

Performance Analysis and Enhanced Chase Decoding of GII-BCH Codes

Xinzheng He, Li Chen, *Senior Member, IEEE*, and Yingquan Wu, *Senior Member, IEEE*

Abstract—Generalized integrated interleaved (GII) codes enable an enhanced error-correction over an array of interleaves (component codes) within a single block. Error-correction performance of GII codes can be further improved by utilizing the soft received information. The existing Chase decoding of GII-BCH codes can achieve a significant coding gain over the hard-decision decoding, but at the cost of complexity. Meanwhile, the existing theoretical characterization of its Chase decoding performance remains complex and partially empirical. This paper introduces a new theoretical characterization for Chase decoding performance of GII-BCH codes. With this characterization, it further proposes two new soft-decision decoding methods for GII-BCH codes, including the enhanced Chase decoding (ECD) and the enhanced concatenated Chase decoding (ECCD). They both identify the decoding rounds that are more likely to declare a decoding failure and prioritize allocating the flipped positions to those rounds. In particular, the latter utilizes codewords of a linear block code to cover the least reliable or the second least reliable positions, further improving the error-correction performance over the ECD. With a similar number of test-vectors, both the ECD and ECCD can outperform the existing Chase decoding for GII-BCH codes.

Index Terms—BCH codes, Chase decoding, generalized integrated interleaved (GII) codes, theoretical performance analysis

I. INTRODUCTION

Advancements in communication technology have led to new requirements for channel coding, which now encompass not only high coding gains but also the need for high throughput. Generalized integrated interleaved (GII) codes [1] are one of the promising candidates that can meet these requirements. An $([m, v], n)$ GII code consists of m interleaves and v nested interleaves, each of length n . The interleaves can either be Reed-Solomon (RS) codes or BCH codes. Decoding of the m interleaves can be performed in parallel. If some interleaves cannot be corrected, linear combinations of the m interleaves can yield v stronger nested interleaves for correcting the remaining errors. Therefore, GII codes can achieve both a high throughput and a high decoding performance. Hard-decision decoding of GII-BCH codes employs the Berlekamp-Massey (BM) algorithm to decode the m interleaves [1] [2].

Xinzheng He is with the School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou 510006, China, and also with Guangdong Province Key Laboratory of Information Security Technology, Guangzhou, 510006, China (e-mail: hexzh9@mail2.sysu.edu.cn;).

Li Chen is with the School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou 510006, China, and also with Guangdong Province Key Laboratory of Information Security Technology, Guangzhou, 510006, China (e-mail: chenli55@mail.sysu.edu.cn).

Yingquan Wu is with Majestic Labs AI, Los Altos, CA, USA (e-mail: yingquan_wu@yahoo.com).

Encoding and decoding of GII codes were proposed in [3]–[7]. Theoretical performance of hard-decision decoding for GII codes was first studied in [1] and later improved in [8].

In order to improve the decoding performance of GII codes, soft received information should be applied to empower the existing algebraic decoding, such as the generalized minimum distance (GMD) decoding [9] and the Chase decoding [10]. The latter is a classic soft-decision decoding that achieves a good trade-off between the error-correction performance and complexity. For BCH codes, Chase decoding flips the η least reliable positions and constructs 2^η test-vectors to ensure all errors that occur in these positions can be corrected. Performance analysis of Chase decoding was studied in [11] [12]. Chase decoding of GII-BCH codes and its decoder architecture were proposed in [13]. The decoding of GII-BCH codes can be divided into $v + 1$ rounds. In decoding round-0, the m interleaves are decoded in parallel. From decoding round-1 to round- v , the nested interleaves are computed to provide more higher order syndromes. The error-correction capability of interleaves vary over different rounds. The Chase decoding of [13] achieves performance improvements by flipping different numbers of the codeword symbols in different decoding rounds. However, it requires that after flipping, the error-correction capability of the current decoding round cannot be greater than the following decoding round.

Chase decoding performance heavily relies on the formulation of test-vectors [14] [15]. A symbol-level-stochastic Chase algorithm (S-SCA) was proposed in [16], which is power-efficient due to random generation of the most likely test-vectors. With a certain number of test-vectors, the S-SCA can achieve near maximum-likelihood (ML) performance. The use of covering codes for selecting the effective test-vectors in the iterative bounded distance decoding algorithms was first proposed in [14], aiming to reduce the decoding complexity while maintaining the decoding performance. Coded Chase decoding [17] utilizes perfect codes, e.g., Hamming codes, as the covering codes to assist the formulation of test-vectors. In particular, it utilizes codewords of a perfect linear block code as the flipping patterns. This results in the minimum Hamming distance between the test-vectors becoming greater. Consequently, with the same number of test-vectors, coded Chase decoding may be able to correct errors over a wider band of a codeword frame. As noted in [17], the conventional Chase decoding and coded Chase decoding can be integrated. For the least reliable positions that have a higher probability of being erroneous, the conventional Chase decoding can be employed to ensure all errors in these positions can be corrected. For the second least reliable positions, coded Chase decoding can be

employed to achieve a broader correction band. In this paper, we call this decoding method concatenated Chase decoding. Consequently, coded Chase decoding can be considered as a special case of concatenated Chase decoding. It has been shown that concatenated Chase decoding can further improve the error-correction performance without any complexity cost [17]. Concatenated Chase decoding is particularly effective when the received information is heavily corrupted. There is lack of theoretical characterization for concatenated Chase decoding performance. This also subsequently limits our understanding on practical deployment of concatenated Chase decoding.

This paper proposes a new theoretical characterization for Chase decoding performance of GII-BCH codes. The more computationally efficient closed form expressions for the decoding error probability are provided, which are also validated by our decoding simulation. This paper further proposes the enhanced Chase decoding (ECD) and the enhanced concatenated Chase decoding (ECCD) for GII-BCH codes. The ECD and the ECCD can outperform the conventional Chase decoding of GII-BCH codes. The major contributions of the paper are as follows.

- 1) This paper proposes a new performance analysis technique for Chase decoding of BCH codes and further extends it for coded Chase decoding and concatenated Chase decoding of the codes. Based on this, the theoretical decoding performance analysis of coded Chase decoding and concatenated Chase decoding of GII-BCH codes is presented in Section III.
- 2) Based on the performance analysis in Section III, this paper further introduces two new Chase decoding methods for GII-BCH codes, including the enhanced Chase decoding (ECD) and the enhanced concatenated Chase decoding (ECCD). They are designed by applying the mentioned theoretical analysis on the decoding performances. With a given SNR, both the ECD and the ECCD can compute the error probability of each decoding round. This enables them to better allocate the flipped positions, resulting in significant performance gains.

The rest of this paper is organized as follows. Section II introduces the prerequisites of this work, including the GII-BCH codes, Chase decoding of GII-BCH codes and concatenated Chase decoding. Section III presents our theoretical performance characterization for Chase decoding and concatenated Chase decoding of GII-BCH codes. The ECD and the ECCD of GII-BCH codes are further proposed in Section IV. Simulation results and comparison with several existing Chase decoding for GII-BCH codes are presented in Section V. Finally, Section VI concludes the paper.

II. PREREQUISITES

This section presents the prerequisites of GII-BCH codes, Chase decoding of GII-BCH codes and concatenated Chase decoding.

A. GII-BCH Codes

Let \mathbb{F}_{2^q} denote the finite field of characteristics two and order q , and σ denote its primitive element. An

$([m, v], n)$ GII-BCH codeword consists of m BCH interleaves $c_0(x), c_1(x), \dots, c_{m-1}(x) \in \mathcal{C}_0(n, k_0, t_0)$. Linear combinations of the m interleaves produce v nested BCH interleaves $\tilde{c}_0(x) \in \mathcal{C}_v(n, k_v, t_v), \tilde{c}_1(x) \in \mathcal{C}_{v-1}(n, k_{v-1}, t_{v-1}), \dots, \tilde{c}_{v-1}(x) \in \mathcal{C}_1(n, k_1, t_1)$, where $k_0 \geq k_1 \geq \dots \geq k_v$ and $t_v \geq \dots \geq t_1 \geq t_0$. They can be defined as

$$\mathcal{C}_G \triangleq \{c(x) = ([c_0(x), c_1(x), \dots, c_{m-1}(x)]), c_i(x) \in \mathcal{C}_0, 0 \leq i \leq m-1; \tilde{c}_j(x) = \sum_{i=0}^{m-1} \alpha(x^{ij})c_i(x) \in \mathcal{C}_{v-j}, 0 \leq j < v\}, \quad (1)$$

where $\alpha(x^\mu)$ is defined by the primitive polynomial $\psi(x)$ of \mathbb{F}_{2^q} through

$$\alpha(x^\mu) = x^\mu \bmod \psi(x), \quad (2)$$

where $\mu \in \mathbb{N}$. Thus, GII-BCH code is of length $N = mn$ and dimension $K = (m-v)k_0 + k_1 + k_2 + \dots + k_v$.

For an $([m, v], n)$ GII-BCH code, decoding can be divided into $v+1$ rounds. In decoding round- b , where $0 \leq b \leq v$, each interleave can correct $t_b = \lfloor (\gamma_b - 1)/2 \rfloor$ errors, where γ_b is the designed minimum distance of \mathcal{C}_b . The hard-decision error-correction capability is $\mathbf{t} = [t_0, t_1, \dots, t_v]$ [1]. Hard-decision decoding of GII-BCH codes employs the BM algorithm [18] to decode the m interleaves. The v nested interleaves can provide the higher order syndromes for the remaining interleaves. Let $y_i(x)$ denote the i -th received word of the transmitted codeword $c_i(x) \in \mathcal{C}_0$. Syndromes of $y_i(x)$ are computed as

$$S_i^j = y_i(\sigma^{j+1}), 0 \leq j < 2t_0. \quad (3)$$

In decoding round-0, i.e., $b = 0$, the error locator polynomial $\Lambda(x)$ for $y_i(x)$ can be computed by the BM algorithm. Inverse of its roots indicate the error locations. For binary BCH codes, decoding is completed once the error locations are determined. At this round, the BM algorithm can correct up to t_0 errors in the received word of each interleave. If all interleaves are correctly decoded, the decoding will terminate. Otherwise, the nested structure needs to be applied through computing the higher order syndromes. They help decode the remaining interleaves. Note that if the number of the remaining interleaves exceeds v , decoding will also terminate but with a failure.

In decoding round- b and $b \neq 0$, there are $v-b+1$ interleaves that have not been successfully decoded. Their corresponding received words are denoted as $y_{l_0}(x), y_{l_1}(x), \dots, y_{l_{v-b}}(x)$. Let $L = \{l_0, l_1, \dots, l_{v-b}\}$ denote the index set of the $v-b+1$ interleaves. Hence, its complementary set L^c denotes the interleaves that have been successfully decoded. The $v-b+1$ nested interleaves can be computed as

$$\tilde{y}_i(x) = \sum_{j \in L} \alpha(x^{ij})y_j(x) + \sum_{j \in L^c} \alpha(x^{ij})c_j(x), \quad (4)$$

where $i = 0, 1, \dots, v-b$. Syndromes of the nested interleave $\tilde{y}_i(x)$ can be computed as

$$\tilde{S}_i^j = \tilde{y}_i(\sigma^{j+1}), 0 \leq j < 2t_b. \quad (5)$$

The higher order syndromes of the remaining interleaves can be computed through syndromes of the nested interleaves by

$$\begin{bmatrix} S_{l_0}^j \\ S_{l_1}^j \\ \vdots \\ S_{l_{v-b}}^j \end{bmatrix} = \begin{bmatrix} 1 & \cdots & 1 \\ \alpha(x^{l_0}) & \cdots & \alpha(x^{l_{v-b}}) \\ \vdots & \ddots & \vdots \\ \alpha(x^{(v-b)l_0}) & \cdots & \alpha(x^{(v-b)l_{v-b}}) \end{bmatrix}^{-1} \begin{bmatrix} \tilde{S}_0^j \\ \tilde{S}_1^j \\ \vdots \\ \tilde{S}_{v-b}^j \end{bmatrix}, \quad (6)$$

(6), where $j = 2t_0, 2t_0 + 1, \dots, 2t_b - 1$. For the $v - b + 1$ remaining interleaves, the number of syndromes has increased to $2t_b$. This enables each interleave to correct up to t_b errors in decoding round- b . If the number of successfully decoded interleaves does not increase in this round, the decoding will also terminate again with a failure. Otherwise, more higher order syndromes can be computed through (6) by entering into the next decoding round, thereby improving the error-correction capability of the remaining interleaves. This process is repeated until all interleaves have been decoded, or a decoding failure is declared.

B. Chase Decoding of GII-BCH Codes

By utilizing the soft received information, decoding performance of GII-BCH codes can be further improved. In its Chase decoding [13], if hard-decision decoding fails in decoding round- b , η_b least reliable positions (LRPs) will be identified in the remaining interleaves. Subsequently, 2^{η_b} test-vectors will be constructed by flipping the decisions at these positions for each of them. By decoding 2^{η_b} test-vectors, each interleave can correct up to $\eta_b + t_b$ errors in the decoding round. Through Chase decoding, the number of remaining interleaves may decrease, which will lead to the next decoding round. When $b \neq 0$, the nested interleave and higher order syndromes should be recomputed for each test-vector. Let η_b denote the number of flipped positions at round- b , then the flipping vector can be denoted as $\boldsymbol{\eta} = [\eta_0, \eta_1, \dots, \eta_v]$. The decoding algorithm is described as in Algorithm 1.

It can be seen that $\boldsymbol{\eta}$ determines the error-correction capability and complexity for GII-BCH codes. Therefore, designing $\boldsymbol{\eta}$ under a certain complexity constraint is critical for Chase decoding of GII-BCH codes. In the existing Chase decoding of GII-BCH codes [13], entries of $\boldsymbol{\eta}$ need to satisfy $t_b + \eta_b < t_{b+1}$. However, this limits the Chase decoding performance, especially under a certain complexity constraint. In our proposed Chase decoding, this conditions is lifted, leading to an enhanced decoding performance.

C. Concatenated Chase Decoding

In the conventional Chase decoding of BCH codes, all 2^{η} test-vectors will be decoded. Despite decoding multiple test-vectors, their decoding outputs are often the same. Therefore, forming an effective set of test-vectors is important for achieving a good decoding performance-complexity trade-off. The use of covering codes in forming the test-vectors was first proposed in [14] for soft-decision bounded distance decoding algorithms. Coded Chase Kötter-Vardy decoding [17] chooses the codewords of a perfect covering code as the Chase flipping patterns. Let $\mathcal{C}_c(\eta, \tau)$ denote a perfect

Algorithm 1: Chase Decoding of GII-BCH Codes

Input: $y_0(x), y_1(x), \dots, y_{m-1}(x)$,
 $\boldsymbol{\eta} = [\eta_0, \eta_1, \dots, \eta_v]$;
Output: $c_0(x), c_1(x), \dots, c_{m-1}(x)$;

- 1 Initialize $L = \emptyset, b = 0$;
- 2 **For** $i = 0, 1, \dots, m - 1$ **do**
- 3 Decode $y_i(x)$ by BM algorithm;
- 4 **If** the decoding declares a failure **then**
- 5 $L = L \cup \{i\}$;
- 6 **Else**
- 7 Return $c_i(x)$;
- 8 **If** $|L| > v$ **then**
- 9 **For** $i \in L$ **do**
- 10 Construct 2^{η_0} test-vectors and decode them;
- 11 **If** the decoding declares a success **then**
- 12 $L = L \setminus \{i\}$ and return $c_i(x)$;
- 13 **If** $|L| > v$ **then**
- 14 Declare a decoding failure;
- 15 **Else**
- 16 $b = |L|$;
- 17 **While** $L \neq \emptyset$ **do**
- 18 **For** $i \in L$ **do**
- 19 Compute the higher order syndromes of
- 20 $y_i(x)$ as in (6) and decode it;
- 21 **If** decoding of $y_i(x)$ declares a success
- 22 **then**
- 23 $L = L \setminus \{i\}$ and return $c_i(x)$;
- 24 **If** $|L|$ does not decrease **then**
- 25 **For** $i \in L$ **do**
- 26 Construct 2^{η_b} test-vectors and decode
- 27 them;
- 28 **If** decoding of $y_i(x)$ declares a success
- 29 **then**
- 30 $L = L \setminus \{i\}$ and return $c_i(x)$;
- 31 **If** $|L|$ does not decrease **then**
- 32 Declare a decoding failure.

sphere-packing code with length η and dimension τ . Theoretical decoding performance of utilizing the codewords of a perfect code as the Chase flipping patterns can be analyzed more straightforwardly. Coded Chase decoding utilizes each codeword of $\mathcal{C}_c(\eta, \tau)$ as the flipping pattern which can cover the η LRPs. There are 2^τ test-vectors, covering η LRPs. When $\tau = \eta$, the conventional Chase decoding can be viewed as a special case of coded Chase decoding. Coded Chase decoding can increase the minimum Hamming distance between test-vectors, enhancing the error-correction capability. Let d_{\min} ($d_{\min} \geq 1$) denote the minimum Hamming distance of

TABLE I
FLIPPING PATTERNS OF CHASE DECODING WITH FOUR FLIPPED POSITIONS AND THOSE OF CODED CHASE DECODING USING THE (7, 4) HAMMING CODE.

Chase decoding	Coded Chase decoding
0000	0000000
0001	0001101
0010	0010111
0011	0011010
0100	0100011
0101	0101110
0110	0110100
0111	0111001
1000	1000110
1001	1001011
1010	1010001
1011	1011100
1100	1100101
1101	1101000
1110	1110010
1111	1111111

$\mathcal{C}_c(\eta, \tau)$. For the conventional Chase decoding, the minimum Hamming distance between the test-vectors is one. For coded Chase decoding that employs $\mathcal{C}_c(\eta, \tau)$ as the covering code, this will be increased to d_{\min} . Note that when there is a severe noise interference, the minimum Hamming distance between a received codeword and a valid codeword tends to be relatively large. Table I illustrates all the flipping patterns of a Chase decoding with four flipped positions and those of a coded Chase decoding using the (7, 4) Hamming code to cover seven LRPCs. It can be seen that the Hamming distance between the test-vectors of coded Chase decoding is greater, spanning a wider error-correction band.

However, it should be aware that coded Chase decoding can only be employed for enhancing the conventional Chase decoding that flips τ positions. Moreover, applying coded Chase decoding is possible to overkill a received word, failing an otherwise successful decoding. For this, the conventional Chase decoding and coded Chase decoding can be integrated. We call this method concatenated Chase decoding [17]. In concatenated Chase decoding, the conventional Chase decoding is employed for flipping the $\tilde{\eta}$ LRPCs, while coded Chase decoding is employed for flipping the η secondary least reliable positions (SLRPCs). This combined flipping approach appears to be effective for enhancing the Chase decoding performance [17]. Under such a setup, coded Chase decoding can be considered as a special case of concatenated Chase decoding with $\tilde{\eta} = 0$.

Concatenated Chase decoding can also be employed to decode GII-BCH codes. Let us assume that each interleave can correct up to $t_b + \tau + \tilde{\eta}$ errors in decoding round- b by flipping $\tau + \tilde{\eta}$ positions. If a good sphere-packing code $\mathcal{C}_c(\eta, \tau)$ is utilized to cover the η SLRPCs, each interleave can correct up to $t_b + \eta + \tilde{\eta}$ errors. It corrects more errors than the conventional Chase decoding.

III. CHASE DECODING PERFORMANCE ANALYSIS

This section proposes a new theoretical decoding performance characterization for Chase decoding of BCH codes and further extends it for coded Chase decoding and concatenated Chase decoding of the codes. Based on this, theoretical decoding performance characterization of coded Chase decoding and concatenated Chase decoding of GII-BCH codes is proposed.

A. Chase Decoding of BCH Codes

For an (n, k, t) BCH code (denoted as $\mathcal{C}(n, k, t)$) with η flipped positions, the error probability of Chase decoding can be characterized as the sum of the probabilities of two types of decoding failures. They include when the number of errors exceeds $t + \eta$, and when the number of errors is between $t + 1$ and $t + \eta$ (but the decoding still fails). To characterize the latter, the existing method of [11] exhibits double integrals, which is computationally challenging.

This paper proposes a new method for the pursuit through simplifying computation while ensuring accuracy. Some of the ordered statistics decoding (OSD) performance characterization techniques of [19] are applied. For a Chase decoding that flips the η LRPCs, all the errors in the η LRPCs can be corrected. Error probability of Chase decoding can be characterized as the probability that the number of errors in the most reliable positions (MRPs) exceeds t . Let us assume the codeword is transmitted over the additive white Gaussian noise (AWGN) channel together with the binary phase shift keying (BPSK) modulation, where noise variance is $N_0/2$ and N_0 is the single side-band power spectrum density. The signal-to-noise ratio (SNR) is defined as E_b/N_0 , where E_b is the transmitted energy per information bit. For simplicity, we assume an all-zero codeword of $\mathcal{C}(n, k, t)$ is transmitted. Let $\mathbf{r} = (r_1, r_2, \dots, r_n) \in \mathbb{R}^n$ denote the channel output and $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \dots, \alpha_n) = (|r_1|, |r_2|, \dots, |r_n|) \in \mathbb{R}^n$ denote its reliability vector. Let R_u and A_u further denote the random variables representing r_u and α_u , respectively. The pdf of R_u and A_u are given by

$$f_R(r) = \frac{1}{\sqrt{\pi N_0}} e^{-\frac{(r-1)^2}{N_0}} \quad (7)$$

and

$$f_A(\alpha) = \begin{cases} 0, & \text{if } \alpha < 0; \\ \frac{e^{-\frac{(\alpha+1)^2}{N_0}}}{\sqrt{\pi N_0}} + \frac{e^{-\frac{(\alpha-1)^2}{N_0}}}{\sqrt{\pi N_0}}, & \text{if } \alpha \geq 0, \end{cases} \quad (8)$$

respectively. Given the Q -function defined as

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{z^2}{2}} dz, \quad (9)$$

the cdf of A_u can be derived as

$$F_A(\alpha) = \begin{cases} 0, & \text{if } \alpha < 0; \\ 1 - Q\left(\frac{\alpha + 1}{\sqrt{N_0/2}}\right) - Q\left(\frac{\alpha - 1}{\sqrt{N_0/2}}\right), & \text{if } \alpha \geq 0. \end{cases} \quad (10)$$

Let $\dot{\alpha} = (\dot{\alpha}_1, \dot{\alpha}_2, \dots, \dot{\alpha}_n)$ denote a sorted reliability vector, where $\dot{\alpha}_1 \geq \dot{\alpha}_2 \geq \dots \geq \dot{\alpha}_n$. Let \dot{A}_u further denote the random variable representing $\dot{\alpha}_u$. Similar to (8), the pdf of \dot{A}_u can be derived as

$$f_{\dot{A}_u}(\dot{\alpha}_u) = \frac{n!}{(u-1)!(n-u)!} \cdot (1 - F_A(\dot{\alpha}_u))^{u-1} \cdot F_A(\dot{\alpha}_u)^{n-u} f_A(\dot{\alpha}_u). \quad (11)$$

For a position with its reliability value ranging in $[\beta, \alpha]$, its error probability $p(\alpha, \beta)$ can be derived as

$$p(\alpha, \beta) = \frac{Q\left(\frac{-2\alpha-2}{\sqrt{2N_0}}\right) - Q\left(\frac{-2\beta-2}{\sqrt{2N_0}}\right)}{Q\left(\frac{-2\alpha-2}{\sqrt{2N_0}}\right) - Q\left(\frac{-2\beta-2}{\sqrt{2N_0}}\right) + Q\left(\frac{2\alpha-2}{\sqrt{2N_0}}\right) - Q\left(\frac{2\beta-2}{\sqrt{2N_0}}\right)}. \quad (12)$$

For Chase decoding that flips the η LRPs, the received word can be categorized into the $n - \eta$ MRPs and the η LRPs, respectively. The probability of having ϵ errors in the MRPs can be derived as

$$p_{\text{MRPs}}(\eta, \epsilon) = \int_0^\infty \binom{n-\eta}{\epsilon} p(\infty, y)^\epsilon (1 - p(\infty, y))^{n-\eta-\epsilon} \cdot f_{\dot{A}_{n-\eta+1}}(y) dy. \quad (13)$$

Errors in the LRPs can be corrected by Chase decoding. Therefore, error probability of Chase decoding can be characterized as the event in which the number of errors in the MRPs exceeds the error-correction capability t , which is

$$p_{\text{Chase}}(t, \eta) = 1 - \sum_{\epsilon=0}^t p_{\text{MRPs}}(\eta, \epsilon). \quad (14)$$

Compared with the method of [11], the theoretical characterization of (14) appears to be more computationally efficient. We will later show it is also accurate.

B. Coded Chase Decoding of BCH Codes

Assume that a perfect sphere-packing code $\mathcal{C}_c(\eta, \tau)$ is employed to cover the η LRPs in coded Chase decoding of an (n, k, t) BCH code. Let \mathbf{f}_μ and \mathbf{e}_ν denote the flipping pattern and the error pattern in the LRPs, respectively, where $1 \leq \mu \leq 2^\tau$ and $1 \leq \nu \leq 2^\eta$. The number of flipping patterns is smaller than that of error patterns. There are certain error patterns that cannot be handled by the decoding. For such error patterns, errors may still remain, or extra errors may occur after the flipping. Both of them are referred to as the erroneous flips. Let us define $\xi(\mathbf{e}_\nu, \mathcal{C}_c(\eta, \tau))$ as the number of erroneous flips as

$$\xi(\mathbf{e}_\nu, \mathcal{C}_c(\eta, \tau)) = \min\{\text{weight}(\mathbf{e}_\nu \oplus \mathbf{f}_\mu), \mathbf{f}_\mu \in \mathcal{C}_c, \mu = 1, 2, \dots, 2^\tau\}, \quad (15)$$

where $\text{weight}(\mathbf{e}_\nu \oplus \mathbf{f}_\mu)$ denotes the number of non-zero entries in $\mathbf{e}_\nu \oplus \mathbf{f}_\mu$. For an error pattern \mathbf{e}_ν , if $\xi(\mathbf{e}_\nu, \mathcal{C}_c(\eta, \tau)) = 0$, it can be corrected by coded Chase decoding. With such a definition, Chase decoding can be considered as a special case of coded Chase decoding, in which all error patterns in the LRPs satisfy $\xi(\mathbf{e}_\nu, \mathcal{C}_c(\eta, \tau)) = 0$. Therefore, the error probability of Chase decoding depends only on the number of errors in the MRPs. For coded Chase decoding, it remains

that the number of flipping patterns is less than that of error patterns. This implies that for certain error patterns, there is no flipping pattern that can completely match them. For those error patterns, $\xi(\mathbf{e}_\nu, \mathcal{C}_c(\eta, \tau)) \neq 0$.

Lemma 1. When a perfect covering code $\mathcal{C}_c(\eta, \tau)$ that can correct at most ω errors is used to cover the η LRPs, the error pattern \mathbf{e}_ν should satisfy $\xi(\mathbf{e}_\nu, \mathcal{C}_c(\eta, \tau)) \leq \omega$.

Proof: For a perfect covering code $\mathcal{C}_c(\eta, \tau)$ that can correct at most ω errors, all length- n binary vectors are contained in the Hamming sphere of a valid codeword. Each vector can be regarded as an error pattern and each valid codeword of $\mathcal{C}_c(\eta, \tau)$ can be regarded as a flipping pattern of coded Chase decoding. Hence, the Hamming distance between each error pattern and its nearest flipping pattern is not greater than ω , i.e., $\xi(\mathbf{e}_\nu, \mathcal{C}_c(\eta, \tau)) \leq \omega$. \square

Based on Lemma 1, it can be seen that there are at most ω erroneous flips occurring in the LRPs. The probability of having ϵ' ($0 \leq \epsilon' \leq \omega$) erroneous flips in the LRPs is

$$p_{\text{LRPs}}(\eta, \epsilon') = \sum_{\nu} p_l(\mathbf{e}_\nu, \eta) \cdot \mathbf{1}_{[\epsilon', \epsilon']}(\xi(\mathbf{e}_\nu, \mathcal{C}_c(\eta, \tau))), \quad (16)$$

where $p_l(\mathbf{e}_\nu, \eta)$ is the probability of an error pattern $\mathbf{e}_\nu = (e_{n-\eta+1}, e_{n-\eta+2}, \dots, e_n)$ occurring in the LRPs [19] and it is defined as

$$p_l(\mathbf{e}_\nu, \eta) = \underbrace{\int_0^\infty \dots \int_0^\infty}_{\eta\text{-weight}(\mathbf{e}_\nu)} \dots \underbrace{\int_{-\infty}^0 \dots \int_{-\infty}^0}_{\text{weight}(\mathbf{e}_\nu)} \left(\frac{n!}{(n-\eta)!} F_A(x_{n-\eta+1})^{n-\eta} \cdot \prod_{z=n-\eta+1}^n f_R(x_z) \cdot \prod_{z=n-\eta+2}^n \mathbf{1}_{[0, |x_{z-1}|]}(|x_z|) \right) \cdot \prod_{n-\eta < z < n, e_z=0} dx_z \prod_{n-\eta < z < n, e_z \neq 0} dx_z. \quad (17)$$

and function $\mathbf{1}_{[a,b]}(x)$ is defined as

$$\mathbf{1}_{[a,b]}(x) = \begin{cases} 1, & \text{if } x \in [a, b]; \\ 0, & \text{if } x \notin [a, b]. \end{cases} \quad (18)$$

Error probability of coded Chase decoding can be characterized as probability of the event where the total number of errors in the MRPs and erroneous flips in the LRPs exceeds the error-correction capability t . That says

$$p_{\text{Coded}}(t, \eta) = \sum_{\epsilon=t+1}^n p_{\text{MRPs}}(\eta, \epsilon) + \sum_{\epsilon=1}^{\omega} \left(p_{\text{MRPs}}(\eta, t - \epsilon + 1) \cdot \sum_{\epsilon'=\epsilon}^{\omega} p_{\text{LRPs}}(\eta, \epsilon') \right). \quad (19)$$

The following Example 1 demonstrates the coded Chase decoding performance characterization.

Example 1. For a (63, 18) BCH code with an error-correction capability $t = 10$, the (7, 4) Hamming code $\mathcal{C}_{\text{Ham}}(7, 4)$ is employed to cover the seven LRPs. Among these LRPs, there are in total 2^7 error patterns. The (7, 4) Hamming code is a perfect code with a minimum Hamming distance of

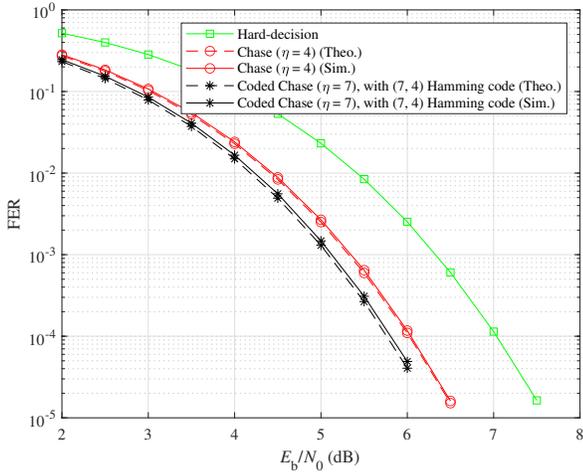


Fig. 1. Error-correction performance of Chase decoding and coded Chase decoding for the (63, 18) BCH code.

three. Based on Lemma 1, there is at most one erroneous flip occurring in the LRPs. Excluding the 16 error patterns that are the Hamming codewords, the remaining error patterns have an erroneous flip. That says

$$\begin{aligned} p_{\text{LRPs}}(7, 1) &= \sum_{\nu} p_l(\mathbf{e}_{\nu}, 7) \cdot \mathbf{1}_{[1,1]}(\xi(\mathbf{e}_{\nu}, \mathcal{C}_{\text{Ham}}(7, 4))) \\ &= \sum_{\mathbf{e}_{\nu} \neq \mathbf{f}_{\mu}} p_l(\mathbf{e}_{\nu}, 7), \end{aligned}$$

where $\mathbf{f}_{\mu} \in \mathcal{C}_{\text{Ham}}(7, 4)$. The decoding error probability can be calculated by (19) as

$$p_{\text{Coded}}(10, 7) = \sum_{\epsilon=11}^{63} p_{\text{MRPs}}(7, \epsilon) + p_{\text{MRPs}}(7, 10)p_{\text{LRPs}}(7, 1).$$

Fig. 1 shows the simulation results and the theoretical characterizations of Chase decoding and coded Chase decoding of the (63, 18) BCH code. The decoding performances (as in frame error rate (FER)) were obtained over the AWGN channel using BPSK. It can be seen that with the same number of test-vectors, coded Chase decoding outperforms the conventional Chase decoding. Coded Chase decoding covers seven LRPs, correcting up to seven errors. Moreover, Fig.1 shows our theoretical characterizations match well with simulation results.

C. Concatenated Chase Decoding of BCH Codes

Concatenated Chase decoding performance can be analyzed similarly. For concatenated Chase decoding, the received word can be categorized into the $n - \eta - \tilde{\eta}$ MRPs, the $\tilde{\eta}$ LRPs and the η SLRPs.

In concatenated Chase decoding, Chase decoding is employed to cover the $\tilde{\eta}$ LRPs, while coded Chase decoding is employed to cover the η SLRPs. Errors in the LRPs can be completely corrected, while the erroneous flips occur in the SLRPs. The probability of having ϵ' ($0 \leq \epsilon' \leq \omega$) erroneous flips in the SLRPs can be derived as

$$p_{\text{SLRPs}}(\eta, \tilde{\eta}, \epsilon') = \sum_{\nu} p_l(\tilde{\mathbf{e}}_{\nu}, \eta + \tilde{\eta}) \cdot \mathbf{1}_{[\epsilon', \epsilon']}(E(\tilde{\mathbf{e}}_{\nu}, \mathcal{C}_c(\eta, \tau))), \quad (20)$$

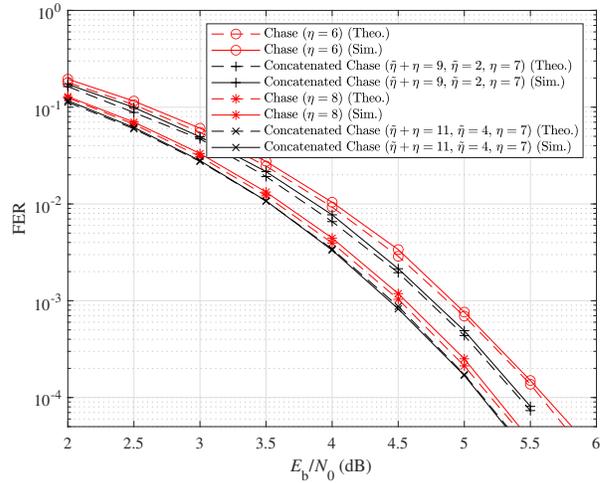


Fig. 2. Error-correction performance of Chase decoding and concatenated Chase decoding for the (63, 18) BCH code.

where $p_l(\tilde{\mathbf{e}}_{\nu}, \eta + \tilde{\eta})$ is the probability of an error pattern $\tilde{\mathbf{e}}_{\nu} = (\tilde{e}_{n-\eta-\tilde{\eta}+1}, \tilde{e}_{n-\eta-\tilde{\eta}+2}, \dots, \tilde{e}_{n-\tilde{\eta}})$ occurring in the SLRPs. Error probability of concatenated Chase decoding can be derived as

$$\begin{aligned} p_{\text{Conc}}(t, \eta, \tilde{\eta}) &= \sum_{\epsilon=t+1}^n p_{\text{MRPs}}(\eta + \tilde{\eta}, \epsilon) \\ &+ \sum_{\epsilon=1}^{\omega} \left(p_{\text{MRPs}}(\eta + \tilde{\eta}, t - \epsilon + 1) \cdot \sum_{\epsilon'=\epsilon}^{\omega} p_{\text{SLRPs}}(\eta, \tilde{\eta}, \epsilon') \right). \end{aligned} \quad (21)$$

The following Example 2 demonstrates this concatenated Chase decoding performance characterization.

Example 2. For a (63, 18) BCH code with an error-correction capability $t = 10$, Chase decoding is employed to cover the four LRPs, while coded Chase decoding is employed to cover the seven SLRPs by the use of the (7, 4) Hamming code. The decoding error probability can be calculated by (21) as

$$\begin{aligned} p_{\text{Conc}}(10, 7, 4) &= \sum_{\epsilon=11}^{63} p_{\text{MRPs}}(11, \epsilon) \\ &+ p_{\text{MRPs}}(11, 10) \cdot p_{\text{SLRPs}}(7, 4, 1). \end{aligned}$$

Fig. 2 compares our simulation results with the theoretical characterizations of Chase decoding and concatenated Chase decoding for the (63, 18) BCH code. It can be seen that the concatenated Chase decoding cannot only enhance error-correction performance but also offer significant flexibility in its application. Coded Chase decoding can only be applied when flipping τ positions. However, concatenated Chase decoding can be applied when at least τ positions need to be flipped. It can be seen that with the same number of test-vectors, concatenated Chase decoding also outperforms Chase decoding, due to its capability of covering more unreliable positions and hence enhancing the correction ability. Again, the theoretical characterizations also match well with simulation results.

$$P_{f_b} = \begin{cases} \sum_{i=v+1}^m \binom{m}{i} p_{\text{Chase}}(t_0, \eta_0)^i \cdot (1 - p_{\text{Chase}}(t_0, \eta_0))^{m-i}, & \text{if } b = 0; \\ \binom{m}{v-b+1} p_{\text{Chase}}(t_b, \eta_b)^{v-b+1} \cdot (1 - p_{\text{Chase}}(t_{b-1}, \eta_{b-1}))^{m-(v-b+1)}, & \text{if } 1 \leq b \leq v. \end{cases} \quad (25)$$

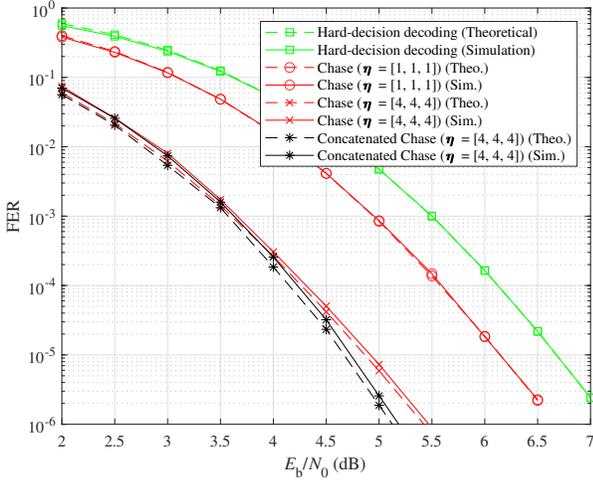


Fig. 3. Error-correction performance of Chase decoding and concatenated Chase decoding for the $([4, 2], 31)$ GII-BCH code.

D. Chase Decoding of GII-BCH Codes

With the theoretical Chase decoding performance characterization for BCH codes in Section III-A, the hard-decision decoding error probability for GII-BCH code can be further defined as [13]

$$P_f = \sum_{b=0}^v P_{f_b}, \quad (22)$$

where P_{f_b} is the error probability of decoding round- b . It is defined as [8]

$$P_{f_b} = \begin{cases} \sum_{i=v+1}^m \binom{m}{i} p_0^i (1 - p_0)^{m-i}, & \text{if } b = 0; \\ \binom{m}{v-b+1} p_b^{v-b+1} (1 - p_{b-1})^{m-(v-b+1)}, & \text{if } 1 \leq b \leq v, \end{cases} \quad (23)$$

where p_b is the error probability of an interleave in decoding round- b . Given p as the decoder input bit error rate, the probability of an interleave having w errors is $\phi_w = \binom{n}{w} p^w (1 - p)^{n-w}$. For hard-decision decoding, $p_b = \sum_{w=t_b+1}^n \phi_w$. For Chase decoding, p_b can be determined by (14). It can then be substituted into (23) and (22) successively in yielding the theoretical Chase decoding error probability for the GII-BCH codes. Therefore,

$$p_b = p_{\text{Chase}}(t_b, \eta_b), \quad (24)$$

where $0 \leq b \leq v$. Furthermore, P_{f_b} of (23) can be recharacterized as in (25).

E. Concatenated Chase Decoding of GII-BCH Codes

Similarly, determining p_b by (19) or (21) can yield the concatenated Chase decoding theoretical error probability of the GII-BCH codes. Assume that a perfect sphere-packing code $\mathcal{C}_c(\eta, \tau)$ is employed in decoding rounds b_1^*, b_2^*, \dots ,

where we denote $\mathbf{B}^* = \{b_1^*, b_2^*, \dots\}$. Note that coded Chase decoding can be considered as a special case of concatenated Chase decoding as when $\tilde{\eta} = 0$. Therefore,

$$p_{\text{Coded}}(t, \eta) = p_{\text{Conc}}(t, \eta, 0). \quad (26)$$

For decoding round-0, its error probability only depends on the decoding error probability of the interleaves. Whether concatenated Chase decoding is applied in decoding round-0 determines the calculation method for P_{f_0} . That says

$$P_{f_0} = \begin{cases} \sum_{i=v+1}^m \binom{m}{i} p_{\text{Chase}}(t_0, \eta_0)^i (1 - p_{\text{Chase}}(t_0, \eta_0))^{m-i}, & \text{if } 0 \notin \mathbf{B}; \\ \sum_{i=v+1}^m \binom{m}{i} p_{\text{Conc}}(t_0, \eta, \eta_0)^i (1 - p_{\text{Conc}}(t_0, \eta, \eta_0))^{m-i}, & \text{if } 0 \in \mathbf{B}. \end{cases} \quad (27a)$$

When $b \geq 1$, its error probability depends on the decoding error probabilities of interleaves in both the current and the previous rounds. Therefore, P_{f_b} can be derived as (27b).

Fig. 3 compares our simulation results with the theoretical characterizations of Chase decoding and concatenated Chase decoding of the $([4, 2], 31)$ GII-BCH code. Its dimension is $K = 49$ and the error-correction capability is $t = [3, 5, 7]$. Concatenated Chase decoding employs the $(7, 4)$ Hamming code as covering code. Fig. 3 shows that theoretical characterizations of Chase decoding and concatenated Chase decoding for GII-BCH codes also match well with the simulation results.

IV. ENHANCED CHASE DECODING OF GII-BCH CODES

Based on the mentioned performance characterization in Section III, this section further introduces two new Chase decoding methods for GII-BCH codes, including the enhanced Chase decoding (ECD) and the enhanced concatenated Chase decoding (ECCD). Armed with the above analysis, we are able to compute the error probability of each decoding round. This helps better allocate the flipped positions, resulting in significant performance gains.

A. Enhanced Chase Decoding of GII-BCH Codes

For an $([m, v], n)$ GII-BCH code whose hard-decision error-correction capability is $t = [t_0, t_1, \dots, t_v]$, composition of the flipping vector η determines the decoding performance and complexity. Therefore, entries of η can be designed under a certain complexity constraint. Intuitively, given the maximum number of test-vectors, the flipped positions should be allocated to the decoding rounds that are more likely to declare a decoding failure, so that the Chase decoding performance can be enhanced. Let $\mathbf{P}_{f_b} = \{P_{f_0}, P_{f_1}, \dots, P_{f_v}\}$ denote the set of the error probability of all decoding rounds, where its entries can be determined by (25). Further let T denote the maximum number of test-vectors as

$$T = m2^{\eta_0} + v2^{\eta_1} + (v-1)2^{\eta_2} + \dots + 2^{\eta_v}. \quad (28)$$

$$P_{f_b} = \begin{cases} \binom{m}{v-b+1} p_{\text{Chase}}(t_b, \eta_b)^{v-b+1} \cdot (1 - p_{\text{Chase}}(t_{b-1}, \eta_{b-1}))^{m-(v-b+1)}, & \text{if } b \notin \mathbf{B}, b-1 \notin \mathbf{B}; \\ \binom{m}{v-b+1} p_{\text{Chase}}(t_b, \eta_b)^{v-b+1} \cdot (1 - p_{\text{Conc}}(t_{b-1}, \eta, \eta_{b-1}))^{m-(v-b+1)}, & \text{if } b \notin \mathbf{B}, b-1 \in \mathbf{B}; \\ \binom{m}{v-b+1} p_{\text{Conc}}(t_b, \eta, \eta_b)^{v-b+1} \cdot (1 - p_{\text{Chase}}(t_{b-1}, \eta_{b-1}))^{m-(v-b+1)}, & \text{if } b \in \mathbf{B}, b-1 \notin \mathbf{B}; \\ \binom{m}{v-b+1} p_{\text{Conc}}(t_b, \eta, \eta_b)^{v-b+1} \cdot (1 - p_{\text{Conc}}(t_{b-1}, \eta, \eta_{b-1}))^{m-(v-b+1)}, & \text{if } b \in \mathbf{B}, b-1 \in \mathbf{B}. \end{cases} \quad (27b)$$

Let T_{Thr} denote the maximum number of test-vectors, which is designed by considering the decoding complexity budget. Based on (22), it can be seen that the Chase decoding error probability of the GII-BCH codes is the sum of the error probability of each decoding round. Based on both (23) and (25), it can be seen that when $b = 0$, P_{f_b} only depends on p_b . When $1 \leq b \leq v$, P_{f_b} depends on both p_b and p_{b-1} . Based on the calculations in (22), (23) and (25), it can be observed that when the FER of GII-BCH codes approaches 10^{-5} and $m < 10$, $(1 - p_{b-1})^{m-(v-b+1)}$ can be approximated as 1, and P_{f_b} can be approximated as

$$P_{f_b} \approx \binom{m}{v-b+1} p_b^{v-b+1}. \quad (29)$$

Therefore, enhancing the error-correction capability of each interleave in decoding round- b can lead to decoding performance enhancement of the GII-BCH codes.

Let $P_{f_{b_{\max}}}$ denote the maximum value of P_{f_b} , i.e.,

$$P_{f_{b_{\max}}} = \max\{P_{f_{b_i}} | i = 0, 1, \dots, v\}. \quad (30)$$

Let $\eta_{b_{\max}}$ further denote the number of flipped positions in decoding round- b_{\max} . Based on the above definition, it can be seen that among all decoding rounds, round- b_{\max} is most likely to declare a decoding failure. A flipped position should be assigned to it. The maximum number of test-vectors T is then updated based on (28). If $T \leq T_{\text{Thr}}$, P_{f_b} and $P_{f_{b_{\max}}}$ will be determined again as in (25) and (30), respectively. Subsequently, the next flipped position can be assigned and T is updated again. If $T > T_{\text{Thr}}$, this flipped position will be relocated. In this case, the error probability set P_{f_b} is updated as

$$P_{f_b} = P_{f_b} \setminus \{P_{f_{b_{\max}}}\}. \quad (31)$$

Afterwards, $P_{f_{b_{\max}}}$ will be determined from the new set as in (30), identifying the decoding round that yields the second highest error probability. The flipped position is assigned to it and T is updated as in (28). The above process terminates once the error probability set P_{f_b} is empty. The flipping vector η is now formed, and the conventional Chase decoding, i.e., Algorithm 1, can be applied to decode the received words. The above described enhanced Chase decoding (ECD) is summarized as in Algorithm 2. Note that calculation of the flipping vector η , i.e., lines 2 to 13, is performed offline. Therefore, the actual decoding complexity and latency remain the same as the conventional Chase decoding. Decoding complexity of a test-vector using BM algorithm is $\mathcal{O}(n^2)$. Therefore, decoding complexity of the ECD in the worst case can be derived as $\mathcal{O}(T_{\text{Thr}} n^2)$, which is the same as the conventional Chase decoding.

The following Example 3 demonstrates the calculation of η

Algorithm 2: Enhanced Chase Decoding (ECD)

Input: $y_0(x), y_1(x), \dots, y_{m-1}(x), T_{\text{Thr}}$;

Output: $c_0(x), c_1(x), \dots, c_{m-1}(x)$;

1 Initialize $T = m + v + v - 1 + \dots + 1$ and

$\eta = [0, 0, \dots, 0]$;

2 **While** $T \leq T_{\text{Thr}}$ **do**

3 Compute $P_{f_b} = \{P_{f_0}, P_{f_1}, \dots, P_{f_v}\}$ as in (25);

4 Determine $P_{f_{b_{\max}}}$ as in (30);

5 $\eta_{b_{\max}} \leftarrow \eta_{b_{\max}} + 1$;

6 Compute T as in (28);

7 **If** $T > T_{\text{Thr}}$ **then**

8 $\eta_{b_{\max}} \leftarrow \eta_{b_{\max}} - 1$;

9 Compute T as in (28);

10 Update P_{f_b} as in (31);

11 **If** $P_{f_b} = \emptyset$ **then**

12 | break;

13 Return to line 4;

14 Employ the **Algorithm 1** to decode

$y_0(x), y_1(x), \dots, y_{m-1}(x)$ with the newly formed η .

in the ECD.

Example 3. For a $([6, 3], 63)$ GII-BCH code with $N = 378$ and $K = 116$, its hard-decision error-correction capability is $\mathbf{t} = [7, 10, 11, 13]$. Let $T_{\text{Thr}} = 40$ and theoretical decoding error probabilities are computed at the SNR of 5.5 dB, the flipping vector η can be determined through Algorithm 2 with its intermediate calculations elaborated as follows. Let us initialize $\eta = [0, 0, 0, 0]$ and $T = 12$. P_{f_b} can be determined as

$$P_{f_b} = \{P_{f_0} = 3.59 \times 10^{-4}, P_{f_1} = 1.08 \times 10^{-6}, \\ P_{f_2} = 2.41 \times 10^{-5}, P_{f_3} = 5.69 \times 10^{-4}\}.$$

Hence, $b_{\max} = 3$. A flipped position is assigned to decoding round-3, $\eta = [0, 0, 0, 1]$ and $T = 13$. Afterwards, P_{f_b} is updated as

$$P_{f_b} = \{P_{f_0} = 3.59 \times 10^{-4}, P_{f_1} = 1.08 \times 10^{-6}, \\ P_{f_2} = 2.41 \times 10^{-5}, P_{f_3} = 2.29 \times 10^{-4}\}.$$

Hence, $b_{\max} = 0$. A flipped position is assigned to decoding round-0, $\eta = [1, 0, 0, 1]$ and $T = 19$. Repeat the above process until $\eta = [2, 0, 0, 3]$, $T = 37$, and

$$P_{f_b} = \{P_{f_0} = 8.45 \times 10^{-6}, P_{f_1} = 1.25 \times 10^{-6}, \\ P_{f_2} = 2.41 \times 10^{-5}, P_{f_3} = 3.43 \times 10^{-5}\}.$$

Hence, $b_{\max} = 3$. A flipped position is assigned to decoding round-3. After this allocation, $T = 45 > T_{\text{Thr}}$. Therefore, this position should be relocated. Based on the above decoding error probability set P_{f_b} , decoding round-2 has the second

highest error probability. Therefore, let $\eta = [2, 0, 1, 3]$ and $T = 39$. At this point, further allocating a flipped position to any decoding round will cause $T > T_{\text{Thr}}$. Therefore, $\eta = [2, 0, 1, 3]$.

B. Enhanced Concatenated Chase Decoding of GII-BCH Codes

With the same number of test-vectors, concatenated Chase decoding can achieve better performance than the conventional Chase decoding. It can also be employed to decode GII-BCH codes. The application of concatenated Chase decoding can achieve higher performance gains without increasing the number of test-vectors. Concatenated Chase decoding is able to correct more errors of interleaves in the decoding rounds which are more likely to declare a decoding failure. However, it is not necessary to employ concatenated Chase decoding in every decoding round. Concatenated Chase decoding that employs the covering code $\mathcal{C}_c(\eta, \tau)$ can only be used in the decoding rounds that have been allocated at least τ flipped positions. For the rest decoding rounds, the conventional Chase decoding is employed. For a decoding round in which concatenated Chase decoding is employed, let \tilde{P}_{f_b} denote the error probability of decoding round- b . Let λ denote the performance gain yielded by concatenated Chase decoding as

$$\lambda = \frac{P_{f_b} - \tilde{P}_{f_b}}{P_{f_b}}. \quad (32)$$

It represents the extent of the performance improvement over the conventional Chase decoding. In particular, if $\lambda > 0$, concatenated Chase decoding in this decoding round can improve performance, and its improvement effect is reflected by the value. In contrast, if $\lambda < 0$, concatenated Chase decoding does not lead to an improvement. This enables us to decide whether to employ concatenated Chase decoding. In our work, concatenated Chase decoding is employed only when $\lambda > 0.1$. Let κ_b denote the flag that whether to employ concatenated Chase decoding in decoding round- b . If $\kappa_b = 1$, the concatenated can be employed in this round. Let we denote $\kappa = [\kappa_0, \kappa_1, \dots, \kappa_v]$. After allocating a flipped position, each decoding round will evaluate λ in order to determine whether to employ the concatenated Chase decoding. The remaining steps are the same as those of Algorithm 2. Recalling the $1_{[a,b]}(x)$ function that is defined as in (18), in the ECCD, T should be calculated as

$$\begin{aligned} T = & m(2^{\eta_0} 1_{[0,0]}(\kappa_0) + 2^{\eta_0 - \eta + \tau} 1_{[1,1]}(\kappa_0)) \\ & + v(2^{\eta_1} 1_{[0,0]}(\kappa_1) + 2^{\eta_1 - \eta + \tau} 1_{[1,1]}(\kappa_1)) \\ & + \dots + (2^{\eta_v} 1_{[0,0]}(\kappa_v) + 2^{\eta_v - \eta + \tau} 1_{[1,1]}(\kappa_v)). \end{aligned} \quad (33)$$

The above described process is summarized as in Algorithm 3. Similar to the ECD, the calculations of η and κ , i.e., lines 2 to 25, are also performed offline. Overall, decoding complexity of the ECCD in the worst case is $\mathcal{O}(T_{\text{Thr}} n^2)$, which is the same as the ECD and the conventional Chase decoding.

In order to better illustrate its key computations, the following Example 4 demonstrates the calculations of η and κ in the ECCD.

Algorithm 3: Enhanced Concatenated Chase Decoding (ECCD)

Input: $y_0(x), y_1(x), \dots, y_{m-1}(x), T_{\text{Thr}}$ and $\mathcal{C}(\eta, \tau)$;
Output: $c_0(x), c_1(x), \dots, c_{m-1}(x)$;

- 1 Initialize $T = m + v + v - 1 + \dots + 1$,
 $\eta = [0, 0, \dots, 0]$, $\kappa = [0, 0, \dots, 0]$;
- 2 **While** $T \leq T_{\text{Thr}}$ **do**
- 3 Compute $P_{f_b} = \{P_{f_0}, P_{f_1}, \dots, P_{f_v}\}$ as in (25) ;
- 4 Determine $P_{f_{b_{\max}}}$ as in (30) ;
- 5 $\eta_{b_{\max}} \leftarrow \eta_{b_{\max}} + 1$;
- 6 Compute T as in (33) ;
- 7 **If** $T > T_{\text{Thr}}$ **then**
- 8 $\eta_{b_{\max}} \leftarrow \eta_{b_{\max}} - 1$;
- 9 Compute T as in (33) ;
- 10 Update P_{f_b} as in (31) ;
- 11 **If** $P_{f_b} = \emptyset$ **then**
- 12 **break** ;
- 13 Return to line 4 ;
- 14 **For** $b = 0, 1, \dots, v$ **do**
- 15 **If** $\eta_b \geq \tau$ **then**
- 16 employ $\mathcal{C}_c(\eta, \tau)$ in decoding round- b ;
- 17 **If** $\eta_b = \tau$ **then**
- 18 $p_b \leftarrow p_{\text{Coded}}(t_b, \eta)$;
- 19 **Else**
- 20 $p_b \leftarrow p_{\text{Conc}}(t_b, \eta, \eta_b - \eta)$;
- 21 Compute \tilde{P}_{f_b} as in (23) ;
- 22 **If** $\lambda = \frac{P_{f_b} - \tilde{P}_{f_b}}{P_{f_b}} > 0.1$ and $\kappa_b = 0$ **then**
- 23 Employ the concatenated Chase decoding in decoding round- b ;
- 24 $\kappa_b = 1$;
- 25 $\eta_b = \eta_b - \tau + \eta$;
- 26 Employ the **Algorithm 1** to decode $y_0(x), y_1(x), \dots, y_{m-1}(x)$ with the newly formed η and κ .

// For the decoding rounds that $\kappa = 1$, concatenated Chase decoding should be employed.

Example 4. For a $[[6, 3], 127]$ GII-BCH code with $N = 762$ and $K = 391$, its hard-decision error-correction capability is $t = [7, 9, 13, 15]$. Let us initialize $\kappa = [0, 0, 0, 0]$. Let $T_{\text{Thr}} = 115$ and theoretical decoding error probabilities calculated at the SNR of 5 dB, while $\eta = [4, 2, 0, 2]$ can be determined through Algorithm 2 with its intermediate calculations elaborated as follows. Let us determine η through Algorithm 3 with the use of the (7, 4) Hamming code as the covering code. When $\eta = [4, 2, 0, 2]$, only decoding round-0 can employ the concatenated Chase decoding. It can be further determined that $\lambda = 0.14$ as through (32). Concatenated Chase decoding can be employed in the round-0. Hence, $\kappa_0 = 1$. Therefore, $\eta = [7, 2, 0, 2]$ and $\kappa = [1, 0, 0, 0]$.

V. SIMULATION RESULTS

This section presents our simulation results of both the ECD and the ECCD of GII-BCH codes over the AWGN

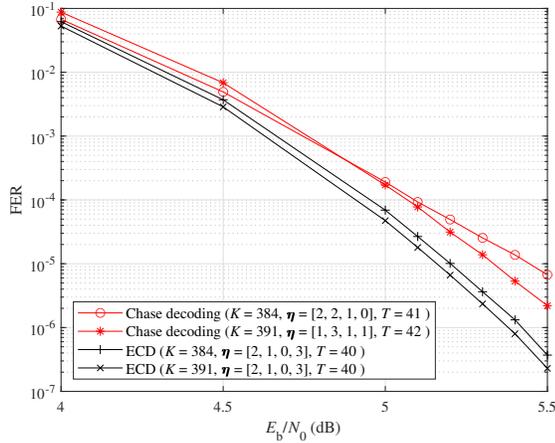


Fig. 4. Comparison between the conventional Chase decoding and the ECD of GII-BCH codes ($m = 6, v = 3, n = 127$).

channel using BPSK modulation. They are compared with the conventional Chase decoding under a similar decoding computational expenditure. In the following discussions, the FER is used to evaluate the decoding performances, which are compared under a similar number of decoding test-vectors. The coding gains are evaluated at the decoding FER of 10^{-5} .

A. Comparison with the Conventional Chase Decoding

Compared with the conventional Chase decoding, the ECD prioritizes allocating flipped positions to the decoding rounds that have an inferior decoding performance.

Fig. 4 compares the conventional Chase decoding and the ECD of GII-BCH codes. For the two GII codes ($[6, 3], 127$), one has a dimension of $K = 391$ and an error-correction capability of $t = [7, 9, 13, 15]$, while the other has a dimension of $K = 384$ and an error-correction capability of $t = [7, 10, 13, 15]$. For both of codes, it is set $T_{\text{Thr}} = 40$ and theoretical decoding error probabilities are calculated at the SNR of 5 dB. We can determine $\eta = [2, 1, 0, 3]$ through the ECD, i.e., Algorithm 2. For the conventional Chase decoding, let $\eta = [2, 2, 1, 0]$ in the case of $K = 391$, and $\eta = [1, 3, 1, 1]$ in the case of $K = 384$. This ensures the maximum number of test-vectors is close to T_{Thr} and $t_b + \eta_b < t_{b+1}$.

Fig. 4 shows that with a similar maximum number of test-vectors, the ECD can achieve a better performance than the conventional Chase decoding. In the case of $K = 384$, the ECD yields a coding gain of 0.25 dB over the conventional Chase decoding. In the case of $K = 391$, it yields a coding gain of 0.3 dB. This is due to the ECD can better allocate the flipped positions for different rounds.

B. Comparison between the ECD and the ECCD

Fig. 5 compares the decoding performance of the ECD and the ECCD for $([4, 2], 63)$ GII-BCH code with dimension $K = 138$ and error-correction capability $t = [3, 6, 10]$. For the ECD, theoretical decoding error probabilities are calculated at the SNR of 4.5 dB. With $T_{\text{Thr}} = 40$, Algorithm 2 yields $\eta = [3, 1, 2]$ and $T = 40$. With $T_{\text{Thr}} = 100$, Algorithm

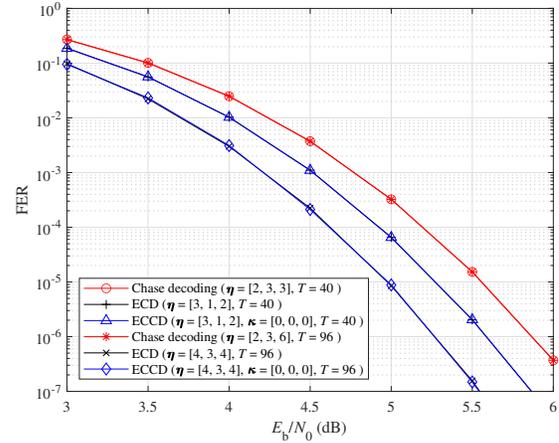


Fig. 5. Comparison between the ECD and the ECCD of the GII-BCH codes ($m = 4, v = 2, n = 63$).

2 yields $\eta = [4, 3, 4]$ and $T = 96$. The ECCD employs the $(7, 4)$ Hamming code as the covering code. For the ECCD, theoretical decoding error probabilities are calculated at the SNR of 4.5 dB. With $T_{\text{Thr}} = 40$, Algorithm 3 yields $\eta = [3, 1, 2]$, $\kappa = [0, 0, 0]$ and $T = 40$. With $T_{\text{Thr}} = 100$, Algorithm 3 yields $\eta = [4, 3, 4]$, $\kappa = [0, 0, 0]$ and $T = 96$. For the conventional Chase decoding, let $\eta = [2, 3, 3]$ in the case of $T_{\text{Thr}} = 40$ and $\eta = [2, 3, 6]$ in the case of $T_{\text{Thr}} = 96$.

In these cases, $\lambda = \frac{P_{fb} - \hat{P}_{fb}}{P_{fb}} > 0.1$ are not satisfied in any decoding round. As a result, concatenated Chase decoding is not applied in the ECCD, and the flipping vector η in the ECCD is the same as in the ECD. Consequently, the ECD and the ECCD exhibit the same decoding performance. In general, when code length is short, performance gain of concatenate Chase decoding is not significant. In case of $T_{\text{Thr}} = 40$, the ECD and the ECCD yield a coding gain of 0.28 dB over the conventional Chase decoding. In case of $T_{\text{Thr}} = 100$, the ECD and the ECCD yield a coding gain of 0.54 dB over the conventional Chase decoding.

The ECCD can further enhance the error-correction performance over the ECD for GII-BCH codes in some cases. Fig. 6 compares the ECD and the ECCD decoding performance for the $([6, 3], 127)$ GII code with dimension $K = 391$ and error-correction capability $t = [7, 9, 13, 15]$. For the ECD, theoretical decoding error probabilities are calculated at the SNR of 4.5 dB. With $T_{\text{Thr}} = 500$, Algorithm 2 yields $\eta = [6, 4, 1, 6]$ and $T = 500$. With $T_{\text{Thr}} = 1000$, Algorithm 2 yields $\eta = [7, 5, 2, 7]$ and $T = 1000$. The ECCD employs the $(7, 4)$ Hamming code as the covering code. For the ECCD, theoretical decoding error probabilities are calculated at the SNR of 4.5 dB. With $T_{\text{Thr}} = 500$, Algorithm 3 yields $\eta = [8, 8, 8, 10]$, $\kappa = [1, 1, 1, 1]$ and $T = 480$. With $T_{\text{Thr}} = 1000$, Algorithm 3 yields $\eta = [9, 9, 9, 11]$, $\kappa = [1, 1, 1, 1]$ and $T = 960$. For the conventional Chase decoding, let $\eta = [1, 3, 1, 9]$ in the case of $T_{\text{Thr}} = 552$, $\eta = [1, 3, 1, 10]$ in the case of $T_{\text{Thr}} = 1064$. For the conventional Chase decoding, as the number of test-vectors increases, the decoding performance does not improve. In this case, the error probability of decoding round-3 is negligible. Therefore, continuing to allocate flipped positions to decoding

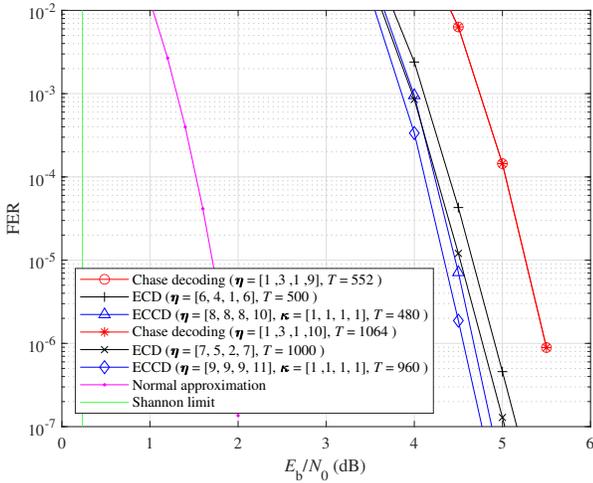


Fig. 6. Comparison between the ECD and the ECCD of the GII-BCH codes ($m = 6, v = 3, n = 127$).

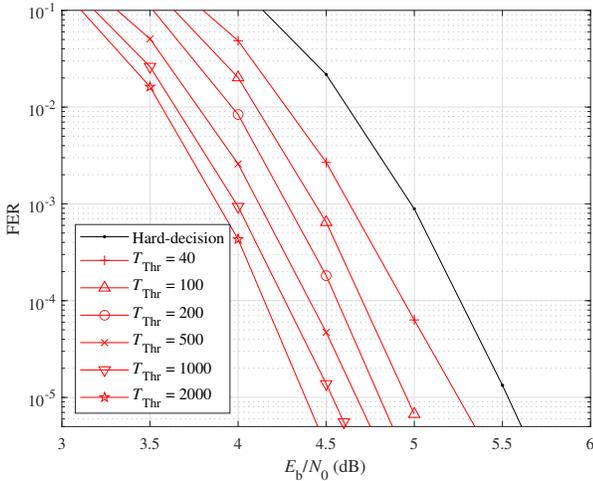


Fig. 7. Decoding performance with different T_{Thr} .

round-3 cannot yield a coding gain. In contrast, both the ECD and the ECCD can improve the error-correction performance without extra complexity cost. The ECCD outperforms the ECD. In the case of $T_{\text{Thr}} = 500$, the ECD yields a coding gain of 0.59 dB over the conventional Chase decoding. The ECCD further yields a coding gain of 0.19 dB over the ECD. A similar improvement can also be observed in the case of $T_{\text{Thr}} = 1000$. However, this is only a code of length 762 bits, incapable of achieving capacity. The soft-decision decoding performances of the codes still fall apart from the theoretical limits.

C. Decoding Performance with Different T_{Thr}

Both the ECD and ECCD performance can be improved by increasing the maximum number of test-vectors, as controlled by T_{Thr} . Fig. 7 shows the decoding performance of different T_{Thr} for ECD of the $([6, 3], 127)$ GII-BCH code, where $K = 391$ and $t = [7, 9, 13, 15]$. It can be observed that as T_{Thr} increases, substantial decoding performance gains can be

achieved. However, this is also at the cost of the decoding complexity that increases proportionally to T_{Thr} .

VI. CONCLUSION

This paper proposes a new theoretical decoding performance characterization for Chase decoding of BCH codes and further extends it for concatenated Chase decoding of the codes. Our theoretical characterizations can match well with the simulation results. Based on this, the theoretical decoding performance characterization of concatenated Chase decoding of GII-BCH codes has been presented. This paper proposes the ECD and the ECCD of GII-BCH codes. They are developed based on our theoretical characterization of Chase decoding performances of the codes. Both the ECD and ECCD can identify the decoding rounds that are more likely to declare a decoding failure and prioritize allocating the flipped positions to those rounds. Our simulation results have shown that, for the GII-BCH codes, both the ECD and the ECCD outperform the conventional Chase decoding with a similar maximum number of test-vectors.

ACKNOWLEDGMENT

This work is supported in part by the National Natural Science Foundation of China (NSFC) with project ID 62471503; and in part by the Natural Science Foundation of Guangdong Province with project ID 2024A1515010213.

REFERENCES

- [1] Y. Wu, "Generalized integrated interleaved codes," *IEEE Trans. Inf. Theory*, vol. 63, no. 2, pp. 1102–1119, Feb. 2017.
- [2] X. Tang and R. Koetter, "A novel method for combining algebraic decoding and iterative processing," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Seattle, USA, Jul. 2006, pp. 474–478.
- [3] Z. Xie and X. Zhang, "Fast nested key equation solvers for generalized integrated interleaved decoder," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 68, no. 1, pp. 483–495, Jan. 2021.
- [4] W. Li, J. Tian, J. Lin and *et al.*, "Modified GII-BCH codes for low-complexity and low-latency encoders," *IEEE Commun. Lett.*, vol. 23, no. 5, pp. 785–788, May 2019.
- [5] Y. Wu, "High-speed LFSR decoder architectures for BCH and GII codes," *IEEE J. Sel. Areas Inf. Theory*, vol. 4, pp. 331–350, Aug. 2023.
- [6] Z. Xie and X. Zhang, "Reduced-complexity key equation solvers for generalized integrated interleaved BCH decoders," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 67, no. 12, pp. 5520–5529, Dec. 2020.
- [7] X. Zhang and Z. Xie, "Efficient architectures for generalized integrated interleaved decoder," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 66, no. 10, pp. 4018–4031, Oct. 2019.
- [8] H. Cui, S. Song, and Z. Wang, "An improved method for performance analysis of generalized integrated interleaved codes," *IEEE Commun. Lett.*, vol. 25, no. 10, pp. 3166–3169, Oct. 2021.
- [9] G. Forney, "Generalized minimum distance decoding," *IEEE Trans. Inf. Theory*, vol. 12, no. 2, pp. 125–131, Apr. 1966.
- [10] D. Chase, "Class of algorithms for decoding block codes with channel measurement information," *IEEE Trans. Inf. Theory*, vol. 18, no. 1, pp. 170–182, Jan. 1972.
- [11] M. Fossorier and S. Lin, "Error performance analysis for reliability-based decoding algorithms," *IEEE Trans. Inf. Theory*, vol. 48, no. 1, pp. 287–293, Jan. 2002.
- [12] D. Agrawal and A. Vardy, "Generalized minimum distance decoding in Euclidean space: performance analysis," *IEEE Trans. Inf. Theory*, vol. 46, no. 1, pp. 60–83, Jan. 2000.
- [13] Y. J. Tang and X. Zhang, "Fast and low-complexity soft-decision generalized integrated interleaved decoder," *IEEE J. Sel. Areas Inf. Theory*, vol. 4, pp. 174–189, Jul. 2023.

- [14] H. Tokushige, T. Koumoto, M. Fossorier and *et al.*, "Selection method of test patterns in soft-decision iterative bounded distance decoding algorithms," *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, vol. 86, no. 10, pp. 2445–2451, Oct. 2003.
- [15] P. S. Nguyen, H. D. Pfister, and K. R. Narayanan, "On multiple decoding attempts for Reed–Solomon codes: A rate-distortion approach," *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 668–691, Feb. 2011.
- [16] H. Mani and S. Hemati, "Symbol-level stochastic Chase decoding of Reed–Solomon and BCH codes," *IEEE Trans. Commun.*, vol. 67, no. 8, pp. 5241–5252, Aug. 2019.
- [17] Y. Wu and J. Ma, "A novel Chase Kötter–Vardy algorithm," *IEEE J. Sel. Areas Inf. Theory*, vol. 4, pp. 232–244, Jul. 2023.
- [18] J. Massey, "Shift-register synthesis and BCH decoding," *IEEE Trans. Inf. Theory*, vol. 15, no. 1, pp. 122–127, Jan. 1969.
- [19] C. Yue, M. Shirvanimoghaddam, B. Vucetic and *et al.*, "A revisit to ordered statistics decoding: Distance distribution and decoding rules," *IEEE Trans. Inf. Theory*, vol. 67, no. 7, pp. 4288–4337, Jul. 2021.