Density Evolution of Low Density Parity Check Codes Over Different Channels

P.Ravikiran[#1], Mehul C. Patel[*2]
[#]*M.Tech&*Assistant Professor*
Department of Electronics, Sardar Vallabhai National Institute of Technology, Surat, India

Abstract

LDPC codes are originally invented by Robert G. Gallager. Later these codes are rediscovered by multiple groups has become the best known forward error correcting codes.We will discuss about density evolution of regular LDPC codes over Binary Erasure Channel(BEC) and Binary Symmetry channel(BSC) under message passing decoding algorithm in this paper. Density Evolution is a technic used to evaluate how good the LDPC codes are working on the given channel and to evaluate how far away the performance is from the Shannon limit.

Keywords– *LDPC codes, BEC, BSC and Density Evolution.*

I. INTRODUCTION

LDPC codes are set of linear block codes that were initiallydiscovered by Robert G. Gallager in the year 1962[1] and after neglecting for almost 35 years these codes were rediscovered by D. J.C. Mackay and Neil in 1996[2]. Because these codes are working at near Shannon limit performance and decoding complexity of these codes is also less, so these codes are foundinmany standers such as WiMAX and DVB-S2extensively [10]. Transmitting data reliably over a noisy channel is a general problem that attracted significant interest. In 1948, Shannon formalized the notions of information, noisy channel and other information theoretic concepts in his seminar paper, 'A Mathematical Theory of Communication'[3]. A communication channel can be represented as a tuple with input and output alphabets with and probabilities of transition from symbol in the input alphabet to a symbol in the output alphabet. In other words, the distribution specifies the receiving probability of an output symbol given that an input symbol was transmitted. A discrete memoryless channel can be described as a channel with discrete set of input and output alphabets and transition probabilities that only depend on the current input symbols being sent.

LDPC codes in their regular manifestation, the working of the codes over the Binary-Input AWGN channel are only little menial to that of serially or parallel concatenated convolution codes. Let us consider an example with one-half (1/2) rate LDPC code with 10000 as the block length, to achieve a bit error probability of 10^{-5} requires E_b/N_0 of roughly 1.4dB, whereas in Turbo codes on par with the same complexity achieve theequal performance at, approximately E_b/N_0 of 0.8dB[3]. Shannon capacity theorem states that in order to obtain a trustworthy transmission one-half (1/2) rate bit per channel used over continuous-input AWGN channel, a E_b/N_0 of atleast 0dB is needed and it can increases to$0.187\mathrm{dB}$, if we limited to binary input.

We knew that any linear equation can be represented with a set of solutions x of a parity check equation $Hx^T = 0^T$. Moreover, if a binary code is used then parity check matrix H takes elements in Galois Field(GF) which was GF(2) and the calculationsare also done over the field. A (d_v, d_c) regular-LDPC code, as originally described by R. G. Gallager[1], is a linear binary code deduced by the specification that eachcodeword bit involves in d_v parity-check equationsi.e in other words there are d_v number of ones in each column and there are d_c codeword bits in every such constraint equation participatedi.e in other words there are d_c number of ones in every row, where the parameters d_v and d_c can be chosen according to the design rate which will be explained in later section. The word 'Low Density' describes that the number of nonzero elements in the H matrix is small compared to number of zeros. Number of one's is small for LDPC parity check matrix Hin particular it is in linear block length n as compared to random linear codes for which the expected number of ones increases like n^2 CITATION Ric01 \l 16393 | [3]}. In this paper we are not confining to one particular LDPC codes rather analyzing the performance of *ensemble* of codes. There are different ways to construct ensemble of LDPC codes, one such way is to consider H matrixof length $n$and choose the parameters (d_v, d_c) so that they satisfy the above said column and row conditions and arrange this ensemble with a uniform probability distribution. A Bipartite graph is used to define an ensemble of LDPC codes and these graphs are also called Tanner graphs. In

section II tanner graphs are mentioned briefly, so that it is easier to analyze the resulting ensemble. There are many ways to construct LDPC codes that approach Shannon capacity using Gallager's original definition by applying many variations and extensions to the original codes. Here two important modifications are mentioned below:a) constructing irregular codes in contrary to regular codes [4], [5], [7], i.e. check equations may involve in different number of variables and variables may participate in different number of checks and b) permitting nodes to represent group bits instead of single bits [6], [1].

There are many decoding algorithms discussed by Gallagerthat can work efficiently and directly on the nodes and links of the Tanner graph (see section II) representing the LDPC codes. In this, one set of nodes called "variable" nodes corresponding to variables which represents the codewordbits x_i, represents columns of H, and the second set of nodes called "check" codes corresponding to constraints which represents the rows of parity check matrix H. In a Tanner graph a bit node is connected to the check node if and only if the respective bit node involves in the equivalent check equation.In a Tanner graph the information is passed on the edges from bit node to check node and vice versa and the decoding algorithms work iteratively on the tanner graph. Every message can be correlated with the codeword bit corresponding to bit nodeto the link conveying the message. And the received messages can be explicated as the approximation of that bitalong with some reliability information. There is some hard decision associated with that message: one can look into the bit's most likely value deduced by the message. We can understand that,if the hard decision is correct then the message received is correct else the message is incorrect.

In general, "threshold" is a critical channel parameter which depends on specific ensemble of codes, the type of channel and the decoding algorithm used. Therefore using these parameters we can calculate threshold. So, we can expect reliable transmission of the messages for any length code if the actual cross over probability is less than this threshold, if decoded by the given decoder for sufficiently large number of iterations. Conversely, if the actual cross over probability is greater than the threshold reliable transmission of the messagesis not possible over that channel with the codes chosen at random from long ensemble. Hence, we should keep in mind that threshold is identical to the random capacity for a given decoder and ensemble of codes.

A short blueprint of the paper is as per the following:Section II gives brief clarification about class of codes utilized, channels and decoding algorithms taken for consideration in this paper. Section III concentrates on the calculation of threshold value. We can observe that an appropriate choice of the messages the obtained threshold value is almostnearer to the Shannon limit. We can say that the calculation of threshold is assured for many families of channels including the Binary Erasure Channel (BEC), Binary Symmetry Channel (BSC), Binary-Input AWGN and the Binary-Input Laplacian (BIL) channel if the channel family can be ordered by some physical degradation.

Density Evolution and Threshold determination: Here $P_e^{sc}(l)$ is defined 'as expected portion of the erroneous messages forwarded in the l^{th} decoding iteration surmising that Tanner graph has no cycles of length $2l$ or less' is calculated by the deterministic algorithm. There subsists a channel parameterσ^*, threshold has the properties mentioned below: if $\sigma < \sigma^*$ then $\lim_{l\to\infty} P_e^{sc}(l) = 0$, if $\sigma > \sigma^*$ then$\lim_{l\to\infty} P_e^{sc}(l)$ does not tend to zero.

II. BASIC TERMINOLOGY AND ASSUMPTIONS

Here we will discuss some of the basic terminologies and assumption that are used in this paper in this section. Section- II starts with defining an *ensemble*, class of *binary memoryless channels* and later we will discuss about *message passing decoders*. Finally we will assume that the transmitted codeword was an all onecodeword.

A. Ensembles:

Assume a binary input code with a block length n, represented as set of solutions x to the given parity equation $Hx^T = 0^T$. Let us construct a Tanner graph with a bit nodes and b constraint nodes, where $b = ad_v/d_c$. Each bit node specifies to onecodeword bit and every constraint node specifies one of the parity check equations. Each edge in the Tanner graph is connected between a bit node and a constraint node. The design rate of the code is

$$r = \frac{a-b}{a} = 1 - \frac{b}{a} = 1 - \frac{d_v}{d_c}$$

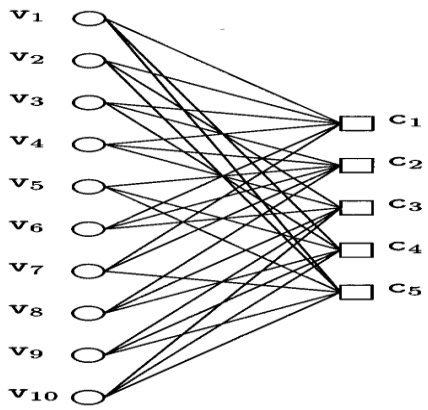

Figure 1 Tanner graph for $(3,6)$ ensemble regular LDPC code with a block length 10.

Since the check equations may not be all independent so the actual design rate may be heigher but we are neglecting this possibility. There are$bd_c = ad_v$ number of links in a Tanner graph, on the left side of the graph$d_v$number of links incident on the bit node and on the right side of the graph$d_c$number of links incident on the constraint node. The Figure-1llustrates (3, 6)-regular LDPC code of length 10.

B. Binary memoryless channels:

In this paper the class of channels we have considered are memoryless channels.Let $\bar{I} = \{\pm 1\}$ be the binary input alphabet which represents +1 as binary 0 and -1 as binary 1 and continuos or discrete output alphabet $\mathcal{O}$.

1. Binary Symmetry Channel (BSC): Consider$\bar{I} = \{\pm 1\}$ and $x_t \in \bar{I} = \{\pm 1\}$ as the input to the channel at time t, $t \in \mathbb{Z}$. Also consider$y_t \in \mathcal{O} = \{\pm 1\}$ as the output of the channel at time t. The parameter ϵ in the BSC is characterised by the relation$y_t = (-1)^{w_t} x_t$, where w_t is a sequence of i.i.d Bernouli random variables. It has the following probabilities $P_r\{w_t = 0\} = 1 - \epsilon$ and $P_r\{w_t = 1\} = \epsilon$.

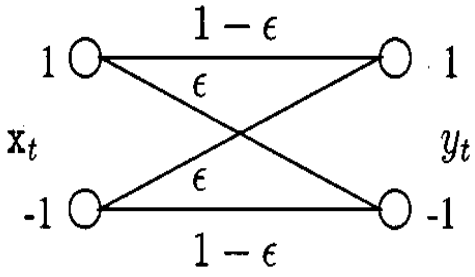

Figure 2 Binary Symmetry Channel with Parameter ϵ

The channel model is shown in figure 2. The Shannon capacity of this channel [8] is

$$C_{BSC}(\dot{o}) = 1 - h(\dot{o}) \quad (1)$$

Where $h(\epsilon) = -(x \log_2 x + (1-x)\log_2(1-x))$ is a entropy function in binary [8, p13].

2. Binary Erasure Channel: Let $\bar{I} = \{0, 1\}$ and $x_t \in \bar{I} = \{0, 1\}$ be the channel input at time t, $t \in \mathbb{Z}$. Let y_t, $y_t \in \mathcal{O} = \{0, ?, 1\}$ be the output at time t. In this model the receiver receiver either correct bit else the bit was not received i.e. erased if the probability isϵ.

The channel model is shown in figure 3. The Shannon capacity of this channel is

$$C_{BEC}(\varepsilon) = 1 - \varepsilon$$

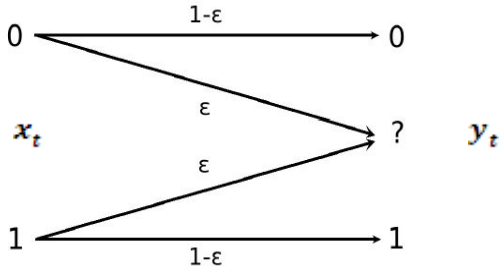

Figure 3 Binary Erasure Channel with parameter ε

***C. Message passing Decoders*:**

Without loosing the abstraction, assume that output alphabet $\mathcal{O}$of the channel is equal to the input alphabet of the decoder. There are many reasonable decoding algorithms for a given code and channel model rooted on message passing which are the set of algorithms we considered. These message passing algorithms will behaves as follows. At the initial time i.e at zero time, every bit node $v_i, i \in [n]$, has an corresponding receiver message η_i, a random variable abstracting values in output alphabet $\mathcal{O}$. Messages are switched between the nodes in the Tanner graph through the links in discrete steps. Initially ech bit node v_i transmits back to every nieghbouring constraint node c_j a message abstractingthe values in message alphabet M. A bit node i sends η_i as its first message, typically at time zero (this needs $\mathcal{O} \sqsubset$). Every check node c_j analyzes the messages it received and they send back to their neighbouring bit nodes v_i a message abstracting values in M. Now every bit node v_i analyses

the messages it received from constraint nodes inconjuction with its corresponding received value r_i to constrcuct new messages which it transmits back to the neighboring constraint nodes. For each iteration l, $l \in \mathbb{N}$, a message passing continues with constraint nodes analyzing and sending, followed by the bit nodes nodes analyzing and sending messages.

A significant requirement on the processing is that a message transmitted from a node u through a neighboring edge e may not depend on the message it received aforetime along edge e. In determining the outgoing message through edge e, there is a significant reason for omitting the incoming message through edge e. Excluding incoming messagearrures that only extrinsic information [10] is passed through, as in Turbo coding terminology. This is one of the important properties of message passing decoders. Analyzing a decoder is possible by using this restriction.

Let $\Psi_v^{(l)}: \mathcal{O} \times {}^{d_v-1} \rightarrow$, $l \geq 1$ describes the bit node message map and $\Psi_c^{(l)}: \mathcal{O} \times {}^{d_c-1} \rightarrow$, $l \geq 0$ describes the constraint node message map as a function of $l \in \mathbb{N}$. The above mentioned functions represents processing done at the bit nodes and check nodes. Due to the above mentioned limitation on the dependency of messages, the outgoing message depends on $d_v - 1$ incoming messages only at the bit node and $d_c - 1$ incoming messages at constraint node.These message maps depends on the numberof iterations of the decoding process. Assume each node of the identical degree supplicates the identical message map for every edge attached to such a node are treated the same. For absoluteness, $\Psi_v^{(0)}: \mathcal{O} \rightarrow$, denotes the first message map i.e. node v_i sends the message $\Psi_v^{(0)}(r_i)$ to its neighbors initially.

***D.Assumptions*:**

To find density evolution, it is assumed that all one codeword was transmitted i.e $+1$ which represents binary zero in BSC and all zero codeword i.e. binary zero in BEC.

III. DENSITY EVOLUTION AND THRESHOLD DERTERMINATION

In this section density evolution and threshold determination is done on BSC and BEC channels assuming that we are using regular LDPC codes. We are also assuming that the tanner graph used was cycle free. We will analyse densities of LDPC codes as a function of number of iteration. In the case of BSC if the sign of the message received along the edge matches with the transmitted message then it is correct else the message is incorrect and in the case of BEC the message sent along the edge is correct if it is received as binary zero else the message is erroneous. In section II-D, we assumed that all one i.e binary zero codeword was transmitted over BSC and all zero codeword were transmitted over BEC.

In section III-A, weare focussing on discrete message alphabets in message passing decoders. As stated previously, in the l[th] iteration the assumedfraction of erroneous messages transmitted can be characterized by a system of recursive function which depends on (d_v, d_c) and the channel parameter. The existence of a Threshold is determined by analyzing this system of recursive function.

***A. Discrete message alphabets*:**

We begin with discrete alphabets in message passing decoders. Consider $P_k^{(0)}$, $k \in \{-q, \ldots, -1, 0, 1, \ldots, +q\}$ as the probability that a message transmitted at zeroth time is equal to k. Let $P_k^{(l)}$, denote that the probability of messages transmitted from bit node to constraint node at iteration l. Similarly $q_k^{(l)}$ denotes the probability of messages transmitted from constraint node to bit node in the l[th] iteration. Gallager shown that the recursions function can be expressed using $P_k^{(l)}$ as a function of $P_k^{(0)}$ and $P_k^{(l-1)}$, the ensemble (d_v, d_c) and the channel parameter.

***B. Density evolution on BSC*:**

Consider a (d_v, d_c) –regular code ensemble on the BSC with input alphabet $\{-1, +1\}$, the channel parameter or the cross over probability ϵ and the output alphabet {-1, +1}. This clearly shows that $p_1^{(0)} = 1 - \epsilon$ and $p_{-1}^{(0)} = \epsilon$. $= \{-1, +1\}$is the message symbol. Message maps do not depend on number of iterations as they are time invariant. They are given by $\Psi_v(m_0) = m_0$, $\Psi_v(m_0, m_1, \ldots, m_{d_v-1}) = -m_0$ if $m_1 = m_2 = \cdots = m_{d_v-1} = -m_0$ and $\Psi_v(m_0, m_1, \ldots, m_{d_v-1}) = -m_0$ otherwise

$$\Psi_c\left(m_1, m_2, \ldots, m_{d_c-1}\right) = \prod_{i=1}^{d_c-1} m_i$$

The above equation can be expressed in words as modulo two sum of the neighbouring bits at the constraint nodes. The bit nodes transmit their incoming values unless all the received messages are same.Then,

$$\left(q_{-1}^{(l)},q_{1}^{(l)}\right)=\Psi_{c}\left(\left(p_{-1}^{l-1},p_{1}^{l-1}\right),.....\left(p_{-1}^{l-1},p_{1}^{l-1}\right)\right)$$

$$=\frac{1}{2}\left(1-\left(1-2p_{-1}^{l-1}\right)^{d_{c}-1},1+\left(1-2p_{-1}^{l-1}\right)^{d_{c}-1}\right)$$

and

$$\left(p_{-1}^{(l)},p_{1}^{(l)}\right)$$

$$=\Psi_{v}\left(\left(q_{-1}^{(l)},q_{1}^{(l)}\right),.....\left(q_{-1}^{(l)},q_{1}^{(l)}\right)\right)$$

$$=\left(p_{1}^{(0)}\left(q_{-1}^{(l)}\right)^{d_{v}-1}+p_{-1}^{(0)}\left(1-\left(q_{1}^{(l)}\right)^{d_{v}-1}\right),\right.$$

$$\left.p_{-1}^{(0)}\left(q_{1}^{(l)}\right)^{d_{v}-1}+p_{1}^{(0)}\left(1-\left(q_{-1}^{(l)}\right)^{d_{v}-1}\right)\right)$$

By substituting p in place of q we obtain,

$$p_{-1}^{(l)}=p_{-1}^{(0)}-p_{-1}^{(0)}\left[\frac{1+\left(1-2p_{-1}^{(l-1)}\right)^{d_{c}-1}}{2}\right]^{d_{v}-1}$$

$$+\left(1-p_{-1}^{(0)}\right)\left[\frac{1-\left(1-2p_{-1}^{(l-1)}\right)^{d_{c}-1}}{2}\right]^{d_{v}-1}$$

From equation 3 we can get threshold values for different pairs of ensembles (d_v,d_c), these threshold values are tabulated in Table-1. Threshold value ϵ^* is the maximum of all values of $p_{-1}^{(0)}\in[0,1]$ such that $\lim_{l\to\infty} p_{-1}^{(l)}=0$.From the above statement for the values of $p_{-1}^{(0)}$ less than ϵ^*, $\lim_{l\to\infty} p_{-1}^{(l)}=0$, otherwise $\lim_{l\to\infty} p_{-1}^{(l)}$ does not tends to zero.

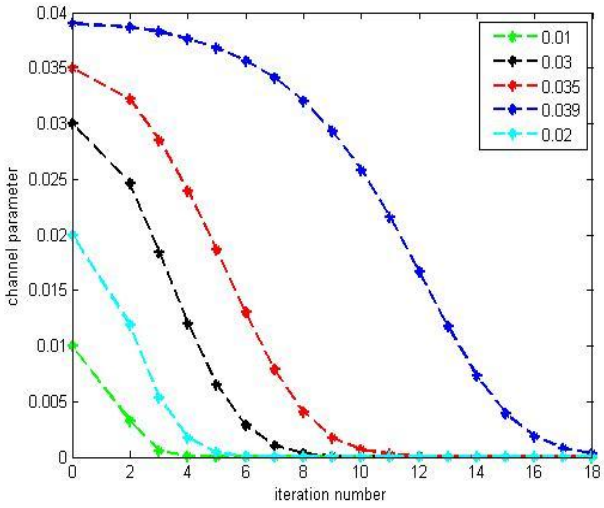

Figure 4 Gives A Threshold Value of 0.04 For (3, 6) Regular Code.

Table 1 Shows Channel Parameter ϵ^* for Different Ensembles over Binary Symmetry Channel

d_v	d_c	rate	ϵ^*
3	6	0.5	0.039
4	8	0.5	0.047
5	10	0.5	0.026
4	6	0.333	0.066

C. Density evolution on BEC:

Consider a (d_v,d_c) –regular code ensemble on the BEC with input alphabet $\{0+1\}$, the channel parameter or the cross over probability ε and the output alphabet {0, ?, 1}[8]. Similarly over BEC, let q_l be the probability of being erasure (?) from constraint node to (3) ode in iteration l and p_l be the probability of being erasure from bit node to constraint node in iteration l. Then

$$q_{l}=1-\left(1-p_{l}\right)^{\left(d_{c}-1\right)}$$

And

$$p_{l}=\varepsilon\left(q_{l-1}\right)^{d_{v}-1}$$

Substitute q_l in the above equation, then

$$p_{l}=\varepsilon\left(1-\left(1-p_{l-1}\right)^{\left(d_{c}-1\right)}\right)^{d_{v}-1} \quad (4)$$

The above equation represents the recursive equation to calculate threshold value over BEC. Threshold value ε^* is the maximum of all value of $p_l \in [0,1]$ such that $\lim_{l\to\infty} p_l = 0$.

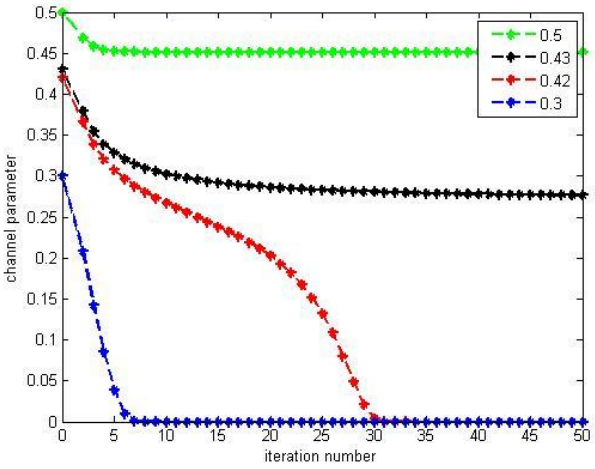

Figure 5 Gives A Threshold Value of 0.4293 For (3, 6) Regular Code

Table 2 Shows Threshold Value ε^* For Different Ensembles Over Binary Erasure Channel

d_v	d_c	rate	ε^*
3	4	0.25	0.6474
3	5	0.4	0.5176
3	6	0.5	0.4293
4	6	0.333	0.5061

V. CONCLUSION

In this paper, we analyzed LDPC codes over message passing decoders over the channels BSC and BEC and we also discussed about the ensemble of LDPC codes. We calculated threshold value for different ensembles of codes over the two channels mentioned. From the calculations we observed that Binary Erasure Channel has more threshold value from the Tabulated values in tables-1 and 2. If the channel parameter is large enough then the channel can retain the correct information even there is more noise effect over the channel during the transmission.

REFERENCES

[1] R. G. Gallager "Low Density Parity Check codes." IRE Transactions on Information Theory, vol 8, no. 1, pp. 21-28, 1962.

[2] D. J. C. MacKay and R. M. Neal, "Near Shannon limit performance of low density parity check codes," in Electronics Letters, vol. 32, no. 18, pp. 1645-1646, 1996.

[3] C. E. Shannon, "A Mathematical Theory of Communication". CSLI publications, 1948. [online] Available: http://math.harvard.edu/~ctm/home/text/others/shannon/entropy/entropy.pdf

[4] Luby, Michael, et al. "Analysis of low density codes and improved designs using irregular graphs." Proceedings of the thirtieth annual ACM symposium on Theory of computing. ACM, 1998.

[5] M. C. Davey and D. MacKay, "Low-density parity check codes over GF(q)," in IEEE Communications Letters, vol. 2, no. 6, pp. 165-167, June 1998.

[6] T. J. Richardson, M. A. Shokrollahi and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," in IEEE Transactions on Information Theory, vol. 47, no. 2, pp. 619-637, Feb 2001.

[7] Johnson, Sarah J."Introducing low-density parity-check codes." University of Newcastle, Australia, 2006.

[8] Valenti, Matthew C., Shi Cheng, and Rohit Iyer Seshadri. "Turbo and LDPC codes for digital video broadcasting." Turbo Code Applications. Springer Netherlands, 2005. 301-319.

[9] Morello, Alberto, and Vittoria Mignone. "DVB-S2: The second generation standard for satellite broad-band services." Proceedings of the IEEE 94.1 (2006): 210-227.

[10] Biglieri, Ezio. Coding for wireless channels. Springer Science & Business Media, 2005.

[11] Johnson, Sarah J. Iterative error correction: Turbo, low-density parity-check and repeat-accumulate codes. Cambridge University Press, 2009.

[12] Loeliger, H-A. "An introduction to factor graphs." IEEE Signal Processing Magazine 21.1 (2004): 28-41.

[13] Zhang, Xiaojie, and Paul H. Siegel. "Quantized iterative message passing decoders with low error floor for LDPC codes." IEEE Transactions on Communications 62.1 (2014): 1-14.