PERFORMANCE OF A SIMPLIFIED SOFT-DISTANCE DECODING ALGORITHM FOR LDPC CODES OVER THE RAYLEIGH FADING CHANNEL

L.J. ARNONE[†], M. LIBERATORI[†], D.M. PETRUZZI[†],
P.G. FARRELL[‡]　and　J. CASTIÑEIRA MOREIRA[§]

† Facultad de Ingeniería, Universidad Nacional de Mar del Plata, Argentina.
‡ Department of Communications Systems, Lancaster University, Lancaster, United Kingdom.
Paddy.Farrell@virgin.net
§ Facultad de Ingeniería, Universidad Nacional de Mar del Plata, Argentina. CONICET.
casti@fi.mdp.edu.ar

Abstract— **In this paper, we investigate the performance of a Soft-Input soft-Output decoding algorithm for LDPC codes that uses Euclidean distance as its metric, in the Rayleigh fading channel. It is found that its Bit Error Rate performance is close to that of traditional decoding algorithms like the Sum-Product algorithm and its logarithmic version. Main characteristics of the proposed algorithm and its modification to perform over the Rayleigh channel are described. This algorithm uses squared Euclidean distance as the metric, does not require knowledge of the signal-to-noise ratio of the received signal, and is less complex to implement than other soft-input, soft-output algorithms.**

Keywords— **LDPC codes, Rayleigh Channel, Soft Distance decoding.**

I. INTRODUCTION

Low-density parity-check (LDPC) codes (MacKay and Neal, 1997; MacKay, 1999) have become one of the most interesting options among block error-correcting codes when a high Bit Error Rate (BER) performance is desired. It is known that by using message passing decoding algorithms, like the classic Sum-product (SP) or its logarithmic version, called LogSP, or the proposed Simplified Soft Distance (SSD) decoding algorithm, which is based on Euclidean Distance metric (Farrell and Castiñeira Moreira, 2009; Castiñeira Moreira and Farrell, 2008) irregular (LDPC) codes can achieve reliable transmission at signal-to-noise ratios (SNR) extremely close to the Shannon limit, on the additive white Gaussian noise (AWGN) channel. They outperform turbo codes of the same block size and code rate (MacKay and Neal, 1997; MacKay, 1999; How *et al.*, 2001). LDPC codes have certain advantages, such as simple descriptions of their code structure and fully parallelizable decoding implementations (How *et al.*, 2001).

However, their potential as capacity achieving codes for other channels, has not been completely established yet. Preliminary results are available which suggest that they can achieve capacity for a wide range of channels.

In How *et al.* (2001) irregular LDPC codes of large lengths were analyzed in a flat uncorrelated Rayleigh fading channel. The channel model assumed in How *et al.* (2001), was a slow (large coherence time) frequency non-selective (flat) Rayleigh fading channel. It is shown that LDPC codes have also an excellent BER performance over the Rayleigh Channel.

We want to study the performance of a decoding algorithm for LDPC codes that operates with iterative interchange of information like the SP algorithm, but utilizes soft distance as its metric. The proposed algorithm uses an antilog-sum as its main operation for performing the interchange of information characteristic of an iterative Soft-Input Soft-Output (SISO) decoding algorithm for LDPC codes. We adopt the Rayleigh channel model used in How *et al.* (2001), that is, a slow memory-less frequency non-selective (flat) Rayleigh fading channel with channel state information (CSI). The channel dynamics are explicitly taken into account by considering a block-independent fading model.

II. BINARY LDPC CODES

LDPC codes belong to a very efficient class of linear block codes. They are constructed by designing a sparse (few nonzero elements scattered among rows and columns) parity check matrix related to a generator matrix G, which is used to encode a message vector m with k elements in a code word $c=G.m$, with n elements.

Any word belonging to the code satisfies the syndrome condition: $H.c=0$. This matrix satisfies certain conditions that provide optimum performance in terms of BER (MacKay and Neal, 1997; MacKay, 1999).

Binary LDPC codes operate with message and codeword vectors defined over the binary field, so that their components belong to the discrete alphabet {0,1}.

Digital transmission has the best performance if polar format is adopted, i.e. bits are transmitted by sending signals from the discrete alphabet {-1,1}. Then, codeword c is sent through the channel as a signal s, where it is affected by the presence of noise and by multipath fading. The aim of the decoding algorithm is to find the vector d, considered as an estimation of the transmitted vector, able to satisfy the condition $H.d=0$.

The LDPC SP decoding algorithm described in MacKay and Neal (1997) and MacKay (1999) operates

over a given bipartite graph (MacKay and Neal, 1997) and makes an estimation of the A Posteriori Probability (APP) of each symbol as a function of the received symbol, and the properties of the channel. In this sense, the traditional decoding algorithm requires from the knowledge of the signal-to-noise ratio present at the channel. The proposed SSD algorithm also operates over a given bipartite graph associated with the LDPC code, but it does not require from the knowledge of the signal-to-noise ratio.

In this paper we present simulation results of the BER performance over the Rayleigh fading channel of two LDPC codes, the $C_{LDPC}(2560,1280)$ LDPC code, and the $C_{LDPC}(204,102)$, whose construction details can be found in Gallager (2012).

III. CHARACTERISTICS OF THE RAYLEIGH FADING CHANNEL

In this paper the channel under study is the Rayleigh fading channel. Medium and large size LDPC codes are decoded using both the LogSP algorithm, and the SSD algorithm.

Rayleigh fading is due to multipath reception. The mobile antenna receives a large number N of waves reflected by obstacles. Due to cancellation effects, the received instantaneous power is defined as a random variable. In these environments, it is reasonable to think that when an impulse signal is transmitted, a pulse train is received.

Since there are multiple propagation paths, an attenuation factor $\alpha_n(t)$ can be associated with each one of them. There is also a time delay $\tau_n(t)$. Both parameters are time dependent because in general terms the channel is time varying.

The transmitted signal can be represented as:

$$s(t) = \Re\left[s_b(t)e^{j2\pi f_c t}\right], \tag{1}$$

where $s_b(t)$ is the baseband signal and f_c is the carrier frequency. The received signal is:

$$y(t) = \sum_n \alpha_n(t)s(t - \tau_n(t)) \cdot \tag{2}$$

By combining Eq. (1) and Eq. (2):

$$y(t) = \Re\left\{\sum_n \alpha_n(t)s_b(t - \tau_n(t))e^{-j2\pi f_c[t-\tau_n(t)]}\right\}. \tag{3}$$

The baseband equivalent of the received signal is

$$y_b(t) = \sum_n \alpha_n(t)e^{-\theta_n(t)}s_b(t - \tau_n(t)), \tag{4}$$

where $\theta_n(t)=2\pi f_c \tau_n(t)$ is the phase of the n-th path. Thus, the baseband impulse response is:

$$\alpha(t) = \sum_n \alpha_n(t)e^{-\theta_n(t)} \cdot \tag{5}$$

The phase related with each path can change in 2π radians when the delay varies $1/f_c$. The wavelength associated with this frequency is much smaller than the distance between devices, so it is reasonable to assume that the phase is a random variable with uniform distribution between 0 and 2π. We also assume that the phases of each path are independent, so that:

$$p(\theta) = \frac{1}{2\pi} \quad -\pi \le \theta \le \pi \cdot \tag{6}$$

On the other hand, if the number of multiple paths is large, we can apply the Central Limit Theorem: each path can be modeled as a complex random Gaussian variable with circular symmetry with time as a variable. This model is known as Rayleigh fading channel.

The channel can be modeled as a circularly symmetric complex Gaussian random variable that is of the form (Tse and Viswanath, 2005):

$$\alpha = \alpha_{Re} + j\alpha_{Im}, \tag{7}$$

where real and imaginary parts are zero mean independent and identically distributed (iid) Gaussian random variables with zero mean and variance σ^2. It is reasonable to suppose that the magnitude of this variable has a Rayleigh distribution for environments with multiple reflectors.

The magnitude α is a vector of normalized Rayleigh fading factors with $E(\alpha^2)=1$ and probability density function (pdf):

$$P(\alpha) = 2\alpha e^{-\alpha^2} \cdot \tag{8}$$

The above description corresponds to he Rayleigh fading channel model. A detailed description of this model can be seen in Tse and Viswanath (2005).

A given coderword c is generated by the LDPC encoder and then transmitted as a signal s over the Rayleigh channel using BPSK modulation. The signal is affected by the random gain and noise present in this channel, and is received as:

$$\mathbf{y} = \boldsymbol{\alpha}.\mathbf{s} + \mathbf{n} \cdot \tag{9}$$

where $\mathbf{y}$ is the received vector, $\boldsymbol{\alpha}$ is the normalized Rayleigh fading vector, $\mathbf{s}$ is the transmitted signal vector and $\mathbf{n}$ is a Gaussian random variable vector, with zero mean and variance σ^2. The product $\boldsymbol{\alpha}.\mathbf{s}$ is the element-wise multiplication between $\boldsymbol{\alpha}$ and $\mathbf{s}$, and it represents the operation described in Eq. (4). Vectors $\mathbf{y}$, $\boldsymbol{\alpha}$, $\mathbf{s}$, $\boldsymbol{\alpha}.\mathbf{s}$ and $\mathbf{n}$ are all of n components.

If the channel scale factor α is known, we operate as with CSI, and a process can be done at the receiver, which consists on performing a compensation of the random gain of the channel, known as gain compensation, by dividing each received symbol by α:

$$\mathbf{r} = \mathbf{y}/\boldsymbol{\alpha} = (\boldsymbol{\alpha}.\mathbf{s}+\mathbf{n})/\boldsymbol{\alpha} = \mathbf{s}+\widetilde{\mathbf{n}} \cdot \tag{10}$$

In Eq. (10), the received vector $\mathbf{r}=\mathbf{y}/\boldsymbol{\alpha}$ is the element - wise division of $\mathbf{y}$ by $\boldsymbol{\alpha}$, and $\tilde{\mathbf{n}}$ is scaled additive noise, taking into account channel coefficients. All vectors in Eq. (10) are of n components.

The independent and identically distributed (i.i.d) fading model is a widely used assumption for frequency nonselective wireless channels. The simulated scheme operates using CSI. At the receiver, availability of the CSI will improve the probability of error (Hall, 1996).

IV. A BRIEF DESCRIPTION OF A SOFT DISTANCE ITERATIVE DECODING ALGORITHM

The soft distance (SD) algorithm (Castiñeira Moreira and Farrell, 2008) is an iterative decoding algorithm that

operates over the factor (Tanner) graph (MacKay and Neal, 1997; MacKay, 1999) of a binary low-density parity-check (LDPC) error-correcting code. As usual, the graph has n symbol nodes and m parity nodes, with edges, as defined by the parity-check matrix of the code, connecting the symbol and parity nodes. The SD algorithm uses Soft distance metric information, which is passed from symbol nodes to parity nodes (the horizontal step in the algorithm), and then from parity nodes to symbol nodes (the vertical step), in an iterative manner.

Assuming the transmission of coded binary information $c=(c_1 \quad c_2 \dots c_j \dots \dots c_n)$ in normalized polar format with signals of amplitudes ± 1, and for a received vector $r=(r_1 \quad r_2 \dots r_j \dots \dots r_n)$, which has been previously gain adjusted since we lie on CSI, we calculate the vectors:

$$d_0^2(j) = (\alpha r_j + 1)^2, \ d_1^2(j) = (\alpha r_j + 1)^2, \quad j = 1,2,...,n .(11)$$

These first soft estimates of the code symbols are used to initialize the algorithm by setting the following coefficients q_{ij}^0 and q_{ij}^1 at each symbol node:

$$q_{ij}^0 = d_0^2(j), \quad q_{ij}^1 = d_1^2(j),$$
$$j = 1,2,...,n, \quad i = 1,2,...,m. \tag{12}$$

These are then passed to the parity-check nodes (first horizontal step) connected to each symbol node, where the following coefficients are computed:

$$r_{ij}^0 = -\log_2 \sum_{s:s_j=0} \left[2^{-\left(\sum_{k \in N(i)\backslash j} q_{ik}^x\right)} \right],$$

$$r_{ij}^1 = -\log_2 \sum_{s:s_j=1} \left[2^{-\left(\sum_{k \in N(i)\backslash j} q_{ik}^x\right)} \right]. \tag{13}$$

$$x \in \{0,1\}$$

Here $N(i)\backslash j$ is the set of indexes of the symbol nodes connected to the parity node h_i, excluding the symbol node s_j. The inner summations add together the q values corresponding to all the possible 0-state or 1-state configurations of the $N(i)\backslash j$ code symbols connected to parity check node h_i, and the negative antilogs of these coefficient sums are then added in the outer summation. Coefficients r_{ij}^0 and r_{ij}^1 are then passed back (first vertical step) to symbol node s_j. At each symbol node s_j the initial values of q_{ij}^0 and q_{ij}^1 are added to the r values passed from each node connected to it, to form a second symbol estimate at each node:

$$q_{ij}^0 = d_0^2(j) + \sum_{k \in M(j)\backslash i} r_{kj}^0, \quad q_{ij}^1 = d_1^2(j) + \sum_{k \in M(j)\backslash i} r_{kj}^1 . \tag{14}$$

Here $M(j)\backslash i$ is the set of indexes of all the parity nodes connected to the symbol node s_j, excluding parity node h_i. These horizontal and vertical steps are then iterated an appropriate number of times to give a final estimate for each received symbol, given by:

$$\hat{d}_j^0 = \arg\min_x \left[d_x^2(j) + \sum_{k \in M(j)} r_{kj}^x \right]. \tag{15}$$
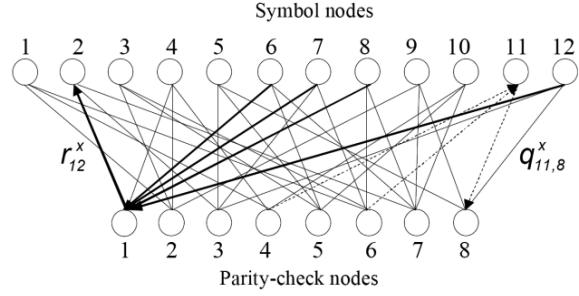
A. Simplified Soft-distance decoding algorithm

Figure 1. An example the bipartite graph of a LDPC code C_{LDPC} (12,4) and calculation procedure of the horizontal (dark line) and vertical (dotted line) steps.

The Sof-Distance Decoding algorithm summarized in the previous section can be simplified, in order to significantly reduce its implementation complexity.

The simplification procedure aims to reduce the complexity associated to determining the number of state configurations required to calculate the expressions in Eq. (13). This problem can be controlled by adapting a technique originally proposed in MacKay and Neal (1997). This involves working with sums and differences of distance antilogs, as the following expressions for the horizontal step indicate. We define:

$$Q_{ij}^x = b^{-q_{ij}^x}, \quad R_{ij}^x = b^{-r_{ij}^x}, \quad x \in \{0,1\}, \tag{16}$$

and

$$A_{ij} = Q_{ij}^0 + Q_{ij}^1 = b^{a_{ij}}, \quad B_{ij} = Q_{ij}^0 - Q_{ij}^1 = (-1)^{s_{ij}} b^{b_{ij}}, \tag{17}$$

where b is the base of the exponential or logarithmic expression and:

$$a_{ij} = \log_b(A_{ij}), \quad b_{ij} = \log_b(B_{ij}),$$

and if $(-q_{ij}^0) \geq (-q_{ij}^1)$ then $s_{ij}=0$ or else $s_{ij}=1$.

Then,

$$R_{ij}^0 = \frac{1}{2}\left(\prod_{k \in N(i)\backslash j} A_{ik} + \prod_{k \in N(i)\backslash j} B_{ik} \right)$$

$$R_{ij}^1 = \frac{1}{2}\left(\prod_{k \in N(i)\backslash j} A_{ik} - \prod_{k \in N(i)\backslash j} B_{ik} \right). \tag{18}$$

The simplification procedure consists on the use of logarithmic operations that are applied to the involved quantities. In this procedure, a logarithm of a sum of terms is solved by using a simplifying expression. Thus, for the case $C=A+B$, $C=b^\gamma$, $A=b^\alpha$, $B=b^\beta$, then $b^\gamma = b^\alpha + b^\beta$ and:

$$\gamma = \max(\alpha,\beta) + \log_b\left(1 + b^{-|\beta-\alpha|}\right) = \max(\alpha,\beta) + f_+(\alpha,\beta) .(19)$$

For the case $C=A-B$

$$\gamma = \max(\alpha,\beta) - \log_b\left(1 - b^{-|\beta-\alpha|}\right) = \max(\alpha,\beta) - f_-(\alpha,\beta) .(20)$$

In an implementation typically developed using programmable logic like Field Programmable Gate Arrays (FPGA), which follows the procedure shown in Table 1, expressions of the form:

$$f_+(\alpha,\beta) = \log_b\left(1 + b^{-|\beta-\alpha|}\right), \quad f_-(\alpha,\beta) = \log_b\left(1 - b^{-|\beta-\alpha|}\right) .(21)$$

are finally calculated by means of look-up tables.

In this simplification procedure, working with the sums and differences of the distance antilogs helps to

reduce overflow and underflow problems, but overflow problems still remain when using a relatively large number of decoding iterations. In order to avoid this, $R_{ij}{}^x$ values are normalized in the following way at the end of each iteration:

$$R_{N,ij}^x = R_{ij}^x \left(\frac{2}{R_{ij}^0 + R_{ij}^1} \right), \quad r_{n,ij}^x = -\log_b(R_{N,ij}^x). \quad (22)$$

The above are the main steps in the simplification of the SD algorithm, which leads to the Simplified Soft-Distance (SSD) algorithm. We will apply this algorithm to LDPC codes operating over the Rayleigh fading channel. Further details of the simplification procedure and of programmable logic implementation for the Aditive White Gaussian Channel can be found in Arnone *et al.* (2009). Table 1 describes steps of the SSD algorithm.

V. SIMULATION RESULTS

Figure 2 shows simulation results obtained for LDPC codes operating over the Rayleigh fading channel. The LDPC code $C_{LDPC}(2560,1280)$ (Gallager, 2012) is decoded using either the proposed SSD algorithm or the LogSP algorithm, with 16 iterations. Additional simulations were done for a shorter LDPC code, the $C_{LDPC}(204,102)$ code (Gallager, 2012), decoded using either the proposed SSD algorithm or the LogSP algorithm with 16 iterations.

Simulation of the LDPC code $C_{LDPC}(2560,1280)$ is done by transmitting 5000 codewords of length 2560, which correspond to messages of length 1280. At approximately 5.5 dB, the decoding using the SSD algorithm presents zero error in this transmitted block, thus the corresponding value of BER does not appear in Fig. 2. Simulation of the LDPC code $C_{LDPC}(204,102)$ is done by transmitting 30000 codewords of length 204, which correspond to messages of length 102. At approximately 9.0 dB, the decoding using the SSD algorithm presents zero error in this transmitted block, thus the corresponding value of BER does not appear in Fig. 2. As it is seen, the BER performance of the proposed SSD decoding algorithm is the same as that of the LogSP algorithm, over the Rayleigh fading channel. In addition, the SSD algorithm has a slight lower complexity than the LogSP algorithm and does not require the knowledge of the signal-to-noise ratio of the channel.

Decoding complexity of the SSD algorithm can be seen in Farrell *et al.* (2011).

VI. CONCLUSIONS

In this paper a squared Euclidean distance SISO decoder has been applied to the decoding of LDPC codes operating over the Rayleigh fading channel. The proposed algorithm is used in its simplified version, called Simplified Soft Distance (SSD) algorithm. It is seen that there is no BER performance degradation of the proposed algorithm with respect to classic decoding algorithms like the SP, or its logarithmic (simplified) version, called LogSP algorithm. The proposed SSD algorithm performs very well in channels like the AWGN channel and the Rayleigh channel, and it offers an even lower

Table 1. Steps of the Simplified Soft-Distance algorithm.

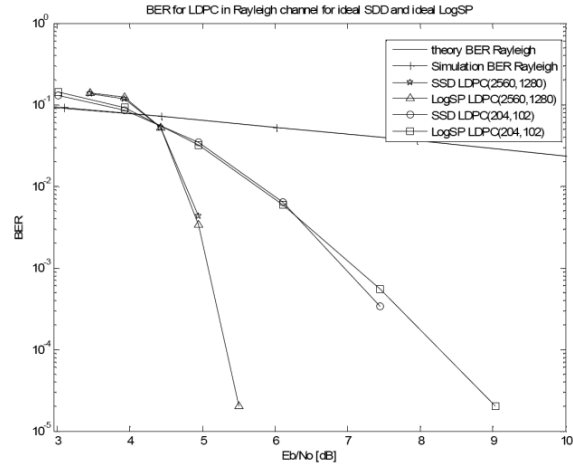| **Initialization** |
| --- |
| $q_{ij}^0 = d_0^2(j)$, $q_{ij}^1 = d_1^2(j)$ |
| **Horizontal Step** |
| $a_{ij} = f_+\left(q_{ij}^0, q_{ij}^1\right)$, $b_{ij} = -f_-\left(q_{ij}^0, q_{ij}^1\right)$, |
| If $\left(-q_{ij}^0\right) \geq \left(-q_{ij}^1\right)$ $s_{ij} = 0$ else $s_{ij} = 1$ |
| $c_{ij} = \sum_{k \in N(i) \backslash j} b_{ik} - \sum_{k \in N(i) \backslash j} a_{ik}$ |
| if $\sum s_{ik}$ is even |
| $r_{n,ij}^0 = -f_+(0, c_{ij})$, $r_{n,ij}^1 = f_-(0, c_{ij})$ |
| if $\sum s_{ik}$ is odd |
| $r_{n,ij}^0 = f_-(0, c_{ij})$, $r_{n,ij}^1 = -f_+(0, c_{ij})$ |
| **Vertical Step** |
| $q_{ij}^0 = d_0^2(j) + \sum_{k \in M(j) \backslash i} r_{n,kj}^0$, $q_{ij}^1 = d_1^2(j) + \sum_{k \in M(j) \backslash i} r_{n,kj}^1$ |
| **Decision** |
| $\hat{d}_0^2(j) = r_{n,ij}^0 + q_{ij}^0$, $\hat{d}_1^2(j) = r_{n,ij}^1 + q_{ij}^1$ |
| If $\hat{d}_1^2(j) < \hat{d}_0^2(j)$ then $\hat{c}(j) = 1$ else $\hat{c}(j) = 0$ |

Figure 2: BER performance of LDPC codes over a Rayleigh fading channel decoding with either the LogSP algorithm or the SSD algorithm.

complexity than the logarithmic version of the SP algorithm (LogSP), but with essentially identical performance. In addition to a lower complexity, the SSD algorithm presents another important advantage that is to not requiring an estimate of the channel SNR, as it happens to the SP algorithm for having its best performance.

Therefore the SDD algorithm appears to be a very suitable Euclidean metric SISO iterative decoding algorithm for LDPC codes, and for use in a wide range of practical applications and channels.

REFERENCES

Arnone, L, J. Castiñeira Moreira and P.G. Farrell, "FPGA implementation of a Euclidean distance metric SISO Decoder," *Tenth International Sympo-*

sium on Communications Theory and Applications, Ambleside, UK (2009).

Castiñeira Moreira, J. and P.G. Farrell. "Soft-Decision Siso Decoding of Error-Control Codes," *XIII Congreso Internacional de Telecomunicaciones, SENACITEL* (2008).

Farrell, P.G. and J. Castiñeira Moreira. "Soft-in, Soft-out (SISO) Decoding with the Euclidean Metric," *Tenth International Symposium on Communications Theory and Applications,* Ambleside, UK, (2009).

Farrell P.G, L. Arnone and J. Castiñeira Moreira, "Euclidean distance soft-input soft-output decoding algorithm for low-density parity-check codes," *IET Communications,* **5**, 2364-2370 (2011).

Gallager, *http://www.inference.phy.cam.ac.uk/mackay/ CodesGallager.html.* Available: On Line (2012).

Hall, E.K., Performance and design of turbo codes over Rayleigh fading channels, MSc Thesis. Virginia University (1996).

How, J., P.H. Siegel and L.B. Milstein, "Performance analysis and code optimization of low density parity check codes on Rayleigh fading channels," *IEEE Journal on Selected Areas in Communications*, **15**, 924–934 (2001).

MacKay, D.J.C. and R.M. Neal. "Near Shannon limit performance of low density parity check codes," *Electronics Letters,* **33,** 457-458 (1997).

MacKay, D.J.C., "Good Error-Correcting Codes based on Very Sparse Matrices," *IEEE Trans Information Theory*, **IT-45,** 399-431 (1999).

Tse, D. and P. Viswanath. *Fundamentals of Wireless Communication.* Cambridge University Press (2005).