IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 1, 2019, accepted April 3, 2019, date of publication April 11, 2019, date of current version April 26, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2910240

Efficient T-EMS Based Decoding Algorithms for High-Order LDPC Codes

JING TIAN, (Student Member, IEEE), SUWEN SONG, JUN LIN, (Senior Member, IEEE), AND ZHONGFENG WANG, (Fellow, IEEE)

School of Electronic Science and Engineering, Nanjing University, Nanjing 210023, China

Corresponding authors: Jun Lin (jlin@nju.edu.cn) and Zhongfeng Wang (zfwang@nju.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61604068, and in part by the Fundamental Research Funds for the Central Universities under Grant 021014380065.

ABSTRACT Non-binary low-density parity-check (NB-LDPC) codes show larger coding gain and lower error floor than their binary counterparts in many cases. However, the existing soft decoding algorithms of NB-LDPC codes generally suffer from high computational complexity. Recently, the trellis-based extended min-sum (T-EMS) decoding algorithm has attracted much attention since it can achieve good decoding performance with high parallelism and low computational complexity. In this paper, we propose two new methods to further reduce the computational complexity and enhance the decoding performance. Firstly, we introduce, with theoretical justification, a universal scheme called threshold-based shrinking (TS) scheme, which facilitates significant reduction of computational complexity for decoding of NB-LDPC codes. Secondly, we present a modified two-extra-column (TEC) scheme and apply it to the T-EMS. Furthermore, a high-performance low-complexity decoding algorithm, named TEC-TEMS algorithm, is obtained. Combining the TS scheme with the TEC-TEMS, the new algorithm, named TS-TEC-TEMS algorithm, achieves much lower computational complexity and has negligible performance loss compared to the TEC-TEMS. For a 256-ary (256, 203) example code, compared to the T-EMS, the computational complexities of the TEC-TEMS and TS-TEC-TEMS algorithms are reduced by more than 50% and nearly 90%, respectively. Moreover, the TEC-TEMS and TS-TEC-TEMS both outperform the T-EMS by about 0.3dB when the frame error rate (FER) is around 10^{-5}.

INDEX TERMS Error correction codes, nonbinary low-density parity-check (NB-LDPC) codes, min-sum decoding algorithm, low computational complexity, forward error correction (FEC).

I. INTRODUCTION

Non-binary low-density parity-check (NB-LDPC) codes over GF(q) were first introduced by Davey and Mackay in 1998 [1], which have been proved to outperform their binary counterparts especially for short and moderate block lengths [2]. The q-ary sum-product algorithm (QSPA) adopted in [1] is extended from the SPA for binary LDPC codes, which can approach the Shannon limit performance. However, its computational complexity is unbearable due to the use of massive multiplications and the vast searching space. The complexity is reduced from $O(q^2)$ to $O(q \log q)$ using the Fast Fourier Transform-SPA (FFT-SPA) algorithm [3]. However, it still suffers large computational complexity from the complicated operations like divisions and multiplications. In order to remove multiplications from the check node processing (CNP) and variable node processing (VNP), the log-SPA [4] was proposed, which firstly changed operations from probability domain to logarithmic domain and introduced the max^* operation for the CNP instead. These simplified algorithms for QSPA bring no performance loss, but they are still hardware-inefficient.

To improve the efficiency of NB-LDPC decoding, several sub-optimal algorithms [5], [6] are proposed based on the logarithmic domain, where only additions and comparisons are involved. Declercq and Fossorier proposed the extended min-sum (EMS) algorithm [5]. In this algorithm, the configurations for the CNP are chosen from the n_m ($n_m < q$) most reliable values of each input vector and at most n_c entries are different from the order-0 configuration. If $n_m = q$ and $n_c = d_c$, the EMS is degenerated as the min-sum (MS) algorithm. Simulation results showed that there is

The associate editor coordinating the review of this manuscript and approving it for publication was Omer Chughtai.

an increasing gap of performance loss between the QSPA and the EMS as n_m and n_c decrease. A scaling factor or offset is introduced to compensate the performance loss. The min-max (MM) algorithm [6] is derived to help prevent the growth of the data width, by replacing the accumulation operations in the CNP of the EMS with comparison operations. But there is an about 0.2dB of bit error rate (BER) loss compared with the SPA. Later, quite a few researchers have spent efforts to further decrease the redundant configurations for the EMS and extend the methods to the MM [7]–[17], whereas visible performance loss is inevitably caused.

Note that the forward-backward scheme for the CNP forms the bottleneck in the EMS and MM algorithms for high-speed applications. Recently, a novel scheme, called trellis-based scheme, has been proposed to enable a high degree of parallelism and low computational complexity with small performance loss. The trellis-based EMS (T-EMS) algorithm [18] was proposed by applying this scheme to the EMS. Then, this scheme is also used for the MM by many researchers to reduce the decoding latency and the computational complexity [19]–[25]. However, their decoding performance is substantially inferior to the T-EMS.

Besides, for the soft decoding of NB-LDPC codes, messages are processed over a complete Galois field GF(q), i.e., to decode a codeword symbol, q candidate symbols require to be taken into consideration. Indeed, vast redundant information is involved, especially for a large q. Several works have been proposed previously to remove the redundancy of CNP, such as the modified EMS [7], X-EMS [12], and μ-EMS [13] algorithms. Nevertheless, all of the candidate symbols are still considered and only the updated metrics are truncated. Additionally, they just concentrate on the complexity reduction of CNP of the EMS and need to truncate and compensate the messages sent to check nodes in each iteration. Meanwhile, there is a growing performance loss when decreasing the number of remaining values.

In this paper, based on the analysis above, we take the T-EMS as the starting point and propose two methods to further reduce the computational complexity and enhance its decoding performance. On the one hand, we propose a novel scheme, named threshold-based shrinking (TS) scheme, with mathematical justification, to facilitate large reduction of computational complexity for NB-LDPC decoding without performance loss in most of the waterfall region. For the first time, we jointly reduce the candidate symbols and their metrics, directly deal with the source messages instead of the updated messages, and obtain a large reduction on complexity. This new scheme can be intuitively shown in Fig. 1. For a certain code, two thresholds are employed in the initialization step; then two corresponding subsets of a Galois field are fixed and adopted in the following iterative decoding. On the other hand, inspired by the idea of the two-extra-column (TEC) scheme for the TMM algorithm [24], we modify the TEC scheme and apply it to the simplified T-EMS algorithm. The new algorithm, called TEC-TEMS, has better decoding performance and lower computational

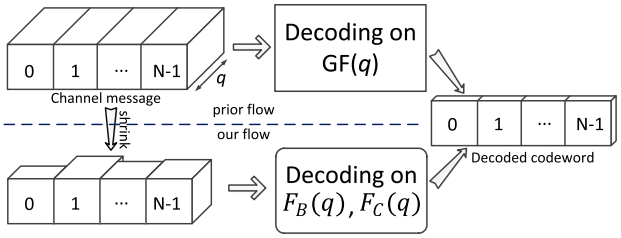


FIGURE 1. Decoding flow comparison between the prior art and our work.

complexity than the T-EMS, which has been preliminarily presented in our conference paper [26]. Combining the TS scheme with the TEC-TEMS algorithm, the proposed TS-TEC-TEMS algorithm further reduces the computational complexity significantly and achieves nearly the same decoding performance as that of the TEC-TEMS. Generally, more than 50% computations can be saved by using this scheme.

The main contributions of this paper are summarized as follows:

1) A universal complexity-reduced scheme, called TS scheme, is proposed with theoretical justification, to enable drastic reduction of computational complexity for general NB-LDPC decoding with negligible performance loss.
2) Based on the T-EMS, a new decoding algorithm, called TEC-TEMS, is developed, which achieves better decoding performance and lower computational complexity than the original T-EMS.
3) We incorporate the TS scheme into the TEC-TEMS, yielding the TS-TEC-TEMS decoding algorithm, which verifies the effectiveness of this scheme. We also provide two other example algorithms, i.e., the TS-EMS and TS-TEMS to demonstrate the feasibility and universality of the TS scheme.

Simulation results and complexity analysis are also included in this paper. The results show that for a (256, 203) example code over GF(2^8), the TEC-TEMS and TS-TEC-TEMS achieve an extra coding gain of about 0.3dB compared with the T-EMS at the FER of nearly 10^{-5}, while the computational complexities of the TEC-TEMS and TS-TEC-TEMS are reduced by more than 50% and nearly 90%, respectively.

The rest of this paper is organized as follows. Section II introduces the notations and several typical NB-LDPC decoding algorithms. The TS scheme for NB-LDPC decoding algorithms is proposed with theoretical justification in Section III. In Section IV, the TEC-TEMSA and TS-TEC-TEMSA are proposed. Simulation results and complexity analysis are presented in Section V. Finally, conclusions are drawn in Section VI.

II. BACKGROUNDS

In this section, basic notations are introduced first. Then, a general decoding process for NB-LDPC codes is presented, and several mainstream decoding algorithms follow.

A. NOTATIONS

Considering an (N, K) NB-LDPC code over GF(q) ($q = 2^p$) with block length N and information length K, parity check matrix $\mathbf{H}$ is a matrix over GF(q) with M rows and N columns. Each element of $\mathbf{H}$ is denoted as $h_{i,j}$, where i and j are the row and column indices, $0 \leq i < M$, and $0 \leq j < N$. The check node degree and variable node degree are denoted as d_c and d_v, respectively. $\mathcal{M}(j)$ is the set of check nodes connected to the j-th variable node (VN$_j$). $\mathcal{N}(i)$ is the set of variable nodes connected to the i-th check node (CN$_i$). If d_v and d_c are constant, such a code is referred to as a (d_v, d_c)-regular code; otherwise, it is an irregular code. $\oplus$ and $\otimes$ respectively denote the addition and multiplication over GF(q).

Let $\mathcal{C} = (c_0, c_1, \ldots, c_{N-1})$ denote a codeword. The received channel probability vector of a codeword symbol c_j is q-tuple, denoted as $P_j = (P_j(0), P_j(1), \ldots, P_j(q-1))$, where its corresponding tentative symbol $z_j = \arg\max_{a \in GF(q)} P_j(a)$. Assuming that X_j is a random symbol of the j-th variable node, the channel log-likelihood ratio (LLR) is defined as

$$L_j(a) = \log \frac{P(X_j = z_j)}{P(X_j = a)}, \tag{1}$$

where $a \in GF(q)$.

A soft message from CN$_i$ (or VN$_j$) to VN$_j$ (or CN$_i$) is denoted as $C_{i,j}(a)$ (or $V_{i,j}(a)$) for $a \in GF(q)$, which is also called c2v (or v2c) soft message. A soft message is also defined as a metric. Let $\lambda_j(a)$ be the j-th posterior message. I_{max} is the maximum number of iterations.

B. DECODING ALGORITHMS FOR NB-LDPC CODES

Algorithm 1 presents the general decoding flow in the logarithmic domain for NB-LDPC codes. After the initialization step, the iterative step follows, which is decomposed into four parts: 1) check node processing (CNP); 2) variable node processing (VNP); 3) tentative decision (TD); and 4) post-processing (PP).

In the CNP, steps 2 and 4 are the direct and inverse permutations, where c denotes the scaling factor. The key operation is expressed by the function Φ, whose input variable is the permuted v2c message $V'_{i,j}(a)$ and output variable is the permuted c2v message $C'_{i,j}(a)$. Different decoding algorithms lead to different expressions of Φ.

In the VNP, before computing v2c messages, the posterior message $\lambda_j(a)$, which is used not only in this part but also in the TD, is firstly updated. Then, $V_{i,j}(a)$ is calculated in Step 6 of Alg. 1.

In the TD, a tentative codeword is firstly computed with updated posterior messages. The decoding process will be terminated if the syndrome vector equals zero vector. On the other hand, a decoding failure is declared when the maximum number of iterations is exceeded.

The PP will be executed unless the decoding is terminated. This part is inevitable since the quantized LLR messages encounter the overflow problem due to the iterative accumulation. The v2c messages are offset by subtracting the minimum values to deal with this problem.

Algorithm 1 General Decoding Flow for NB-LDPC Codes

Input: H, I_{max}, $L_j(a)$, z_j.

Initialization:

1: Set $V_{i,j}(a) = L_j(a)$ and $k = 0$.

Iteration:

1) Check node processing:

2: $V'_{i,j}(h_{i,j} \otimes a) = V_{i,j}(a)$

3: $C'_{i,j}(a) = \Phi(V'_{i,j}(a))$

4: $C_{i,j}(h_{i,j}^{-1} \otimes a) = c \cdot C'_{i,j}(a)$

2) Variable node processing:

5: $\lambda_j(a) = L_j(a) + \sum_{i \in \mathcal{M}(j)} C_{i,j}(a)$

6: $V_{i,j}(a) = \lambda_j(a) - C_{i,j}(a)$

3) Tentative decision:

7: $z'_j = \arg\min_{a \in GF(q)} \lambda_j(a)$

8: If syndrome $s = H \cdot (z')^T = 0$, return z'.

9: If $k = I_{max}$, declare a decoding failure.

4) Post-processing:

10: $V_{i,j}(a) = V_{i,j}(a) - \min_{a \in GF(q)} V_{i,j}(a)$

11: Then, $k \rightarrow k + 1$ and skip back to Step 2.

Output: z'_j.

The expressions of function Φ for several typical decoding algorithms are outlined as follows:

- **Min-Sum Decoding Algorithm** The MS is a kind of sub-optimal decoding algorithm evolving from the QSPA, where the complex operations like the multiplications are removed. The Φ function is computed as:

$$C'_{i,j}(a) = \min \sum_{\substack{a_{j'} \in \mathcal{L}(i|a_j = a), \\ j' \in \mathcal{N}(i)/j}} V'_{i,j'}(a_{j'}), \tag{2}$$

  where $\mathcal{L}(i|a_j = a)$ denotes the set of configurations verifying the check node i such that $a_j = a$ for $a \in GF(q)$ and $j \in \mathcal{N}(i)$.
- **Extended Min-Sum Decoding Algorithm** Based on the MS, the EMS [5] is proposed to reduce the space of $\mathcal{L}(i|a_j = a)$, as:

$$C'_{i,j}(a) = \min \sum_{\substack{a_{j'} \in conf(n_m, n_c), \\ j' \in \mathcal{N}(i)/j}} V'_{i,j'}(a_{j'}), \tag{3}$$

  where $conf(n_m, n_c)$ is a sub-space of $\mathcal{L}(i|a_j = a)$. n_m is the number of selected values from each and every incoming message vector and n_c is the largest number of deviations from the order-0 configuration. Later, $conf(n_m, n_c)$ is constantly studied and simplified [8], [14], [16].
- **Min-Max Decoding Algorithm** The MM [6] is the variant of the EMS, replacing the accumulation operation

with the comparison operation by using the infinity norm theory, shown as:

$$C'_{i,j}(a) = \min \max_{\substack{a_{j'} \in conf(n_m, n_c), \\ j' \in \mathcal{N}(i)/j}} V'_{i,j'}(a_{j'}). \tag{4}$$

- **Trellis-Based Extended Min-Sum Decoding Algorithm** Note that the above presented decoding algorithms are based on the forward-backward scheme, requiring a large number of cycles for the decoding. The trellis-based scheme, benefiting low-complexity high-speed decoder designs, is another kind of schemes, whose Φ function is shown in Alg. 2. For convenience, it transfers the operations from normal domain to delta domain using the v2c hard message b and realizes the inverse transformation using the c2v hard message *beta*. The key computation is temporarily represented by function Ψ. For T-EMS [18], an extra column ΔW is added to realize parallel processing. The formulated process of Ψ has two steps as shown in the following:

$$\Delta W(a) = \min_{\eta(a) \in conf(n_r, n_c)} \sum_{t=1}^{d_c} \Delta V'_{i,j_t}(\eta_t(a)), \tag{5}$$

$$\Delta C'_{i,j_t}(\eta_t^P(a) \oplus a) = \min\{\Delta C'_{i,j_t}(\eta_t^P(a) \oplus a), \Delta W(a) - \Delta V'_{i,j_t}(\eta_t^P(a))\}, \tag{6}$$

where the n_r most reliable v2c messages of each row are chosen, and $\eta^P(a)$ is the configuration that has the metric $\Delta W(a)$. The vacant positions are filled with either the first minimum or the second minimum values of the corresponding rows. In [18], n_r is set to 2 and n_c to 3.

- **Trellis-Based Min-Max Decoding Algorithm** The TMM [19] is an extended version of T-EMS. The main difference lies in the way to compute the extra column ΔW, which is shown as follows:

$$\Delta W(a) = \min_{\eta(a) \in conf(1,2)} \max_{t} \Delta V'_{i,j_t}(\eta_t(a)). \tag{7}$$

In addition, many simplifications for (7) are proposed [20]–[25]. In [24], equation (6) is also modified by filling the vacant positions with a second extra column instead of the first or second minimum values of rows when necessary.

III. THRESHOLD-BASED SHRINKING METHOD FOR NB-LDPC DECODING ALGORITHMS

In this section, a novel scheme, which can be applied to generic decoding algorithms of NB-LDPC codes, is proposed. This scheme is called as threshold-based shrinking (TS) scheme, where two thresholds are introduced to truncate the channel messages and help build two subsets of Galois fields. By limiting the decoding processing on the two subsets, the computations of the whole decoding are effectively reduced. We will first separately present the two thresholds of this scheme with mathematical justification, and then explain how to combine it together with existing soft decoding algorithms. Without loss of generality, decoding algorithms processing on the logarithmic domain are considered in the following discussion.

Algorithm 2 Trellis-Based Decoding Scheme

Input: $V'_{i,j}(a)$.

The v2c hard messages:

for $j \in \mathcal{N}(i)$ **do**

  $b_j = \arg \min_{a \in GF(q)} V'_{i,j}(a)$

end for

for $j \in \mathcal{N}(i)$ **do**

  The c2v hard messages:

  $\beta_j = \sum_{j' \in \mathcal{N}(i)/j}^{\oplus} b_{j'}$

  Normal domain to delta domain:

  $\Delta V'_{i,j}(a \oplus b_j) = V'_{i,j}(a)$

end for

Computing c2v messages:

$\Delta C'_{i,j}(a) = \Psi(\Delta V'_{i,j}(a))$

Delta domain to normal domain:

$C'_{i,j}(a) = \Delta C'_{i,j}(a \oplus \beta_j)$

Output: $C'_{i,j}(a)$.

A. MATHEMATICAL ANALYSIS

1) THE FIRST THRESHOLD

For decoding algorithms of LDPC codes over GF(q), a codeword symbol is selected out of q symbols. A larger q means more redundant channel information with respect to a codeword symbol.

The first threshold is denoted as T_B, to truncate channel messages at the initialization step. And then, only the remaining messages will be used in the iterative step. The purpose is to possibly abandon the redundant messages while keeping the codeword symbol in. The following definition is used for convenience:

Definition 1: Let $\mathbf{A}$ *be a channel LLR vector of size q. A shrunk version* $\mathbf{B}$ *of* $\mathbf{A}$ *is composed of values smaller than the threshold* T_B.

According to Definition 1, $\mathbf{B}$ is the remaining vector truncated by T_B and its complementary vector $\mathbf{B}'$ is the discarded one. The feasibility and validity will be justified in the following.

The representation of $\mathbf{B}$ is written as:

$$B = \{A[a] \in A \mid A[a] < T_B,\ a \in GF(q)\}. \tag{8}$$

Based on (1), assume there exists a probability P_{T_B} that makes the following equation hold:

$$\log \frac{P_m}{P_{T_B}} = T_B, \tag{9}$$

where P_m denotes the maximum probability. Equation (9) can be rewritten as follows:

$$P_{T_B} = \frac{P_m}{e^{T_B}}. \tag{10}$$

Since $0 < P_m < 1$, we have:

$$0 < P_{T_B} = \frac{P_m}{e^{T_B}} < \frac{1}{e^{T_B}}. \tag{11}$$

It should be noted that, when q increases, the upper bound in (11) is much less possible to be approached. For simplicity, we still use this equation to draw a rough conclusion.

Given that the number of elements in $\mathbf{B}$ is n_B, the number of elements in $\mathbf{B}'$ is $q-n_B$. $q-n_B$ can be regarded as a variable to characterize the degree of computational complexity reduction. The total probability of the discarded symbols (elements of B') satisfies the following equation:

$$\sum^{q-n_B} P_{B'} \le (q-n_B)\cdot P_{T_B} < \frac{q-n_B}{e^{T_B}} \le \varepsilon, \tag{12}$$

where ε is an error-tolerant factor, which is expected to be as small as possible. To achieve a smaller ε, we can increase T_B or decrease $q-n_B$. However, larger $q-n_B$ is desired to reduce more computations. According to Definition 1, the selection of T_B directly influences the vector $\mathbf{B}$. The larger T_B is, the more elements $\mathbf{B}$ may have, i.e., the larger n_B might be. And then $q-n_B$ may be decreased. On the other hand, $q-n_B$ is also related to the order of a Galois field and a given channel model which has many variables. Hence, $q-n_B$ is a function of these variables and we temporarily express this function as $q-n_B = f(T_B, q, \ldots)$. Then the rightmost inequality of (12) can be rewritten as:

$$\frac{f(T_B, q, \ldots)}{e^{T_B}} \le \varepsilon. \tag{13}$$

Note that the AWGN has been widely used in literatures on information theory. Usually, it is true that if a decoding algorithm of LDPC codes is good over the AWGN, it will also be good over other channels, like the Rayleigh-fading channel. To further explore the relationship of T_B and the function f, without loss of generality, we take the additive white Gaussian noise (AWGN) channel with binary phase shift keying (BPSK) modulation as an example. We denote the j-th received modulated noisy codeword vector as $y_j = (y_{j,0}, \ldots, y_{j,r}, \ldots, y_{j,p-1})$ for $0 \le j < N$ and $0 \le r < p$. The received channel probability is formulated as:

$$P(X_j = a|y_j) = \prod_{r=0}^{p-1} \frac{1}{\sqrt{2\pi}\sigma} \exp\{-\frac{(y_{j,r}-(1-2a_r))^2}{2\sigma^2}\}, \tag{14}$$

where a_r is the r-th bit of the symbol a and σ^2 is the channel noise variance. The corresponding channel LLR of (1) can be reformulated as:

$$\begin{aligned} L_j(a) &= \log \frac{P(X_j = z_j|y_j)}{P(X_j = a|y_j)} \\ &= \frac{2}{\sigma^2}\cdot \sum_{r=0}^{p-1} y_{j,r}(a_r - z_{j,r}). \end{aligned} \tag{15}$$

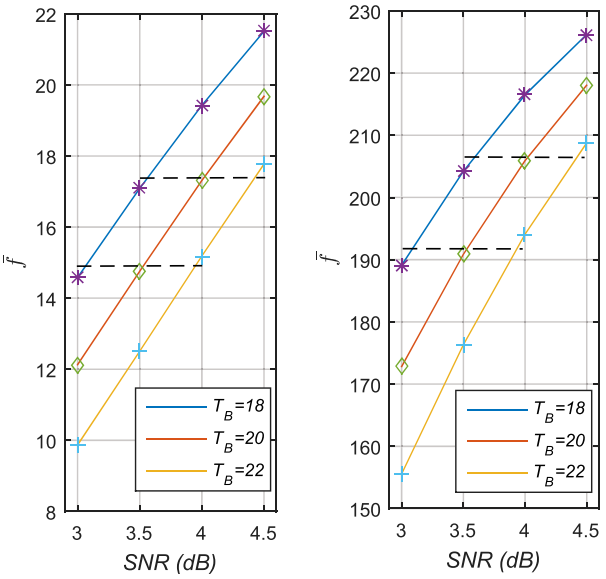


FIGURE 2. The relationships between $\bar{f}$ and *SNR* for two high order codes, with T_B equal to 18, 20, and 22. (a) For the 32-ary (837, 726) code. (b) For the 256-ary (256, 203) code.

The relationship between the channel noise variance and the signal-to-noise ratio (SNR) in dB is as follows:

$$SNR = 10\log_{10}(\frac{1}{2R\sigma^2}), \tag{16}$$

where R is the code rate. Then, equation (15) is transformed as:

$$L_j(a) = 4R\cdot 10^{\frac{SNR}{10}}\cdot \sum_{r=0}^{p-1} y_{j,r}(a_r - z_{j,r}). \tag{17}$$

Replacing vector $\mathbf{A}$ with vector $\mathbf{L}_j$, function f represents the number of LLR values where $L_j(a) \ge T_B$ for $a \in GF(q)$. Hence, the variables of f, besides T_B and q, also include R, SNR, $y_{j,r}$, and $(a_r - z_{j,r})$. Due to the nature of randomness of variable $y_{j,r}$, the value of f for a channel LLR vector is also random. There is no point in calculating an individual message. Therefore, we use a sufficiently large frames of codeword samples to get an average value for function f as $\bar{f}$.

Let us turn back to (17). Note that the channel LLR $L_j(a)$ is in exponential growth with the variable SNR and in linear growth with the other variables, which means that $L_j(a)$ is more sensitive to SNR. Using Monte-Carlo simulation, we firstly try to study the relationship between $\bar{f}$ and SNR for specific T_B. If $L_j(a)$ is not smaller than a T_B, these kind of values will be counted and averagely calculated for a codeword symbol. The statistically counting results are shown in Fig. 2, where two high order codes are considered and 10^5 frames of either one are simulated.

From Fig. 2, we can see that if $T_B = 22$ and $SNR = 4$, roughly 50% redundant symbols will be reduced for the (837, 726) code and more than 70% redundant symbols reduction for the (256, 203) code will be obtained, while keeping the

error-tolerant factor smaller than 10^{-8}. In terms of the trend, $\bar{f}$ increases almost linearly with *SNR* for a given T_B. Moreover, the curves shown in Fig. 2(a) or 2(b) are nearly in parallel with each other and the gaps between adjacent curves are almost the same, which inspire us to build two mathematical models for the first threshold T_B as below:

- **Constant Model** Considering the simplicity of implementation, T_B is set as a constant value:

$$T_B = T_0, \tag{18}$$

  where T_0 is a positive value.

By using this model, equation (13) can be expressed as:

$$\frac{\bar{f}(SNR)}{e^{T_0}} \leq \varepsilon, \tag{19}$$

where $\bar{f}$ is a linearly increasing function of *SNR*, which may deteriorate the decoding performance as *SNR* increases. Fortunately, as ε decreases exponentially with T_0 but increases linearly with $\bar{f}$, it is possible to choose a proper T_0 to achieve an acceptable ε and meanwhile keep a relatively small $\bar{f}$.

- **Linear Model** The linear model is designed for better decoding performance, which is presented as follows:

$$T_B = a \cdot x + b, \tag{20}$$

  where a ($a > 0$) and b are the coefficients, and x denotes the variable *SNR*.

In this way, the relationships between $\bar{f}$ and *SNR* can be illustrated as the dashed lines as in Fig. 2. Take the lower dashed line of Fig. 2(a) as an example. Using the coordinates $SNR = 3$, $T_B = 18$ and $SNR = 3.5$, $T_B = 20$, a line as $T_B = 4 \cdot SNR + 6$ is constructed. This line almost parallelizes the X-axis and $\bar{f}$ is nearly equal to 15. Meanwhile, we can reformulated (13) as follows:

$$\frac{\bar{f}_0}{e^{a \cdot SNR + b}} \leq \varepsilon, \tag{21}$$

where $\bar{f}_0$ can be approximately regarded as a constant. It should be noted that when *SNR* is small, the decoding performance of a code is mainly affected by the noise. The negative impact introduced by the truncation operation is negligible. But when *SNR* increases, the noise is reduced and the small error classification probability caused by this operation may gradually play a part. However, with this model, the upper bound ε can be exponentially reduced as *SNR* grows. It means that the proposed linear model for T_B has a quite small impact on the decoding performance even with a large *SNR* while achieving a significant complexity reduction.

2) THE SECOND THRESHOLD

It should be noted that the CNP is the most computationally complicated step in the whole decoding and the main task of this part for a symbol is to find a configuration that satisfies the minimal checksum in a vast searching space (q^{d_c-2} configurations in total) made up of v2c messages. Additionally, this process usually shows robustness when finding an optimal configuration, i.e., if a sub-optimal configuration is used, the decoding performance may not be affected. Many simplifications have been proposed to reduce the redundancy as introduced before. All of them require to sort the inputs of CNP in each iteration. In this subsection, we propose a novel scheme to effectively decrease the computational complexity of the CNP without any extra operations like sorting or truncating operations in the iteration step. The mathematical analysis is as follows.

Firstly, we reformulate Step 6 of Alg. 1 for the v2c message as

$$V_{i,j}(a) = L_j(a) + \sum_{i' \in \mathcal{M}(j)/i} C_{i',j}(a). \tag{22}$$

Notice that, $V_{i,j}(a)$ is dominated by the channel LLR $L_j(a)$ and tuned by an iteratively-updated accumulation term $\sum_{i' \in \mathcal{M}(j)/i} C_{i',j}(a)$, which motivates us to efficiently simplify the inputs of CNP by directly making a prior truncation based on the channel messages at the initialization. As the redundancy of CNP is much larger than those of the other steps, we try to use another smaller threshold to more aggressively truncate the channel messages, and denote this new threshold as T_C, where $T_C < T_B$. Then, the following assumption is proposed:

Assumption 1: If a channel LLR $L_j(a)$ for $a \in GF(q)$ satisfies the inequality $L_j(a) \geq T_C$, this value and the corresponding symbol will be removed at the initialization step, and then, those symbols and their corresponding soft messages will be no longer involved in the following iterative calculations of the CNP.

According to this assumption, the CNP of the existing decoding algorithms can be effectively simplified. Following simulation results will show that the performance has negligible degradation when a relatively small T_C is adopted. Similarly, the constant and linear models can be easily constructed as:

- **Constant Model**

$$T_C = T_1, \tag{23}$$

  where T_1 is an arbitrary positive value.
- **Linear Model**

$$T_C = e \cdot x + g, \tag{24}$$

  where e ($e > 0$) and g are the coefficients, and x denotes the variable *SNR*.

B. THRESHOLD-BASED SHRINKING SCHEME FOR NB-LDPC DECODING ALGORITHMS

The TS scheme is developed from the two thresholds T_B and T_C introduced above. Based on these two thresholds, two subsets of a Galois field GF(q) are defined as:

$$F_B(q) \triangleq \{a \in GF(q) \mid L_j(a) < T_B\},$$
$$F_C(q) \triangleq \{a \in GF(q) \mid L_j(a) < T_C\}.$$

TABLE 1. Four TS schemes formed by combining different models of T_B and T_C.

Schemes	CC	LC	CL	LL
T_B (model)	constant	linear	constant	linear
T_C (model)	constant	constant	linear	linear

A macroscopic illustration of our scheme for NB-LDPC decoding is shown in Fig. 1. More precisely, let us employ the two fields in Alg. 1. At the initialization step, the two fields are firstly obtained. Then, in the CNP step, the c2v messages over field $F_B(q)$ are computed by using the v2c messages over field $F_C(q)$. In the rest steps, the posterior messages and tentative decisions are updated only when the corresponding symbols are in field $F_B(q)$, and the v2c messages are updated and normalized on field $F_C(q)$. It should be noted that, for the CNP, the configurations containing abandoned elements are ignored. Particularly, if all of the configurations of a symbol are ignored, an approximate value will be chosen, where this kind of symbol is usually not the final codeword symbol. According to the mathematical models for T_B and T_C, four TS schemes (the CC, LC, CL, LL schemes) can be obtained as shown in Table 1. The results by using the constant combination (namely, the CC scheme, $T_B = T_0$ and $T_C = T_1$) have been presented in our conference paper [27]. They show a large complexity reduction with negligible performance loss at the FER of about above 10^{-5} (corresponding to the BER of about 10^{-7}), which conforms to our aforementioned analysis. A further discussion for these schemes will be given later.

Compared with the conventional truncation (CT) scheme used like in the EMS [7], the essential difference is that the TS based algorithm has irregular number of kept messages after the initialization processing while the CT based algorithm has a regular structure all the time. This irregularity helps the proposed TS scheme have the following advantages:

- The TS scheme consumes fewer computations than the CT scheme. The TS scheme only truncates the channel LLR messages using two thresholds at the initialization step and then two subsets of Galois field are built and fixed for the iterative step, whereas the CT scheme has to sort the v2c and c2v messages in the initial and iterative steps to keep the regularity.
- To reduce the computational complexity, the CT scheme only considers the CNP, but in the TS scheme, besides the CNP, the other parts are also covered.
- There is no performance gap between the algorithm with the TS scheme and the original algorithm in most of the waterfall region, but the algorithm with the CT scheme suffers from an increasing decoding gap when the parameter n_m is decreased.

IV. PROPOSED NB-LDPC DECODING ALGORITHMS

A. TEC-TEMS ALGORITHM

The TEC-TEMS is developed based on the T-EMS [18], inspired by the idea of the TEC-TMM [24]. As the min-max operation of the TMM itself reduces the accuracy of metrics, the performance improvement is limited in [24]. Note that the original motivation to replace the min-sum operation with the min-max operation is to help prevent the growth of the data width, which works well in the forward-backward scheme that requires recursive accumulation operations. Nevertheless, in the trellis scheme, these configurations are simultaneously calculated and usually no more than three terms are included. There is no need to use this kind of approximation for the trellis scheme considering the performance degeneration.

Algorithm 3 Two-Extra-Column T-EMS Algorithm

Input : $\Delta V'_{i,j}(a)$

1 $\{m_1(a), I(a)\} = Min\{\Delta V'_{i,j_k}(a)|_{k=1}^{d_c}\}$
2 $\{\Delta W_1(a), \Delta W_2(a), p_1(a), p_2(a)\} = 2\_min\{\sum_{\eta_k(a)\in conf(1,2)} (m_1(\eta_k(a)|_{k=1,2}))\}$
3 $\Delta W_2(a) = \min\{\Delta W_2(a), T_{TEC}\}$
4 **for** $k = 1$ **to** d_c **do**
5 **if** $k \neq p_1(a)$ && $k \neq p_2(a)$ **then**
6
7 $\Delta C'_{i,j_k}(a) = \Delta W_1(a)$
8 **else**
9 $\Delta C'_{i,j_k}(a) = \Delta W_2(a)$
10
11 **end if**

Output: $\Delta C'_{i,j}(a)$

Therefore, we here use the T-EMS as the baseline to increase the precision of the reliable metrics to improve the decoding performance. On the other hand, since the metrics are nonnegative numbers and the reliability of a metric decreases along with the increase of the metric, we modify the TEC scheme by suppressing the precision of the large metrics to prevent the growth of the data width. We can simply use a threshold to truncate a metric to fulfill it. To achieve a lower computational complexity, we apply the modified TEC to a simplified T-EMS where $n_r = 1$ and $n_c = 2$ and denote the new algorithm as TEC-TEMS. The detailed process of Ψ function for the TEC-TEMS is presented in Alg. 3.

In Step 1, as we set n_r to 1, function *Min* only needs to find the minimal value and its corresponding index in each row. $conf(1, 2)$ in Step 2 covers configurations with at most 2 deviations based on these minimum values. Then, function 2_*min* is used to compute the two extra columns (ΔW_1 and ΔW_2) and the corresponding positions of ΔW_1 (p_1 and p_2). The min operation in Step 3 is used to truncate the second extra column, where T_{TEC} denotes the threshold, set as T_C in our simulations. Steps 3-10 are to obtain the c2v messages by using the two extra columns and the positions.

As the original T-EMS ($n_r = 2$ and $n_c = 3$) in [18] is well-known for its good decoding performance and high degree

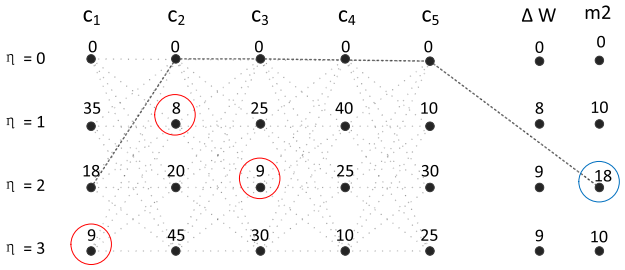


FIGURE 3. An example to compute two extra columns by the T-EMS, with $d_c = 5$ over GF(4).

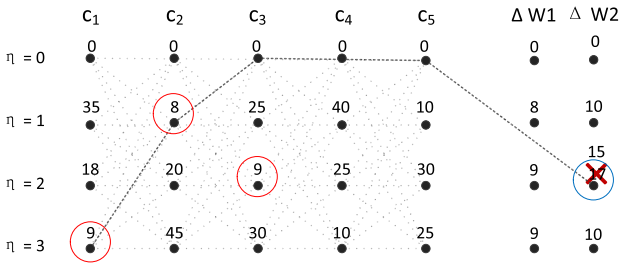


FIGURE 4. An example to compute two extra columns by the TEC-TEMS, with $d_c = 5$ over GF(4).

Algorithm 4 Threshold-Based Shrinking TEC-TEMS Algorithm

Input : $\Delta V'_{i,j}(a),\ a \in F'_C(q)$

1 **for** $a \in GF(q)$ **do**
2 $\quad I(a) = 0$
3 $\quad \{m_1(a), I(a)\} = min\{\Delta V'_{i,j_k}(a)|_{k=1}^{d_c},\ a \in F'_C(q)\}$
4 $\quad \Delta W_1(a) = T_{TS},\ \Delta W_2(a) = T_{TS},\ p_1(a) = 0,\ p_2(a) = 0$
5 $\quad \{\Delta W_1(a), \Delta W_2(a), p_1(a), p_2(a)\} = 2\_min\{\sum_{\eta_k(a) \in conf'(1,2)} (m_1(\eta_k(a)|_{k=1,2}))\}$
6 **for** $k = 1$ **to** d_c **do**
7 $\quad$ **for** $a \in F'_B(q)$ **do**
8 $\quad\quad$ **if** $k \neq p_1(a)$ && $k \neq p_2(a)$ **then**
9
10 $\quad\quad\quad \Delta C'_{i,j_k}(a) = \Delta W_1(a)$
11 $\quad\quad$ **else**
12 $\quad\quad\quad \Delta C'_{i,j_k}(a) = \Delta W_2(a)$
13
14 $\quad\quad$ **end if**

Output: $\Delta C'_{i,j}(a),\ a \in F'_B(q)$

of parallelism, to better demonstrate the effectiveness of the proposed TEC-TEMS, we will further discuss the difference between the T-EMS and the TEC-TEMS. Assume the second minimum value vector computed by the T-EMS as a second extra column m_2. We can illustrate a CN processed by the T-EMS and TEC-TEMS as shown in Figs. 3 and 4, where the CN has a check node degree of 5 over GF(4) and η denotes the symbol in delta domain ranging from 0 to $q-1$.

We can see that the first extra columns vector ΔW and vector ΔW_1, computed by the T-EMS and TEC-TEMS respectively, are equal to each other, either of which represents the most reliable metrics of the leftmost corresponding symbols. The second extra columns are used to approximate the second most reliable metrics. It shows that vector m_2 and vector ΔW_2 are almost the same except the circled location where $\eta = 2$. The configuration set of $\eta = 2$ of this CN equals {9 (0⊕2), 17 (1⊕3), 18 (0⊕2), 18 (1⊕3), 19 (1⊕3), ... }, where the values in the parenthesis are the configuration symbols for $\eta = 2$. Hence, the second most reliable metric of $\eta = 2$ equals 17, and the TEC-TEMS finds the correct values. The statistical data in [24] also shows that the values selected by the TEC scheme are closer to the real second most reliable values than directly using the first or the second minimum v2c values of rows. We set the threshold T_{TEC} to 15 so $\Delta W_2(2)$ is equal to 15 instead of 17, which could effectively reduce the data width while keeping the first and second most reliable metrics distinguished. It is generally true that the error performance is heavily influenced by the choices of the second most reliable metrics for the vacant positions (p_1 and p_2). We can see from the simulation results in the next section that though n_r is set to 1 and n_c to 2 in our proposed algorithm, the TEC-TEMS still shows better decoding performance than the original T-EMS ($n_r = 2$ and $n_c = 3$) due to the use of the modified TEC scheme, especially for high-order NB-LDPC codes.

B. TS-TEC-TEMS ALGORITHM

As mentioned before, the proposed TS scheme can be applied to almost all of the NB-LDPC decoding algorithms. In this subsection, we employ this scheme in the proposed TEC-TEMS as an example and get a new algorithm named TS-TEC-TEMS algorithm as shown in Alg. 4, where $F'_B(q)$ and $F'_C(q)$ are the fields of $F_B(q)$ and $F_C(q)$ after permutation and transformation, respectively. Step 2 is to initialize $I(a)$ to zero as a flag signal, which will be used in Step 5. If $I(a) = 0$, a configuration $\eta_k(a)$ containing the corresponding a will be ignored; configurations not involving such elements make up the set $conf'(1, 2)$. If the case where there is no configuration remained for a symbol happens, an approximation will be given as in Step 4, where T_{TS} is an undetermined value and can be optimized by Monte-Carlo simulation. In the following simulations, T_{TS} is set as T_C. At last, the c2v messages over field $F_B(q)$ are computed and output.

Note that the TS scheme used in CNP directly affects the choice of $m_1(a)$ and $I(a)$ (Step 3 of Alg. 4). If none of the values in vectors ΔW_1 and ΔW_2 are approximately chosen, the decoding performance will be the same as that of the TEC-TEMS. Additionally, because the configurations with large metrics are removed by the TS scheme, the proposed TS-TEC-TEMS owns good decoding performance and meanwhile effectively suppresses the growth of the data width without consuming extra comparisons.

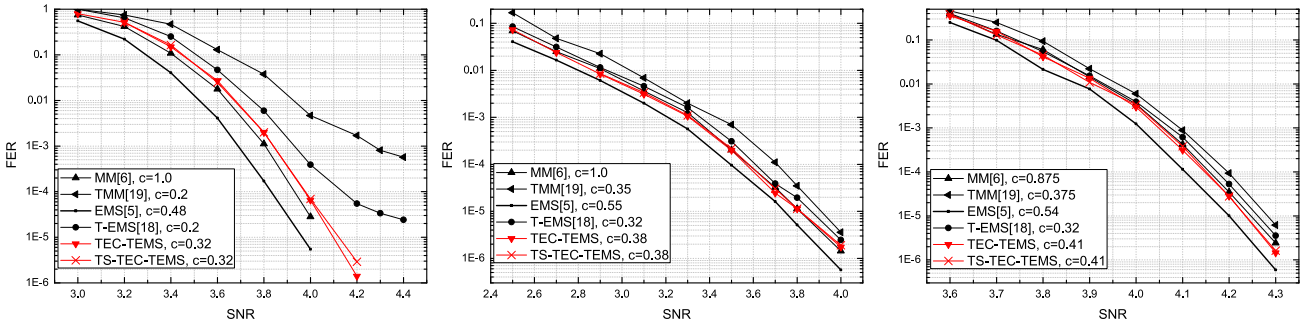


FIGURE 5. The decoding performance of NB-LDPC decoding algorithms for three codes. (a) For the (256, 203) code $\mathcal{C}_1$ over GF(2^8). (b) For the (64, 41) code $\mathcal{C}_2$ over GF(2^6). (c) For the (837, 726) code $\mathcal{C}_3$ over GF(2^5).

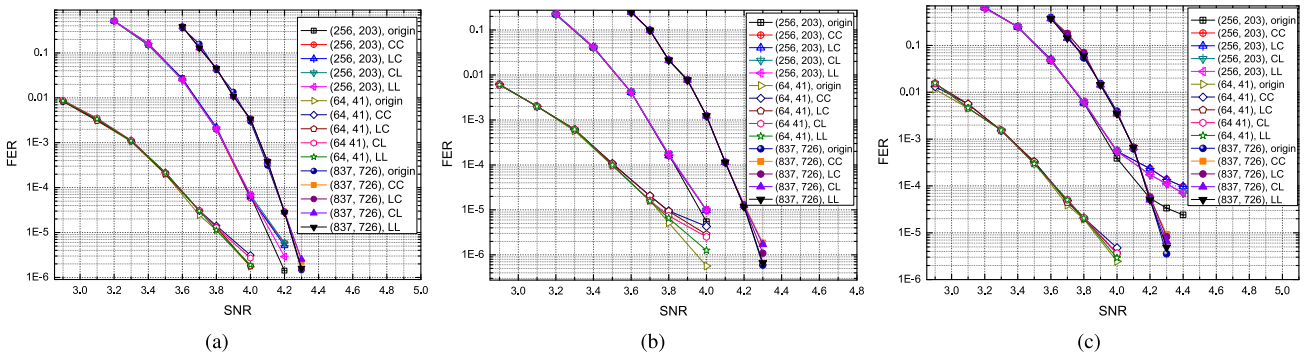


FIGURE 6. Performance comparisons of decoding algorithms applied with the four TS schemes (i.e., the CC, LC, CL, LL schemes) for the three codes. (a) The TS-TEC-TEMS. (b) The TS-EMS. (c) The TS-TEMS.

V. SIMULATION RESULTS AND COMPLEXITY ANALYSIS

A. SIMULATION RESULTS

In this subsection, the simulation results of the proposed decoding algorithms will be given. We have done extensive experiments. In these results, we have selected three NB-LDPC codes with different code rates over different Galois fields. The first code $\mathcal{C}_1$ is a (4, 16)-regular (256, 203) NB-LDPC code over GF(2^8) with a rate of 0.79. The second code $\mathcal{C}_2$ is a rate-0.64 (4, 8)-regular (64, 41) NB-LDPC code over GF(2^6). The third code $\mathcal{C}_3$ is a (4, 27)-regular (837, 726) NB-LDPC code over GF(2^5) with a rate of 0.87. The first two codes are constructed based on the parallel bundles of lines in Euclidean geometries by using geometry decomposition [28], and the last code is structured by array dispersion [29]. The characteristics of these codes, like the Shannon limit, can also be referred to in the corresponding references. As the proposed two algorithms are modifications to the T-EMS with theoretical justification given in the paper, we think these examples are sufficient to explain the effectiveness of the proposed algorithms. We take the BPSK over the AWGN channel into account. The maximum number of iterations I_{max} is set to 50.

As shown in Fig. 5, the proposed TEC-TEMS and TS-TEC-TEMS algorithms are simulated for the three codes. The EMS [5], T-EMS [18], MM [6], and TMM [19] algorithms are also simulated for performance comparisons. The scaling factors (c) are also appended in the figures. The adopted parameters for the TS-TEC-TEMS for the three codes are the same, namely, ($a = 4$, $b = 4$) for T_B and ($e = 3$, $g = 2$) for T_C, which can be well tuned to achieve a good tradeoff between the decoding performance and computational complexity. The results show that the code rate mainly decides the waterfall region of a code as usual. The EMS has the best decoding performance among them, which is also very close to that of the QSPA and the Shannon limit shown in [5]. For the proposed TEC-TEMS and TS-TEC-TEMS in each sub-figure, we can see that their decoding performances are almost the same, comparable to that of the MM, slightly inferior to that of the EMS, and better than those of the T-EMS and TMM, especially for NB-LDPC codes over high-order Galois fields. For example, both of them obtain an about 0.3dB extra coding gain compared to the T-EMS for the (256, 203) code.

To practically verify the universality of the proposed TS schemes, two other TS based example algorithms, i.e. the TS-EMS and TS-TEMS, are also designed and simulated. The performance comparisons of three decoding algorithms applied with the four TS schemes for the three codes are shown in Fig. 6. The adopted parameters of different schemes for different algorithms with different codes are presented in Table 2. The simulation results show that the decoding performances of the TS based decoding algorithms are strictly

TABLE 2. Parameters and coefficients used for simulations in Fig. 6.

Algorithms	the (256, 203) code					the (64, 41) code					the (837, 726) code				
	T_0	T_1	(a,b)	(e,g)	c	T_0	T_1	(a,b)	(e,g)	c	T_0	T_1	(a,b)	(e,g)	c
TS-TEC-TEMS	20	12	$(4,4)$	$(3,2)$	0.32	18	12	$(4,4)$	$(3,2)$	0.38	20	12	$(4,4)$	$(3,2)$	0.41
TS-EMS	20	12	$(4,4)$	$(3,2)$	0.48	18	12	$(4,4)$	$(3,2)$	0.55	20	12	$(4,4)$	$(3,2)$	0.54
TS-TEMS	20	16	$(4,4)$	$(3,6)$	0.2	18	16	$(4,4)$	$(3,5)$	0.32	20	16	$(4,4)$	$(3,4)$	0.32

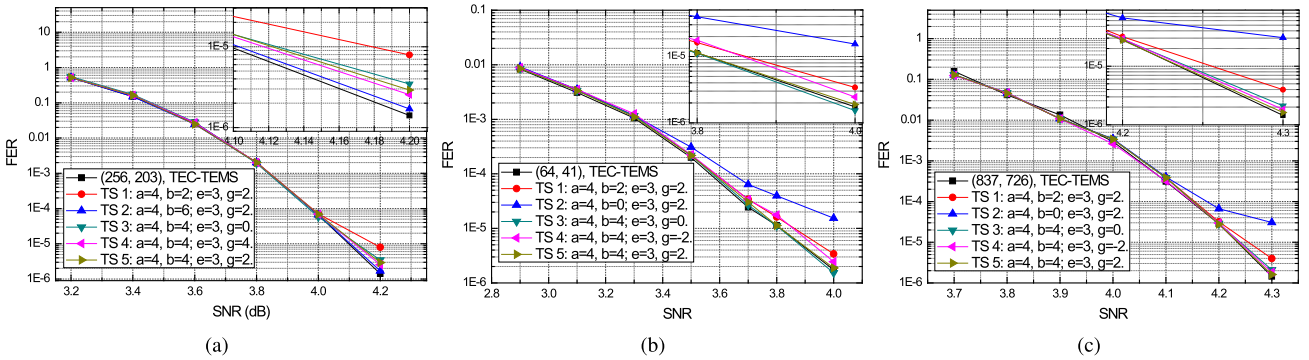


FIGURE 7. The decoding performance of TEC-TEMS and TS-TEC-TEMS for three NB-LDPC codes. (a) For the (256, 203) code over GF(2^8). (b) For the (64, 41) code over GF(2^6). (c) For the (837, 726) code over GF(2^5).

the same as those of the original ones even when the FER is about 10^{-5}. Then, usually, the LL based algorithms still perform well even in a much lower error rate. However, the CC based algorithms show slight error floors in most cases. And the LC and CL based algorithms have performances between the LL and CC based algorithms. These performance degradations indicate that the small error classification probability plays a non-trivial role in these regions, just as analyzed in Section III. Note that there is no performance gap between the original algorithms and the new ones in most of the waterfall regions, different from the EMS in [7].

As the LL scheme is more practical than other schemes, we are going to introduce the way we use to select its parameters. As explained in Section III-A, when setting the parameter $\bar{f}$ as constant, the linear model for T_B is obtained based on the dashed lines parallelizing to the X axis shown in Fig. 2. In those cases, the parameter a is fixed. For the two example codes in Fig. 2, a is equal to 4. And b is used to shift the curve up and down. Based on the constraint shown in (21), when a specific upper bound for the error classification probability is chosen, b can be decided correspondingly. Different from T_B, T_C cannot be directly deduced. Of course, any effective methods can be used to optimize this parameter. However, the Monte Carlo method is usually widely used and very suitable to optimize the parameters of NB-LDPC decoding algorithms, and the compared algorithms in the literatures were only optimized by using the Monte Carlo method. Thus, we adopt the estimated density evolution simulated by means of Monte Carlo estimations of densities to determine e and g of its linear model, using a sufficiently large number of samples. In order to discuss the separate effectiveness of T_B and T_C for the decoding performance, five sets of parameters for the TS-TEC-TEMS are given for the three different NB-LDPC codes shown as in Fig. 7. The TEC-TEMS curve is also provided as a baseline. Generally, these curves in each sub-figure show consistent performances with the original ones when the FER is not too low in the waterfall regions. As the SNR continuously increases, the divergence gradually occurs. These differences are highlighted in these partial enlargements. To further detail the influence caused by the two thresholds, the five sets, marked as TS 1, TS 2, TS 3, TS 4, and TS 5, can be classified into two groups. The first group including the TS 1, TS 2, and TS 5, is used to verify the effect of the first threshold T_B. The second group, covering the TS 3, TS 4, and TS 5, is designed to test the impact of T_C. For fair comparisons, only the constant coefficients, i.e. the parameters b and g, are changed with an interval of 2. We can see that a significant performance improvement appears in the first group when increasing the value of b. But this kind of improvement is not obvious in the second group as g increases. In contrast, we notice that there is a better decoding performance with a smaller g in some cases, such as the TS 3 curve versus the TS 5 curve in Fig. 7(b) and the TS 4 curve versus the TS 3 curve in Fig. 7(c). It means that the proposed approximation related to T_C has a compensation for the cases with overestimation caused by the original estimation method. It can also explain the phenomenon where the TS based algorithms perform slightly better than the original ones in some cases. In other words, there is a robustness when choosing the coefficients of T_C. In brief, in order to effectively reduce the computational complexity and achieve a comparable decoding performance, the first threshold T_B cannot be too small, but the second threshold T_C can be aggressively reduced. A tradeoff can be made between the complexity and performance based on the two thresholds.

TABLE 3. Analysis of computational complexity per iteration for the proposed algorithms and the T-EMS.

	T-EMS [18]	**TEC-TEMS**	**TS-TEC-TEMS**[1]
FA	$(3d_c-2)M+(3q-1)\delta$	$(3d_c-2)M+2q\delta$	$(3d_c-2)M+(n_B+n_C)\delta$
FM	$(2q+2)\delta$	$(2q+2)\delta$	$(n_B+n_C+2)\delta$
RM	$q\delta$	$q\delta$	$n_B\delta$
RA	$(4q-1)\delta+4(q/2-1)(q-1)M$	$3q\delta+(q/2-1)(q-1)M$	$(n_B+2n_C)\delta+(t-1)(q-1)M$
CS	$(q-1)N+4(q-1)\delta$ $+(7q/2+2d_c-10)(q-1)M$	$(q-1)N+3(q-1)\delta$ $+(3q/2+d_c-4)(q-1)M$	$(q+n_B)N/I_{max}$ $+(n_B-1)N+(2n_B+n_C-3)\delta$ $+(3t+n_C^r-5)(q-1)M$

FA: Finite-Field Addition FM: Finite-Field Multiplication
RM: Real Multiplication RA: Real Addition CS: Comparison and Selection
[1] t denotes the average number of remaining pairs of configurations for a symbol.
$n_C^r = n_C \cdot \frac{d_c}{q}$.

B. COMPLEXITY ANALYSIS

Note that the T-EMS is the starting point of the TEC-TEMS and TS-TEC-TEMS. We will analyze and compare the computational complexity of them in the following.

According to Algs. 1-4 and the description for T-EMS in [18], the computational operations of the three algorithms can be summarized into five categories: the finite-field addition (FA), finite-field multiplication (FM), real multiplication (RM), real addition (RA), and comparison and selection (CS). Because the proposed two methods have an impact on the whole decoding, we will take the computation complexity of the complete decoding process into consideration and count the number of operations by per iteration. The computational complexity of the initialization step is averagely distributed to each iteration. The number of nonzero values in $\mathbf{H}$ matrix is denoted as $\delta = d_c M = d_v N$.

For the TEC-TEMS, we can first focus on Alg. 1. It can be easily calculated that, except the Φ function, the other parts take $(d_c-1)M$ FAs, $(2q+1)\delta$ FMs, $q\delta$ RMs, $3q\delta$ RAs, and $(q-1)(N+\delta)$ CSs per iteration. In Alg. 2, Step 2 can be divided into two sub-steps as:

$$b'_j = \arg\min_{a\in GF(q)} V_{i,j}(a), \tag{25}$$

$$b_j = b'_j \otimes h_{i,j}. \tag{26}$$

Note that the first sub-step has been computed in Step 10 of Alg. 1. Therefore, the trellis-based decoding for the Φ function costs $(2d_c-1)M+2q\delta$ FAs and δ FMs to compute the v2c and c2v hard messages, and the direct and inverse domain transformations. The TEC-TEMS for the Ψ function is shown in Alg. 3. Step 1 requires $(d_c-1)(q-1)M$ CSs and Steps 3-10 need $(q-1)(2\delta+M)$ CSs. Back to Step 2, the worst case is that the $q/2-1$ pairs of configurations for a symbol have different column indexes and all need to be computed. Considering that case, it takes $(q/2-1)(q-1)M$ RAs and $(q-3)(q-1)M$ CSs. In addition, comparing the column indexes requires $(q/2-1)(q-1)M$ CSs.

For the proposed TS-TEC-TEMS, according to the introduction in Section III, the initial step averagely takes $(q+n_B)N/I_{max}$ CSs per iteration. Thanks to the use of the two fields $F_B(q)$ and $F_C(q)$, in the iterative step, except the Ψ function, the complexity of the other parts can be computed by replacing the parameter q with n_B or n_C. Algorithm 4 shows the computing process of the Ψ function. In Step 3, the number of the v2c messages of a row for a symbol is averagely computed as $n_C^r = n_C \cdot \frac{d_c}{q}$. Then, this step costs $(n_C^r-1)(q-1)M$ CSs. Steps 6-14 take $2(n_B-1)\delta$ CSs. For Step 5, we assume t as the average number of remaining pairs of configurations for a symbol, which can be statistically obtained by the Monte-Carlo simulation. The complexity of this step can be formulated as $(x-1)(q-1)M$ RAs and $(3x-4)(q-1)M$ CSs.

TABLE 4. Numerical analysis of the computational complexity.

the 256-ary (256, 203) code			
	T-EMS [18]	**TEC-TEMS**	**TS-TEC-TEMS**
FA	788,352 (1.00)	527,232 (0.67)	70,927.4 (0.09)
FM	526,336 (1.00)	526,336 (1.00)	10,031.4 (0.13)
RM	262,144 (1.00)	262,144 (1.00)	51,312.6 (0.20)
RA	9,338,112 (1.00)	2,859,072 (0.31)	911,098.9 (0.10)
CS	16,091,520 (1.00)	7,311,360 (0.45)	2,631,379.9 (0.16)
Total	27,006,464 (1.00)	11,486,144 (0.43)	3,734,750.2 (0.14)
the 32-ary (837, 726) code			
	T-EMS [18]	**TEC-TEMS**	**TS-TEC-TEMS**
FA	327,856 (1.00)	224,068 (0.68)	79,568.3 (0.24)
FM	220,968 (1.00)	220,968 (1.00)	76,468.3 (0.35)
RM	107,136 (1.00)	107,136 (1.00)	48,646.4 (0.45)
RA	655,836 (1.00)	379,068 (0.58)	148,442.9 (0.23)
CS	1,040,763 (1.00)	610,235 (0.59)	305,890.0 (0.29)
Total	2,352,559 (1.00)	1,541,475 (0.66)	659,017.9 (0.28)

Computing the computational complexity of the T-EMS is referred to [18], where n_r is 2 and n_c is 3. The computations per iteration of the three algorithms are concluded in Table 3. Obviously, the complexity of the TEC-TEMS is less than that of the T-EMS, and that of the TS-TEC-TEMS is also significantly reduced. To be more precise, the numerical analyses for $\mathcal{C}_1$ and $\mathcal{C}_3$ codes are presented in Table 4. The parameters n_B, n_C, and t for the TS-TEC-TEMS with the LL scheme (TS 5) are statistically counted with 10^5 frames of either code when the FER is about 10^{-5}, as shown in Table 5.

The total numbers of operations are also listed in Table 4 as a reference. Based on the normalization in the brackets, compared with the T-EMS, the complexity of the proposed

TABLE 5. The statistical values of the parameters for NB-LDPC codes at the FER of lower than 10^{-4}.

Code	(256, 203)	(837, 726)
SNR (dB)	4	4.2
n_B	50.11	14.53
n_C	16.28	6.31
t	51.64	15.97

TABLE 6. The extra coding gains between the proposed algorithms and the T-EMS.

code	T-EMS [18]	TEC-TEMS	TS-TEC-TEMS
(256, 203)	0	+0.3dB	+0.3dB
(837, 726)	0	+0.03dB	+0.03dB

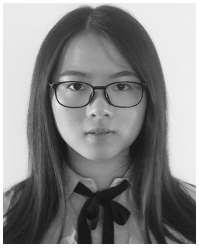
TEC-TEMS is reduced by more than 50% for $\mathcal{C}1$ and by about 35% for $\mathcal{C}3$. The complexity of the proposed TS-TEC-TEMS is further significantly reduced. Generally, more than half of computations can be saved by the introduced TS scheme. Compared to the T-EMS, nearly 90% complexity is reduced for code $\mathcal{C}1$ and more than 70% complexity for code $\mathcal{C}3$. Meanwhile, the proposed algorithms achieve better FER performance than the T-EMS. For example, as listed in Table 6, an extra coding gain of about 0.3dB is obtained for $\mathcal{C}1$ and that of about 0.03dB for $\mathcal{C}3$ at the FER of about 10^{-5}.

VI. CONCLUSION

In this paper, we have proposed two methods to reduce the computational complexity and enhance the decoding performance for the T-EMS algorithm of NB-LDPC codes. At first, the TS scheme is presented with theoretical justifications, which facilitates large complexity reduction for the decoding of NB-LDPC codes. Then, the TEC scheme is modified and applied to the T-EMS algorithm. The proposed two new algorithms, the TEC-TEMS and TS-TEC-TEMS, both achieve better decoding performance and lower computational complexity than the T-EMS that is well-known in achieving good decoding performance with low computation complexity. Considering a 256-ary (256, 203) example code, compared to the T-EMS, both the TEC-TEMS and TS-TEC-TEMS achieve an extra coding gain of about 0.3dB. In addition, the computational complexity of the TEC-TEMS is reduced by more than 50% and the TS-TEC-TEMS has almost 90% computational complexity reduction.

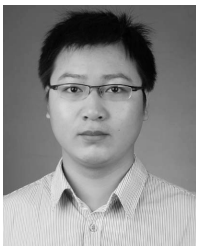
REFERENCES

[1] M. C. Davey and D. MacKay, "Low-density parity check codes over GF(q)," *IEEE Commun. Lett.*, vol. 2, no. 6, pp. 165–167, Jun. 1998.
[2] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Trans. Inf. Theory*, vol. 45, no. 2, pp. 399–431, Mar. 1999.
[3] L. Barnault and D. Declercq, "Fast decoding algorithm for LDPC over GF(2q)," in *Proc. IEEE Inf. Theory Workshop*, Mar./Apr. 2003, pp. 70–73.
[4] H. Wymeersch, H. Steendam, and M. Moeneclaey, "Log-domain decoding of LDPC codes over GF(q)," in *Proc. IEEE Int. Conf. Commun.*, vol. 2, Jun. 2004, pp. 772–776.
[5] D. Declercq and M. Fossorier, "Decoding algorithms for nonbinary LDPC codes over GF (q)," *IEEE Trans. Commun.*, vol. 55, no. 4, pp. 633–643, Apr. 2007.
[6] V. Savin, "Min-max decoding for non binary LDPC codes," in *Proc. IEEE Int. Symp. Inf. Theory*, Jul. 2008, pp. 960–964.
[7] A. Voicila, D. Declercq, F. Verdier, M. Fossorier, and P. Urard, "Low-complexity decoding for non-binary LDPC codes in high order fields," *IEEE Trans. Commun.*, vol. 58, no. 5, pp. 1365–1375, May 2010.
[8] E. Boutillon and L. C. Canencia, "Bubble check: A simplified algorithm for elementary check node processing in extended min-sum non-binary LDPC decoders," *IEEE Electron. Lett.*, vol. 46, no. 9, pp. 633–634, Apr. 2010.
[9] J. Lin, J. Sha, Z. Wang, and L. Li, "Efficient decoder design for nonbinary quasicyclic LDPC codes," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 57, no. 5, pp. 1071–1082, May 2010.
[10] J. Lin, J. Sha, Z. Wang, and L. Li, "An efficient VLSI architecture for nonbinary LDPC decoders," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 57, no. 1, pp. 51–55, Jan. 2010.
[11] K. He, J. Sha, and Z. Wang, "Nonbinary LDPC code decoder architecture with efficient check node processing," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 59, no. 6, pp. 381–385, Jun. 2012.
[12] X. Ma, K. Zhang, H. Chen, and B. Bai, "Low complexity X-EMS algorithms for nonbinary LDPC codes," *IEEE Trans. Commun.*, vol. 60, no. 1, pp. 9–13, Jan. 2012.
[13] S. Zhao, Z. Lu, X. Ma, and B. Bai, "A variant of the EMS decoding algorithm for nonbinary LDPC codes," *IEEE Commun. Lett.*, vol. 17, no. 8, pp. 1640–1643, Aug. 2013.
[14] E. Boutillon, L. Conde-Canencia, and A. A. Ghouwayel, "Design of a GF(64)-LDPC decoder based on the EMS algorithm," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 60, no. 10, pp. 2644–2656, Oct. 2013.
[15] F. Cai and X. Zhang, "Relaxed min-max decoder architectures for non-binary low-density parity-check codes," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 21, no. 11, pp. 2010–2023, Nov. 2013.
[16] P. Schläfer, N. Wehn, M. Alles, T. Lehnigk-Emden, and E. Boutillon, "Syndrome based check node processing of high order NB-LDPC decoders," in *Proc. 22nd Int. Conf. Telecommun. (ICT)*, Apr. 2015, pp. 156–162.
[17] Q. Huang, L. Song, and Z. Wang, "Set message-passing decoding algorithms for regular non-binary LDPC codes," *IEEE Trans. Commun.*, vol. 65, no. 12, pp. 5110–5122, Dec. 2017.
[18] E. Li, D. Declercq, and K. Gunnam, "Trellis-based extended min-sum algorithm for non-binary LDPC codes and its hardware structure," *IEEE Trans. Commun.*, vol. 61, no. 7, pp. 2600–2611, Jul. 2013.
[19] J. O. Lacruz, F. García-Herrero, D. Declercq, and J. Valls, "Simplified trellis min–max decoder architecture for nonbinary low-density parity-check codes," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 23, no. 9, pp. 1783–1792, Sep. 2015.
[20] J. O. Lacruz, F. Garcia-Herrero, J. Valls, and D. Declercq, "One minimum only trellis decoder for non-binary low-density parity-check codes," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 62, no. 1, pp. 177–184, Jan. 2015.
[21] J. O. Lacruz, F. Garcia-Herrero, and J. Valls, "Reduction of complexity for nonbinary LDPC decoders with compressed messages," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 23, no. 11, pp. 2676–2679, Nov. 2015.
[22] J. O. Lacruz, F. G. Herrero, M. J. Canet, and J. Valls, "High-performance NB-LDPC decoder with reduction of message exchange," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 24, no. 5, pp. 1950–1961, May 2016.
[23] J. O. Lacruz, F. G. Herrero, M. J. Canet, and J. Valls, "Reduced-complexity nonbinary LDPC decoder for high-order Galois fields based on trellis min-max algorithm," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 24, no. 8, pp. 2643–2653, Aug. 2016.
[24] H. P. Thi and H. Lee, "Two-extra-column trellis min–max decoder architecture for nonbinary LDPC codes," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 25, no. 5, pp. 1787–1791, May 2017.
[25] H. P. Thi and H. Lee, "Basic-set trellis Min–Max decoder architecture for nonbinary LDPC codes with high-order Galois fields," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 26, no. 3, pp. 496–507, Mar. 2018.
[26] S. Song, J. Lin, J. Tian, and Z. Wang, "A reduced complexity decoding algorithm for NB-LDPC codes," in *Proc. IEEE 17th Int. Conf. Commun. Technol. (ICCT)*, Oct. 2017, pp. 127–131.
[27] J. Tian, J. Lin, and Z. Wang, "An efficient NB-LDPC decoding algorithm for next-generation memories," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2018, pp. 1–5.
[28] L. Zeng, L. Lan, Y. Y. Tai, B. Zhou, S. Lin, and K. A. S. Abdel-Ghaffar, "Construction of nonbinary cyclic, quasi-cyclic and regular LDPC codes: A finite geometry approach," *IEEE Trans. Commun.*, vol. 56, no. 3, pp. 378–387, Mar. 2008.
[29] B. Zhou, J. Kang, S. Song, S. Lin, K. Abdel-Ghaffar, and M. Xu, "Construction of non-binary quasi-cyclic LDPC codes by arrays and array dispersions," *IEEE Trans. Commun.*, vol. 57, no. 6, pp. 1652–1662, Jun. 2009.



JING TIAN (S'18) received the B.S. degree in microelectronics from Nanjing University, Nanjing, China, in 2015, where she is currently pursuing the M.S. and Ph.D. integrated degrees in information and communication engineering. Her research interests include forward error correction decoding algorithm and VLSI design for digital signal processing.

SUWEN SONG received the B.S. degree in electronic information science and technology from Nanjing University, Nanjing, China, in 2017, where she is currently pursuing the M.S. and Ph.D. integrated degrees. Her research interests include error correction coding and decoding algorithms, especially for LDPC codes. Moreover, she also has interest in their efficient VLSI designs and implementations.



JUN LIN (S'14–SM'18) received the B.S. degree in physics and the M.S. degree in microelectronics from Nanjing University, Nanjing, China, in 2007 and 2010, respectively, and the Ph.D. degree in electrical engineering from Lehigh University, Bethlehem, in 2015.

From 2010 to 2011, he was an ASIC Design Engineer with AMD. In summer 2013, he was an Intern with Qualcomm Research, Bridgewater, NJ, USA. In 2015, he joined the School of Electronic Science and Engineering, Nanjing University, where he is currently an Associate Professor. His current research interest includes low-power high-speed VLSI design, specifically the VLSI design for digital signal processing and cryptography. He is a member of the Design and Implementation of Signal Processing Systems (DISPS) Technical Committee of the IEEE Signal Processing Society. He was a co-recipient of the Merit Student Paper Award at the IEEE Asia Pacific Conference on Circuits and Systems, in 2008. He was a recipient of the 2014 IEEE Circuits and Systems Society (CAS) Student Travel Award.

ZHONGFENG WANG (M'00–SM'05–F'16) received the B.E. and M.S. degrees from Tsinghua University, Beijing, China, and the Ph.D. degree from the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN, USA, in 2000.

He joined Nanjing University as a Distinguished Professor in 2016, after serving Broadcom Corp., as a leading VLSI Architect, for nine years. From 2003 to 2007, he was an Assistant Professor with the School of EECS, Oregon State University, Corvallis, OR, USA. Previously, he was with National Semiconductor Corporation. He is a world-recognized expert on VLSI for Forward Error Correction Codes. He has published nearly two hundreds of technical papers, has edited one book (VLSI), and has filed tens of U.S. patent applications and disclosures. He was a recipient of the IEEE Circuits and Systems (CAS) Society VLSI Transactions Best Paper Award, in 2007. In the current record, he had five papers ranked among top twenty most downloaded manuscripts in the IEEE TRANSACTIONS ON VLSI SYSTEMS, since 2007. During his tenure at Broadcom, he has contributed significantly on 10Gbps and beyond high-speed networking products. In addition, he has made critical contributions in defining FEC coding schemes for 100Gbps and 400Gbps Ethernet standards. So far, his technical proposals have been adopted by many networking standard specs. His current research interests are in the area of VLSI for High-Speed Signal Processing Systems.

Dr. Wang has served as a Technical Program Committee Member, the Session Chair, the Track Chair, and a Review Committee Member for many IEEE and ACM conferences. In 2013, he has served in the Best Paper Award Selection Committee for the IEEE Circuits and System Society. Since 2004, he has been serving as an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS I (TCAS-I), TCAS-II, and the IEEE TRANSACTIONS ON VLSI SYSTEMS for many terms.