Iterative Decoding of Non-Binary Cyclic Codes Using Minimum-Weight Dual Codewords

Jiongyue Xing †, Martin Bossert ‡, Sebastian Bitzer ‡, Li Chen †
† School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou, China
‡ Institute of Communications Engineering, Ulm University, Ulm, Germany
Email: xingjyue@mail2.sysu.edu.cn, {martin.bossert, sebastian.bitzer}@uni-ulm.de, chenli55@mail.sysu.edu.cn

***Abstract*—This paper proposes a novel shift-sum decoding scheme for non-binary cyclic codes. Using minimum-weight dual codewords and their cyclic shifts, a reliability measure can be yielded as an indicator for the error position and the error magnitude. Based on this shift-sum decoding concept, a hard-decision iterative decoding algorithm is proposed, which can correct errors beyond half of the code's minimum Hamming distance. By utilizing reliability information from the channel, a soft-decision iterative decoding algorithm is further introduced to improve the decoding performance. These two shift-sum based iterative decoding algorithms are realized with polynomial multiplication and integer (or real number) comparisons, which are hardware-friendly. Simulation results on Reed-Solomon codes and non-binary BCH codes show the decoding potential of the proposed algorithms.**

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

I. Introduction

Cyclic codes, e.g., Reed-Solomon (RS) codes and BCH codes, are widely applied for data transmissions due to their simple encoding and efficient decoding algorithms. Currently, the Berlekamp-Massey (BM) algorithm [1] is applied in practice, which can correct errors up to half of the code's minimum Hamming distance. The interpolation based algebraic list decoding algorithm, i.e., the so-called Guruswami-Sudan (GS) algorithm [2] can correct errors beyond the above distance bound. Its soft-decision advancement was proposed in [3]. By utilizing the BM output, Wu further proposed an improved list decoding algorithm for RS and BCH codes [4], which exhibits a lower complexity than the GS algorithm [2]. However, their complexity remains high, limiting their practical applications. Utilizing soft information, Chase decoding [5], information set decoding [6] and order statistics decoding [7] generate a number of decoding trials, yielding a better performance with a moderate complexity. By adapting or extending the parity-check matrix of the code, several belief-propagation based algorithms have also been proposed to improve the decoding performance of cyclic codes [8]–[10].

Recently, it has been shown that near maximum-likelihood (ML) decoding performance for Reed-Muller and BCH codes can be achieved by decoding with a large number of minimum-weight dual codewords (MWDCs) [11]–[13]. In [12] and [13], a novel concept of shift-sum decoding of cyclic codes was proposed. With cyclically different MWDCs and their proper shifts, a reliability measure can be generated and utilized as the basis for various decoding algorithms. In this paper, we extend the shift-sum decoding to the non-binary cyclic codes, e.g., RS and non-binary BCH (NB-BCH) codes. Based on this concept, a hard-decision iterative shift-sum (HISS) algorithm is proposed, showing its error-correction capability beyond half of the code's minimum Hamming distance. With received soft information, a soft-decision iterative shift-sum (SISS) algorithm is further introduced to improve the decoding performance. It should be pointed out that the HISS algorithm is realized with polynomial multiplication and integer comparisons, while the SISS algorithm replaces the latter with real number comparisons. They are well suited for hardware implementation. Simulation results on RS and NB-BCH codes show that the proposed algorithms outperform the bounded minimum-distance decoding, e.g., the BM algorithm [1], with a significant coding gain. Moreover, the HISS algorithm achieves the same decoding performance as the GS algorithm [2] for RS codes. So far, decoding of NB-BCH codes has been sparsely reported in literature. This work provides some new performance insights for the community.

II. Prerequisites

Let $\mathbb{F}_q = \{\sigma_0, \sigma_1, \ldots, \sigma_{q-1}\}$ denote a finite field of size q with a primitive element α, and $\mathbb{F}_q[x]$ denote the univariate polynomial ring defined over $\mathbb{F}_q$. Note that σ_0 is set to the zero element. For simplicity, we restrict to binary extension fields in this paper, i.e., $q = 2^s$ and $s \in \mathbb{Z}^+$. Furthermore, we only consider codes with length $n = 2^s - 1$. Let $\mathcal{C}(2^p; n, k, d)$ be a cyclic code defined over $\mathbb{F}_{2^p}$ with dimension k and minimum distance d, where $p = 1, 2, \ldots, s$. When $p = 1$, $\mathcal{C}$ is a binary BCH code. For $p = s$, $\mathcal{C}$ 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 $\underline{c} = (c_0, c_1, \ldots, c_{n-1}) \in \mathcal{C}(2^p; n, k, d)$ denote a codeword. It can also be written as a polynomial $c(x) = c_0 + c_1 x + \cdots + c_{n-1}x^{n-1}$. Without ambiguity, we use both of the representations to denote a codeword in the following. The support of $\underline{c}$ (or $c(x)$) is defined as $\mathrm{sup}(\underline{c}) = \mathrm{sup}(c(x)) = \{j \mid c_j \neq 0, \forall j\}$. The weight of $\underline{c}$ (or $c(x)$) is $\mathrm{wt}(\underline{c}) = \mathrm{wt}(c(x)) = |\mathrm{sup}(c(x))|$.

Definition I. Given codewords $c_1(x), c_2(x) \in \mathcal{C}$, they are cyclically different if $c_2(x) \neq \alpha^j c_1(x) x^{-h} \mod (x^n - 1), \forall j$ and $\forall h \in \mathrm{sup}(c_1(x))$.

Encoding of cyclic codes is defined by its generator polynomial $g(x)$. Given a message polynomial $f(x) = f_0 + f_1 x +$

$\cdots + f_{k-1}x^{k-1} \in \mathbb{F}_{2^p}[x]$, codeword $c(x)$ is generated by

$$c(x) = f(x)g(x). \tag{1}$$

In this paper, we consider two typical non-binary cyclic codes, 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 $g_{\text{RS}}(x)$ can be defined as

$$g_{\text{RS}}(x) = \prod_{j=1}^{d_{\text{RS}}-1} (x - \alpha^j). \tag{2}$$

The 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. Let the cyclotomic cosets be $K_j = \{j \cdot (2^p)^i \mod n, i = 0, 1, \ldots, \frac{s}{p} - 1\}$, where j is the smallest number in K_j. Although we can get n cosets for $j = 0, 1, \ldots, n-1$, two cosets are either identical or disjoint. The cardinality of K_j satisfies $|K_j| \leq \frac{s}{p}$. Now the generator polynomial $g_{\text{BCH}}(x)$ can be defined as

$$g_{\text{BCH}}(x) = \prod_{i \in \mathcal{K}} (x - \alpha^i), \tag{3}$$

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

III. The Shift-Sum Decoding

In this section, we introduce the basic idea of the shift-sum decoding for the non-binary cyclic codes. The shift-sum process utilizes a number of cyclically different MWDCs to determine the error positions and the error magnitudes.

Assume that τ errors have occurred at the positions $e_1, e_2, \ldots, e_\tau$ in the n transmitted codeword symbols c_j and the error symbols are from nonzero elements of $\mathbb{F}_{2^p}$. We can denote this error polynomial by

$$\varepsilon(x) = \varepsilon_{e_1} x^{e_1} + \varepsilon_{e_2} x^{e_2} + \cdots + \varepsilon_{e_\tau} x^{e_\tau}. \tag{4}$$

Hence, $r(x) = c(x) + \varepsilon(x)$ is the received polynomial.

Let $b(x) = \beta_{b_1} x^{b_1} + \beta_{b_2} x^{b_2} + \cdots + \beta_{b_{d^\perp}} x^{b_{d^\perp}}$ denote a codeword of dual code with weight $d^\perp$. Since the dual code is also linear and cyclic, we can assume $\beta_{b_1} = 1$ and $b_1 = 0$, i.e., $b(x) = 1 + \beta_{b_2} x^{b_2} + \cdots + \beta_{b_{d^\perp}} x^{b_{d^\perp}}$. The support of this polynomial is $\sup(b(x)) = \{b_1, b_2, \ldots, b_{d^\perp}\}$. Let polynomial $w(x)$ be the multiplication of the dual codeword $b(x)$ with the received polynomial $r(x)$, i.e.,

$$\begin{aligned} w(x) &= r(x)b(x) \\ &= (c(x) + \varepsilon(x))b(x) \\ &= \varepsilon(x)b(x) \mod (x^n - 1), \end{aligned} \tag{5}$$

where $c(x)b(x) = 0 \mod (x^n - 1)$. Polynomial $w(x)$ can be further elaborated as

$$\begin{aligned} w(x) =& \beta_{b_1} x^{b_1} \varepsilon(x) + \cdots + \beta_{b_{d^\perp}} x^{b_{d^\perp}} \varepsilon(x) \mod (x^n - 1) \\ =& \varepsilon_{e_1} x^{e_1} + \varepsilon_{e_2} x^{e_2} + \cdots + \varepsilon_{e_\tau} x^{e_\tau} + \\ & \beta_{b_2} \varepsilon_{e_1} x^{e_1 + b_2} + \cdots + \beta_{b_2} \varepsilon_{e_\tau} x^{e_\tau + b_2} + \\ & \vdots \\ & \beta_{b_{d^\perp}} \varepsilon_{e_1} x^{e_1 + b_{d^\perp}} + \cdots + \beta_{b_{d^\perp}} \varepsilon_{e_\tau} x^{e_\tau + b_{d^\perp}}, \end{aligned} \tag{6}$$

where the exponents are calculated $\mod n$. It can be seen that any non-zero coefficient of polynomial $w(x)$ is an error (at its original position) or a shifted scalar error. We can shift the non-zero coefficients of $w(x)$ (which are shifted scalar errors) back to their original positions by multiplying $w(x)$ with $\frac{x^{-h}}{\beta_h}$, where $h \in \{b_2, b_3, \ldots, b_{d^\perp}\}$ denotes the shift. Therefore, $d^\perp$ polynomials $\frac{x^{-h}}{\beta_h} w(x)$ can be obtained, denoted as $w_h(x), \forall h \in \sup(b(x))$. Note that $w_0(x) = w(x)$.

Assume we have L cyclically different dual codewords $b^{(\ell)}(x) = 1 + \beta_{b_2}^{(\ell)} x^{b_2} + \cdots + \beta_{b_{d^\perp}}^{(\ell)} x^{b_{d^\perp}}$ of weight $d^\perp$, where $\ell = 1, 2, \ldots, L$. Each of them can produce a polynomial $w^{(\ell)}(x)$ by

$$w^{(\ell)}(x) = r(x)b^{(\ell)}(x) = \varepsilon(x)b^{(\ell)}(x) \mod (x^n - 1). \tag{7}$$

With $d^\perp$ cyclic shifts, $Ld^\perp$ polynomials can be yielded, i.e.,

$$w_h^{(\ell)}(x) = \frac{x^{-h}}{\beta_h^{(\ell)}} r(x)b^{(\ell)}(x) \mod (x^n - 1), \tag{8}$$

where $h \in \sup(b^{(\ell)}(x))$. For each $w_h^{(\ell)}(x)$, its coefficient $w_{h,j}^{(\ell)}$ can be written as

$$w_{h,j}^{(\ell)} = \frac{1}{\beta_h^{(\ell)}} \sum_{u \in \sup(b^{(\ell)}(x))} \beta_u^{(\ell)} r_{(j+h-u) \mod n}, \tag{9}$$

where $j = 0, 1, \ldots, n-1$. From (6), we know that value of $w_{h,j}^{(\ell)}$ is an indicator for the error position and the error magnitude. In order to characterize the value of $w_{h,j}^{(\ell)}$, we define the following function as

$$T(\ell, i, j, h) = \begin{cases} 1, & \text{if } w_{h,j}^{(\ell)} = \sigma_i, \\ 0, & \text{otherwise}, \end{cases} \tag{10}$$

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

$$\phi_{i,j} = \sum_{\ell=1}^{L} \sum_{h \in \sup(b^{(\ell)}(x))} T(\ell, i, j, h). \tag{11}$$

Note that for each j, summation of $\phi_{i,j}$ is a constant, i.e.,

$$\sum_{i=0}^{2^p - 1} \phi_{i,j} = Ld^\perp. \tag{12}$$

A larger value of $\phi_{i,j}$ except $\phi_{0,j}$ implies that an error value σ_i is more likely to occur at position j. Therefore, $\phi_{i,j}$ can be considered as a reliability measure for the erroneous or the non-erroneous positions. The following example shows the

property of $\phi_{i,j}$.

Example 1. Given an RS code $\mathcal{C}(8;7,3,5)$ [1], assume codeword $c(x) = \alpha^6 + \alpha^4 x + \alpha^4 x^2 + \alpha^3 x^3 + \alpha^6 x^5 + \alpha^3 x^6$ is transmitted and $r(x) = \alpha^6 + \alpha^5 x + \alpha^4 x^2 + \alpha^3 x^3 + \alpha^3 x^6$ is received. There are 5 cyclically different minimum-weight codewords of dual code $\mathcal{C}^{\perp}(8;7,4,4)$, which are

$$b^{(1)}(x) = 1 + \alpha x + \alpha^5 x^2 + \alpha^2 x^6,$$
$$b^{(2)}(x) = 1 + x + x^3 + x^6,$$
$$b^{(3)}(x) = 1 + \alpha^2 x^2 + \alpha^5 x^3 + \alpha x^6,$$
$$b^{(4)}(x) = 1 + \alpha^2 x + \alpha^4 x^2 + \alpha^3 x^5,$$
$$b^{(5)}(x) = 1 + \alpha^3 x^2 + \alpha^5 x^4 + \alpha^6 x^6.$$

After performing the shift-sum decoding, we have

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

where $\phi_{i,j}$ is an entry at row-i column-j. Note that $\sum_{i=0}^{7} \phi_{i,j} = Ld^{\perp} = 20, \forall j$. It can be seen that $\phi_{1,1}$ and $\phi_{5,5}$ have the maximum value 10. 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. Therefore, the error polynomial is $\varepsilon(x) = x + \alpha^6 x^5$.□

In order to ensure the accuracy of the determined error positions and magnitudes, a large number of cyclically different MWDCs are necessary. In this paper, we utilize the Lee-Brickell algorithm [14] to formulate a heuristic searching scheme. By randomly generating an error vector $\underline{\varepsilon}$ of weight $\leq d^{\perp}$, the Lee-Brickell algorithm seeks a codeword whose Hamming distance to $\underline{\varepsilon}$ is minimal. If a codeword is found, we check whether its weight is $d^{\perp}$ and it is cyclically different from the earlier found codewords. The process continues until a large number of the cyclically different MWDCs are found.

IV. The Iterative Shift-Sum Decoding Algorithms

This section proposes two decoding algorithms based on the above shift-sum approach, the HISS and the SISS algorithms. Based on $\phi_{i,j}$, the algorithms determine several positions and magnitudes to update the received $r(x)$. If $r(x)$ is not a codeword, recalculate $\phi_{i,j}$ and update $r(x)$ again. This process will be iteratively performed until a codeword is found.

A. The HISS Algorithm

Based on the shift-sum decoding, the reliability measure $\phi_{i,j}$ is obtained. Since $\sigma_0 = 0$, $w_{h,j}^{(\ell)} = \sigma_0$ indicates position j is correct and vice versa. Based on (12), a smaller $\phi_{0,j}$ implies a larger $\sum_{i=1}^{2^p-1} \phi_{i,j}$, i.e., more original errors and shifted scalar errors occur at position j. Therefore, r_j is more likely an erroneous symbol. The HISS algorithm will modify $r(x)$ by determining several positions which are more likely to be erroneous at each iteration. Firstly, we sort $\phi_{0,0}, \phi_{0,1}, \ldots, \phi_{0,n-1}$ in an ascending order to yield a new sequence $j_0^{(1)}, j_1^{(1)}, \ldots, j_{n-1}^{(1)}$ such that

$$\phi_{0,j_0^{(1)}} \leq \phi_{0,j_1^{(1)}} \leq \cdots \leq \phi_{0,j_{n-1}^{(1)}}. \tag{13}$$

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

$$\varphi_{j_0^{(2)}} \geq \varphi_{j_1^{(2)}} \geq \cdots \geq \varphi_{j_{n-1}^{(2)}}. \tag{14}$$

Afterwards, we define two index sets as $\Lambda^{(1)} = \{j_0^{(1)}, j_1^{(1)}, \ldots, j_{\lambda-1}^{(1)}\}$ and $\Lambda^{(2)} = \{j_0^{(2)}, j_1^{(2)}, \ldots, j_{\lambda-1}^{(2)}\}$, where λ is a predefined parameter to determine the number of updated positions at each iteration. Further let $\Lambda = \Lambda^{(1)} \cap \Lambda^{(2)}$ denote the index set of the updated positions. The corresponding updated polynomial is defined by

$$\gamma(x) = \sum_{j \in \Lambda} \gamma_j x^j, \tag{15}$$

where $\gamma(x) \in \mathbb{F}_{2^p}(x)$. Now the received polynomial $r(x)$ can be refined by $r(x) \leftarrow r(x) + \gamma(x)$. If $r(x) \in \mathcal{C}(2^p; n, k, d)$, a codeword is found, and the HISS algorithm will terminate and output $r(x)$. Otherwise, perform the shift-sum decoding to recalculate $\phi_{i,j}$ and find the new 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.

Algorithm 1 The HISS Algorithm

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

For $I = 1$ to $I_{\max}$
  Initialize $\phi_{i,j} = 0, \forall (i,j)$;
  For $\ell = 1$ to L
    For $j = 0$ to $n-1$
      For $h \in \sup(b^{(\ell)}(x))$ **do**
        Determine $w_{h,j}^{(\ell)}$ as in (9);
        Determine $\phi_{i,j}$ as in (10) (11);
      End For
    End For
  End For
  Determine Λ and $\gamma(x)$;
  Refine $r(x) \leftarrow r(x) + \gamma(x)$;
  If $r(x) \in \mathcal{C}(2^p; n, k, d)$, terminate and output $r(x)$;
End for

Remark 1. The HISS algorithm needs polynomial multiplication and integer comparisons, which is of practical interest.

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

B. The SISS Algorithm

The SISS algorithm utilizes soft information from the channel, in which the reliability measure is defined for $w_{h,j}^{(\ell)}$. Assume codeword $\underline{c} = (c_0, c_1, \ldots, c_{n-1})$ is transmitted through a memoryless channel and $\underline{\mathsf{r}} = (\mathsf{r}_0, \mathsf{r}_1, \ldots, \mathsf{r}_{n-1}) \in \mathbb{R}^n$ is the received symbol vector. By assuming $\Pr[c_j = \sigma_i] = \frac{1}{2^p}$, an *a posteriori* probability matrix $\mathbf{\Pi} \in \mathbb{R}^{2^p \times n}$ with entries $\pi_{i,j} = \Pr[c_j = \sigma_i \mid \mathsf{r}_j]$ can be observed, where $0 \le i \le 2^p - 1$ and $0 \le j \le n-1$. Note that $\sum_i \pi_{i,j} = 1, \forall j$. Let π_j^{I} denote the largest value of each column j such that the reliability of each hard-decision received symbol r_j can be defined by

$$y_j = \frac{\pi_j^{\mathrm{I}}}{1 - \pi_j^{\mathrm{I}}}. \tag{16}$$

Now we can define the reliability measure for $w_{h,j}^{(\ell)}$ as

$$\zeta_{h,j}^{(\ell)} = \min_{\substack{u \in \sup(b^{(\ell)}(x)) \\ j \neq (j+h-u) \mod n}} y_{(j+h-u) \mod n}. \tag{17}$$

Note that $j \neq (j+h-u) \mod n$ can be simplified as $h \neq u$. With the same MWDCs $b^{(\ell)}(x)$, we calculate $\phi_{i,j}$ as in (11), while (10) is redefined by

$$T(\ell, i, j, h) = \begin{cases} \zeta_{h,j}^{(\ell)}, & \text{if } w_{h,j}^{(\ell)} = \sigma_i, \\ 0, & \text{otherwise.} \end{cases} \tag{18}$$

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 polynomial $r(x)$. With the slight modification of $T(\ell, i, j, h)$, the SISS algorithm can yield a significant improvement, as the following simulation results show.

Remark 2. The SISS algorithm is realized with polynomial multiplication and real number comparisons, which is also hardware-friendly.

V. Simulation Results

This section presents simulation results of the proposed algorithms over the Q-ary symmetric channel (QSC) with error probability ρ and the additive white Gaussian noise (AWGN) channel with BPSK modulation. The HISS and the SISS algorithms with the maximum iteration number $I_{\max}$ are denoted as HISS $(I_{\max})$ and SISS $(I_{\max})$, respectively.

A. The Q-ary Symmetric Channel

In the QSC, performance of the HISS algorithm can be obtained in a semi-analytical manner. Let $M(\tau)$ denote the number of simulated events with τ errors and $F(\tau)$ denote the number of decoding failure among $M(\tau)$ events. With this information, the word error rate (WER) for decoding in case of a QSC with ρ is

$$\mathrm{WER}(\rho) = \sum_{\tau=1}^{n} \frac{F(\tau)}{M(\tau)} \binom{n}{\tau} \rho^{\tau} (1-\rho)^{n-\tau}. \tag{19}$$

Note that we can only simulate several significant values of τ to obtain the WER since the small-weight (or large-weight) errors are obviously correctable (or uncorrectable). This reduces the simulation time.

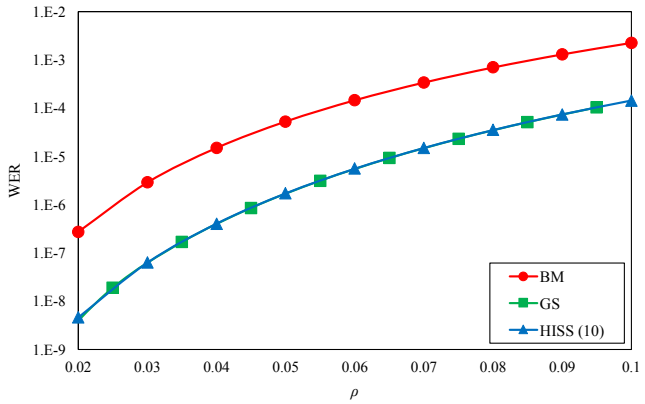

Fig. 1. WER for the RS code $\mathcal{C}(16; 15, 5, 11)$ versus error probability ρ.

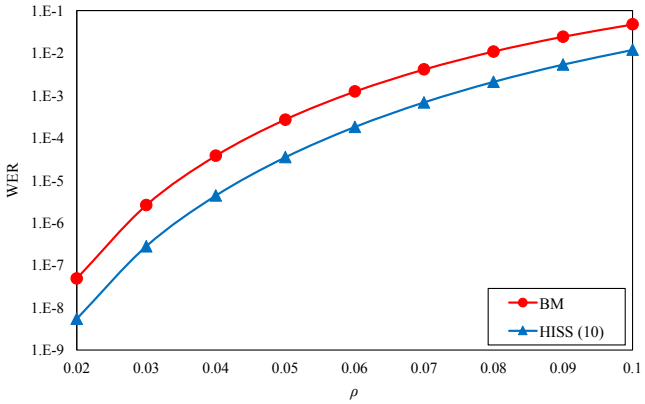

Fig. 2. WER for the NB-BCH code $\mathcal{C}(4; 63, 27, 21)$ versus error probability ρ.

Figs. 1 and 2 show the WER for the RS code $\mathcal{C}(16; 15, 5, 11)$ and the NB-BCH code $\mathcal{C}(4; 63, 27, 21)$, respectively. Note that $I_{\max}$ is set to 10 for the HISS algorithm. For the RS code, the BM algorithm can correct errors up to half of the minimum distance, i.e., five symbol errors. In the HISS algorithm, we use $L = 335$ cyclically different dual codewords of weight $d^{\perp} = 6$. Fig. 1 shows that the HISS algorithm performs the same as the GS list decoding algorithm [2], outperforming the BM algorithm by a factor of 100 in the WER. Note that the GS implementation was from [15], and when the output list contained several candidates with the same distance to $r(x)$, a random one was selected. For each iteration of the HISS algorithm, it needs $Ln(d^{\perp})^2$ finite field multiplications. While the GS decoding complexity is $O(m^6 n^3)$, where m is the interpolation multiplicity [2]. Therefore, the HISS algorithm is less complex than the GS algorithm. For the NB-BCH code, the BM algorithm can correct up to 10 symbol errors. We have found 183 cyclically different dual codewords of weight $d^{\perp} = 14$. Fig. 2 shows that the proposed algorithm performs better than the BM algorithm by a factor of nearly 10. These results demonstrate that the HISS algorithm can correct errors beyond the half distance bound.

B. The AWGN Channel

Fig. 3 shows decoding performance of the proposed algorithms for the RS code $\mathcal{C}(16; 15, 5, 11)$. The upper bound and lower bound of the ML decoding [16], denoted as MLUB and

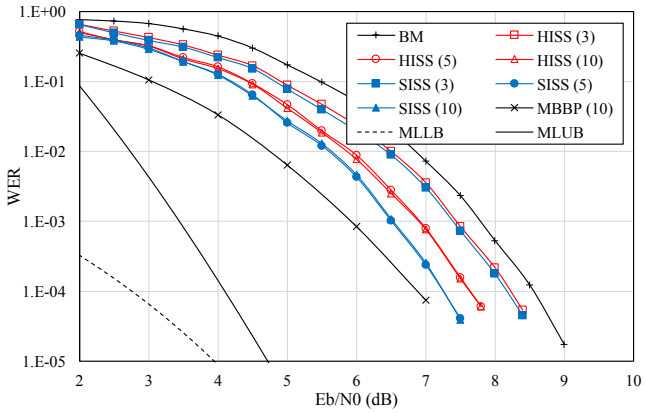

Fig. 3. Performance of the RS code $\mathcal{C}(16; 15, 5, 11)$ over the AWGN channel.

MLLB, respectively, are also shown for comparison. As $I_{\max}$ increases, performance of the HISS and the SISS algorithms improve, outperforming the BM algorithm. When $I_{\max} = 10$, the HISS algorithm yields a gain of 0.9 dB at the WER of 10^{-4}. By utilizing soft information from the channel, the SISS algorithm performs better than its hard-decision counterpart, obtaining an extra 0.3 dB performance gain. But it is still 3.1 dB away from MLUB at the WER of 10^{-4}. It can also be observed that the HISS (or SISS) algorithm with five iterations exhibits a similar performance as that with ten iterations, implying that most of the errors can be corrected at the first few iterations. Although the multiple-bases belief-propagation (MBBP) algorithm [10] with 10 iterations performs better than the proposed algorithms, it requires a significantly higher decoding complexity.

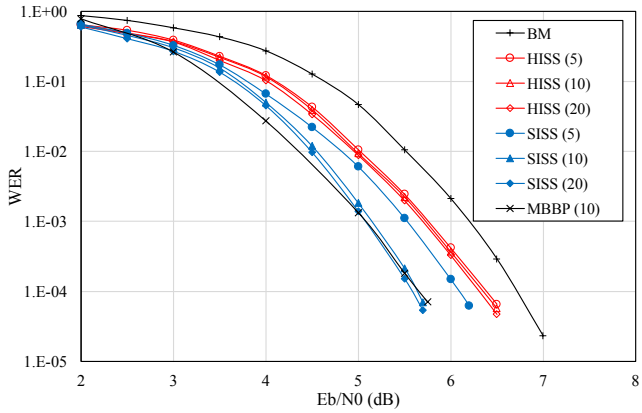

Fig. 4. Performance of the NB-BCH code $\mathcal{C}(4; 63, 27, 21)$ over the AWGN channel.

Fig. 4 shows decoding performance of the NB-BCH code $\mathcal{C}(4; 63, 27, 21)$. Again, their performance improve as $I_{\max}$ enlarges. When $I_{\max} = 20$, the HISS and the SISS algorithms yield a better performance than the BM algorithm with a gain of 0.4 dB and 1.2 dB at the WER of 10^{-4}, respectively. It can be seen that the soft-decision decoding improvement of the NB-BCH code is larger than that of the RS code. This is because the symbol reliability y_j is more precise for the NB-BCH code whose symbols are defined over a smaller finite field. For this code, the SISS algorithm performs similarly as the MBBP algorithm with a much lower complexity.

VI. Conclusion

This paper has investigated the shift-sum decoding of non-binary cyclic codes. By multiplying the MWDCs with the received polynomial, a reliability measure can be yielded as an indicator for the error positions and the error magnitudes. The HISS and the SISS algorithms have been further proposed to show the performance potential of the novel shift-sum decoding concept, outperforming the conventional BM algorithm. Moreover, they can be realized with only polynomial multiplication and some integer (or real number) comparisons, which are of practical interest. Simulation results on the RS and the NB-BCH codes have been provided to validate the decoding capability of the proposed algorithms.

Acknowledgement

This work is sponsored by National Natural Science Foundation of China (NSFC) with project ID 61671486 and International Program for Ph.D. Candidates, Sun Yat-sen University.

References

[1] J. Massey, "Shift register synthesis and BCH decoding," *IEEE Trans. Inform. Theory*, vol. 15, no. 1, pp. 122–127, Jan. 1969.
[2] V. Guruswami and M. Sudan, "Improved decoding of Reed-Solomon and algebraic-geometric codes," *IEEE Trans. Inform. Theory*, vol. 45, no. 6, pp. 1757–1767, Sept. 1999.
[3] R. Koetter and A. Vardy, "Algebraic soft-decision decoding of Reed-Solomon codes," *IEEE Trans. Inform. Theory*, vol. 49, no. 11, pp. 2809–2825, Nov. 2003.
[4] Y. Wu, "New list decoding algorithms for Reed-Solomon and BCH codes," *IEEE Trans. Inform. Theory*, vol. 54, no. 8, pp. 3611–3630, Aug. 2008.
[5] D. Chase, "Class of algorithms for decoding block codes with channel measurement information," *IEEE Trans. Inform. Theory*, vol. 18, no. 1, pp. 170–182, Jan. 1972.
[6] B. Dorsch, "A decoding algorithm for binary block codes and j-ary output channels," *IEEE Trans. Inform. Theory*, vol. 20, no. 3, pp. 391–394, May 1974.
[7] M. P. Fossorier and S. Lin, "Soft-decision decoding of linear block codes based on ordered statistics," *IEEE Trans. Inform. Theory*, vol. 41, no. 5, pp. 1379–1396, Sept. 1995.
[8] J. Jiang and K. R. Narayanan, "Iterative soft-input soft-output decoding of Reed-Solomon codes by adapting the parity-check matrix," *IEEE Trans. Inform. Theory*, vol. 52, no. 8, pp. 3746–3756, Aug. 2006.
[9] I. Dimnik and Y. Be'ery, "Improved random redundant iterative HDPC decoding," *IEEE Trans. Commun.*, vol. 57, no. 7, pp. 1982–1985, Jul. 2009.
[10] 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.
[11] E. Santi, C. Hager, and H. D. Pfister, "Decoding Reed-Muller codes using minimum-weight parity checks," in *Proc. IEEE Int. Symp. Inform. Theory (ISIT)*, Vail, U.S.A, pp. 1296–1300, Jun. 2018.
[12] 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, pp. 263–268, Feb. 2019.
[13] M. Bossert, "On decoding using codewords of the dual code," *arXiv preprint:2001.02956*, Jan. 2020.
[14] P. J. Lee and E. F. Brickell, "An observation on the security of McEliece's public-key cryptosystem," in *Workshop Theory Appl. Crypto. Tech. (Eurocrypt 1988)*, Davos, Switzerland, pp. 275–280, May 1988.
[15] SageMath, Sage Mathematics Software System (Version 7.6), The Sage Development Team (2018), http://www.sagemath.org.
[16] 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.*, Monticello, U.S.A, pp. 290–299, Sept. 2004.