High Performance Steganographic Scheme Applying Time-Varying Convolutional Embedding Codes

CHI-YUAN LIN[1] and JYUN-JIE WANG[2]
Department of Computer Science and Information Engineering,
National Chin Yi University of Technology, Taichung 41170, Taiwan, ROC
Email: [1]chiyuan@ncut.edu.tw,[2]jjwang@ncut.edu.tw

Abstract: - A matrix embedding code was developed as a commonly used steganographic technique in which a parity-check matrix is used to perform embedding. However, a drawback of high decoding complexity for linear block codes by using the maximum-likelihood algorithm is unrealistic. This paper proposes a simple and effective trellis embedding scheme for binary messages. Compared with a matrix embedding algorithm that uses linear block codes, the proposed scheme is more appropriate for embedding messages in the case of linear block codes with a long length. The proposed algorithm uses time-varying convolutional codes as the embedding method and yields a favorable structure of time-varying convolutional codes for steganography. The proposed method employs maximum-likelihood decoding based on trellis construction to identify the coset leader of convolutional codes for large payloads. The experimental results show that the embedding efficiency of the proposed scheme is substantially superior to that of the scheme using linear block codes.

Key-Words: - Steganography, matrix embedding, embedding efficiency, linear block code, convolutional codes

1 Introduction

As public network communication progresses increasingly, a large amount of data must be transmitted using numerous reliable approaches, one of which is steganography. The main requirements for a steganographic scheme are security and embedding efficiency. Embedding efficiency directly influences security; thus, steganography emphasizes its embedding efficiency. To obtain high security in communication, the embedding efficiency must be high; in other words, the average number of embedded bits per change is low. One effective steganographic technique involves using matrix embedding (ME) codes [1],[4]. Constructing structured codes with an embedding efficiency close to the theoretical bound is a crucial open problem that entails two concerns. First, embedding schemes require structured codes of a sufficiently long length that possess an excellent parity-check matrix or generator matrix. Second, structured codes are more computationally efficient, and efficient encoding and decoding procedures are developed on the basis of structured codes.

Through coding theory, linear block codes provide a general approach to improving embedding efficiency in steganography. This technique was first proposed by Crandall [2] and Bierbrauer [3] in early 1998. Long low-density generator-matrix (LDGM) embedding codes [5] with iterative decoding based on the bias propagation algorithm have been presented. In addition to LDGM embedding codes, constructed codes with a fast decoding algorithm have been developed [6],[7],[8], and Filler, Judas, and Fridrich [9] proposed a practical embedding scheme using syndrome-trellis codes (STCs), which are represented in a dual domain of convolutional codes. The method is employed to perform embedding procedures by using a trellis structure based on a parity-check matrix rather than a generator matrix. This study developed a new trellis-based embedding scheme in which the complexity is of linear time and space for steganography. This paper proposes an embedding technique designated as the time-varying convolutional codes because it is developed on the basis of a convolutional code. In contrast to the ME codes, the time-varying convolutional codes are embedded with a time-

varying trellis structure and then decoded using a Viterbi algorithm, one of the maximum-likelihood (ML) algorithms. Because a trellis structure is a time variant, the decoding complexity varies linearly, not exponentially as in the ML approach, with linear blocks when the Viterbi algorithm is performed.

The remainder of this paper is organized as follows: Section 2 briefly discusses the theory limit of binary data hiding and quantization for linear codes. The major work on the proposed optimal time-varying convolutional embedding codes algorithm is described in Section 3, and Section 4 provides experimental results and constructive discussions. Finally, Section 5 concludes this paper.

2 Performance of Linear Codes

The goal of the binary embedding scheme is to quantize a source subject to a theoretical bound on the amount of distortion. Fig. 1 illustrates an embedding model and extracting model.

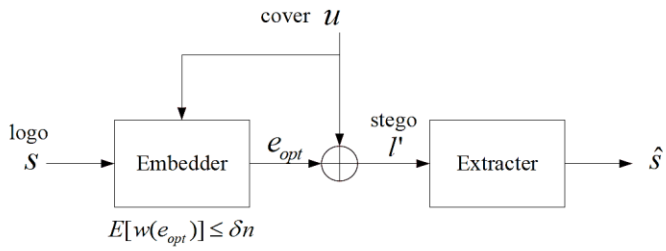

Fig. 1 Embedding model and extracting model.

Under the assumption that a logo $s \in \{0,1\}^m$ embedded into a cover $u \in \{0,1\}^n$ is transmitted to the receiver, then the optimal stego $l' = u + e_{opt}$ is provided by an embedder; that is, a message l', modified from u, corresponding to the syndrome s. Given an optimal toggle e_{opt}, a symmetric Bernoulli source adds to it some cover u, that is, $l' = u + e_{opt}$. Even though the embedder knows the cover u, it cannot simply cancel this known interference because of the constraint on the average number of 1's, which cannot exceed $n\delta$, where $0 \le \delta \le 1/2$. In this study, the optimal or minimum quantized error was defined as $e_{opt} = d_H(l', u)$, where $d_H(\cdot)$ denotes a Hamming distance between a setgo l' and cover u. The rate-distortion function $R(\delta) = 1 - h(\delta)$ is requested to be achieved, where δ denotes the bound of average distortion and $h(d) = d\log_2(1/d) + (1-d)\log_2(1-(1-d))$ denotes a binary entropy function, by an (n, k) linear code C with a code rate $R_c = k/n \approx R(\delta)$. Theoretically, the codeword of a linear code C can be regarded as a quantized message set $C = \{\hat{u}\}$, with δ as the average distance between an arbitrary cover set U an upper bound of the embedding capacity is then determined as

$$\max_{E[d(C,U)] \le n\delta} h(C \mid U) = h(\delta). \qquad (1)$$

In case a well-designed linear code exists, the aforementioned theoretical upper bound can be approached using an associated embedder. For any $0 \le \delta \le 1/2$, $\varepsilon > 0$, and sufficiently high n, there exists an (n, k) embedding code of rate $R_c < R(\delta) + \varepsilon$ that satisfies

$$\frac{1}{n}E[d(u,\hat{u})] = \frac{1}{n}E[w(e_{opt})] < \delta + \varepsilon. \qquad (2)$$

The remaining major concern entails seeking a parity-check matrix with a well-behaved (n, k) linear code and a code rate $R_c = k/n$. Furthermore, with an embedding rate requested in such a linear code C, the aforementioned equation can then be rewritten as

$$h(\delta) \approx 1 - k/n = m/n. \qquad (3)$$

Because $m \approx nh(\delta)$, 2^m cosets are employed to reach aforementioned R_m. Given an embedding rate for an (n, k) embedding code, the minimum average distortion is up to

$$\delta = h^{-1}(m/n) = h^{-1}(R_m) \qquad (4)$$

where $h^{-1}(*)$ is the inverse function of the binary entropy function h.

For a symmetric Bernoulli source and a source sequence of n bits, the average quantization distortion per bit is defined as

$$d_{avg} = \frac{D}{n} = \frac{E[d_H(\hat{u}, u)]}{n} \qquad (5)$$

where D is the average Hamming distortion between a quantized codeword $\hat{u} \in C$ and arbitrary vector u. The lower bound δ of each bit average distortion in blocks can be written as $d_{avg} \geq \delta$. When the binary data embedding of a sequence of length n bits is performed, the embedding efficiency is defined as

$$\eta = \frac{R_m}{d_{ave}} = \frac{m}{D}. \qquad (6)$$

For such a linear code C, the embedding efficiency between both the efficiency bound and ML-detected algorithms can be related as

$$\frac{m}{nh^{-1}(R_m)} \geq \eta \Rightarrow D_{opt} \geq nh^{-1}(R_m), \qquad (7)$$

where D_{opt} represents the distortion level estimated in the ML-detected algorithms.

For a given $(n, k, \lambda_{\min})$ linear block code C with an embedding rate $R_m = (n-k)/n$, the packing radius of spheres is obtained using

$$t = \left\lfloor \frac{\lambda_{\min} - 1}{2} \right\rfloor. \qquad (8)$$

Using (8), the number of Voronoi set E_0 of linear block codes can be divided as

$$2^{nR_m} = V_2(n, t) + \Delta(n, t), \qquad (9)$$

where $V_2(n, t) = \sum_{i=0}^{t} \binom{n}{i}$ and $\Delta(n, t)$ are the Hamming spheres of radius t and residue without the Hamming sphere $V_2(n,t)$. Furthermore, the covering radius R of linear block codes was defined as

$$t_R \triangleq \max_{u \in F_2^n} \min_{c \in C} d_H(u,c) \qquad (10)$$

An all-zero codeword in C was adopted to show that spheres for some various radii exist. Fig. 2 illustrates the spheres.

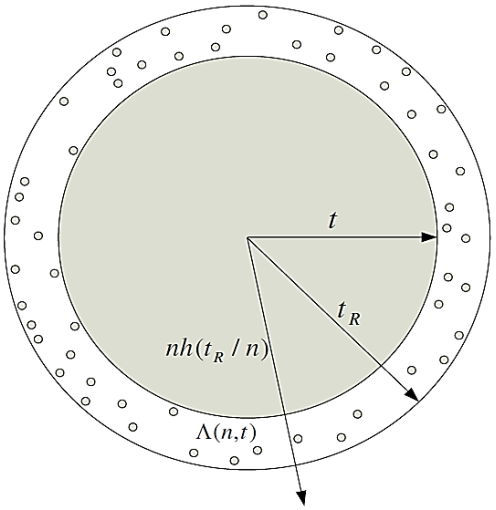

Fig. 2 Covering radius for block codes.

By applying information theory, the inequality of the spheres can be obtained using

$$2^{nh(t_R/n)} \geq V_2(n, R_t) \geq 2^{nR_m} \geq V_2(n, t) \qquad (11)$$

; that is,

$$h(t_R/n) \geq \frac{\log_2 V_2(n, R_t)}{n} \geq R_m \geq \frac{\log_2 V_2(n, t)}{n}. \qquad (12)$$

Finally, the precise upper and lower bounds were obtained using

$$\frac{\log_2 V_2(n, R_t)}{n} \geq R_m \geq \frac{\log_2 V_2(n, t)}{n}, \qquad (13)$$

where $V_2(n, t_R)$ is the Hamming sphere of radius t_R. Using (13) divided by the average changes of embedding d_{avg} to determines the bound of embedding efficiency as

$$\frac{\log_2 V_2(n, R_t)}{d_{avg} n} \geq \eta \geq \frac{\log_2 V_2(n, t)}{d_{avg} n}. \qquad (14)$$

For a linear block code with sufficiently long length, d_{avg} is difficult to obtain. However, the average changes of embedding of sphere $V_2(n, t)$ and $V_2(n, t_R)$ can be effortless determined. To efficiently measure the embedding efficiency for a certain linear block code C, a more precise efficiency bound for a sufficiently long linear block code is necessary. Subsequently, a realistic measure bound of embedding efficiency is introduced.

The average embedding changes D_{avg}, the changes of embedding per n -bit block, by using linear block codes C. are discussed. A linear embedded code C can control the number of error bits between t and t_R. Upon finding all the $\binom{n}{t}$ sequences in the Voronoi set E_0 of coset leader, the rest are of the weight between t and t_R. However, such weight of set $\Delta(n,t)$ is assumed to be difficult to obtain in this study. The average changes of embedding within the Voronoi set E_0 is bounded as follows:

$$D_{opt} = \frac{\sum_{i=0}^{t} i\binom{n}{i} + \sigma\left(2^m - \sum_{i=0}^{t}\binom{n}{i}\right)}{2^m}. \quad (15)$$

The average sequence distortion of either optimal decoding, or ML decoding, of any linear code is not lower than D_{opt}. Assume D_t and D_{t_R} are the average changes of embedding corresponding to sphere $V_2(n,t)$ and $V_2(n,t_R)$. For a sphere of radius t', the average density of sphere is expressed as

$$D_{t'} = \sum_{i=0}^{t'} i\binom{n}{i} / \sum_{i=0}^{t'}\binom{n}{i}. \quad (16)$$

The average changes of embedding within E_0 are written as follows:

$$D_{t_R} \geq D_{opt} \geq D_t, \quad (17)$$

where D_{opt} is the average changes per block for the matrix embedding scheme using the ML decoding strategy. For a matrix embedding, D_{opt} is approached using a (n,k) linear code with the ML decoding strategy. For the statistical embedding changes per block, the embedded logo vector l, derived from s_l, is uniformly picked and then the toggle vector x is obtained by cover u subtracting l. In addition, the expected number $E[w(e_{opt})]$ of embedding changes is uniformly distributed within F_q^m with respect to the number of coset leaders of various weights. In other words, the toggle x is uniform in each coset and the coset leader, obtained using $e_{opt} = f(Hx^T)$. Finally, the average embedding changes per block are expressed as

$$\begin{aligned} D_{avg} \approx D_{opt} &= E[w_H(f(Hx^T))] \\ &= \frac{\sum_{x\in F_q^n} w_H(f(Hx^T))}{q^n} \\ &= \frac{\sum_{i=0}^{q^m-1} w_H(e_{opt,i}) q^k}{q^n} \\ &= \frac{\sum_{i=0}^{q^m-1} w_H(e_{opt,i})}{q^m}. \end{aligned} \quad (18)$$

Finally, inverse (16) and multiply by R_m as

$$\frac{R_m}{D_{t_R}}(\eta_{low}) \leq \eta \leq \frac{R_m}{D_t}(\eta_{up}). \quad (19)$$

3 Optimal Quantizing by Using Time-Varying Convolutional Codes

3.1. Quantizing by using convolutional codes

As shown in Fig. 3, a module operation performs the quantization of an arbitrary vector $u \in F_2^n$ by a linear block code C.

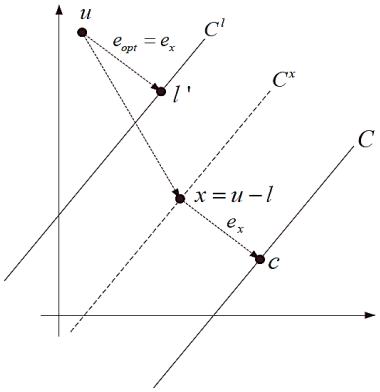

Fig. 3 u Mod C operation.

This section details the quantization under the condition that C^l, the coset of the code C, replaces C here. Considering a coset C^l of C with a corresponding syndrome s_l, an arbitrary cover vector $u \in C^l$, is then quantized by C^l. It is intended to locate $l' \in C^l$, the vector closest to u, and the minimal error between l' and u is represented as

$$\begin{aligned} e_{opt} &= u \bmod C^l \\ &= u + l \bmod (C^l + l) \\ &= x \bmod C \\ &= f(Hx^T). \end{aligned} \quad (20)$$

In case l is the coset leader within C^l, the linear code C is obtained using the sum of the coset leader l and C^l, the solution to the equation $l = H^{-1}s_l$, but in most cases, the coset leader is not required, which can be instead determined using an arbitrary vector defined in C^l. Ultimately, e_{opt} is obtained by performing a decoding function.

An arbitrary host vector $u \in C^u \subset F_2^n$ exists within the coset C^u. The vector is referred to as the logo vector, a known binary vector s_l of length $n-k$ bits that is intended for embedding. The coset leader $e_{opt} \in C^x \subset F_2^n$ must be located within a set C^x, closest to u, with logo vector s_l, discovered. The syndrome s_x is then determined using $H(l+u)^T$, where $l = H^{-1}s_l$. From the perspective of quantization, the coset leader e_{opt} can be discovered through a module operation, expressed as

$$\begin{aligned} e_{opt} &= u \ \bmod C^l \\ &= l + u \ \bmod C \\ &= x \ \bmod C. \end{aligned} \quad (21)$$

Suppose that a sequence $x \in C^x$ exists that satisfies $s_x = Hx$, and represents a coset C^x of the code C. x with the minimal weighting is intended to be sought; that is, e_{opt}, which is expressed as

$$e_{opt} = x + \arg\min_{c \in C} d_H(c, x). \quad (22)$$

Once discovered, the coset leader $e_{opt} \in C^x$ is added to the host as u, $l' = u + e_{opt}$. Essentially, $l' \in C^l \subset F_2^n$ is the sequence closest to the sequence u within F_2^n dimensional space and contains the logo sequence s_l.

Finally, the secret message s_l is extracted as $s_l = H(l')^T$ at the receiver, according to the aforementioned embedding procedures and then presented as the following algorithm:

==========

Algorithm Optimal embedding algorithm:

Encoder: Given a symbol s_l and a host vector u, a vector l', closet to the vector u corresponding to the syndrome s_l, is located as follows:

1. In syndrome domain, derived from s_l, the vector l in C^l, is added to s_u to obtain s_x.
2. The vector x is then decoded using the ML algorithm into a codeword c as follows:
   $$\hat{c} = \arg\min_{c \in C} d(c, x).$$
   and adding x to $\hat{c}$ yields e_{opt}.
3. Or, to solve the linear equations, $H_x = s_x$ yields e_{opt}.
4. The output l' is obtained by adding e_{opt} to u.
   $$l' = u + e_{opt}$$

Decoder: Recover the logo s_l by y and H. (i.e., the embedded data is then extracted by performing

$$s_l = Hl')$$

==========

A notation must be defined to describe the embedding of a convolutional code-based algorithm. Assume that a convolutional code is a nonsystematic generator matrix, which can translate into a systematic generator matrix by using elementary row operations. Alternatively, the convolutional code can be reached in a systematic recursive form. In this study, a convolutional code was used to embed the binary message as follows:

A convolutional code Λ with a generator matrix $G(D)$ is defined as

$$\Lambda = \{c(D) = v(D)G(D)\}, \quad (23)$$

where information sequence is $v(D) \in F_2^k(D)$ and codeword sequence is $c(D) \in F_2^n(D)$. The codeword $c(D) \in \Lambda$, is closest to a random binary sequence $u(D)$ with respect to the Hamming distance over a binary symmetric source. The convolutional code Λ was used to generate the minimum error sequence $e_u(D)$ from a quantization perspective as follows:

$$\begin{aligned} e_u(D) &= \arg \min_{c(D)\in\Lambda} d(c(D), u(D)) \\ &= u(D) - Q(u(D)) \\ &\triangleq u(D) \bmod \Lambda \end{aligned} \quad (24)$$

where the $Q(x(D))$ is a quantizer as follows:

$$Q(u(D)) = c'(D) \in \Lambda, \quad (25)$$

$c(D) \in \Lambda$ exists and

$$d_H(u(D)) - c'(D)) \le d_H(u(D) - c(D)). \quad (26)$$

The nearest neighbor quantizer $Q(\cdot)$, which was interpreted as the minimal error vector $e_u(D)$ in quantizing $u(D)$ by Λ and $Q(\cdot)$, can be realized using the Viterbi algorithm for convolutional codes with a trellis structure. Finally, the Voronoi cell of Λ was defined as the set

$$V_0 = \{e_{opt}(D)\} = \{u(D) : Q(u(D)) = 0\}. \quad (27)$$

Consider the use of algebraic for a coset code of a convolutional code. Assume a shifted coset code Λ^l of a convolutional code Λ, where Λ^l is defined as the sum of Λ and a minimal error sequence $e_{opt}(D)$. Subsequently, by using Λ^l, an arbitrary binary sequence $u(D)$ is quantized by coset code Λ^l as

$$\begin{aligned} e_{opt}(D) &= u(D) \bmod \Lambda^l \\ &= u(D) - l(D) \bmod (\Lambda^l - l(D)) \\ &= x(D) \bmod \Lambda \\ &= x(D) - Q(x(D)), \end{aligned} \quad (28)$$

where the shift sequence $l(D) \in \Lambda^l$; that is, $l(D) = c(D) + e_{opt}(D)$ and $e_{opt}(D)$ denote the error sequence or coset leader sequence in quantizing toggle sequence $x(D)$ by Λ.

Assume that cover sequence $u(D)$ is uniformly distributed in $F_2^N(D)$; subsequently, the toggle sequence $x(D)$, which is obtained by subtracting message sequence $l(D)$ from cover sequence $u(D)$, is also uniformly distributed. The minimal distance sequence $e_{opt}(D)$ between cover sequence $u(D)$ and message sequence l is equal to (30). By quantizing a random binary sequence $x(D)$ by Λ^l, an average quantized distortion level is represented as

$$\begin{aligned} D_{avg} &= E[w(x(D)Q(x(D)))] \\ &= E[w(e_{opt}(D))]. \end{aligned} \quad (29)$$

Similar to the linear block codes, the optimal toggle vector must be determined. The task can be performed using systematic convolutional codes. A simpler method relative to the systematic coding approach is message embedding by using linear block codes requiring a coset vector l associated with s_l. The method in which the toggle vector was obtained in a systematic block code binary embedding was applied to the systematic convolutional code binary embedding. The embedding procedure for systematic convolutional codes is presented as follows:

With a message syndrome sequence $s_l(D)$ of length $N(nR_c)$, determining the sequence $l(D) \in \Lambda^l$ of length Nn is necessary with the syndrome s_l as the linear codes. For a special $(n,1)$ systematic convolutional code case, a generator matrix $G_s(D)$, is defined as

$$G_s(D) = [1 \;\; g_1(D) g_2(D) ... g_m(D)], \quad (30)$$

where $m = n - 1$. The transposition of $G(D)$ yields

$$H_s(D) = \begin{bmatrix} g_1(D) & 1 & 0 & \cdots & 0 \\ \vdots & 0 & 1 & \cdots & 0 \\ g_m(D) & \vdots & & \ddots & 1 \end{bmatrix}, \quad (31)$$

where $H_s(D)$ is a $m \times n$ matrix and embedded sequence $s_l(D) \in F_2^m(D)$ is derived as

$$H_s(D) l(D)^T = s_l(D). \quad (32)$$

It is necessary to solve

$$l(D)^T = (H_s(D))^{-1} s_l(D)^T. \quad (33)$$

In general, this equation is highly complex to solve. Because of the systematic encoder, $l(D) = [0...0 \ \ s_l(D)]$ of size $1 \times n$ can be solved. In addition, the toggle sequence $x(D)$ is obtained by $u(D)$ subtraction to $l(D)$. The embedder quantizes the arbitrary toggle sequence $x(D)$ to generate the optimal stego sequence $l'(D)$ as

$$\begin{aligned} l'(D) &= u(D) - (x(D) \bmod \Lambda(D)) \\ &= u(D) - (x(D) - Q(x(D))). \end{aligned} \quad (34)$$

Finally, the sequence $l'(D)$, closest to the sequence $u(D)$, corresponding to the syndrome $s_l(D)$ is derived as

$$l'(D) = u(D) - e_{opt}(D). \quad (35)$$

At the receiver, the message sequence $s_l(D)$ is extracted as $s_l(D) = H_s(D)(l'(D))^T$.

The following example is based on a systematic convolutional code to illustrate the nested convolutional embedding algorithm and describe the embedding procedure. Consider an embedded message sequence $s_l = [1,1,1,1,1]$ and cover sequence $u = [11,01,11,01,11]$. Because the systematic convolutional codes, specified by parity-check matrix $H_s(D) = [1 + D^2 \ \ 1]$, are used, the solution $l(D)$ corresponding to $H_s(D)l(D) = s_l(D)$ was easily obtained. Subsequently, a systematic convolutional code binary embedding was performed in the code domain. Under the assumption that $s_l = (l_1, l_2, ..., l_M)$ is the symbol intended for embedding, the vector $l = (0, l_1, 0, l_2, ..., 0, l_M)$ represents a sequence, that is, a (2,1) systematic convolutional code with the syndrome s_l . $l = [0,1,0,1,0,1,0,1,0,1]$ and the toggle sequence x corresponding to s_x and within the coset C^x can be easily obtained.

$$\begin{aligned} x &= u - l \\ &= [1,1,0,1,1,1,0,1,1,1] - [0,1,0,1,0,1,0,1,0,1] \\ &= [1,0,0,0,1,0,0,0,1,0]. \end{aligned}$$

The optimal toggle sequence $e_{opt}(D)$, corresponding to syndrome $s_x(D)$, can be discovered by performing Viterbi decoding of $x(D)$, as follows:

$$\begin{aligned} e_{opt}(D) &= x(D) - Q(x(D)) \\ &= [1,1,0,0,1,0,0,0,1,0] - [1,0,0,0,1,0,0,0,1,0] \\ &= [0,1,0,0,0,0,0,0,0,0], \end{aligned}$$

where Viter $(\cdot)$ is a Viterbi decoding function. The procedure for finding an optimal toggle sequence $e_{opt}(D)$. Finally, the stego sequence was obtained using

$$\begin{aligned} l'(D) &= u(D) - e_{opt}(D) \\ &= [11,01,11,01,11] + [0,1,0,0,0,0,0,0,0,0] \\ &= [10,01,11,01,11]. \end{aligned}$$

In the receiver, the message sequence $s_l(D)$ was reconstructed as

$$\begin{aligned} s_l(D) &= H_s(D) l'(D) \\ &= \begin{bmatrix} 11 & & & & \\ 00 & 11 & & & \\ 10 & 00 & 11 & & \\ & 10 & 00 & 11 & \\ & & 10 & 00 & 11 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \end{bmatrix} \\ &= [1,1,1,1,1]^T. \end{aligned}$$

A crucial factor of convolutional embedding codes is how the optimal generator matrix for large payloads is found. The subsequent section provides numerous optimal generator matrices by simulation.

3.2. Optimal time-varying convolutional codes

An algebraic construction for binary embedding scheme is presented as follows: The memory M parity-check matrix of a time-varying, periodical, rate $R = k/n$ binary convolutional code C is the semi-infinite matrix expressed as

$$H^T = \begin{pmatrix} H_0^{(0)} & H_1^{(1)} & \cdots & H_{t-1}^{(M-1)} & H_t^{(M)} \\ & H_1^{(0)} & \cdots & H_{t-1}^{(M-2)} & H_t^{(M-1)} \\ & & \ddots & \vdots & \vdots \end{pmatrix}, \quad (36)$$

where $H_t^{(m)}$, $m = 0,1,...,M$, $t = 0,1,...$ are $n \times (n-k)$ binary submatrices , $H_t^{(0)}$ should have a full rank and $H_t^{(m)} = H_{t+T}^{(m)}$ for all t, and T is the period of the code. Let $u_{[0,n]} = (u_0, u_1, ..., u_n)$ be a segment of an information sequence and $v_{[0,n]} = (v_{0,1}, v_{0,2}, v_{1,1}, v_{1,2}, ..., v_{n,1}, v_{n,2},)$ be the corresponding segment of the encoded sequence, which satisfies the equation

$$v_{[0,n]} H_{[0,n]}^T = 0, \quad n = 0,1,\cdots. \quad (37)$$

Because $h_1^{(0)}(n) = 1$ and $h_2^{(0)}(n) = 1$,

$$v_{n,1} = u_n \quad (38)$$

and

$$v_{n,2} = \sum_{i=0}^{M} h_1^{(i)}(n) u_{n-i,1} + \sum_{i=0}^{M} h_2^{(i)}(n) u_{n-i,2}. \quad (39)$$

The realization can be implemented as shown in Fig. 4.

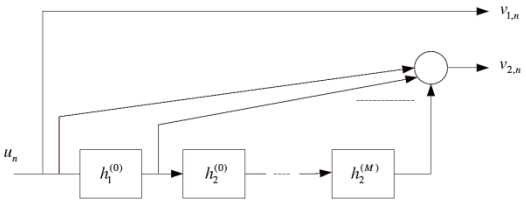

Fig. 4 Time-varying convolutional code encoder.

Generally, the rate $R = k/n$ encoder can be implemented using n length $M+1$ shift registers in parallel and time-varying connections from the register stages to the modulo 2 adders. The quantizer is used to generate the quantized error e_u from the module operation. Certainly, given a linear code C , the codeword $\hat{u} \in C$ is closest to a random binary sequence u with respect to the Hamming distance over a symmetric Bernoulli source. The problem is formulated as follows:

$$\begin{aligned} e_u &= f(s_u) \\ &= f(Hu^T) \\ &= \arg\min_{\hat{u} \in C} d(\hat{u}, u) - u \\ &= Q_{opt}(u) - u, \end{aligned} \quad (40)$$

where $f(\cdot)$ is a ML decoding function and $Q_{opt}(\cdot)$ denotes an optimal quantization function, which is generated using ML decoding of a linear block code.

4 Simulation Results

The simulation was performed to address the problem of constructing a (2,1) favorable time-varying convolutional embedding family of codes with Viterbi decoding. The method presented in Section 3 was used to embed the binary messages and compare embedding efficiency with other embedding algorithms, including time-invarying convolutional codes, BCH codes, simple codes, and Golay codes [4]. The following experiments were simulated on code length $N = 2000$: uniform logo sequence and cover sequence. To obtain the optimal embedding efficiency for time-varying convolutional codes, the parameter of the generator matrix of that was simulated using a computer, as shown in Table 1. Table 1 shows a comparison of parameters of generator matrix in memory $m = 2,3,4$ for time-variant and time-invariant convolutional codes. Table 1 presents the design of an optimal generator of time-variant and time-invariant convolutional codes. The embedding efficiency of these optimal designs is illustrated in Fig. 5.

Some families of block embedding codes as a function of the inverse embedding rate are shown in Fig. 5. In the embedding efficiency corresponding to the approximate inverse embedding rate, the time-varying convolutional embedding codes exhibited superior embedding efficiency, and the trend of a high inverse embedding rate was achieved. By using a full search, some optimal systematic time-varying convolutional embedding codes were found. The time-varying period was 2, as shown in Table 1 and Fig. 5.

5 Conclusion

Proposed in this work is an alternative to a binary embedding algorithm, that is, time-varying convolutional embedding codes. In comparison with an ME codes, it demonstrates a double advantage of (i) being able to perform the ML decoding in the case of a sufficiently large code, i.e. a superior embedding efficiency and (ii) an easily alterable embedding rate to meet various application requirements. The proposed method renders an (2,1) time-varying convolutional code and embedding rate up to $R_m = 0.5$ to perform an embedding task. A good source code built upon the time varying convolutional code structure will be addressed in the near future.

Acknowledgment

This study was supported by the Ministry of Science and Technology, Taiwan, R.O.C. under contract 103-2221-E-167-012.

References

[1] P. Moulin and R. Koetter, "Data-hiding codes," Proc. IEEE, vol. 93, no. 12, pp. 2083-2126, Dec. 2005.

[2] R. Crandall. Some notes on steganography. Steganography Mailing List, available from http://os.inf.tu-dresden.de/westfeld/crandall.pdf, 1998.

[3] J. Bierbrauer, On Crandall's Problem [Online]. Available: http://www.ws.binghamton.edu/fridrich/covcodes.pdf 1998.

[4] J. Fridrich and D. Soukal, "Matrix embedding for large payloads," IEEE Trans. Information Forensics and Security, vol. 1, pp. 390-395, 2006.

[5] M. J. Wainwright, "Sparse Graph Codes for Side Information and Binning," IEEE Signal Process. Mag., vol. 24, no. 5, pp. 47-57, Sep. 2007.

[6] J. J. Wang and H. Chen, "A Suboptimal Embedding Algorithm With Low Complexity for Binary Data Hiding," IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pp. 165-168, June 2012.

[7] J. J. Wang, H. Chen, C. Y. Lin, and T. Y. Yang, "An embedding strategy for large payload using convolutional embedding codes," IEEE International Conference on ITS Telecommunications (ITST), pp. 365-369, Nov. 2012.

[8] J. J. Wang, H. Chen, and C. Y. Lin, "An Adaptive Matrix Embedding Technique for Binary Hiding With an Efficient LIAE Algorithm," WSEAS Transactions on Signal Processing, Issue 2, Volume 8, April 2012.

[9] T. Filler, J. Judas, and J. Fridrich, "Minimizing additive distortion in steganography using syndrome-trellis codes," IEEE Trans. Information Forensics and Security, vol. 6, pp. 920-935,2011.

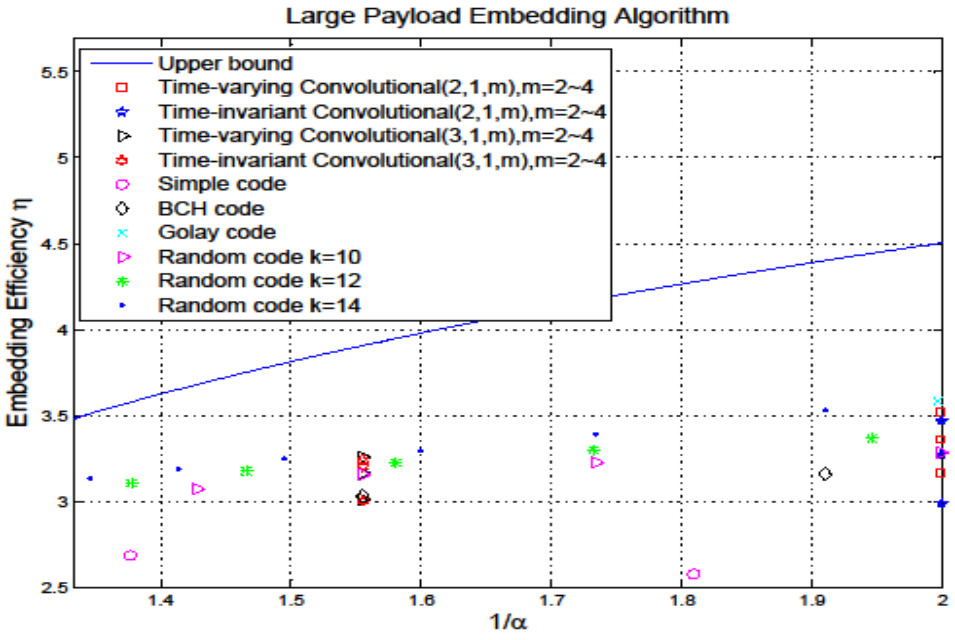

Fig. 5 Some optimal systematic time-varying convolutional embedding codes.

Table 1 The optimal generators of time-varying and time-invariant convolutional code with systematic form were found using a computer search (TI and TV denote time-invarying convolutional code and time-varying convolutional code, respectively).

Parameters / Codes	Period T	Generator G	Embedding efficiency η
TI (r=1/2,L=3)	1	[111]	2.98
TI (r=1/2,L=4)	1	[1111]	3.25
TI (r=1/2,L=5)	1	[10111]	3.47
TI (r=1/3,L=3)	1	[110,101]	3.01
TI (r=1/3,L=4)	1	[1011,1101]	3.23
TI (r=1/3,L=5)	1	[10101,10011]	3.25
TV(r=1/2,L=3)	2	[111][101]	3.21
TV(r=1/2,L=4)	2	[1101][1111]	3.32
TV(r=1/2,L=5)	2	[11011][11111]	3.52
TV(r=1/3,L=3)	2	[111101][111011]	3.03
TV(r=1/3,L=4)	2	[1111,1001][1101,1111]	3.22
TV(r=1/3,L=5)	2	[11111,11011][11011,11101]	3.26