Study regarding the density evolution of messages and the characteristic functions associated of a LDPC code

S Drăghici¹², O Proștean¹, E Răduca², C Hațiegan², I Hălălae², I Pădureanu³, M Nedeloni² and L (Barboni) Hațiegan³
¹Politehnica University of Timisoara, Department of Automation and Applied Informatics, Vasile Parvan str., no. 2, 300223 Timisoara, Romania
²Eftimie Murgu University of Resita, Department of Electric Engineering and Informatics, Traian Vuia str., no. 1-4, 320085 Resita, Romania
³Eftimie Murgu University of Resita, Department of Mechanical and Materials Engineering, Traian Vuia str., no. 1-4, 320085 Resita, Romania

E-mail: s.draghici@uem.ro

Abstract. In this paper a method with which a set of characteristic functions are associated to a LDPC code is shown and also functions that represent the evolution density of messages that go along the edges of a Tanner graph. Graphic representations of the density evolution are shown respectively the study and simulation of likelihood threshold that render asymptotic boundaries between which there are decodable codes were made using MathCad V14 software.

1. Introductive notions about LDPCs

LDPC (Low Density Parity Check) codes represent a class of block linear error correcting codes put forward by R.G. Gallager in 1963 [1] whose usefulness has been rediscovered in the last two decades.

The essential difference as opposed to the other classes of block codes consists in the fact that the control and parity matrix H is sparsely populated (sparse) with binary values of 1 logic, and a greater prevalence of values of 0 logic, which facilitates an easier decoding, because LDPCs are a class of codes that facilitates parallel calculus, they work well for high speeds of data transmission (WiMax radio channels, ATM technologies for data transfer, radio communications in outer space and so on). Even when the H matrix does not have values of 1 logic sparsely distributed in its structure, it can be brought to a sparsely populated shape/ pattern, by linear combinations of rows and columns. Furthermore the H matrix can be divided in sub-matrix blocks of H, which are sparsely populated blocks.

Through all this the main code is divided into sub-codes that are easier to decode, and thus the speed of data transmission remains very high even when multiple decoders are used. The LDPC codes are regular and irregular and are characterized by the set of parameters \((n, d_v, d_c)\), where \(n\) is the length of the codeword. In regular LDPCs the H matrix is composed of exact \(d_v\) Hamming weights in columns and exact \(d_c\) Hamming weights in lines.

The LDPCs codes are represented by a bipartite Tanner graph (regular or irregular). In this kind of representation, in general, on the left hand side we have the variable nodes associated with bits in the code words, and on the right hand side, the parity control nodes, associated with the equations used to verify the parity. This two part graph is an ensemble characterised by two vectors [2-4]:
\[ \lambda = (\lambda_1, \lambda_2, \ldots, \lambda_d) \text{ and } \rho = (\rho_1, \rho_2, \ldots, \rho_d) \quad (1) \]

Where \( \lambda_i \) represents the fraction between the edges associated to the variable nodes with \( i \) degrees of freedom; \( \rho_j \) represents the fraction of Tanner graph edges associated parity check nodes with \( j \) degrees of freedom; Taking into account the relationship (1) the rate \( r \) of a LDPC code is defined by [2], [3]:

\[ r(\lambda, \rho) = 1 - \frac{\sum_{j \geq 2} \frac{\rho_j}{j}}{\sum_{i \geq 2} \frac{\lambda_i}{i}} = 1 - \frac{L}{N} \quad (2) \]

Where \( L \) is the total number of variable nodes, \( N \) is the total number of parity check nodes, \( d_i \) is the number of degrees of freedom of a variable node (which is the same for every variable node in the case of a regular LDPC), and \( d_c \) is the number of the degrees of freedom of a parity node.

A particular case is the regular Tanner - graph (associated to a regular LDPC), when [2], [3], [5]:

\[ N \cdot d_v = L \cdot d_c \text{ and } r = 1 - \frac{d_v}{d_c} \quad (3) \]

In practice it’s usual for the variable nodes to be represented by circles, and those of parity by squares, in the graphic representation of a bipartite Tanner graph. In the LDPC code study relevant is also the number \( E \) of the bipartite Tanner graph edges. Thus, for a Tanner graph with \( L \) variable nodes, with each having \( i \) degrees of freedom, the total number of edges \( E_v \), starting from these nodes is [2], [3]:

\[ L \cdot \sum_{j \geq 2} \frac{\lambda_j}{j} = L \cdot \frac{\lambda_i}{i} \int_0^1 \lambda(x)dx \quad (4) \]

Where the \( x \) variable is a continuous or meshed variable specific to a probable continuous or meshed distribution, determined by the type of data transmission channel used.

At the same time, one calculates the number of \( E_c \) edges exiting the parity check nodes, so the rate of a LDPC code can be written thus:

\[ r(\lambda, \rho) = 1 - \frac{\int_0^1 \rho(x)dx}{\int_0^1 \lambda(x)dx} \quad (5) \]

In the study of LDPCs binary symmetric channels are used (B.S.C.), binary erasure channels (B.E.C.), binary input additive gaussian noise channels (B.I.A.W.G.N.C.), binary input Laplacean channels (B.I.L.C.), and others.

2. Obtaining the evolution density function

2.1. Presenting the characteristic function, the L.L.R. reports, and the cost functions

The decoding algorithms for LDPCs used to consist in passing the messages along the edges of a Tanner graph, from the variable nodes \( v \) towards the parity check nodes \( c \), and back. For the graphic representation of this decoding procedure some cost function associated to the \( E_v \) and \( E_c \) edges of a Tanner graph are defined:
In Figure 1, $\theta$ is used to represent the function that corresponds to the initial message, $v$ and $c$ being variable nodes and parity check nodes. The cost functions depend on the iteration number, that is the message sent to a $l$ – from the variable node $v$ to the parity node $c$, is represented by the cost function $m_{vc}^{(l)}$. These are specific to the Gallager A and B decoding algorithms, which consist of evolution density transport and which supply the estimate number of incorrect messages that go through a Tanner graph at each iteration. This is how the notion of passing messages and entry and output alphabet appeared (see Figure 1). For mathematical reasons the cost functions of $m$ parameter are defined by the log-likelihood ratios (L.L.R.) [2], [3], [6]:

$$m = \ln \frac{P(x = 1 \mid y)}{P(x = -1 \mid y)} = \ln \frac{P(y \mid x = 1)}{P(y \mid x = -1)}$$

In relation (6) Bayes rule is applied, and the random $x$ and $y$ variables can be, for example, the entrances and exits of a B.S.C. as the one represented in the next figure [2]:

![Figure 2. The structure of a B.S.C.](image)

In Figure 2, $p$ represents the likelihood of erroneous transmission of a symbol in a word code. The notation 1 was chosen in place of 0 logic because in the functions characterizing the evolution density of LDPCs the atypical function $\text{sgn}(x)$ intervenes, which characterizes the error in decoding and defines the move from space $\{0, 1\}$ to space $\{-1, +1\}$ according to relation (7):

$$\text{sgn}(x) = \begin{cases} 0, & \text{if } x \geq 0; \\ 1, & \text{if } x \leq 0; \end{cases}$$

Figure 1. The cost functions $m$ corresponding to the edges of a Tanner graph.
It can be seen in \( x = 0 \) that \( \text{sgn}(x) \) is both zero and one with a probability of \( 1/2 \). From a mathematical standpoint it is practically an indetermination. Coming back to Relation (6), if we rewrite \( m = L(x/y) = L(y/x) \) - according to Bayes rule, than:

\[
\ln L(x/y_1, y_2, \ldots, y_l) = \sum_{i=1}^{l} \ln L(x/y_i)
\]

(8)

Where \( y_1, y_2, \ldots, y_l \) are independent random variable;
We note:

\[
\not L = L(x_i/y_i) \text{ and } \not l = \ln L
\]

(9)

Also, it is noted:

\[
p = 2 \cdot P(x_1 = 0/y_1) - 1, \quad q = 2 \cdot P(x_2 = 0/y_2) - 1
\]

(10)

The relation (10) determines the following relation:

\[
2 \cdot P(x_1 \oplus x_2 = 0/y_1, y_2) - 1 = p \cdot q
\]

(11)

Where \( x_1 \oplus x_2 \) is the sum of module 2 over field \( GF(2) \) - Galois Field [7];

From the relation (9) and (10) we observe that:

\[
2 \cdot P(x_i = 0/y_i) - 1 = \frac{L-1}{L+1} = \tanh\left(\frac{l}{2}\right)
\]

(12)

Taking into account relations (11) and (12) and extrapolating, we come to:

\[
\ln L(x_1 \oplus \ldots \oplus x_i/y_1, \ldots, y_l) = \ln \left[ 1 + \left( \prod_{i=1}^{l} \tanh\left(\frac{l_i}{2}\right) \right) \right]
\]

(13)

Where \( \not l_i = \ln L(x_i/y_i) \);
As a consequence of relation (13) a function \( \gamma(x) \) is defined, that characterizes the evolution density map [2]:

\[
\gamma : ( -\infty, +\infty ) \rightarrow GF(2) \times [0, +\infty) \text{ and } \gamma(x) := (\gamma_1(x), \gamma_2(x)) := \left( \text{sgn}(x), -\ln \tanh\left(\frac{|x|}{2}\right) \right)
\]

(14)

The act of defining the function \( \gamma(x) \) in relation (14) represents a functional transformed (Transform – T [9]) and makes the transition from the pairs space \((x_i, y_i) - \text{of code words, in the LLR space. In the LLR space the function } sgn(x) \text{ defines the erroneous reception or not of a code word (determined by the presence or lack of modification of the reception signal), and the } \ln \tanh\left(\frac{|x|}{2}\right) \text{ function represents the amplitude of the received signal. Sugestive to those presented above are the following Figures [3], [8]:} \)
Figure 3. The evolution density correspondent to variable node \( v \)

\[
\left( x_d, y_d \right) = \left( \sum_{i=1}^{d-1} x_i \pmod{2}, \sum_{i=1}^{d-1} y_i \pmod{2} \right)
\]  
(15)

In consequence of transform \(-T\) defined in the relation (14), relation (15) can be written thus:

\[
tanh \left( \frac{l_d}{2} \right) = \prod_{i=1}^{d-1} \tanh \left( \frac{l_i}{2} \right)
\]  
(16)

Taking into account the relations (15) și (16), the decoding process and the density map are represented suggestively in figure:

Figure 5. Transform-\( T; T^{-1} \) is transformed in reverse \( T \)

From Figure 5 it can be observed \( l_d \) is:
(17)

and \( I_i \) and \( I_d \) is the one given by the relations (13) and (16) respectively.

Returning to relation (14) which represents the evolution density map, it is demonstrated that [2] this function \( \gamma(x) \) is the likely allotment density of a function \( F(x) \) defined by [2], [9]:

\[
P\{z \in (-\infty, x]\} = F(x)
\]

where \( z \) is a random variable with values between \((−\infty,+\infty)\), and \( F(x) \) is continuous and nonnegative.

By defining a function \( G(x) \) similar to \( F(x) \), the convolution between the two functions in the real numbers domain \( \mathbb{R} \) can be determined by [2], [9]:

\[
(F \otimes G)(x) = \int_{\mathbb{R}} F(x-y) dG(y) = \int_{\mathbb{R}} G(x-y) dF(y)
\]

One demonstrates [10], with the Lesbegue-Stieltjes relations used in cumulative distribution functions that the convolution of likelihood functions \( F \otimes G \) corresponds to the convolution of their derivatives. Using the same Lesbegue-Stieljes relations we can define a function \( G(s,x) \) that associates \( s \) to the evolution density map defined through \( \gamma(x) \), thus [2], [11]:

\[
G(s,x) := \chi_{\{s=0\}} \cdot G^0(x) + \chi_{\{s=1\}} \cdot G^1(x)
\]

where \( \chi_{\{s\}} = \chi_{\{\gamma_i(x)=s\} \in \{0,1\}} \) - namely to \( \gamma_i(x) \) and \( \chi_{\{s=0\}} = 1 \) for \( s = 0 \) and is nil otherwise; also, \( \chi_{\{s=1\}} = 1 \) for \( s = 1 \) and is nil otherwise.

The function \( G(s,x) \) shown in relation (20) is defined by the variation domain of the \( x \) variable with values in the \( GF(2) \times [0, +\infty) \) set and is also a characteristic function, by associating \( F(x) \) to \( \gamma(x) \). This is represented by means of an additional functional operator \( \Gamma(s, F(x)) :\)

\[
\Gamma(s, F(x)) = \chi_{\{s=0\}} \cdot \Gamma_0(F(x)) + \chi_{\{s=1\}} \cdot \Gamma_1(F(x)), \text{ where } \Gamma_0(F(x)) = 1 - F\left(-\ln \tanh \left(\frac{x}{2}\right)\right),
\]

\[
\text{and } \Gamma_1(F(x)) = \left(\ln \tanh \left(\frac{x}{2}\right)\right).
\]

As likelihood functions, the relations (21) can be written as:

\[
\Gamma_0(F(x)) = P\{\gamma_1(z)=0, \gamma_2(z) \leq x\} = P\left\{z \geq -\ln \tanh \left(\frac{x}{2}\right)\right\}
\]

\[
\Gamma_1(F(x)) = P\{\gamma_1(z)=1, \gamma_2(z) \leq x\} = P\left\{z \leq \ln \tanh \left(\frac{x}{2}\right)\right\}
\]

This is important because if \( G_{\gamma_1} \) and \( G_{\gamma_2} \) are two characteristic functions of the \( \gamma \) type, then a function marked by a random value \( z = z_1 + z_2 \) corresponds to a characteristic function determined by \( G_{\gamma_1} \otimes G_{\gamma_2} \) [2].
The description of the evolution density for LDPC codes is achieved with an operator function of the $\Gamma(s, F(x))$ type, which is the characteristic function associated with the LDPC code. So, for a certain class of LDPC codes the group of characteristic functions specific to that class of codes is defined.

The function $\Gamma(s, F(x))$ has the quality of permitting its mirrored opposite.

Also, given that $G$ and $H$ are two characteristic functions of the $\Gamma$ type, that satisfy the relations (20), then the convolution between the $G$ and $H$ distribution types is [2]:

\[
G \otimes H = \mathcal{X}_{[\gamma=0]} \left[ \left[ (G^0 \otimes H^0) + \left( G^1 \otimes H^1 \right) \right] + \mathcal{X}_{[\gamma=1]} \left[ \left[ (G^0 \otimes H^1) + \left( G^1 \otimes H^0 \right) \right] \right] \right]
\]

(23)

2.2. Describing the evolution density of the LDPC codes with the help of characteristic function

The $P_l$ and $Q_l$ functions, are associated to the probable densities given by the cost functions $m^{(i)}_w$ and $m^{(i)}_v$, where $P_l$ and $Q_l$ are probabilities. By observing Figure 1 the equations for the messages under belief propagation can be written [3]:

\[
m^{(i)}_w = \begin{cases} m_w, & \text{if } l = 0 \\ m_w + \sum_{r \in \mathbb{C}} m^{(r)}_r & \text{if } l \geq 1 \\ \gamma^{-1} \left( \sum_{\mathbb{C}} \gamma^{(m^{(i-1)}_r)} \right) & \end{cases}
\]

(24)

where $C_v$ is the parity node set that converge from the variable node $v$, $V_c$ is the variable node set that converge from the $c$ parity node, and $m_0$ is the initial likelihood density (that corresponds to an initial LLR for a $x = \pm 1$ bit).

By reanalyzing the (4) and (5), one can write:

\[
\lambda(x) = \sum_{i \geq 2} \lambda_i \cdot x^{i-1} \quad \text{and} \quad \rho(x) = \sum_{i \geq 2} \rho_i \cdot x^{i-1}
\]

(25)

Taking into account the (14), (21), (22) and (25) relations, as well as the association between the characteristic function $F(x)$ and the function $\gamma(x)$, we obtain [2]:

\[
Q_l = \Gamma^{-1}(\rho(\Gamma(P_{l-1}))) = \Gamma^{-1} \left( \sum_{i \geq 2} \rho_i \cdot (\Gamma(\Gamma(P_{l-1})))^{\otimes(i-1)} \right)
\]

(26)

where $\Gamma^{-1}$ is the reverse of the characteristic function $\Gamma$.

Relation (26) expresses the fact that the density of the message passed from check node $c$ to variable node $v$ at round $l$ is equal to $Q_l$. Also we write that the density of the message passed from variable node $v$ to check node $c$ at round $l$ is equal to $P_l$.

Taking into account the function $\gamma(x)$ definition and the fact that the convolution of the characteristic functions $G_{z_1}$ and $G_{z_2}$ has as correspondent the sum of the random variables $z_1$ and $z_2$, as well as the relation (24), it is determined that:

\[
P_l = P_0 \otimes \lambda(Q_l) = P_0 \otimes \sum_{i \geq 2} \lambda_i (Q_l)^{\otimes(i-1)}
\]

(27)

where $P_0$ is the density of the random variable describing the channel.

By combining the (26) and (27) relations, the evolution density for the LDPCs is obtained through a recursive mathematical formula with values of $P_l$ and $P_{l-1}$ described with the help of the $\Gamma$ characteristic functions:
\[ P_l = P_0 \otimes \lambda \left( \Gamma^{-1}(\rho(\Gamma(P_{l-1}))) \right) \]  \hfill (28)

where \( l \) being the iteration value.

3. **Representing and simulating the evolution density for a particular LDPC case**

To being with we shall present graphically in MathCad 14 the likelihood thresholds \((f_0(x))\) and \((f_1(x))\):

\[ f_0(x) := -\ln \left( \tanh \left( \frac{x}{2} \right) \right) \]  \hfill (29)

\[ f_1(x) := \ln \left( \tanh \left( \frac{x}{2} \right) \right) \]  \hfill (29')

Those functions determined by the relation (22) for different values of the random variable \( x \), have the followings representation:

\[ \begin{array}{c}
\includegraphics[width=0.5\textwidth]{figure6.png} \\
\text{Figure 6. The simulation of the } f_0 \text{ and } f_1 \text{ functions for } x \text{ with values between } -10 \div 10
\end{array} \]

\[ \begin{array}{c}
\includegraphics[width=0.5\textwidth]{figure7.png} \\
\text{Figure 7. The simulation of the } f_0 \text{ and } f_1 \text{ functions for } x \text{ with values between } -25 \div 25
\end{array} \]
It can be observed that the larger the values of $x$, the more the likelihood thresholds go towards zero. Therefore, the bigger the code dimension, the better its decoding goes.

We simulate and represent the initial evolution density (that corresponds to the zero iteration) for an entrance binary channel which is affected by white Gaussian noise (BIAWGN), and that has the initial evolution density determined by the following relation [2]:

$$P_0(l) = \frac{\sigma^2}{8 \cdot \pi} \cdot e^{\left(-\frac{2 \sigma^2}{\sigma^2}\right)}$$

(30)

where $\sigma$ represents the signal - noise ratio, and $l$ the iteration value.

We have chosen to simulate the evolution density for the BIAWGN channel as well as for the case of a regular LDPC, because, in this context the evolution density of the messages has a continuous distribution, which makes it easier the represent in MathCad 14 [12], as opposed to a discrete distribution. In the simulation different values for $\sigma$ will be given. Also, taking into account relation (28) we shall simulate $P_0$ for different iterative steps ($P_{01}$, $P_{02}$ and $P_{03}$, $P_{11}$, $P_{21}$ and $P_{31}$): $\sigma_1 := 0.8$, $\sigma_2 := 0.5$, $\sigma_3 := 1$
\[ e = 2.71, \pi = 3.1415, 1 \approx -10.50 \]

\[
P_{01}(l) = \sqrt{\frac{\sigma_1^2}{8\pi e}} \left(1 - \frac{2}{(\sigma_1^2)}(\sigma_2)^2\right)
\]

\[
P_{02}(l) = \sqrt{\frac{\sigma_2^2}{8\pi e}} \left(1 - \frac{2}{(\sigma_2^2)}(\sigma_3)^2\right)
\]

\[
P_{03}(l) = \sqrt{\frac{\sigma_3^2}{8\pi e}} \left(1 - \frac{2}{(\sigma_3^2)}(\sigma_3)^2\right)
\]

\[ P_{11}(l) := P_{01}(l)^2, P_{21}(l) := P_{01}(l)^3, P_{31}(l) := P_{01}(l)^5 \]

**Figure 10.** The simulation of the evolution density of messages

### 4. Results, conclusions, observations, future research directions

One notices that the more utterances there are in Figure 10, \( P_l \) decreases, therefore \( \ln P_l \) increases, thus the evolution density of the messages that go through a Tanner graph also increases. The simulation was carried out for a regular LDPC \((d_e = 3, d_v = 6)\), with a \( 1/2 \) rate. A more simple LDPC was chosen, because the iteration computing according to the relation (28) is very complex. Aproximations have been carried out, taking into account that \( F(x) \) is a function with a cumulative distribution [9]:

\[
\lim_{x \to \infty} F(x) = 1
\]

Therefore we may consider \( F(x) \) being constant on several portions.

For the same values \((d_e = 3, d_v = 6)\) we have considered in relation (28) \( \lambda(x) = x^2 \) and \( \rho(x) = x^5 \). Thus, we have simplified greatly the recursive relation of evolution density. As a consequence of this fact, the decoding algorithms are linear.

The evolution density represents the essence of the decoding process of LDPCs, the recursive function by which it is expressed being directly associated to a covering mode of a Tanner graph by the code words. It is an analysis method based on asymptotic edges.

Based on this principle all decoding algorithms (soft or hard) are implemented. The most known are the Gallager A and B [1] algorithms. In the last years there have been implemented algorithms
taking into account the rapid Fourier and Laplace transformations of the characteristic functions, this
being an actual and future study problem.

The study of the characteristic functions and of the evolution density is important because the
implementing of all soft and hard decoding algorithms at LDPCs is carried out with their help.

Acknowledgment
The work has been funded by the Sectoral Operational Programme Human Resources Development
2007-2013 of the Ministry of European Funds through the Financial Agreement
POSDRU/159/1.5/S/132395.

References
[1] Gallager R G 1963 Low Density Parity-Check Codes, M.I.T. Press, Massachusetts, USA
[2] Richardson T J, Shokrollahi A and Urbanke R L 2001 Design of Capacity-Approaching
Irregular Low-Density Parity-Check Codes, IEEE Transactions on Information Theory 47(2)
619-637
[3] Richardson T J and Urbanke R L 2001 The Capacity of Low Density Parity Check Codes under
Message-Passing Decoding, IEEE Transactions on Information Theory 47(2) 599-618
[4] Drăghici S Scientific report No. 1 – Ph.D. thesis, unpublished
[5] https://www.youtube.com/watch?v=_45M-dO99-M&list=PL85783E9E7B5F2C0D&index=31,
existing at 27.01.2016
[6] Richardson T J and Urbanke R 2004 Fixed points and stability of density evolution,
Communications in Information and Systems 4(1) 103-116
[7] Draghici S, Proștean O, Răduca E, Anghel-Drugărin C, Chioncel C, Răduca M and Rudolf C
2015 The relations between the mathematic ideal, the generating polynome and the matrixes
G, H used in encodings, Procedia - Social and Behavioral Sciences 197 2032-2039
[8] Shokrollahi A 2003 LDPC Codes: An Introduction Technical Report Digital Fountain Inc. -
Zürich
[9] https://en.wikipedia.org/wiki/Cumulative_distribution_function, existing at 27.01.2016
[10] https://en.wikipedia.org/wiki/Lebesgue%2E%E2%80%93Stieltjesintegration
[11] https://www.youtube.com/watch?v=bGup-R--bEU&index=32&list=PL85783E9E7B5F2C0D,
existing at 27.01.2016
[12] Macfarlan K H and McClain S T 2011 The use of Mathcad functions in teaching an
undergraduate aerospace propulsion course ASME Turbo Expo Conference, Vancouver,
Canada, June 6-10, pp 505-510