IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON ADVANCES IN CHANNEL CODING FOR 5G AND BEYOND

Received October 10, 2018, accepted November 8, 2018, date of publication November 14, 2018,
date of current version December 18, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2881243

Capacity-Approaching Non-Binary Turbo Codes: A Hybrid Design Based on EXIT Charts and Union Bounds

TOSHIKI MATSUMINE, (Student Member, IEEE), AND HIDEKI OCHIAI, (Senior Member, IEEE)
Department of Electrical and Computer Engineering, Yokohama National University, Yokohama 240-8501, Japan
Corresponding author: Toshiki Matsumine (matsumine-toshiki-tk@ynu.jp)

This work was supported by JSPS KAKENHI under Grant JP16H02345.

ABSTRACT In this paper, we introduce a novel design approach for capacity-approaching non-binary turbo codes. There are two important factors that impact the performance of turbo codes in general: 1) the convergence behavior of iterative decoding in the low signal-to-noise ratio (SNR) and 2) the error-floor effect in the high SNR. We thus design the non-binary turbo codes by means of the EXIT charts and truncated union bounds. We first reduce the search space of component recursive convolutional codes by the analysis based on the truncated union bounds in conjunction with the uniform interleaver, followed by its optimization through the EXIT chart analysis. The construction of the EXIT chart for non-binary turbo codes with fixed code coefficients is a non-trivial task by the fact that these messages have multiple parameters to identify. Therefore, we develop a new EXIT chart analysis for non-binary messages which does not rely on any specific message model. It is demonstrated through computer simulations that the well-designed non-binary turbo codes achieve a better performance than their binary counterparts as well as the conventional non-binary LDPC codes of the same field size. Furthermore, the code design is extended to high-order modulation, and our turbo codes designed for quadrature amplitude modulation are shown to outperform the conventional turbo trellis coded modulation schemes.

INDEX TERMS Turbo codes, non binary codes, EXIT charts, union bounds, coded modulation.

I. INTRODUCTION

Binary turbo and low-density parity-check (LDPC) codes [1]–[3] can asymptotically approach the channel capacity as the frame size increases, but a specific code design that can improve their finite-length performance should be of practical importance. Subsequently, non-binary turbo and LDPC codes have been proposed in [4] and [5] and the related recent results have shown that these codes can significantly improve the performance of their binary counterparts with short-to-moderate block length [6]. The price is its increasing decoding complexity mainly due to the metric calculation associated with non-binary symbols. By exploiting the fast Fourier transform (FFT), the computational complexity of decoding can be limited to $O(q \log q)$ when the finite field size q is a power-of-two [7], [8], but the decoding complexity may remain significant when the field size q is large.

In this paper, we consider the design issue of non-binary turbo codes that consist of two identical convolutional encoders defined over GF(q) with $q = 2^p$ and $p > 1$. The two major issues are the convergence performance of iterative decoding in low signal-to-noise ratio (SNR) and the resulting error floor in high SNR, and optimizing the minimum Hamming distance of constituent codes without considering their convergence behavior may not be sufficient for designing the best parallel concatenated code. For this reason, we propose a two-step optimization algorithm. In the first step, we identify the good component recursive convolutional codes (RCCs) from a distance spectrum viewpoint by analyzing their corresponding truncated union bound. Since the distance spectrum of turbo codes is not unique and depends on the specific interleaver structure, the concept of *uniform interleaver* [9] is utilized for simplicity of analysis. After reducing the search space by the first step, the second step selects the best component RCCs in terms of decoding convergence by tracking the message distribution during the iterative decoding. The resulting codes thus achieve good

performance in terms of both convergence behavior and error floor.

The convergence analysis of the iterative decoder has been extensively studied in the literature. The *density evolution* [10] is a powerful tool for analyzing the convergence property of iterative decoder. Following the concentration theorem by Luby *et al.* [11], a rigorous approach for iterative decoding threshold analysis under the cycle-free assumption is developed by Richardson and Urbanke [10]. Subsequently, the Gaussian approximation was proposed in [12] and [13], which treats the message as Gaussian random variables. By tracking only the mean of a message density, the analysis can be made much simpler even with reasonable accuracy. The Gaussian approximation for binary LDPC codes was then extended to the non-binary case in [14]. For turbo codes, the analysis based on Gaussian approximation called extrinsic information transfer (EXIT) chart was first developed by ten Brink in [15]. Later, a similar analysis based on the extrinsic information SNR was developed in [13] and [16]. These studies enable us to estimate the iterative decoding threshold of turbo decoders with a reasonable effort.

For non-binary codes, the convergence analysis is in general a non-trivial task, since the message for these codes is no longer specified by a scalar value. In [14] and [17], the Gaussian approximation-based analyses for the non-binary LDPC code ensemble have been studied, assuming the *permutation-invariance* property of the message. This assumption significantly simplifies the analysis by treating the message as the Gaussian random variable with a single parameter. This useful property, however, holds only for the *code ensemble* and thus may not be directly applicable to the design of *specific* codes. Furthermore, the EXIT chart analysis for non-binary turbo trellis coded modulation (TTCM) proposed in [18] and [19] may fail to capture the actual performance as the constellation size increases, mainly stemming from the fact that the multi-dimensional distribution of the *a priori* information is not carefully taken into consideration. In this work, we develop a new method to estimate the decoding convergence for non-binary turbo codes with fixed code coefficients. Our approach is to generate message samples used for evaluating the EXIT curve by the Monte-Carlo method instead of modeling the multi-dimensional message distribution. To do this, in addition to the conventional maximum *a posteriori* (MAP) decoder used for evaluation of the EXIT curve, we introduce an auxiliary MAP decoder that will be used for generating message samples. Then, we perform the actual MAP decoding with only two decoders to identify the convergence, thereby requiring much less computational complexity compared to actual iterative decoding.

Monte-Carlo density evolution may be another approach to analyze the convergence of the non-binary message with multiple parameters [20]–[22]. It is similar to our approach in that it does not assume any specific message model. However, the main difference is that Monte-Carlo density evolution requires iterative message-passing procedure to evaluate the convergence behavior, whereas our approach does not require such a procedure. Since our EXIT chart analysis only requires actual decoding operation at two constituent MAP decoders to identify the convergence, the analysis is much simpler, while retaining the accuracy of the analysis.

Turbo codes over non-binary fields or rings, as well as multi-binary input turbo codes have been studied so far in [5], [23], [24], and [24]–[30]. However, how to design good non-binary turbo codes, especially in the case of large field size, remains an open problem. Specifically, the convergence analysis of the iterative decoding for *specific* non-binary turbo codes has not been studied so far, due to their intractable message distribution. Non-binary turbo codes over high-order fields have been recently proposed in [30], which are derived from a particular protograph sub-ensemble of the (2, 3) regular LDPC code ensemble. Their main advantage would be the development of the corresponding turbo encoder that is simpler than the corresponding non-binary LDPC code. To the contrary, our design approach directly focuses on the structure of turbo codes, and furthermore, it is extendable to the code design for high-order modulation.

Very recently, capacity-approaching LDPC and polar code designs for coded modulation together with their constellation shaping have been studied in [31]–[34]. We do not address a constellation shaping to fill the gap between the Shannon limit and the constellation constrained capacity in this work. Nevertheless, since our code design for coded modulation does not assume any modulation format, it may be applicable to a non-equispaced signal constellation to achieve shaping gain, as is adopted in the ATSC 3.0 standard [35].

This paper is a substantial extension of our conference paper [36] and provides a new method for EXIT chart analysis without assuming any specific message model, whereas the permutation-invariance property of the message was assumed in [36] for simplicity. Furthermore, the code design is extended to the high-order modulation, where the optimality of our turbo codes is demonstrated even in a high spectral efficiency regime. Although another conference paper by the authors [37] also addressed coded modulation based on non-binary turbo codes, its code design relies on the conventional EXIT chart analysis; furthermore, the field size of the non-binary convolutional encoder in [37] should be identical to the constellation size, and thus is not applicable to the case with BPSK transmission.

In summary, the main contributions of this paper are as follows:

- The EXIT chart-based analysis for specific non-binary turbo codes is developed. Since we focus on specific codes (i.e., with fixed code coefficients), the common assumption of the message distribution valid for the code ensemble [14], [17] does not hold. Therefore, we propose a novel method of analyzing the decoding convergence without any assumption on the message distribution.
- A code search algorithm that efficiently finds the best codes in terms of both the convergence of iterative decoding and the resulting error floor is introduced.

The code search procedure consists of two steps that are based on the truncated union bounds and the proposed EXIT chart analysis.
- Numerical results in terms of frame error rate (FER) performance over the binary input additive white Gaussian noise (AWGN) channel reveal that our non-binary turbo codes outperform binary turbo codes, polar codes, as well as non-binary LDPC codes defined over similar field size. Furthermore, the code design is extended to the coded modulation system and we demonstrate that our turbo codes with quadrature amplitude modulation (QAM) can outperform the conventional TTCM.

This paper is organized as follows. The non-binary turbo codes defined over finite fields are introduced in Section II, where the non-binary messages as well as channel models are also described. In Section III, we introduce our proposed EXIT chart analysis for non-binary turbo codes with fixed code coefficients, followed by Section IV that derives the truncated union bound. Section V is devoted to identification of optimal component codes through the proposed two-step optimization algorithm based on the combination of EXIT charts and union bounds. In Section VI, the code design is extended to the case with high-order modulation. The simulation results with binary and high-order modulation in Section VII reveal that the proposed turbo codes can achieve the FER performance superior to the conventional schemes. Finally, Section VIII concludes our work.

II. TURBO CODES OVER FINITE FIELD

We begin with the description of our proposed turbo codes based on recursive convolutional codes (RCCs) over the finite field.

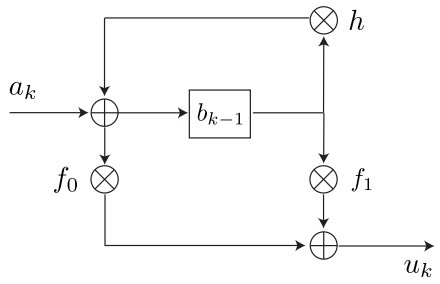


FIGURE 1. The recursive convolutional code defined over GF(2^p).

A. CONVOLUTIONAL CODES OVER FINITE FIELD

Figure 1 shows a block diagram of the convolutional code with a single memory element defined over GF(2^p). Both feedback and feedforward coefficients, denoted by h and $\{f_0, f_1\}$, respectively, are chosen from GF(2^p). The output u_k of the encoder is expressed by a function of a given input symbol $a_k \in$ GF(2^p) and the previous symbol $b_{k-1} \in$ GF(2^p) stored in the memory as

$$u_k = f_1 \otimes b_{k-1} \oplus f_0 \otimes (a_k \oplus h \otimes b_{k-1}), \tag{1}$$

where $\oplus$ and $\otimes$ denote addition and multiplication over GF(2^p), respectively, and thus $u_k \in$ GF(2^p).

Similar to conventional binary turbo codes, our non-binary turbo codes consist of two identical constituent RCCs and a single symbol-wise interleaver as shown in Figure 2.

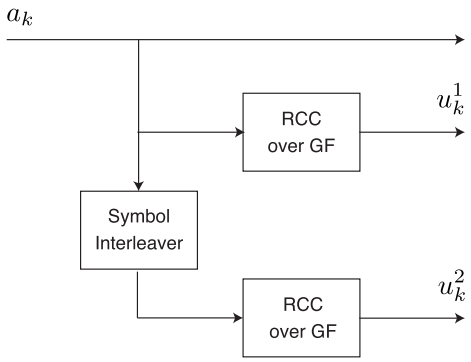


FIGURE 2. Parallel concatenation of the RCCs over the finite field.

B. CHANNEL MODEL AND ITS SYMMETRY

Throughout this paper, we consider the transmission over the symmetric-output AWGN channel. In what follows, we consider the transmission of 2^p-ary symbol $v \in$ GF(2^p) with either BPSK modulation or 2^p-ary modulation.

1) BINARY MODULATION

We first consider the transmission of 2^p-ary symbol in binary format. In this case, symbol v is mapped onto p BPSK symbols denoted by $\boldsymbol{x} = [x_0, x_1, \ldots, x_{p-1}] \in \{-1, 1\}^p$. with its noisy version given by $\boldsymbol{y} = [y_0, y_1, \ldots, y_{p-1}] \in \mathbb{R}^p$. The channel log-likelihood ratio (LLR) is defined as $\boldsymbol{L}^{\mathrm{ch}} = [L_0^{\mathrm{ch}}, L_1^{\mathrm{ch}}, \ldots, L_{2^p-1}^{\mathrm{ch}}]$ where

$$L_i^{\mathrm{ch}} = \log \frac{P(\boldsymbol{y} \mid v = i)}{P(\boldsymbol{y} \mid v = 0)} \tag{2}$$

with $i \in$ GF(2^p)$\backslash 0 = \{1, \ldots, 2^p - 1\}$. (Note that with this definition $L_0^{\mathrm{ch}} = 0$.)

Under the assumption that each bit derived from the same non-binary symbol is affected by statistically independent noise, the symbol-wise LLR vector is expressed as

$$L_i^{\mathrm{ch}} = \sum_{j=0}^{p-1} \log \frac{P(y_j \mid v = i)}{P(y_j \mid v = 0)}. \tag{3}$$

For a binary-input AWGN channel, the bit-wise LLR can be modeled by the independent and identically distributed (i.i.d.) Gaussian random variable with mean $2/\sigma^2$ and variance $4/\sigma^2$, where σ^2 is the variance of the channel noise. Therefore, the channel LLR is also Gaussian distributed, since it consists of the summation of i.i.d. Gaussian random variables.

The transmission of 2^p-ary symbols over binary-input AWGN is derived from the 2^p-ary-input symmetric-output memoryless channel [14, Definition 1], which is an extension

of the binary-input symmetric-output channel to the 2^p-ary input case.

2) 2^p-ary MODULATION

Let $\mathcal{X} \in \{X_0, X_1, \ldots, X_{2^p-1}\}$ denote the set of 2^p-ary constellation points, with each element corresponding to one of GF(2^p) symbols. The symbol $v \in$ GF(2^p) has a one-to-one correspondence with the transmitted signal $\boldsymbol{x} = [x]$, $x \in \mathcal{X}$, and the received signal is denoted by $\boldsymbol{y} = [y]$, $y \in \mathbb{C}$. In this case, the channel LLR is expressed as

$$L_i^{\mathrm{ch}} = \log \frac{P(y \mid x = X_i)}{P(y \mid x = X_0)}. \tag{4}$$

For high-order modulation such as QAM, the channel is not in general symmetric. Therefore, we assume the use of the *i.i.d. channel adapter* that enforces the channel to be symmetric for simplicity of analysis. The use of high-order modulation together with the channel adapter is discussed in Section VI.

C. MAP DECODING

The proposed non-binary turbo codes can be efficiently decoded by the symbol-wise MAP decoder [38]. Each constituent symbol-based MAP decoder operates with the symbol LLR defined as $\boldsymbol{L} = [L_0, L_1, \ldots, L_{2^p-1}]$ where

$$L_i = \log \frac{P(a = i)}{P(a = 0)}. \tag{5}$$

The number of the trellis states and that of the branches in each state for our RCCs are both equal to 2^p.

In the following section, we assume the symmetry of the MAP decoding. Since we consider the symmetric channel, this assumption indicates that the extrinsic information is symmetric. The definition of symmetry of non-binary message is defined as [14]

$$P(\boldsymbol{L} \mid v = i) = e^{-L_i} P(\boldsymbol{L} \mid v = 0), \tag{6}$$

where $i \in$ GF(2^p)\0. Under the symmetric message assumption, the probability of decoding error is independent of the transmitted codeword, and thus we can perform the analysis assuming that the all-zero codeword is transmitted.

III. EXIT CHART ANALYSIS FOR NON-BINARY TURBO CODES

The EXIT chart is a useful tool for the system design and analysis [39], with which we can estimate the convergence behavior of the iterative decoding from the mutual information transfer characteristic of the constituent MAP decoder. In order to evaluate this characteristic, it is necessary to define a specific model that represents the message distribution and Gaussian distribution is often adopted for its simplicity. For non-binary turbo codes with fixed coefficients, however, such a modeling approach may not be accurate. Therefore, in this section, we propose a new EXIT chart analysis where message samples are generated by the Monte-Carlo approach. We first describe how to generate these messages, followed by the detailed description of the proposed EXIT chart analysis.

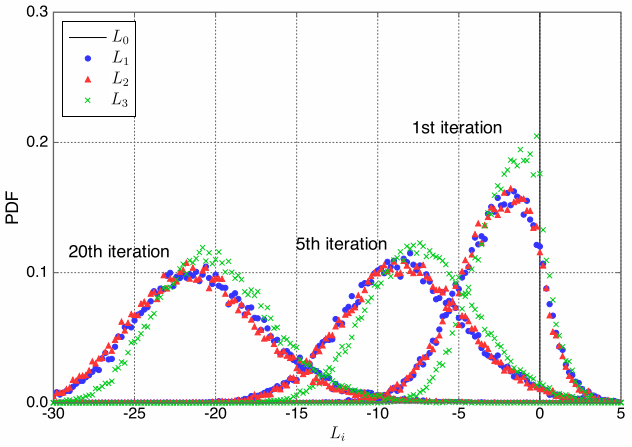


FIGURE 3. Evolution of empirically measured extrinsic information histogram with its iteration for turbo codes over GF(4).

A. PROPOSED MODEL FOR EXIT CHART ANALYSIS

For binary codes, the message in the iterative decoding is expressed by a single scalar value, and its distribution is well approximated by a simple Gaussian distribution. On the other hand, the message for non-binary codes are in vector form and modeling such a vector is not straightforward. Figure 3 shows the evolution of histogram of the message vector $\boldsymbol{L}$ with iterations for GF(4) turbo codes with BPSK modulation obtained by simulations. From this figure, we observe that each element of message vector has its own distinct parameter. Since these parameters depend on specific codes, their analytical identification prior to the actual code search would be infeasible. Therefore, we attempt to produce the message samples by a simple Monte-Carlo approach.

Figure 4 shows the proposed EXIT chart model for evaluating the mutual information transfer characteristic of the constituent MAP decoder where the two streams are concatenated through the interleaver. At each stream, an information sequence is encoded by a test non-binary turbo encoder, i.e., a recursive systematic convolutional (RSC) encoder, and its output is mapped onto the constellation point. After that, they are transmitted over the AWGN channels with different noise levels σ and σ'. This indicates that the reliabilities of channel LLRs used in each MAP decoder are different.

Although a single MAP decoder is used to obtain the EXIT curve in binary case, we use two MAP decoders. The auxiliary MAP decoder shown at the bottom is used for the purpose of generating the *extrinsic* information $\boldsymbol{L}_e'$ that serves as the *a priori* information $\boldsymbol{L}_a$ at the next decoder after interleaving. We note that the permutation of the message does not change the mutual information. This operation is unnecessary if the message distribution is already known and the specific message model is assumed. However, since this is not the case for our model, we generate the message by

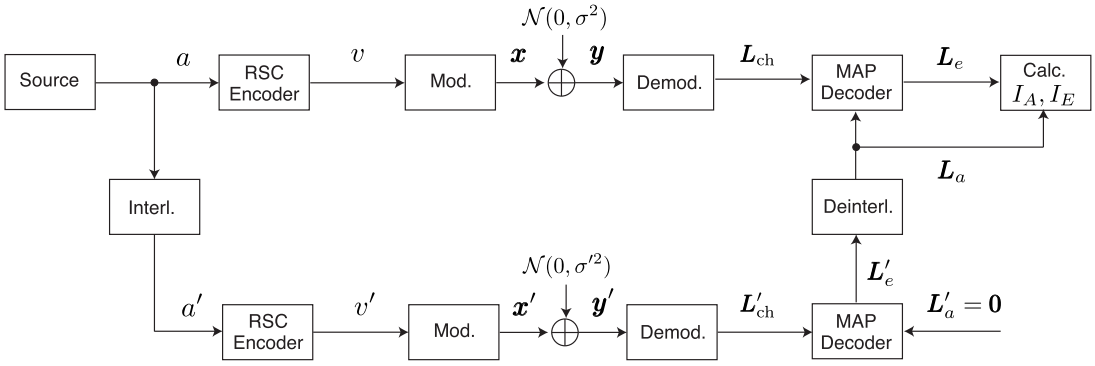


FIGURE 4. The EXIT chart analysis model.

actual simulation. It is also important to note that we do not use the *a priori* information at this decoder, i.e., $\boldsymbol{L}'_a$ is always zero, and thus the knowledge of the message distribution is unnecessary. MAP decoding is then performed at the upper decoder to produce the *extrinsic* information $\boldsymbol{L}_e$ and obtain the decoder characteristic.

B. EXIT CHART ANALYSIS

1) DEFINITION OF MUTUAL INFORMATION

EXIT chart analysis tracks the mutual information of the message in the iterative decoding. Let $I(a;\boldsymbol{L})$ denote the mutual information between the information symbol $a \in \mathrm{GF}(2^p)$ and the corresponding LLR $\boldsymbol{L}$. The mutual information corresponding to the *a priori* and *extrinsic* information can be expressed as

$$I_A = E[I(a;\boldsymbol{L}_a)], \quad I_E = E[I(a;\boldsymbol{L}_e)], \tag{7}$$

respectively, where the expectation is taken over the information sequence. Similarly, we also denote $I'_A = I(a;\boldsymbol{L}'_a)$ and $I'_E = I(a;\boldsymbol{L}'_e)$ for $\boldsymbol{L}'_a$ and $\boldsymbol{L}'_e$, respectively.

Assuming the ergodicity of the extrinsic information, the mutual information $I(a;\boldsymbol{L})$ is calculated by

$$I(a;\boldsymbol{L}) = 1 - E\left[\log_q\left(1+\sum_{i=1}^{q-1} e^{L_i}\right)\middle| a=0\right], \tag{8}$$

where the statistical averaging will be replaced by time averaging [17]. Note that we consider the conditional distribution $P(\boldsymbol{L} \mid a = 0)$ assuming the symmetry of the message, and $I(a;\boldsymbol{L})$ takes the value of $0 \le I(a;\boldsymbol{L}) \le 1$ since the base is q. Also, the mutual information of (8) is related with probabilities of non-zero symbols, i.e., symbol error probability, and the mutual information increases as the corresponding symbol error probability reduces.

2) RELATIONSHIP BETWEEN I_A AND σ'

Naturally, the question is how to select the parameter σ'. In the conventional EXIT chart analysis, the parameter of the extrinsic information corresponding to the desired mutual information I_A is obtained by using the inverse J-function [15]. However, in our case, we perform actual MAP decoding to generate $\boldsymbol{L}_a$, and thus the relationship between I_A and σ' depends on specific code coefficients. Furthermore, it is difficult to evaluate this relationship for each code in the code search. Therefore, we set the range of σ' to examine based on its channel capacity.

The relationship between I_A and σ' depends on the characteristic of the auxiliary MAP decoder in Figure 4. We represent this decoder function by $I_A = f(\sigma')$. Although this decoder function may also have $\boldsymbol{L}'_a$ as its parameter, the *a priori* information at this decoder is always set to zero, i.e., $\boldsymbol{L}'_a = \boldsymbol{0}$ in our analysis. Therefore, we do not take $\boldsymbol{L}'_a$ into account.

Note that I_A is monotonically decreasing with σ'. In other words, we have the following theorem:

Theorem 1: The decoder function $f(\sigma)$ is monotonically decreasing.

Proof: The proof can be derived in the similar way to that of [13, Proposition 1] and based on the fact that the auxiliary MAP decoder is optimal in the sense that for a given σ, it minimizes the symbol error and, equivalently, maximizes $f(\sigma)$. □

3) DETAILED DESCRIPTION OF ANALYSIS

Next, we describe how to select the parameter σ' in our analysis. Let $\sigma(\gamma)$ denote the noise level corresponding to the channel SNR γ expressed in decibel. Since the constituent MAP decoder will not improve the mutual information at the channel SNR below the channel capacity, we set σ' above $\sigma(\gamma_{\rm cap})$, where $\gamma_{\rm cap}$ is the minimum channel SNR required to achieve channel capacity given in decibel. In this work, we first set σ' to the noise level corresponding to $\gamma_{\rm cap}$ plus an offset (0.5 dB in our case) and change it in the logarithm domain. We empirically set the maximum value of the channel SNR to $\gamma_{\rm cap} + 3.5$ dB, and adjust it by the step of 0.5 dB, i.e., $\sigma' \in \{\sigma(\gamma_{\rm cap} + 0.5\,\mathrm{dB}), \sigma(\gamma_{\rm cap} + 1.0\,\mathrm{dB}), \ldots, \sigma(\gamma_{\rm cap} + 3.5\,\mathrm{dB})\}$. Note that by Theorem 1, I_A increases if σ' decreases, and *vice versa*. Although I_A may not be linearly decreasing with respect to σ' in general, the linearity property is not important for the purpose of identifying the convergence.

The above parameter selection will be adopted in the EXIT chart analysis, regardless of code coefficients, the field size, and the modulation format.

We note that our setting of σ' may not cover the full range of $0 \leq I_A \leq 1$, i.e., we may have $f(\sigma(\gamma_{\text{cap}}+3.5\,\text{dB})) > 0$ and $f(\sigma(\gamma_{\text{cap}}+0.5\,\text{dB})) < 1$ in general. Nevertheless, the decoder characteristic around $I_A = 0$ and $I_A = 1$ would not be critical to identify the decoding convergence.

In the iterative decoding, the extrinsic information output from a constituent MAP decoder serves as the *a priori* information for the next decoder. Since our iterative decoder consists of two identical MAP decoders, it is sufficient to examine the relationship between I_A and I_E at the constituent MAP decoder to estimate the decoding convergence. If $I_A < I_E$ is always satisfied for every value of I_A, it indicates that the extrinsic information reaches the maximum level, and thus error free performance could be achieved as decoding iteration increases for a given channel noise level σ.

IV. DERIVATION OF THE UNION BOUND

By assuming the transmission of the all-zero codeword, in this section we derive the union bound on the FER for non-binary turbo codes. The distance spectrum in terms of Hamming distance is sufficient for evaluation of the performance of turbo codes over binary-input AWGN channel. Nevertheless, we focus on the Euclidean distance considering its extension to the non-binary signaling such as high-order QAM. Union bound on the word error probability of turbo codes over AWGN channel is given by

$$\begin{aligned} P_e &\leq \sum_{d=d_{\min}}^{\infty} A_d^{\text{Turbo}} P_d \\ &= \sum_{d=d_{\min}}^{\infty} A_d^{\text{Turbo}} Q\left(\sqrt{\frac{d^2}{2N_0}}\right) \end{aligned} \tag{9}$$

where A_d^{Turbo} is the number of the codewords whose output has the Euclidean distance of d from that of the all-zero codeword. Note that when BPSK signals $\{-1, 1\}$ are used as a modulation format, the squared Euclidean distance of symbol $x \in \text{GF}(2^p)$ from symbol 0 is four times the Hamming weight of binary image of x. Also, P_d is the pairwise error probability (PEP) that the decoder makes an erroneous decision by selecting an error sequence of Euclidean distance d, and $d_{\min}$ is the code minimum Euclidean distance. Upon finding the error coefficient of non-binary turbo codes A_d^{Turbo}, we first derive the weight distribution of the constituent non-binary RCCs. Since non-binary turbo codes are linear codes, we approximate the complete transfer function with all the paths that diverge from the zero state of both constituent RCCs and re-marge into the zero state after some steps [9]. It is shown in [9] that this approximation is accurate when the interleaver length is significantly larger (i.e., 10 times) than the number of memory elements.

A. DISTANCE SPECTRUM OF COMPONENT RECURSIVE CONVOLUTIONAL CODES

We start with deriving the 2^p-dimensional error coefficient $A_{\boldsymbol{\omega},z}$ for the constituent non-binary RCCs based on its trellis representation, which corresponds to the number of the codewords having the input weight vector $\boldsymbol{\omega} = (\omega_1, \omega_2, \ldots, \omega_{2^p-1})$ and squared Euclidean distance z of the parity symbol from the correct path. Each element ω_i, with $i \in \{1, 2, \ldots, 2^p - 1\}$, represents the number that the ith symbol appears in the information sequence. The information symbol length will be denoted by N in what follows.

Upon calculation of the error coefficients for symbol-interleaved turbo codes, it is important to note that one should take into account the symbol-wise input weight vector of the constituent RCCs which indicates the number of non-zero symbols in the input sequence. However, the knowledge of the symbol-wise parity weight is not required. Therefore, we store only the squared Euclidean distance of parity symbols from the correct path. In the following, the squared Euclidean distance of the parity symbol x from symbol 0 is denoted by $\mathcal{E}(x)$. As mentioned, $\mathcal{E}(x)$ is four times the corresponding Hamming weight for BPSK modulation.

Following the notation in [19], we first denote the number of the paths reaching the state S at the time index t as $A_{t,S,\boldsymbol{\omega},z}$, where $\boldsymbol{\omega}$ is the input symbol weight vector and z is the squared Euclidean distance of parity symbol from the correct path. For each time index t, the coefficient $A_{t,S,\boldsymbol{\omega},z}$ is updated recursively as

$$A_{t,S,\boldsymbol{\omega},z} = \sum_{S',S:u_t,x_t} A_{t-1,S',\boldsymbol{\omega}',z'}, \tag{10}$$

where u_t is the input symbol that triggers transition from the state S' (at $t-1$) to the state S (at t), x_t is the corresponding output symbol, and the vector of the input weight stored in the state S' is given by $\boldsymbol{\omega}' = (\omega'_1, \omega'_2, \ldots, \omega'_{2^p-1})$. In this calculation process, we update the input weight vector as

$$\omega_i = \omega'_i + 1, \quad \text{for } i = u_t. \tag{11}$$

Similarly, we update the parity weight z as

$$z = z' + \mathcal{E}(x_t), \tag{12}$$

where z' is the squared Euclidean distance stored in the state S'. Finally, we approximate the error coefficients of the constituent RCCs, denoted by $A_{\boldsymbol{\omega},z}^{\text{RCC}}$, as

$$A_{\boldsymbol{\omega},z}^{\text{RCC}} = A_{N,0,\boldsymbol{\omega},z}. \tag{13}$$

The above approximation ensures that the complete distance spectrum is characterized by all the paths of length N that once diverge from the zero state and then re-merge into the zero state.

B. DISTANCE SPECTRUM OF AVERAGE TURBO CODES

Based on the error coefficients of the constituent RCCs, the error coefficients of non-binary turbo codes will be derived in

what follows. Considering the fact that the error coefficients of turbo codes depend on the specific interleaver structure, we assume the use of *uniform interleaver* [9] for the sake of simplicity. The total number of non-zero symbols in the information sequence is denoted by $\omega = \omega_1 + \ldots + \omega_{2^p-1}$. Then, the error coefficient of the equivalent block code with the uniform interleaver of size N can be calculated as

$$A_d^{\text{Turbo}} = \sum_{(e,z_1,z_2):\sqrt{e+z_1+z_2}=d} \frac{(N-\omega)!\,\omega_1!\cdots\omega_{2^p-1}!}{N!} \times A^{\text{RCC}}_{\boldsymbol{\omega},z_1} \cdot A^{\text{RCC}}_{\boldsymbol{\omega},z_2}, \tag{14}$$

where $A^{\text{RCC}}_{\boldsymbol{\omega},z}$ is the error coefficient of component convolutional codes and $e = \sum_{i=1}^{2^p-1} \mathcal{E}(i)\,\omega_i$ is the total squared Euclidean distance of input sequence from the all-zero sequence. The union bound for average turbo codes can be obtained by substituting (14) into (9). Note that the summation of (14) is over all the combinations of (e, z_1, z_2) that satisfy $d = \sqrt{e + z_1 + z_2}$.

C. TRUNCATED UNION BOUND

Since our objective of using union bound is to estimate the error floor, the error event that has large output Euclidean distance from all-zero codeword may be excluded. Furthermore, calculation of the distance spectrum of the constituent RCCs based on the above procedure would become computationally challenging as the number of memory elements and codeword length increase. Therefore, we truncate the error events corresponding to large output Euclidean distances. Specifically, we perform trellis-based search with the M-algorithm, where only M branches in each trellis section associated with the smallest metrics are connected to the node at the next level.

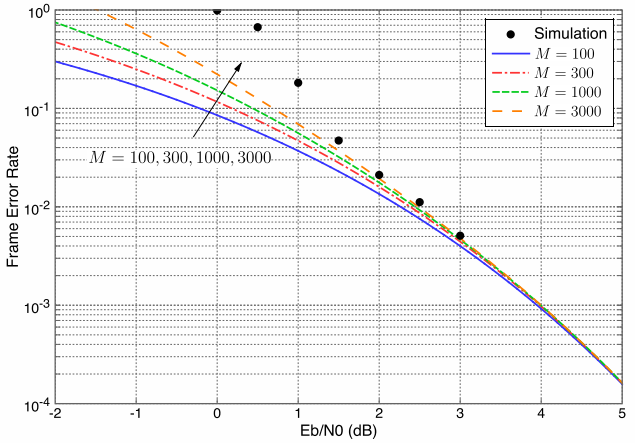


FIGURE 5. Simulated FER for the rate-1/3 non-binary turbo codes over GF(4) with BPSK modulation and the corresponding truncated union bounds with $N = 100$ and various values of parameter M.

The computational complexity of the trellis-based search with the M-algorithm depends on the choice of the trellis length N and parameter M. Figure 5 demonstrates the effect of the parameter M on the error rate performance of turbo codes over GF(4) with BPSK modulation. From this figure, we observe that a large number of M is not necessarily required to capture the error floor. This indicates that the search for the error floor with the M-algorithm requires neither high computational complexity, nor large memory to store the information on M branches. For the code search presented in the following section, we set $N = 100$ and $M = 1000$. This trellis length is sufficient to capture the error floor performance, since the length of error events that affect the error floor is typically much shorter than 100. However, the required value of M to estimate the error floor may depend on the field size, and we set $M = 1000$ for all the cases for simplicity.

V. COMPONENT CODE SEARCH

For parallel or serially concatenated codes, the component codes have to be chosen properly in order to achieve a good performance. In this section, the code search algorithm for finding component code is described based on the EXIT charts and truncated union bounds. Since the objective in this section is to design the non-binary turbo codes that can achieve a good iterative decoding threshold as well as the asymptotic performance simultaneously, we perform the code search in two steps. We first apply the truncated union bound presented in Section IV to all the possible sets of code coefficients and analyze the asymptotic performance. After that, we apply the proposed EXIT chart analysis described in Section III to the reduced search space and choose the best codes that can achieve good trade-off between the iterative decoding threshold and asymptotic performance.

There are mainly two reasons for applying the truncated union bound in the first step, rather than the EXIT charts. The first reason is that the union bound analysis always finds the best codes in terms of the error floor, whereas the results of the EXIT chart analysis is probabilistic, since it resorts to the Monte-Carlo simulation. Therefore, in order to perform an analysis with acceptable accuracy, the proposed EXIT chart analysis may require a long information sequence, which results in high computational complexity of analysis. As the second reason, the truncated union bound, on the other hand, can be easily derived (less time consuming) by applying the M-algorithm as explained in the previous section. Since the main objective of the first step is to reduce the large search space to a limited number of candidates, the truncated union bound would be more appropriate.

In the first step, we aim to find codes that achieve good error floor performance, which is equivalent to maximizing $d_{\min}$ and minimizing $A^{\text{Turbo}}_{d_{\min}}$ in (14). If multiple codes achieve the same values of $d_{\min}$ and $A^{\text{Turbo}}_{d_{\min}}$, the next term $A^{\text{Turbo}}_{d_{\min}+1}$ is compared, where higher priority will be given to the codes having smaller coefficients. In this manner, we may find the best codes in terms of the error floor among all possible combinations of code coefficients.

Note that the codes that have the lowest error floor performance do not necessarily have the best decoding convergence performance. Therefore, we temporarily store the best N_{temp}

TABLE 1. Best memory-1 component codes for BPSK modulation.

Field	Encoder $(h,\ f_0,\ f_1)$	Primitive polynomial	Threshold		
			Estimated (dB)	Actual (dB)	Gap (dB)
GF(8)	$(\alpha,\ \alpha^2,\ \alpha^4)$	$p(x) = x^3 + x + 1$	-0.08	0.16	0.24
GF(16)	$(\alpha^{12},\ \alpha^4,\ \alpha^8)$	$p(x) = x^4 + x + 1$	-0.20	-0.07	0.13
GF(32)	$(\alpha^{18},\ \alpha^{13},\ \alpha^5)$	$p(x) = x^5 + x^2 + 1$	-0.21	-0.12	0.09
GF(64)	$(\alpha^6,\ \alpha^{41},\ \alpha^{15})$	$p(x) = x^6 + x + 1$	-0.18	-0.05	0.13

candidates from the above process, and select the best one in terms of both decoding convergence and error floor by applying the EXIT chart analysis to all of the N_{temp} candidates.

More specifically, in the second step, for each of N_{temp} candidates we compute the channel SNR where its EXIT curve becomes open. We denote this channel SNR, i.e., the iterative decoding threshold by SNR_{TH}. For each set of tentative test coefficients, we first assign the SNR value at the channel capacity of the corresponding code rate to the initial value. Then, the EXIT curve is computed. If the EXIT curve has an open tunnel, then the SNR_{TH} is set to the current channel SNR. If the EXIT curve is not open, the channel SNR is increased by 0.1 dB and EXIT curve is computed again. We apply these processes to all N_{temp} candidates. Finally, the component code that achieves the smallest metric SNR_{TH} will be selected as the best code.

Our proposed code search procedure is summarized as follows:

- Step 1 (Union bound)
  1) Derive the error coefficients of turbo code A_d^{Turbo}.
  2) If the minimum distance $d_{\min}$ or corresponding coefficients A_d^{Turbo} of tentative test coefficients is within the top N_{temp} candidates, the coefficient is stored.
  3) The above process continues until all possible candidates are tested.
- Step 2 (EXIT chart)
  1) Set the initial channel SNR to the corresponding SNR suggested by the channel capacity.
  2) Compute the EXIT curve.
  3) If the EXIT curve has an open tunnel, SNR_{TH} is set to the current channel SNR. Otherwise, increment the channel SNR by 0.1 dB and go back to 2).
  4) Apply the above process to all N_{temp} candidates and compute SNR_{TH}.
  5) Select the best code that achieves the lowest value of SNR_{TH}.

In the code search process, we use the following setup: the information length for EXIT chart analysis is 10000 symbols, and the number of tentative candidates is $N_{\text{temp}} = 30$. The best memory-1 component codes for non-binary turbo codes found by the above procedure are tabulated in Table 1.

VI. HIGH-ORDER MODULATION WITH CHANNEL ADAPTER

In this section, we consider the transmission of 2^p-ary constellation matched to 2^p-ary codewords of turbo codes over the AWGN channel. Since the symmetry property may not hold, the analysis assuming the all-zero codeword may not be accurate. Therefore, we resort to the i.i.d. channel adapter [40], which was proposed as an analytical tool that enforces the channel to be symmetric. This technique is also applicable to the non-binary case, where channel codes are defined over GF(2^p). At the output of the encoder, the channel adapter adds a symbol chosen from GF(2^p) with equal probability to a coded symbol, and this effect is taken into account at the receiver by permuting the channel LLR prior to decoding. Since the channel adapter averages out the effect of a coded symbol over GF(2^p), the physical channel can be seen as symmetric. (The reader is referred to [14, Th. 7] for the details and the proof of symmetry property.) Thanks to this adapter, the physical channel would become symmetric regardless of the modulation format, and we thus perform the analysis assuming that the all-zero codeword is transmitted analogous to the case with binary modulation.

Although the EXIT chart analysis developed in Section III is applicable to the cases of high-order modulation in a straightforward manner, a slight modification is required in the derivation of the union bound. Here, we consider $A_{\boldsymbol{\omega},e,z}$ for the case with channel adapter, which now describes the number of the codewords that have the input weight vector $\boldsymbol{\omega}$ as well as the input and output squared Euclidean distances, denoted by e and z, respectively, from the all-zero sequence. The *input* squared Euclidean distance e from the correct path is introduced here, since unlike the symmetric channel, the input squared Euclidean distance from the all-zero codeword is not determined by the input symbol weight vector due to the channel adapter, and thus the evaluation of e separately from the input symbol weight vector $\boldsymbol{\omega}$ is necessary.

By introducing the i.i.d. symbols v_t^0 and v_t^1 that are uniformly distributed over GF(2^p) and independent of the output symbol x_t, the error coefficient $A_{\boldsymbol{\omega},e,z}$ can be averaged over all possible combinations of the i.i.d. symbols v_t^0 and v_t^1. Consequently, it can be calculated for each time index t as

$$A_{t,S,\boldsymbol{\omega},e,z} = \frac{1}{2^{2p}} \sum_{v_t^0} \sum_{v_t^1} \sum_{S',S:u_t+v_t^0,x_t+v_t^1} A_{t-1,S',\boldsymbol{\omega}',e,z'}, \tag{15}$$

where, similar to (10), the transition from the state S' (at $t-1$) to the state S (at t) is considered with u_t representing the input symbol and x_t the corresponding output symbol. The difference from that in Section IV is that now the i.i.d. symbols v_t^0 and v_t^1 are added to the input u_t and the

output x_t, respectively. In this case, the squared Euclidean distance of the parity symbol x_t from symbol 0, i.e., $\mathcal{E}(x_t)$ used in (12), is expressed as

$$\mathcal{E}(x_t) = \left| X_{0+v_t^1} - X_{x_t+v_t^1} \right|^2, \tag{16}$$

where $|\cdot|$ indicates the Euclidean norm. The squared Euclidean distance of systematic symbol s_t from symbol 0 is also updated in a similar way to that of parity symbol based on $\mathcal{E}(s_t) = \left| X_{0+v_t^0} - X_{s_t+v_t^0} \right|^2$. Consequently, the error coefficient of turbo codes is given by the form analogous to (14).

Since the i.i.d. channel adapter averages out the effect of feedforward coefficients, the union bound depends only on the feedback coefficient. Therefore, we first optimize the feedback coefficient by the union bound and best feedforward coefficients are chosen by the EXIT charts.

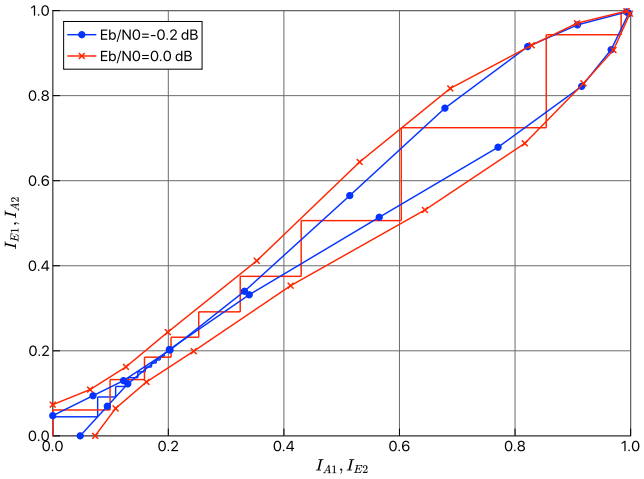


FIGURE 6. The EXIT curve and average decoding trajectory of rate-1/3 non-binary turbo codes over GF(64) with information length of 10^4 symbols.

VII. SIMULATION RESULTS

A. RESULTS OF EXIT CHART ANALYSIS

Figure 6 compares the EXIT chart as well as the resulting decoding trajectory of the optimized rate-1/3 non-binary turbo codes defined over GF(64) for two different channel SNRs. The code coefficients employed here are listed in Table 1. The axis labels I_{Ai} and I_{Ei} in the figure indicate the mutual information of the *a priori* and *extrinsic* information at the ith MAP decoder, respectively. From this figure, we observe that the EXIT curve well agrees with the actual decoding trajectory and this indicates the accuracy of our analysis. Also, it is observed that the tunnel of the EXIT chart becomes open at the channel SNR around -0.2 dB. From this result, one may predict that the iterative decoding threshold of this code would be around -0.2 dB, which is 0.3 dB away from the channel capacity.

In Table 1, we summarize the iterative decoding threshold estimated through the proposed EXIT chart analysis, the actual channel SNR where the FER curve of extensive simulation reaches FER$= 10^{-2}$, and their gaps. In this simulation, the information length is set as 10^5 bits, and decoding iteration is 20. For GF(8) codes, since the error floor appears above the FER of 10^{-2}, the SNR where the FER curve reaches 10^{-1} is evaluated. From Table 1, it is observed that the gap between the estimated threshold by the EXIT chart analysis and the actual SNR where FER of 10^{-2} is achieved is around 0.1 dB for $2^p > 8$. These gaps will decrease as the information length and decoding iteration increase. These results indicate that the proposed EXIT chart analysis is valid for estimating the behavior of non-binary turbo code in the low SNR region without extensive Monte-Carlo simulation.

B. PERFORMANCE OVER BINARY-INPUT AWGN CHANNELS

We evaluate the FER performance of the proposed non-binary turbo codes assuming the binary-input AWGN channel. The performance is compared with the following three different approaches: 1) 16-state *binary* turbo codes with the optimized generator polynomial 37-33 [41], 2) ultra-sparse non-binary (2, 4)-LDPC codes, and 3) binary polar codes with the cyclic redundancy check (CRC) aided successive list decoding (CASCL) [42]. For these codes, the information length and code rate are set as 1024 and 1/2, respectively. In the case of turbo codes, the code rate of 1/2 is achieved by puncturing of the original rate-1/3 turbo codes, and Log-MAP decoding with 10 iteration count is performed. The so-called S-interleaver with the spread value $S = 22$ is employed. The location of non-zero elements in the parity-check matrix of non-binary LDPC codes is optimized by progressive edge-growth [43], whereas the non-zero elements are randomly selected [44]. We note that when field size is large, e.g., GF(256), the performance improvement by the optimization of non-zero elements is marginal [44]. The maximum number of iteration for the sum-product decoding is 50. The polar codes used here are constructed using the density evolution based on convolutions with the design SNR of $E_b/N_0 = -1.6$ dB, and the list size for CASCL is chosen as $L = 32$ along with 16-bit CRC.

The results are compared in Figure 7. As a benchmark of the performance of finite length codes, we have also plotted Gallager's random coding bound [45]. It is observed from this figure that the proposed non-binary turbo codes achieve the FER of 10^{-3} at the same channel SNR as the non-binary (2, 4)-LDPC codes designed over GF(256) and outperform binary turbo codes by approximately 0.15 dB. Considering the fact that the decoding complexity of non-binary codes substantially increases as the field size increases, this result demonstrates the superiority of our non-binary turbo codes in terms of trade-off between the achievable performance and resulting decoding complexity.

To verify the effectiveness of our code design algorithm, we have also plotted the performance of non-binary turbo codes designed only by the union bound analysis in Figure 7, which is labeled as *UB only*, in contrast to the proposed hybrid design algorithm labeled as *UB + EXIT*. As observed, the

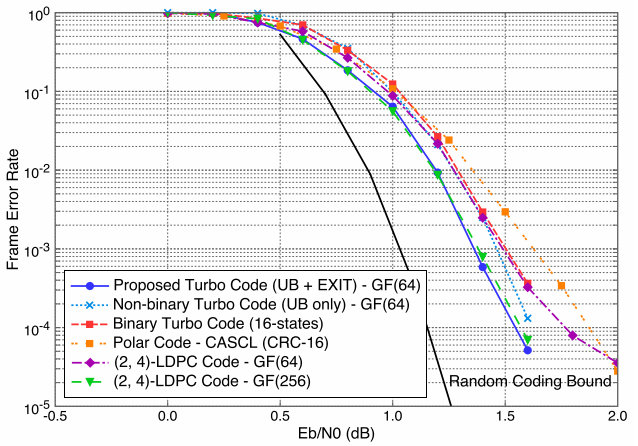


FIGURE 7. Performance comparison of the proposed non-binary turbo codes with other capacity-approaching codes (Information length is 1024 bits and code rate is around 1/2) .

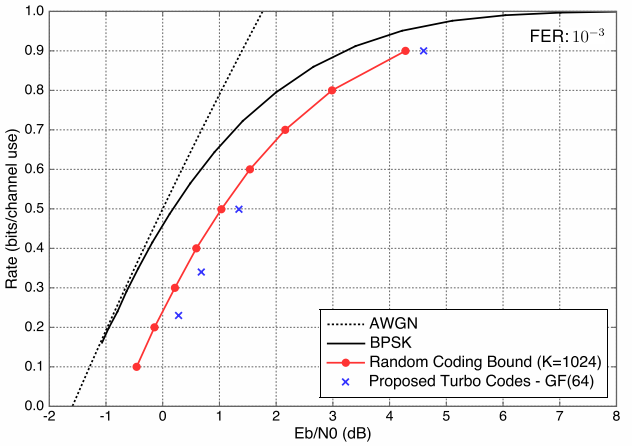


FIGURE 8. Achievable rate of the proposed turbo codes over GF(64) with BPSK modulation.

performance gain of the proposed design algorithm over the design method that only takes the distance spectrum property into account is about 0.15 dB at the FER of 10^{-3}. This demonstrates the effectiveness of our two-step optimization algorithm. Note that we can also design the turbo codes based only on the EXIT chart analysis, but it may be computationally demanding and thus impractical due to its substantial search space as mentioned earlier.

C. EXTENSION TO LOWER CODE RATE

Although the code rate of turbo codes can be enhanced through puncturing, it is in general difficult for traditional binary turbo codes to achieve capacity-approaching performance with low code rate, e.g., less than 1/3. However, there are important applications of low-rate codes, e.g., multiple-access scheme such as code-division multiple-access (CDMA) and interleave-division multiple-access (IDMA) [46]. Therefore, we show the design example of non-binary turbo codes defined over GF(2^p) whose code rate is given by $m/(3p)$, where m is less than p.

The low code rate of turbo codes can be achieved by restricting the cardinality of its input 2^m to the strict subset of the field GF(2^p), where $m < p$. The input binary sequence is mapped onto the strict subset of GF(2^p) such that the minimum Hamming distance is maximized. In this case, the trellis still has 2^p states and each state has 2^m branches.

Here we consider the design of rate-2/9 turbo codes over GF(64) with $m = 4$ as an example. In this case, we map the 4-bit sequence to the subset of GF(64) symbols such that the minimum Hamming distance is maximized. To do this, we perform this mapping by using the generator matrix of the punctured (6, 4) Hamming code, which has the minimum distance of 2. In this paper, we obtain the low-rate codes by applying the above method to the optimized codes in Table 1, and thus we do not perform additional optimization.

Figure 8 shows the achievable rate of the proposed variable-rate turbo codes with the same simulation setup as that demonstrated in Figure 7, where we plot the required channel SNR for each code to achieve the FER of 10^{-3}. The code rates of 9/10 and 1/2 are achieved by symbol-wise puncturing and 2/9 is designed based on the method described above. We can see from this figure that at any code rate, the proposed codes can achieve the performance within approximately 0.3 dB from the random coding bound. This result demonstrates high flexibility of the proposed turbo codes in terms of code rate design as well as the optimality of its performance.

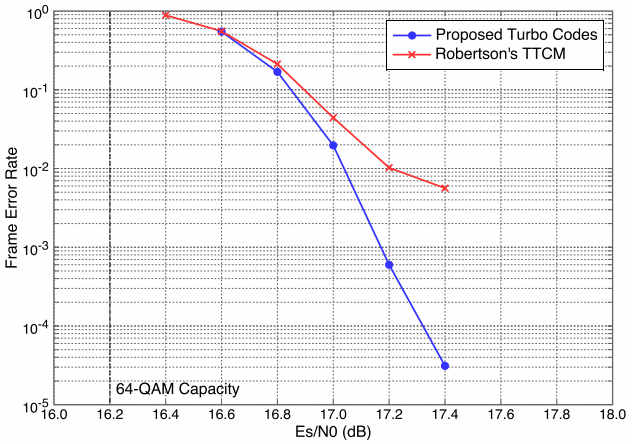


FIGURE 9. Frame error rate performance comparison with the spectral efficiency of 5 bits/symbol. Information block size is 5 × 1000 bits and the decoding iteration count is 8.

D. PERFORMANCE WITH HIGH-ORDER MODULATION

Figure 9 shows the performance comparison with Robertson's TTCM [47] in terms of the FER performance with the spectral efficiency of five bits per symbol. The vertical dashed line in the figure indicates the corresponding constellation constrained capacity (16.2 dB). The TTCM consists of two identical rate-5/6 TCM encoders with 8 trellis states employing 64-QAM that are optimized in [47]. Based on our

code search algorithm modified to high-order modulation, the proposed turbo code is designed over GF(64) with its code coefficients (α^{26}, α^{6}, α^{21}), where the primitive element α is associated with the primitive polynomial in Table 1. The rate-5/6 turbo codes for the proposed scheme are obtained via symbol-wise puncturing, where the puncturing pattern is optimized by the exhaustive search. Information length is set as 5000 bits and decoding iteration is 8. The interleaver employed here is S-random interleaver. We have performed the optimization based on the EXIT chart analysis for a fixed mapping pattern.

From Figure 9, it is observed that the proposed non-binary turbo codes achieve a remarkable gain compared to the conventional TTCM scheme. Specifically, even though Robertson's TTCM suffers from its high error floor, our non-binary turbo codes do not exhibit such an error floor even at the FER of 10^{-4}.

VIII. CONCLUSION

We have tackled with the design issues of non-binary turbo codes. Since the EXIT chart analysis for specific non-binary turbo codes is a non-trivial task, we have proposed a novel approach that does not require the *a priori* knowledge on the message distribution. The two-step code design algorithm based on the EXIT charts and union bounds has been proposed. Simulation results have demonstrated that our proposed non-binary turbo codes achieve performance comparable to the non-binary LDPC codes even with much less decoding complexity. Furthermore, its application to high-order modulation based on the channel adapter has been presented, which can outperform the conventional TTCM scheme.

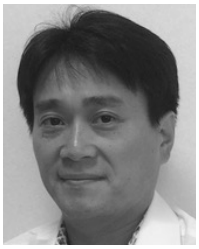
Finally, since we have performed the optimization assuming uniform interleaver in this work, the joint optimization of component RCCs and the interleaver structure may have a room for further performance improvement.

REFERENCES

[1] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes," in *Proc. Int. Conf. Commun. (ICC)*, May 1993, pp. 1064–1070.
[2] R. G. Gallager, "Low-density parity-check codes," *IRE Trans. Inf. Theory*, vol. 8, no. 1, pp. 21–28, Jan. 1962.
[3] D. J. C. MacKay and R. M. Neal, "Near Shannon limit performance of low density parity check codes," *IEEE Electron. Lett.*, vol. 32, no. 18, pp. 1645–1646, Aug. 1996.
[4] 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.
[5] J. Berkmann, "Symbol-by-symbol map decoding of nonbinary codes," in *Proc. ITG-Fachtagung Codierung Quelle, Kanal Ubertragung*, 1998, pp. 95–100.
[6] G. Liva, L. Gaudio, T. Ninacs, and T. Jerkovits. (Oct. 2016). "Code design for short blocks: A survey." [Online]. Available: https://arxiv.org/abs/1610.00873
[7] D. J. C. MacKay and M. C. Davey, "Evaluation of Gallager codes for short block length and high rate applications," in *Proc. IMA Workshop Codes, Syst., Graph. Models*, 1999, pp. 113–130.
[8] S. Hongzin and J. R. Cruz, "Reduced-complexity decoding of Q-ary LDPC codes for magnetic recording," *IEEE Trans. Magn.*, vol. 39, no. 2, pp. 1081–1087, Mar. 2003.
[9] S. Benedetto and G. Montorsi, "Unveiling turbo codes: Some results on parallel concatenated coding schemes," *IEEE Trans. Inf. Theory*, vol. 42, no. 2, pp. 409–428, Mar. 1996.
[10] T. J. Richardson and R. L. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 599–618, Feb. 2001.
[11] M. Luby, M. Mitzenmacher, A. Shokrollah, and D. Spielman, "Analysis of low density codes and improved designs using irregular graphs," in *Proc. 30th Annu. ACM Symp. Theory Comput. (STOC)*, 1998, pp. 249–258.
[12] S.-Y. Chung, T. J. Richardson, and R. L. Urbanke, "Analysis of sum-product decoding of low-density parity-check codes using a Gaussian approximation," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 657–670, Feb. 2001.
[13] H. E. Gamal and A. R. Hammons, "Analyzing the turbo decoder using the Gaussian approximation," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 671–686, Feb. 2001.
[14] G. Li, I. J. Fair, and W. A. Krzymien, "Density evolution for nonbinary LDPC codes under Gaussian approximation," *IEEE Trans. Inf. Theory*, vol. 55, no. 3, pp. 997–1015, Mar. 2009.
[15] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, no. 10, pp. 1727–1737, Oct. 2001.
[16] D. Divsalar, S. Dolinar, and F. Pollara, "Iterative turbo decoder analysis based on density evolution," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 5, pp. 891–907, May 2001.
[17] A. Bennatan and D. Burshtein, "Design and analysis of nonbinary LDPC codes for arbitrary discrete-memoryless channels," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 549–583, Feb. 2006.
[18] J. Kliewer, S. X. Ng, and L. Hanzo, "Efficient computation of EXIT functions for nonbinary iterative decoding," *IEEE Trans. Commun.*, vol. 54, no. 12, pp. 2133–2136, Dec. 2006.
[19] S. X. Ng, O. R. Alamri, Y. Li, J. Kliewer, and L. Hanzo, "Near-capacity turbo trellis coded modulation design based on EXIT charts and union bounds," *IEEE Trans. Commun.*, vol. 56, no. 12, pp. 2030–2039, Dec. 2008.
[20] V. Savin, "Split-extended LDPC codes for coded cooperation," in *Proc. Int. Symp. Inf. Theory Appl. (ISITA)*, Oct. 2010, pp. 151–156.
[21] B. M. Kurkoski, K. Yamaguchi, and K. Kobayashi, "Single-Gaussian messages and noise thresholds for decoding low-density lattice codes," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun./Jul. 2009, pp. 734–738.
[22] M. Takai and K. Ishibashi. (Feb. 2018). "Repeat-accumulate signal codes." [Online]. Available: https://arxiv.org/abs/1802.01133
[23] A. C. Reid, T. A. Gulliver, and D. P. Taylor, "Rate-1/2 component codes for nonbinary turbo codes," *IEEE Trans. Commun.*, vol. 53, no. 9, pp. 1417–1422, Sep. 2005.
[24] J. Berkmann, "On turbo decoding of nonbinary codes," *IEEE Commun. Lett.*, vol. 2, no. 4, pp. 94–96, Apr. 1998.
[25] M. Ferrari and S. Bellini, "Rate variable, multi-binary turbo codes with controlled error-floor," *IEEE Trans. Commun.*, vol. 57, no. 5, pp. 1209–1214, May 2009.
[26] C. Douillard and C. Berrou, "Turbo codes with rate-$m/(m+1)$ constituent convolutional codes," *IEEE Trans. Commun.*, vol. 53, no. 10, pp. 1630–1638, Oct. 2005.
[27] G. S. White and D. J. Costello, "Construction and performance of q-ary turbo codes for use with M-ary modulation techniques," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2000, p. 220.
[28] J. A. Briffa and H. G. Schaathun, "Non-binary turbo codes and applications," in *Proc. 5th Int. Symp. Turbo Codes Rel. Topics (ISTC)*, Sep. 2008, pp. 294–298.
[29] H. Balta, C. Douillard, and R. Lucaciu, "Multi-non-binary turbo codes," *J. Wireless Commun. Netw.*, vol. 2013, p. 279, Dec. 2013, doi: 10.1186/1687-1499-2013-279.
[30] G. Liva, E. Paolini, B. Matuz, S. Scalise, and M. Chiani, "Short turbo codes over high order fields," *IEEE Trans. Commun.*, vol. 61, no. 6, pp. 2201–2211, Jun. 2013.
[31] F. Steiner, G. Böcherer, and G. Liva, "Protograph-based LDPC code design for bit-metric decoding," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2015, pp. 1089–1093.
[32] F. Steiner, G. Liva, and G. Böcherer. (Aug. 2017). "Ultra-sparse non-binary LDPC codes for probabilistic amplitude shaping." [Online]. Available: https://arxiv.org/abs/1708.05558
[33] T. Prinz *et al.*, "Polar coded probabilistic amplitude shaping for short packets," in *Proc. IEEE 18th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Jul. 2017, pp. 1–5.

[34] O. İşcan, R. Böhnke, and W. Xu, "Shaped polar codes for higher order modulation," *IEEE Commun. Lett.*, vol. 22, no. 2, pp. 252–255, Feb. 2018.
[35] N. S. Loghin and J. Zöllner, B. Mouhouche, D. Ansorregui, J. Kim, and S.-I. Park, "Non-uniform constellations for ATSC 3.0," *IEEE Trans. Broadcast.*, vol. 62, no. 1, pp. 197–203, Mar. 2016.
[36] T. Matsumine and H. Ochiai, "A design of non-binary turbo codes over finite fields based on Gaussian approximation and union bounds," in *Proc. IEEE Veh. Technol. Conf. (VTC-Spring)*, Jun. 2018, pp. 1–5.
[37] T. Matsumine and H. Ochiai, "A new turbo coded modulation approach exploiting non-binary field," in *Proc. IEEE Radio Wireless Symp. (RWS)*, Jan. 2018, pp. 72–75.
[38] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inf. Theory*, vol. IT-20, no. 2, pp. 284–287, Mar. 1974.
[39] M. El-Hajjar and L. Hanzo, "EXIT charts for system design and analysis," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 127–153, 1st Quart., 2014.
[40] J. Hou, P. H. Siegel, L. B. Milstein, and H. D. Pfister, "Capacity-approaching bandwidth-efficient coded modulation schemes based on low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 49, no. 9, pp. 2141–2155, Sep. 2003.
[41] S. Benedetto, R. Garello, and G. Montorsi, "A search for good convolutional codes to be used in the construction of turbo codes," *IEEE Trans. Commun.*, vol. 46, no. 9, pp. 1101–1105, Sep. 1998.
[42] I. Tal and A. Vardy, "List decoding of polar codes," *IEEE Trans. Inf. Theory*, vol. 61, no. 5, pp. 2213–2226, May 2015.
[43] X.-Y. Hu, E. Eleftheriou, and D. M. Arnold, "Regular and irregular progressive edge-growth tanner graphs," *IEEE Trans. Inf. Theory*, vol. 51, no. 1, pp. 386–398, Jan. 2005.
[44] C. Poulliat, M. Fossorier, and D. Declercq, "Design of regular (2, d_c)-LDPC codes over GF(q) using their binary images," *IEEE Trans. Commun.*, vol. 56, no. 10, pp. 1626–1635, Oct. 2008.
[45] R. G. Gallager, *Information Theory and Reliable Communication*. Hoboken, NJ, USA: Wiley, 1968.
[46] L. Ping, L. Liu, K. Wu, and W. K. Leung, "Interleave division multiple-access," *IEEE Trans. Wireless Commun.*, vol. 5, no. 4, pp. 938–947, Apr. 2006.
[47] P. Robertson and T. Wörz, "Bandwidth-efficient turbo trellis-coded modulation using punctured component codes," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 2, pp. 206–218, Feb. 1998.

TOSHIKI MATSUMINE (S'16) received the B.E. and M.E. degrees from Yokohama National University, Yokohama, Japan, in 2015 and 2017, respectively, where he is currently pursuing the Ph.D. degree in information and communication engineering. His current research interests include channel coding and coded modulation.



HIDEKI OCHIAI (SM'16) received the Ph.D. degree in information and communication engineering from The University of Tokyo, Tokyo, Japan, in 2001. From 2001 to 2003, he was a Research Associate with the Department of Information and Communication Engineering, University of Electro-Communications, Tokyo. Since 2003, he has been with the Department of Electrical and Computer Engineering, Yokohama National University, Yokohama, Japan, where he is currently a Professor. He served as an Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS from 2007 to 2011 and the IEEE WIRELESS COMMUNICATIONS LETTERS from 2011 to 2016.