mathematics

MDPI

Article

An Embedding Strategy Using Q-Ary Convolutional Codes for Large and Small Payloads

Jyun-Jie Wang [1], Chi-Yuan Lin [1,*], Sheng-Chih Yang [1], Hsi-Yuan Chang [2] and Yin-Chen Lin [3]

[1] Department of Computer Science and Information Engineering, National Chin-Yi University of Technology, Taichung 411030, Taiwan; jjwang@ncut.edu.tw (J.-J.W.); scyang@ncut.edu.tw (S.-C.Y.)

[2] Department of Electrical Engineering, Institute of Computer and Communication Engineering, National Cheng Kung University, Tainan 701, Taiwan; hi168.hi168@yahoo.com.tw

[3] Ph. D. Program, Prospective Technology of Electrical Engineering and Computer Science, National Chin-Yi University of Technology, Taichung 411030, Taiwan; sla9t2003@student.ncut.edu.tw

* Correspondence: chiyuan@ncut.edu.tw

Abstract: Matrix embedding (ME) code is a commonly used steganography technique, which uses linear block codes to improve embedding efficiency. However, its main disadvantage is the inability to perform maximum likelihood decoding due to the high complexity of decoding large ME codes. As such, it is difficult to improve the embedding efficiency. The proposed q-ary embedding code can provide excellent embedding efficiency and is suitable for various embedding rates (large and small payloads). This article discusses that by using perforation technology, a convolutional code with a high embedding rate can be easily converted into a convolutional code with a low embedding rate. By keeping the embedding rate of the (2, 1) convolutional code unchanged, convolutional codes with different embedding rates can be designed through puncturing.

Keywords: q-ary codes; matrix embedding; optimal design; maximum decoding; convolutional codes

check for updates

Citation: Wang, J.-J.; Lin, C.-Y.; Yang, S.-C.; Chang, H.-Y.; Lin, Y.-C. An Embedding Strategy Using Q-Ary Convolutional Codes for Large and Small Payloads. *Mathematics* **2021**, *9*, 1577. https://doi.org/10.3390/math9131577

Academic Editors: Her-Terng Yau, Bo-Lin Jian and Junseok Kim

Received: 26 April 2021
Accepted: 29 June 2021
Published: 4 July 2021

1. Introduction

Among the numerous steganography techniques that have been developed, matrix embedding (ME) [1,2] provides high undetectability and embedding efficiency, which result in efficient steganographic security. Steganography refers to embedding data to conceal objects such as images, videos, or audio. In steganography, the covered object is modified to obtain a stego.

Numerous ME codes based on covering codes [3–6] have been developed because they exhibit high embedding efficiency due to their favorable structural characteristics, such as excellent weight distribution of the coset leaders of linear codes. In [5], several coverage code series are constructed using factorized block-by-block direct sum (BDS). BDS(6) and BDS(8) provided the highest embedding efficiency. The use of nonlinear covering codes considerably improved efficiency. Fridrich et al. [7] proposed an ME-based embedding technique that comprises two types of linear block codes, namely simplex codes and a random code. The technique exhibited high efficiency for large payloads [7], which resulted in superior steganographic security. Furthermore, they use structured simple codes (including decoding by using fast Hadamard decoding) to obtain effective ME codes and approach the efficiency limit of large payloads. Generally, good ME codes are based on suitable linear block codes that are long enough. Due to the complexity of maximum likelihood (ML) decoding, it is difficult to determine the coset preamble of a large linear block code. Numerous approaches using structured codes [8–13] have been developed.

In coding theory, the best ME code (ME code that can approach the upper limit of the embedding efficiency of the rate-distortion function) requires a well-structured code and a sufficiently long effective decoding algorithm, such as a low-density generator matrix code [14]. Researchers have developed a great number of embedding techniques in adaptive steganography. The study in [15] proposes an adaptive algorithm called the linear independent approximation embedding (LIAE) algorithm. The LIAE algorithm has the ability to perform data embedding at an arbitrarily specified cover location. The method presented in this study used a family of convolutional codes known as convolutional embedding (CE) codes for q-ary payloads. The CE code can be used as an alternative approach to the theoretical upper limit of embedding efficiency. The CE code is based on a grid structure and Viterbi decoding (this is an ML algorithm). The CE code is suitable for encoding a payload with a sufficiently large block length to increase the embedding efficiency and change the embedding rate. Additionally, the optimal design of current CEs can be used to obtain the embedding scheme. Moreover, a puncturing technique is suitable for altering the embedding rate of CE codes. For the q-element payload, the CE code can be easily obtained at an embedding rate of 1/2 CE by using a piercing strategy. Experimental results show that the embedding efficiency of CE code is better than that of ME code.

The rest of this article is organized as follows. The second section briefly introduces the basic theory and the scope of the embedding scheme. The third section introduces the embedding algorithm of q-element payload using CE. The fourth part provides experimental results and constructive analysis of the performance of various embedded algorithms. Finally, Section 5 presents conclusions.

2. Preliminaries

1. Cover for multitone images

The q-ary convolutional codes were applied to multitone images. The procedure for constructing the proposed embedding scheme involves the following aspects: (i) how to generate multiple-level tone images from a grayscale image; and (ii) how to construct an embedding system using an optimal decoding algorithm.

Error diffusion is a popular halftoning technique. Two-level representations are used in this technique to replace the original grayscale image or color image. This technique can be considered a generalization of multiple tones. Let $g_{x,y}$ and $h_{x,y}$ be coordinates of a grayscale image and a multiple-tone image, respectively, after quantizing point $g_{x,y}$. The quantization error $e_{x,y}$ is expressed as $e_{x,y} = g_{x,y} - h_{x,y}$. The multiple-tone point $h_{x,y}$ can be obtained using the following expression:

$$h_{x,y} = \begin{cases} t_1, & g_{x,y} \geq \lfloor 255/n \rfloor \\ t_2, & \lfloor 255/n \rfloor \leq g_{x,y} \leq 2\lfloor 255/n \rfloor \\ & \vdots \\ t_n, & (n-1)\lfloor 255/n \rfloor \leq g_{x,y} \leq 255 \end{cases} \tag{1}$$

To transform the grayscale image into a multiple-tone image, an error filter is used in all filtering areas to obtain the final multiple-tone image. By contrast, in the recovery procedure, the multiple-tone image is recovered as the grayscale image. A low-pass filter is used to filter the points in the grayscale image and to obtain a continuous tone image.

2 Embedding scheme and efficiency bound

The goal of the binary embedding scheme is to quantify the source limited by the distortion theory. The embedded model and extraction model are shown in Figure 1.

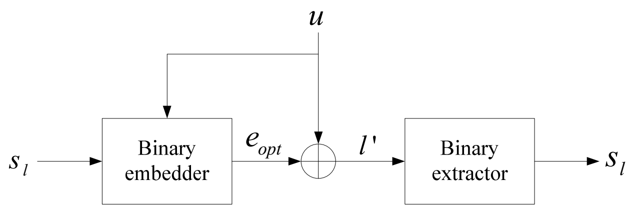


Figure 1. Block diagram of a binary embedding system.

Under the assumption that a logo s_l embedded into a cover u is transmitted to the receiver, the optimal stego $l' = u - e_{opt}$ is provided by the embedder. Thus, a message l', which is modified from u, corresponds to syndrome s_l. Given a cover u, which is a Bernoulli-1/2 process in the binary symmetric source, it subtracts some toggle e; thus $l' = u - e$. Even though the embedder knows the cover u, it cannot simply cancel this known interference due to the constraint that the average number of 1s cannot exceed $n\delta$, where n is the block length and $0 \leq \delta \leq 0.5$. We define the optimal or minimum quantized error $e_{opt} = d_H(l', u)$, where $d_H(\cdot)$ denotes the Hamming distance between stego l' and cover u. The optimal quantization error e_{opt} is the optimal modified vector such that the host u and stego l' are of optimal quantization error. The rate distortion can be calculated as $R(d) = 1 - h(d)$, where d denotes the bound 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 $R_c = k/n \approx R(\delta)$. Thus, the embedding rate is $R_m = (n-k)/n \approx h(\delta)$. Therefore, for a given good linear code with embedding rate $h(\delta)$, optimal distortion δ can be approached. Theoretically, the codeword of a linear code C can be regarded as a quantized message set $C = \{c\}$, with δ as the average distance between an arbitrary cover set $U = \{u\}$. The upper bound of the embedding capacity can thus be expressed as follows:

$$\max_{E[d(C,U)] \leq n\delta} h(C|U) = h(\delta) \tag{2}$$

where $h(\delta)$ is the embedding rate corresponding to the optimal distortion δ. If a well-designed linear code exists, then the theoretical upper bound can be approached by an associated embedder. However, the major concern is to determine a parity-check matrix with a well-behaved (n, k) linear code and a code rate R_c. Furthermore, with the embedding rate requested in such a linear code C, the aforementioned equation can then be expressed as follows:

$$h(\delta) \approx 1 - k/n = m/n \tag{3}$$

For a binary symmetric source and an n bit source sequence $u \in \{0, 1\}^n$, the average distortion per bit is defined as follows:

$$d_{avg} = \frac{E[d(\hat{u}, u)]}{n} = \frac{D_{avg}}{n} \tag{4}$$

where $\hat{u}$ represents a quantized codeword existing in code C, and D_{avg} is the average Hamming distortion between $\hat{u}$ and u per block. For an (n, k) linear block code, the minimum average distortion can be expressed as follows:

$$\delta = h^{-1}(m/n) = h^{-1}(R_m), \tag{5}$$

where $h^{-1}(.)$ is the inverse function of the binary entropy function h. The aforementioned equation is the rate-distortion function. The lower bound δ of average distortion for each bit in a code block is $\delta \leq d_{avg} = D/n$. The lower bound δ of each bit average distortion in blocks is displayed in Figure 2.

When performing the binary data embedding of a sequence of length n bits, the embedding efficiency is defined as follows:

$$\eta = \frac{R_m}{d_{avg}} = \frac{m}{D} \tag{6}$$

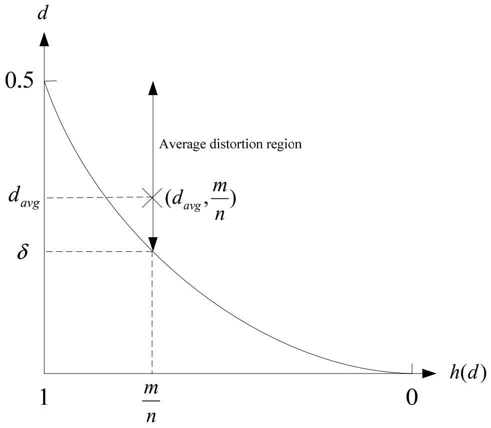


Figure 2. Rate-distortion function.

3. Embedding Algorithm for Small and Large Payloads

Binary data embedding was achieved by using a standard array as follows: with $(n,\ k)$ linear code C, we developed a standard array with a size of $2^{n-k} \times 2^k$, as displayed in Figure 3.

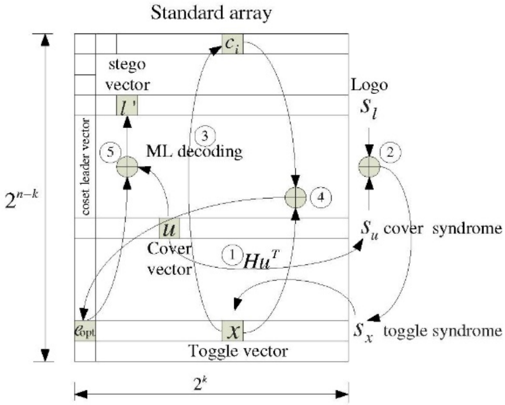


Figure 3. Standard array for the embedding algorithm.

Alternatively, the required coset leader can be determined precisely to perform binary data embedding or optimal embedding. An $(n,\ k)$ linear code C can be characterized with a parity-check matrix H of size $(n-k) \times n$ as follows:

$$C = \left\{ r | Hr^T = 0 \right\} \tag{7}$$

where the sequence is $r \in F_q^n$. Based on (7), the syndrome s of the sequence r is defined as $s = Hr^T$. Furthermore, the set composed of all the sequences r corresponding to identical s is referred to as the coset of code C and is defined as follows:

$$C^s = \left\{ r | Hr^T = s \right\} = \{ c + e | c \in C \} \tag{8}$$

where e denotes the coset leader in the standard array. The term s can be derived through H from an arbitrary sequence r, and e can be expressed using an ML decoding function as follows:

$$e_{opt} = f\left(Hr^T \right) = f(s) \tag{9}$$

where $f(\cdot)$ represents the decoding function of the linear codes. Using ML decoding, the coset leader e is added to r to recover the codeword c, which is closest to the sequence r.

As displayed in Figure 3, for convolutional codes, it is necessary to determine the minimal toggle vector, e_{opt}, namely the coset leader, for a convolutional code C in the vector domain to solve the equation $x = u - l$, where $l^T = H^{-1}s_l$. We considered the following simple embedding method. Using a systematic form CE in the vector domain, the equation can be used to solve the following expression:

$$e_{opt} = f(s_x) = f(H_s x^T) = f(H_s(u-l)^T).$$

Assuming that $l = (s_l, 0, \cdots, 0)$ is a solution for $l^T = H^{-1}s_l$, toggle vector x can be determined immediately with u and s_l; toggle $x = u - l$ can also be determined immediately. This section focuses on the efficient identification of the toggle vector using a systematic encoding technique. A symbol must be defined to describe the embedding of algorithms based on convolutional codes. Assuming that the convolutional code is a non-system generator matrix, it can be converted into a system generator matrix using basic row operations. Alternatively, the code can be expressed in a system recursive form. We use CE to embed binary messages as follows:

An embedding scheme with small payloads is used for numerous applications. However, for a case of (n, k) CE codes with a low embedding rate, the trellis structure has high branches per state because of a large k, which indicates that a complex mechanism is required when performing the Viterbi algorithm. To avoid this disadvantage, we constructed a CE code at a low embedding rate. The $(2, 1)$ CE code was obtained through puncturing. In the time domain, we constructed an embedding rate $R_e = (\tau - N)/(2\tau - N)$, where τ is the puncturing period and $N = 0, 1, \cdots, \tau - 1$ is the number of deleted bits. Systematic recursive CE codes can be obtained by puncturing the output of a $(2, 1)$ convolutional code with the puncturing matrix P as follows:

$$P = \begin{bmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,\tau} \\ p_{2,1} & p_{2,2} & \cdots & p_{2,\tau} \end{bmatrix} = \begin{bmatrix} p_1 \\ p_2 \end{bmatrix}. \tag{10}$$

Based on (10), we selected N and τ to obtain the required R_e. In puncturing matrix P, if $p_{i,j} = 1$, the corresponding output bit from the CE code is embedded. Otherwise, the corresponding output bit from the CE code is deleted. To construct a systematic CE code by puncturing, the embedding algorithm must first locate the matrix l corresponding to message s_l with length $\tau - N$ bits in period τ. However, because of a systematic encoder, $s_l = \{s_{l,i}\}$ is set in assigned locations with respect to the set of indices $S \subset \{1, 2, \cdots, \tau\}$. Here, $|S| = \tau - N$ of the second-row sequence p_2 in period τ. We located $\tau - N$ assigned indexes in s_l corresponding to the location $p_{2,i}$ as follows:

$$(l)_{pun} = \begin{bmatrix} l_{1,1} & l_{1,2} & \cdots & l_{1,T} \\ l_{2,1} & l_{2,2} & \cdots & l_{2,T} \end{bmatrix}, \tag{11}$$

where $l_{1,i} = 0$ and $l_{2,i}$ is expressed as follows:

$$l_{2,i} = \begin{cases} s_{l,i} & \text{, if } i \in S \\ 0 & \text{, if } i \in S \end{cases}. \tag{12}$$

Thus, provided a cover matrix u corresponding to P, we can obtain the toggle matrix as $x = u - l$ in a puncturing period τ.

A notation must be defined to describe the embedding of a convolutional code-based algorithm. Here, CE is a nonsystematic generator matrix that can be translated into a systematic generator matrix using elementary row operations. Alternatively, it can be expressed in systematic recursive form. We used a CE to embed the binary message as follows:

A CE with a generator matrix $G(D)$ is defined as follows:

$$\Lambda = \{c(D) = v(D)G(D)\} \tag{13}$$

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

$$\begin{aligned} e_u &= \text{argmin}\, d(c(D), u(D)) \\ &= u(\mathrm{D}) - Q(u(D)) \\ &= u(D) \bmod \Lambda \end{aligned} \tag{14}$$

where the $Q(x(D))$ is the quantizer, which can be expressed as follows:

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

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

$$d_H(u(D) - c\prime(D)) \leq d_H(u(D) - c(D))$$

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

$$V_0 = \{e_{opt}(D)\} = \{u(D) : Q(u(D)) = 0\}. \tag{15}$$

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

$$\begin{aligned} e_{opt}(D) \quad &\triangleq u(D) \bmod \Lambda^l \\ &= u(D) - l(D) \bmod (\Lambda^l - l) \\ &= x(D) \bmod \Lambda \\ &= x(D) - Q(x(D)) \end{aligned} \tag{16}$$

where the shift sequence $l(D) \in \Lambda^l$, that is, $l(D) = c(D) + e_{opt}(D)$ and $e_{opt}(D)$, denotes the error sequence or coset leader sequence in quantizing toggle sequence $x(D)$ by Λ. It is assumed that cover sequence $u(D)$ is uniformly distributed in $F_2^n(D)$; moreover, the toggle sequence $x(D)$, which is obtained by subtracting message sequence $l(D)$ from cover sequence $u(D)$, is also uniformly distributed. The minimum distance sequence $e_{opt}(D)$ between cover sequence $u(D)$ and message sequence $l(D)$ is equal to (16). By quantizing a random binary sequence $x(D)$ by Λ^l, an average quantized distortion level is represented as follows:

$$\begin{aligned} D_{avg} \quad &= E[w(x(D) = Q(x(D)))] \\ &= E[\sum_{\forall i} w(x_i(D) - Q(x_i(D)))] \\ &= E[\sum_{\forall i} w(e_{opt,i}(D))] \end{aligned} \tag{17}$$

Similar to the linear block codes, the optimal toggle vector must be determined using convolutional systematic codes. A simple method similar to the systematic coding approach is data embedding using linear block codes with 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 CE binary embedding. The embedding procedure for systematic CE is as follows:

For a message syndrome sequence $s_l(D)$ of length $N(nR_c)$, it is necessary to determine sequence $l(D) \in \Lambda^l$ of length Nn with syndrome s_l as the linear code. For a special $(n, 1)$ systematic convolutional code case, a generator matrix $G_s(D)$ can be defined as follows:

$$G_s(D) = [1\; g_1(D)\; g_2(D)\; \cdots\; \mathrm{g}_m(D)] \tag{18}$$

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

$$H_s(D) = \begin{bmatrix} g_1(D) & 1 & 0 & \cdots & 0 \\ \vdots & 0 & 1 & \cdots & 0 \\ g_m(D) & \vdots & \vdots & \ddots & 1 \end{bmatrix} \tag{19}$$

where $H_s(D)$ is an $m \times n$ matrix and embedded sequence $s_l(D) \in F_q^m(D)$ and is derived using the following expression:

$$H_s(D) l(D)^T = s_l(D). \tag{20}$$

which is used to solve the following expression:

$$l(D)^T = H_s(D)^{-1} s_l(D)^T$$

This equation is complex. Due to the systematic encoder, $l(D)^T = [0 \,\cdots\, 0 \; s_l(D)]$ of size $1 \times n$ can be solved. Furthermore, the toggle sequence $x(D)$ is obtained by subtracting $u(D)$ from $l(D)$. Embedder Λ quantizes the arbitrary toggle sequence $x(D)$ to generate the optimal stego sequence $l'(D)$ as follows:

$$\begin{aligned} l'(D) &= u(D) - (x(D) \bmod \Lambda(D)) \\ &= u(D) - (x(D) - Q(x(\mathrm{D}))) \end{aligned} \tag{21}$$

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

$$l'(D) = u(D) + e_{opt}(D) \tag{22}$$

At the receiver, message sequence $s_l(D)$ is extracted as follows: $s_l(D)^T = H_s(D)(l'(D))^T$. To illustrate the nested CE algorithm, the following example is based on a systematic convolutional code to describe the embedding procedure, displayed as follows. Consider an embedded message sequence $s_l = [1, 1, 1, 1, 1]$ and a cover sequence $u = [11, 01, 11, 01, 11]$. As the systematic convolutional codes are used, we easily obtain solution $l(D)$ corresponding to $H_s(D)l(D) = s_l(D)$. Subsequently, a systematic CE binary embedding is performed. Assuming that $s_l = (l_1, l_2, \cdots, l_M)$ is the symbol intended for embedding, vector $l = (0, l_1, 0, l_2, \cdots, 0, l_M)$ represents a sequence, that is, a (2, 1) systematic CE with the syndrome s_l, $l = (0, 1, 0, 1, 0, 1, 0, 1, 0, 1)$, and the toggle sequence corresponding x to s_x and falling within the coset C^x can be determined as follows:

$$\begin{aligned} x &= l + u \\ &= (0, 1, 0, 1, 0, 1, 0, 1, 0, 1) + (1, 1, 0, 1, 1, 1, 0, 1, 1, 1) \\ &= (1, 0, 0, 0, 1, 0, 0, 0, 1, 0). \end{aligned}$$

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:

$$\begin{aligned} e_{opt}(D) &= Viter(x(D)) + x(D) \\ &= (1, 1, 0, 0, 1, 0, 0, 0, 1, 0) + (1, 0, 0, 0, 1, 0, 0, 0, 1, 0) \\ &= (0, 1, 0, 0, 0, 0, 0, 0, 0, 0) \end{aligned} \tag{23}$$

where $Viter(\cdot)$ is a Viterbi decoding function. The procedure for finding an optimal toggle sequence $e_{opt}(D)$ is displayed in as above. Finally, the stego sequence can be obtained as follows:

$$\begin{aligned} l'(D) &= u(D) + e_{opt}(D) \\ &= (11, 01, 11, 01, 11) + (01, 00, 00, 00, 00) \\ &= (10, 01, 11, 01, 11) \end{aligned} \tag{24}$$

In the receiver, we reconstructed the message sequence $s_l(D)$ as follows:

A crucial factor of CE codes is the method for determining the optimal generator matrix for large payloads.

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

4. Optimal Design for Q-Ary CEs

We used q-ary CE codes to achieve a high-performance ME. A structured code is commonly required in an efficient embedding algorithm. The ML decoding algorithm can be used to determine the optimal embedding algorithm. Optimal ML decoding (Viterbi decoding) can be performed with an existing CE. The CE features a sufficiently larger length of codeword compared with the block code, in which a short block of fixed length is used as a codeword set. A good CE code with a sufficiently large codeword length has $2^{nh(\delta)}$ cosets.

Next, the nonbinary embedding algorithm using CE codes, which involved modification of the cover samples by $\pm\lfloor (q-1)/2 \rfloor$, was demonstrated. The algorithm was applied to an arbitrary selection of cover location. Although the proposed scheme used the embedding algorithm over F_q, it can be used in the nonbinary domain for various applications for increasing embedding efficiency. We used q-ary random codes and searched the generator matrix to implement the nonbinary embedding algorithm. By applying $\pm\lfloor (q-1)/2 \rfloor$ embedding, we could generate embedding codes with optimal embedding efficiency and obtain numerous designs of generator polynomials of CE codes over F_q, as displayed in Figures 4–10.

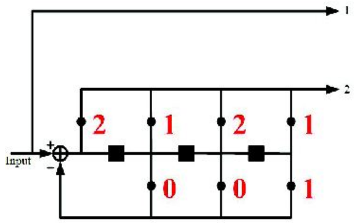


Figure 4. 3-Ary, constraint length = 4, η = 4.37.

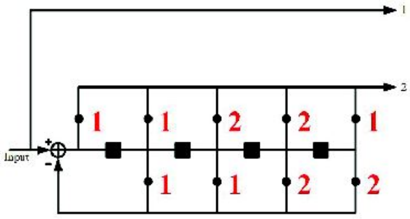


Figure 5. 3-Ary, constraint length = 5, η = 4.53.

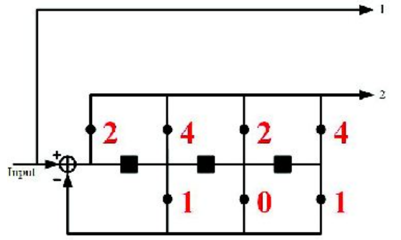


Figure 6. 5-Ary, constraint length = 4, η = 5.1.

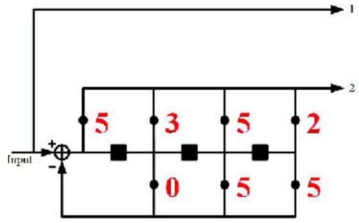


Figure 7. 7-Ary, constraint length = 4, η = 5.42.

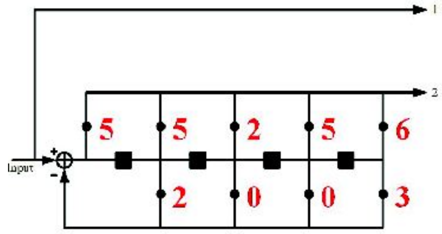


Figure 8. 7-Ary, constraint length = 5, η = 5.56.

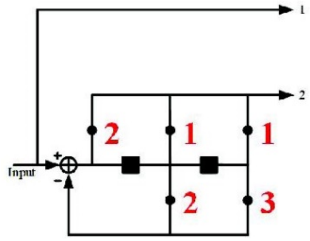


Figure 9. 5-Ary, constraint length = 3, η = 4.8.

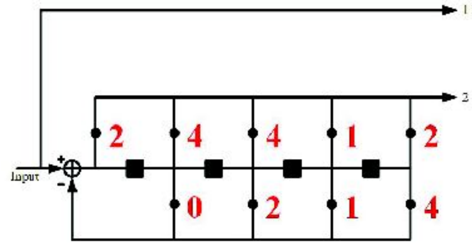


Figure 10. 5-Ary, constraint length = 5, η = 5.1.

Constructing a structured q-ary CE code with embedding efficiency close to the theoretical limit is a key open problem, involving the following aspects: (1) the embedding scheme requires a structured code of sufficient length, and must have an excellent parity check matrix or generator matrix; (2) the structured code is computationally efficient, and an effective encoding/decoding process has been developed based on the structured code.

5. Simulation Results

In this section, we describe how the packet form of q-ary CE codes is used in the application field. Consider the initial first packet data in Figure 11. This packet information is embedded into the $\pm\lfloor (q-1)/2 \rfloor$ least significant bit channel.

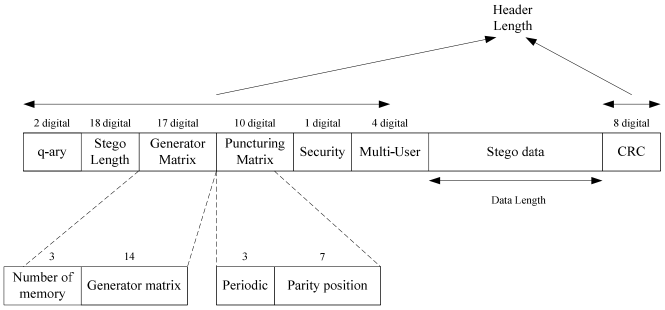


Figure 11. Packet form in the application field.

Next, Figure 12 presents an example of the practical packet form.

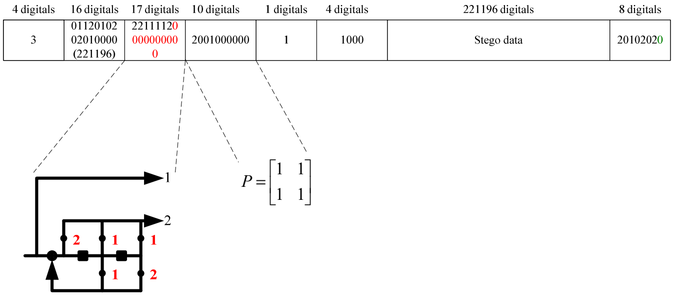


Figure 12. Example of the packet form.

Figure 13 displays the graphical user interface (GUI) of image steganography performed using MATLAB for embedding various images of different sizes over the proposed packet form. Total embedding involves cryptography with a random key. The embedding system includes the cyclic redundancy check (CRC) detection model. In the model, we examined the sensitivity of computer simulation results to the various images of different sizes to represent the rand errors and retransmission in the simulation. In the CRC model, it is assumed that the size of image capable for embedding can be protected for total length of the packet. The party check of CRC mode uses the following standard of ITU-IEEE:

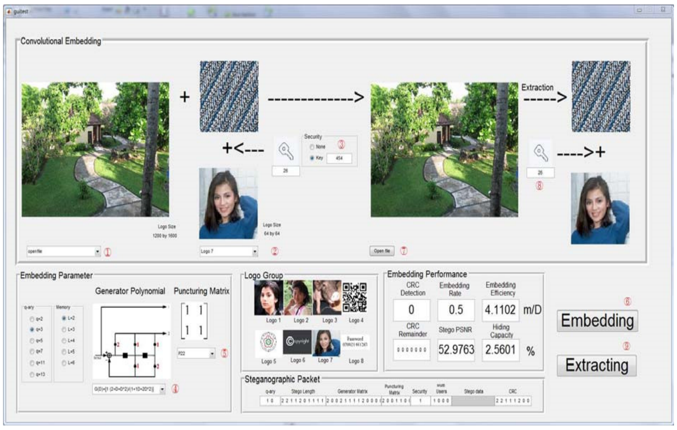
$$g(x) = 1 + x^2 + x^3 + x^7 + x^8$$

$$g(x) = 1 + 2x + x^2 + x^7$$

$$g(x) = 3 + 4x + x^5$$

$$g(x) = 5 + 6x + 3x^2 + x^3 + x^4$$

The GUI embedding system has the following characteristics: (1) image visualization: the image logo message stream is embedded with a q-ary cover and the packet of the embedding message is used in the security system. (2) Optimal design: some reordering or permutation of the image cover can be used to optimize q-ary CE codes. (3) Error detection mode: the component is used to protect the message packet from the attack channel.



Puncturing Matrix:
$$\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

CRC Detection: 0; Embedding Rate: 0.5; Embedding Efficiency: 4.1102 m/D

CRC Remainder: 0000000; Stego PSNR: 52.9763; Hiding Capacity: 2.5601 %

Steganographic Packet: q-ary 10; Stego Length 2211201111; Generator Matrix 2002111120000; Puncturing Matrix 200110; Security 1; multi Users 1000; CRC 22111200

Figure 13. GUI of the image steganography system.

Figure 13 indicates that the recovery logo messages in the GUI were the same as the original logo message. Thus, the transmitted and received message were the same. The square of CRC indicated simulation results that use the above party check polynomial to protect the packet, and the generator polynomial with various sizes was selected. The generator polynomials of q-ary convolutional with q = 3–7 and length 2–5 were used in the simulation. For convolutional codes with low embedding rate, the grid structure has a large number of branches per state, which means that a smaller number of metric operations are required to execute the Viterbi algorithm, and vice versa.

For constructing a high complexity code, a convolutional code with a high embedding rate is structured using a convolutional code with a low embedding rate through puncturing. Ultimately, the embedding rate of a (2, 1) convolutional code is maintained constant to design a convolutional code with various embedding rates through puncturing, and the complexity of the designed convolutional code is compared with that of the (2, 1) convolutional code. Moreover, the work of [16] proposed that the LDGM embedding codes and the embedding efficiency was the best performance for the study in steganography. Figure 14 shows the comparison of embedding efficiency between [16] and this study.

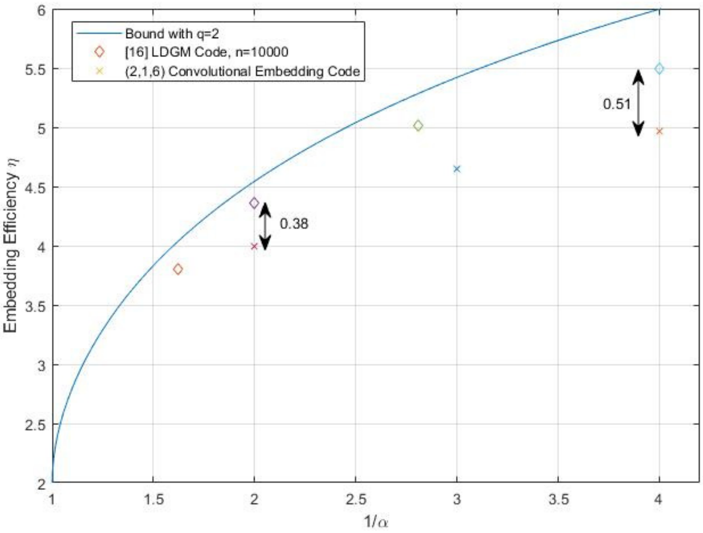


Figure 14. The embedding efficiency between [16] and convolutional embedding codes.

6. Conclusions

A novel decoding method based on q-ary convolutional codes for ME in steganography was proposed in this study. Generally, the q-ary embedding scheme is applied to multiple-tone images. The q-ary level and the simulation of optimal embedding is run using the full search method. In q-ary CE, we used a q value of 3–7 and a code length of 3–5. The proposed method not only performed optimal decoding but also achieved optimal embedding efficiency for ME convolutional codes. The Viterbi decoding procedure was also used for this study. Moreover, the operation can be performed using a GUI for embedding applications.

Author Contributions: Conceptualization, J.-J.W., C.-Y.L., and H.-Y.C.; methodology, J.-J.W., C.-Y.L., and H.-Y.C.; software, J.-J.W. and H.-Y.C.; validation, S.-C.Y.; formal analysis, C.-Y.L.; investigation, S.-C.Y.; resources, C.-Y.L., S.-C.Y., H.-Y.C., and Y.-C.L.; data curation, J.-J.W. and H.-Y.C.; writing—original draft preparation, J.-J.W.; writing—review and editing, J.-J.W.; visualization, Y.-C.L.; supervision, S.-C.Y.; project administration, C.-Y.L.; funding acquisition, C.-Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Crandall, R. Some Notes on Steganography. Post on *Steganography Mailing List*. 1998. Available online: http://os.inf.tu-dresden.de/west-feld/crandall.pdf (accessed on 18 June 2021).
2. Bierbrauer, J. On Crandall's Problem. 1998, unpublished. Available online: http://www.ws.binghamton.edu/fridrich/covcodes.pdf (accessed on 18 June 2021).
3. Galand, F.; Kabatiansky, G. Information hiding by coverings. In Proceedings of the 2003 IEEE Information Theory Workshop (Cat. No.03EX674), Paris, France, 31 March–4 April 2003 ; pp. 151–154. [CrossRef]
4. Zhang, W.; Wang, S.; Zhang, X. Improving embedding efficiency of covering codes for applications in steganography. *IEEE Commun. Lett.* **2007**, *11*, 680–682. [CrossRef]
5. Bierbrauer, J.; Fridrich, J. Constructing good covering codes for applications in Steganography. In *LNCS Transactions on Data Hiding and Multimedia Security*; Springer: Berlin/Heidelberg, Germany, 2008; Volume 4920, pp. 1–22.
6. Fridrich, J.; Filler, T. Practical methods for minimizing embedding impact in steganography. In Proceedings of the Security, Steganography, and Watermarking of Multimedia Contents IX, San Jose, CA, USA, 26 February 2007; Volume 6050, p. 2V3.
7. Fridrich, J.; Soukal, D. Matrix embedding for large payloads. *IEEE Trans. Inf. Forensics Secur.* **2006**, *1*, 390–395. [CrossRef]
8. Tseng, Y.-C.; Chen, Y.-Y.; Pan, H.-K. A secure data hiding scheme for binary images. *IEEE Trans. Commun.* **2002**, *50*, 1227–1231. [CrossRef]
9. Li, R.Y.; Au, O.C.; Lai, K.K.; Yuk, C.K.; Lam, S.-Y. Data hiding with tree based parity check. In Proceedings of the IEEE International Conference, Beijing, China, 2–5 July 2007; pp. 635–638.
10. Li, R.Y.; Au, O.C.; Yuk, C.K.M.; Yip, S.-K.; Lam, S.-Y. Halftone Image Data Hiding with Block-Overlapping Parity Check. In Proceedings of the 2007 IEEE International Conference on Acoustics, Speech and Signal Processing—ICASSP' 07, Honolulu, HI, USA, 15–20 April 2007; Volume 2, pp. 193–196.
11. Chen, J.; Zhu, Y.; Shen, Y.; Zhang, W. Efficient Matrix Embedding Based on Random Linear Codes. In Proceedings of the MINES 2010, Jiangsu, China, 4–6 November 2010; pp. 879–883.
12. Gao, Y.; Li, X.; Yang, B. Employing optimal matrix for efficient matrix embedding. In Proceedings of the IIH-MSP2009, Kyoto, Japan, 12–14 September 2009; pp. 161–165.
13. Sch¨onfeld, D.; Winkler, A. Embedding with syndrome coding based on BCH codes. In Proceedings of the ACM 8th Workshop on Multimedia and Security, Geneva, Switzerland, 26–27 September 2006; pp. 214–223.
14. Wainwright, M.J. Sparse graph codes for side information and binning. *IEEE Signal Process. Mag.* **2007**, *24*, 47–57. [CrossRef]
15. Wang, J.J.; Chen, H. An Adaptive Matrix Embedding Technique for Binary Hiding with an Efficient LIAE Algorithm. *WSEAS Trans. Signal Process.* **2012**, *8*, 64–75.
16. Filler, T.; Fridrich, J. Binary quantization using Belief Propagation with decimation over factor graphs of LDGM codes. *arXiv* **2007**, arXiv:0710.0192.