ABSTRACT: Steganography is the art of secret communication. Since the advent of modern steganography, in the 2000s, many approaches based on error correcting codes (Hamming, BCH, RS, STC ...) have been proposed to reduce the number of changes in the roof while inserting the maximum bit.

In this paper we propose a new steganography scheme based on the polar codes. The scheme works according to two steps. The first offers a stego vector from given cover vector and message.

1 INTRODUCTION

The steganography is a technique allowing hiding an information in a medium (image, sound or video) unsuspected so that it was undetectable. To reach this objective it is indispensable to use a technique in order to reduce the distortion induced by the hiding of the secret message. The matrix embedding technique introduced by Crandall [1] has allowed the definition of steganography schemes that minimize the embedding impact. The first implementation was created with the work of Westfeld [2] in which the Hamming code has been used. Afterwards the BCH codes [3, 4], the Reed-Solomon codes [5] and the STC codes [6] are used in steganography. The combination of the techniques of LSB, of matrix embedding and wet paper has allowed realizing more effective and more reliable steganography schemes. Our works is a contribution of schemes of minimization of embedding impact.

This paper is organized as following. Section 1 describes the concepts of matrix embedding and minimization of embedding impact. In Section 3 we study the linear programming. The polar codes used for the implementation of our scheme are presented in Section 4. In Section 5 we propose our scheme based on the polar codes. Section 6 show the results obtained when the scheme is applied on images. Explications of these results are also given in this Section. Section 7 concludes the paper.

2 STEGANOGRAPHY AND MATRIX EMBEDDING

2.1 Steganography

Steganography or the art of secret communication aims to hide a message in an apparently innocuous cover medium.

Steganography schemes are characterized by different parameters. The insertion capacity represents the maximum number of bits that can be inserted in a cover medium. The rate is the number of bits of the message by inserted support element and the change density defines the proportion of modified components of the cover. The embedding efficiency is the number of bits of the message by distortion unit. This is the ratio of the rate by the density change. This last characteristic is used to evaluate the performance of a steganography scheme. We say that a steganography scheme is even better than its insertion efficiency is great.

2.2 Distortion measure with the PSNR

The PSNR (Peak Signal Noise Ratio) is a distortion measure between two images. It is calculate from MSE (Mean Square Error) and is expressed in dB. Let $I_o$ and $I_r$ be respectively the images original and reconstructed images of same length $M \times N$.

The PSNR and the MSE are given by:

$$PSNR(I_o, I_r) = 10 \log_{10} \frac{D^2}{MSE(I_o, I_r)}$$

$$MSE(I_o, I_r) = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} (I_o(i,j) - I_r(i,j))^2$$

where $D$ is the dynamic (the maximum value of a pixel). If the pixels are coded with 8 bits $D = 2^8 - 1 = 255$. 

KEYWORDS: STEGANOGRAPHY, MATRIX EMBEDDING, POLAR CODES, LINEAR PROGRAMMING, WET PAPER CODES
More the value of PSNR is greater; more the images compared are similar. A PSNR of more than 35 dB between two images means that there is no visible difference between these two images [7]. If the PSNR is less than 20 dB the two images are very different.

2.3 The principle of matrix embedding

Consider the cover vector \( v \) consists of the LSBs of the cover image, the stego vector \( y \), the vector of changes \( e \) \((y = v + e)\), the secret message \( m \) and the parity check matrix \( H \) correcting code errors used. The principle of matrix embedding is to find the stego vector \( y \) closest to \( v \) such that \( yH^T = m \). By replacing \( y \) by \( v + e \) we will have \( vH^T + eH^T = m \iff eH^T = m - vH^T \).

The objective of the sender is to find the vector \( e \) of minimum weight in the coset \( C(m - vH^T) \) (the set of the vectors of size \( n \) and syndrome \( m - vH^T \)) and then add it with \( v \) to find \( y \). At the reception, to find \( m \), the decoding is just done by the matrix product \( m = yH^T \).

2.4 Minimization of embedding impact

We still consider the vectors defined above. Assuming that the changes do not interact with each other, the total embedding impact is the sum of the embedding impact at each pixel [6]:

\[
D(v, y) = \sum_{i=1}^{n} \rho_i |v_i - y_i|, \tag{3}
\]

with \( 0 \leq \rho_i \leq \infty \) the cost the change of the pixel \( v_i \) into \( y_i \). The goal is for the sender to insert its binary message \( m \in \{0, 1\}^m \) so that the distortion \( D \) is minimized.

The functions of insertion and extraction are defined by:

\[
\text{Emb}(v, m) = \arg \min_{y \in \mathcal{C}(m)} D(v, y) \tag{4}
\]

\[
\text{Ext}(y) = yH = m \tag{5}
\]

where \( H \in \{0, 1\}^{(n-k)\times n} \) is a parity check matrix of the code \( \mathcal{C}(n, k) \) and \( \mathcal{C}(m) = \{x \in \{0, 1\}^n \mid xH^T = m\} \) is the coset corresponding to the syndrome \( m \).

3 LINEAR PROGRAMMING

The linear programming is a central domain of optimization. An optimization problem highlights variables, constraints on these variables and a criterion to optimize. It can be formulated as follows:

\[
\min_{x \in \mathcal{C}} \max_{s.t.} f(x)
\]

with s.t : subject to, \( f \) the criterion to optimize (objective function), \( x \) the variable and \( \mathcal{C} \) the set of constraints (feasible set).

A linear program can be written either in the canonical form or in the standard form (obtained from the canonical form).

| Canonical form | Standard form |
|----------------|---------------|
| \( \min f_{s.t}(x) = e^T x \) | \( \min f_{s.t}(x) = c^T x \) |
| \( \begin{cases} A x \geq b \\ x \geq 0 \end{cases} \) | \( \begin{cases} A'x' = b \\ x' \geq 0 \end{cases} \) |

Before solving a linear programming problem, we must begin by putting it in standard form with the introduction of discard variables that allow setting the expression of constraints in the form of a linear equations system. The solving of a linear program can be done by using the simplex method or methods of interior points.

3.1 Simplex Method

This method was developed in the late 40s by G. Danzig and solves linear programs. To avoid calculating the solutions of all linear systems \( m \times m \) extracts from \( A \), we may use the simplex algorithm. This algorithm is based on the following approach presented in [8]: starting from a vertex representing the initial solution, we traverse the whole of the vertices of the set of feasible solutions \( S \) (a polyhedron) by determining if the current vertex is optimal and if not the case, we move to adjacent vertex that optimizes the objective function. Starting of a vertex representing the initial solution, we move from extreme point (vertex) to extreme point along the frontier of the polyhedron and since the number of extreme points is finite, the algorithm is called combinatorial.

3.2 Methods of interiors points

The 1984 publication of the work of Karmarkar [9] gave rise to interior point methods which are intended to reduce the complexity observed in the simplex algorithm. The interior point methods start from an interior point (initial solution) to the domain of feasible solutions, then using a fixed strategy determines an approximate value of the optimal solution [10]. The movement is made along the direction that gives the best qualifying improvement of the objective function. In general, the direction is inside the polyhedron and the method is called "nonlinear". The
advantages of these methods compared to the simplex method are robustness, polynomial complexity and fast convergence to the real problems of large sizes.

Clearly, a method of solving a linear program is even faster than the initial solution is close to the optimal value sought. We will see in Section 7 that our optimization problem is particular.

4 THE POLARS CODES

4.1 Usual notations

Let $W: X \rightarrow Y$ the channel B-DMC (Binary input-Discrete Memoryless Channel) on which the transmission takes place. The sets $X$ and $Y$ respectively represent the input and output alphabets of the channel $W$. $W(y|x)$ is the transition probability such that $x \in X$ and $y \in Y$. The channel $W^n$ correspond to $n$ uses of $W$. The operation $\oplus$ defines the modulo-2 addition and $\otimes$ Kronecker product. $a^n_i$ denote the line vector $(a_1, ..., a_n)$ where $n = 2^p$, $p$ being a positive integer. $a^n_{i,o}$ is the subvector off indices $(a_{2i-1})_{1 \leq i \leq n/2}$ and $a^n_{i,e}$ those of even indices $(a_{2i})_{1 \leq i \leq n/2}$.

4.2 Definitions

The symmetric capacity (bits/s) of the channel B-DMC $W$ is defined as following [11]:

$$ I(W) = \sum_{y \in Y} \sum_{x \in X} \frac{1}{2} W(y|x) \log_2 \frac{W(y|x)}{\frac{1}{2} W(y|0) + \frac{1}{2} W(y|1)} $$

(6)

and Bhattacharyya parameter or reliability of the channel $W$ is given by:

$$ Z(W) = \sum_{y \in Y} \sqrt{W(y|0)W(y|1)} $$

(7)

The polar coding is based on these two parameters.

Two types of usual symmetrical channels are Binary Erasure Channel (BEC) and Binary Symmetric Channel (BSC). A B-DMC channel is a BSC if $Y = \{0,1\}$, $W(0|0) = W(1|1)$ and $W(1|0) = W(0|1)$ and a BSC if $W(y|0)W(y|1) = 0$ or $W(y|0) = W(y|1)$.

For any B-DMC $W$, we have [11]:

$$ \log_2 \frac{1}{1 + Z(W)} \leq I(W) \leq \sqrt{1 - Z(W)^2} $$

(8)

$$ I(W)^2 + Z(W)^2 \leq 1 \leq I(W) + Z(W) $$

(9)

The parameters $I(W)$ and $Z(W)$ take their values in $[0,1]$ and more verify the following equivalences:

$I(W) \approx 1$ is equivalent to $Z(W) \approx 0$ and (perfect canal)

$I(W) \approx 0$ is equivalent to $Z(W) \approx 1$ (completely noisy)

From these two equivalences we can say that, to know the properties of a B-DMC channel, it suffices to study one of the two parameters. The larger $I(W)$ is the better is the channel and vice versa. On the other hand the channel with the smallest value of $Z(W)$ is the most reliable. We will focus particularly on $Z(W)$ more easy to manipulate.

4.3 Channel polarization and transformation

4.3.1 Channel polarization

Polarization constitutes the base of the construction of polar codes. It consists in synthesizing of $n$ independent copies of a given B-DMC $W$ and other channels to create others $(W^n_i)_{1 \leq i \leq n}$. The polarization appears in the sense that $I(W^n_i)$ tends to 0 or 1 depending on whether $I(W^n_i)$ is closer to 0 or 1. The operation of channel polarization is constituted of two step: the channels combination and channel splitting.

4.3.1.1 The channels combination

This is to group $n$ copies of a given B-DMC $W$ channel in a given channel $W^n$. The combination for the level $p = 1$ associated $n = 2^p$ independent copies of $W_1 = W$ to form the channel $W_2: X^2 \rightarrow Y^2$.

![Figure 1: Construction of the channel $W_2$.](image)

The following relations are established from Figure 1: $x_1 = u_1 \Theta u_2$ and $x_2 = u_3$. Thus $x_1^n$ and $u_1^n$ are linked by the relation $x_1^n = u_1^n G_2$ with $G_2 = [\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}]$.

The generalization of the channels combination procedure with any $p \geq 1$ ($n = 2^p$) is given by a combination of two independent copies of the channel $W_{n/2}$ to form $W_n$. The first step is the passage from the input $u_1^n$ of $W_n$ to $s_1^n$ such that $s_{2i-1} = u_{2i-1} \Theta u_{2i}$ and $s_{2i} = u_{2i}$ for $1 \leq i \leq n/2$. The permutation matrix $R_n$ transforms $s_1^n$ into $v_1^n = (s_{1,o}^n, s_{1,e}^n)$, the input for the two copies of $W_{n/2}$.
Figure 2: Construction of the channel $W_n$ from two copies of $W_{n/2}$.

The relation of polar coding is 

$$x_1^n = u_1^n G_n$$  \hspace{1cm} (10)

The matrix $G_n = B_n G_2^\otimes p$ is called generator matrix, $B_n$ a permutation matrix and $G_2^\otimes p$ the product of Kronecker of $p$ copies of a matrix $G_2$.

4.3.1.2 The channel splitting

After the channels combination the next step of the channel polarization is to subdivide the channel $W_n$ into $n$ channel $W_n^{(i)} : \mathcal{X} \rightarrow \mathcal{Y}^n \times \mathcal{X}^{i-1}$ defined by the following transition probabilities:

$$W_n^{(i)}(y_1^n, u_1^{i-1} | u_i) = \sum_{u_{i+1} \in \mathcal{X}^{n-i}} \frac{1}{2^{n-i}} W_n(y_1^n | u_1^{i-1})$$

If $u_1^n$ is uniform on $\mathcal{X}^n$ then $W_n^{(i)}$ is the channel really seen by $u_i$ : all happens as if each input bit $u_i$ borrows the channel $W_n^{(i)}$ to give $(y_1^n, u_1^{i-1})$ as shown in Figure 3.

4.3.2 The recursive channel transformation

The process of transformation can be generalized recursively:

$$(W_n^{(i)}, W_n^{(i)}) \rightarrow \text{construct} (W_{2n}^{(2i-1)}, W_{2n}^{(2i)})$$

The recurrence relations used for the construction of the polar codes are \cite{11}:

$$Z(W_{2n}^{(2i-1)}) \leq 2Z(W_{2n}^{(i)}) + Z(W_n^{(i)}), \hspace{1cm} (11)$$
$$Z(W_{2n}^{(2i)}) = Z(W_n^{(i)})^2. \hspace{1cm} (12)$$

with equality if $W$ is a BEC.

These two relations will be used for the construction of polar codes in Section 6.

4.4 The polar coding

The principle of polar coding is to create a system of coding allowing to access to each channel $W_n^{(i)}$ individually and send the data through those more reliable that is to say those for which $Z(W_n^{(i)})$ is more near to 0.

Consider a given subset $A$ of dimension $k$ and his et son complementary $A^c$ in $\{1, \ldots, n\}$. We will fix $A$ ($A^c$ is also fixed) and $u_A^c$ by letting $u_A$ variable. The vector $u_A$ is called information vector and $u_A^c$ frozen vector (its bits are fixed). In general we choose $u_A^c = 0_1^{n-k}$ because the choice of $u_A^c$ don’t affect the performances of a symmetric channel \cite{11}. The set $A$ is chosen such that $Z(W_n^{(i)}) \leq Z(W_n^{(j)})$ for any $i \in A$ and $j \in A^c$.

4.4.1 The construction of polar codes

To construct a polar code we need as inputs the channel B-DMC $W$ on which the code is applied, the block length $n$ and dimension $k$ of the code. The algorithm of construction provide as output the information set $A \subset \{1, \ldots, n\}$ of size $k$ such that the value of $\sum_{i \in A} Z(W_n^{(i)})$ is minimal.

4.4.2 Polar encoding

We use the relation (10) for encode a data word $u_1^n$ in a codeword $x_1^n$. An interpretation of the various operations of permutation which consists $G_n$ gives us the following expression \cite{11}:

$$G_n = R_n(G_2 \otimes G_{n/2}) = B_n G_2^\otimes p \hspace{1cm} (13)$$

with $B_n$ a permutation matrix defined by:

$$B_n = R_n(I_2 \otimes R_{n/2})(I_4 \otimes R_{n/4}) \cdots (I_{n/2} \otimes R_2)$$
\[ R_n \left( I_2 \otimes B_{n/2} \right) \]  

From these two relations we try to define more explicitly the permutation matrices \( B_n \) and \( R_n \). For this we will use the indicial notation of the vectors.

Let \( a_i^n \) be a vector then the element \( a_i \) is noted by \( a_{b_1b_2 \cdots b_p} \) where \( b_1b_2 \cdots b_p \) correspond to the binary representation of \( i - 1 \).

- The matrix \( R_n \) acts on a vector by performing a cyclic shift of index bits to the left. For example if \( v_1^n = u_1^n R_n \) then \( v_{b_1b_2 \cdots b_p} = u_{b_2 \cdots b_p b_1} \).
- The permutation matrix \( B_n \), consisting of \( p \) shift matrices, performs \( p \) cyclic shifts to the left in accordance with the following process: for the \( i \)th shift we consider the first \( p - i + 1 \) bits from the left to the right. That is to reverse the order of index bits. If \( v_1^n = u_1^n B_n \) we have \( v_{b_1b_2 \cdots b_p} = u_{b_p \cdots b_2 b_1} \).

We will use in our practical part (Section 5) recursion formulas (11) and (12) to find the permutation matrix \( B_n \) and the generator matrix \( G_n \) of the polar code.

### 4.4.3 The decoding of polar codes

Several types of decoding of polar codes exist namely Successive Cancellation (SC) [11], the Linear Programming (LP) decoding [12] and Belief Propagation (BP) decoding, etc. Among these types of decoding the most powerful is the SC but because of the probability of the channels involved in its implementation its application in steganography is not yet possible. Therefore we use the LP decoding for the formulation of our steganographic scheme.

### 5 STEGANOGRAPHIC SCHEMA BASED ON POLAR CODES

Before defining the scheme we describe first the construction of polar codes in the steganographic context.

#### 5.1 The different steps of the construction of polar codes for the steganography

The construction of the polar codes for the steganography can be summed up in true steps as shown by Figure 4.

![Figure 4: The scheme of the construction of a polar code.](image)

**Step 1**: the calculation of reliability of the channels.

We saw in the previous section that the choice of sets \( A \) and \( A^c \) depends on the values of the parameters of reliability \( Z(W_n^{(i)}) \). For a symmetric channel we have relations (11) and (12), with equality in (11) for the case of a BEC channel. Steganographic channel BSC. Consider that our channel \( W \) is chosen such that equality in (11) is satisfied. Thus we can calculate all reliability parameters recursively. Relations (11) and (12) become:

\[
Z(W_n^{(i)}) = 2Z \left( W_n^{(i+1)/2} \right) - Z \left( W_n^{(i+1)/2} \right)^2 \quad \text{if } j \text{ is even}
\]

\[
Z(W_n^{(j)}) = Z \left( W_n^{(j+1)/2} \right)^2 \quad \text{if } j \text{ is odd}
\]

The initial value is given by:

\[
Z(W_n^{(1)}) = Z(W) = \sum_{y \in \{0,1\}} \sqrt{W(y|0)W(y|1)} = 2\sqrt{p_e(1-p_e)}
\]

where \( p_e \) is the error probability of the channel \( W \), \( p_e = W(0|1) = W(1|0) \) et \( 1 - p_e = W(0|0) = W(1|1) \).

**Step 2**: determination of the sets of information bits and redundancy bits.

We select channels with the parameters of the lowest reliabilities (the most reliable channels) for data bits. The indices of all these channels form information bits \( A \). Its cardinality is equal to the dimension \( k \) of the code considered polar and the other channels (there are \( n-k \)) carry redundancy bits. Their indices constitute \( A^c \).

**Step 3**: generation of the parity check matrix \( H \).

A common element in the construction of any steganographic scheme is the parity check matrix \( H \). Indeed it is used both for insertion to extraction of messages in a cover object. To determine the parity check matrix of a polar code we can use the lemma given by Korada and al. in [12, Lemma 1] which says that if the bits of redundancy vector \( u_{A^c} \) are set to 0 then the parity check matrix is given by the columns of the matrix generator \( G_n \) of the polar code whose indices are in \( A^c \).

A polar code of length \( n = 2^p \) and dimension \( k \) has a generator matrix of size \((n,n)\), \( G_n = B_n \otimes_{2^p} \). For example, if \( p = 2 \) we have:

\[
G_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}, \quad G_2^3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}
\]

and \( G_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \)
where $B_4$ is the matrix that permute lines 2 and 3 of $G_2^{\otimes 3}$.

The calculation of the reliability parameters give

$$Z(W_4^{(1)}) = 0.9744, \quad Z(W_4^{(2)}) = 0.7056,$$

$$Z(W_4^{(3)}) = 0.5904 \quad \text{and} \quad Z(W_4^{(4)}) = 0.1296.$$

$$Z(W_4^{(1)}) \geq Z(W_4^{(2)}) \geq Z(W_4^{(3)}) \geq Z(W_4^{(4)}).$$

If we choose $k = 1$, one canal will be used for the information then $A = \{4\}$ and $A^c = \{1, 2, 3\}$. The parity check matrix will be defined as following:

$$H^T = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad \text{and} \quad H = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}.$$

5.2 Steganography scheme with a polar code

In this party of the paper we consider that the change of any pixel produces the same distortion (constant profile). Thus we are interested in the number of bits changed during insertion. The research of the stego vector will follow two phases.

5.2.1 First method proposed

We will draw our inspiration from the work of Fridrich and al. [6] in which the proposed steganography scheme is strongly related to the form of the parity check matrix of the STC code used. Indeed, by observing the parity check matrix $H$ of polar code and its transpose $H^T$, we can make the following remarks:

1) the columns of $H$ are pairwise independent;

2) if we scan the columns of $H^T$, the position at which it meets the first non-zero coefficient (equal to 1) differs from that of other columns of $H^T$;

3) more starting from the last column and starting with the first line, the position at which we meets the first non-zero coefficient is the first met on this line; so, the last position equal to 1 on this line if one starts from the left;

With these remarks, we will define a first steganographic scheme.

Consider the matrix product $yH^T = m$. The decomposition of this system gives us the following equations:

$$y_i = y_{i+1}H^T_{i+1,j} + \cdots + y_nH^T_{nj} + m_j, \quad \text{for} \quad j = 1, \cdots, n.$$

Let $i$ be the position of the first 1 met on the column $j$, the system of equations above becomes:

$$y_i = y_{i+1}H^T_{i+1,j} + \cdots + y_nH^T_{nj} + m_j \quad (15)$$

To determine $y_i$ in the above equation we must first find $y_{i+1}$ such that $H^T_{i+1,j} = 1$. We will assume that these positions correspond to the locked positions of $v$. In this case $v_{i+1} = y_{i+1}$. Therefore, before calculating the elements $y_i$ of the initial stego vector $y$, we first assign it as initial value the cover vector $y \leftarrow v$. The changes of certain positions $y_i$ of $y$ will occur as and when we travel over the columns.

At the end of this process, we have a stego vector $y$ arising from $d$ modifications of the cover vector $v$.

For a clearer explanation we will use an example. Consider the cover vector $v = (v_1, \cdots, v_8)$ and the message $m = (m_1, \cdots, m_8)$ to produce the stego vector $y = (y_1, \cdots, y_8)$. To apply the method described above we use a polar code of length $n = 2^p = 8$ ($p = 3$) and dimension $k = 4$. The set of information bits $A = \{4, 6, 7, 8\}$ and the set of the frozen bits $A^c = \{1, 2, 3, 5\}$. The parity check matrix is:

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

and $H^T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$. (16)

With the relation (15) we have the following system:

$$\begin{cases} y_2 = y_4 + y_6 + y_8 + m_4 \\ y_3 = y_4 + y_7 + y_8 + m_3 \\ y_5 = y_6 + y_7 + y_8 + m_2 \\ y_1 = y_2 + y_3 + y_4 + y_5 + y_6 + y_7 + y_8 + m_1 \end{cases}$$

The vector $y$ must be initialized to the cover vector $v$. Consider the first equation; the calculation of the coefficient $y_2$ requires knowledge of the coefficients $y_4$, $y_6$, and $y_8$, $a$. We consider these positions as fixed so $y_4 = v_4$, $y_6 = v_6$ and $y_8 = v_8$. In the same we fix the coefficient $y_7 = v_7$ to calculate $y_3$ by using $y_4$ and $y_8$ that have been fixed at the previous step. The calculation of is done using $y_6$, $y_7$ and $y_8$ previously fixed. The last coefficient $y_1$ is determined with coefficients either already fixed or already calculated. Hence the necessity to start from the last column of $H^T$ otherwise we could not calculate $y_1$ without fixing all the other coefficients $y_i$, $2 \leq i \leq 8$; that would be absurd.

In this example, for a vector cover $v = (0, 1, 1, 0, 1, 0, 1, 0)$ and a message $m = (1, 0, 0, 1)$ we
have the following stego vector \( y = (1, 1, 1, 0, 1, 0, 1, 0) \) and the corresponding error vector \( e = (1, 0, 0, 0, 0, 0, 0, 0) \). We have inserted a message of four bits to just modify one of the cover edit one. Hence the embedding efficiency is 4.

Note that the scheme described above provides an insertion rate of up to 100% (we can insert a number of bits equal to the size of the medium) but the density change is also great.

The application of the method described above gives us a solution satisfying \( yH^T = m \) but it is not necessarily the best. To test the optimality of the solution obtained, we compare the number of changes with the value \((n - k)/2\). If it is less than \((n - k)/2\) then the solution found with the first method is optimal otherwise it is not necessary optimal. In order to ensure to find the optimal solution, i.e. that which provides the stego vector closest to the cover vector \( v \) satisfying \( yH^T = m \), we will define, from the solution already obtained, a method that offers the optimal solution.

5.2.2 The second method

The objective with this method is to find the vector cover \( y \) closest to \( v \) that we can have by using the polar code \( C(n, k) \). Let \( y_p \) be the stego the vector found by applying the first approach \( (y_p = v + e_p) \). The question is how to find, from \( y_p \), the optimal vector \( y_{opt} \) verifying \( yH^T = m \). So we have to create an algorithm that, initialized to \( y_p \), converges to \( y_{opt} \). In other words, considering the error vectors, the algorithm should, from \( e_p \), provide the error vector corresponding to \( e_{opt} \). Thus we look out the error vector \( e \) of minimum weight satisfying \( eH^T = m - vH^T = s \).

Recapitulation:
- we have a starting solution \( e_p \) \( \iff \) initial solution,
- we search a vector \( e \) of minimal weight \( \iff \) problem of minimization,
- verifying \( eH^T = m - vH^T = s \) \( \iff \) constraints.

Considering these three points we have an optimization problem, particularly a minimization problem under constraints of equality with \( e_p \) as initial solution. Our optimization problem is defined for binary vectors and without inequality constraints. As we seek to minimize the weight of the vector \( e \), so to define the scalar product of the objective function we consider the vector unit cost \( (c = \{1\}) \). Find the vector \( e \) of minimum weight amounts to finding the vector realizing the minimum of the scalar product with the vector \( c \). This gives us as the objective function \( f(e) = \langle c, e \rangle \), it is the function to minimize. From there we can define our optimization problem as follows:

\[
\arg\min_{e} f(e) = \langle c, e \rangle = c^T e
\]

\[
\begin{cases}
 e \in \{0, 1\}^n \text{ a binary vector} \\
 eH^T = m - vH^T = s \\
 e_p \text{ initial solution } \iff e_pH^T = s
\end{cases}
\]

This problem is that of a linear optimization with constraints equalities written in standard form with an additional constraint that the vector \( e \) is constituted of binary elements. It can be solved by different methods for solving linear programming problems such as the simplex and the interior points methods explained in Section 3.

The decoding we use is similar, in the formulation of the problem, which of linear programs known but differs in the procedure of finding the solution. Indeed the LP decoding [12] uses error probabilities of the transmission channel in the formulation of the linear program. This is not exploitable in this context because in our steganography channel all bits of the cover vector are may be modified with equal probability. This is why we have not used the successive cancellations (S.C) decoding.

The application of the optimization method provides the optimal solution of our steganography problem. In order to see more clearly consider two examples.

- Let \( m = (0, 0, 1, 0) \) be the message to hide in the cover vector \( v = (0, 1, 0, 1, 0, 0, 0, 1) \). We use a polar code of length \( n = 8 \) and dimension \( k = 4 \) and the parity check matrix is that given in (16).

\[
\rightarrow \text{The first solution gives an error vector } e_p = (1, 1, 0, 0, 1, 0, 0, 0) \text{ and the corresponding stego vector } y_p = (1, 0, 0, 1, 1, 0, 0, 1).
\]

\[
\rightarrow \text{The optimization of the solution given by the above method gives the following results: the error vector } e_{opt} = (0, 0, 0, 0, 0, 1, 0, 0) \text{ and the optimal stego vector } y_{opt} = (0, 1, 0, 1, 0, 1, 0, 1).
\]

The first method provides an embedding efficiency \( eft_{inst} = (n - k)/d = 4/3 \) whereas the second offers an embedding efficiency \( eft_{inst} = 4/1 = 4 \).

- Consider \( v = (1, 0, 1, 1, 1, 0, 0, 1) \) and \( m = (0, 0, 1, 1) \).

The results of the first approach follow:

\[
\rightarrow \text{the error vector } e_p = (0, 1, 0, 0, 0, 0, 0, 0),
\]
the stego vector \( y_p = (1, 1, 1, 1, 1, 0, 1) \).

The optimization provides:
- error vector \( e_{opt} = (0, 1, 0, 0, 0, 0, 0, 0) \),
- optimal stego vector \( y_{opt} = (1, 1, 1, 1, 1, 0, 1) \).

The two methods produce the same solution with an embedding efficiency equal to 4.

The scheme can be broadly summarized by the following figure.

![Figure5: Representation of the proposed steganographic scheme.](image)

### 5.2.3 Calculation of embedding efficiency

We will calculate the embedding efficiency of our scheme for the case of a cover vector of size \( n = 8 \) and a message of \( n - k = 4 \) bits. Thus \( p = 3 \) and \( k = 4 \).

\[
H = \begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\
0 & 1 & 0 & 1 & 0 & 1 & 0 & 1
\end{bmatrix}
\]

By examining the columns of \( H_j \) and those of \( H \) combining them two to two we have the following equalities:

\[
\begin{align*}
H_1 + H_2 &= H_3 + H_4 = H_5 + H_6 = H_7 + H_8 = 0 \quad \text{and} \quad 0 \quad 0 \quad 1 \\
H_1 + H_3 &= H_2 + H_4 = H_5 + H_7 = H_6 + H_8 = 0 \quad \text{and} \quad 0 \quad 1 \quad 0 \\
H_1 + H_4 &= H_2 + H_3 = H_5 + H_6 = H_7 + H_8 = 0 \quad \text{and} \quad 0 \quad 1 \quad 1 \\
H_1 + H_5 &= H_2 + H_6 = H_3 + H_7 = H_4 + H_8 = 0 \quad \text{and} \quad 1 \quad 0 \quad 0 \\
H_1 + H_6 &= H_2 + H_5 = H_3 + H_8 = H_4 + H_7 = 0 \quad \text{and} \quad 1 \quad 0 \quad 1 \\
H_1 + H_7 &= H_2 + H_8 = H_3 + H_5 = H_4 + H_6 = 0 \quad \text{and} \quad 1 \quad 1 \quad 0 \\
H_1 + H_8 &= H_2 + H_7 = H_3 + H_6 = H_4 + H_5 = 0 \quad \text{and} \quad 1 \quad 1 \quad 1
\end{align*}
\]

The syndrome \( s = m - vH^T \) of size 4 is equal to:
- the zero vector (\( s = 0_4^T \)) with a probability of \( \frac{1}{2^4} \) \( \rightarrow \) no (0) change of the cover vector;
- a column of \( H \) with a probability of \( \frac{8}{2^4} \) (8 different vectors representing the 8 column of \( H \) on the \( 2^4 \) possibles of \( GF(2^4) \)) \( \rightarrow \) one (1) change;
- a sum of two distinct columns of \( H \) with a probability of \( \frac{7}{2^4} \) (7 vectors obtained by summation two to two of the different columns of \( H \)) \( \rightarrow \) two (2) changes.

The average number of changes made by the insertion of the message is:

\[
nbr_{change} = 0 \times \frac{1}{16} + 1 \times \frac{8}{16} + 2 \times \frac{7}{16} = \frac{11}{8} = 1.375.
\]

Thus the embedding efficiency is:

\[
\text{eff}_{emb} = \frac{\text{size}(m)}{\text{nbr}_{change}} = \frac{4}{\frac{11}{8}} = \frac{32}{11} \approx 2.91.
\]

For a relative payload \( \alpha = \frac{m}{n} = \frac{4}{8} = \frac{1}{2} \).

This value of the embedding efficiency is much greater than \( (n - k)/2 = 2 \) and constitutes the largest possible using a binary code with the same characteristics \( (n = 8 \) and \( k = 4) \) for a constant profile.

### 5.2.4 Optimality condition of the proposed scheme

The proposed scheme is not suitable for messages of size less than or equal to \( p = \log_2(n) \), with \( n \) the block length of the polar code. Then the messages we want that treatment with this scheme provides a minimum number of changes must verify size \( (m) = m > p \). To satisfy this requirement we can adjust either the size of the message or that of the vector cover. Because the message is given in advance, it would be easier and wiser to choose the second option which consists of choosing the cover vector so that its size \( n \) satisfies the criterion \( \log_2(n) < m \). This criterion imposed is logic since for \( m \leq p \) some columns of the parity check matrix \( H \) are related (they are identical because we are in the binary case) which favors more changes. Note that even messages of size \( m \leq p \) can be inserted and extracted at reception but the minimum number of changes is no longer guaranteed.

To illustrate what we have said we choose \( p = 3 \) \( (n = 2^3 = 8) \). If we take \( k = 5 \) (i.e. \( m = n - k = 3 = p \)),

\[
H = \begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 1 & 1 & 0 & 0 & 1 & 1
\end{bmatrix}
\]

with \( A_c = \{1, 2, 3\} \) and \( A = \{4, 5, 6, 7, 8\} \).

The successive columns of \( H \) are 2 to 2 identical.
5.3 Steganographic scheme with wet paper

In the scheme we have defined, we considered the case of a constant profile. We will show in this section that the proposed scheme can be adapted to the case of wet paper.

In the case where the costs of embedding changes \( \rho_l = 1 \) (constant profile), the minimization of the distortion \( D \) amounts to minimize the number of modifications of the cover vector by seeking the stego vector \( y \) the closest to \( v \). On the other hand if the \( \rho_l \) are arbitrary in \([0, \infty]\) we have to define the scheme for the minimization of the number of changed positions by taking into account constraints of the cost values \( \rho_l \). So we have to modify the pixels where the changes are less perceptible (with the smallest values of \( \rho_l \)). Fridrich and al. have identified three profile types for steganographic schemes [6]: the constant profile \( (\rho_l(x) = 1) \), the linear profile \( (\rho_l(x) = 2x) \), and the square profile \( (\rho_l(x) = 3x^2) \). We will develop a technique of steganography to minimize the embedding impact for a given distortion.

Consider an arbitrary profile defined by \( \rho = \{\rho_l\}_{l=1}^{n} \) [13]. Our goal is to adapt the proposed scheme previously to this general case of the distorted profiles. J. Fridrich and al. [6] have proposed two methods to apply their scheme to the case of wet papers. The first consists of locking a certain number of positions \( t \) (wet elements) of stego object and change only the bits corresponding to the dry elements. The success of this method depends, of course, on the number of locked positions and the type of encoding used. The value of \( t \) should not exceed the dimension \( k \) of the code as defined also in [4] and [5]. Otherwise the search of stego vector would provide no solution. The second technique consists to lock the positions at which the changes will be most visible. Fridrich and al. have also shown that this method is well adapted in practice. If the number of wet elements is greater than \( k \) we allow ourselves to change some\(^2\). To propose a method of steganography with codes polar wet paper we will use the latter approach which is more practical and suitable for our scheme.

Since our scheme consists of two parts, the second improves the first, it is thus necessary to see how each of these two methods is applied to the case of wet papers.

Recall that the first method is closely related to the form of the matrix parity check of the polar code and exploits the steganography relation \( yH^T = m \) by locking some positions of the cover vector. Consequently it is independent of the type of the considered profile, and thus applies identically to the case of constant profile. In this case if the solution offered by this method corresponds to the optimal one it will not be necessary to apply the second method.

Concerning the second method whose implementation depends on the profile used, changes should be made to define a scheme to wet paper steganography. The problem is the same as in the case of constant profile (optimization problem, specifically minimization), we will use the same principle of linear programming to find our optimal solution. The initial solution and the constraints have not changed. What does change here is the objective function. Indeed, in the case of constant profile, the goal was to minimize the Hamming weight of error vector \( e \), while for an arbitrary profile, the goal is to minimize the distortion function (3). Rewrite this relation and the functions of insertion and extraction depending on the change vector \( e \):

\[
D(e) = \sum_{i=1}^{n} \rho_i e_i,
\]

where \( |v_i - y_i| = e_i \) and \( 0 \leq \rho_i \leq \infty \) the modification cost of the LSB \( v_i \) of a pixel into \( y_i \).

\[
Emb(v, m) = \arg \min_{e \in C} D(e)
\]

\[
Ext(y) = yH^T = m \iff eH^T = s = m - vH^T
\]

We can see that our objective function \( f(e) = \langle c, e \rangle \), which must be written as a vector scalar product between the cost vector of the linear program and the variable \( e \), appears well in the expression of \( D(e) \). Consequently the cost elements of the vector are represented by changes the costs of pixels during the insertion.

Let \( c = \rho = \{\rho_l\}_{l=1}^{n} \) and we find the same form of linear program. And the resolution can be done in the same way that in the case of constant profile.

Note that we can also apply the first approach which consists in fixing \( t \) bits (wet elements) of the cover vector by assigning values \( \rho_l = \infty \) (large values in practice) and values \( \rho_l = 1 \) to dry elements. But the condition \( t \leq k \) is needed in this case.

6 EXPERIMENTAL RESULTS

To verify the efficiency of our scheme and the invisibility of the hidden messages using this scheme, we have tested it on different images in gray scale PGM format size \((512 \times \)

---

1 The three types of profile are particulars cases of this consideration.
2 If we consider that the number of wet elements is equal to \( t > k \), we can change \( t - k \) of them.
512) pixels. The images are taken from the BOSS database.

To make the message less detectable, we choose to permute the pixels of the cover image before making the insertion. Remember that we had a permutation matrix $B_n$, square and of dimension a power of 2, that permutes the rows of a given square matrix. And it happens that our images are of size $512 \times 512$ and 512 is a power of 2. We can use $B_n$ for the permutation. Thus the changes will be spread over isolated pixels of the image making it less detectable the secret message inserted and thus allowing a more secure insertion. After insertion, it is necessary to find the original order of pixels of the cover image. To achieve this we still use the matrix $B_n$. Since it is invertible and equal to its own inverse, it suffices just to repeat the same operation as in the permutation (matrix product of $B_n$ by the matrix of the cover image). This choice of permutation is an example among many others (we might use the matrix $R_n$ for example) and may be secretly shared between the sender and receiver.

First we insert a message of size 3 ko or 4576 bits in an image (10.pgm) BOSS database. The first and most simple evaluation to do concerns the visual imperceptibility. The changes in the stego image are invisible to the naked eye as shown in Figures 6. Hence the first and main goal of steganography is achieved. If we are confronted with a passive attacker we have clearly seen that by comparing the cover and stego images on the one hand and their histogram on the other hand, for an attacker to "semi-active", the distinction between the two images is almost impossible. Because there is a very small difference between the histograms of the cover and stego images that is very difficult to perceive. This difference is more perceptible if the size of the message to be inserted increases. The scheme is even more secure that the attacker has only the stego image to see if it contains a secret message or not. He should therefore use much more sophisticated means to reach to detect the presence of secret message.

Like any good steganographic scheme, ours allows to recipient to extract the secret message in full, without any alteration. What we can see well with the example given above. The message inserted in the cover image (Figure 7) is identical to that extracted from the stego image (Figure 8).

![Figure 6: The cover image 10.ppm and stego image 10_stego.png and their histograms.](image)

![Figure 7: Inserted message.](image)

---

3 BOSS (Break Our Stego-System): the competition for the pour les attacks of the steganographic schemas.
To evaluate the performance of our scheme we calculated the MSE and the PSNR with the equations (1) and (2). We randomly generate 10 messages of different size that we insert in 5 images (1.pgm, 10.pgm, 100.pgm, and 1000.pgm 10000.pgm). We averaged the MSE and PNSR and the results are shown in Figure 9.

The value of PNSRs vary between 55 dB and 66 dB. That is greater than 55 dB. Prospect of improvement can guide us in the search of scheme in a single step. To get it we can sink to use the LP decoding by changing the polytope \( y \mathbf{H}^T = m \). In the definition of our scheme we used a generator matrix \( G_2 \) constructed from a sub-matrix \( G_{2^p} \). Korara and al. [14] showed that it is possible to use a sub-matrix \( G_{2^p} \). We can try to see what would give a construction from \( G_2 \) by drawing inspiration from the construction of the parity check matrix \( H \) of the STC [6] from the sub-matrix \( \tilde{H} \) of various sizes.

**REFERENCES:**

[1] R. Crandall, “Some notes on steganography”, Posted on Steganography Mailing List (1998), available http://os.inf.tu-dresden.de/~westfeld/crandall.pdf.

[2] A. Westfeld, “High capacity despite better steganalysis (F5 – a steganographic algorithm)”, In: Moskowitz, I.S. (ed.) IH 2001. LNCS, vol. 2137, pp. 289–302, Springer, Heidelberg (2001).

[3] Schönfeld, D., Winkler, “A Embedding with syndrome coding based on BCH codes”, in: Proceedings of the 8th ACM Workshop on Multimedia and Security, pp. 214 – 223, 2006.

[4] Rongyue Zhang, Vasilii 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 Heidelberg 2009.

[5] F. Galand and C. Fontaine, “How Reed-Solomon Codes Can Improve Steganographic Schemes”, in Information Hiding, Rennes, France, 2009.

[6] Tomáš Filler, Jan Judas and 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.

[7] Xuanwen Luo, Qiang Cheng, Joseph Tan, “A Lossless Data Embedding Scheme For Medical in Application of e- Diagnosis,” Proceedings of the 25th Annual International Conference of the IEEE EMBS Cancun, Mexico. September 17-21, 2003.

[8] AAID DJAMEL, “Étude numérique comparative entre des méthodes de résolution d'un problème de transport à quatre indices avec capacités,” Thèse, École Doctorale de Mathématiques, pôle de Constantine, 2010. [Online]: http://bu.umc.edu.dz/theses/math/AAI5587.pdf

[9] Narendra Karmarkar (1984). “A New Polynomial Time Algorithm for Linear Programming”, Combinatorial, Vol 4, nr. 4, p. 373–395.

[10] Gilles Savard GERAD, “Introduction aux méthodes de points intérieurs,” Extrait de notes de cours, Département de mathématiques et génie industriel Ecole Polytechnique de Montréal.

[11] E. Arıkan, “Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memory-less channels”, IEEE Trans. Inform. Theory, vol. IT-55, pp. 3051–3073, July 2009.

[12] N. Goela, S. B. Korada, and M. Gastpar, “On LP Decoding of Polar Codes,” submitted to IEEE Trans. Information Theory, 2010.

[13] Tomáš Pevný, 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.

[14] S.B. Korada, E. Sasoglu, and R. Urbanke, “Polar Codes: Characterization of Exponent, Bounds and Construction”, submitted to IEEE Trans. Inform. Theory, January 2009 (http://arxiv.org/abs/0901.0536).