# USING OF LDPC CODES IN STEGANOGRAPHY

**I. DIOP[1], S .M FARSSI[1], M.CHAUMONT[2,3,4],O. KHOUMA, H. B DIOUF, K .TALL[1], K .SYLLA1**

**[1]Polytechnic School Of Cheikh Anta Diop University Dakar Sénégal**
**[2]Nimes University De, F-30021 Nîmes Cedex 1, France**
**[3]Montpellier 2 University, Umr5506-Lirmm, F-34095 Montpellier Cedex 5, France**
**[4]Cnrs, Umr5506-Lirmm, F-34392 Montpellier Cedex 5, France**

Email: idydiop@yahoo.fr;farsism@yahoo.com

## ABSTRACT

Steganography is the art of secret communication [1] Since the advent of modern steganography, in the 2000s, many approaches based on error correcting codes (Hamming, BCH, RS ...) have been proposed to reduce the number of changes in the roof while inserting the maximum bit. Jessica Fridrich's works have shown that sparse codes best approach the theoretical limit of efficiency of insertion. Our research works are a continuation of those on low-density codes (LDGM) proposed by T. Filler in 2007. In this paper we propose a new approach to correcting codes using LDPC codes rather than LDGM. The complexity of our approach is much less than that of T. Filler which makes it usable in practice.

**Keywords:** *LDPC Codes, Encoding, Decoding, Steganography.*

## 1 INTRODUCTION

Steganography is the art of secret communication. Steganography consists in hiding a message into a medium-trivial for example, a picture, video, sound, so that this insertion is statistically undetectable.

One of the assumptions made before 2011, was to say that it was sufficient to minimize the number of modification of the medium to ensure maximum security of the scheme. This assumption casts doubt over question the competition BOSS. That is to say, while minimizing the number of changes the study of error correcting codes to insert a message is an interesting problem. Many steganographic schemes based on the principle of "embedding matrix" (there is a use hijacked correcting codes) have been proposed in the past BCH, RS ... [2] [3] [4] These patterns are usually far from the theoretical limit of efficiency.

Our work continues the same approach to minimize the number of changes by providing support host-based approach using LDPC error correcting codes. Our approach has the advantage of being less complex than the LDGM approach [5]while being very close to the theoretical limit of efficiency of insertion. In Section 2 we will recall the principle of operation of the LDPC codes. In Section 3 we will explain our approach. In Section 4 we will present the experimental results.

## 2 CODES LDPC

### 2.1 Definition

A code LDPC is a code which matrix of parity checks H is of weak density. The weak density is explained by the fact why there is more number of “0” that number of “1” in the matrix [6].A code LDPC can be represented in matrix and graphic form called bipartite graph (or Tanner). Thus We have:

$$\mathbf{H} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{pmatrix}$$

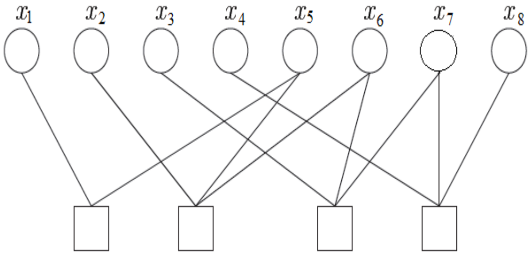

Figure 1: Bipartite graph of a code LDPC

In this graph the rows (check nodes) are represented by squares, the columns (variables nodes) by circles and the "1" by arcs.

There are two families of codes LDPC: regular codes and irregular codes.

Regular codes LDPC are the codes in which the number of "1" in each row and the number of "1" in each column are constant. By extension, irregular codes LDPC are the codes defined by matrices of parity check where the number of "1" in each row or column is not constant. The irregularity of these codes is specified through two polynomials λ(x) and ρ(x).

$$\lambda(x) = \sum_{i\geq 1} \lambda_i \, x^{i-1} \qquad (1)$$

$$\rho(x) = \sum_{i\geq 2} \rho_i \, x^{i-1} \qquad (2)$$

Where $\lambda_i$ (resp. $\rho_i$) the proportion of the number of branches connected to the variables nodes characterizes (resp. with the check nodes) of degree i compared to the full number of branch. The degree is defined like the number of branches connected to a node.

## 2.2 Encoding

Works of T.J. Richardson and R.L Urbanke [7] showed that the matrix of control must undergo a pretreatment before the encoding operation. The goal of the pretreatment is to put the matrix H in a lower almost triangular form, as shown in the Figure 2, by using only permutations of the rows or columns. This matrix is made up of 6 hollow sub matrices, noted as A, B, C, D, E and of a matrix T under lower triangular of size m-g × m-g . Once completed the pretreatment of H, the principle of the encoding is based on the resolution of the system represented by the following matrix equation:

$$cH^T = 0 \qquad (3)$$

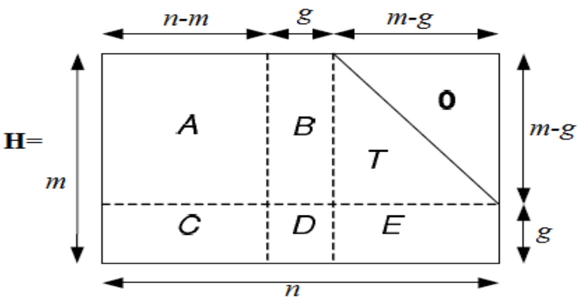

*Figure 2: Representation in pseudo form triangular lower of the matrix H*

The algorithm of pretreatment is described in a following way:

The preprocessing algorithm is described below succinctly [8]

*1* - [Triangulation] performs the permutations of rows or columns of an approximation of the matrix H as lower triangular.

$$H = \begin{pmatrix} A & B & T \\ C & D & E \end{pmatrix}$$

With a gap g small if possible. We will see in the following section how this can be effectively accomplished.

*2* - [Control of row] to use Gaussian elimination to carry out the pre-multiplication indeed.

$$\begin{pmatrix} I & 0 \\ -ET^{-1} & I \end{pmatrix} \begin{pmatrix} A & B & T \\ C & D & E \end{pmatrix}$$

$$= \begin{pmatrix} A & B & T \\ -ET^{-1}B + C & -ET^{-1}B + D & 0 \end{pmatrix}$$

In order to check that $-ET^{-1}B + D$ is non-singular and executant of other permutations of column to the need to ensure this property.

When solving the equation (3), the code word sought is divided into three parts: $c = (\mathbf{d}, \mathbf{r_1}, \mathbf{r_2})$ , where d is the systematic part (that is to say an element the canonical basis in the vector space of dimension n-m as shown in Figure 2) where the desired redundancy bits are separated into two vectors r1 and r2 of respective sizes g and m-g . After multiplication on the right by the matrix $\begin{pmatrix} I & 0 \\ -ET^{-1} & I \end{pmatrix}$, the equation (3) becomes:

$$A\mathbf{d}^T + B\mathbf{r_1}^T + T\mathbf{r_2}^T = 0 \qquad (4)$$

$$(-ET^{-1}A + C)\mathbf{d}^T + (-ET^{-1}B + D)\mathbf{r_1}^T = 0 \qquad (5)$$

The equation (5) makes it possible to find $\mathbf{r_1}^T$ while reversing $\Phi = -ET^{-1}B + D$. Then the equation (4) makes it possible to find $\mathbf{r_2}^T$. Many expensive operations in time can be made once for all in a pretreatment. All the operations repeated during the encoding have a complexity in $\mathrm{O(n)}$ except for the multiplication of $(-ET^{-1}A + C)\mathbf{d}^T$ by the square matrix $(-\Phi^{-1})$ of size g x g which after insertion is not hollow any more from where a complexity in $\mathrm{O(g^2)}$. T.J. Richardson and R.L Urbanke also shown that we can obtain a value of g equal to a weak fraction of n: $\mathrm{g} = \alpha n$ where $\alpha$ is a sufficiently low coefficient so that $\mathrm{O(g^2)} \ll \mathrm{O(n)}$ for values of n going until $10^5$.

Thus, the complexity of the approach of encoding complexity is O (n).

### 2.3 Decoding

The decoding of codes LDPC is carried out starting from the iterative algorithms of which more used is the algorithm of Belief Propagation (BP). At each iteration, there is exchange of messages between the variables nodes and the check nodes, on the same arc of the bipartite graph.

The algorithm of Belief Propagation consists in updating initially the variables nodes then the check nodes and finally to make a decision based . (see work of Jean-Baptiste Doré [8]) .

The update of the messages $\mathrm{m_{vc}}$ resulting from the variable node v with iteration i is calculated in the following way:

$$m_{vc}^{i} = v_0 + \sum_{c' \in C_v \backslash c} m_{c'v}^{i-1} \qquad (6)$$

Where $v_0$ represent the log likelihood ratio of probability resulting from the observation $y_v$ at output of the channel:

$$v_0 = \ln \frac{Pr\ (y_v | v = 0)}{Pr\ (y_v | v = 1)} \qquad (7)$$

And where $C_v$ represents the whole of the check nodes connected to the variable node *v*. With the first iteration, the message coming from the check nodes are null.

The update of the messages $m_{cv}$ resulting from the node of control *C* to iteration *i* is calculated in the following way:

$$m_{cv}^{i} = 2\tanh^{-1}\left( \prod_{v' \in C_c \backslash v} \tanh\left( \frac{m_{v'c}^{i-1}}{2} \right) \right) \qquad (8)$$

Where $C_c$ represents the whole of the variable nodes connected to the check node *c*.

## 3 PRINCIPLE OF THE SCHEME BASED ON CODES LDPC

### 3.1 Minimization embedding impact

When we want to hide a message by using a steganography scheme, we take $\mathrm{x} \in \mathrm{F}_2^n$ and this sequence is changed into a steganography image $\mathrm{y} \in \mathrm{F}_2^n$. And the message can be represented by $\mathrm{m} \in \mathrm{F}_2^m$ with $m < n$.

Indeed, to measure the similarity between the cover vector and the steganography image, we use the additive function of distortion which is defined by:

$$d(\mathrm{x},\mathrm{y}) = \|\mathrm{x} - \mathrm{y}\| = \sum_{i=1}^{n} \rho_i |\mathrm{x}_i - \mathrm{y}_i| \qquad (9)$$

Where $\rho_i$ or the not-negative number is the cost embedding of the pixels to be changed which belongs to the interval [0 1].

And the efficiency embedding is given by the following relation:

$$e(\mathrm{x},\mathrm{y}) = \frac{m}{d(\mathrm{x},\mathrm{y})} \qquad (10)$$

In the case of codes LDPC, the matrix of parity check **H** is used for encoding and even for decoding. For that, let us take $C(\mathbf{m}) = \{\mathbf{v} \in \mathrm{F}_2^n | \mathbf{Hv} = \mathbf{m}\}$ the coset[1] corresponding to the syndrome $\mathbf{m} \in \mathrm{F}_2^m$ (**m** is the secret message). The embedding and the extraction of the message can be given by:

$$\mathrm{y} = Emb(\mathrm{x},\mathrm{m}) \triangleq \arg \min_{\mathrm{u} \in C(\mathrm{m})} \|\mathrm{x} - \mathrm{v}\| \qquad (11)$$

$$Ext(\mathrm{m}) = \mathrm{Hy} = \mathrm{m} \qquad (12)$$

Let us take a vector u member of the coset such as.

$$\min_{\mathrm{v} \in C(\mathrm{m})} \|\mathrm{x} - \mathrm{v}\| = \min_{\mathrm{c} \in C} \|\mathrm{x} - \mathrm{u} - \mathrm{c}\| \qquad (13)$$

c is the code word which is determined by the equation (13). The sender must find the vector u in order to satisfy the equation (15).

[1] The whole of the words of code which have the same syndrome

To reduce the embedding impact to the minimum, the sender must find a vector there nearer to x.

A binary code $C$ used for the embedding of a message has a rate $R = \frac{n-m}{n} = 1 - \alpha$ and $\alpha = \frac{m}{n}$ is the payload. Consequently, we can write the higher limit of the rate of distortion by using the average embedding distortion [9]:

$$\alpha = 1 - R \leq H(d/n) \qquad (14)$$

Where $H$ is the entropy function which is defined by:

$$H(x) = -x\log_2 x - (1-x)\log_2(1-x)$$

According to the definition of the effectiveness of insertion, we can obtain the higher limit:

$$e \leq \frac{\alpha}{H^{-1}(\alpha)} \qquad (15)$$

### 3.2 The scheme proposed

T. Filler [5] proposed the use of LDGM codes to minimize the number of changes to support a host. T. Filler showed that the insertion is equivalent to the problem of quantifying binary. To solve this T. Filler used the BP algorithm applied to LDGM codes.
Our work is an extension of the work of T. Filler with, for our purposes, the use of LDPC codes. The use of LDPC codes overcomes the problem of complexity of implementation of the scheme T. Filler and its cumbersome implementation. In against part, the same way as T. Filler, our scheme is very close to the theoretical limit.

We will seek to minimize the number of pixels to change to insert a message into a cover image. We take as representative of the cover image the binary vector x consists of the LSB (Least to represent bits) of the pixels to insert the selected message.

The insertion of the message x is then performed as:
[Treatment of the matrix of control] Pre-treatment of the matrix parity check is performed as explained in Section 2.2. Note that the method proposed by T. Filler does not go through the pre-treatment.
[Calculation of a vector u] We calculate the vector u as a member of the coset:
$u= P^{T}.m$ (12)
where $P^{T}$ is the transpose of the matrix H 'obtained through preprocessing algorithm described in Section 2.2. This differs from the approach of Filler. In fact to determine u, we made a treatment at the generator matrix G LDGM code with the introduction of two matrices which assumes the existence Pr matrix that permutes the rows, the matrix switches Pc columns.
Filler led to the following equation:

$$(PrGPc)^{T}=G'^{T}=\begin{pmatrix} A & B & T \\ C & D & E \end{pmatrix}$$

Where $\begin{pmatrix} A & B & T \\ C & D & E \end{pmatrix}$ triangular matrix as closely as defined Section 2.1.
To make this work, he was inspired by the work of Richardson and Urbanke finally he asked:
$u = Pr^{-1}. (m, \underline{0})$ or $\underline{0}$ is a vector that it concatenates the message m for the matrix multiplication is possible.
We see that this approach is very tedious for the determination of the vector u, we propose a much simpler method with LDPC codes.
Treatment of the parity check matrix is made by Richardson and Urbanke without using matrices that allow either swap the rows or columns of the matrix.
3. [Calculation of the vector c] Calculate c = (d,r1, r2), with the systematic part of (that is to say an element of the canonical basis in the vector space of dimension n-m as shown in Figure 2) .The vectors r1 and r2 are determined as explained in Section 2.2.
4. [determination of the vector change r] We determine the vector r approaching x-u-c as described in equation (13) BP by running the algorithm that takes as input and returns the $x - u - c$ codeword r. Note that T. Filler also uses the BP algorithm with the same approach for the vector of change r. Once the vector r changes is obtained, we calculate the vector y =r + u.

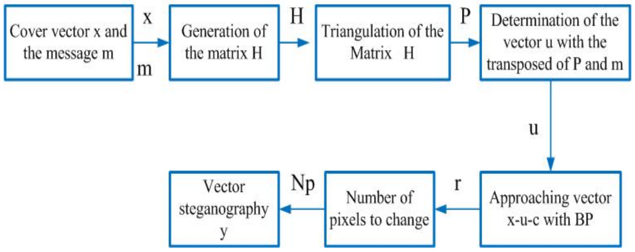
Cover vector x and the message m
x
m
Generation of the matrix H
H
Triangulation of the Matrix H
P
Determination of the vector u with the transposed of P and m
u
Approaching vector x-u-c with BP
r
Number of pixels to change
Np
Vector steganography y

*Figure 3: Diagram based on codes LDPC*

Figure 3 shows a representation of the different steps in the process of inserting a message into a vector coverage using the codes

## 4 PRESENTATION AND ANALYZES OF THE RESULTS

The construction of an LDPC code is to be filled with non-zero values the parity matrix. To optimize the construction of LDPC codes, we use the three-step approach of Claude Berrou(optimization profiles of irregularity, size optimization cycles, code selection by the pulse method).Figure 4 shows a representation of a parity check matrix of an LDPC code in accordance with these principles.

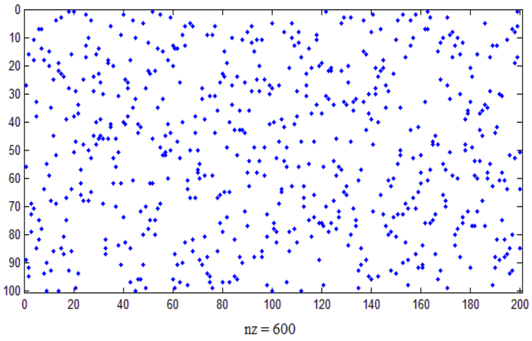

*Figure 4: Example of matrix of parity LDPC*

In Figure 4, the matrix is of size 100 x 200, that is to say 100 rows and 200 columns. The non-zero (nz = none zero in the figure**)** is represented by "point blue" and is in number600. The rate is $\tau$ = 600 / ((100 x 200)) = 0.03.For this example, 3% of the matrix elements are nonzero.

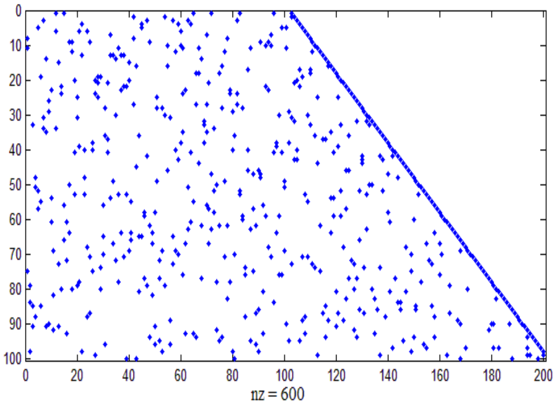

*Figure 5: Representation form almost triangular matrix of parity*

Figure 5 shows a representation of the parity check matrix of the output of the preprocessing algorithm described in Section 2.2.

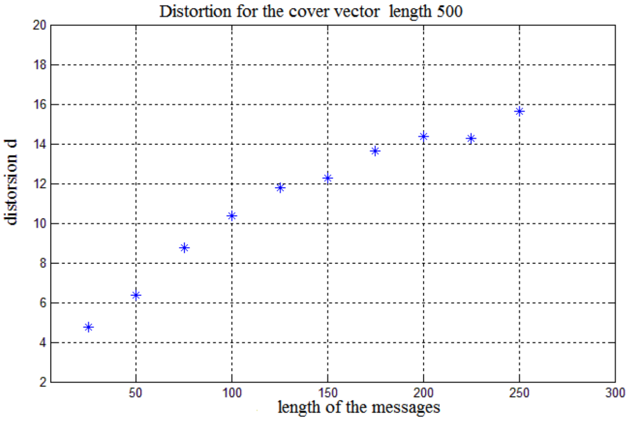

*Figure **6**: Distortion for the various messages hidden in a cover vector length 500*

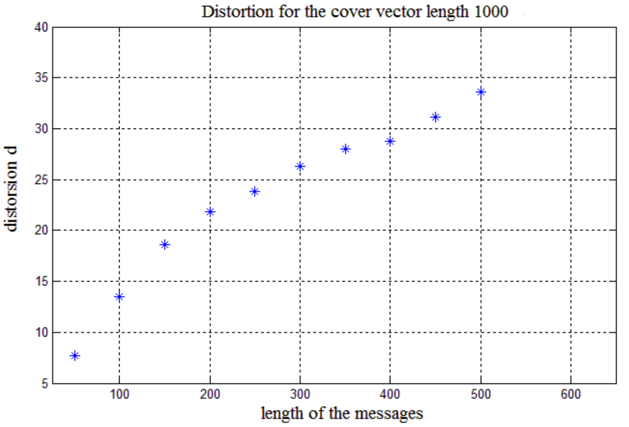

*Figure **7**: Distortion for the various messages hidden in a cover vector length 1000*

Figure 7: Distortion for different messages inserted into a vector of length 1000

Figures 6 and 7 give a representation of the different values of the distortion (number of pixels changed) in the vectors of length 500 bits cover bits and 1000 respectively according to the length of the message. Messages and vectors of coverage are randomly generated. For the vector of 500 bits, messages inserted have sizes ranging from 25 to 250. The upper bound is equal to 250 bits because the message size must satisfy the following inequality: $2m \leq n$ with m and n the respective sizes of the message and the vector coverage. For the vector of 1000 bits, messages inserted have sizes ranging from 50 to 500. Note that the vectors of coverage can be considered as the LSB of the pixels of a cover image.We note that the distortion is almost linear in the size of the message

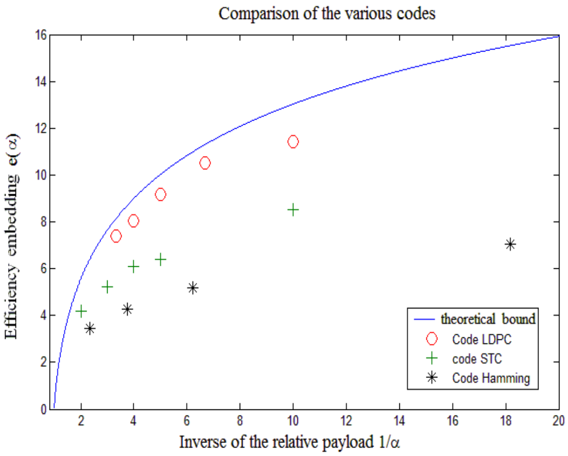

*Figure 8: Comparison of performance between the codes The theoretical upper limit of the efficiency of insertion [9]*

$$e \leq \frac{\alpha}{H^{-1}(\alpha)},$$

With $\alpha = \frac{m}{n}$ the payload H is the entropy function is defined by:

$$H(x) = -x\log_2 x - (1-x)\log_2(1-x)$$

Figure 8 compares the performance of insertion of LDPC codes, Hamming and STC and the route of the theoretical efficiency limit of integration. The values of the efficiency of insertion for the LDPC codes are obtained by inserting 10 messages in a vector of size n =1000. The performance figures for the insertion code STC are obtained by running the source code of Tomáš Filler and Jessica Fridrich [9])(with an adjustment cost of detectability constant). The efficiency values of Hamming codes are determined by the formula $e = \frac{p}{1-2^{-p}}$ with p an integer. Figure 8 confirms the hypothesis that Jessica Fridrich codes approach very sparse matrices prepared the theoretical limit of efficiency of insertion. LDPC codes are good candidates for steganography by minimizing the number of pixels changed

## 5. CONCLUSION

In this paper, we reiterated the principle of operation of the LDPC codes. Then we presented an approach of "embedding matrix" based on the approach of T. Filler, but which complexity is greatly reduced by a pre-treatment of control matrix. The results confirm that the binary LDPC codes are used to insert a message by minimizing the number of pixels changed. In comparison with the binary codes STC [10](with use of a constant cost of detectability), LDPC codes are more efficient because they come more to the theoretical limit of efficiency of insertion.

Our future research directed towards the use of photographers codes [11]from the LDPC codes that have a much more hollow configuration. In addition, we also plan to study non-binary LDPC codes. Finally, as noted in the introduction, current patterns of steganography safest take into account the detectability of each pixel during insertion. It is therefore necessary to include this map of detectability in the creation of new code (the code STC is an example of code taking into account the map of detectability) [12]

## REFERENCES:

[1] Jessica Fridrich, *Steganography In digital media principles, Algorithms, and Application,* Binghamton University, State University of New York, Cambridge University Press, 2010.

[2] Rongyue Zhang, Vasiliy Sachnev, Hyoung Joong Kim, *Fast BCH syndrome coding for steganography* ; S. Katzenbeisser and A.-R. Sadeghi(Eds.), IH 2009, LNCS 5806, pp. 44-58, Springer-Verlag Berlin Heiderbelg 2009.

[3] Vasiliy Sachnev, Hyoung Joong Kim, Rongyue Zhang, *Less detectable JPEG steganography method based on heuristic optimization and BCH syndrome coding*; MM& Sec '09, Princeton, New Jersey, USA, Septembre 2009.

[4] F. Galand C. Fontaine. *How can Reed-Solomon codes improve steganographic schemes. In Information Hidding*, Rennes, France, 2009.

[5] Tomas Filler, *Minimizing Embedding Impact in Steganography Using Low Density Codes*, Thesis, Department of Electrical and Computer Engineering, SUNY Binghamton, USA, 2006/2007.

[6] Claude BERROU, *Codes et turbocodes* ; 1e édition, Springer - Verlag, France, 2007

[7] T.J. Richardson and R.L Urbanke, « Efficient Encoding of Low-DensityParity-Check Codes », *IEEE Trans. Inform. Theory*, vol. 47, pp. 638-656, February 2001.

[8] Jean-Baptiste Doré, « Optimisation conjointe de codes LDPC et de leurs architectures de décodage et mise en œuvre sur FPGA », Thèse pour obtenir le grade de Docteur à l'INSA de Rennes, Spécialité : Electronique, Soutenue le 26 Octobre 2007.

[9] Tomáš Filler, Jan Judasand Jessica Fridrich*, Minimizing Additive Distortion in steganography Using Syndrome-Trellis Codes*, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 6, NO. 3, SEPTEMBER 2011.

[10] Tomáš Filler, Jan Judas et Jessica Fridrich, *Minimizing Embedding Impact in Steganography using Trellis-Coded Quantization*, Department of Electrical and Computer Engineering SUNY Binghamton, Binghamton, NY 13902-6000, USA 2010.

[11] TODD K MOON, *ERROR correction CODING Mathematical Methods and Algorithms,* Utah State University, Copyright ©2005 by John Wiley &Sons, Inc.

[12] Tomáš Pevnỳ, Tomáš Filler and Patrick Bas, *Using High-Dimensional Image Models to Perform Highly Undetectable Steganography*, Czech Technical University in Prague, Czech Republic; State University of New York in Binghamton, NY, USA; CNRS-LAGIS, Lille, France, 2010.