w_q(\boldsymbol{a}_j) \geqslant \delta_j$ for $2 \leqslant j \leqslant i$. Thus, $w_q(\boldsymbol{c}_i) \geqslant d_1 + \sum_{j=2}^{i} \delta_j$.

Similarly, if $\boldsymbol{c}_1 \in \mathcal{C}_1^\ell$ and $\boldsymbol{c}_1 \notin \mathcal{C}_1^{\ell+1}$ for some $\ell$, $2 \leqslant \ell \leqslant i-1$, we have $w_q(\boldsymbol{c}_1) \geqslant d_\ell$ and $w_q(\boldsymbol{a}_j) \geqslant \delta_j$ for $\ell + 1 \leqslant j \leqslant i$. Thus, $w_q(\boldsymbol{c}_i) \geqslant d_\ell + \sum_{j=\ell+1}^{i} \delta_j$.

If $\boldsymbol{c}_1 \in \mathcal{C}_1^i$, we have $w_q(\boldsymbol{c}_1) \geqslant d_i$, so $w_q(\boldsymbol{c}_i) \geqslant d_i$. ∎

From Theorem 18, we have the following corollary on the minimum distance.

**Corollary 19.** *Let $\mathcal{A}_i$, $2 \leqslant i \leqslant M$, have minimum distance $\delta_i \geqslant d_i - d_{i-1}$. From Construction 2, the code $\mathcal{C}_i$, $1 \leqslant i \leqslant M$, has minimum distance $D_i = d_i$.*

*Proof:* First, from Theorem 18, we have $D_i \geqslant \min\{d_1 + \sum_{j=2}^{i} \delta_j, d_2 + \sum_{j=3}^{i} \delta_j, \ldots, d_{i-1} + \delta_i, d_i\} = d_i$. On the other hand, since there exists a codeword $\boldsymbol{c}_1 \in \mathcal{C}_1^i$ such that $w_q(\boldsymbol{c}_1) = d_i$ and $w_q(\boldsymbol{c}_i) = d_i$, we have $D_i \leqslant d_i$. ∎

The following example gives three-level rate-compatible codes from Construction 2.

**Example 10.** Consider a set of nested binary BCH codes $\mathcal{C}_1^3 = [15, 5, 7]_2 \subset \mathcal{C}_1^2 = [15, 7, 5]_2 \subset \mathcal{C}_1^1 = [15, 11, 3]_2$. Choose a set of auxiliary codes $\mathcal{A}_2 = [5, 4, 2]_2$ and $\mathcal{A}_3 = [7, 6, 2]_2$.

Then, from Construction 2 and Corollary 19, we obtain three-level rate-compatible codes $\mathcal{C}_1 = [15, 11, 3]_2 \prec \mathcal{C}_2 = [20, 11, 5]_2 \prec \mathcal{C}_3 = [27, 11, 7]_2$. □

We present correctable error-erasure patterns for $\mathcal{C}_i$, $1 \leqslant i \leqslant M$. For $1 \leqslant j \leqslant i \leqslant M$, let $t_j$ and $\tau_j$ denote the number of errors and erasures in the sub-block $\boldsymbol{y}_j$ of the received word $\boldsymbol{y} \in (\mathbb{F}_q \cup \{?\})^{N_i}$, which corresponds to a transmitted codeword $\boldsymbol{c}_i \in \mathcal{C}_i$.

**Lemma 20.** *The code $\mathcal{C}_i$ obtained from Construction 2 can correct any combined error and erasure pattern that satisfies the following condition:*

$$2t_1 + \tau_1 \leqslant d_i - 1,$$
$$2t_i + \tau_i \leqslant \delta_i - 1.$$

*Proof:* The proof is similar to that of Theorem 6, so it is omitted. ∎

### B. Construction 3

In this construction for $M$-level rate-compatible codes, we use a set of the following component codes.

1) Choose a set of nested codes $\mathcal{C}_1^M \subset \mathcal{C}_1^{M-1} \subset \cdots \subset \mathcal{C}_1^1 = \mathcal{C}_1 = [n_1, k, d_1]_q$, where $\mathcal{C}_1^i = [n_1, n_1 - \sum_{m=1}^{i} v_m, d_i]_q$ for $1 \leqslant i \leqslant M$. The encoder of $\mathcal{C}_1$ is denoted by $\mathcal{E}_{\mathcal{C}_1} : \mathbb{F}_q^k \to \mathcal{C}_1$. These nested codes are the same as those used in Construction 1 and Construction 2.

2) Choose a set of $[n_i, v_i, \delta_i]_q$ auxiliary codes $\mathcal{A}_i$, $2 \leqslant i \leqslant M$. The corresponding encoder is denoted by $\mathcal{E}_{\mathcal{A}_i} : \mathbb{F}_q^{v_i} \to \mathcal{A}_i$.

**Construction 3: Encoding Procedure**

**Input:** A length-$k$ vector $\boldsymbol{u}$ of information symbols over $\mathbb{F}_q$.
**Output:** A codeword $\boldsymbol{c}_i \in \mathcal{C}_i$ over $\mathbb{F}_q$, for $i = 1, \ldots, M$.

1: $\boldsymbol{c}_1 = \mathcal{E}_{\mathcal{C}_1}(\boldsymbol{u})$.
2: $\boldsymbol{s}_i = \boldsymbol{c}_1 H^T_{\mathcal{C}_1^i|\mathcal{C}_1^{i-1}}$ for $i = 2, 3, \ldots, M$.
3: **for** $i = 2, \ldots, M$ **do**
4: $\quad \boldsymbol{a}_i = \mathcal{E}_{\mathcal{A}_i}(\boldsymbol{s}_i)$.
5: $\quad \boldsymbol{c}_i = (\boldsymbol{c}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_i)$.
6: **end for**

The code parameters of $\mathcal{C}_i$ from Construction 3 are given by the following Theorem.

**Theorem 21.** *From Construction 3, the code $\mathcal{C}_i$, $1 \leqslant i \leqslant M$, has code length $N_i = \sum_{j=1}^{i} n_j$, dimension $K_i = k$, and minimum distance $D_i \geqslant \min\{d_1 + \delta_2, d_2 + \delta_3, \ldots, d_{i-1} + \delta_i, d_i\}$.*

*Proof:* The code length and dimension are clear. In the following, we prove the minimum distance $D_i$.

From the construction, any nonzero codeword $\boldsymbol{c}_i \in \mathcal{C}_i$ is associated with a certain nonzero codeword $\boldsymbol{c}_1$.

If $\boldsymbol{c}_1 \in \mathcal{C}_1^1$ and $\boldsymbol{c}_1 \notin \mathcal{C}_1^2$, we have $w_q(\boldsymbol{c}_1) \geqslant d_1$ and $w_q(\boldsymbol{a}_2) \geqslant \delta_2$. Thus, $w_q(\boldsymbol{c}_i) \geqslant d_1 + \delta_2$.

Similarly, if $\boldsymbol{c}_1 \in \mathcal{C}_1^\ell$ and $\boldsymbol{c}_1 \notin \mathcal{C}_1^{\ell+1}$ for some $\ell$, $2 \leqslant \ell \leqslant i-1$, we have $w_q(\boldsymbol{c}_1) \geqslant d_\ell$ and $w_q(\boldsymbol{a}_{\ell+1}) \geqslant \delta_{\ell+1}$. Thus, $w_q(\boldsymbol{c}_i) \geqslant d_\ell + \delta_{\ell+1}$.

If $\boldsymbol{c}_1 \in \mathcal{C}_1^i$, we have $w_q(\boldsymbol{c}_1) \geqslant d_i$, so $w_q(\boldsymbol{c}_i) \geqslant d_i$. ∎

The following corollary from Theorem 21 shows that the minimum distances of the constructed rate-compatible codes can be determined exactly under some condition.

**Corollary 22.** *Let $\mathcal{A}_i$, $2 \leqslant i \leqslant M$, have minimum distance $\delta_i \geqslant d_M - d_{i-1}$. From Construction 3, the code $\mathcal{C}_i$, $1 \leqslant i \leqslant M$, has minimum distance $D_i = d_i$.*

*Proof:* First, from Theorem 21, we have $D_i \geqslant \min\{d_1 + \delta_2, d_2 + \delta_3, \ldots, d_{i-1} + \delta_i, d_i\} = \min\{d_M, d_i\} = d_i$. On the other hand, since there exists a codeword $\boldsymbol{c}_1 \in \mathcal{C}_1^i$ such that $w_q(\boldsymbol{c}_1) = d_i$ and $w_q(\boldsymbol{c}_i) = d_i$, we have $D_i \leqslant d_i$. ∎

We provide an example of three-level rate-compatible codes from Construction 3.

**Example 11.** Consider a set of nested binary BCH codes $\mathcal{C}_1^3 = [15, 5, 7]_2 \subset \mathcal{C}_1^2 = [15, 7, 5]_2 \subset \mathcal{C}_1^1 = [15, 11, 3]_2$. Choose a set of auxiliary codes $\mathcal{A}_2 = [8, 4, 4]_2$ and $\mathcal{A}_3 = [3, 2, 2]_2$.

Then, from Construction 3 and Corollary 22, we obtain three-level rate-compatible codes $\mathcal{C}_1 = [15, 11, 3]_2 \prec \mathcal{C}_2 = [23, 11, 5]_2 \prec \mathcal{C}_3 = [26, 11, 7]_2$. □

Finally, we give correctable error-erasure patterns for $\mathcal{C}_i$, $1 \leqslant i \leqslant M$. For $1 \leqslant j \leqslant i \leqslant M$, let $t_j$ and $\tau_j$ denote the number of errors and erasures in the sub-block $\boldsymbol{y}_j$ of the received word $\boldsymbol{y} \in (\mathbb{F}_q \cup \{?\})^{N_i}$, corresponding to a transmitted codeword $\boldsymbol{c}_i \in \mathcal{C}_i$.

**Lemma 23.** *The code $\mathcal{C}_i$ obtained from Construction 3 can correct any combined error and erasure pattern that satisfies the following condition:*

$$2t_1 + \tau_1 \leqslant d_i - 1,$$
$$2t_j + \tau_j \leqslant \delta_j - 1, \ \forall\, 2 \leqslant j \leqslant i.$$

*Proof:* The proof is similar to that of Theorem 6, so it is omitted. ∎

**Remark 7.** Compared to Construction 1, Construction 2 and Construction 3 do not use nested auxiliary codes. Thus, they possess relatively lower encoding and decoding complexities, which is an attractive feature for practical applications.

Using the same nested binary BCH codes $\mathcal{C}_1^3 = [15, 5, 7]_2 \subset \mathcal{C}_1^2 = [15, 7, 5]_2 \subset \mathcal{C}_1^1 = [15, 11, 3]_2$ as component codes, in Example 1, Example 10, and Example 11, we construct three-level rate-compatible codes $\mathcal{C}_1 \prec \mathcal{C}_2 \prec \mathcal{C}_3$ from Construction 1, Construction 2, and Construction 3, respectively. It is shown that the code $\mathcal{C}_3 = [26, 11, 7]_2$ from Example 1 has a higher rate than that of the code $\mathcal{C}_3 = [27, 11, 7]_2$ from Example 10, and the code $\mathcal{C}_2 = [20, 11, 5]_2$ from Example 1 has a higher rate than that of the code $\mathcal{C}_2 = [23, 11, 5]_2$ from Example 11. Therefore, for this specific three-level case, merely from the perspective of code rates, Construction 1 is better than Construction 2 and Construction 3. However, these three constructions provide distinct correctable error and erasure patterns.

For the two-level case, Construction 1, Construction 2, and Construction 3 are all the same and in fact can be seen as the linear realization of Construction X [26]. For the case of multiple levels, in contrast to Construction 1, it seems that Construction 2 and Construction 3 cannot be used to construct capacity-achieving rate-compatible codes.

The following example shows three-level rate-compatible codes generated by Construction 1, Construction 2, and Construction 3, respectively. Although they have similar code parameters, in terms of the performance over the BSC, Construction 1 gives the best results.

**Example 12.** *Construction 1*: Consider a set of nested binary BCH codes $\mathcal{C}_1^3 = [511, 193]_2 \subset \mathcal{C}_1^2 = [511, 259]_2 \subset \mathcal{C}_1^1 = [511, 340]_2$ that can correct 43, 30, and 20 errors, respectively. Choose two auxiliary shortened BCH codes $\mathcal{A}_2^3 = [205, 49]_2 \subset \mathcal{A}_2^2 = [205, 81]_2$ that can correct 23 and 18 errors, respectively, and also choose $\mathcal{A}_3^3 = [368, 98]_2$ which corrects 36 errors. Then, from Construction 1, we obtain three-level rate-compatible codes $\mathcal{C}_1 = [511, 340]_2 \prec \mathcal{C}_2 = [716, 340]_2 \prec \mathcal{C}_3 = [1084, 340]_2$.

*Construction 2*: Consider a set of nested binary BCH codes $\mathcal{C}_1^3 = [511, 229]_2 \subset \mathcal{C}_1^2 = [511, 259]_2 \subset \mathcal{C}_1^1 = [511, 340]_2$ that can correct 38, 30, and 20 errors, respectively. Choose an auxiliary code $\mathcal{A}_2 = [205, 81]_2$ which corrects 18 errors, and also choose $\mathcal{A}_3 = [372, 111]_2$ which corrects 31 errors. Then, from Construction 2, we obtain three-level rate-compatible

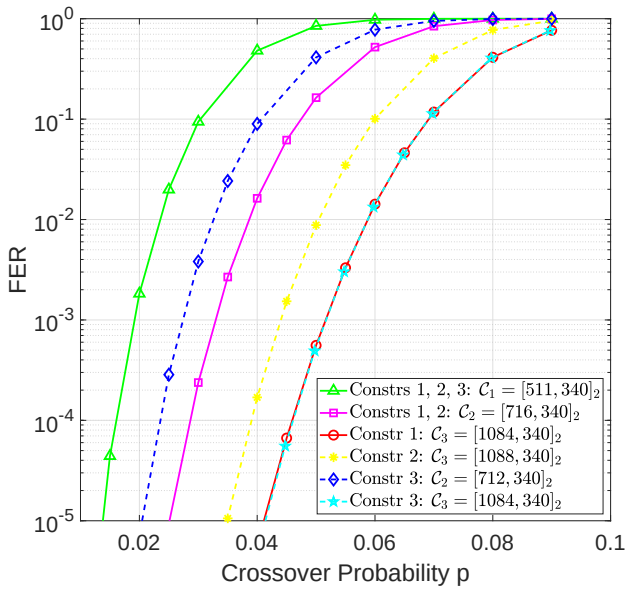



Fig. 7. FER performance of three-level rate-compatible codes obtained from Constructions 1, 2, and 3 over the BSC.

codes $\mathcal{C}_1 = [511, 340]_2 \prec \mathcal{C}_2 = [716, 340]_2 \prec \mathcal{C}_3 = [1088, 340]_2$.

*Construction 3*: Consider a set of nested binary BCH codes $\mathcal{C}_1^3 = [511, 193]_2 \subset \mathcal{C}_1^2 = [511, 295]_2 \subset \mathcal{C}_1^1 = [511, 340]_2$ that can correct 43, 26, and 20 errors, respectively. Choose an auxiliary code $\mathcal{A}_2 = [201, 45]_2$ which corrects 23 errors, and also choose $\mathcal{A}_3 = [372, 102]_2$ which corrects 36 errors. Then, from Construction 3, we obtain three-level rate-compatible codes $\mathcal{C}_1 = [511, 340]_2 \prec \mathcal{C}_2 = [712, 340]_2 \prec \mathcal{C}_3 = [1084, 340]_2$.

The rate-compatible codes $\mathcal{C}_1 \prec \mathcal{C}_2 \prec \mathcal{C}_3$ obtained from Construction 1, Construction 2, and Construction 3 have almost the same code parameters, but their performances over the BSC are different and are plotted in Fig. 7. It is obvious that Construction 1, Construction 2, and Construction 3 all start with the same code $\mathcal{C}_1$. For $\mathcal{C}_2$, Construction 1 and Construction 2 give the same performance, and both of them are better than Construction 3; for $\mathcal{C}_3$, Construction 1 is comparable to Construction 3, but outperforms Construction 2. □

## VIII. Conclusion

We presented a new algebraic construction for rate-compatible codes with increasing minimum distances. We also proved that our construction can generate capacity-achieving rate-compatible codes by using proper component codes, validating the optimality of the construction. With polar codes as component codes, we established the equivalence between our construction and the construction of capacity-achieving rate-compatible polar codes in [15].

Our parity-check matrix based approach enables us to conveniently obtain the combinatorial properties (e.g., minimum distance) of the constructed rate-compatible codes, as well as the correction capability of algebraic decoders. Our construction is also very general. Many linear codes, e.g., BCH, RS, and LDPC codes, can be used as component codes, and several examples illustrating their use were given.

We considered several applications of the codes. We evaluated the performance of high-rate, two-level rate-compatible codes based on BCH codes and LDPC codes on MLC flash memories. The results suggest that the constructed rate-compatible codes can effectively prolong the lifetime of the memories. We also evaluated the performance of two-level and three-level rate-compatible binary codes on BSCs over a range of code rates, and we gave an example of a two-level rate-compatible code over $\mathbb{F}_{64}$ used on a 64-ary symmetric channel.

Finally, two variations of Construction 1 were proposed and compared to the original construction.

## Appendix A
## Proof of Theorem 2

**Theorem 2.** *There exist $M$-level rate-compatible codes $\mathcal{C}_1 \prec \mathcal{C}_2 \prec \cdots \prec \mathcal{C}_M$, where $\mathcal{C}_i = [N_i = \sum_{j=1}^{i} n_j,\ k,\ \geqslant d_i]_q$ for $1 \leqslant i \leqslant M$, if the following inequalities are satisfied for all $1 \leqslant i \leqslant M$,*

$$d_i = \max\left\{d : \sum_{m=0}^{d-2} \binom{\sum_{j=1}^{i} n_j - 1}{m} (q-1)^m < \frac{q^{\sum_{j=1}^{i} n_j - k}}{M}\right\}. \tag{2}$$

*Proof:* We first define an $(N_M - k) \times N_M$ matrix $\Phi_M$ over $\mathbb{F}_q$ in the following block lower triangular form,

$$\Phi_M = \begin{bmatrix} H_{1,1} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ H_{2,1} & H_{2,2} & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ H_{M-1,1} & H_{M-1,2} & \cdots & H_{M-1,M-1} & \mathbf{0} \\ H_{M,1} & H_{M,2} & \cdots & H_{M,M-1} & H_{M,M} \end{bmatrix},$$

where $H_{1,1}$ is an $(n_1 - k) \times n_1$ matrix. For $2 \leqslant i \leqslant M$, the matrix $H_{i,1}$ has size $n_i \times n_1$. For $2 \leqslant i \leqslant M$ and $2 \leqslant j \leqslant i$, the matrix $H_{i,j}$ has size $n_i \times n_j$.

For $1 \leqslant i \leqslant M$, we assign the upper left $(N_i - k) \times N_i$ submatrix of $\Phi_M$, denoted by $\Phi_i$, to be a parity-check matrix of $\mathcal{C}_i$. For example, the matrices $H_{1,1}$ and $\Phi_M$ are parity-check matrices of $\mathcal{C}_1$ and $\mathcal{C}_M$, respectively.

Now, we show how to construct $H_{i,j}$, $1 \leqslant i \leqslant M$ and $1 \leqslant j \leqslant i$, such that each code $\mathcal{C}_i$ has its desired code parameters.

First, for $2 \leqslant i \leqslant M$, we choose $H_{i,i}$ to be the $n_i \times n_i$ identity matrix. For $3 \leqslant i \leqslant M$ and $2 \leqslant j \leqslant i-1$, we choose $H_{i,j}$ to be an arbitrary matrix in $\mathbb{F}_q^{n_i \times n_j}$. Next, we construct columns of $H_{i,1}$, $1 \leqslant i \leqslant M$, iteratively, as the technique used in the proof of the GV bound. We use $\boldsymbol{h}_\ell(i)$, $1 \leqslant \ell \leqslant n_1$ and $1 \leqslant i \leqslant M$, to denote the $\ell$th column of the matrix $\Phi_i$ which is a parity-check matrix of $\mathcal{C}_i$. Assume that we have already added the leftmost $\ell - 1$ columns of the matrix $\Phi_M$. In order to show that in $\mathbb{F}_q^{N_M - k}$ there is a vector that can be used as the $\ell$th column $\boldsymbol{h}_\ell(M)$ of the matrix $\Phi_M$, we only need to show

that the total number of bad vectors is less than $q^{N_M-k}$. We count the number of bad vectors as follows.

For the code $\mathcal{C}_1$, it requires that every $d_1-1$ columns in $\Phi_1$ are linearly independent. A bad vector for the $\ell$th column $\boldsymbol{h}_\ell(1)$ in $\Phi_1$ is a vector that can be expressed as a linear combination of $d_1-2$ columns in the preceding $\ell-1$ columns. There are at most a total of $\sum_{m=0}^{d_1-2}\binom{\ell-1}{m}(q-1)^m$ such bad vectors, so we exclude at most $T_1(\ell)=\sum_{m=0}^{d_1-2}\binom{\ell-1}{m}(q-1)^m\times q^{\sum_{j=2}^{M}n_j}$ bad vectors for the column $\boldsymbol{h}_\ell(M)$.

Similarly, for the code $\mathcal{C}_i$, $2\leqslant i\leqslant M$, it requires that every $d_i-1$ columns in $\Phi_i$ are linearly independent. A bad vector for the $\ell$th column $\boldsymbol{h}_\ell(i)$ in $\Phi_i$ is a vector that can be expressed as a linear combination of $d_i-2$ columns in the preceding $\ell-1+\sum_{j=2}^{i}n_j$ selected columns, so we have at most a total of $\sum_{m=0}^{d_i-2}\binom{\ell-1+\sum_{j=2}^{i}n_j}{m}(q-1)^m$ such bad vectors. Then, we exclude at most $T_i(\ell)=\sum_{m=0}^{d_i-2}\binom{\ell-1+\sum_{j=2}^{i}n_j}{m}(q-1)^m\times q^{\sum_{x=i+1}^{M}n_x}$ bad vectors for the column $\boldsymbol{h}_\ell(M)$.

Since we assume that the inequalities (2) are satisfied, we have $T_i(\ell)<\frac{q^{N_M-k}}{M}$ for $1\leqslant i\leqslant M$ and $1\leqslant\ell\leqslant n_1$. Thus, we have $\sum_{i=1}^{M}T_i(\ell)<q^{N_M-k}$, which indicates that a good column $\boldsymbol{h}_\ell(M)$ can be found. ∎

## Appendix B
## Proof of Theorem 4

**Theorem 4.** *Let $\mathcal{C}_1$ be an $[n_1,k,d_1]_q$ code with weight enumerator $A(s)=\sum_{w=0}^{n_1}A_w s^w$, where $A_w$ is the number of codewords of Hamming weight $w$. There exist two-level rate-compatible codes $\mathcal{C}_1\prec\mathcal{C}_2=[N_2=n_1+n_2,\ k,\ \geqslant d_2]_q$, if*

$$\sum_{w=1}^{d_2-1}B_w<q^{n_2},$$

*where $B_w=\frac{1}{q-1}\sum_{m=1}^{w}A_m\binom{n_2}{w-m}(q-1)^{w-m}$, for $1\leqslant w\leqslant N_2$.*

*Proof:* Let an $(n_1-k)\times n_1$ matrix $H_1$ represent a parity-check matrix of $\mathcal{C}_1$. Assume that $\mathcal{C}_2$ has a parity-check matrix $H_2$ in the form

$$H_2=\begin{bmatrix}H_1 & \mathbf{0}\\ H & I\end{bmatrix},$$

where $H$ is an $n_2\times n_1$ matrix and the matrix $I$ represents the $n_2\times n_2$ identity matrix. Construct an ensemble of $(N_2-k)\times N_2$ matrices $\{H_2\}$ by using all $n_2\times n_1$ matrices $H$ over $\mathbb{F}_q$. We then assume a uniform distribution over the ensemble $\{H_2\}$.

We say a matrix $H_2$ is bad, if there exists a vector $\boldsymbol{x}\in\mathbb{F}_q^{N_2}$ such that $\boldsymbol{x}H_2^T=\mathbf{0}$ and $0<w_q(\boldsymbol{x})<d_2$. Thus, we only need to prove the probability $\Pr\{H_2\text{ is bad}\}<1$, i.e., not all $H_2$ are bad.

Define sets $\mathcal{B}'=\{\boldsymbol{x}\in\mathbb{F}_q^{N_2}:\boldsymbol{x}[H_1,\mathbf{0}]^T=\mathbf{0}\}$, $\mathcal{B}''=\{\boldsymbol{x}\in\mathcal{B}':w_q(\boldsymbol{x})>0,\text{ and the leading nonzero entry of }\boldsymbol{x}\text{ is }1\}$, and $\mathcal{B}=\{\boldsymbol{x}\in\mathcal{B}'':w_q\big(\pi_{[n_1]}(\boldsymbol{x})\big)>0\}$. We also define $B_w=|\{\boldsymbol{x}\in\mathcal{B}:w_q(\boldsymbol{x})=w\}|$. It is clear that $B_w=\frac{1}{q-1}\sum_{m=1}^{w}A_m\binom{n_2}{w-m}(q-1)^{w-m}$, for $1\leqslant w\leqslant N_2$. Now, we have

$$\begin{aligned}
&\Pr\{H_2\text{ is bad}\}\\
=&\Pr\{\text{For some }\boldsymbol{x}\in\mathcal{B}',\ 0<w_q(\boldsymbol{x})<d_2,\ \boldsymbol{x}[H,I]^T=\mathbf{0}\}\\
=&\Pr\{\text{For some }\boldsymbol{x}\in\mathcal{B}'',\ 0<w_q(\boldsymbol{x})<d_2,\ \boldsymbol{x}[H,I]^T=\mathbf{0}\}\\
=&\Pr\{\text{For some }\boldsymbol{x}\in\mathcal{B},\ 0<w_q(\boldsymbol{x})<d_2,\ \boldsymbol{x}[H,I]^T=\mathbf{0}\}\\
\overset{(a)}{\leqslant}&\sum_{\boldsymbol{x}\in\mathcal{B}\text{ and }0<w_q(\boldsymbol{x})<d_2}\Pr\{\boldsymbol{x}[H,I]^T=\mathbf{0}\}\\
=&\frac{\sum_{w=1}^{d_2-1}B_w}{q^{n_2}},
\end{aligned}$$

where step (a) follows from the union bound. ∎

## Appendix C
## Proofs of Lemmas 15, 16, 17

**Lemma 15.** *In Algorithm 2, for $2\leqslant i\leqslant M$, the matrix $G_{i-1}H_i^T$ is a $v_i\times v_i$ full rank matrix; that is, the matrix $B_i$ exists and is also a full rank matrix.*

*Proof:* The proof is based on contradiction.

Assume that $G_{i-1}H_i^T$ is not a full rank matrix. Then, there must exist a full rank matrix $U$ such that the first row of $UG_{i-1}H_i^T$ is all-zero. Denote the first row of $UG_{i-1}$ by a vector $\boldsymbol{g}$. We have $\boldsymbol{g}H_i^T=\mathbf{0}$. We also have $\boldsymbol{g}H_j^T=\mathbf{0}$, for all $1\leqslant j<i$, due to the duality between the two matrices $G(\mathcal{C}_{i-1})$ and $H(\mathcal{C}_{i-1})$. Therefore, we get $\boldsymbol{g}\in\ker\Big(H(\mathcal{C}_i)\Big)$.

On the other hand, due to the duality between the two matrices $G(\mathcal{C}_i)$ and $H(\mathcal{C}_i)$, $\ker\Big(H(\mathcal{C}_i)\Big)$ is the row space of $G(\mathcal{C}_i)$, i.e., $\ker\Big(H(\mathcal{C}_i)\Big)=\mathcal{R}\Big(G(\mathcal{C}_i)\Big)$. However, the vector $\boldsymbol{g}$ does not in the row space of $G(\mathcal{C}_i)$, i.e., $\boldsymbol{g}\notin\mathcal{R}\Big(G(\mathcal{C}_i)\Big)$, so we get a contradiction.

Thus, our assumption is violated. ∎

**Lemma 16.** *For $1\leqslant i\leqslant M$, using the matrices $H_1',H_2',\ldots,H_i'$ from Algorithm 2, the matrix $H'(\mathcal{C}_i)$ in (14) is a parity-check matrix of $\mathcal{C}_i$.*

*Proof:* We need to show the row space of $H'(\mathcal{C}_i)$ is equal to the row space of $H(\mathcal{C}_i)$, i.e., $\mathcal{R}\Big(H'(\mathcal{C}_i)\Big)=\mathcal{R}\Big(H(\mathcal{C}_i)\Big)$. The proof is based on mathematical induction.

First, observe that $H'(\mathcal{C}_1)=H(\mathcal{C}_1)=H_1$, so $H'(\mathcal{C}_1)$ is a parity-check matrix of $\mathcal{C}_1$.

Second, we have $H'(\mathcal{C}_2)=\begin{bmatrix}H'(\mathcal{C}_1)\\ B_2H_2\end{bmatrix}$ and $H(\mathcal{C}_2)=\begin{bmatrix}H(\mathcal{C}_1)\\ H_2\end{bmatrix}$. Since $B_2$ is a full rank matrix according to Lemma 15 and $\mathcal{R}\Big(H'(\mathcal{C}_1)\Big)=\mathcal{R}\Big(H(\mathcal{C}_1)\Big)$ from the previous step, we have $\mathcal{R}\Big(H'(\mathcal{C}_2)\Big)=\mathcal{R}\Big(H(\mathcal{C}_2)\Big)$. Thus, $H'(\mathcal{C}_2)$ is a parity-check matrix of $\mathcal{C}_2$.

Now, for $i$, $2 \leqslant i \leqslant M-1$, assume that $\mathcal{R}\Big(H'(\mathcal{C}_i)\Big) = \mathcal{R}\Big(H(\mathcal{C}_i)\Big)$, we show $\mathcal{R}\Big(H'(\mathcal{C}_{i+1})\Big) = \mathcal{R}\Big(H(\mathcal{C}_{i+1})\Big)$. To see this, the matrix $H'(\mathcal{C}_{i+1})$ has the following form

$$H'(\mathcal{C}_{i+1}) = \begin{bmatrix} H'(\mathcal{C}_i) \\ H'_{i+1} \end{bmatrix} = \begin{bmatrix} H'(\mathcal{C}_i) \\ B_{i+1}H_{i+1} + F_{i+1} \end{bmatrix},$$

where the matrix $F_{i+1} = \sum_{j=2}^{i} T_{j,i+1}H'_j$ from Algorithm 2, so it is easy to see $\mathcal{R}\Big(F_{i+1}\Big) \subseteq \mathcal{R}\Big(H'(\mathcal{C}_i)\Big)$. Compare $H'(\mathcal{C}_{i+1})$ to $H(\mathcal{C}_{i+1})$ which is in the form

$$H(\mathcal{C}_{i+1}) = \begin{bmatrix} H(\mathcal{C}_i) \\ H_{i+1} \end{bmatrix}.$$

Since $B_{i+1}$ is a full rank matrix according to Lemma 15, and $\mathcal{R}\Big(H'(\mathcal{C}_i)\Big) = \mathcal{R}\Big(H(\mathcal{C}_i)\Big)$ from our assumption, we conclude that $\mathcal{R}\Big(H'(\mathcal{C}_{i+1})\Big) = \mathcal{R}\Big(H(\mathcal{C}_{i+1})\Big)$. ∎

**Lemma 17.** *For $2 \leqslant i \leqslant M$, we have $\pi_{\mathcal{N}_i}(\boldsymbol{u}) = \boldsymbol{s}_i = \boldsymbol{u}G(\mathcal{C}_1)H_i'^T$.*

*Proof:* For $2 \leqslant i \leqslant M$, we have

$$\boldsymbol{s}_i = \boldsymbol{u}G(\mathcal{C}_1)H_i'^T = \boldsymbol{u}\begin{bmatrix} G_1H_i'^T \\ G_2H_i'^T \\ \vdots \\ G_MH_i'^T \end{bmatrix}.$$

Thus, to prove $\boldsymbol{s}_i = \pi_{\mathcal{N}_i}(\boldsymbol{u})$, we need to show $G_{i-1}H_i'^T = I$ and $G_jH_i'^T = \mathbf{0}$ for $1 \leqslant j \leqslant M$ and $j \neq i-1$. From Lemma 16, we have $G_jH_i'^T = \mathbf{0}$ for $j \geqslant i$. Therefore, we only need to show $G_{i-1}H_i'^T = I$ and $G_jH_i'^T = \mathbf{0}$ for $j < i-1$. The proof is based on mathematical induction.

For $i = 2$, we prove $G_1H_2'^T = I$.

Since $H'_2 = B_2H_2$, we directly have

$$G_1H_2'^T = G_1H_2^TB_2^T = G_1H_2^T(G_1H_2^T)^{-1} = I.$$

For $i = 3$, we prove $G_2H_3'^T = I$ and $G_1H_3'^T = \mathbf{0}$.

The matrix $H'_3$ has the following form,

$$\begin{aligned} H'_3 =& H_3(2) + T_{2,3}H_2(2) \\ =& B_3H_3 + T_{2,3}H'_2. \end{aligned}$$

Thus, we have

$$G_2H_3'^T = G_2H_3^TB_3^T + G_2H_2'^TT_{2,3}^T \overset{(a)}{=} I,$$

where step (a) follows from $B_3 = (H_3G_2^T)^{-1}$ and $G_2H_2'^T = \mathbf{0}$ according to Lemma 16. We also have

$$\begin{aligned} G_1H_3'^T =& G_1H_3^TB_3^T + G_1H_2'^TT_{2,3}^T \\ \overset{(a)}{=}& G_1H_3^TB_3^T + T_{2,3}^T \overset{(b)}{=} \mathbf{0}, \end{aligned}$$

where step (a) is due to $G_1H_2'^T = I$ from the previous step and step (b) follows from $T_{2,3} = -H_3(2)G_1^T = -B_3H_3G_1^T$.

Now, for $n = 2, 3, \ldots, i$, assume that $G_{n-1}H_n'^T = I$ and $G_jH_n'^T = \mathbf{0}$ for $1 \leqslant j < n-1$. In the following, we will show that $G_iH_{i+1}'^T = I$ and $G_jH_{i+1}'^T = \mathbf{0}$ for $1 \leqslant j < i$.

The matrix $H'_{i+1}$ has the following form,

$$\begin{aligned} H'_{i+1} =& H_{i+1}(2) + \sum_{k=2}^{i} T_{k,i+1}H_k(k) \\ =& B_{i+1}H_{i+1} + \sum_{k=2}^{i} T_{k,i+1}H'_k. \end{aligned}$$

First, we have

$$G_iH_{i+1}'^T = G_iH_{i+1}^TB_{i+1}^T + \sum_{k=2}^{i} G_iH_k'^TT_{k,i+1}^T \overset{(a)}{=} I,$$

where step (a) follows from $B_{i+1} = (H_{i+1}G_i^T)^{-1}$ and $G_iH_k'^T = \mathbf{0}$ for $2 \leqslant k \leqslant i$ according to Lemma 16.

Second, for $1 \leqslant j < i$, we have

$$\begin{aligned} G_jH_{i+1}'^T =& G_jH_{i+1}^TB_{i+1}^T + G_j\sum_{k=2}^{i} H_k'^TT_{k,i+1}^T \\ =& G_jH_{i+1}^TB_{i+1}^T + G_j\sum_{k=2}^{j} H_k'^TT_{k,i+1}^T + G_jH_{j+1}'^TT_{j+1,i+1}^T \\ &+ G_j\sum_{k=j+2}^{i} H_k'^TT_{k,i+1}^T \\ \overset{(a)}{=}& G_jH_{i+1}^TB_{i+1}^T + G_jH_{j+1}'^TT_{j+1,i+1}^T \\ \overset{(b)}{=}& G_jH_{i+1}^TB_{i+1}^T + T_{j+1,i+1}^T \overset{(c)}{=} \mathbf{0}, \end{aligned}$$

where step (a) is due to

1) $G_j\sum_{k=2}^{j} H_k'^TT_{k,i+1}^T = \mathbf{0}$ according to Lemma 16.
2) $G_j\sum_{k=j+2}^{i} H_k'^TT_{k,i+1}^T$ according to our assumption.

Step (b) follows from $G_jH_{j+1}'^T = I$ based on our assumption. Step (c) follows from $T_{j+1,i+1} = -B_{i+1}H_{i+1}G_j^T$. ∎

## REFERENCES


[1] W. Alltop, "A method for extending binary linear codes," *IEEE Trans. Inf. Theory*, vol. 30, no. 6, pp. 871–872, Nov. 1984.

[2] E. Arikan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3051–3073, July 2009.

[3] R. Bez, E. Camerlenghi, A. Modelli, and A. Visconti, "Introduction to flash memory," *Proceedings of the IEEE*, vol. 91, no. 4, pp. 489–502, Apr. 2003.

[4] Y. Cai, S. Ghose, E. F. Haratsch, Y. Luo, and O. Mutlu, "Error characterization, mitigation, and recovery in flash-memory-based solid-state drives," *Proceedings of the IEEE*, vol. 105, no. 9, pp. 1666–1704, Sept. 2017.

[5] Y. Cai, E. F. Haratsch, O. Mutlu, and K. Mai, "Error patterns in MLC NAND flash memory: Measurement, characterization, and analysis," in *Proc. Design, Automation & Test in Europe Conference & Exhibition (DATE)*, Dresden, Germany, Mar. 2012, pp. 521–526.

[6] T.-Y. Chen, K. Vakilinia, D. Divsalar, and R. D. Wesel, "Protograph-based raptor-like LDPC codes," *IEEE Trans. Commun.*, vol. 63, no. 5, pp. 1522–1532, May 2015.

[7] R. Comroe and D. Costello, "ARQ schemes for data transmission in mobile radio systems," *IEEE J. Sel. Areas Commun.*, vol. 2, no. 4, pp. 472–481, July 1984.

[8] G. I. Davida and S. M. Reddy, "Forward-error correction with decision feedback," *Information and Control*, vol. 21, no. 2, pp. 117–133, Sept. 1972.

[9] M. A. De Boer, "Almost MDS codes," *Designs, Codes and Cryptography*, vol. 9, no. 2, pp. 143–155, Oct. 1996.

[10] M. El-Khamy, J. Hou, and N. Bhushan, "Design of rate-compatible structured LDPC codes for hybrid ARQ applications," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 6, pp. 965–973, Aug. 2009.

[11] R. G. Gallager, *Information Theory and Reliable Communication*. New York: Wiley, 1968.

[12] N. Goela, S. B. Korada, and M. Gastpar, "On LP decoding of polar codes," in *Proc. IEEE ITW*, Dublin, Ireland, Aug.–Sept. 2010, pp. 1–5.

[13] J. Ha, J. Kim, and S. W. McLaughlin, "Rate-compatible puncturing of low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 50, no. 11, pp. 2824–2836, Nov. 2004.

[14] J. Hagenauer, "Rate-compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Trans. Commun.*, vol. 36, no. 4, pp. 389–400, Apr. 1988.

[15] S.-N. Hong, D. Hui, and I. Marić, "Capacity-achieving rate-compatible polar codes," *IEEE Trans. Inf. Theory*, vol. 63, no. 12, pp. 7620–7632, Dec. 2017.

[16] P. Huang, Y. Liu, X. Zhang, P. H. Siegel, and E. F. Haratsch, "Syndrome-coupled rate-compatible error-correcting codes," in *Proc. IEEE Inf. Theory Workshop (ITW)*, Kaohsiung, Taiwan, Nov. 2017, pp. 454–458.

[17] ——, "Syndrome-coupled rate-compatible error-correcting codes for flash memories," in *Proc. 9th Annual Non-Volatile Memories Workshop (NVMW)*, San Diego, CA, USA, Mar. 2018.

[18] S. B. Korada, "Polar codes for channel and source coding," Ph.D. dissertation, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland, 2009.

[19] H. Krishna and S. Morgera, "A new error control scheme for hybrid ARQ systems," *IEEE Trans. Commun.*, vol. 35, no. 10, pp. 981–990, Oct. 1987.

[20] L. C. Lee, "New rate-compatible punctured convolutional codes for Viterbi decoding," *IEEE Trans. Commun.*, vol. 42, no. 12, pp. 3073–3079, Dec. 1994.

[21] B. Li, D. Tse, K. Chen, and H. Shen, "Capacity-achieving rateless polar codes," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Barcelona, Spain, July 2016, pp. 46–50.

[22] J. Li and K. R. Narayanan, "Rate-compatible low density parity check codes for capacity-approaching ARQ schemes in packet data communications," in *Proc. International Conference on Communications, Internet and Information Technology (CIIT)*, St. Thomas, US Virgin Islands, Nov. 2002, pp. 201–206.

[23] S. Lin and D. J. Costello, *Error Control Coding*. Upper Saddle River, NJ: Prentice Hall, 2004.

[24] S. Lin and P. Yu, "A hybrid ARQ scheme with parity retransmission for error control of satellite channels," *IEEE Trans. Commun.*, vol. 30, no. 7, pp. 1701–1719, July 1982.

[25] R. Liu, P. Spasojevic, and E. Soijanin, "Punctured turbo code ensembles," in *Proc. IEEE Inf. Theory Workshop (ITW)*, Paris, France, Mar.–Apr. 2003, pp. 249–252.

[26] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. New York: Elsevier, 1977.

[27] M. Mondelli, S. H. Hassani, I. Marić, D. Hui, and S.-N. Hong, "Capacity-achieving rate-compatible polar codes for general channels," in *Proc. IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, San Francisco, CA, USA, Mar. 2017, pp. 1–6.

[28] R. Roth, *Introduction to Coding Theory*. New York: Cambridge University Press, 2006.

[29] D. N. Rowitch and L. B. Milstein, "On the performance of hybrid FEC/ARQ systems using rate compatible punctured turbo (RCPT) codes," *IEEE Trans. Commun.*, vol. 48, no. 6, pp. 948–959, June 2000.

[30] W. Ryan and S. Lin, *Channel Codes: Classical and Modern*. New York: Cambridge University Press, 2009.

[31] E. Şaşoğlu, "Polar codes for discrete channels," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Cambridge, Massachusetts, Jul. 2012, pp. 2137–2141.

[32] E. Şaşoğlu, E. Telatar, and E. Arikan, "Polarization for arbitrary discrete memoryless channels," in *Proc. IEEE Inf. Theory Workshop (ITW)*, Taormina, Sicily, Italy, Oct. 2009, pp. 144–148.

[33] W. C. Schmid and R. Schürer. (2014) MinT: Table for Linear Codes. [Online]. Available: http://mint.sbg.ac.at/index.php

[34] M. M. Shihab, J. Zhang, M. Jung, and M. Kandemir, "Revenand: A fast-drift-aware resilient 3d nand flash design," *ACM Trans. Archit. Code Optim.*, vol. 15, no. 2, pp. 17:1–17:26, May 2018. [Online]. Available: http://doi.acm.org/10.1145/3184744

[35] ——, "Addressing fast-detrapping for reliable 3d nand flash design," in *Proc. 10th Annual Non-Volatile Memories Workshop (NVMW)*, San Diego, CA, USA, Mar. 2019.

[36] V. Taranalli, H. Uchikawa, and P. H. Siegel, "Channel models for multi-level cell flash memories based on empirical error analysis," *IEEE Trans. Commun.*, vol. 64, no. 8, pp. 3169–3181, Aug. 2016.

[37] T. Van Nguyen, A. Nosratinia, and D. Divsalar, "The design of rate-compatible protograph LDPC codes," *IEEE Trans. Commun.*, vol. 60, no. 10, pp. 2841–2850, Oct. 2012.

[38] S. B. Wicker and M. J. Bartz, "Type-II hybrid-ARQ protocols using punctured MDS codes," *IEEE Trans. Commun.*, vol. 42, no. 234, pp. 1431–1440, Feb./Mar./Apr. 1994.

[39] E. Yaakobi, L. Grupp, P. H. Siegel, S. Swanson, and J. K. Wolf, "Characterization and error-correcting codes for TLC flash memories," in *Proc. IEEE International Conference on Computing, Networking and Communications (ICNC)*, Maui, HI, USA, Jan.–Feb. 2012, pp. 486–491.

[40] E. Yaakobi, J. Ma, L. Grupp, P. H. Siegel, S. Swanson, and J. K. Wolf, "Error characterization and coding schemes for flash memories," in *Proc. IEEE Global Communications Conference (GLOBECOM) Workshops*, Miami, FL, USA, Dec. 2010, pp. 1856–1860.

[41] M. R. Yazdani and A. H. Banihashemi, "On construction of rate-comptible low-density parity-check codes," *IEEE Communications Letters*, vol. 8, no. 3, pp. 159–161, Mar. 2004.


**Pengfei Huang** (S'13–M'20) received the B.E. degree in electrical engineering from Zhejiang University, Hangzhou, China, in 2010, the M.S. degree in electrical engineering from Shanghai Jiao Tong University, Shanghai, China, in 2013, and the Ph.D. degree in electrical engineering from the University of California San Diego, CA, USA, in 2018. From 2014 to 2018, he was associated with the Center for Memory and Recording Research. Since 2018, he has been with Western Digital Corporation, where he is engaged in research and development on enterprise solid-state drives. His current research interests are coding for distributed storage systems and non-volatile memories.

**Yi Liu** (S'16) received the B.S. degree in physics from Peking University, Beijing, China, in 2014. He is currently working toward the Ph.D. degree with the Department of Electrical and Computer Engineering, University of California, San Diego, where he is associated with the Center for Memory and Recording Research. His current research interests are coding for costly channel and non-volatile memories.

**Xiaojie Zhang** (S'05–M'13) received the B.S. degree in Electrical Engineering from Shanghai Jiao Tong University, Shanghai, China, in 2004, the M.S. degree in Electrical Engineering from Seoul National University, Seoul, Korea, in 2006, and the Ph.D. degree in Electrical Engineering from the University of California in San Diego, La Jolla, in 2012. From 2006 to 2008, he was a system engineer in Samsung Electronics, Suwon, Korea. In 2013, He worked for Samsung Research American on research and development of 5G system. From 2013 to 2018, he worked on data integrity of non-volatile memory storage, including error detection, error correction, and error handling in start-up companies. Since 2019, he has been with Uber Advanced Technologies Group working on autonomous vehicle system design and development. His research interests include cross-layer wireless communication system design, error correction coding theory and its applications, artificial intelligence and self-driving

vehicle.

**Paul H. Siegel** (M'82–SM'90–F'97–LF'19) received the S.B. and Ph.D. degrees in mathematics from Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, in 1975 and 1979, respectively. He held a Chaim Weizmann Postdoctoral Fellowship with the Courant Institute, New York University, New York, NY, USA. He was with the IBM Research Division, San Jose, CA, USA, from 1980 to 1995. He joined the faculty of the University of California, San Diego, CA, USA, in July 1995, where he is currently a Distinguished Professor of Electrical and Computer Engineering in the Jacobs School of Engineering. He is affiliated with the Center for Memory and Recording Research where he holds an Endowed Chair and served as Director from 2000 to 2011. His research interests include information theory and communications, particularly coding and modulation techniques, with applications to digital data storage and transmission. He was a Member of the Board of Governors of the IEEE Information Theory Society from 1991 to 1996 and again from 2009 to 2014. He served as Co-Guest Editor of the May 1991 Special Issue on "Coding for Storage Devices" of the IEEE Transactions on Information Theory. He served the same Transactions as Associate Editor for Coding Techniques from 1992 to 1995, and as Editor-in-Chief from July 2001 to July 2004. He was also Co-Guest Editor of the May/September 2001 two-part issue on "The Turbo Principle: From Theory to Practice" and the February 2016 issue on "Recent Advances in Capacity Approaching Codes" of the IEEE Journal on Selected Areas in Communications. He is a member of the National Academy of Engineering. He was the 2015 Padovani Lecturer of the IEEE Information Theory Society. He was the corecipient of the 2007 Best Paper Award in Signal Processing and Coding for Data Storage from the Data Storage Technical Committee of the IEEE Communications Society. He was the corecipient of the 1992 IEEE Information Theory Society Paper Award and the 1993 IEEE Communications Society Leonard G. Abraham Prize Paper Award.

**Erich F. Haratsch** is Managing Technologist at Seagate Technology, where he leads a team that develops new hardware architectures and firmware algorithms for solid state disks. His research interests include signal processing, error correction, compression, media management, flash translation layer, performance, power and quality of service optimizations for data storage. Earlier in his career, he developed signal processing and error correction technologies for hard disk drive controllers at LSI Corporation and Agere Systems, which shipped in millions of devices. He started his engineering career at Bell Labs Research, where he invented new chip architectures for Gigabit Ethernet over copper and optical communications. He is a frequent speaker at leading industry events, is the author of over 40 peer-reviewed journal and conference papers and holds more than 100 U.S. patents. He received the Outstanding Patents award from Seagate Technology in 2015. He has been on the Program Executive Council of the Flash Memory Summit since 2018. He earned his M.S. and Ph.D. degrees in electrical engineering from the Technical University of Munich (Germany).