Convolutional Code Theory based Steganography Technique

M.I. Garcia-Planas, L.E. Um
Universitat Polit`ecnica de Catalunya Departament de Matem`atiques Mineria 1, Barcelona Spain
University of Douala Faculty of Sciences Douala, 24157 Campus PK17, Cameroon

Received: August 24, 2021. Revised: February 9, 2022. Accepted: February 21, 2022. Published: March 11, 2022.

Abstract: Steganography, as it is known, is a technique to hide a secret message within a message or collection of data that is not secret, and a problem in mathematics is to decipher the secret included in the message, to solve this problem a good tool It is the theory of codes. Unlike the existing works that use block codes to hide information using the steganographic process, in this work, we propose the use of convolutional coding theory in steganography to encrypt and decrypt messages methods to decrypt messages. Here, we suggest a steganographic protocol based on convolutional codes in which they are defined as discrete linear dynamical systems with which the properties on controllability and observability characteristic of linear systems theory can be applied, in particular the properties of output observability character which can be easily described using matrix language. The proposed decoding algorithm used for dissimulation is a linear decoding method, which has decreased both the time and space complexity, compared to the Viterbi decoding algorithm, sometimes used in other cases; indeed, we go from 2^{h}.n to $2^{h=2}$.n, in memory space (with h: constraint height, and n: length of cover object). Moreover, the time complexity is better, while we can also denote that with the convolutional approach, we intend to take advantage of the time-depending transaction.

Key–Words: Steganography, Convolutional codes, Coding, Decoding, Syndrome, Linear Systems

I. INTRODUCTION

Based on the results obtained in the work of the doctoral thesis [20] and on the results presented in [21], we have extended the work proposing different convolutionally-based stegoschemes and designing better stegoschemes based on convolutional codes under linear systems theory. In this paper, we present these results that are an application of convolutional coding theory in steganography. As we know, block codes are largely used for dissimulation of informations in a steganographic process, such as in [2, 5], where the syndrome coding method is used for most cases.

Here, we suggest a steganographic protocol based on convolutional codes defined under linear systems theory. As it is known, convolutional codes can be given by a quadruple of matrices (A, B, C, D) representing discret linear dynamical systems (see [6]), which can be the support for an embedding protocol of a message m into a cover object u, with as little modification as possible. The use of convolutional codes allows using the structural properties of systems theory that facilitate decoding [7]. The structural properties of the convolutional codes are useful to find conditions on the convolutional code that allow to maintain the imperceptibility of the change in the cover, regardless of the cover or the message to be hidden;

This paper is organized as follows: right after the introduction, the second section presents steganography in general; then, the third will be about conceiving some convolutional codes under linear systems, the fourth one will be reviewing the relationship between steganography and coding, as well as the syndrome coding method. The last section is concerned with steganography and convolutional coding, where we suggest both an embedding and an extracting algorithm, for which the space and time complexity of the dissimulation are very much decreased compared to the trellis' approach, and illustrated by examples. That new method invoked is based upon a new decoding method specifically designed for convolutionals, which is linear and inspired by the linear systems point of view of convolutional codes.

II. STEGANOGRAPHY

Steganography is a form of protection of communication, as it is known as a technique being used in order to protect some information to be exchanged by hiding its original existence, onto some digital files, could it be photographies or videograms. As it is well known, cryptography is the technique and science behind the protection of messages and information to be transmitted, whereas the idea of steganography is actually to prevent a nasty observer to even detect the need for that protection beforehand, and it is also depending on the situations, as for instance in places where cryptography cannot be used. Sometimes, it is also possible to mix both techniques for protection of communication and information as well. The classic example known to illustrate use of a steganographic scheme is the prisoner's problem. In that scheme, Alice and Bob are two prisoners imprisoned in separate cells, and need to plan their escape; they can only communicate under the surveillance of a warden, so undetectability is imperative throughout their exchange [5].

Characteristics of a steganographic scheme

A steganographic scheme is characterized by some necessary conditions and components which are: (see [13, 18, 19])

i) the choice of a communication support

ii) the message to be embedded

iii) embedding function

iv) extracting function

v) optional steganographic key-management

The embedding and extracting functions as their names show consist of functions responsible for hiding messages or information. For digital steganography, as it is our case, the purpose is to hide or embed a sequence of bits in that digital cover, within some conditions such as making sure that the change is not visually perceptible. Knowing that the choice of covers is quite large in digital media (graphic files, messages, etc), and also dictated by the nature of the information to embed, the performance of a steganographic method can be assessed over a certain cover object mainly by its average distortion and its embedding rate.

For instance, a very popular method used in digital steganography is called Least Significant Bit (LSB) steganography, and it consists of hiding information within a graphic file, by replacing the least significant bits of specifically selected pixels by message bits, in such a way that they are "visually imperceptible". ([10, 11])

Definition 1. *Let k and n be two integers, and A a finite field. A digital steganographic scheme S of type $[k,n]$ over a finite alphabet A is a pair of functions:*

$$\begin{array}{llll} emb: & A^n \times A^k & \longrightarrow A^n \\ rec: & A^n & \longrightarrow A^k \end{array}$$

such that

$$rec(emb(c,m)) = m \textit{ for all } c \in A^n \textit{ and } m \in A^k,$$

with m being the secret message, and c the cover vector.

The scheme is denoted as: $S(\mathrm{emb}, \mathrm{rec})$.
Note that if $c' = emb(c,m)$, then $rec(c') = m$

For the scheme S, the following characteristics are of importance:

a) the cover length n

b) the embedding capacity k

c) the embedding radius r, defined by:

$$r = \max\{d(c, emb(c,m)) \mid c \in A^n, m \in A^k\}.$$

where d is the Hamming distance

d) the average number of embedding changes R_a, given by:

$$R_a = \frac{1}{q^{kn}} \sum d(c, emb(c,m))$$

where $q = \sharp A$; (see [13])

Proposition 2 ([15])**.** *Let $S = (emb, rec)$ be a steganographic scheme of type $[n,k]$ defined over an alphabet A. Then:*

1) the map rec is surjective;

2) for fixed $c \in A^n$, the map $emb(c,-) : A^k \longrightarrow A^n$ is injective.

In particular, $k \leq n$.

Proof. From the condition $rec(emb(c,m)) = m$. □

Knowing that the purpose of a stegoscheme is to embed as much information as possible, with as few changes as possible, we have the definition of a proper scheme;

Definition 3. *A steganographic scheme $S = (emb, rec)$ is said to be* proper *if the number of changes produced in the cover is the minimum possible allowed by the recovering map.*

$$\begin{array}{l} d(c, emb(c,m)) = \\ d(c, rec^{-1}(m)), \textit{ for all } c \in A^n \textit{ and } m \in A^k \end{array}$$

where the notation $rec^{-1}(m)$ is used to denote an arbitrary recovery of m.

We have the following proposition:

Proposition 4 ([15]). *Let $S = (emb, rec)$ be a steganographic scheme of type $[n, k]$ over A. There exists a proper stegoscheme $S^* = (emb^*, rec)$ of the same type $[n, k]$ such that $R_a(S^*) \leq R_a(S)$.*

Proof. The proof of the proposition can be found in [15] . □

III. SYNDROME CODING IN STEGANOGRAPHY

In practice, we are interested in practical methods that can embed an m-bit message in an n-element cover, while keeping the expected distortion $E[D(x, Emb(x, m))]$ as small as possible [5]. In syndrome coding, the embedding and extraction mappings are realized using a binary linear code $\mathcal{C}$ of length n and dimension $n - m$:

$$\begin{array}{rcl} Emb(x, m) & = & arg\ min_{y \in \mathcal{C}(m)}\ D(x, y) \\ Ext(y) & = & \mathbb{H}y, \end{array}$$

where $\mathbb{H} \in \{0, 1\}^{m \times n}$ is a parity-check matrix of the code $\mathcal{C}$, $\mathcal{C}(m) = \{z \in \{0, 1\}^n, \mathbb{H}z = m\}$ is the coset corresponding to syndrome m, and all operations are in binary arithmetic.

Let us first assume that the set of single-letter distortions is bounded by a constant, $0 \leq \rho_i < C$. Let us denote that the following distortion profiles are of interest in steganography, and will be used for benchmarking the proposed methods:

- the *constant profile*, $\rho(x) = 1$, when all pixels have the same impact on detectability when changed;
- the *linear profile*, $\rho(x) = 2x$, when the distortion is related to a quantization error uniformly distributed on $\left[-\dfrac{Q}{2}, \dfrac{Q}{2}\right]$ for some quantization step $Q > 0$;
- and the *square profile*, $\rho(x) = 3x^2$, which can be encountered when the distortion is related to a quantization error that is not uniformly distributed.

IV. CONVOLUTIONAL CODES UNDER LINEAR SYSTEMS POINT OF VIEW

A convolutional code is a type of error-correcting code in which each k-bit information symbol (each k-bit string) to be encoded is transformed into an n-bit symbol, k/n being the code rate and the transformation is a function of the last information symbols contained in the memory of the physical encoder. Those are block codes over polynomial rings.([3])

Definition 5. *[16] A convolutional code of length n and dimension k is a k-rank submodule of $\mathbb{F}_q^n[z]$.*

Corollary 6 ([17]). *Let C be a convolutional code.*

Then there exists a polynomial matrix $G(z)$ (called encoder) of size $n \times k$ and having maximal rank such that

$$\mathcal{C} = \{v(z) \mid \exists u(z) \in \mathbb{F}^k[z], v(z) = G(z)u(z)\}. \quad (1)$$

We denote by ν_i the maximum of all degrees of each of the polynomials of each line $i, i = 1, \ldots, n$, we define the complexity of the encoder as $\overline{\delta} = \sum_{i=1}^{n} \nu_i$, and finally we define the complexity of a convolutional code $\delta(\mathcal{C})$ as the maximum of all degrees of the largest minors of $G(z)$ that we will write simply by δ if no confusion is possible [9].

Given a convolutional code $\mathcal{C}$ with its encoding matrix $G(z)$, there always exists its first-order representation, and a quadruple (A, B, C, D) associated to $G(z)$ given by the equations

$$\left\{ \begin{array}{rl} x(t+1) & = Ax(t) + Bu(t) \\ y(t) & = Cx(t) + Du(t) \end{array} \right. \quad (2)$$

where $A \in M_{\delta}(\mathbb{F})$, $B \in M_{\delta \times k}(\mathbb{F})$, $C \in M_{p \times \delta}(\mathbb{F})$, $D \in M_{p \times k}(\mathbb{F})$ (with $p = n - k$) are constant matrices over the field $\mathbb{F}$, and $u(t) \in \mathbb{F}^k$, $x(t) \in \mathbb{F}^{\delta}$, $y(t) \in \mathbb{F}^p$ are the input, state and output vectors, respectively; (t is a discrete variable: $t \in \mathbb{N}$).

We will denote a system simply as the quadruple of matrices (A, B, C, D).

Theorem 7. *Let $C \subset \mathbb{F}^n$ be a k/n convolutional code, of complexity of convolutional code δ. Then, there exist matrices K, L of sizes $(\delta + n - k) \times \delta$, and a matrix M, of size $(\delta + n - k) \times n$, with entries in $\mathbb{F}$, such that the convolutional code C is defined by*

$$\begin{array}{rl} C = & \{u(z) \in \mathbb{F}^n[z] | \exists x(z) \in \mathbb{F}^{\delta}[z] : \\ & zKx(z) + Lx(z) + Mu(z) = 0\}. \end{array}$$

From Theorem 7 and taking into account the following proposition

Proposition 8. *Let (K_1, L_1, M_1) be another representation of the convolutional code $\mathcal{C}$. Then there exist invertible matrices T and S of adequate sizes, such that*

$$(K_1, L_1, M_1) = (TKS^{-1}, TLS^{-1}, TM).$$

We have the following corollary

Corollary 9. *The triple* (K, L, M) *can be written as:*

$$K = \begin{pmatrix} -I_\delta \\ 0 \end{pmatrix},\ L = \begin{pmatrix} A \\ C \end{pmatrix},\ M = \begin{pmatrix} 0 & B \\ -I_{n-k} & D \end{pmatrix}.$$

And we deduce the following one:

Corollary 10.

$$\begin{aligned} \mathcal{C} = \ & \{v(z) \in \mathbb{F}^n[z] \mid \exists x(z) \in \mathbb{F}^\delta[z] : \\ & \left(\begin{smallmatrix} zI-A & 0 & -B \\ -C & I & -D \end{smallmatrix}\right) \begin{pmatrix} x(z) \\ v(z) \end{pmatrix} = 0\} \end{aligned}$$

With initial condition $x(0) = 0$, a solution of system (2) can be obtained by making use of the Z-transform. Let $u(z)$, $x(z)$, $y(z)$ be the Z-transforms of the variables u, x, y of a time- invariant linear system. Then by applying the Z-transform to the equations of the system we have

$$\begin{cases} zx(z) & = Ax(z) + Bu(z) \\ y(z) & = Cx(z) + Du(z) \end{cases} \tag{3}$$

and as a result we have

$$y(z) = (C(zI_\delta - A)^{-1}B + D)u(z), \tag{4}$$

which is the transfer function of the system, and $C(zI_\delta - A)^{-1}B + D$ is the transfer matrix.

Control concepts: Controllability and Observability

In control systems theory the major concepts are controllability and observability, introduced by R. Kalman in 1960 ([12]).

Definition 11. *A linear system* (A, B, C, D) *is a controllable system if the controllability matrix*

$$\mathcal{C} = \begin{pmatrix} B & AB & A^2B & \dots & A^{\delta-1}B \end{pmatrix} \tag{5}$$

of the system has full rank δ*, where* δ *is the complexity of the code.*
Equivalently (Hautus test [8]), a linear system (A, B, C, D) *is controllable if and only if*

$$\text{rank } \begin{pmatrix} zI_\delta - A & B \end{pmatrix} = \delta, \textit{ for all } z \in \overline{\mathbb{F}}, \tag{6}$$

where $\overline{\mathbb{F}}$ *denotes the algebraic closure of* $\mathbb{F}$*.*

Definition 12. *A linear system* (A, B, C, D) *is said to be observable if the observability matrix*

$$\mathcal{O} = \begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{\delta-1} \end{pmatrix} \tag{7}$$

of the system has full rank δ*,*
or equivalently (Hautus test), a linear system (A, B, C, D) *is observable if and only if*

$$\text{rank } \left(\begin{smallmatrix} zI-A \\ C \end{smallmatrix}\right) = \delta, \textit{ for all } z \in \overline{\mathbb{F}}, \tag{8}$$

where $\overline{\mathbb{F}}$ *denotes the algebraic closure of* $\mathbb{F}$*.*

Calling $T_\ell(A, B, C, D)$ (that we simply write T_ℓ if no confusion is possible) the matrix

$$T_\ell = \begin{pmatrix} C & D & & & & \\ CA & CB & D & & & \\ CA^2 & CAB & CB & D & & \\ \vdots & & & \ddots & \ddots & \\ CA^\ell & CA^{\ell-1}B & CA^{\ell-2}B & \dots & CB & D \end{pmatrix}. \tag{9}$$

We have the following.

Proposition 13. *A system* (A, B, C, D) *is output observable if and only if the matrix* T_ℓ *has full row rank for all* $\ell \in \mathbb{N}$*.*

Proof. The proof of the proposition can be found in ([6]) . □

V. STEGANOGRAPHY AND CODING

There are some interesting steganographic protocols that have already been defined from coding theory, considering the fact that error-correcting codes are used in order to detect and/or correct errors, during data transfer. If we consider for instance some of the methods involving the existence of the parity check matrix, we can implement syndrome coding. Considering a steganographic protocol within the spatial domain of gray scale image, inspired by [14]. This approach suggests to divide the cover block into blocks of equal sizes.

For instance, let us consider the following protocol having the cover object v whose LSB (Least Significant Bits) values are given by $v = \{v_0, v_1, \dots, v_n\}$ over $\mathbb{F}_2^n$, the message $m = \{m_0, m_1, \dots, m_t\}$ with $t < n$ over $\mathbb{F}_2^t$, the code given by its parity matrix $\mathbb{H}$.
Embedding m into v produces the stego object $r = \{r_0, r_1, \dots, r_n\}$, given by the relation:

$$m = r.\mathbb{H}^t \tag{10}$$

In order to extract m, it is the equation (10) that is used. After embedding, some of the bits of the cover block are modified (either 0 or 1); if we consider e, the flip pattern representing the modified bits of that cover block, the stego object in polynomial form is given by:

$$r(X) = v(X) + e(X) \tag{11}$$

From both equations (10) and (11), we have:

$$m - v.\mathbb{H}^t = e.\mathbb{H}^t \tag{12}$$

which gives us the extraction formula (formula to recover the original message m, which had been previously embedded).

Example 14. *In the case of the $F5$ algorithm [22], the technique used with an $[n, n-k, 1]$-code consists of embedding k bits into an n-length cover sequence by changing at most 1 bit.*

This method is called the syndrome coding, and from a steganographic point of view, we need to find a minimal number of flips of $e(X)$ to decrease modification.

VI. STEGANOGRAPHY AND CONVOLUTIONAL CODING

Before going any further, let us recall some notions involved within the construction of the steganographic protocol based on coding.

A. The purposes and interest

From what we already get out of the traditional steganographic procedure, the idea is to suggest an efficient steganographic protocol that is implementable on convolutional encoding/decoding. As we know, there exists several steganographic protocols defined over error-correcting block codes, within the decoding method actually used to detect and correct errors, in order to introduce a minimum amount of errors, as few as possible [15]. We are using the same approach on our own steganographic model here, with the twist and particularity of convolutional codes, which requires for the sequential characteristic of the implementation, for instance embedding of a sequence while transmission of a message, file or image during an undetermined, or semi-infinite sequence of time. We are also inspired by [4], and [1] which still goes a very different direction. Indeed, the general plan is to introduce “as little modification as possible”, onto the cover sequence, in order to embed another digital sequence (preferably, of less length).

As we are trying to get there, some key points we have to cover are:

1. the conditions for the steganographic scheme to be established, which means rec and emb functions to be right, and well described.
2. the conditions for the modified subsections, which is the bound on the flipping bits, the bits that are being altered while embedding, in order to alter the least bits possible;
3. the classical bound of imperceptibility as far as the embedding radius r.

B. Syndrome-trellis codes and Implementation

The idea here [5] is to create efficient syndrome-coding schemes for an arbitrary $\alpha \leq \frac{1}{2}$, since many results suggest that secure payload for digital image steganography is always far below $\frac{1}{2}$. The idea is to select the parity-check matrix $\mathbb{H}$ in a special form that allows representing every solution of $\mathbb{H}y = m$ as a path through a trellis. The optimal y closest to x is then found using the Viterbi algorithm.

In this case, the parity-check matrix $\mathbb{H}$ is obtained by placing small submatrices $\widehat{\mathbb{H}}$ of size $h \times w$, next to each other and shifted down by one row, which leads to a sparse, banded $\mathbb{H}$. The height h of the submatrix, also called the constraint height, is a design parameter that affects the algorithm speed and efficiency (typically, $6 \leq h \leq 15$). The width of $\widehat{\mathbb{H}}$ is dictated by the desired relative payload α: if α is equal to $\frac{1}{k}$ for some $k \in \mathbb{N}$, select $w = k$. For general payloads α, find k such that $\frac{1}{k+1} < \alpha < \frac{1}{k}$. The matrix $\mathbb{H}$ will contain a mix of submatrices of width k and $k+1$ so that the final matrix $\mathbb{H}$ is of size $[\alpha n] \times n$. In this way, we can create a parity-check matrix for an arbitrary rational $\alpha \leq \frac{1}{2}$.

So, we assume $\alpha = \frac{1}{w}$ and thus the matrix $\mathbb{H}$ is of the size $b \times (b.w)$, where b is the number of copies of $\widehat{\mathbb{H}}$ in $\mathbb{H}$.

C. Proposition of different convolutionallybased Stegoschemes

In this subsection, we show our construction and implementation of the steganographic scheme based on convolutional codes, most specifically based on the convolutional codes approach based on linear systems, unlike in [4]. As already defined earlier, we consider our convolutional codes, by their realization representation given by the quadruple of matrices (A, B, C, D). Before going any further, let us introduce some notions we use throughout the process.

Definition 15. *The quasi-syndrome denoted by s is the value from which we choose the estimated pertubation e for embedding m into u, at each step $t \in \{0, 1, \ldots, \ell\}$ of the embedding process.*
It is given by: $s(t) = \mathbb{H}e(t)$.

Let (A, B, C, D) be a system representing a convolutional code. We denote by τ the minimum number of linearly-dependent columns of D.

By analogy to block coding theory, τ can be related to the minimal distance of a block code whose parity check matrix is represented by D.

For the realization representation scheme, we are using the decoding method and protocol implemented for this specific representation of convolutionals, by involving the output-observability matrix T_ℓ given by the matrix (9). Indeed, the decoding procedure in this case consists of solving the system:

$$T_\ell \begin{pmatrix} x(0) \\ u \end{pmatrix} = y. \qquad (13)$$

We recall that it is usual to consider the initial state of the system $x(0) = 0$, as in our case for instance; therefore, our new output-observability matrix is reduced to:

$$\widehat{T}_{\ell-1} = \begin{pmatrix} D & & & & \\ CB & D & & & \\ CAB & CB & D & & \\ \vdots & & \ddots & \ddots & \\ CA^{\ell-1}B & CA^{\ell-2}B & \dots & CB & D \end{pmatrix} \qquad (14)$$

In order to do the embedding, the process goes by considering the output-observability matrix as the parity check matrix. The model of steganography we build is inspired by the (A, B, C, D)-representation of the convolutional code. Knowing that we are normally using its structure on convolutional codes for the decoding, step by step ([7]), when it comes down to steganography it is our base for the embedding function. In this case, we decide to approach it in a sequential fashion which means that for each step of the time-related steganographic process, at each $t = 1, \dots, \ell$, the protocol consists of “embedding” the message sequence, by altering lightly the cover sequence with some error, in order to build the stego-sequence. In order to do so, we need to figure out the best sequence corresponding to the flipping bits that minimizes the modification, which corresponds to the coset leader of the list of potential “error vectors $e(t)$” for embedding $m(t)$ in $u(t)$ by the formula: $u(t) + e(t)$.

When it comes to the retrieval of the hidden message, that is when the actual output-observability matrix explicitly appears for the solving of the corresponding equation ([6]). Indeed, the recovery process of the embedded message consists of the encoding of the stego-sequence. On the other hand, it is also an analogy of the syndrome steganographic protocol method, in order to retrieve the hidden message; actually, the idea is to extract at each step from the stego-sequence, using the control ”block of matrices“, each part of the current embedded message.

The following algorithm provides the method for the embedding process.

Embedding algorithm for embedding function emb
Input: Message m, Cover sequence u
Output: Stego-sequence St

```
1: if rank $D$ is row maximal then
2:     for t:=0 to $\ell$ do
3:         $s(t) = m(t) - Du(t) - \sum_{k=0}^{t-1} CA^{t-1-k}B(u(k) + e(k))$;
4:         list $sysdTableD = \{e(t)\}$;
5:         compute $\min_{e \in \text{list}} w(e(t))$;
6:         pick and store one corresponding $e(t)$;
7:         $St(t) = u(t) + e(t)$;
8:     end for
9:     $St = (St(0), \dots, St(\ell))$
10: else
11:     Choose an adequate code with $D$ corresponding
12: end if
```

The next algorithm is used to extract the embedded message.

Extracting algorithm for recovery function rec

```
1: $m = \widehat{T}_{\ell-1}\left(u + e\right)$
2: for t:=0 to $\ell$ do
3:     $m(t) = Du(t) + De(t) + \sum_{k=0}^{t-1} CA^{t-1-k}B(u(k) + e(k))$; In $\mathbb{F}_2$, let us consider the code $\mathcal{C}(A, B, C, D)$ defined by the list $sysdTableD = \{e(t)\}$;
4: end for
5: $m(t) = (m(0), \dots, m(\ell))$
```

Example 16.

$$A = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}, B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix},$$

$$C = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \end{pmatrix}, D = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}$$

We observe that the system is output-observable.

Let us consider the message to be embedded be $m = (m(0), m(1), m(2))$*, and is done so within those sections, at each step, in a cover sequence denoted by* u*.*

Let us consider $m = (10,\ 00,\ 01)$

We consider that given each input sequence u*, we will try to find out characteristics of each sequence* e *of the flipping bits that were added to* u*, in order to embed the message* m*. The corresponding quasi-syndrome is denoted by* s*.*

Consider the decoding matrix for the convolutional codes given by:

$$\widehat{T}_{\ell-1} = \begin{pmatrix} D & & & & \\ CB & D & & & \\ CAB & CB & D & & \\ \vdots & & \ddots & \ddots & \\ CA^{\ell-1}B & CA^{\ell-2}B & \dots & CB & D \end{pmatrix}$$

At each step here: $0, 1, \dots, \ell$*, we will be try to evaluate our error value* e

We already have: $m = \widehat{T}_{\ell-1}\left(u + e\right)$

As inputs, we have m *and* u*.*

First of all, let us assess the coset of potential errors e; as a general formula, m and u are given by: $m(t) = \sum_{k=0}^{t-1} CA^{t-1-k}B(u(k)+e(k)) + D(u(t)+e(t))$; which means that:

$$De(t) = s(t) = m(t) - \sum_{k=0}^{t-1} CA^{t-1-k}B(u(k)+e(k)) - Du(t)$$

Going through all possible cases of syndromes (or cover sequences), we get the coset of errors e.

s	*e*
0 0	*0 0 0*
	1 0 0
1 0	*0 1 1*
	1 1 1
0 1	*0 0 1*
	1 0 1
1 1	*0 1 0*
	1 1 0

Therefore, at each step, there is always a sequence e that can be used for embedding such that: $w(e) \leq 2$.

Let us work with u = (111 010 001) having $m = (10,\ 00,\ 01)$.

At step $t = 0$, we have: $D(u(0) + e(0)) = m(0)$

Then, $s = (0\,0)$, and $e(0) = (0\,0\,0)$

At step $t = 1$, we have: $D(u(1) + e(1)) = m(1) - CB(u(0) + e(0))$

Then, $s = (0\,0)$, we pick $e(0) = (0\,0\,0)$ and $e(1) = (0\,0\,0)$

At step $t = 2$, we have: $D(u(2) + e(2)) = m(2) - CAB(u(0) + e(0)) - CB(u(1) + e(1))$

Then, $s = (1\,0)$, and $e(2) = (0\,1\,1)$.

*Then, for u = **(1 1 1, 0 1 0, 0 0 1)** , we can embed m = (10, 00, 01) with the flip pattern: e = **(0 0 0, 0 0 0, 0 1 1)**.*

For this operation, we embedded 6 bits in a 9-length cover sequence by changing 2 bits.

D. Design of better stegoschemes based on convolutional codes under linear systems theory

This specific subsection is about the design, and implementation of a revamped version of the steganographic scheme we suggested earlier, based on the convolutional codes approach based on linear systems, with their realization representation given by the quadruple of matrices (A, B, C, D). In fact, in order to obtain a desirable design, with performance close to the bound, we will be inspired in our implementation and new scheme, with the work before us, from J. Fridrich and al, pertaining the approach from which we derive greater evaluation for the *relative paoyload* α, the *embedding efficiency* e=$\frac{\alpha}{d}$, with relation to the best *profile* we decide to adapt to our examples.

From the previous paragraph, given that our output-observability matrix is used as the parity-check matrix, from the embedding process, the model of steganography we build is yet inspired by the (A, B, C, D)-representation of convolutional codes. However, in order to obtain similar performance outputs as in [5], we will have to incorporate major changes, especially in the hope of respecting the banded structure of $\mathbb{H}$.
For our new parity-check matrix, we will consider cases of A being the null matrix, i.e

$$A = 0$$

. Indeed, we are aware that $\mathbb{H}$ is built up on submatrices $\widehat{\mathbb{H}}$, with special *constraint height h*, as well as *width w* from which we compute *the relative payload* α, and experience great *embedding efficiency* $e = \frac{\alpha}{d}$.
So, for our new model, in order to be in alignment with the sparse banded representation of $\mathbb{H}$, we will only consider such cases, as mentioned earlier.
Which leads to the decoding matrix:

$$\widehat{T}_{\ell-1} = \mathbb{H} = \begin{pmatrix} D & & & & \\ CB & D & & & \\ & CB & D & & \\ & & \ddots & \ddots & \\ & & & CB & D \end{pmatrix}$$

In our own construction of the parity-check matrix, the submatrix $\widehat{\mathbb{H}}$ is given by:

$$\widehat{\mathbb{H}} = \begin{pmatrix} D \\ CB \end{pmatrix}$$

Therefore, the new algorithms providing methods for embedding, as well as recovery are:

Embedding algorithm for embedding function emb
Input: Message m, Cover sequence u
Output: Stego-sequence St

```
if rank $D$ is row maximal then
    for t:=0 to $\ell$ do
```

```
        $s(t) = m(t) - Du(t) - CB(u(t-1) + e(t-1))$;
        list $sysdTableD = \{e(t)\}$;
        compute $\min_{e \in \text{list}} w(e(t))$;
        store one corresponding, coset leader $e_{leader}(t)$;
        $St(t) = u(t) + e_{leader}(t)$;
    **end for**
    $St = (St(0), \ldots, St(\ell))$
**else**
    Choose an adequate code with $D$ corresponding
**end if**
```

The next algorithm is used to extract the embedded message.

Extracting algorithm for recovery function rec

```
$m = \widehat{T}_{\ell-1}\left(u + e\right)$
**for** t:=0 to $\ell$ **do**
    $m(t) = Du(t) + De(t) + CB(u(t-1) + e(t-1))$;
    In $\mathbb{F}_2$, let us consider the code $\mathcal{C}(A, B, C, D)$ defined by the list $sysdTableD = \{e(t)\}$;
**end for**
$m(t) = (m(0), \ldots, m(\ell))$
```

Implementation design details
For our implementation instances, we will be respecting performance measures, as suggested in [5].
Thus, our *constraint height* h and *relative payloads* α are selected w.r.t. $h \in \{6, \ldots, 12\}$ and $\alpha = \dfrac{1}{w}$, with $w \in \{2, \ldots, 20\}$

Example 17.

$$A = \left(0\right), B = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 \end{pmatrix}$$

$$C = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}, D = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

We observe that the system is output-observable.

Let us consider the message to be embedded be $m = (m(0), m(1), m(2))$*, and is done so within those sections, at each step, in a cover sequence denoted by* u*.*

Let us consider $m = (111,\ 100,\ 001)$

We consider that knowing each input sequence u*, we will try to find out characteristics of each sequence* e *of the flipping bits that were added to* u*, in order to embed the message* m*. The corresponding quasi-syndrome is denoted by* s*.*

Consider the decoding matrix for the convolutional codes given by:

$$\widehat{T}_{\ell-1} = \begin{pmatrix} D & & & & \\ CB & D & & & \\ & CB & D & & \\ & & \ddots & \ddots & \\ & & & CB & D \end{pmatrix}$$

At each step here: $0, 1, \ldots, \ell$*, we will be try to evaluate our error value* e

We already have: $m = \widehat{T}_{\ell-1}\left(u + e\right)$

As inputs, we have m *and* u*.*

First of all, let us assess the coset of potential errors e*; as a general formula,* m *and* u *are given by:* $m(t) = CB(u(t-1) + e(t-1)) + D(u(t) + e(t))$*; which means that:*

$$De(t) = s(t) = m(t) - CB(u(t-1) + e(t-1)) - Du(t)$$

Going through all possible cases of syndromes (or cover sequences), we get the coset of errors e*. Here, we only store the ones with minimum weight.*

s	*e (w < d)*
000	*00000*
001	*00001*
010	*00010*
100	*01000*
011	*00100*
101	*10000*
110	*01010*
111	*01100*

Therefore, at each step, there is always a sequence e *that can be used for embedding such that:* $w(e) \leq d - 1$ *(*$d = 3$*).*

Let us work with $u = (11011\ 01000\ 10101)$ *having* $m = (111,\ 100,\ 001)$*.*

At step $t = 0$*, we have:* $D(u(0) + e(0)) = m(0)$

At step $t = 1$*, we have:* $D(u(1) + e(1)) = m(1) - CB(u(0) + e(0))$

At step $t = 2$*, we have:* $D(u(2) + e(2)) = m(2) - CB(u(1) + e(1))$

Then, at step $t = 0$*,* $s = (1\,0\,1)$*, and* $e(0) = (1\,0\,0\,0\,0)$*;*

at step $t = 1$*,* $s = (0\,0\,0)$*, and* $e(1) = (0\,0\,0\,0\,0)$*;*
at step $t = 2$*,* $s = (0\,0\,0)$*, and* $e(2) = (0\,0\,0\,0\,0)$
Then, for

$$u = \boldsymbol{(11011,\ 01000,\ 10101)},$$

we can embed

$$m = (111,\ 100,\ 001)$$

with the flip pattern:

$$e = \boldsymbol{(10000,\ 00000,\ 00000)}.$$

For this operation, we embedded 9 *bits in a* 15*-length cover sequence by changing only* 1 *bit.*

For this particular case, our submatrix

$$\widehat{\mathbb{H}} = \begin{pmatrix} D \\ CB \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

is of size: $h \times w$*, with* $h = 6$*, and* $w = 5$*. Then, we can deduce that the* relative payload $\alpha = \frac{1}{5}$*, and the* embedding efficiency $e = \frac{\alpha}{d}$*, with* d *being the expected per-element distortion. That element been given by:*

$$d(\alpha) = \frac{E[D(x, Emb(x, m))]}{n}$$

, we can assess our final embedding efficiency e.
Considering we are working with the constant profile, in this case, because we only changed 1 *bit,* $d = \frac{1}{15}$*, and* $e = 3$*.*

Example 18. *Given the syndrome table, and remembering that we store the syndromes of minimum weight:*

s	*e (w < d)*
0 0 0	***0000000***
0 0 1	***0010000***
0 1 0	***0100000***
1 0 0	***1000000***
0 1 1	***0000100***
1 0 1	***0000001***
1 1 0	***0001000***
1 1 1	***0000010***

Let us consider instead the convolutional code, given by:

$$A = (0)\,,\ B = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

$$C = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}, D = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}$$

At step $t = 0$*, we have:* $D(u(0) + e(0)) = m(0)$

At step $t = 1$*, we have:* $D(u(1) + e(1)) = m(1) - CB(u(0) + e(0))$

At step $t = 2$*, we have:* $D(u(2) + e(2)) = m(2) - CB(u(1) + e(1))$

Then, at step $t = 0$*,* $s(0) = (1\,0\,0)$*, and* $e(0) = (1\,0\,0\,0\,0\,0\,0)$*,*
at step $t = 1$*,* $s = (0\,1\,1)$*, and* $e(1) = (0\,0\,0\,0\,1\,0\,0)$
at step $t = 2$*,* $s = (1\,0\,1)$*, and* $e(2) = (0\,0\,0\,0\,0\,0\,1)$
for the same $u = (1\,1\,0\,0\,1\,1\,0,\ 0\,0\,1\,0\,0\,1\,1,\ 1\,0\,0\,1\,0\,1\,0)$*, we can embed* $m = (110,\ 010,\ 011)$ *with the flip pattern:* $e = (1\,0\,0\,0\,0\,0\,0,\ 0\,0\,0\,0\,1\,0\,0,\ 0\,0\,0\,0\,0\,0\,1)$

For this operation, we embedded 9 *bits in a* 21*-length cover sequence by changing only* 3 *bits.*

For this particular case, our submatrix

$$\widehat{\mathbb{H}} = \begin{pmatrix} D \\ CB \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}$$

is of size: $h \times w$*, with* $h = 6$*, and* $w = 7$*. Then, we can deduce that the* relative payload $\alpha = \frac{1}{7}$*, and the* embedding efficiency $e = \frac{\alpha}{d}$*, with* d *being the expected per-element distortion. That element been given by:*

$$d(\alpha) = \frac{E[D(x, Emb(x, m))]}{n}$$

, we can assess our final embedding efficiency e.
Considering we are working with the constant profile, in this case, because we only changed 3 *bit,* $d = \frac{1}{7}$*, and* $e = 1$*.*

D.1 Functions embedding and recovery

In order to define our convolutional code for the proper stegoscheme, according to our protocol, the following necessary condition is requested.

Proposition 19. *Let (A, B, C, D) be a representation of a convolutional code $\mathcal{C}$, with $A \in M_\delta(\mathbb{F})$, $B \in M_{\delta\times k}(\mathbb{F})$, $C \in M_{p\times\delta}(\mathbb{F})$, $D \in M_{p\times k}(\mathbb{F})$ (with $D \neq 0$ and $p = n - k$). Let $p < k$.*
A necessary condition for building a stegoscheme from $\mathcal{C}$ is that rank D *be row maximal.*

D.2 Conditions on the modified subsections of the cover

Here are some conditions applying to the modification of the cover sequence when embedding.

Proposition 20. *Let (A, B, C, D) be a representation of a convolutional code $\mathcal{C}$ for a steganographic scheme S. Then, at each step t of the convolutional sequence, the flipping sequence e introduced for embedding m in u can be given by the formula:*

$$De(t) = m(t) - Du(t) - CB(u(t-1) + e(t-1))$$

Proof. Having that the embedding of m is given by: $m = \widehat{T}_{\ell-1}\big(u + e\big)$, and having: $\widehat{T}_{\ell-1} = \begin{pmatrix} D & & & & \\ CB & D & & & \\ & CB & D & & \\ & & \ddots & \ddots & \\ & & & CB & D \end{pmatrix}$, we can deduce the result. □

D.3 Classical bound of imperceptibility

Proposition 21. *Let (A, B, C, D) be a representation of a convolutional code $\mathcal{C}$ for a steganographic scheme S. Let us consider embedding m in u, with error sequence e. Let D have all of its columns non-zero and distinct.*
Then, for each step t of the convolutional sequence: $\exists\, e(t)$ such that $w(e(t)) \leq d - 1$.

Proof. Considering $\widehat{T}_{\ell-1}$ the control matrix, the embedding of m is given by: $m = \widehat{T}_{\ell-1}\big(u + e\big)$;

at each step t, we have: $De(t) = s(t)$.

Let us consider D_j the columns of D; knowing that d is the minimal number of linearly-dependent columns of D, for all t, for $s(t) \neq 0$, there exists n columns D_j such that: $\sum_{j=1}^{n} D_j = s(t) = De(t) \neq 0$, with $n \leq \tau - 1$; therefore, $w(e(t)) = n \leq \tau - 1$; and for $s(t) = 0$, there is always $e(t) = 0$ which verifies: $w(e(t)) = 0 \leq \tau - 1$.

From there, we can deduce the result. □

This proposition follows from the precedent one.

Proposition 22. *Let us consider a steganographic scheme S given by a convolutional code (A, B, C, D), and functions* emb *and* rec*. Let D have all of its columns non-zero and distinct. Then, within an ℓk-length cover sequence, we can embed at most ℓp-length message by modifying at most $\ell(\tau - 1)$ bits*

D.4 Analysis and results

So, in this work, we suggested a decoding method for convolutional codes under the linear systems theory, codes with the parity-check matrix being $\mathbb{H}$, and that would be of size: $b \times wb$, with b, being the number of copies of $\widehat{\mathbb{H}}$, and w being the number of columns of $\widehat{\mathbb{H}}$. So, we learn from [5], that in order to find a good submatrix, that has embedding algorithm performance close to the bound, the profile is not a necessity. But, we already know from both our embedding and extracting procedures, that for the decoding method, the syndrome approach, the time complexity is linear, due to the linear decoding method, as well as the space complexity that decreases from $2^h.n$ to $2^{h/2}.n$.
Indeed, for the embedding process, we need the coset of errors for each syndrome, but we only need to store the leader, which is the one of minimal weight. So, knowing the length n of the cover vector, h being the constraint height, since: $\widehat{\mathbb{H}}$ is implemented such that:

$$\widehat{\mathbb{H}} = \begin{pmatrix} D \\ CB \end{pmatrix}$$

, then the number of leaders to be stored are $2^{h/2}$, just like the number of syndromes. Which brings the space complexity to a total of $\dfrac{\sqrt{2}^h.n}{8}$ bytes.

VII. CONCLUSIONS

We developed a steganographic model, based on the representation of convolutional codes within linear systems theory. In fact, the idea resides on considering the output-observability matrix, along with the encoding/decoding procedures, used for the convolutional codes. Indeed, the embedding and recovery algorithms inspired by this algebraic computational method, are based upon a novel decoding method, built on the linear systems properties and tools, and the implementation of convolutional codes from the linear systems theory approach. The stegoscheme, that was in order to enable us to implement steganography for time-related transactions, protection of communication during an unspecified time, revealed interesting results, such as linear time complexity, as well as an ability to hide a tremendous amount of information, with very little modification. We can also denote an improvement of space complexity as well, going from $2^h.n$ to $2^{\frac{h}{2}}.n$, in memory space (with h: constraint height, and n: length of cover object), while implementing conditions for maximal bounds , within specified conditions. This work is just a beginning, but shows a lot of possibilities within the subject, possibilities for reduction of both time and space complexity, and expansion of opportunities in steganography itself.

REFERENCES

[1] Z. Bao, Y. Guo, X. Li, Y. Zhang, M. Xu, X. Luo. A robust image steganography based on the concatenated error correction encoder and discrete cosine transform coefficients, Journal of Ambient Intelligence and Humanized Computing 11, 2020, pp.1–13. DOI: 10.1007/s12652-019-01376-1

[2] R, Dariti, El M. Souidi. An application of linear error-block codes in steganography. International Journal of Digital Information and Wireless Communications 1(2), 2012, pp. 426–433.

[3] P. Elias. Coding for noisy channels. IRE Conv. Rec 4, 1955, pp. 37–46.

[4] T. Filler, J. Judas, J. Fridrich. Minimizing Embedding Impact in Steganography using Trellis-Coded Quantization. In Proceedings Volume 7541, Media Forensics and Security II 7541, 2006, pp. 1–11. DOI: 10.1117/12.838002

[5] J. Fridrich. Steganography in Digital Media - Principles, Algorithms, and Applications. Cambridge Univ. Press, 2009, pp. 1–76.

[6] M.I. Garcia-Planas, El M. Soudi, L.E. Um. Convolutional codes under control theory point of view. Analysis of output-observability. Recent Advances in Circuits, Communications & Signal Processing, 2013, pp. 131–137. DOI: 10.1109/JNS3.2013.6595475

[7] M.I. Garcia-Planas, El M. Soudi, L.E. Um. Decoding Algorithm for Convolutional Codes under Linear Systems Point of View. Recent Advances in Circuits, Systems, Signal Processing and Communications, Publisher: Wseas, 2014, pp. 17–24. DOI: 10.1007/978-3-319-55589-8.

[8] M. Hautus. Controllability and observability condition for linear autonomous systems. Proceedings of Nedderlandse Akademie voor Wetenschappen, Series A 72, 1969, pp. 443–448.

[9] R. Johannesson, K.Sh. Zigangirov. Fundamentals of Convolutional Coding. New York: IEEE Press, 1999, pp. 163–237.

[10] H. Jouhari. New Steganographic Schemes using Binary and Quaternary Codes. Ph.D thesis, Universit ´e Mohammed V-Agdal, Morocco, 2013, pp. 33–44.

[11] H. Jouhari, El M. Souidi. Application of Cyclic Codes over Z4 in Steganography. Journal of Applied Mathematical Sciences 6(139), 2012, pp 6911–6925.

[12] R.E. Kalman. Contribution to the theory of optimal control. Bolet´ın de de la Sociedad Matem´atica Mexicana 5, 1960, pp.102–119. DOI: 10.4236/ojop.2013.23010

[13] E. Martinez-Moro. Algebraic Geometry Modeling in Information Theory. World Scientific, 2012, pp. 87–92.

[14] M.O. Medeni, El M. Souidi, A Novel Steganographic Protocol from Error-correcting Codes. Journal of Information Hiding and Multimedia Signal Processing 1, 2010.

[15] C. Munuera. Steganography From A Coding Theory Point Of View. Algebraic Geometry Modeling in Information Theory, World Scientific, 2013, pp. 83–128.

[16] J. Rosenthal. Connections between linear systems and convolutional codes. Springer, (2000). DOI: 10.1007/978-1-4613-0165-3□2

[17] J. Rosenthal, E,V, York. BCH convolutional codes. IEEE Trans. Inform. Theory 45(6), 1999, pp. 1833–1844.

[18] Sheelu, B. Ahuja. An Overview of Steganography. IOSR Journal of Computer Engineering (IOSRJCE) 11(1), 2013, pp. 15–19. DOI: 10.9790/0661- 1111519

[19] C.P. Sumathi, T. Santanam, G. Umamaheswari. A Study of Various Steganographic Techniques Used for Information Hiding. International Journal of Computer Science & Engineering Survey (IJCSES) 4(6), 2013, pp. 9–12.

[20] L.E. Um. A contribution to the theory of convolutional codes from systems theory piont of view. Doctoral Thesis, UPC, Departament de Matem`atica Aplicada I, 2015.

[21] L.E. Um, M.I. Garcia-Planas. Applying the theory of convolutional codes to the study of steganography. Proceedings of International Conference on Applied Linear Algebra, Probability and Statistics, ALAPS 2020, pp 55–56.

[22] A. Westfield. F5 - A steganographic algorithm. Proc. of the Information Hiding 4th International Workshop 2137, 2001, pp. 289–302. DOI: 10:1007=3-540-45496-921

Creative Commons Attribution License 4.0 (Attribution 4.0 International, CC BY 4.0)