Pertanika J. Sci. & Technol. 27 (1): 33 - 47 (2019)

SCIENCE & TECHNOLOGY

Journal homepage: http://www.pertanika.upm.edu.my/

Non-Binary Serial Turbo LDPC Codes Combined with High Order Constellations

Latifa Mostari[1]* and Abdelmalik Taleb-Ahmed[2]

[1]*Department of Electronics, Faculty of Technology, Hassiba Benbouali University of Chlef, Chelf 02000, Algeria*
[2]*IEMN UMR CNRS 8520, Polytechnic University of Hauts-de-France, Valenciennes 59313, France*

ABSTRACT

Given the increasing number of applications requiring high data transmission, this is the reason for the use of high order constellations such as Quadrature Amplitude Modulation (QAM). However, communication systems using QAM require a high signal to noise ratio. To overcome this disadvantage, it is interesting to combine high error correction codes such as Low-Density Parity-Check (LDPC) codes with QAM. Although the LDPC codes are good codes for a system using QAM, concatenation of these codes with iterative decoding is still attractive to construct more powerful codes. In this context, we propose the non-binary serial turbo LDPC code. It is obtained by a serial combination of two identical regular non-binary LDPC codes, separated by an interleaver introducing the diversity. Regular codes were used to avoid the complexity of irregular codes despite that they have better performance than the regular code. Simulation results show that the performance of non-binary serial turbo LDPC code, with 16-QAM, 64-QAM and 256-QAM constellations using Gray mapping under Gaussian and Rayleigh channels, are higher than that of non-binary LDPC codes.

Keywords: Iterative decoding, Low-Density Parity-Check codes, non-binary, serial concatenation, turbo-code

ARTICLE INFO

Article history:
Received: 02 November 2017
Accepted: 19 June 2018
Published: 24 January 2019

E-mail addresses:
l.mostari@univ-chlef.dz (Latifa Mostari)
Abdelmalik.Taleb-Ahmed@univ-valenciennes.fr (Abdelmalik Taleb-Ahmed)
* Corresponding author

INTRODUCTION

With the Internet democratization, mobile, user requirements become increasingly large and diverse. Faced with such requirements, digital communications are an essential solution now. One solution among others, is to increase the spectral efficiency while guaranteeing an unchanged transmission quality.

ISSN: 0128-7680
e-ISSN: 2231-8526

In 1948, Shannon (1948) proved that there was a limit spectral efficiency that we could not overcome if we wanted a transmission without errors. Shannon was able to give a limit without giving the code to correct errors. In order to realize the coding solution, intensive research efforts have been made worldwide. The key is to realize a code to get closer to the Shannon limit, and also to achieve a good trade-off performance/complexity. Until the 80s, the code that achieved the Shannon limit with reasonable complexity was not yet introduced. Two large error correcting code families were imposed: the block codes which were subdivided into several types and convolution codes (Elias, 1955).

It can be shown that the performance of a binary code can be improved with the increasing of the block length, but a large block length increases the complexity (Beermann et al., 2013 & Moision, 2013). One way to resolve this problem is to use concatenated codes. The code concatenation is the combination, in parallel, in serial or hybrid, two or more error correcting codes, convolutional or block, of small to moderate lengths.

Berrou (1993) showed that the performance of concatenated codes could be improved with an iterative decoding. This new scheme of code, called turbo-code can achieve the Shannon limit. Turbo-codes may block turbo-codes or convolutional turbo-codes (Berrou et al., 1993& Pyndiah, 1998) depending on the type of concatenated codes. Thus, depending on the type of concatenation, parallel or serial, we can have parallel or series turbo codes.

After the power of iterative decoding, which was highlighted by the invention of turbo codes. The binary LDPC, which had been neglected because of their complexity, for many years since they were introduced by Gallager in 1962 (Gallager, 1962 & Gallager, 1963), had been rediscovered by Mackay (MacKay et al., 1995) in 1995 Spieleman and others (Sipser et al., 1996) in 1996. LDPC codes are linear block codes based on low-density parity-check matrices, that is to say that the number of non-zero elements of the matrix is much less than the number of 0.

A significant contribution was introduced by Luby and others in 1997 (Luby et al., 1979) which introduced and set the irregular LDPC codes. These later have the main characteristic to perform better than regular code. LDPC codes can be regular or irregular according to the regular or irregular distribution of non-zero elements in the matrix. An LDPC code is called regular if the number of non-zeros elements in each columns and/or in each row of the matrix H, is constant. But if the number of non-zeros in each row or column are not constant the code is called an irregular LDPC code.

In 2002 Davey and Mackey (2002) studied the non-binary LDPC codes. LDPC codes can be binary or non-binary codes according to the non-zeros elements in the parity check matrix. If the non-zeros elements in matrix H, are binary (or non-binary), LDPC codes are binary (or non-binary). Non-binary LDPC codes are designed in high order Galois Fields GF(q) where q is the cardinality of the Galois field. The non-binary LDPC codes perform better than their binary equivalents when the coded block is low to moderate length, or when the modulation used has a high order stats. However, the advantages of using non-binary

LDPC codes involve a significant increase in decoding complexity. More the Galois Field order is higher the complexity becomes important. For a Galois Field GF(q), the complexity is of order $O(q^2)$. Similarly, the memory required for storing messages is of order $O(q)$.

LDPC codes are represented by their parity check matrix, and by a graphical representation, called the Tanner graph corresponds to the parity check matrix. The Tanner graph is a bipartite graph composed of two types of nodes: variable nodes representing the symbols of the coded block and the check nodes represent parity check equations. These two types of nodes are connected by branches according to the non-zero elements of the matrix H. The number of variable and check nodes corresponds, respectively, to the number of matrix columns and rows.

The Tanner graph is used as a transmission medium by the decoder. At first all variable nodes are initialized. After, each check node receives messages arriving from the variable nodes that are connected by her branches, then calculates and sends the resulting message that is related to all messages except the input message that the resulting message will be sent. Then, these same operations are performed by the variable nodes.

In Davey and Mckay (2002), the authors proposed the first practical iterative decoding algorithm for non-binary LDPC codes. This algorithm, called Sum-Product Algorithm (SPA), is an optimal iterative decoding with computational complexity. Several algorithms have been proposed to reduce the complexity of the non-binary SPA (Barnault et al., 2003; Wymeersch et al., 2004; Spagnol et al., 2009), each one with a particular performance/ complexity trade-off, such as: FFT-SPA (Fast Fourier Transform), Min-Sum Algorithm, Extended Min-Sum algorithm (Declercq et al., 2007; Voicila et al., 2010) and the Min-Max Algorithm (Savi, 2008), the Simplified Min-Sum Algorithm (Wang et al., 2013).

Many applications have adopted LDPC codes as industry standards, such as WLANs (IEEE 802.11n), WiMAX (IEEE 802.16e), WiFi, DVB-S2, 10 GBase-T Ethernet (IEEE 802.3an) and the ITU-T standard for networking over power lines, phone lines, and coaxial cable (G.hn/G.9960) (Chandrasetty et al., 2011)

An increasing number of applications requires high-speed transmission without increasing the bandwidth of the transmission channel, i.e. high spectral efficiency transmissions, while guaranteeing an unchanged transmission quality. This is the reason for the use of a system combining a high-order constellation with high errors correcting code. For this system, the QAM, is highly recommended as a high order constellation.

LDPC codes are selected as candidate for 5th generation wireless communications (5G) (Tahir et al., 2017). It is essential to develop a new error correction coding technique for 5G and Satellite communication systems.

Although non-binary LDPC codes are good error-correcting codes for a system using a higher order constellation, QAM, concatenation of these codes with iterative decoding is still attractive to a construct powerful errors correcting codes (Mostari et al., 2018; AlMuaini et al., 2013; Hung et al., 2011; Kumar et al., 2013) with reasonable complexity (Mostari et al., 2018).

The original LDPC codes concatenated in parallel PCGCs (Parallel Concatenated Gallager Codes), were introduced in (Behairy et al., 2000) as a class of concatenated codes in which two LDPC codes are irregular binary LDPC codes having different parameters interact in parallel without interleavers. The interleaver runs as a permutation, it changes the weight distribution of the code. It is therefore useful in increasing the minimum distance of the code. In Behairy et al. (2014) and Wang et al. (2012) a serial concatenation of binary irregular LDPC codes, is also introduced.

The authors (Behairy et al., 2000) showed how the different components LDPC codes with different parameters affected the overall performance in a Gaussian channel. Although they had limited their description of PCGC to a code rate equals to 1/3 by combining two LDPC codes of code rate equals to 1/2, they predicted that the conclusions are easily extended to the case where three or more codes are used as presented in (Behairy et al., 2014). Also, the authors (Behairy et al., 2000) showed that the interleaver was not necessary when the LDPC code was concatenated with another. To study the interleaving effect between component LDPC codes, a PCGC has been modified to use an interleaver as presented in (Belgheit et al., 2012) for irregular codes. However, the irregular LDPC codes have an error floor and a higher coding complexity than regular codes, although they are more efficient than regular code.

In this work, we studied the concatenation of two identical regular non-binary LDPC codes arranged in serial, using an interleaver between two LDPC codes that composed it, and we performed the decoding operation iteratively between the component codes. In our simulation, we used a high order constellation using Gray mapping under Gaussian and Rayleigh channels.

The rest of the paper is organized as follows. Section 2 introduces the non-binary LDPC encoding. Gray-QAM mapping and demapping is studied in section 3. FFT-SPA algorithm that used in our simulation is introduced in section 4. In sections 5 and 6, the parallel turbo LDPC encoding and decoding are investigated, respectively. Finally, the simulation results and concluding remarks are given in section 7 and 8, respectively.

MATERIALS AND METHODS

Non-Binary LDPC Encoding

Non-binary LDPC codes are defined by their parity check matrix ($M \times N$)-H, where the non-zero elements in this matrix belong to the Galois Field GF(2^p) (p>1). The symbols of information block, of size (N-M), belong to GF(2^p). An encoder output can be expressed as a sequence of symbols in GF(2^p).

The encoding is doing by several methods. In this work, one used the known encoding method by LU decomposition of H, this encoding type is systematic. It means that the codeword $C = [C_1\ C_2\ ...\ C_N]$, is as follows:

$C = [C_R\ C_I]$

Where $C = [C_1\ C_2 \ldots C_{N\text{–}M}$ is the information block of size (N-M) and $C_R = [C_1\ C_2 \ldots C_{N\text{–}M}$ is the redundancy block of size M. Therefore, the code rate is given by R=(N-M)/N.

At the encoder output, each non-binary codeword $C = [C_1\ C_2 \ldots C_N]$ is converted to binary block $U = [u_1\ C_2 \ldots u_{p \times N}]$. Then, binary codeword generated, is mapped by 2^{2m}-QAM-Gary mapping, where m is an integer.

Gray-Qam Mapping and Demapping

2^{2m}-QAM transmit, at each time 2^{2m} binary symbols. Each set of $2m$ binary symbols is associated to a symbol $c = a + jb$, where a and $b \in \{\pm1, \pm3, \pm5, \ldots, 2m \pm 1\}$. After passing through the transmission channel, the observation relating to the symbol c is represented by the symbols $c' = a' + jb'$. The transmitted symbols are better follow a Gray mapping, it allows to affirm that there is usually only one erroneous symbol.

The simplest diagram of a digital transmission system as part of the association of an LDPC code and a 2^{2m}-QAM, is given in the Figure 1.

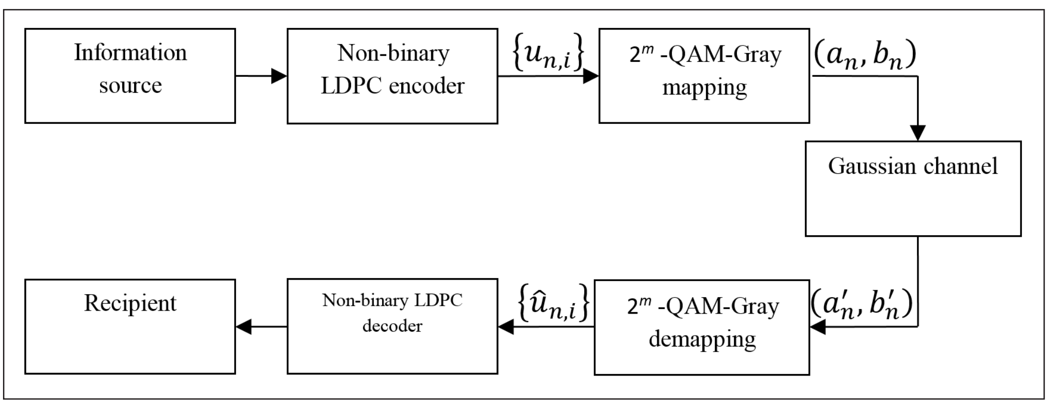

Figure 1. Diagram of a digital transmission system

At the reception, 2^{2m}-QAM-Gray demapping treat each symbols c' representative of the symbols c to extract $2m$ samples $\{\check{u}_{n,i}\}$, $i \in \{1, \ldots, 2m\}$ each representative of a binary symbol $u_{n,i}$. The sample $\check{u}_{n,i}$, the soft output demapping, is obtained using two relationships, LLR($u_{n,i}$) (Log-Likelihood Ratio) (Mostari et al., 2017) or APP($u_{n,i}$) (A Posteriori Probability). In this work, one used APP computation:

For a Gaussian transmission channel, with the noise variance σ^2, the m relations in phase eventually lead to the following expressions (Barnault et al., 2003):

$$APP(u_{n,i}=0)=\frac{\sum_{j=1}^{2^{m-1}} exp\left\{-\frac{1}{2\sigma^2}\left(a'_n-a^0_{i,j}\right)^2\right\}}{\sum_{j=1}^{2^{m-1}} exp\left\{-\frac{1}{2\sigma^2}\left(a'_n-a^0_{i,j}\right)^2\right\}+\sum_{j=1}^{2^{m-1}} exp\left\{-\frac{1}{2\sigma^2}\left(a'_n-a^1_{i,j}\right)^2\right\}}, \qquad i\in\{1,\dots,m\}$$

$$APP(u_{n,i}=1)=1-APP(u_{n,i}=0), i\in\{1,\dots,m\}$$

Where $a^k_{i,j}$ are possible values of the symbol a_n when the symbol $u_{n,i}$ to be transmitted has the value k (k = 0 or 1).

Similarly, for a Gaussian channel, the p relations in the quadrature path eventually lead to the following expressions (Moon, 2005):

$$APP(u_{n,i}=0)=\frac{\sum_{j=1}^{2^{m-1}} exp\left\{-\frac{1}{2\sigma^2}\left(b'_n-b^0_{i,j}\right)^2\right\}}{\sum_{j=1}^{2^{m-1}} exp\left\{-\frac{1}{2\sigma^2}\left(b'_n-b^0_{i,j}\right)^2\right\}+\sum_{j=1}^{2^{m-1}} exp\left\{-\frac{1}{2\sigma^2}\left(b'_n-b^1_{i,j}\right)^2\right\}}, i\in\{m+1,\dots,2m\}$$

$$APP(u_{n,i}=1)=1-APP(u_{n,i}=0), i\in\{m+1,\dots,2m\}$$

Where $b^k_{i,j}$ are possible values of the symbol b_n when the symbol $u_{n,i}$ to be transmitted has the value k (k = 0 or 1).

Since non-binary LDPC decoding uses the soft output demapping of non-binary symbols, each p sample $\check{u}_{n,i}$ is used to obtain the soft output demapping of a non-binary symbol a, $a \in GF(2^p)$. Therefore, we obtain a bloc F of N components from a block of length pN. Each components in F is a vector of length 2^p:

$$F_i=\begin{pmatrix} APP(0) \\ APP(1) \\ \vdots \\ APP(2^p-1) \end{pmatrix}, i\in\{1,\dots,N\}$$

with $APP(a)=\prod_{i=1}^{p} APP(u_i)$

Non-Binary LDPC Decoding: FFT-SPA

FFT-SPA (Wang et al., 2013) initializes each variable node v_n in the Tanner graph by the 2^p possible APPs of non-binary symbols. Messages $a_{m,n}$, $m \in \{1, \dots, M\}$, and $n \in \{1, \dots, N\}$ in variable nodes are given by:

$$a_{m,n} = F_n$$

After, each check node c_m in the Tanner graph receives messages $a_{m,n}$ arriving from variable nodes that are connected by their branches, then calculates and sends the resulting message that is related to all messages except the input message that the resulting message will be sent:

$$\beta_{m,n}(h_{ij} \otimes a) = FFT^{-1}\left(\prod_{n' \in N_m/n} FFT\left(\alpha_{m,n'}(h_{ij} \otimes a)\right)\right)$$

Where N_m is the set of check nodes with $h_{ij} \neq 0$ (h_{ij}, $i \in \{1, \ldots, M\}$ and $j \in \{1, \ldots, N\}$ represent elements of H).

Then, each variable node v_n receives messages $\beta_{m,n}$ arriving from check nodes that are connected by their branches, then calculates and sends the resulting message that is related to all messages except the input message that the resulting message will be sent:

$$\alpha_{m,n}(a) = \delta_{m,n}\, \gamma_n(a) \prod_{m' \in M_n/m} \beta_{m',n}(a)$$

Where M_n is the set of variable nodes with $h_{ij} \neq 0$.

Then, a posteriori information associated to each variable node is calculated before taking a decision.

$$\tilde{\gamma}_n(a) = \delta_n \gamma_n(a) \prod_{m' \in M_n} \beta_{m',n}(a)$$

The decision is given by

$$Z_n = \underset{a}{\operatorname{argmax}}\left(\tilde{\gamma}_n(a)\right)$$

Finally, after a number of iterations or in case the syndrome is zero, the algorithm stops.

Serial Turbo LDPC Coding

The serial turbo encoder is built using a serial concatenation of two systematic component encoders separated by an interleaver noted π. In this work, each component encoder is non-binary regular LDPC encoder.

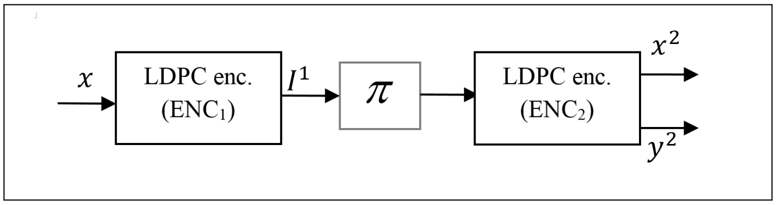

Figure 2. Serial turbo LDPC encoder

Figure 2 represents the block diagram of a rate R_{SC} serial turbo LDPC encoder (Berrou, 2007), where R_{SC} is given by:

$R_{sc} = R_1 . R_2$

where R_1 is the code rate of the fist component encoder ENC_1 and R_2 is the code rate of the second component encoder ENC_2.

The first component encoder ENC_1 encodes the information block x of size N-M, = [x_1 x_2 ... x_{N-M}], using a parity check matrix H_1 of size , and generates the coded information block of size N:

$$I^1 = [y^1\ x^1] = [y_1^1\ y_2^1 \dots y_M^1\ x_1^1\ x_2^1 \dots x_{N-M}^1]$$

where x_1 is a systematic bloc $x^1 = x$, and y^1 is a parity block.

The second encoder ENC_2 uses the interleaved bloc $I_{interleaved}^1$, of size N, using a parity check matrix H_2 of size (L – N) ×, and generates the coded information block of size L:

$$I^2 = [y^2\ x^2] = [y_1^2\ y_2^2 \dots y_{L-N}^2\ x_1^2\ x_2^2 \dots x_N^2]$$

where x^2 is an interleaved systematic bloc $x^2 = x_{interleaved}^1 = x_{interleaved}$, and y^2 is a parity block. Thus, the turbo LDPC encoder encodes the information block x of size N-M, [x_1 x_2 ... x_{N-M}], and generates the coded information block of size L:

$$[y^2\ x^2] = [\ y_1^2\ y_2^2 \dots y_{L-N}^2\ x_1^2\ x_2^2 \dots x_N^2]$$

Serial Turbo LDPC Decoding

A serial turbo-decoder (Berrou, 2007) presented at Figure 3, consists of two decoders DEC_1 and DEC_2 associated respectively to the codes ENC_1 and ENC_2 disposed in serial, of an interleaver and a deinterleaver noted π^{-1}.

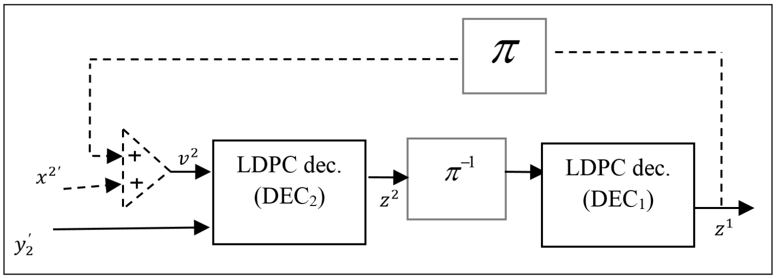

Figure 3. Serial turbo LDPC decoder

Each component LDPC decoder is decoded by using soft-input soft-output decoding algorithm, as described in section 4, using the "FFT-Sum-Product Algorithm".

In the proposed non-binary turbo LDPC decoder, each non-binary turbo LDPC code contains two non-binary LDPC decoders decoded iteratively. Therefore, each turbo iteration, $iter_{turbo}$ of the non-binary turbo LDPC code contains multiple LDPC iterations $iter_{ldpc}$.

The turbo LDPC decoder receives the soft observations $x^{2'}$ and $y^{2'}$ and estimates the message transmitted. where $x^{2'}$ denotes the received block corresponding to the interleaved systematic information block, while $y^{2'}$ denotes the received blocks corresponding to the parity block of the second decoders.

In the first iteration $iter_{turbo}$, the first decoder DEC_2 generates the soft information block $z^2 = [z_1^2\ z_2^2 \dots z_N^2]$, after a fixed number of iterations $iter_{ldpc}$, using the following received block:

$$[y^{2'}\ x^{2'}] = [y_1^{2'}\ y_2^{2'} \dots y_{L-N}^{2'}\ x_1^{2'}\ x_2^{2'} \dots x_N^{2'}]$$

The second decoder DEC_1 generates the soft information block $Z^1 = [z_1^1\ z_2^1 \dots z_N^1]$, after a fixed number of iterations $iter_{ldpc}$, using, $z^2_{deinterleaved}$, the deinterleaved block of the information generated by the first decoder DEC_2,

For next iterations $iter_{turbo}$, the first decoder DEC_2 uses, $z^1_{interleaved}$, the interleaved block of the block generated by the second decoder DEC_1 to do the decoding. The turbo LDPC decoder input becomes:

$$[y^{2'}\ v^2] = [y_1^{2'}\ y_2^{2'} \dots y_{L-N}^{2'}\ v_1\ v_2 \dots v_N]$$

Where $v^2 = x^{2'} + z^1_{interleaved}$

Decoding stops at the end of a fixed number of iteration, and the final decision comes from DEC_1. One iteration corresponds to one pass from DEC_2 to DEC_1.

RESULTS AND DISCUSSION

Simulations

In this section, we discuss the performance of non-binary serial turbo LDPC codes and non-binary LDPC codes constructed on GF(4), with the same block length and code rate, combined with high order constellations (16-QAM, 64-QAM and 256-QAM) using Gray mapping, over Gaussian and Rayleigh channels. Simulation results are given in terms of Bit Error Rate (BER) versus E_b / N_0, where E_b is the energy per information and N_0 is the spectral density noise, using Matlab.

The non-binary LDPC code is made by a parity check matrix with the parameters ($w_c = 4$, $M = 1024$, $N = 1536$) decoded by FFT-SPA, and the serial turbo LDPC code is composed of two regular non-binary LDPC codes decoded by FFT-SPA: the inner-code is a rate 1/2 non-binary LDPC code with the parameters ($w_c = 2$, $M = 512$, $N = 1536$) , and the outer-LDPC code is a rate 2/3 non-binary LDPC code with the parameters $w_c = 4$, $M = 1024$, $N = 1536$). The number of iterations in turbo LDPC code is set to 2 using 2 iterations in each component non-binary LDPC code, ($iter_{turbo} = 2$, $iter_{ldpc} = 2$), and the maximum number of a single LDPC code is set to 2, $iter_{ldpc} = 2$.

Simulation results in Figures 4,5 and 6 show that the proposed code outperforms the single LDPC codes. In order to investigate the performance of serial turbo LDPC code in a Rayleigh channel, performance comparison is conducted on a Rayleigh channel, with 16-QAM, 64-QAM and 256-QAM, respectively in Figures 7, 8 and 9.

The simulation results presented in all figures show that turbo LDPC code outperforms LDPC code. Turbo LDPC code needs less number of iterations to achieve the same performance of single LDPC code. As seen in Figures 7, 8 and 9, the coding gain between a single LDPC code and a serial turbo LDPC code increases in a Rayleigh channel. This increase is due to the interleaver in the serial turbo LDPC code. It means that the interleaver has a good effect in fading channels.

As mentioned before, achieving high spectral efficiency using high order constellations with high quality is the key requirement of future wireless systems and mobile

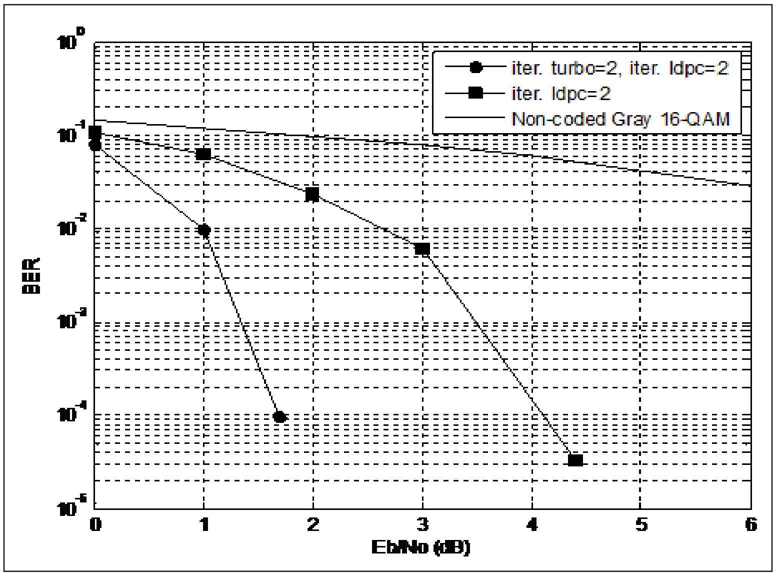

Figure 4. Performance comparison of a rate 1/3 non-binary turbo LDPC code with a rate 1/3 non-binary LDPC code associated with 16-QAM constellation under Gaussian channel

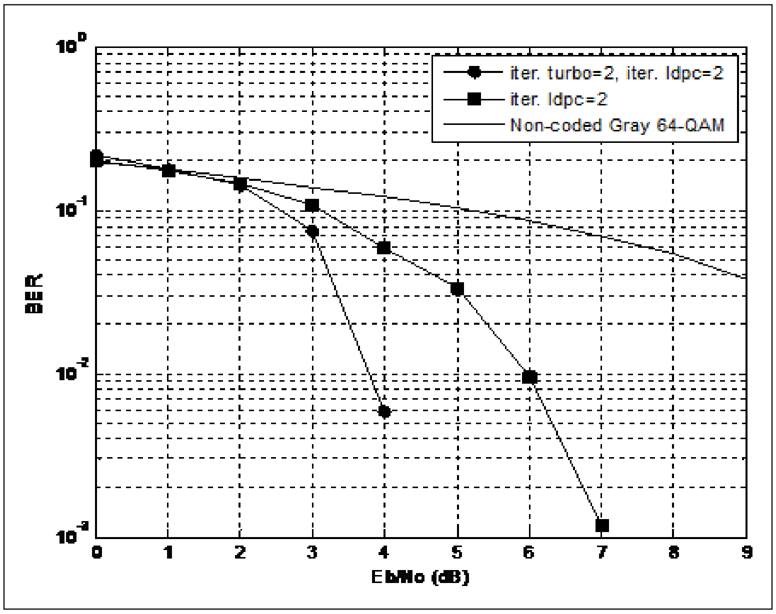

Figure 5. Performance comparison of a rate 1/3 non-binary turbo LDPC code with a rate 1/3 non-binary LDPC code associated with 64-QAM constellation under Gaussian channel

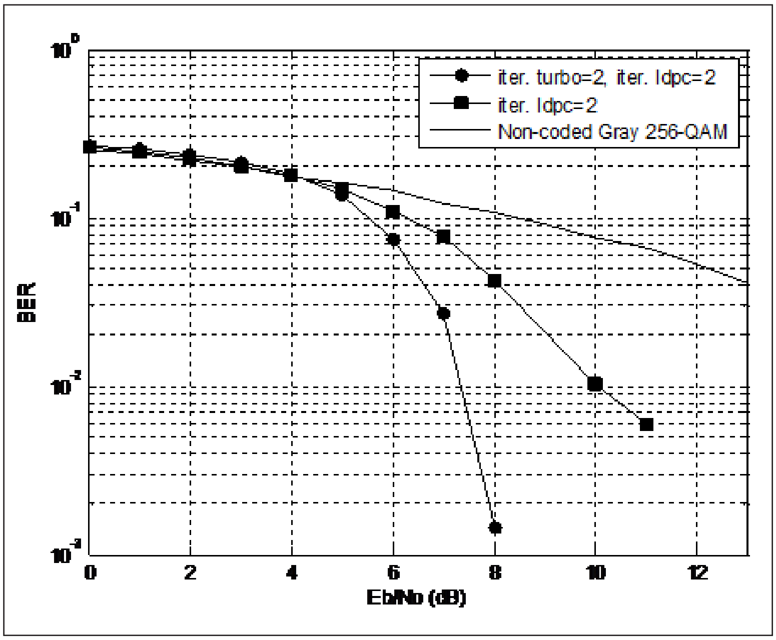

Figure 6. Performance comparison of a rate 1/3 non-binary turbo LDPC code with a rate 1/3 non-binary LDPC code associated with 256-QAM constellation under Gaussian channel

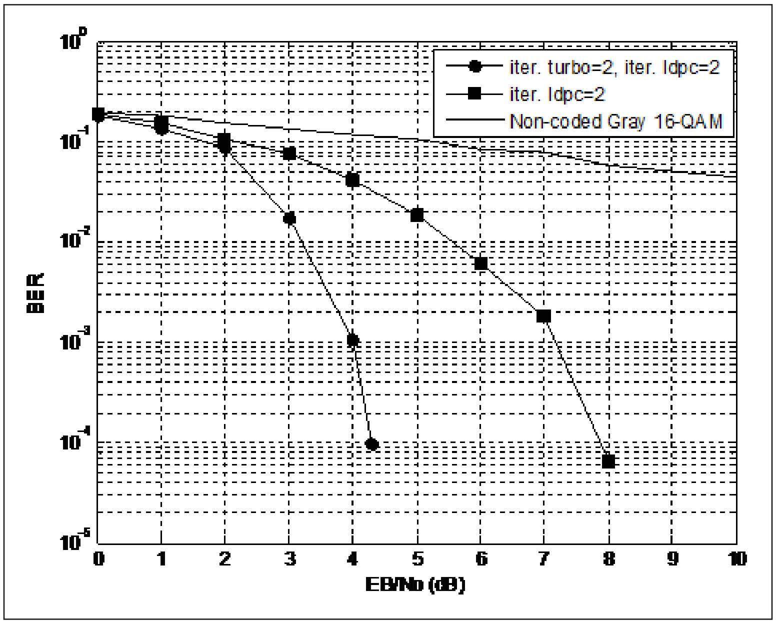

Figure 7. Performance comparison of a rate 1/3 non-binary turbo LDPC code with a rate 1/3 non-binary LDPC code associated with 16-QAM constellation under Rayleigh channel

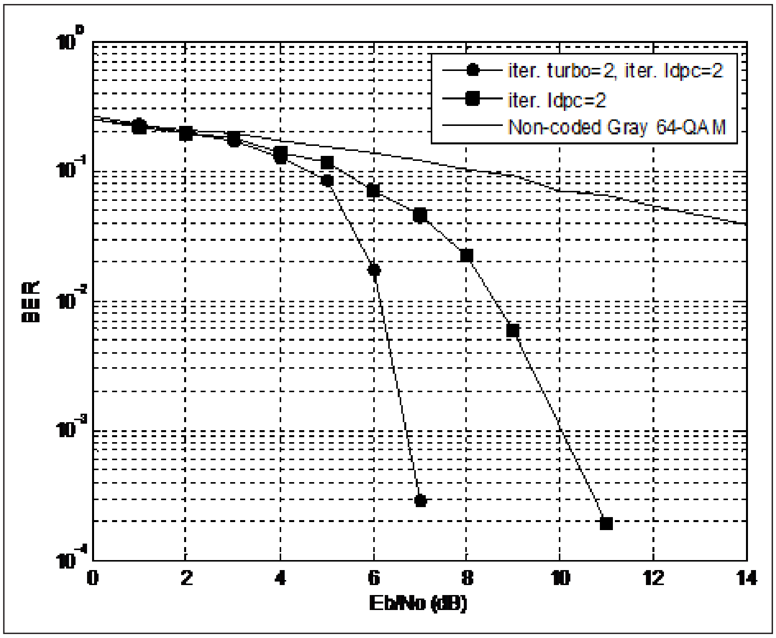

Figure 8. Performance comparison of a rate 1/3 non-binary turbo LDPC code with a rate 1/3 non-binary LDPC code associated with 64-QAM constellation under Rayleigh channel

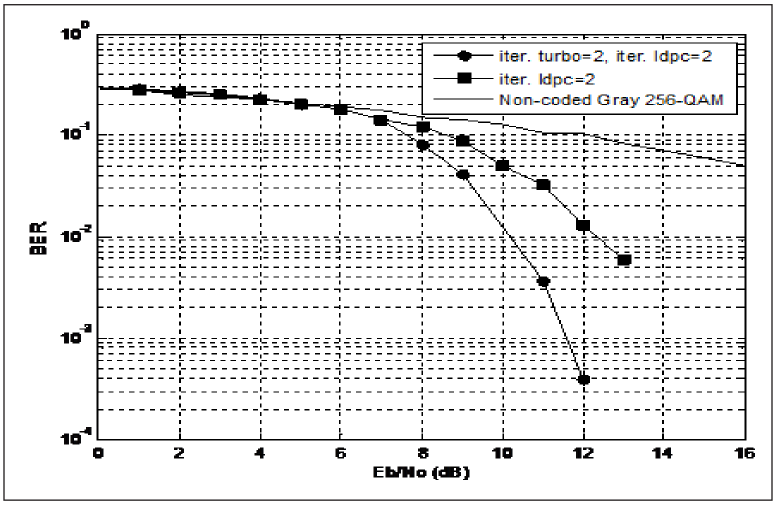

Figure 9. Performance comparison of a rate 1/3 non-binary turbo LDPC code with a rate 1/3 non-binary LDPC code associated with 256-QAM constellation under Rayleigh channel

communications. Thus, the transmission with high spectral efficiency needs a high performance error correcting code such as LDPC codes. Simulation results show that the performance of the proposed code, with higher order constellations (16-QAM, 64-QAM and 256-QAM) using Gray mapping, under Gaussian and Rayleigh channels has higher performance than a single non-binary LDPC code with the same block length. Also, we show that the interleaver in the proposed code has a positive effect on the performance. Therefore, the proposed code is a good error correcting code for high spectral efficiency system.

CONCLUSION

In this work, we proposed a non-binary turbo LDPC code. It is an error correcting code scheme based on the serial concatenation of non-binary LDPC. Simulation results show that the performance of non-binary turbo LDPC code, with high order constellation using Gray mapping under Gaussian and Rayleigh channels, is higher than the performance of a rate 1/3 non-binary LDPC code.

REFERENCES

AlMuaini, S. A., Al-Dweik, A., & Al-Qutayri, M. (2013). BER performance of turbo product LDPC codes with non-sequential decoding. In *2013 6th Joint IFIP on Wireless and Mobile Networking Conference (WMNC)* (pp. 1-6). Dubai, United Arab Emirates.

Barnault, L., & Declercq, D. (2003). Fast decoding algorithm for LDPC over GF (2/sup q/). In *Proceedings of Information Theory Workshop* (pp. 70-73). Paris, France.

Beermann, M., Monró, E., Schmalen, L., & Vary, P. (2013, October). High speed decoding of non-binary irregular LDPC codes using GPUs. In *2013 IEEE Workshop on Signal Processing Systems (SiPS)* (pp. 36-41). Taipei City, Taiwan.

Behairy, H. M., & Benaissa, M. (2013). Multiple parallel concatenated gallager codes: Code design and decoding techniques. *IETE Journal of Research, 59*(6), 659-664.

Behairy, H., & Chang, S. C. (2000). Parallel Concatenated Gallager Codes. *Electronics Letters, 36*(24), 2025-2026.

Belgheit, B., Boukelif, A., Lakhder, A. M., & Kamline, M. (2012). Parallel concatenated Gallager codes matrix and the effect of interleaver. *International Journal of Electronics, 99*(9), 1281-1289.

Berrou, C. (2007). *Codes et Turbo-Codes*. Paris: Springer.

Berrou, C., Glavieux, A., & Thitimajshima, P. (1993). Near Shannon limit error-correcting coding and decoding: Turbo-codes. 1. In *Proceedings of ICC '93 - IEEE International Conference on Communications* (Vol. 2, pp. 1064-1070). Geneva, Switzerland.

Chandrasetty, V. A., & Aziz, S. M. (2011). FPGA implementation of a LDPC decoder using a reduced complexity message passing algorithm. *Journal of Networks, 6*(1), 36-45.

Davey, M., & MacKay, D. (2002). Low-Density Parity Check Codes over GF(*q*). *IEEE Communications Letters, 2*(6), 165–167.

Declercq, D., & Fossorier, M. (2007). Decoding Algorithms for Non binary LDPC Codes over GF(*q*). *IEEE Transactions on Communications, 55*(4), 633-643.

Elias, P. (1955). Coding for noisy channels. *IRE Convention Record, 3*(4), 37-46.

Gallager, R. G. (1962). Low-Density Parity-Check Codes. *IEEE Transaction Information Theory, 8*(1), 21–28.

Gallager, R. G. (1963). *Low-density parity-check codes*. Cambridge: MIT Press.

Hung, J. H., Shyu, J. S., & Chen, S. G. (2011, May). A New High-Performance and Low-Complexity Turbo-LDPC Code. In *2011 International Conference on Multimedia and Signal Processing (CMSP)* (Vol. 1, pp. 68-72). Guilin, Guangxi, China.

Kumar, R. P., & Kshetrimayum, R. S. (2013, February). An Efficient Methodology for Parallel Concatenation of LDPC codes with reduced complexity and decoding delay. In *2013 National Conference on Communications (NCC)* (pp. 1-5). New Delhi, India.

Luby, M. G., Mitzenmacher, M., Shokrollahi, M. A., Spielman, D. A., & Stemann, V. (1997, May). Practical loss-resilient codes. In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing* (pp. 150-159). Texas, USA.

MacKay, D. J., & Neal, R. M. (1995, December). Good codes based on very sparse matrices. In *IMA International Conference on Cryptography and Coding* (pp. 100-111). Cirencester, UK.

Moon, T. K. (2005). *Error Correction Coding Mathematical Methods and Algorithms*. New Jersey: John Wiley & Sons, Inc.

Moision, B. (2013). Decoding complexity and performance of short-block LDPC codes over GF(q). *Interplanetary Network Progress Report, 194*, 1-16.

Mostari, L., Taleb-Ahmed, A., & Bounoua, A. (2018). High performance LDPC codes for high spectral efficiency transmission. *Internetworking Indonesia Journal, 10*(1), 09-15.

Mostari, L., & Taleb-Ahmed, A. (in press). High performance short-block binary regular LDPC codes. *Alexandria Engineering Journal.*

Mostari, L., & Taleb-Ahmed, A. (2017). Simplified log likelihood ratio for binary LDPC codes. *Journal of Engineering Science and Technology Review, 07*(1), 37-40.

Pyndiah, R. (1998). Near Optimum Decoding of Product Codes: Block Turbo-Codes. *IEEE Transactions on Communications, 46*(8), 1003-1010.

Savin, V. (2008, July). Min-Max decoding for non binary LDPC codes. In *IEEE International Symposium on Information Theory (ISIT 2008)* (pp. 960-964). Toronto, ON, Canada.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal, 27*(3), 379-423.

Sipser, M., & Spieleman, D. A. (1996). Expender Codes. *IRE Transaction on Information Theory, 42*(6), 1710-1722.

Spagnol, C., Popovici, E., & Marnane, W. (2009) .Hardware implementation of *GF* (*2m*) LDPC decoders. *IEEE Transactions Circuits Systems I: Regular Papers, 56*(12), 2609-2620.

Tahir, B., Schwarz, S., & Rupp, M. (2017, May). BER comparison between Convolutional, Turbo, LDPC, and Polar codes. In *2017 24th International Conference on Telecommunications (ICT)* (pp. 1-7). Limassol, Cyprus.

Voicila, A., Declercq, D., Verdier, F., Fossorier, M., & Urard, P. (2010). Low-complexity decoding for non-binary LDPC codes in high order fields. *IEEE Transactions on Communications, 58*(5), 1365-1375.

Wang, C., Chen, X., Li, Z., & Yang, S. (2013). A Simplified min-sum decoding algorithm for non-binary LDPC codes. *IEEE Transactions on Communications, 61*(1), 24–32.

Wymeersch, H., Steendam, H., & Moeneclaey, M. (2004, June). Log-domain decoding of LDPC codes over GF (q). In *2004 IEEE International Conference on Communications* (Vol. 2, pp. 772-776). Paris, France.

Wang, Z., & Zhang, M. (2012, April). A serial concatenated scheme for LDPC code to achieve better error correction performance. In *2012 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet)* (pp. 1587-1589). Yichang, China.