Shift-Sum Decoding of Non-Binary Cyclic Codes

Jiongyue Xing, Martin Bossert, *Fellow, IEEE*, Li Chen, *Senior Member, IEEE*, Jiasheng Yuan, and Sebastian Bitzer, *Graduate Student Member, IEEE*

***Abstract*— This paper proposes a novel shift-sum decoding method for non-binary cyclic codes, which only requires finite field operations but yields advanced decoding performance. Using the cyclically different minimum-weight dual codewords (MWDCs) and their proper shifts, a frequency matrix can be obtained as a reliability metric for identifying the error positions and magnitudes. By analyzing the statistical distributions of the matrix entries, the rationale for the shift-sum decoding's advanced error-correction capability is revealed. Based on this decoding method, a hard-decision iterative shift-sum (HISS) decoding algorithm is first proposed. It can correct errors beyond half of the code's minimum Hamming distance. By further utilizing the reliability information obtained from the channel, a soft-decision iterative shift-sum (SISS) decoding algorithm is then proposed to improve the decoding performance. Both the HISS and the SISS algorithms are realized only with polynomial multiplications and numerical comparisons, which are hardware-friendly. To further improve the error-correction performance, the HISS and SISS algorithms can be integrated in a Chase decoding mechanism for handling the test-vectors. Simulation results on Reed-Solomon (RS) and non-binary BCH (NB-BCH) codes show that the proposed algorithms yield a competent decoding and complexity performances in comparison with the existing decoding algorithms.**

***Index Terms*— Iterative decoding, minimum-weight dual codewords, non-binary cyclic codes, shift-sum decoding.**

I. Introduction

CYCLIC codes, including Reed-Solomon (RS) codes and Bose-Chaudhuri-Hocquenghem (BCH) codes, are widely used for data transmission due to their algebraic structure which results in efficient encoding and decoding algorithms [3]. In particular, their encoding can be implemented by a linear shift register circuit. For practical systems, syndrome based decoding is applied, including the Berlekamp-Massey (BM) algorithm [4], [5] and the extended Euclidean algorithm [6]. These can correct errors up to half of the code's minimum Hamming distance. Interpolation based algebraic list decoding, or the so-called Guruswami-Sudan (GS) algorithm [7], can correct errors beyond this limit with a polynomial-time complexity. The decoding performance can be further enhanced by the algebraic soft-decision (ASD) decoding, or the so-called Kötter-Vardy (KV) algorithm [8]. By utilizing the BM decoding output, Wu further proposed an improved list decoding algorithm for both RS and BCH codes [9], which exhibits a lower complexity than the GS algorithm. However, in comparison to syndrome based decoding, the complexity of interpolation based decoding remains high, limiting its practical applications. Utilizing the soft information obtained from the channel, Chase decoding [10], information set decoding [11] and ordered statistics decoding (OSD) [12] generate multiple decoding trials, yielding an enhanced decoding performance but with a complexity that is exponential in nature. Furthermore, integrating the KV algorithm into the Chase decoding of RS codes was proposed in [13] and [14] to reduce the decoding complexity. Recently, a low-complexity Chase decoding utilizing the basis reduction interpolation was introduced to reduce the decoding latency [15]. In [16], a syndrome based fast Chase decoding was proposed to reduce the decoding complexity from the Gröbner basis perspective. Based on the statistical distribution of the distance from codeword estimates to the received information, several stopping and discarding rules were presented to facilitate the OSD algorithm [17].

Belief propagation (BP) is an efficient decoder with good performance for low-density parity-check (LDPC) codes [18]. However, its error-correction ability falls short when decoding cyclic codes since the parity-check matrix contains too many short cycles. To alleviate the impact of short cycles, the adaptive BP (ABP) algorithm [19], [20], [21] first performs Gaussian elimination (GE) on the binary parity-check matrix, eliminating some of the short cycles. This limits the propagation of unreliable information during the BP iterations. The ABP algorithm can help improve the reliability of the received information. Algebraic decoding algorithms, e.g., the BM or the KV algorithms, utilize this improved information to produce an enhanced decoding performance. Recently, the perturbation with scheduling [22] and the concatenation with single parity-check codes [23] were proposed to further improve the performance of the above mentioned ABP algorithms. However, the GE process cannot be conducted

Manuscript received 5 November 2022; revised 11 June 2023; accepted 20 September 2023. Date of publication 2 October 2023; date of current version 22 January 2024. This work was supported by the National Natural Science Foundation of China (NSFC) under Project ID 62071498. An earlier version of this paper was presented in part at the 2020 IEEE International Symposium on Information Theory [DOI: 10.1109/ISIT44484.2020.9174258] and in part at the 2021 IEEE International Symposium on Information Theory [DOI: 10.1109/ISIT45174.2021.9518200]. *(Corresponding author: Li Chen.)*

Jiongyue Xing and Li Chen are with the School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou 510006, China (e-mail: xingjyue@mail2.sysu.edu.cn; chenli55@mail.sysu.edu.cn).

Martin Bossert is with the Institute of Communications Engineering, Ulm University, 89081 Ulm, Germany (e-mail: martin.bossert@uni-ulm.de).

Jiasheng Yuan was with the School of Electronics and Communication Engineering, Sun Yat-sen University, Guangzhou 510006, China. He is now with Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China, Shenzhen 518110, China (e-mail: yuanjsh@mail2.sysu.edu.cn).

Sebastian Bitzer is with the Institute for Communications Engineering, Technical University of Munich, 80333 Munich, Germany (e-mail: sebastian.bitzer@tum.de).

Communicated by I. Tal, Associate Editor for Coding and Decoding.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TIT.2023.3321173.

Digital Object Identifier 10.1109/TIT.2023.3321173

in parallel, resulting in a high decoding latency. In order to overcome this challenge, Halford and Chugg [24] proposed a random redundant iterative decoding by utilizing a redundant parity-check matrix and its permutation groups, while Hehn et al. [25] introduced the multiple-bases BP (MBBP) algorithm which utilizes several parity-check matrices for parallel BP decoding. Integrating the above two approaches, Dimnik and Be'ery [26] further presented an improved random redundant iterative decoding, exhibiting a near maximum-likelihood (ML) decoding performance for short cyclic codes.

In [27], minimum-weight dual codewords (MWDCs) are utilized for decoding linear codes to correct errors beyond half of the minimum Hamming distance bound. This idea has also been applied to decode Reed-Muller (RM) codes, showing that near-ML decoding performance can be achieved [28]. In [29], a novel concept of shift-sum decoding for binary cyclic codes was proposed. It utilizes a number of cyclically different MWDCs and their proper shifts to generate a reliability measure which can be considered as the fundamental metric for various decoding algorithms. The shift-sum decoding process only requires polynomial multiplications and numerical operations, which is of practical interest. It also allows the information set decoding of BCH codes to achieve the ML decoding performance [30]. Recently, the cyclic property was also explored in [31] and [32] to construct a neural list decoder for BCH and punctured RM codes.

In this paper, we generalize the idea of [29] to the non-binary case, which is also motivated by its application to channels suffering from burst errors. The main contributions of this work are summarized as follows:

1) The shift-sum operation for non-binary cyclic codes is first proposed. It utilizes a number of cyclically different MWDCs and their cyclic shifts. Each of them can produce a syndrome polynomial whose coefficients indicate the positions of errors (up to a cyclic shift) and their magnitudes (up to a multiplication by scalar). By counting the number of different coefficients at each position, a frequency matrix is formulated to identify the erroneous positions and their magnitudes.
2) The plausibility analysis of the shift-sum operation is presented. It looks into the statistical distribution of the frequency matrix's entries. They are categorized into four cases with characterizations of the probability and expectation of their occurrence. It reveals the rationale for the shift-sum decoding's advanced error-correction capability. Its application to binary cyclic codes improves the recent results of expectation characterization in [30] and matches with the numerical results.
3) Based on the above analysis, a hard-decision iterative shift-sum (HISS) algorithm is proposed. Further utilizing the received soft information, a soft-decision iterative shift-sum (SISS) algorithm is also introduced to improve the decoding performance. The two decoding algorithms only require polynomial multiplications, additions and comparisons, presenting a hardware-friendly operation nature. The HISS and SISS algorithms are further integrated into the Chase decoding to yield an improved error-correction performance.
4) Simulation results for classical non-binary cyclic codes, including RS and non-binary BCH (NB-BCH) codes, show that the HISS and the SISS algorithms perform better than the bounded minimum-distance decoding [4], [5], and Chase decoding substantiated by the HISS and the SISS algorithms can significantly outperform the ASD algorithm [8]. It is also shown that for RS codes, the HISS algorithm achieves the same decoding performance as the GS algorithm [7], demonstrating its capability in correcting errors beyond half of the code's minimum Hamming distance. Complexity analysis shows the advantage of the proposed algorithms over two interpolation based decoding algorithms, namely, Kötter's interpolation [33] and the basis reduction interpolation [34]. It is also worthwhile to mention that so far, decoding of NB-BCH codes has been sparsely reported in literature. This work also provides some new performance insights for the codes.

The rest of this paper is organized as follows. Section II provides some preliminaries for non-binary cyclic codes. Section III describes non-binary shift-sum decoding. Section IV presents a plausibility analysis of the shift-sum decoding. Section V introduces the proposed HISS and SISS algorithms, together with their Chase-decoding based variants. Comprehensive simulation results and complexity analysis are presented in Section VI, followed by our conclusions in Section VII.

II. Background Knowledge

This section presents the definition of cyclic codes and its encoding process. Let $\mathbb{F}_q = \{\sigma_0, \sigma_1, \ldots, \sigma_{q-1}\}$ denote a finite field of size q with a primitive element α, where σ_0 designates the zero element. Let $\mathbb{F}_q[x]$ denote the univariate polynomial ring defined over $\mathbb{F}_q$. For simplicity, we only consider codes defined over finite fields of characteristic two with length $n = 2^s - 1$, where $s \in \mathbb{Z}^+$. Let $\mathcal{C}(2^p; n, k, d)$ denote a cyclic code defined over $\mathbb{F}_{2^p}$ with length n, dimension k and the minimum Hamming distance d, where $p = 1, 2, \ldots, s$. Note that when $p = 1$, $\mathcal{C}$ is a binary BCH code; when $p = s$, $\mathcal{C}$ is an RS code; otherwise, $\mathcal{C}$ is an NB-BCH code. Its dual code is denoted as $\mathcal{C}^{\perp}(2^p; n, n-k, d^{\perp})$. Let

$$\underline{c} = (c_0, c_1, \ldots, c_{n-1}) \in \mathcal{C}(2^p; n, k, d) \tag{1}$$

denote a codeword, which can also be written as a polynomial

$$c(x) = c_0 + c_1 x + \cdots + c_{n-1} x^{n-1}, \tag{2}$$

where $c_j \in \mathbb{F}_{2^p}, \forall j$. In the following, both representations will be interchangeably used to denote the same codeword. Note that for any codeword $c(x) \in \mathcal{C}(2^p; n, k, d)$ and any dual codeword $c^{\perp}(x) \in \mathcal{C}^{\perp}(2^p; n, n-k, d^{\perp})$, we have

$$c(x)c^{\perp}(x) = 0 \mod (x^n - 1). \tag{3}$$

The support of $\underline{c}$ (or $c(x)$) is defined as

$$\sup(\underline{c}) = \sup(c(x)) = \{j \mid c_j \neq 0, \forall j\}. \tag{4}$$

The weight of $\underline{c}$ (or $c(x)$) is

$$\mathrm{wt}(\underline{c}) = \mathrm{wt}(c(x)) = |\sup(c(x))|. \tag{5}$$

Given a dual codeword $c^{\perp}(x) \in \mathcal{C}^{\perp}(2^p; n, n-k, d^{\perp})$, it is called a minimum-weight dual codeword (MWDC) if $\mathrm{wt}(c^{\perp}(x))) = d^{\perp}$.

Definition 1: For any two codewords $c_1(x)$ and $c_2(x)$ of $\mathcal{C}(2^p; n, k, d)$, they are cyclically different if $c_2(x) \neq \alpha^j c_1(x) x^{-h} \mod (x^n - 1), \forall j, h \in \mathbb{Z}^2$. Otherwise, they are cyclically equivalent.

Encoding of cyclic codes can be realized by its generator polynomial $g(x)$, where $g(x) \in \mathbb{F}_{2^p}[x]$. Given a message polynomial

$$f(x) = f_0 + f_1 x + \cdots + f_{k-1} x^{k-1} \tag{6}$$

and $f(x) \in \mathbb{F}_{2^p}[x]$, the corresponding codeword polynomial $c(x)$ is generated by

$$c(x) = f(x) g(x). \tag{7}$$

In this paper, two classical non-binary cyclic codes are considered, namely, RS codes and NB-BCH codes. Their generator polynomials are described as follows.

For an RS code $\mathcal{C}(2^s; n, k, d_{\text{RS}})$, where $d_{\text{RS}} = n - k + 1$, its generator polynomial $g_{\text{RS}}(x)$ is defined as

$$g_{\text{RS}}(x) = \prod_{j=1}^{d_{\text{RS}}-1} (x - \alpha^j). \tag{8}$$

Its dual code is also an RS code $\mathcal{C}^{\perp}(2^s; n, n-k, d_{\text{RS}}^{\perp})$, where $d_{\text{RS}}^{\perp} = k + 1$.

NB-BCH codes can be regarded as the sub-field sub-codes of RS codes [3]. Let the cyclotomic cosets be $K_j = \{j \cdot (2^p)^i \mod n, i = 0, 1, \ldots, \frac{s}{p} - 1\}$. It can be seen that for $j_1, j_2 \in \{0, 1, \ldots, n-1\}$, either $K_{j_1} = K_{j_2}$ or $K_{j_1} \cap K_{j_2} = \emptyset$. The cardinality of K_j satisfies $|K_j| \leq \frac{s}{p}$ and $|K_0| = 1$. The generator polynomial $g_{\text{NB-BCH}}(x)$ is defined by

$$g_{\text{NB-BCH}}(x) = \prod_{i \in \mathcal{K}} (x - \alpha^i), \tag{9}$$

where $\mathcal{K}$ is a union set of several distinct cosets K_j and $g_{\text{NB-BCH}}(x) \in \mathbb{F}_{2^p}[x]$. The NB-BCH code has length $n = 2^s - 1$, dimension $k = n - \deg g_{\text{NB-BCH}}(x)$ and its designed minimum distance is $d_{\text{NB-BCH}}$ if $g_{\text{NB-BCH}}(x)$ has $d_{\text{NB-BCH}} - 1$ consecutive roots over $\mathbb{F}_{2^s}$. With different choices of $\mathcal{K}$, we can construct different NB-BCH codes with different parameters and properties.

III. The Shift-Sum Operation

This section proposes the shift-sum operation for decoding non-binary cyclic codes. It utilizes a number of cyclically different MWDCs to create a frequency matrix for determining the error positions and their magnitudes.

Let $\mathcal{E} = \{e_1, e_2, \ldots, e_\tau\}$ denote a set of τ error positions and its complementary set is $\mathcal{E}^{\text{c}} = \{0, 1, \ldots, n-1\} \backslash \mathcal{E}$. Let ε_{e_i} further denote the error magnitude at position e_i, where $\varepsilon_{e_i} \in \mathbb{F}_{2^p} \backslash \{0\}$ and $i = 1, 2, \ldots, \tau$. The error polynomial can be written as

$$\varepsilon(x) = \varepsilon_{e_1} x^{e_1} + \varepsilon_{e_2} x^{e_2} + \cdots + \varepsilon_{e_\tau} x^{e_\tau}. \tag{10}$$

Therefore, as a result of transmitting a codeword $c(x) \in \mathcal{C}$ over a channel, we receive at its output

$$\begin{aligned} r(x) &= c(x) + \varepsilon(x) \\ &= r_0 + r_1 x + \cdots + r_{n-1} x^{n-1}. \end{aligned} \tag{11}$$

Let

$$\beta(x) = \beta_{b_1} x^{b_1} + \beta_{b_2} x^{b_2} + \cdots + \beta_{b_{d^{\perp}}} x^{b_{d^{\perp}}} \tag{12}$$

denote a codeword polynomial of dual code $\mathcal{C}^{\perp}(2^p; n, n-k, d^{\perp})$. Based on Section II, it is an MWDC. The support of this polynomial is $\sup(\beta(x)) = \{b_1, b_2, \ldots, b_{d^{\perp}}\}$. Since the dual code is also linear and cyclic, we can assume w.l.o.g. that $\beta_{b_1} = 1$ and $b_1 = 0$, i.e.,

$$\beta(x) = 1 + \beta_{b_2} x^{b_2} + \cdots + \beta_{b_{d^{\perp}}} x^{b_{d^{\perp}}}. \tag{13}$$

Since

$$c(x)\beta(x) = 0 \mod (x^n - 1), \tag{14}$$

the syndrome polynomial $w(x)$ that is associated with the received word polynomial $r(x)$ is defined as

$$\begin{aligned} w(x) &= r(x)\beta(x) \\ &= (c(x) + \varepsilon(x))\beta(x) \\ &= \varepsilon(x)\beta(x) \mod (x^n - 1). \end{aligned} \tag{15}$$

The polynomial $w(x)$ can also be written as in (16), shown at the bottom of the next page, where the exponents are calculated $\mod n$. It can be seen that $w(x)$ breaks down into $d^{\perp}$ cyclically equivalent error polynomials. Any non-zero coefficient of $w(x)$ is an error at its original position or a scalar error at its shifted position. Multiplying $w(x)$ by $\frac{x^{-h}}{\beta_h}$, we can restore the h-th such polynomial to its original form $\varepsilon(x)$, where $h \in \{b_1, b_2, b_3, \ldots, b_{d^{\perp}}\}$ denotes the shift. As a result, $d^{\perp}$ syndrome polynomials

$$w_h(x) = \frac{x^{-h}}{\beta_h} w(x) \tag{17}$$

can be obtained. Note that $w_0(x) = w(x)$ and $h \in \sup(\beta(x))$.

Assume there are L cyclically different MWDCs in $\mathcal{C}^{\perp}$, which are written as

$$\beta^{(\ell)}(x) = 1 + \beta_{b_2}^{(\ell)} x^{b_2} + \cdots + \beta_{b_{d^{\perp}}}^{(\ell)} x^{b_{d^{\perp}}}, \tag{18}$$

where $\ell = 1, 2, \ldots, L$. Similar to (15), each of them can produce a syndrome polynomial $w^{(\ell)}(x)$ as

$$\begin{aligned} w^{(\ell)}(x) &= r(x)\beta^{(\ell)}(x) \\ &= \varepsilon(x)\beta^{(\ell)}(x) \mod (x^n - 1). \end{aligned} \tag{19}$$

With $d^{\perp}$ cyclic shifts of all $\beta^{(\ell)}(x)$, $Ld^{\perp}$ syndrome polynomials can be generated by

$$\begin{aligned} w_h^{(\ell)}(x) &= \frac{x^{-h}}{\beta_h^{(\ell)}} r(x)\beta^{(\ell)}(x) \mod (x^n - 1) \\ &= w_{h,0}^{(\ell)} + w_{h,1}^{(\ell)} x + \cdots + w_{h,n-1}^{(\ell)} x^{n-1}, \end{aligned} \tag{20}$$

where $h \in \sup(\beta^{(\ell)}(x))$. Note that for each $\ell \in \{1, 2, \ldots, L\}$, $h \in \sup(\beta^{(\ell)}(x))$ and $j \in \{0, 1, \ldots, n-1\}$, the coefficient $w_{h,j}^{(\ell)}$ is determined by

$$w_{h,j}^{(\ell)} = \frac{1}{\beta_h^{(\ell)}} \sum_{u \in \sup(\beta^{(\ell)}(x))} \beta_u^{(\ell)} r_{(j+h-u) \mod n}. \tag{21}$$

Based on (16), it is realized that the value of $w_{h,j}^{(\ell)}$ can be regarded as an indicator for the error position and its magnitude. In order to characterize the value of $w_{h,j}^{(\ell)}$, we further introduce function $T(\ell, i, j, h)$ as

$$T(\ell, i, j, h) = \begin{cases} 1, & \text{if } w_{h,j}^{(\ell)} = \sigma_i, \\ 0, & \text{otherwise}, \end{cases} \tag{22}$$

where $i = 0, 1, \ldots, 2^p - 1$. The main idea is to be based on $w_{h,j}^{(\ell)}$ and on the frequency count of each element σ_i at position j. This frequency is denoted by

$$\phi_{i,j} = \sum_{\ell=1}^{L} \sum_{h \in \sup(\beta^{(\ell)}(x))} T(\ell, i, j, h), \tag{23}$$

which is a statistical measure over all $Ld^{\perp}$ syndrome polynomials. Note that the summation is performed over the integer domain. Since cyclic shifts and summations of $T(\ell, i, j, h)$ play such a central part in the proposed decoding method, it is named the shift-sum operation. Let $\mathbf{\Phi}$ further denote the frequency matrix with entry $\phi_{i,j}$ as

$$\mathbf{\Phi} = \begin{bmatrix} \phi_{0,0} & \phi_{0,1} & \cdots & \phi_{0,n-1} \\ \phi_{1,0} & \phi_{1,1} & \cdots & \phi_{1,n-1} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{2^p-1,0} & \phi_{2^p-1,1} & \cdots & \phi_{2^p-1,n-1} \end{bmatrix}. \tag{24}$$

Note that for each column $j \in \{0, 1, \ldots, n-1\}$ of $\mathbf{\Phi}$,

$$\sum_{i=0}^{2^p-1} \phi_{i,j} = Ld^{\perp}. \tag{25}$$

Entry $\phi_{i,j}$ can be categorized into two classes, i.e., $\{\phi_{0,j}, \forall j\}$ and $\{\phi_{i,j}, \forall j | i \neq 0\}$. On one hand, $\phi_{0,j}$ can be regarded as an indicator to determine whether r_j is erroneous. A smaller value of $\phi_{0,j}$ means that r_j is more likely to be erroneous. Alternatively, due to (25), a larger value of $\phi_{i,j}$ $(i \neq 0)$ also indicates r_j is erroneous with an error magnitude of σ_i. Therefore, $\phi_{i,j}$ can be considered as a reliability metric for identifying the error positions and magnitudes. This is an important observation that will be used in the iterative decoding introduced in Section V. The following example demonstrates the property of $\phi_{i,j}$.

Example 1: Given an RS code $\mathcal{C}(8; 7, 3, 5)$,[1] assume codeword polynomial $c(x) = \alpha^6 + \alpha^4 x + \alpha^4 x^2 + \alpha^3 x^3 + \alpha^6 x^5 + \alpha^3 x^6$ is transmitted and the received polynomial is $r(x) = \alpha^6 + \alpha^5 x + \alpha^4 x^2 + \alpha^3 x^3 + \alpha^3 x^6$. There are five cyclically different MWDCs in $\mathcal{C}^{\perp}$, whose polynomial expressions are

$$\begin{aligned} \beta^{(1)}(x) &= 1 + \alpha x + \alpha^5 x^2 + \alpha^2 x^6, \\ \beta^{(2)}(x) &= 1 + x + x^3 + x^6, \\ \beta^{(3)}(x) &= 1 + \alpha^2 x^2 + \alpha^5 x^3 + \alpha x^6, \\ \beta^{(4)}(x) &= 1 + \alpha^2 x + \alpha^4 x^2 + \alpha^3 x^5, \\ \beta^{(5)}(x) &= 1 + \alpha^3 x^2 + \alpha^5 x^4 + \alpha^6 x^6. \end{aligned}$$

For $\ell = 1$, the coefficients $w_{h,j}^{(\ell)}$ are listed as follows.

- $h = 0 : w_{0,0}^{(1)} = \alpha, w_{0,1}^{(1)} = 1, w_{0,2}^{(1)} = \alpha, w_{0,3}^{(1)} = \alpha^5, w_{0,4}^{(1)} = \alpha, w_{0,5}^{(1)} = \alpha^6, w_{0,6}^{(1)} = 1.$
- $h = 1 : w_{1,0}^{(1)} = \alpha^6, w_{1,1}^{(1)} = 1, w_{1,2}^{(1)} = \alpha^4, w_{1,3}^{(1)} = 1, w_{1,4}^{(1)} = \alpha^5, w_{1,5}^{(1)} = \alpha^6, w_{1,6}^{(1)} = 1.$
- $h = 2 : w_{2,0}^{(1)} = \alpha^3, w_{2,1}^{(1)} = 1, w_{2,2}^{(1)} = \alpha^3, w_{2,3}^{(1)} = \alpha, w_{2,4}^{(1)} = \alpha^2, w_{2,5}^{(1)} = \alpha^3, w_{2,6}^{(1)} = \alpha^2.$
- $h = 6 : w_{6,0}^{(1)} = \alpha^5, w_{6,1}^{(1)} = \alpha^6, w_{6,2}^{(1)} = \alpha^5, w_{6,3}^{(1)} = \alpha^6, w_{6,4}^{(1)} = \alpha^3, w_{6,5}^{(1)} = \alpha^6, w_{6,6}^{(1)} = \alpha^4.$

Similarly, for $\ell = 2$, the coefficients $w_{h,j}^{(\ell)}$ are listed as follows.

- $h = 0 : w_{0,0}^{(1)} = 1, w_{0,1}^{(1)} = \alpha^2, w_{0,2}^{(1)} = 1, w_{0,3}^{(1)} = 0, w_{0,4}^{(1)} = \alpha^2, w_{0,5}^{(1)} = \alpha^6, w_{0,6}^{(1)} = \alpha^6.$
- $h = 1 : w_{1,0}^{(1)} = \alpha^2, w_{1,1}^{(1)} = 1, w_{1,2}^{(1)} = 0, w_{1,3}^{(1)} = \alpha^2, w_{1,4}^{(1)} = \alpha^6, w_{1,5}^{(1)} = \alpha^6, w_{1,6}^{(1)} = 1.$
- $h = 3 : w_{3,0}^{(1)} = 0, w_{3,1}^{(1)} = \alpha^2, w_{3,2}^{(1)} = \alpha^6, w_{3,3}^{(1)} = \alpha^6, w_{3,4}^{(1)} = 1, w_{3,5}^{(1)} = \alpha^2, w_{3,6}^{(1)} = 1.$
- $h = 6 : w_{6,0}^{(1)} = \alpha^6, w_{6,1}^{(1)} = 1, w_{6,2}^{(1)} = \alpha^2, w_{6,3}^{(1)} = 1, w_{6,4}^{(1)} = 0, w_{6,5}^{(1)} = \alpha^2, w_{6,6}^{(1)} = \alpha^6.$

For the remaining three MWDC polynomials, the coefficients $w_{h,j}^{(\ell)}$ can also be determined through the same process. After performing the shift-sum operations over all the MWDC polynomials, the following frequency matrix is obtained as

$$\mathbf{\Phi} = \begin{bmatrix} 5 & 1 & 4 & 4 & 5 & 1 & 4 \\ 3 & 10 & 2 & 2 & 1 & 1 & 5 \\ 2 & 1 & 3 & 2 & 2 & 1 & 1 \\ 3 & 1 & 1 & 1 & 2 & 2 & 2 \\ 1 & 2 & 4 & 4 & 3 & 2 & 2 \\ 2 & 2 & 2 & 2 & 2 & 10 & 2 \\ 1 & 2 & 3 & 2 & 3 & 1 & 2 \\ 3 & 1 & 1 & 3 & 2 & 2 & 2 \end{bmatrix}.$$

[1]It is assumed that $\mathbb{F}_8$ is defined by the primitive polynomial $\alpha^3 + \alpha + 1$. Moreover, let $\mathbb{F}_8 = \{\sigma_0, \sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \sigma_6, \sigma_7\} = \{0, 1, \alpha, \alpha^3, \alpha^2, \alpha^6, \alpha^4, \alpha^5\}$.

$$\begin{aligned} w(x) = \; & \varepsilon(x) + \beta_{b_2} x^{b_2} \varepsilon(x) + \cdots + \beta_{b_{d^{\perp}}} x^{b_{d^{\perp}}} \varepsilon(x) \mod (x^n - 1) \\ = \; & \varepsilon_{e_1} x^{e_1} + \varepsilon_{e_2} x^{e_2} + \cdots + \varepsilon_{e_\tau} x^{e_\tau} + \\ & \beta_{b_2} \varepsilon_{e_1} x^{e_1 + b_2} + \beta_{b_2} \varepsilon_{e_2} x^{e_2 + b_2} + \cdots + \beta_{b_2} \varepsilon_{e_\tau} x^{e_\tau + b_2} + \\ & \vdots \\ & \beta_{b_{d^{\perp}}} \varepsilon_{e_1} x^{e_1 + b_{d^{\perp}}} + \beta_{b_{d^{\perp}}} \varepsilon_{e_2} x^{e_2 + b_{d^{\perp}}} + \cdots + \beta_{b_{d^{\perp}}} \varepsilon_{e_\tau} x^{e_\tau + b_{d^{\perp}}}, \end{aligned} \tag{16}$$

Note that $\sum_{i=0}^{7}\phi_{i,j} = Ld^{\perp} = 20, \forall j$. It can be observed that $\phi_{1,1} = 10$ and $\phi_{5,5} = 10$, which are the largest values. Therefore, we can consider symbols r_1 and r_5 are more likely to be erroneous. The corresponding error magnitudes are $\sigma_1 = 1$ and $\sigma_5 = \alpha^6$, respectively. Consequently, the error polynomial is $\varepsilon(x) = x + \alpha^6 x^5$. The codeword polynomial can be recovered by $c(x) = r(x) - \varepsilon(x)$. Furthermore, it can be seen that $\phi_{0,1}$ and $\phi_{0,5}$ are smaller than the rest of the entries in the first row. This also implies that r_1 and r_5 are more likely to be the erroneous symbols. □

IV. Plausibility Analysis

This section provides the plausibility analysis of the proposed shift-sum operation, which looks into the statistical distribution of the frequency matrix's entries. In particular, given τ errors, the expectation of $\phi_{i,j}$ at the erroneous and non-erroneous positions are characterized. We will first conduct this analysis on non-binary codes, followed by its simplification to binary case. This analysis improves the results given in [30].

A. Non-Binary Codes

Since each of the syndrome polynomials $w_h^{\ell}(x)$ contributes equally to the frequency matrix, we will simply notate them as $w(x) = \sum_{j=0}^{n-1} w_j x^j$ in the following analysis. Moreover, we write out $\varepsilon(x)$ as $\varepsilon(x) = \sum_{j=0}^{n-1}\varepsilon_j x^j$. Based on (15) and (16), its coefficient w_j can be written as

$$w_j = \varepsilon_j + \beta_{b_2}\varepsilon_{j-b_2} + \beta_{b_3}\varepsilon_{j-b_3} + \cdots + \beta_{b_{d^{\perp}}}\varepsilon_{j-b_{d^{\perp}}}, \quad (26)$$

where the subscripts of ε are calculated $\mod n$. The plausibility analysis aims to determine the probability of w_j being ε_j when $j \in \mathcal{E}$, and the probability of w_j being zero when $j \in \mathcal{E}^{\mathrm{c}}$.

Let $\xi_m \in \mathbb{F}_{2^p}\backslash\{0\}$ for $m \in \mathbb{Z}^+$. We define A_t as the probability of $\sum_{m=1}^{t}\xi_m$ being nonzero, i.e.,

$$A_t \triangleq \Pr\Big(\sum_{m=1}^{t}\xi_m \neq 0\Big), \quad (27)$$

where $t \in \mathbb{Z}^+$. Note that the summation of ξ_m is performed over $\mathbb{F}_{2^p}$. The following lemma characterizes A_t.

Lemma 1: Suppose ξ_m is uniformly drawn from $\mathbb{F}_{2^p}\backslash\{0\}$. For $t \in \mathbb{Z}^+$, we have

$$A_t = 1 - \frac{1}{2^p} + \frac{1}{2^p}\Big(\frac{1}{2^p-1}\Big)^{t-1}(-1)^{t-1}. \quad (28)$$

Proof: If $\sum_{m=1}^{t}\xi_m = 0$, based on $\xi_{t+1} \neq 0$, $\sum_{m=1}^{t+1}\xi_m \neq 0$. This means that $\Pr(\sum_{m=1}^{t+1}\xi_m \neq 0 | \sum_{m=1}^{t}\xi_m = 0) = 1$. Otherwise, if $\sum_{m=1}^{t}\xi_m \neq 0$, $\sum_{m=1}^{t+1}\xi_m = 0$ if and only if $\sum_{m=1}^{t}\xi_m = -\xi_{t+1}$. Since ξ_{t+1} is uniformly drawn from $\mathbb{F}_{2^p}\backslash\{0\}$, $\Pr(\sum_{m=1}^{t+1}\xi_m \neq 0 | \sum_{m=1}^{t}\xi_m \neq 0) = \frac{2^p-2}{2^p-1}$. Therefore, based on the law of total probability, the relationship between A_t and A_{t+1} is

$$\begin{aligned}
&A_{t+1}\\
&\triangleq \Pr\Big(\sum_{m=1}^{t+1}\xi_m \neq 0\Big)\\
&= \Pr\Big(\sum_{m=1}^{t+1}\xi_m \neq 0 \Big| \sum_{m=1}^{t}\xi_m = 0\Big)\cdot\Pr\Big(\sum_{m=1}^{t}\xi_m = 0\Big) +\\
&\Pr\Big(\sum_{m=1}^{t+1}\xi_m \neq 0 \Big| \sum_{m=1}^{t}\xi_m \neq 0\Big)\cdot\Pr\Big(\sum_{m=1}^{t}\xi_m \neq 0\Big)\\
&= (1 - A_t) + \frac{2^p-2}{2^p-1}A_t.
\end{aligned}$$

The above equation can be manipulated as

$$A_{t+1} - \frac{2^p-1}{2^p} = -\frac{1}{2^p-1}\big(A_t - \frac{2^p-1}{2^p}\big).$$

Hence, $\{A_t - \frac{2^p-1}{2^p}, \forall t\}$ forms a geometric sequence. With the initial condition of $A_1 = 1$, we can obtain (28). ■

Based on (22) and (23), frequency matrix entries $\phi_{i,j}$ can be categorized into the following four cases based on their positions and the corresponding values. The probability and expectation of these cases occurring are analyzed accordingly. During the analysis, error magnitude ε_{e_m} is assumed to be uniformly drawn from $\mathbb{F}_{2^p}\backslash\{0\}$, where $m = 1, 2, \ldots, \tau$.

Case 1: Let $j \in \mathcal{E}$ and $w_j = \varepsilon_j \neq 0$. In this case, for the erroneous positions j, $w_j = \varepsilon_j$, and hence $\sum_{m=2}^{d^{\perp}}\beta_{b_m}\varepsilon_{j-b_m} = 0$. The probability of this event occurring is analyzed as follows. Suppose there exist t (at most $\tau - 1$) nonzero values among $\varepsilon_{j-b_2}, \varepsilon_{j-b_3}, \ldots, \varepsilon_{j-b_{d^{\perp}}}$. The other $\tau - 1 - t$ nonzero values would appear in the remaining ε_j. They do not affect the value w_j. Since $\beta_{b_m} \neq 0$, there exist t nonzero values $\beta_{b_m}\varepsilon_{j-b_m}$ in the summation, which are also uniformly drawn from $\mathbb{F}_{2^p}\backslash\{0\}$. Based on Lemma 1, the sum of probabilities of ε_{j-b_m}'s, for which the weighted sum is nonzero, i.e., $\sum_{m=2}^{d^{\perp}}\beta_{b_m}\varepsilon_{j-b_m} \neq 0$, are $A_t\binom{d^{\perp}-1}{t}\binom{n-d^{\perp}}{\tau-1-t}$. Since one error has occurred at position j, the remaining $\tau - 1$ errors should appear at the other $n - 1$ positions, resulting in $\binom{n-1}{\tau-1}$ possibilities. Consequently, the probability of Case 1 occurring is

$$P_1(\tau) = 1 - \frac{\sum_{t=1}^{\tau-1}A_t\binom{d^{\perp}-1}{t}\binom{n-d^{\perp}}{\tau-1-t}}{\binom{n-1}{\tau-1}}, \quad (29)$$

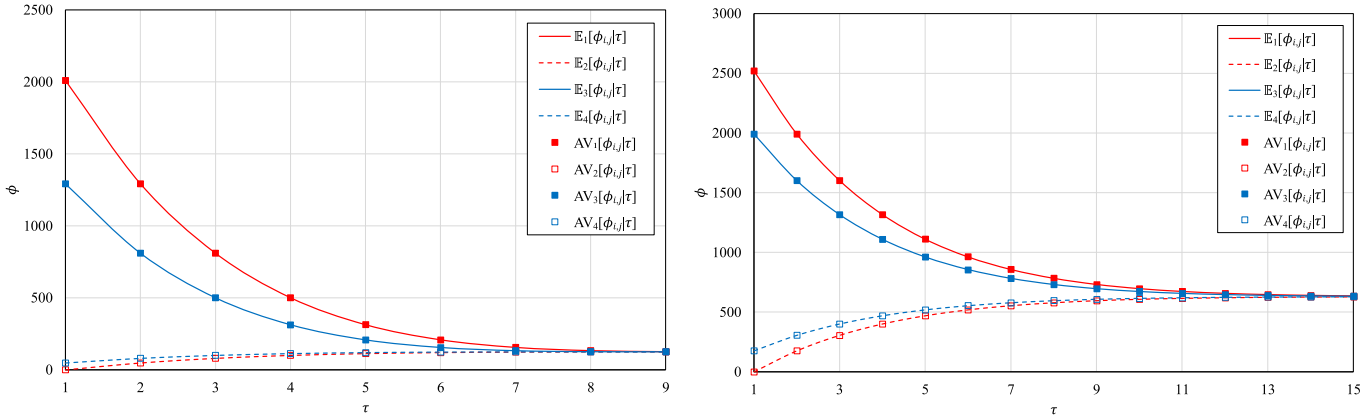
where $t \leq d^{\perp} - 1$ and $\tau - 1 - t \leq n - d^{\perp}$. Ranging over all MWDCs and their cyclic shifts, the expectation of $\phi_{i,j}$ is

$$\mathbb{E}_1[\phi_{i,j}|\tau] = Ld^{\perp}\cdot P_1(\tau), \quad (30)$$

where $j \in \mathcal{E}$ and $i = \arg_{i'}\{\sigma_{i'} = \varepsilon_j\}$.

Case 2: Let $j \in \mathcal{E}$ and $w_j \neq \varepsilon_j$. In this case, for the erroneous positions j, $w_j \neq \varepsilon_j$, and hence $\sum_{m=2}^{d^{\perp}}\beta_{b_m}\varepsilon_{j-b_m} \neq 0$. Since $w_j \in \mathbb{F}_{2^p}\backslash\{\varepsilon_j\}$, similar to Case 1, the probability of Case 2 occurring is

$$P_2(\tau) = \frac{1}{2^p-1}\cdot\frac{\sum_{t=1}^{\tau-1}A_t\binom{d^{\perp}-1}{t}\binom{n-d^{\perp}}{\tau-1-t}}{\binom{n-1}{\tau-1}}, \quad (31)$$



(a) RS code $\mathcal{C}(16; 15, 5, 11)$ (b) NB-BCH code $\mathcal{C}(4; 63, 27, 21)$

Fig. 1. Plausibility analysis of the non-binary codes.

where $t \leq d^{\perp} - 1$ and $\tau - 1 - t \leq n - d^{\perp}$. Therefore, the expectation of $\phi_{i,j}$ is

$$\mathbb{E}_2[\phi_{i,j}|\tau] = Ld^{\perp} \cdot P_2(\tau), \tag{32}$$

where $j \in \mathcal{E}$ and $i \in \{i' \mid \sigma_{i'} \neq \varepsilon_j\}$.

Case 3: Let $j \in \mathcal{E}^{\mathrm{c}}$ and $w_j = 0$. In this case, for the non-erroneous positions j, $w_j = 0$, and hence $\sum_{m=2}^{d^{\perp}} \beta_{b_m} \varepsilon_{j-b_m} = 0$. Note that in this case, $\varepsilon_j = 0$. Assume that there are t (at most τ) nonzero values among $\varepsilon_{j-b_2}, \varepsilon_{j-b_3}, \ldots, \varepsilon_{j-b_{d^{\perp}}}$. Based on Lemma 1, the sum of probabilities of ε_{j-b_m} for $\sum_{m=2}^{d^{\perp}} \beta_{b_m} \varepsilon_{j-b_m}$ being nonzero are $A_t\binom{d^{\perp}-1}{t}\binom{n-d^{\perp}}{\tau-t}$. Meanwhile, τ errors should occur at the remaining $n-1$ positions, resulting in $\binom{n-1}{\tau}$ possibilities. Therefore, the probability of Case 3 occurring is

$$P_3(\tau) = 1 - \frac{\sum_{t=1}^{\tau} A_t\binom{d^{\perp}-1}{t}\binom{n-d^{\perp}}{\tau-t}}{\binom{n-1}{\tau}}, \tag{33}$$

where $t \leq d^{\perp} - 1$ and $\tau - t \leq n - d^{\perp}$. Consequently, the expectation of $\phi_{i,j}$ is

$$\mathbb{E}_3[\phi_{i,j}|\tau] = Ld^{\perp} \cdot P_3(\tau), \tag{34}$$

where $j \in \mathcal{E}^{\mathrm{c}}$ and $i = 0$.

Case 4: Let $j \in \mathcal{E}^{\mathrm{c}}$ and $w_j \neq 0$. In this case, for the non-erroneous positions j, $w_j \neq 0$, and hence $\sum_{m=2}^{d^{\perp}} \beta_{b_m} \varepsilon_{j-b_m} \neq 0$. Since $w_j \in \mathbb{F}_{2^p} \backslash \{0\}$, the probability of Case 4 occurring is

$$P_4(\tau) = \frac{1}{2^p - 1} \cdot \frac{\sum_{t=1}^{\tau} A_t\binom{d^{\perp}-1}{t}\binom{n-d^{\perp}}{\tau-t}}{\binom{n-1}{\tau}}, \tag{35}$$

where $t \leq d^{\perp} - 1$ and $\tau - t \leq n - d^{\perp}$. Finally, the expectation of $\phi_{i,j}$ is

$$\mathbb{E}_4[\phi_{i,j}|\tau] = Ld^{\perp} \cdot P_4(\tau), \tag{36}$$

where $j \in \mathcal{E}^{\mathrm{c}}$ and $i \neq 0$.

Example 2 Continuing from Example 1, we know $\mathcal{E} = \{1, 5\}$ and $\mathcal{E}^{\mathrm{c}} = \{0, 2, 3, 4, 6\}$. Therefore, $\phi_{1,1}$ and $\phi_{5,5}$ are categorized in Case 1, while $\{\phi_{i,1}|\forall i\} \backslash \phi_{1,1}$ and $\{\phi_{i,5}|\forall i\} \backslash \phi_{5,5}$ are categorized in Case 2. Similarly, $\phi_{0,0}$, $\phi_{0,2}$, $\phi_{0,3}$, $\phi_{0,4}$ and $\phi_{0,6}$ are categorized in Case 3, while the remaining elements are categorized in Case 4.

Remark 1: For Case 1 and Case 2, we have

$$P_1(\tau) + (2^p - 1)P_2(\tau) = 1 \tag{37}$$

and

$$\mathbb{E}_1[\phi_{i,j}|\tau] + (2^p - 1)\mathbb{E}_2[\phi_{i,j}|\tau] = Ld^{\perp}. \tag{38}$$

While for Case 3 and Case 4,

$$P_3(\tau) + (2^p - 1)P_4(\tau) = 1 \tag{39}$$

and

$$\mathbb{E}_3[\phi_{i,j}|\tau] + (2^p - 1)\mathbb{E}_4[\phi_{i,j}|\tau] = Ld^{\perp}. \tag{40}$$

Both (38) and (40) vindicate the conclusion of (25).

Fig. 1 shows plausibility analysis of two non-binary cyclic codes, the RS code $\mathcal{C}(16; 15, 5, 11)$ and the NB-BCH code $\mathcal{C}(4; 63, 27, 21)$. Their dual codes are RS code $\mathcal{C}(16; 15, 10, 6)$ and NB-BCH code $\mathcal{C}(4; 63, 36, 14)$, respectively. The analytical results are compared with the average values (AV) that were obtained through simulations by running 10 000 decoding events for each τ. These average values are denoted as $\mathrm{AV}_1[\phi_{i,j}|\tau]$, $\mathrm{AV}_2[\phi_{i,j}|\tau]$, $\mathrm{AV}_3[\phi_{i,j}|\tau]$ and $\mathrm{AV}_4[\phi_{i,j}|\tau]$ for the four cases, respectively. Note that we have used 335 and 180 cyclically different MWDCs for the RS and NB-BCH codes, respectively. It can be seen that our characterizations on the expectations of $\phi_{i,j}$ match well with the simulation results. The discrepancy between $\mathbb{E}_1[\phi_{i,j}|\tau]$ and $\mathbb{E}_2[\phi_{i,j}|\tau]$ yields the capability on distinguishing the most likely error magnitude and the other elements at the erroneous positions. Similarly, the discrepancy between $\mathbb{E}_3[\phi_{i,j}|\tau]$ and $\mathbb{E}_4[\phi_{i,j}|\tau]$ yields the capability on determining the non-erroneous positions. Fig. 1 shows that for the RS code, when $1 \leq \tau \leq 7$, $\mathbb{E}_1[\phi_{i,j}|\tau] > \mathbb{E}_2[\phi_{i,j}|\tau]$ and $\mathbb{E}_3[\phi_{i,j}|\tau] > \mathbb{E}_4[\phi_{i,j}|\tau]$. While for the NB-BCH code, this property holds for the region of $1 \leq \tau \leq 13$. These results reveal that even when the number of errors is greater than half of the code's minimum Hamming distance, it is still possible to utilize $\phi_{i,j}$ to identify the erroneous positions and further correct the errors. This observation vindicates the shift-sum decoding's advanced error-correction capability.

On the other hand, the discrepancy between $\mathbb{E}_1[\phi_{i,j}|\tau]$ and $\mathbb{E}_2[\phi_{i,j}|\tau]$ (or $\mathbb{E}_3[\phi_{i,j}|\tau]$ and $\mathbb{E}_4[\phi_{i,j}|\tau]$) is very large for a small number of errors. It guarantees errors to be corrected since the erroneous positions can be determined uniquely. In the opposite, the discrepancy reduces as the number of errors increases, making it less confident to identify the error positions. An iterative shift-sum decoding would be necessary, in which the most likely errors can first be corrected. By iteratively reducing the number of errors in the received word polynomial $r(x)$, the discrepancy will increase again. The shift-sum decoding is then capable to correct errors beyond the above half distance bound. This leads to the iterative shift-sum decoding which will be described in the next section.

B. Binary Codes

We now apply the above analysis to the case of binary cyclic codes. For this case, the error polynomial is

$$\varepsilon(x) = \sum_{j=0}^{n-1} \varepsilon_j x^j, \tag{41}$$

where $\varepsilon_j = 1$ if $j \in \mathcal{E}$ and $\varepsilon_j = 0$ if $j \in \mathcal{E}^{\mathrm{c}}$. Meanwhile, the dual codeword polynomial of (13) with weight $d^{\perp}$ can be simplified into

$$\beta(x) = 1 + x^{b_2} + \cdots + x^{b_{d^{\perp}}}. \tag{42}$$

Therefore, coefficients of the syndrome polynomial $w(x)$ become

$$w_j = \varepsilon_j + \varepsilon_{j-b_2} + \cdots + \varepsilon_{j-b_{d^{\perp}}}, \tag{43}$$

where the subscripts of ε are calculated $\mod n$ and w_j is either 0 or 1.

Recently, a plausibility analysis of the shift-sum decoding for binary BCH codes has been presented in [30]. Given τ errors, the expectation of $\mathrm{wt}(w(x))$ is

$$\mathbb{E}[\mathrm{wt}(w(x))] = \frac{n \sum_{t=1,t \text{ is odd}}^{\tau} \binom{d^{\perp}}{t}\binom{n-d^{\perp}}{\tau-t}}{\binom{n}{\tau}}, \tag{44}$$

where $t \le d^{\perp}$ and $\tau - t \le n - d^{\perp}$. Let ϕ^{e} and ϕ^{c} denote the frequency of one among all coefficients w_j for $j \in \mathcal{E}$ and $j \in \mathcal{E}^{\mathrm{c}}$, respectively. Their expectation can be determined by [30]

$$\mathbb{E}_0[\phi^{\mathrm{e}}|\tau] = \frac{\mathbb{E}[\mathrm{wt}(w(x))]}{\tau} L \tag{45}$$

and

$$\mathbb{E}_0[\phi^{\mathrm{c}}|\tau] = \frac{(d^{\perp}-1)\cdot \mathbb{E}[\mathrm{wt}(w(x))]}{n-\tau} L, \tag{46}$$

respectively. However, as Fig. 2 shows, these characterizations deviate from the simulation results as τ increases. The accuracy can be improved by degenerating the above non-binary analysis to the binary case. Note that in case of binary codes, the definitions of ϕ^{e} and ϕ^{c} are equivalent to Case 1 and Case 4, respectively.

Corollary 2: For binary codes, i.e., $p = 1$, $A_t = 1$ if t is odd while $A_t = 0$ if t is even.

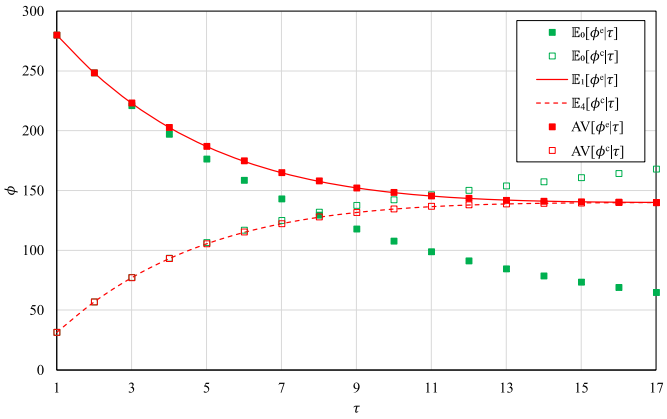


Fig. 2. Plausibility analysis of the BCH code $\mathcal{C}(2; 63, 24, 15)$.

Therefore, when $j \in \mathcal{E}$ and $\sigma_i = w_j = 1$, (30) is simplified to

$$\begin{aligned}\mathbb{E}_1[\phi^{\mathrm{e}}|\tau] &\triangleq \mathbb{E}_1[\phi_{i,j}|\tau] \\ &= Ld^{\perp} \cdot \left(1 - \frac{\sum_{t \text{ is odd}} \binom{d^{\perp}-1}{t}\binom{n-d^{\perp}}{\tau-1-t}}{\binom{n-1}{\tau-1}}\right).\end{aligned} \tag{47}$$

When $j \in \mathcal{E}^{\mathrm{c}}$ and $\sigma_i = w_j = 1$, (36) becomes

$$\mathbb{E}_4[\phi^{\mathrm{c}}|\tau] \triangleq \mathbb{E}_4[\phi_{i,j}|\tau] = Ld^{\perp} \cdot \left(\frac{\sum_{t \text{ is odd}} \binom{d^{\perp}-1}{t}\binom{n-d^{\perp}}{\tau-t}}{\binom{n-1}{\tau}}\right). \tag{48}$$

Fig. 2 shows the plausibility analysis of the binary BCH code $\mathcal{C}(2; 63, 24, 15)$, whose dual code is also a BCH code $\mathcal{C}(2; 63, 39, 8)$. There are $L = 35$ cyclically different MWDCs with $d^{\perp} = 8$. The average values of ϕ^{e} and ϕ^{c}, denoted as $\mathrm{AV}[\phi^{\mathrm{e}}|\tau]$ and $\mathrm{AV}[\phi^{\mathrm{c}}|\tau]$, respectively, were obtained through running 10 000 decoding events for each τ. It can be seen that $\mathbb{E}_0[\phi^{\mathrm{e}}|\tau]$ and $\mathbb{E}_0[\phi^{\mathrm{c}}|\tau]$ deviate from the empirical results as τ increases. This is because the plausibility analysis of [30] is reached based on a coarse estimation of the weight of $w(x)$. Instead of directly computing $\mathrm{wt}(w(x))$, our analysis derives the expectation of coefficient w_j being nonzero for erroneous and non-erroneous positions, respectively. Therefore, our characterizations of $\mathbb{E}_1[\phi^{\mathrm{e}}|\tau]$ and $\mathbb{E}_4[\phi^{\mathrm{c}}|\tau]$ match well with the simulated $\mathrm{AV}[\phi^{\mathrm{e}}|\tau]$ and $\mathrm{AV}[\phi^{\mathrm{c}}|\tau]$, respectively. These characteristics improve over the results of [30], i.e., $\mathbb{E}_0[\phi^{\mathrm{e}}|\tau]$ and $\mathbb{E}_0[\phi^{\mathrm{c}}|\tau]$.

V. Iterative Shift-Sum Decoding Algorithms

This section first proposes two iterative decoding algorithms based on the above shift-sum method. They are the HISS and the SISS algorithms. By further integrating into the Chase decoding, the HISS or SISS algorithms can be utilized to decode the test-vectors, resulting in an enhanced error-correction capability.

A. The HISS Algorithm

The shift-sum decoding yields a reliability measure by multiplying the MWDCs and their cyclic shifts with the received

word polynomial $r(x)$, and further counting the frequency of the coefficients at each position. This measure can be utilized to determine the error positions and magnitudes so as to update $r(x)$ and reduce its containing errors. This process will be iteratively performed until a codeword is found (the errors in $r(x)$ have been removed), or the maximum iteration number $I_{\max}$ is reached.

With the above mentioned shift-sum decoding, the reliability measures $\phi_{i,j}$ can be obtained. Since $\sigma_0 = 0$, $w_{h,j}^{(\ell)} = \sigma_0$ indicates position j is correct and vice versa. Based on (25), a smaller $\phi_{0,j}$ implies a larger $\sum_{i=1}^{2^p-1} \phi_{i,j}$, which implies that the original errors and the shifted scalar multiples are likely to occur at position j. Therefore, r_j is more likely to be a corrupted symbol. The HISS algorithm will iteratively modify $r(x)$ by determining the possible error positions and magnitudes at each iteration. At the beginning, $\phi_{0,0}, \phi_{0,1}, \ldots, \phi_{0,n-1}$ are sorted in an ascending order, yielding a new sequence $j_0^{(1)}, j_1^{(1)}, \ldots, j_{n-1}^{(1)}$ such that

$$\phi_{0,j_0^{(1)}} \leq \phi_{0,j_1^{(1)}} \leq \cdots \leq \phi_{0,j_{n-1}^{(1)}}. \tag{49}$$

Secondly, let $\varphi_j = \max\{\phi_{i,j} \mid i = 1, 2, \ldots, 2^p - 1\}$ and $\gamma_j = \sigma_{\mathsf{m}_j}$, where $\mathsf{m}_j = \arg\max_i\{\phi_{i,j} \mid \forall i, i \neq 0\}$. Note that a larger φ_j also indicates that position j is more likely to be erroneous and γ_j would be the corresponding error magnitude. By sorting $\varphi_0, \varphi_1, \ldots, \varphi_{n-1}$ in a descending order, we can obtain another sequence $j_0^{(2)}, j_1^{(2)}, \ldots, j_{n-1}^{(2)}$ such that

$$\varphi_{j_0^{(2)}} \geq \varphi_{j_1^{(2)}} \geq \cdots \geq \varphi_{j_{n-1}^{(2)}}. \tag{50}$$

By introducing λ as a positive integer, the following two index sets can be defined as $\Lambda^{(1)} = \{j_0^{(1)}, j_1^{(1)}, \ldots, j_{\lambda-1}^{(1)}\}$ and $\Lambda^{(2)} = \{j_0^{(2)}, j_1^{(2)}, \ldots, j_{\lambda-1}^{(2)}\}$. Further let $\Lambda = \Lambda^{(1)} \cap \Lambda^{(2)}$ denote the index set of the updated positions in $r(x)$. The received polynomial $r(x)$ is iteratively updated as

$$r(x) \leftarrow r(x) - \gamma(x), \tag{51}$$

where

$$\gamma(x) = \sum_{j \in \Lambda} \sigma_{\mathsf{m}_j} x^j \in \mathbb{F}_{2^p}(x) \tag{52}$$

is the corresponding updated polynomial. Since $\lambda \geq |\Lambda|$, λ can be regarded as an upper bound on the number of updated positions of $r(x)$ in one iteration. If $r(x) \in \mathcal{C}(2^p; n, k, d)$, a codeword is found,[2] and the HISS algorithm will terminate and output $r(x)$. Otherwise, the shift-sum decoding will be performed to recalculate $\phi_{i,j}$ and determine the newly updated polynomial $\gamma(x)$. The decoding continues until a codeword is found or the maximum iteration number $I_{\max}$ is reached. The HISS algorithm is summarized in Algorithm 1, where $r(x)$ is first checked to see whether it is a valid codeword before the iterative decoding.

Remark 2: The HISS algorithm requires only polynomial multiplications and integer comparisons, which is of practical interest. Note that the HISS algorithm exhibits some similarities to the symbol flipping decoding of non-binary LDPC codes [35], as they both reach parity-check conditions for the iterative decoding. In particular, the HISS algorithm utilizes $\Lambda^{(1)}$ and $\Lambda^{(2)}$ to identify the updated positions and their corresponding magnitudes, while the symbol flipping algorithm determines the updated positions based on the extrinsic information that is calculated from the parity-check equations.

[2] If $r(x)\beta^{(\ell)}(x) = 0 \mod (x^n - 1)$ for any ℓ, $r(x) \in \mathcal{C}(2^p; n, k, d)$.

Algorithm 1 The HISS Algorithm

Input: $r(x)$, $\beta^{(\ell)}(x)$ for $\ell = 1, 2, \ldots, L$, $I_{\max}$, λ;
Output: $r(x) \in \mathcal{C}(2^p; n, k, d)$ or a decoding failure;

```
If r(x) ∈ C(2^p; n, k, d), terminate and output r(x);
For I = 1 to I_max
    Initialize φ_{i,j} = 0, ∀(i, j);
    For ℓ = 1 to L
        For j = 0 to n − 1
            For h ∈ sup(β^(ℓ)(x)) do
                Determine w_{h,j}^(ℓ) as in (21);
                Determine φ_{i,j} as in (22) (23);
            End For
        End For
    End For
    Determine Λ and γ(x) as in (52);
    Update r(x) ← r(x) − γ(x);
    If r(x) ∈ C(2^p; n, k, d), terminate and output r(x);
End For
Return decoding failure.
```

B. The SISS Algorithm

With codeword $\underline{c} = (c_0, c_1, \ldots, c_{n-1})$ being transmitted, let $\underline{\mathsf{r}} = (\mathsf{r}_0, \mathsf{r}_1, \ldots, \mathsf{r}_{n-1}) \in \mathbb{R}^n$ denote the received symbol vector. By assuming $\Pr(c_j = \sigma_i) = \frac{1}{2^p}$, an *a posteriori* probability (APP) matrix $\mathbf{\Pi} \in \mathbb{R}^{2^p \times n}$ can be observed, where its entries are denoted as

$$\pi_{i,j} = \Pr(c_j = \sigma_i \mid \mathsf{r}_j), \tag{53}$$

where $0 \leq i \leq 2^p - 1$ and $0 \leq j \leq n - 1$. Note that $\sum_i \pi_{i,j} = 1, \forall j$. Let $\pi_j^{\mathrm{I}} = \max\{\pi_{i,j}.\forall i\}$. The reliability of each hard-decision received symbol r_j can be defined as

$$\omega_j = \frac{\pi_j^{\mathrm{I}}}{1 - \pi_j^{\mathrm{I}}}. \tag{54}$$

For the HISS algorithm, coefficient $w_{h,j}^{(\ell)}$ contributes 1 to the frequency matrix as (22) shows, limiting the error-correction performance. The SISS algorithm aims to utilize the soft information obtained from the channel to enhance the decoding performance, in which an improved weight is determined for $w_{h,j}^{(\ell)}$. Inspired by the BP decoding of LDPC codes [18], we can utilize the soft information of received symbols r_j which are involved in computing $w_{h,j}^{(\ell)}$. Based on (21), the value of $w_{h,j}^{(\ell)}$ depends on $r_{(j+h-u) \mod n}, \forall u \in \sup(\beta^{(\ell)}(x))$. Rather than setting it to 1 as in the HISS algorithm, we can define the contribution of $w_{h,j}^{(\ell)}$ to the frequency matrix as

$$\zeta_{h,j}^{(\ell)} = \min_{\substack{u \in \sup(\beta^{(\ell)}(x)) \\ j \neq (j+h-u) \mod n}} \omega_{(j+h-u) \mod n}, \tag{55}$$

where $j \neq (j+h-u) \mod n$ means that the soft information of r_j is not considered for the weight. Note that it can be simplified into $h \neq u$. The definition of $\zeta_{h,j}^{(\ell)}$ can be regarded as the extrinsic information of $w_{h,j}^{(\ell)}$ obtained from the ℓ-th MWDC and its h-th cyclic shift. With the MWDCs $\beta^{(\ell)}(x)$, the frequency metrics $\phi_{i,j}$ can again be determined as in (23), where its function $T(\ell,i,j,h)$ is redefined as

$$T(\ell,i,j,h) = \begin{cases} \zeta_{h,j}^{(\ell)}, & \text{if } w_{h,j}^{(\ell)} = \sigma_i, \\ 0, & \text{otherwise.} \end{cases} \tag{56}$$

Note that all of $\phi_{i,j}$ are real number. Similar to the HISS algorithm, the SISS algorithm determines the updated set Λ and the updated polynomial $\gamma(x)$ so as to refine the received word polynomial $r(x)$ iteratively. With the modification of $T(\ell,i,j,h)$ as shown in (56), the SISS algorithm can determine the erroneous positions and magnitudes more precisely, yielding a significantly improved decoding performance. We will discuss more on this in the section of simulation results.

Remark 3: Similar to the HISS algorithm, the SISS algorithm is realized with polynomial multiplications and real number comparisons, which is also hardware-friendly.

C. Chase Decoding

In order to further improve the decoding performance, Chase decoding [10] can be employed, in which the above mentioned HISS or SISS algorithm is utilized to decode the test-vectors. Based on matrix $\mathbf{\Pi}$, let $\mathsf{m}_j^{\mathrm{I}} = \arg\max_i\{\pi_{ij}\}$ and $\mathsf{m}_j^{\mathrm{II}} = \arg\max_{i,i\neq\mathsf{m}_j^{\mathrm{I}}}\{\pi_{ij}\}$, the two most likely decisions for r_j are

$$r_j^{\mathrm{I}} = \sigma_{\mathsf{m}_j^{\mathrm{I}}} \text{ and } r_j^{\mathrm{II}} = \sigma_{\mathsf{m}_j^{\mathrm{II}}}. \tag{57}$$

Sort the reliability ω_j of (54) in a descending order, yielding a new symbol index sequence $j_0, j_1, \ldots, j_{n-1}$. It indicates

$$\omega_{j_0} \geq \omega_{j_1} \geq \cdots \geq \omega_{j_{n-1}}. \tag{58}$$

The decision on r_j is more reliable if ω_j is greater, and vice versa. By identifying η unreliable symbols, the reliable symbol index set $\Theta = \{j_0, j_1, \ldots, j_{n-\eta-1}\}$ can be defined. Subsequently, its complementary set $\Theta^{\mathrm{c}} = \{j_{n-\eta}, j_{n-\eta+1}, \ldots, j_{n-1}\}$ contains the index of the unreliable symbols, and $|\Theta^{\mathrm{c}}| = \eta$. Two decisions can be chosen for each of the η unreliable symbols. Consequently, 2^η test-vectors can be formulated, which are denoted by

$$\underline{r}_v = (r_{j_0}^{(v)}, r_{j_1}^{(v)}, \ldots, r_{j_{n-\eta-1}}^{(v)}, r_{j_{n-\eta}}^{(v)}, \ldots, r_{j_{n-1}}^{(v)}), \tag{59}$$

where $v = 1, 2, \ldots, 2^\eta$ and

$$r_j^{(v)} = \begin{cases} r_j^{\mathrm{I}}, & \text{if } j \in \Theta, \\ r_j^{\mathrm{I}} \text{ or } r_j^{\mathrm{II}}, & \text{if } j \in \Theta^{\mathrm{c}}. \end{cases} \tag{60}$$

For each test-vector $\underline{r}_v$, the proposed HISS or SISS algorithm can be utilized to decode. Since all test-vectors can be decoded in parallel, this Chase decoding not only yields an improved error-correction performance, but also maintains a low decoding latency. Note that if the decoding yields multiple codeword candidates, the one whose modulated symbol sequence has the minimum Euclidean distance to $\underline{\mathsf{r}}$ will be chosen as the output. Substantiated by the HISS and the SISS algorithms, these two Chase decoding are further named as the CHISS and the CSISS algorithms, respectively.

Remark 4: In order to ensure the accuracy of the determined erroneous positions and magnitudes, a sufficient number of cyclically different MWDCs are needed. In this paper, we utilize the Lee-Brickell algorithm [36] to formulate a heuristic search of the MWDCs as follows. By randomly generating an error vector $\underline{\varepsilon}$ of weight less than or equal to $d^\perp$, the Lee-Brickell algorithm seeks a codeword whose Hamming distance to $\underline{\varepsilon}$ is minimal. If a nonzero codeword is found, we check whether its weight is $d^\perp$ and whether it is cyclically different from the earlier found codewords. The process continues until a sufficient number of the cyclically different MWDCs are found. However, except for the RS codes, the number of cyclically different MWDCs for most of cyclic codes is unknown. Given an RS code $\mathcal{C}(2^s; n, k, d_{\mathrm{RS}})$, it has [37]

$$L_{\mathrm{RS}} = \frac{1}{n} \sum_{j|\mathrm{GCD}(n-k-1,n)} \tilde{\varphi}(j) \binom{n/j}{(n-k-1)/j} \tag{61}$$

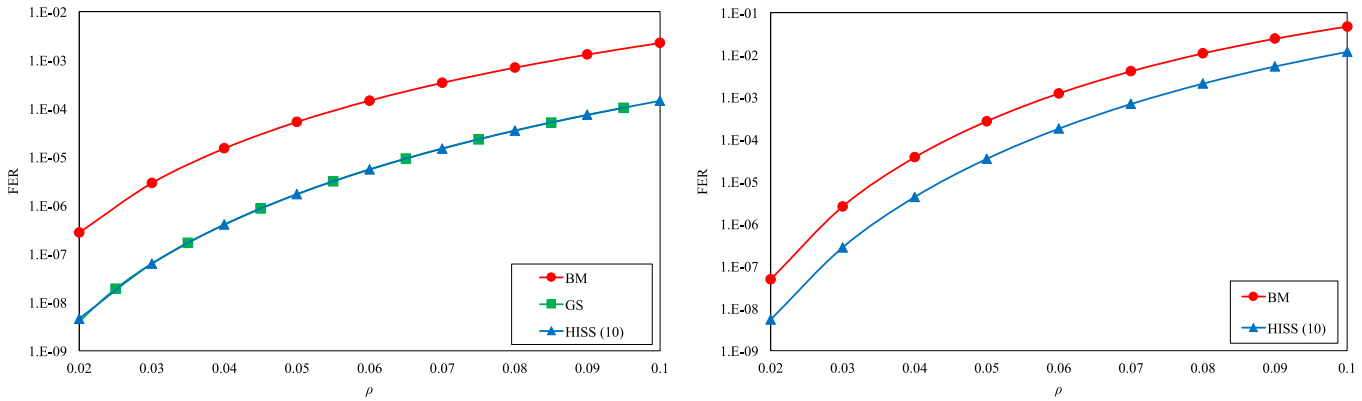
cyclically different MWDCs, where $\tilde{\varphi}(\cdot)$ is the Euler's totient function and $\mathrm{GCD}(n-k-1, n)$ denotes the greatest common divisor (GCD) between $n-k-1$ and n. To the best of our knowledge, a systematic and efficient construction of all cyclically different MWDCs is yet to be developed.

VI. Simulation Results

This section presents the simulation results of the proposed algorithms over three conventional channels, i.e., the memoryless Q-ary symmetric channel, the additive white Gaussian noise (AWGN) channel and the Rayleigh fading channel. The latter two cases use the binary phase-shift keying (BPSK) modulation. The HISS and the SISS algorithms with a maximum iteration number of $I_{\max}$ are denoted as HISS $(I_{\max})$ and SISS $(I_{\max})$, respectively. For each iteration, the maximum number of updated positions is set as $\lambda = \lfloor \frac{d}{4} \rfloor$. Moreover, the CHISS and the CSISS algorithms are denoted as CHISS $(I_{\max}, \eta)$ and CSISS $(I_{\max}, \eta)$, respectively, where η is the number of unreliable symbols. Note that as we have pointed out by Remark 4, finding the cyclically different MWDCs remains heuristic for most cyclic codes. This would be even more challenging for long codes. Hence, we have only simulated short codes to demonstrate the decoding effectiveness of the proposed algorithms. In the following discussions, the coding gains are evaluated at the decoding frame error rate (FER) of 10^{-4}.

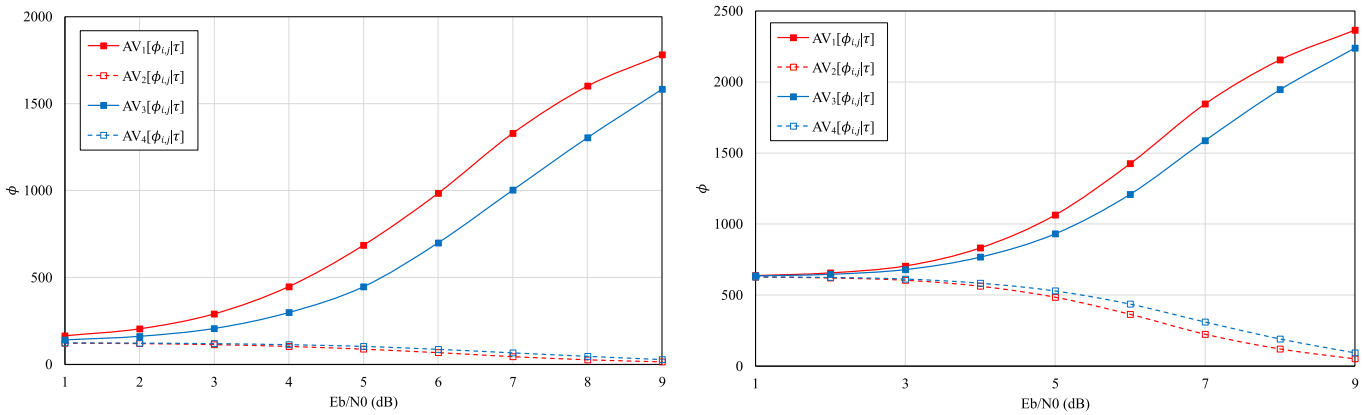
A. Memoryless Q-Ary Symmetric Channel

The Q-ary symmetric channel with an error probability of ρ is defined by taking a Q-ary symbol as its input and outputting either the unchanged input symbol with a probability of $1-\rho$ or one of the other $Q-1$ symbols with a probability of $\frac{\rho}{Q-1}$. During the simulations, we use $Q = 2^p$. For this channel, the performance of the HISS algorithm can be obtained in



(a) RS code $\mathcal{C}(16; 15, 5, 11)$ (b) NB-BCH code $\mathcal{C}(4; 63, 27, 21)$

Fig. 3. Decoding performance of the HISS algorithm over the Q-ary symmetric channel.



(a) RS code $\mathcal{C}(16; 15, 5, 11)$ (b) NB-BCH code $\mathcal{C}(4; 63, 27, 21)$

Fig. 4. Plausibility results of the non-binary codes over the AWGN channel.

a semi-analytical manner. Let $M(\tau)$ denote the number of simulated events with τ errors and $F(\tau)$ denote the number of decoding failures among $M(\tau)$ events. With this information, the decoding FER can be determined by[3]

$$\mathrm{FER}(\rho) = \sum_{\tau=1}^{n} \frac{F(\tau)}{M(\tau)} \binom{n}{\tau} \rho^{\tau}(1-\rho)^{n-\tau}. \tag{62}$$

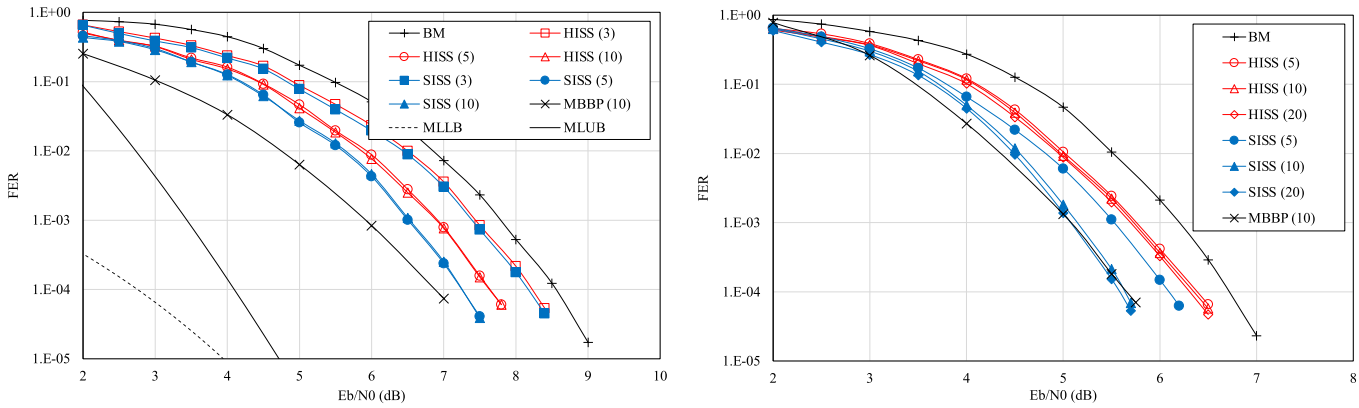
Fig. 3 shows the HISS decoding performance of the RS code $\mathcal{C}(16; 15, 5, 11)$ and the NB-BCH code $\mathcal{C}(4; 63, 27, 21)$ with $I_{\max} = 10$. For the RS code, the BM algorithm can correct up to five symbol errors, while the GS algorithm can correct at most seven symbol errors with an interpolation multiplicity of eight [7]. For the HISS algorithm, $L = 335$ cyclically different MWDCs of weight $d^{\perp} = 6$ are utilized. Note that (61) can validate the number of all cyclically different MWDCs for the RS code $\mathcal{C}(16; 15, 5, 11)$ is 335. Fig. 3(a) shows that the HISS algorithm performs the same as the GS algorithm, outperforming the BM algorithm by a factor of 100 in the FER. During the GS implementation, when the output list contained several candidates with the same Hamming distance to $r(x)$, a random one was selected. For the NB-BCH code, the BM algorithm can correct up to ten symbol errors. We have found 180 cyclically different MWDCs with weight $d^{\perp} = 14$. Note that this may not be the total number of MWDCs for the NB-BCH code $\mathcal{C}(4; 63, 27, 21)$. Fig. 3(b) shows that the proposed algorithm performs better than the BM algorithm by a factor of nearly 10. These results reveal that the HISS algorithm can correct errors beyond half of the code's minimum Hamming distance, demonstrating the advanced decoding potentials of the shift-sum decoding.

[3]Note that we can only simulate several important values of τ to obtain the FER performance since the small-weight (or large-weight) errors are obviously correctable (or uncorrectable). This can significantly reduce the simulation time.

B. AWGN Channel

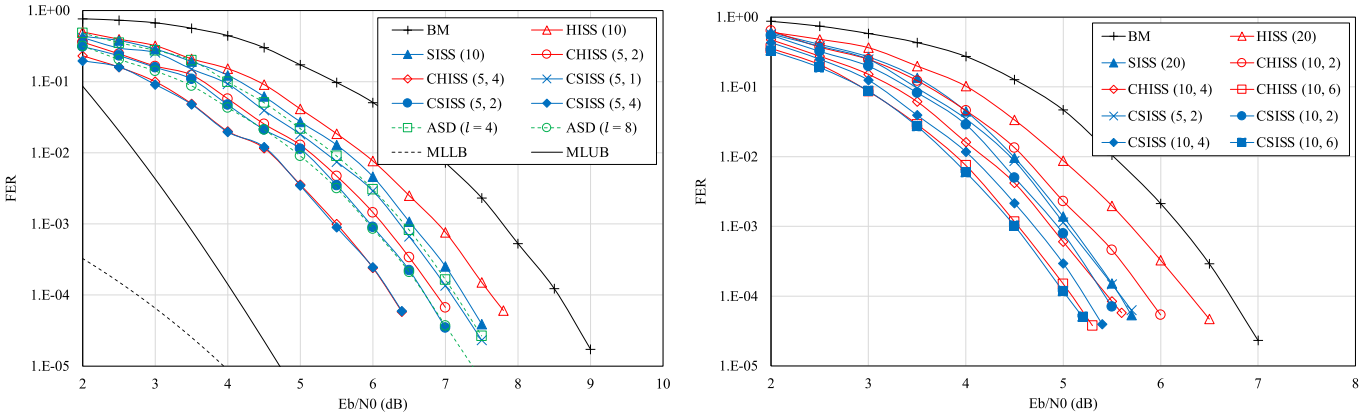
In this paper, cyclic codes are defined over finite fields of characteristic two. Hence, each codeword symbol c_j can be represented by its binary form as $(c_{j,0}, c_{j,1}, \ldots, c_{j,p-1}) \in \mathbb{F}_2^p$, where $j = 0, 1, \ldots, n-1$. Assume they are transmitted over an AWGN channel with two-sided power spectral density $N_0/2$ using BPSK modulation. The signal-to-noise ratio (SNR) is defined as E_b/N_0, where E_b is the transmitted energy per information bit.

Fig. 4 shows the numerical results on the statistical distribution of the frequency matrix's entries $\phi_{i,j}$ over the AWGN channel. It can be seen that as E_b/N_0 increases, the discrepancy between $\mathrm{AV}_1[\phi_{i,j}|\tau]$ and $\mathrm{AV}_2[\phi_{i,j}|\tau]$ (or $\mathrm{AV}_3[\phi_{i,j}|\tau]$



(a) RS code $\mathcal{C}(16; 15, 5, 11)$ (b) NB-BCH code $\mathcal{C}(4; 63, 27, 21)$

Fig. 5. Decoding performance of the HISS and the SISS algorithms over the AWGN channel.



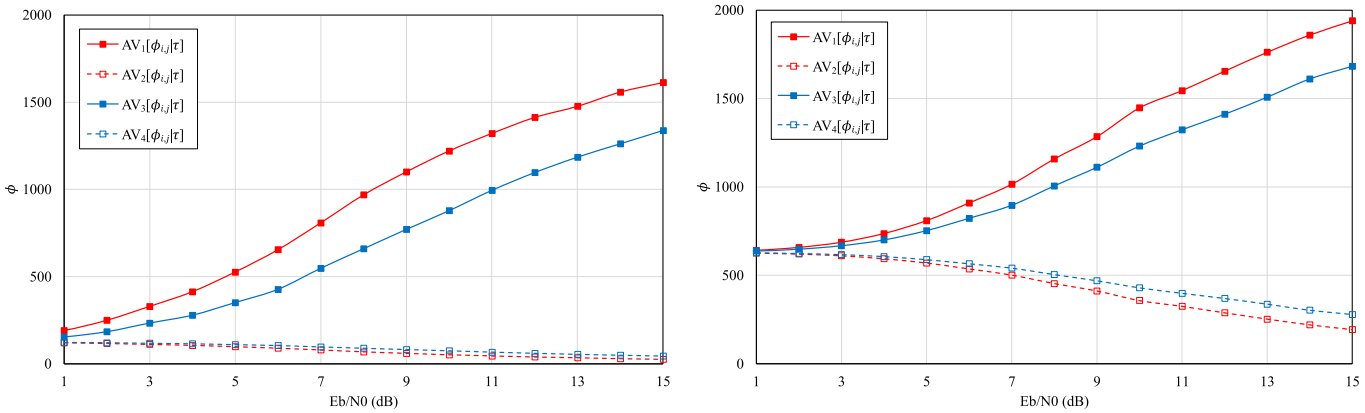
(a) RS code $\mathcal{C}(16; 15, 5, 11)$ (b) NB-BCH code $\mathcal{C}(4; 63, 27, 21)$

Fig. 6. Decoding performance of the CHISS and the CSISS algorithms over the AWGN channel.

and $\mathrm{AV}_4[\phi_{i,j}|\tau]$) also increases for both the RS and the NB-BCH codes. This is due to the number of errors is small when the SNR is high. Although a theoretical plausibility analysis remains impossible, these empirical results vindicate the proposed shift-sum based decoding algorithms can correct errors over the AWGN channel.

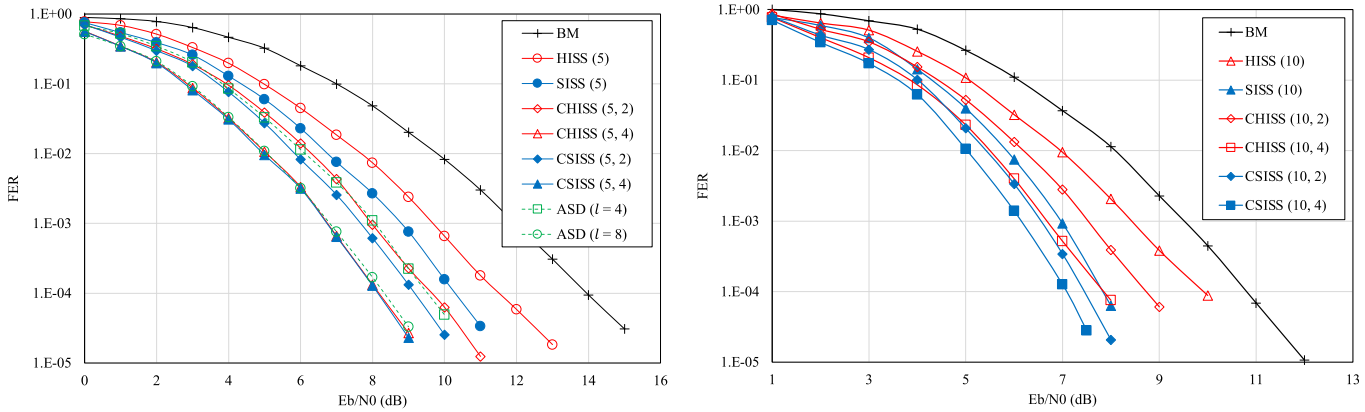
Fig. 5(a) shows the decoding performance of the HISS and the SISS algorithms for the RS code $\mathcal{C}(16; 15, 5, 11)$. The ML decoding upper and lower bounds [38], denoted as MLUB and MLLB, are shown for comparisons. As $I_{\max}$ increases, performance of the HISS and the SISS algorithms improve, outperforming the conventional BM algorithm. When $I_{\max} = 10$, the HISS algorithm yields a coding gain of 0.9 dB over the BM algorithm. By utilizing the soft information obtained from the channel, the SISS algorithm outperforms its hard-decision counterpart, exhibiting an extra 0.3 dB performance gain. But it is still 3.1 dB away from MLUB at the FER of 10^{-4}. It can also be observed that the HISS (or SISS) algorithm can achieve little gain by increasing the iteration number beyond five. This implies that most of the errors have been corrected within the first few iterations. Although the MBBP decoding algorithm [25] with ten iterations performs better than the proposed algorithms, it exhibits a significantly higher decoding complexity which requires both floating point operations and multiple BP decoding trials. Fig. 5(b) shows the decoding performance of the NB-BCH code $\mathcal{C}(4; 63, 27, 21)$. Again, their performance improve as $I_{\max}$ increases. When $I_{\max} = 20$, the HISS and the SISS algorithms can yield 0.4 dB and 1.2 dB coding gains over the BM algorithm, respectively. By comparing Figs. 5(a) and 5(b), it can be seen that the soft-decision decoding achieves greater coding gains than its hard-decision counterpart for the NB-BCH code than for the RS code. Note that the NB-BCH code is defined over a smaller finite field. This results in the symbol reliability metric ω_j, which is a product of the corresponding bit-wise reliabilities, being more reliable. For this code, the SISS algorithm performs similarly as the MBBP algorithm, but with a much lower complexity.

Fig. 6 further provides the decoding performance of the CHISS and the CSISS algorithms. For the RS code, Fig. 6(a) shows the CSISS algorithm with $\eta = 2$ yields a similar error-correction capability as the ASD decoding algorithm with an output list size $l = 8$ [8]. By increasing η to four, both of the Chase decoding based algorithms perform the same and exhibit a coding gain of 2.3 dB over the BM algorithm. For the NB-BCH code, Fig. 6(b) shows that as



(a) RS code $\mathcal{C}(16; 15, 5, 11)$ (b) NB-BCH code $\mathcal{C}(4; 63, 27, 21)$

Fig. 7. Plausibility results of the non-binary codes over the Rayleigh fading channel.



(a) RS code $\mathcal{C}(16; 15, 5, 11)$ (b) NB-BCH code $\mathcal{C}(4; 63, 27, 21)$

Fig. 8. Decoding performance over the Rayleigh fading channel.

η increases, the CHISS and the CSISS algorithms yield a better decoding performance, outperforming the HISS and the SISS algorithms as well as the BM algorithm, with significant coding gains. However, the gap between two Chase decoding variants becomes smaller, which is similar as the RS code. That says Chase decoding yields a better utilization of soft information than the SISS algorithm which only constructs the frequency matrix $\mathbf{\Phi}$ based on soft information. Furthermore, considering the SISS and the CSISS algorithms with the same total iteration number,[4] the CSISS algorithm with $I_{\max} = 5$ and $\eta = 1$ outperforms the SISS algorithm with $I_{\max} = 10$ for the RS code, while the CSISS algorithm with $I_{\max} = 5$ and $\eta = 2$ performs similarly as the SISS algorithm with $I_{\max} = 20$ for the NB-BCH code. Note that the CSISS decoding performance can be further improved by increasing η but it does not hold for the SISS algorithm by increasing $I_{\max}$. This demonstrates that the Chase decoding can yield a better trade-off between the decoding capability and complexity.

C. Rayleigh Fading Channel

The Rayleigh fading channel is memoryless with Doppler shift. It is a fast fading channel, in which the fading coefficients are Rayleigh distributed with a mean value of 1.25 and a variance of 0.43. During the simulations, we assumed coherent detection, i.e., the channel state information and the power allocation are known by both the transmitter and receiver. Fig. 7 shows numerical results on the statistical distribution of the frequency matrix's entries $\phi_{i,j}$ over the Rayleigh fading channel. Similar to the AWGN channel, when E_b/N_0 improves, the discrepancy between $\mathrm{AV}_1[\phi_{i,j}|\tau]$ and $\mathrm{AV}_2[\phi_{i,j}|\tau]$ (or $\mathrm{AV}_3[\phi_{i,j}|\tau]$ and $\mathrm{AV}_4[\phi_{i,j}|\tau]$) becomes larger for both the RS and the NB-BCH codes. Therefore, the proposed shift-sum based decoding algorithms can iteratively correct errors to recover the transmitted message. Fig. 8 shows the decoding performance of the proposed algorithms over the Rayleigh Fading channel. It shows that for the RS code, the HISS and the SISS algorithms outperform the BM algorithm with a coding gain of 2.5 dB and 3.6 dB, respectively. While for the NB-BCH code, they can respectively yield a gain of 0.8 dB and 2.8 dB. The Chase decoding can further improve the performance, yielding at most 5.8 dB for the RS code and 3.7 dB for the NB-BCH code over the BM algorithm. Finally, it should be pointed out that for the RS code, when $\eta = 4$, both the Chase decoding algorithms outperform the ASD decoding algorithm with $l = 8$.

[4]The total iteration number of the CSISS algorithm is defined as $I_{\max} \cdot 2^{\eta}$.

TABLE I
AVERAGE NUMBER OF ITERATIONS AND COMPLEXITY IN DECODING THE RS CODE $\mathcal{C}(16; 15, 5, 11)$

E_b/N_0 (dB)		0	1	2	3	4	5	6	7	8
HISS (3)	Iteration	2.96	2.91	2.81	2.59	2.26	1.80	1.36	1.01	0.73
	Complexity	6.25×10^5	6.14×10^5	5.93×10^5	5.48×10^5	4.77×10^5	3.81×10^5	2.87×10^5	2.14×10^5	1.55×10^5
HISS (5)	Iteration	3.88	3.73	3.47	3.03	2.49	1.88	1.38	1.01	0.73
	Complexity	8.21×10^5	7.87×10^5	7.34×10^5	6.41×10^5	5.26×10^5	3.98×10^5	2.91×10^5	2.14×10^5	1.55×10^5
HISS (10)	Iteration	3.93	3.77	3.49	3.05	2.51	1.89	1.39	1.01	0.73
	Complexity	8.30×10^5	7.96×10^5	7.38×10^5	6.45×10^5	5.29×10^5	4.01×10^5	2.94×10^5	2.14×10^5	1.55×10^5
SISS (3)	Iteration	2.96	2.91	2.81	2.60	2.28	1.84	1.37	1.02	0.73
	Complexity	6.25×10^5	6.15×10^5	5.94×10^5	5.50×10^5	4.81×10^5	3.88×10^5	2.90×10^5	2.15×10^5	1.55×10^5
SISS (5)	Iteration	3.83	3.69	3.46	3.04	2.52	1.93	1.39	1.02	0.73
	Complexity	8.08×10^5	7.78×10^5	7.30×10^5	6.41×10^5	5.32×10^5	4.07×10^5	2.94×10^5	2.16×10^5	1.55×10^5
SISS (10)	Iteration	3.87	3.73	3.49	3.05	2.52	1.93	1.39	1.02	0.73
	Complexity	8.18×10^5	7.88×10^5	7.37×10^5	6.45×10^5	5.33×10^5	4.07×10^5	2.94×10^5	2.16×10^5	1.55×10^5

TABLE II
AVERAGE NUMBER OF ITERATIONS AND COMPLEXITY IN DECODING THE NB-BCH CODE $\mathcal{C}(4; 63, 27, 21)$

E_b/N_0 (dB)		0	1	2	3	4	5	6	7	8
HISS (5)	Iteration	4.99	4.90	4.63	3.79	2.67	1.76	1.22	0.95	0.71
	Complexity	1.19×10^7	1.17×10^7	1.10×10^7	9.02×10^6	6.37×10^6	4.21×10^6	2.91×10^6	2.26×10^6	1.70×10^6
HISS (10)	Iteration	9.81	9.36	8.14	5.59	3.09	1.83	1.22	0.95	0.71
	Complexity	2.34×10^7	2.23×10^7	1.94×10^7	1.33×10^7	7.37×10^6	4.37×10^6	2.91×10^6	2.26×10^6	1.70×10^6
HISS (20)	Iteration	18.70	17.48	14.59	8.80	3.80	1.94	1.22	0.95	0.71
	Complexity	4.46×10^7	4.16×10^7	3.47×10^7	2.10×10^7	9.05×10^6	4.63×10^6	2.91×10^6	2.26×10^6	1.70×10^6
SISS (5)	Iteration	5.00	4.91	4.63	3.86	2.78	1.82	1.24	0.95	0.71
	Complexity	1.19×10^7	1.17×10^7	1.10×10^7	9.21×10^6	6.63×10^6	4.34×10^6	2.96×10^6	2.26×10^6	1.70×10^6
SISS (10)	Iteration	9.81	9.30	7.92	5.31	2.98	1.84	1.24	0.95	0.71
	Complexity	2.34×10^7	2.22×10^7	1.89×10^7	1.26×10^7	7.11×10^6	4.38×10^6	2.96×10^6	2.26×10^6	1.70×10^6
SISS (20)	Iteration	18.50	17.10	13.68	7.74	3.27	1.86	1.24	0.95	0.71
	Complexity	4.41×10^7	4.07×10^7	3.26×10^7	1.84×10^7	7.78×10^6	4.43×10^6	2.96×10^6	2.26×10^6	1.70×10^6

D. Complexity Analysis

Herein, the decoding complexity of both the HISS and the SISS algorithms will be analyzed. It is measured by the amount of finite field multiplications in decoding a codeword. For each iteration, the essential computation of the HISS algorithm is the calculation of coefficients $w_{h,j}^{(\ell)}$, which requires $d^{\perp}$ multiplications. Note that n coefficients need to be computed, and L MWDCs and their $d^{\perp}$ cyclic shifted codewords are utilized in the decoding. Hence, the decoding requires $Ln(d^{\perp})^2$ finite field multiplications, resulting in an asymptotic complexity of $O(Ln(d^{\perp})^2)$. Considering the maximum number of iterations $I_{\max}$ is needed, the worst-case complexity would be $O(I_{\max}Ln(d^{\perp})^2)$. The SISS algorithm utilizes soft information to generate the weight of coefficient $w_{h,j}^{(\ell)}$, resulting in $\phi_{i,j}$ of Step 8 in Algorithm 1 being a real value. Meanwhile, the determination of the updated positions and magnitudes, e.g., Step 12 in Algorithm 1, is different. The HISS algorithm performs the integer comparison, while the SISS algorithm performs the real value comparison. However, in terms of finite field multiplications, the SISS algorithm exhibits the same decoding complexity as the HISS algorithm.

Table I shows the average number of iterations and complexity in decoding the RS code $\mathcal{C}(16; 15, 5, 11)$ over the AWGN channel. These results were obtained by running 10 000 decoding events for each E_b/N_0. It can be seen that these two algorithms yield a similar convergence on the average iteration number. As E_b/N_0 increases, the average number decreases, as a valid codeword is more likely to be produced at an earlier stage. When $E_b/N_0 \geq 8$ dB, the average iteration number is less than one. This is because some of the received word is already a valid codeword without incurring the decoding. When E_b/N_0 reduces, the average iteration number still would not reach $I_{\max}$ since the algorithms usually can find a valid but incorrect codeword after the first few iterations. This also explains the reason why Fig. 5 shows that the decoding performance cannot be improved by increasing the decoding iterations. Table I also shows the average decoding complexity of the RS code, demonstrating the decoding complexity decreases as the E_b/N_0 increases. For the GS decoding, it can correct seven errors with an interpolation multiplicity of eight. Its actual complexity depends on the interpolation approach. The asymptotic complexity of Kötter's interpolation [33] is $O(n^2l^5)$, where l is the maximum output list size. Its empirical average decoding complexity is 9.72×10^6. When using the basis reduction interpolation [34], the asymptotic complexity is $O(n(n-k)l^5)$. Its empirical average complexity is 2.37×10^6. Therefore, the proposed algorithms yield a lower decoding complexity.

Finally, Table II shows the average number of iterations and complexity in decoding the NB-BCH code $\mathcal{C}(4; 63, 27, 21)$. Again, it can be observed that the HISS and the SISS algorithms exhibit similar average iteration number and decoding complexity over the E_b/N_0 region, and these values decrease

with the increase of E_b/N_0. Compared with Tables I and II, we observe that when E_b/N_0 is low, the average number of iterations for decoding the NB-BCH code is much closer to $I_{\max}$ than for decoding the RS code. This difference is caused by the difference of their codebook cardinalities. For the RS code $\mathcal{C}(16; 15, 5, 11)$, it has $16^5 = 1\,048\,576$ codewords, while the NB-BCH code $\mathcal{C}(4; 63, 27, 21)$ has $4^{27} \approx 1.80 \times 10^{16}$ codewords. Therefore, it would be much easier for the iterative decoding to find a valid RS codeword than to find a valid NB-BCH codeword. During this research, we have noticed that even when the SNR is low, the decoding can still find a valid but incorrect RS codeword. This is not the case for the NB-BCH code. Hence, when E_b/N_0 is low, the NB-BCH code needs to perform more decoding iterations than the RS code.

VII. Conclusion

This paper has proposed the shift-sum operation for decoding non-binary cyclic codes. By multiplying a number of MWDCs and their cyclic shifts with the received word polynomial, a frequency matrix can be yielded as a reliability metric for identifying the error positions and magnitudes. Plausibility analysis of the shift-sum operation has been provided, which derives the probability distributions and expectations of the frequency matrix entries, explaining its advanced decoding capability. The HISS and the SISS algorithms have been further proposed to show the performance potentials of the novel shift-sum decoding method. Moreover, they can be realized with only polynomial multiplications and numerical comparisons, which are friendly for practical implementation. It should be highlighted that the HISS algorithm achieves the same advanced decoding performance as the GS algorithm, but yields a lower decoding complexity. To further improve the error-correction capability, the Chase decoding algorithms have been proposed, in which the HISS or the SISS algorithm is utilized to decode the test-vectors, resulting in a significantly improved decoding performance. They can also outperform the ASD decoding algorithm. Simulation results on the RS and the NB-BCH codes have been provided to verify the decoding performance and complexity advantages of the proposed decoding approaches.

References

[1] J. Xing, M. Bossert, S. Bitzer, and L. Chen, "Iterative decoding of non-binary cyclic codes using minimum-weight dual codewords," in *Proc. IEEE Int. Symp. Inform. Theory (ISIT)*, Los Angeles, CA, USA, Jun. 2020, pp. 333–337.

[2] J. Yuan, J. Xing, and L. Chen, "Plausibility analysis of shift-sum decoding for cyclic codes," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2021, pp. 652–657.

[3] T. K. Moon, *Error Correction Coding: Mathematical Methods and Algorithms*. Hoboken, NJ, USA: Wiley, 2020.

[4] E. Berlekamp, *Algebraic Coding Theory*. New York, NY, USA: McGraw-Hill, 1968.

[5] J. Massey, "Shift-register synthesis and BCH decoding," *IEEE Trans. Inf. Theory*, vol. IT-15, no. 1, pp. 122–127, Jan. 1969.

[6] Y. Sugiyama, M. Kasahara, S. Hirasawa, and T. Namekawa, "A method for solving key equation for decoding Goppa codes," *Inf. Control*, vol. 27, no. 1, pp. 87–99, Jan. 1975.

[7] V. Guruswami and M. Sudan, "Improved decoding of Reed–Solomon and algebraic-geometric codes," *IEEE Trans. Inf. Theory*, vol. 45, no. 1, pp. 1757–1767, Mar. 1999.

[8] R. Koetter and A. Vardy, "Algebraic soft-decision decoding of Reed–Solomon codes," *IEEE Trans. Inf. Theory*, vol. 49, no. 11, pp. 2809–2825, Nov. 2003.

[9] Y. Wu, "New list decoding algorithms for Reed–Solomon and BCH codes," *IEEE Trans. Inf. Theory*, vol. 54, no. 8, pp. 3611–3630, Aug. 2008.

[10] D. Chase, "Class of algorithms for decoding block codes with channel measurement information," *IEEE Trans. Inf. Theory*, vol. IT-18, no. 1, pp. 170–182, Jan. 1972.

[11] B. Dorsch, "A decoding algorithm for binary block codes and J-ary output channels," *IEEE Trans. Inf. Theory*, vol. IT-20, no. 3, pp. 391–394, May 1974.

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

[13] X. Zhang, Y. Zheng, and Y. Wu, "A chase-type Koetter–Vardy algorithm for soft-decision Reed–Solomon decoding," in *Proc. Int. Conf. Comput., Netw. Commun. (ICNC)*, Jan. 2012, pp. 466–470.

[14] X. Zhang and Y. Zheng, "Generalized backward interpolation for algebraic soft-decision decoding of Reed–Solomon codes," *IEEE Trans. Commun.*, vol. 61, no. 1, pp. 13–23, Jan. 2013.

[15] J. Xing, L. Chen, and M. Bossert, "Low-complexity chase decoding of Reed–Solomon codes using module," *IEEE Trans. Commun.*, vol. 68, no. 10, pp. 6012–6022, Oct. 2020.

[16] Y. Shany and A. Berman, "A Gröbner-bases approach to syndrome-based fast chase decoding of Reed–Solomon codes," *IEEE Trans. Inf. Theory*, vol. 68, no. 4, pp. 2300–2318, Apr. 2022.

[17] C. Yue, M. Shirvanimoghaddam, B. Vucetic, and Y. Li, "A revisit to ordered statistics decoding: Distance distribution and decoding rules," *IEEE Trans. Inf. Theory*, vol. 67, no. 7, pp. 4288–4337, Jul. 2021.

[18] R. G. Gallager, "Low-density parity-check codes," *IRE Trans. Inf. Theory*, vol. 8, no. 1, pp. 21–28, Jan. 1962.

[19] J. Jiang and K. R. Narayanan, "Iterative soft decoding of Reed–Solomon codes," *IEEE Commun. Lett.*, vol. 8, no. 4, pp. 244–246, Apr. 2004.

[20] J. Jiang and K. R. Narayanan, "Iterative soft-input soft-output decoding of Reed–Solomon codes by adapting the parity-check matrix," *IEEE Trans. Inf. Theory*, vol. 52, no. 8, pp. 3746–3756, Aug. 2006.

[21] M. El-Khamy and R. J. McEliece, "Iterative algebraic soft-decision list decoding of Reed–Solomon codes," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 3, pp. 481–490, Mar. 2006.

[22] L. Deng et al., "Perturbed adaptive belief propagation decoding for high-density parity-check codes," *IEEE Trans. Commun.*, vol. 69, no. 4, pp. 2065–2079, Apr. 2021.

[23] Y. Jing, W. Zhang, H. Wang, Y. Chang, and Y. Liu, "Improved adaptive belief propagation decoding of Reed–Solomon codes with SPC codes," *IEEE Commun. Lett.*, vol. 26, no. 7, pp. 1464–1468, Jul. 2022.

[24] T. R. Halford and K. M. Chugg, "Random redundant iterative soft-in soft-out decoding," *IEEE Trans. Commun.*, vol. 56, no. 4, pp. 513–517, Apr. 2008.

[25] T. Hehn, J. B. Huber, O. Milenkovic, and S. Laendner, "Multiple-bases belief-propagation decoding of high-density cyclic codes," *IEEE Trans. Commun.*, vol. 58, no. 1, pp. 1–8, Jan. 2010.

[26] I. Dimnik and Y. Be'ery, "Improved random redundant iterative HDPC decoding," *IEEE Trans. Commun.*, vol. 57, no. 7, pp. 1982–1985, Jul. 2009.

[27] M. Bossert and F. Hergert, "Hard- and soft-decision decoding beyond the half minimum distance—An algorithm for linear codes (Corresp.)," *IEEE Trans. Inf. Theory*, vol. IT-32, no. 5, pp. 709–714, Sep. 1986.

[28] E. Santi, C. Hager, and H. D. Pfister, "Decoding Reed–Muller codes using minimum-weight parity checks," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2018, pp. 1296–1300.

[29] M. Bossert, "An iterative hard and soft decision decoding algorithm for cyclic codes," in *Proc. 12th Int. ITG Conf. Syst. Commun. Coding (SCC)*, Rostock, Germany, Feb. 2019, pp. 263–268.

[30] M. Bossert, R. Schulz, and S. Bitzer, "On hard and soft decision decoding of BCH codes," *IEEE Trans. Inf. Theory*, vol. 68, no. 11, pp. 7107–7124, Nov. 2022.

[31] X. Chen and M. Ye, "Cyclically equivariant neural decoders for cyclic codes," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Jul. 2021, pp. 1771–1780.

[32] X. Chen and M. Ye, "Improving the list decoding version of the cyclically equivariant neural decoder," in *Proc. IEEE Int. Symp. Inform. Theory (ISIT)*, Espoo, Finland, Jun. 2022, pp. 2344–2349.

[33] R. Koetter, "On algebraic decoding of algebraic-geometric and cyclic codes," Ph.D. dissertation, Dept. Elect. Eng., Univ. Linköping, Linköping, Sweden, 1996.

[34] K. Lee and M. O'Sullivan, "List decoding of Reed–Solomon codes from a Gröbner basis perspective," *J. Symb. Comput.*, vol. 43, no. 9, pp. 645–658, Sep. 2008.
[35] S. Wang, Q. Huang, and Z. Wang, "Symbol flipping decoding algorithms based on prediction for non-binary LDPC codes," *IEEE Trans. Commun.*, vol. 65, no. 5, pp. 1913–1924, May 2017.
[36] P. J. Lee and E. F. Brickell, "An observation on the security of McEliece's public-key cryptosystem," in *Proc. Workshop Theory Appl. Crypto. Tech.*, Davos, Switzerland, May 1988, pp. 275–280.
[37] G. Polya and R. C. Read, *Combinatorial Enumeration of Groups, Graphs, and Chemical Compounds*. Berlin, Germany: Springer, 2012.
[38] M. El-Khamy and R. J. McEliece, "Bounds on the average binary minimum distance and the maximum likelihood performance of Reed–Solomon codes," in *Proc. 42nd Allerton Conf. Commun. Control Comput.*, Sep. 2004, pp. 290–299.

Jiongyue Xing received the B.Sc. degree in communication engineering and the Ph.D. degree in information and communication engineering from Sun Yat-sen University, Guangzhou, China, in 2015 and 2020, respectively. From December 2018 to December 2019, he was a Visiting Ph.D. Student with the Institute of Communication Engineering, Ulm University, Germany. Since 2020, he has been a Researcher with the Hong Kong Theory Laboratory, Central Research Institute, 2012 Laboratories, Huawei Technology Company Ltd. He received the Outstanding Doctoral Dissertation Award from the Chinese Institute of Electronics Information Theory Society in 2020. His research interests include channel coding and data communications.

Martin Bossert (Fellow, IEEE) received the Dipl.-Ing. degree in electrical engineering from the Technical University of Karlsruhe, Germany, in 1981, and the Ph.D. degree from the Technical University of Darmstadt, Germany, in 1987. After a one-year DFG scholarship with Link-Ping University, Sweden, he joined AEG Mobile Communication, where he was involved in the specification and development of the GSM system. Since 1993, he has been a Professor with Ulm University, Germany. He is currently a Senior Professor with the Institute of Communications Engineering. He is the author of several textbooks and the coauthor of more than 200 papers. He has been a member of the IEEE Information Theory Society Board of Governors from 2010 to 2012 and he has been appointed as a member of the German National Academy of Sciences (Leopoldina) in 2013. Among other awards and honors, he received the Vodafone Innovationspreis in 2007. His research interests include reliable and secure data transmission. His main focus is on decoding of codes with reliability information and coded modulation.

Li Chen (Senior Member, IEEE) received the B.Sc. degree in applied physics from Jinan University, Guangzhou, China, in 2003, and the M.Sc. degree in communications and signal processing and the Ph.D. degree in communications engineering from Newcastle University, U.K., in 2004 and 2008, respectively. From 2007 to 2010, he was a Research Associate with Newcastle University. In 2010, he returned to China as a Lecturer with the School of Information Science and Technology, Sun Yat-sen University, Guangzhou. From 2011 to 2012, he was a Visiting Researcher with the Institute of Network Coding, The Chinese University of Hong Kong, where he was an Associate Professor and a Professor from 2011 and 2016. Since 2013, he has been the Associate Head of the Department of Electronic and Communication Engineering (ECE). From July 2015 to October 2015, he was a Visitor with the Institute of Communications Engineering, Ulm University, Germany. From October 2015 to June 2016, he was a Visiting Associate Professor with the Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN, USA. From 2017 to 2020, he was the Deputy Dean of the School of Electronics and Communication Engineering. His research interests include information theory, error-correction codes, and data communications. He is a Senior Member of the Chinese Institute of Electronics (CIE). He is a member of the IEEE Information Theory Society Board of Governors and its External Nomination Committee and the Chair of its Conference Committee. He is also the Chair of the IEEE Information Theory Society Guangzhou Chapter. He has been organizing several international conferences and workshops, including the 2018 IEEE Information Theory Workshop (ITW) at Guangzhou and the 2022 IEEE East Asian School of Information Theory (EASIT) at Shenzhen, for which he is the General Co-Chair. He is also the TPC Co-Chair of 2022 IEEE/CIC International Conference on Communications in China (ICCC) with Foshan. He is an Associate Editor of IEEE TRANSACTIONS ON COMMUNICATIONS.

Jiasheng Yuan received the B.Sc. degree in electronic information science and technology and the M.Sc. degree in communication and information systems from Sun Yat-sen University, Guangzhou, China, in 2019 and 2021, respectively. His research interests include coding theory and data communications.

Sebastian Bitzer (Graduate Student Member, IEEE) received the B.Sc. and M.Sc. degrees in electrical engineering from Ulm University, Germany, in 2018 and 2021, respectively. He is currently pursuing the Ph.D. degree with the Coding and Cryptography Group, Institute of Communications Engineering, Technical University of Munich (TUM), under the supervision of Prof. Wachter-Zeh. His research interests include coding theory and cryptography.