Advance Access publication on 3 November 2011 doi:10.1093/comjnl/bxr111

Transparency-Orientated Encoding Strategies for Voice-over-IP Steganography

Hui Tian[1,*], Hong Jiang[2], Ke Zhou[3] and Dan Feng[3]

[1]*College of Computer Science and Technology, National Huaqiao University, Xiamen 361021, China*
[2]*Department of Computer Science and Engineering, University of Nebraska-Lincoln, Lincoln, NE 68588-0150, USA*
[3]*School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China*
[*]*Corresponding author: htian@hqu.edu.cn*

Embedding transparency is one of the most important criteria for steganography. This paper mainly focuses on transparency-orientated encoding strategies for steganography on Voice over IP (VoIP). Our analysis of the existing encoding strategies proposed for steganography on storage media reveals that they have limited applicability in VoIP-based steganography because they enhance embedding transparency at the expense of a decreased embedding rate (ER). In this paper, we propose three encoding strategies based on digital logic for steganography on VoIP. Differing from the existing approaches, our strategies reduce the embedding distortion by improving the similarity between the cover and the covert message using digital logical transformations, instead of reducing the ER. Therefore, in contrast, our strategies improve the embedding transparency without sacrificing the embedding capacity. Of these three strategies, one adopts logical operations, one employs circular shifting operations and the third combines the operations of the first two. All these schemes are evaluated experimentally through their prototype implementations that are compared with some existing methods in a prototypical covert communication system based on VoIP (called StegVoIP). The experimental results show that the proposed strategies can provide better embedding performance than the existing approaches in terms of embedding transparency and ER.

Keywords: steganography; transparency; similarity; voice over IP; digital logic

Received 29 June 2011; revised 1 October 2011
Handling editor: Wojciech Mazurczyk

1. INTRODUCTION

Steganography, a technology of information hiding, has captured the imagination of researchers for many years [1]. So far, various studies on steganography have been carried out on storage media, including image [2], video [3], audio [4] and text [5]. In recent years, a new steganographic cover, Voice over IP (VoIP) [6], has attracted significant attention, which has two main advantages over storage media. Firstly, the real-time nature of VoIP provides better security for secret messages by virtue of its instantaneity, because it does not give eavesdroppers sufficient amount of time to detect possible abnormity due to hidden messages. In fact, the recognition of a specific VoIP traffic in enormous network traffic is a very perplexing and challenging problem, let alone detecting possible covert messages. Secondly, VoIP can be considered a multidimensional carrier in that both the packet protocol headers and the payload data can be used to hide data. Due to the above advanced characteristics, steganography over VoIP can provide an alternative solution for secure covert communication. However, if the technique is used by lawbreakers (e.g. terrorists), it will pose a threat to network security and bring a new challenge to network forensics (a subbranch of digital forensics), because unauthorized information can be easily smuggled through firewalls and other monitor equipment without being undetected. Therefore, it can be often considered as a double-edged sword. The trade-off between the benefits and threats involves many complex ethical,

legal and technological issues [7]. Nevertheless, steganography over VoIP is still a worthy subject of thorough studies, given the potential advantages.

Recently, many researchers have carried out useful research on steganography over VoIP [7–16]. Most of them [7–13] mainly focus on the embedding methods and/or prototypical implementations of steganography over VoIP. However, our observation and empirical study suggest that the encoding problem for VoIP-based steganography is also very important. Generally speaking, encoding strategies can be divided into four categories according to their objectives.

(i) *Capacity-orientated strategy* (*CoS*). CoS aims to increase the information capacity of the covert message. It often involves certain compression codes, such as T-code, Run Length Code, etc.

(ii) *Reliability-orientated strategy* (*RoS*). RoS aims to assure the correct extraction of covert messages at the receiver side. It often involves certain check codes [14] (e.g. Checksum, CRC and MD5) or error-correcting codes (e.g. BCH code, Hamming code and Turbo code).

(iii) *Security-orientated strategy* (*SoS*). SoS aims to enhance the security of the covert message, namely, to avoid or minimize the possibility of unauthorized extraction. In Ref. [10], the authors introduced traditional cryptographies for embedded messages. Owing to the time-consuming characteristic of traditional cryptographies, this encryption operation is often designed to be carried out offline before the embedding process, which is efficient for transmitting bulk covert messages. However, it is not suitable for the real-time exchange of secret short messages that are popular in interactive scenarios. In our previous studies, we employed pseudo-random binary sequences, namely, the pseudo-random sequence (PRS) produced by an improved Mersenne Twiste algorithm [14] and the m-sequence [16], to encrypt the covert messages with the Exclusive OR (XOR) operation, which can strike a good balance between adequate security and low latency for real-time services.

(iv) *Transparency-orientated strategy* (*ToS*). ToS aims to reduce the distortion and thereby enhance the steganographic transparency. Generally, a given steganographic method is believed to possess an inherent transparency. In other words, it is commonly believed that the distortion induced by a given steganographic method is settled. However, we argue that the transparency of a given steganographic method can be improved by proper encoding. In this paper, we mainly focus on investigating some encoding strategies used to enhance the transparency of steganographic methods for VoIP.

Huang *et al.* [11] first focus on steganographic distortion in VoIP-based steganography. In their method, a PRS is used to guide the selection of embedding samples. Although this method reduces the distortion of the cover speech at the expense of halving the maximum embedding capacity, it points to the significance of ToS for VoIP-based steganography. In fact, ToS has proved its crucial importance in years of practical application in steganography over storage media [17, 18]. Many effective strategies have been developed in the literature. For example the random interval method (RIM) [19–21] can spread the secret message over the cover in a rather random manner with less distortion using a pseudo-random number generator; the random position method (RPM) [22], on the other hand, dynamically determines substitution bits with given probability thresholds, which can decrease the amount of substitution bits and thereby reduce the distortion. To a certain degree, the method in Ref. [11] can be viewed as a special case of the RPM. Tseng *et al.* [23] presented a secure data hiding approach for binary images, which can conceal $\lfloor \log_2(mn+1) \rfloor$ bits of data in a binary image block of size $m \times n$ by modifying at most two bits. Meanwhile, Westfeld [24] first introduced the matrix encoding (ME) strategy into his F5 algorithm, which can embed l bits into $2^l - 1$ pixels with no more than one bit changed. However, all these strategies enhance embedding transparency at the cost of decreasing the embedding rate (ER) (or the practical embedding capacity), which may not be a significant concern if the cover capacity of the host object (e.g. image) is sufficiently large. Unfortunately, in VoIP-based steganography scenarios, the length of the cover speech depends on the conversation, which is often irregular and unpredictable. Further, the embedding capacity is more significant if VoIP-based steganography is used to construct covert communication. Therefore, the application of the approaches mentioned above in VoIP-based steganography scenarios is clearly limited. In recent years, some researchers also proposed some improved ME strategies (e.g. simplex code [25] and random linear code [26]) to increase the payload of steganography based on storage media. However, due to their relatively high computational complexities, they cannot also be applied in real-time steganography over VoIP.

Our previous work [15] concludes that the embedding transparency can be enhanced by taking into account the similarity between the covers and the secret messages. Further, our observations through research and experiments suggest that the similarity between the covers and the embedded messages can be significantly increased by exploiting some transformations of the embedded messages. Therefore, we are motivated to propose the encoding strategies based on digital logical transformations for steganography over VoIP, which can reduce the embedding distortion and enhance the embedding transparency as a consequence while maintaining the maximum ER. Although our strategies are applicable to various steganographic approaches, we chose to take the well-known Least-Significant-Bits (LSBs) steganography as example for ease of presentation.

The rest of this paper is organized as follows. Metrics for measuring the embedding performance are defined in Section 2. Section 3 first introduces existing storage media-based

approaches to improve the embedding transparency of steganography over VoIP and theoretically analyzes their steganographic performance. The proposed encoding strategies based on digital logic are presented in Section 4. Section 5 describes the prototypical implementations of the proposed strategies and evaluates their steganographic performances by comparing them with the existing approaches. Finally, Section 6 concludes the paper with remarks on the main contributions of the paper and directions of future study.

2. MEASURE CRITERIA

For the convenience of evaluating the performance of different encoding methods in the rest of the paper, this section will define steganography over VoIP and introduce some measuring criteria for ToS, including similarity, distortion and efficiency.

Based on the definition of steganography [21], the standard VoIP-based steganography with symmetric key can be formally defined as follows:

DEFINITION 1 (VoIP-BASED STEGANOGRAPHY). *Given the quintuple* $\emptyset =< C, M, K, E_K, R_K >$, *where* C *is the set of VoIP packets (VoIP stream),* M *is the set of secret messages with* $|C| \geq |M|$, K *is the set of secret symmetric keys,* E_K: $C \times M \times K \rightarrow C^*$ *represents the embedding function in which* C^* *is the set of VoIP packets with secret messages, and* R_K: $C^* \times K \rightarrow M$ *represents the restituting function. For* $\forall c \in C$, $m \in M$ *and* $k \in K$, *a system with the property of* $R_K(E_K(c, m, k), k) = R_K(c^*, k) = m$ *is called a VoIP-based steganography system.*

To evaluate the similarity between the cover and the corresponding stego-object (the cover embedded with the secret message), the authors in Ref. [21] first formally introduced the similarity function as follows:

DEFINITION 2 (SIMILARITY FUNCTION). *Let* C *be a non-empty set. A function sim:* $C^2 \rightarrow (-\infty, 1]$ *is called the similarity function on* C, *for* $x, y \in C$, *if* $x = y$, $sim(x, y) = 1$; *otherwise,* $sim(x, y) < 1$.

According to this definition, the transparency criterion for steganography can be formally defined as, for $\forall c \in C$ and $m \in M$, maximizing the value of the following function:

$$f(c, m) = \text{sim}(c, c^*) - 1 = \text{sim}(c, E_K(c, m, k)) - 1, \quad (1)$$

where $f(c, m) \in (-\infty, 0]$. If $c = m$, $\text{sim}(c, m) = 1$, then $f(c, m)$ reaches its maximum value 0. That is to say, if the bit stream of the cover is exactly the same as the bit stream of secret messages to be embedded, the steganography has the best transparency without any induced distortion; otherwise, the steganography inevitably induces a certain degree of distortion. The better the similarity between c and m, the less will be the distortion induced and the higher will be the transparency obtained.

Although this theoretical definition points out that the objective of improving the steganographic transparency is to increase the similarity between the cover and the secret message, it cannot be employed directly to measure the practical similarity or the embedding transparency. We thereby give an alternative definition for similarity that is measurable as follows:

DEFINITION 3 (MEASURABLE SIMILARITY). *Given bit set* $X = \{x_1, x_2, \ldots, x_N\}$ *and* $Y = \{y_1, y_2, \ldots, y_N\}$, *where* N *is the length of* X *and* Y, *the similarity between* X *and* Y *can be determined as follows:*

$$Sim(X, Y) = \sum_{i=1}^{N} (\overline{x_i \oplus y_i}), \quad (2)$$

where '$\oplus$*' is the bitwise XOR operation and '*$\bar{x}$*' is the bitwise NOT operation of bit* x. *This equation means that the similarity between* X *and* Y *can be measured by the number of identical bits between them. If we assume that* M *is the bit set of a given secret message and* C *exactly represents the bit set of a given cover (e.g. LSBs set) employed to hide* M, *then the similarity between* M *and* C *is Sim(*M, C*). If* X *and* Y *are each part of* M*and* C, *respectively, the value of Sim(*X, Y*) is called the partial similarity value (PSV) between* M *and* C.

Further, we can accordingly define the distortion function as follows:

DEFINITION 4 (DISTORTION FUNCTION). *Given bit set* $X = \{x_1, x_2, \ldots, x_N\}$ *and* $Y = \{y_1, y_2, \ldots, y_N\}$, *where* N *is the length of* X *and* Y, *the distortion between* X *and* Y*can be determined as follows:*

$$Dis(X, Y) = \sum_{i=1}^{N} (x_i \oplus y_i) \cdot \eta_i, \quad (3)$$

where η_i *is the distortion factor, indicating the distortion induced by changing* y_i. *This equation means that the distortion between* X *and* Y *is measured by the sum of distortion factors of changed bits. However, it is an intricate problem to quantify the embedding impact and determine the distortion factor for each cover bit. In this paper, we give an alternative distortion measure function as follows:*

$$\begin{aligned} Dis(X, Y) &= N - \sum_{i=1}^{N} (\overline{x_i \oplus y_i}) \\ &= \sum_{i=1}^{N} (x_i \oplus y_i). \end{aligned} \quad (4)$$

The equation means that the distortion between X and Y can be measured by the number of different bits (i.e. the hamming distance) between them, which is an approximate estimate, based on the assumption that the distortion factors of all cover bits are identical. We can justify this assumption for

the following two reasons. Firstly, we often employ the LSBs as the cover, of which the embedding impacts are quite small and slightly different from each other. Secondly, the function is mainly used to compare encoding strategies that are performed with the same cover bits. Clearly, the more the cover bits are changed, the higher is the distortion induced. Thus, it is reasonable to employ the hamming distance between the bits to be embedded (i.e. the secret message bits or their transformed versions) and the same cover bits to evaluate the distortions induced by different encoding strategies.

For convenience, we often normalize the distortion value, and consequently obtain the bits-change rate (BCR), which is the ratio between the number of bits changed and the total number of bits, namely,

$$\mathrm{BCR} = \frac{\mathrm{Dis}(X, Y)}{N} = \frac{\sum_{i=0}^{N}(x_i \oplus y_i)}{N}. \tag{5}$$

BCR, instead of the distortion value calculated by Equation (4), will be employed to evaluate the embedding transparency of a given encoding strategy in the following sections.

In addition, as mentioned above, the ER is another significant measurement criterion.

DEFINITION 5 (EMBEDDING RATE). *If the cover C is used to hide the secret message M, where $|C| \geq |M|$, then the ER can be calculated as follows:*

$$ER = \frac{|M|}{|C|} \tag{6}$$

ER is also called the usage rate of the cover. Usually, $|M|$ denotes the practical embedding capacity of a given method. Therefore, ER can indicate the practical embedding capability of a given method.

To synthetically evaluate the steganographic performance in terms of both the embedding transparency and the ER, we define the effective bit-change rate (EBCR) as follows.

DEFINITION 6 (EFFECTIVE BIT-CHANGE RATE). *If the cover C is used to hide the secret message M and thereby transformed into a stego-object C^*, where $|C| \geq |M|$ and $|C| = |C^*|$, then the EBCR is the ratio between the BCR and the ER, namely,*

$$EBCR = \frac{BCR}{ER} = \frac{Dis(C, C^*)/|C|}{|M|/|C|} = \frac{Dis(C, C^*)}{|M|}. \tag{7}$$

Obviously, EBCR denotes the average number of cover bits that need to be changed for embedding 1 bit of the secret message. Further, the distortion between the cover and the stego-object can be determined as follows:

$$\begin{aligned} \mathrm{Dis}(C, C^*) &= |C| \times \mathrm{BCR} \\ &= \frac{|M|}{\mathrm{ER}} \times \mathrm{BCR} \\ &= |M| \times \mathrm{EBCR}. \end{aligned} \tag{8}$$

The equation indicates that the embedding distortion essentially depends on the practical embedding capacity and the EBCR.

3. ANALYSIS OF PREVIOUS APPROACHES

The previous approaches [19–24] for steganography based on storage media are not well suited for steganography based on VoIP, especially covert communication, because they tend to sacrifice the embedding capacity of the cover in varying degrees in exchange for improving the embedding transparency. For the convenience of comparing them with our strategies detailed in the next section, this section briefly introduces their possible applications in the VoIP-based steganography without considering the requirement of embedding capacity, and analyze their theoretical steganographic performance.

For the convenience of the following presentation, we assume that $M = \{m_1, m_2, \ldots, m_L\}$ is the bit set of a given secret message, where L is the length of the secret message; $C = \{c_1, c_2, \ldots, c_N\}$ is the LSBs set in the VoIP stream and the embedding result is denoted by $C^* = \{c_1^*, c_2^*, \ldots, c_N^*\}$, where N is the total number of LSBs.

3.1. Random interval method

The RIM improves the embedding transparency by spreading the secret message over the cover in a random manner [19–21]. To use the RIM, both communicating parties share the same knowledge of a pseudo-random number generator and a stego-key k as a seed, and so they can gain an identical random sequence $K = \{k_1, k_2, \ldots, k_L\}$ to guide both the embedding process and the restituting process. Formally, the embedding process can be described as follows:

$$\begin{aligned} C^* &= E_K(M, C, K) \\ &= \sum_{i=1}^{L}\left(\sum_{j=n_{i-1}+1}^{n_i-1} c_j + (m_i \otimes c_{n_i})\right) + \sum_{i=n_L+1}^{N} c_i, \end{aligned} \tag{9}$$

where $\otimes$ is the operation of bit substitution, i.e. $\forall\ y_1, y_2 = 0$ or 1, $y_1 \otimes y_2 = y_1$; and n_i is the index of elements in C and C^*, which can be determined by the following formula:

$$n_i = \begin{cases} 0 & \text{if } i = 0, \\ n_{i-1} + k_i & \text{else if } 1 \leq i \leq L, \end{cases} \tag{10}$$

where $n_0 = 0$ is a meaningless index and used to be the initial value of the recursion. The key point of the embedding process is to substitute c_{n_i} with m_i. Accordingly, the restituting process can be described as follows:

$$M = R_K(C^*, K) = \sum_{i=1}^{L} c_{n_i}^*. \tag{11}$$

It is worth noting that the above algorithm is based on the assumption that the cover is long enough to hide the secret

message, namely,

$$n_L = \sum_{i=1}^{L} k_i \leq N, \tag{12}$$

which is also the significant precondition for the application of the RIM. Although it is very difficult to learn exactly how long the cover is sufficient, we can roughly estimate it if the value range of each element k_i in K is predefined. For example if each k_i is limited within 1 to $k_{\max}$, then we can confirm that the covers with the length no $< L \cdot k_{\max}$ are enough to hide L bits of secret messages. Clearly, the ER in the RIM is L/N; that is, the embedding capacity of the RIM is only L/N times of that of the traditional LSBs method. Moreover, because the embedded messages are usually encrypted before the embedding process, they can be considered as a random bit stream in which the values are evenly distributed between 0 and 1. Therefore, the average $\mathrm{Dis}(C, C^*) = 0.5 \times |M| = 0.5L$, average EBCR $= 0.5$ and average BCR $= L/2N$. Generally, the average BCR of the simple LSBs method is 0.5. In contrast, the RIM decreases the BCR and thereby enhances the embedding transparency.

3.2. Random position method

As its name suggests, the RPM determines the substitution positions in a random manner. In the RPM, each bit in the cover set C is endued with a substitution probability in advance; that is, a probability set $P = \{p_i | p_i \in [0, 1], i = 1, 2, \ldots, N\}$ is predefined and shared between both the sender and the receiver. Moreover, both the communicating parties create an identical random sequence $K = \{k_i | k_i \in [0, 1], i = 1, 2, \ldots, N\}$ via a shared pseudo-random number generator with the same stego-key k as a seed. At the sender side, if $k_i \leq p_i$, then c_i can be used to hide a bit of secret message; otherwise, c_i is not employed. Accordingly, at the receiver side, a bit of secret message can be collected from c_i^* if $k_i \leq p_i$; and c_i^* is ignored if $k_i > p_i$. From the above embedding process, we can learn that the average embedding capacity of the RPM can be represented as

$$|M|_{\mathrm{avg}} = \sum_{i=1}^{N} p_i. \tag{13}$$

So, the average ER is given as follows:

$$\mathrm{ER}_{\mathrm{avg}} = \frac{|M|_{\mathrm{avg}}}{|C|} = \frac{\sum_{i=1}^{N} p_i}{N}. \tag{14}$$

Further, since the bits of the embedding message are usually equally distributed between 0 and 1, the average EBCR is 0.5. Accordingly, the average BCR is

$$\begin{aligned} \mathrm{BCR}_{\mathrm{avg}} &= \mathrm{EBCR}_{\mathrm{avg}} \times \mathrm{ER}_{\mathrm{avg}} \\ &= \frac{\sum_{i=1}^{N} p_i}{2N}. \end{aligned} \tag{15}$$

And the average embedding distortion is

$$\begin{aligned} \mathrm{Dis}_{\mathrm{avg}}(C, C^*) &= |M|_{\mathrm{avg}} \times \mathrm{EBCR}_{\mathrm{avg}} \\ &= \sum_{i=1}^{N} \frac{p_i}{2}. \end{aligned} \tag{16}$$

In contrast to the traditional LSBs method, the RPM decreases not only the BCR but also the ER, because the RPM does not change the EBCR. However, we may balance the BCR (embedding distortion) and the ER (embedding capacity) by properly setting substitution probabilities. An ideal approach is to set the substitution probability of each bit in C according to its impact on the perceptual effect. Unfortunately, the evaluation of this impact is often a very difficult problem, and so the substitution probabilities are usually set identically. In this case, let the substitution probabilities be p; then $|M|_{\mathrm{avg}} = N \cdot p$; $\mathrm{ER}_{\mathrm{avg}} = p$; $\mathrm{BCR}_{\mathrm{avg}} = p/2$ and $\mathrm{Dis}_{\mathrm{avg}}(C, C^*) = N \cdot p/2$.

Further, there are two approaches that can be considered as the special cases of the RPM. In the proposed approach in Ref. [11], a PRS is used to determine embedding positions (the embedding LSBs). If the current value of the PRS is '1', the corresponding LSB is chosen to hide 1 bit of secret messages; otherwise, the corresponding LSB is not employed. Since the values in the PRS are evenly distributed between 0 and 1, the selection probability of each bit is 0.5; that is, this method will halve the maximum embedding capacity, i.e. $|M|_{\mathrm{avg}} = 0.5N$. For this reason, this approach can reduce the embedding distortion. In the other special case, LSBs are divided into some groups with the same length l. One bit in each group is randomly chosen to hide 1 bit of secret messages; that is, the selection probability of each bit is $1/l$. Accordingly, $|M|_{\mathrm{avg}} = N/l$; $\mathrm{ER}_{\mathrm{avg}} = 1/l$; $\mathrm{BCR}_{\mathrm{avg}} = 1/2l$; and $\mathrm{Dis}_{\mathrm{avg}}(C, C^*) = N/2l$. Therefore, this approach enhances the embedding transparency at the expense of decreasing the practical embedding capacity to $1/l$ times of the maximum embedding capacity.

3.3. TSENG algorithm

The TSENG algorithm (named with the surname of the first author) [23] is a secure encoding algorithm proposed for steganography in binary images. In this algorithm, a given binary image is often divided into some blocks sized $m \times n$, and a binary matrix and a weight matrix are used as secret keys to protect the hidden information. Generally, the TSENG algorithm can conceal $\lfloor \log_2(mn+1) \rfloor$ bits of data in each block by modifying at most 2 bits. In VoIP-based scenarios, TSENG algorithm is also applicable, if we view all LSBs in a given cover speech as a whole. Accordingly, the involved matrices are changed into vectors. An improved TSENG algorithm for steganography over VoIP can be typically described as follows.

Assume that a given LSB group C' includes l LSBs, which is a subset of C. Here, we regard C' as a bit vector, namely, $C' = (c'_1, c'_2, \ldots, c'_l)$. Moreover, both communicating parties share the key vector $K' = (k'_1, k'_2, \ldots, k'_l)$, which is a subset

of the pseudo-random key sequence K. The sender wants to hide r bits of secret messages (denoted by $M' \subset M$) into C', where $2^r - 1 \leq l$. In addition, there is a secret weight vector W shared between the sender and the receiver. The W is a column vector denoted by $W = (w_1, w_2, \cdots, w_l)^{\mathrm{T}}$, where $w_i \in Z = \{1, 2, \cdots, 2^r - 1\}$, $i = 1, 2, \ldots, l$; and each element of Z must appear in W at least once. For example $C' = (0, 1, 1, 0, 1, 1, 0, 1)$, $K' = (1, 0, 1, 0, 1, 0, 0, 1)$ and $W = (1, 2, 3, 4, 5, 6, 7, 3)$, where $r = 3$. The encoding steps are as follows:

Step 1: Calculate $(C' \oplus K') \times W$, denoted by *Sum*;

Step 2: For each $w = 1, 2, \ldots, 2^r - 1$, define the set $S_w = \{i | ([W]_i = w \wedge [C' \oplus K']_i = 0) \vee ([W]_i = 2^r - w \wedge [C' \oplus K']_i = 1)\}$. We find that S_w includes some element indices of C'. If any bit whose index is included in S_w is flipped, the result of Step 1, *Sum*, will be increased by w or decreased by $2^r - w$ (namely, increased by w under mod 2). In the following discussion, we do not distinguish between S_{w_1} and S_{w_2} if $w_1 \bmod 2 = w_2 \bmod 2$. Further, it has been proved that only one of S_w and $S_{2^r - w}$ is a null set at most, i.e., $S_w = \emptyset \Rightarrow S_{2^r - w} \neq \emptyset$; and $S_{2^r - w} \neq \emptyset$ [23].

Step 3: Calculate $d \equiv (M')_{10} - Sum \pmod{2^r}$, where $(M')_{10}$ denotes the decimal value of r bits of secret messages M'. If $d = 0$, then C' need not to be modified; otherwise, the following steps are executed to transform C' to C'^*: (a) choose an $h \in \{1, 2, \ldots, 2^r - 1\}$, such that $S_{hd} \neq \emptyset$ and $S_{-(h-1)d} \neq \emptyset$; specially, we define $S_0 \neq \emptyset$, which means no bits need to be flipped. Accordingly, if $S_d \neq \emptyset$, we can choose S_d and S_0; (b) randomly select an index $i \in S_{hd}$ and flip the bit c'_i; and (c) randomly select an index $j \in S_{-(h-1)d}$ and flip the bit c'_j. To sum up, if $d = 0$, no bits in C' need to be flipped; if $d \neq 0$ and $S_d \neq \emptyset$, only one bit in C' needs to be flipped; if $d \neq 0$ and $S_d = \emptyset$, two non-empty sets $S_{hd}$and $S_{-(h-1)d}$ should be selected. In other words, two bits in C' need to be flipped.

For the example given above, $Sum = 9$, $C' \oplus K' = (1, 1, 0, 0, 0, 1, 0, 0)$; if $M' =$ "001"$((M')_{10} = 1)$, then $d = 0$, namely, no bits in C' need to be changed; if $M' =$ "101"$((M')_{10} = 5)$, then $d = 4$. Since $S_4 = \{4\}$, we can flip c'_4, accordingly, we can obtain $C'^* = (0, 1, 1, 1, 1, 1, 0, 1)$; if $M' =$ "010"$((M')_{10} = 2)$, then $d = 1$; since $S_1 = \emptyset$, we have to choose another two non-empty sets. It is not hard to find that (S_2, S_7), (S_3, S_6) and (S_4, S_5) are alternatives. Typically, if we choose $S_2 = \{6\}$ and $S_7 = \{1, 7\}$, then we can obtain $C'^* = (1, 1, 1, 0, 1, 0, 0, 1)$ or $(0, 1, 1, 0, 1, 0, 1, 1)$.

Correspondingly, at the receiving side, the receiver can first extract C'^* from the VoIP stream and obtain M' by calculating $(C'^* \oplus K') \times W \pmod{2^r}$.

To analyze the performance of the TSENG algorithm, let us assume that C is divided into many vectors, such as C'_1, $C'_2, \ldots, C'_Q$, and l_i is the length of C'_i, so that

$$N = \sum_{i=1}^{Q} l_i. \tag{17}$$

Then the maximum embedding capacity is

$$|M|_{\max} = \sum_{i=1}^{Q} \lfloor \log_2(l_i + 1) \rfloor. \tag{18}$$

Accordingly, the maximum ER is

$$\mathrm{ER}_{\max} = \frac{\sum_{i=1}^{Q} \lfloor \log_2(l_i + 1) \rfloor}{\sum_{i=1}^{Q} l_i}. \tag{19}$$

Since the BCR of C'_i is no more than $2/l_i$, the total BCR satisfies the following relation:

$$\mathrm{BCR} = \frac{\sum_{i=1}^{Q} \mathrm{Dis}(C'_i, C'^*_i)}{\sum_{i=1}^{Q} l_i} < \frac{2Q}{\sum_{i=1}^{Q} l_i} \tag{20}$$

Moreover, the total EBCR satisfies the following relation,

$$\mathrm{EBCR} = \frac{\sum_{i=1}^{Q} Dis(C'_i, C'^*_i)}{\sum_{i=1}^{Q} \lfloor \log_2(l_i + 1) \rfloor} < \frac{2Q}{\sum_{i=1}^{Q} \lfloor \log_2(l_i + 1) \rfloor}. \tag{21}$$

From these equations, we can learn that the sizes of vectors have great impacts on both the embedding capacity and the induced embedding distortion. Hence, an optimal balance between them may be achieved by properly setting the values of vector sizes.

3.4. ME strategy

Westfeld [24] first introduced the hamming-code-based ME strategy into his F5 algorithm for steganography in JPEG images, which can embed l bits into $2^l - 1$ pixels with not more than 1 bit changed. In VoIP-based scenarios, let us assume that r bits of messages $M' = \{m'_1, m'_2, \ldots, m'_r\}$ will be embedded into a given LSB group $C' = \{c'_1, c'_2, \ldots, c'_l\}$, where $l = 2^r - 1$, $M' \subset M$ and $C' \subset C$. It is well known that the secret message is often encrypted in advance. Here we assume that M is the ciphertext of secret messages for ease of presentation. We can employ this ME strategy by the following steps:

Step 1: Assign the dependencies with the binary coding of i to c'_i; regard each binary coding as a column vector $B_i = (b_{i1}, b_{i2}, \ldots, b_{ir})^{\mathrm{T}}$, where

$$i = \sum_{j=1}^{r} b_{ij} \cdot 2^{(j-1)} \quad b_{ij} = 0 \text{ or } 1. \tag{22}$$

The encoding matrix A consists of all these vectors, namely,

$$A = (B_1, B_2, \ldots, B_l) = \begin{bmatrix} b_{11} & b_{21} & \ldots & b_{l1} \\ b_{12} & b_{22} & \ldots & b_{l2} \\ \vdots & \vdots & \ldots & \vdots \\ b_{1r} & b_{2r} & \ldots & b_{lr} \end{bmatrix}. \quad (23)$$

Step 2: For each row in A, calculate

$$x_j = \begin{cases} 0, & m'_j = \oplus_{i=1}^{l}(c'_i \cdot b_{ij}) \\ 1, & m'_j \neq \oplus_{i=1}^{l}(c'_i \cdot b_{ij}) \end{cases} \quad 1 \leq j \leq r, \quad (24)$$

where $\oplus_{i=1}^{l}$ represents continuous XOR operations.

Step 3: Calculate the following expression:

$$X = \sum_{j=1}^{r} x_j \cdot 2^{j-1}. \quad (25)$$

If $X = 0$, there are no bits needed to be modified in C'; otherwise, the X-th bit c'_X needs to be flipped, namely, $C'^* = \{c'_1, c'_2, \ldots, 1 - c'_X, \ldots, c'_l\}$.

The following example concretely shows the steps. Assume $r = 3$, $l = 7$, $M' = \{0, 0, 1\}$, $C' = \{0, 1, 1, 0, 0, 1, 1\}$. According to Step 1, we can get the encoding matrix

$$A = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

Further, according to Equation (24), we can obtain $x_1 = 0$, $x_2 = 0$, $x_3 = 1$. Due to $X = 0 \times 1 + 0 \times 2 + 1 \times 4 = 4$, c'_4 needs to be flipped, namely, $C'^* = \{0, 1, 1, 1, 0, 1, 1\}$.

The restituting approach is very simple. The receiver first extracts C'^* from the VoIP stream, and obtains each bit of embedded messages by calculating the following expression:

$$m'_j = \oplus_{i=1}^{l}(c'^*_i \cdot b_{ij}) \quad 1 \leq j \leq r. \quad (26)$$

For the given example, we can easily restitute $m'_1 = 0$, $m'_2 = 0$ and $m'_3 = 1$, namely, $M' = \{0, 0, 1\}$.

For each LSBs group, the ER can be calculated as follows:

$$\text{ER} = \frac{r}{l} = \frac{r}{2^r - 1}. \quad (27)$$

Further, we can obtain the mathematical expectation of $\text{Dis}(C', C'^*)$ as follows,

$$E(\text{Dis}(C', C'^*)) = \frac{l}{l+1} = \frac{2^r - 1}{2^r}. \quad (28)$$

Accordingly, the mathematical expectation of the BCR is

$$E(\text{BCR}) = \frac{\text{Dis}(C', C'^*)}{l} = \frac{1}{2^r}. \quad (29)$$

TABLE 1. The concrete embedding performance of ME strategy.

r	$l = 2^r - 1$	ER (%)	E(BCR) (%)	E(EBCR) (%)
1	1	100.00	50.00	50.00
2	3	66.67	25.00	37.50
3	7	42.86	12.50	29.17
4	15	26.67	6.25	23.44
5	31	16.13	3.13	19.38
6	63	9.52	1.56	16.41
7	127	5.51	0.78	14.17
8	255	3.14	0.39	12.45
9	511	1.76	0.20	11.09
10	1023	0.98	0.10	9.99

And, the mathematical expectation of the EBCR is

$$E(\text{EBCR}) = \frac{\text{Dis}(C', C'^*)}{r} = \frac{2^r - 1}{r \cdot 2^r}. \quad (30)$$

Table 1 shows the steganographic performances for different values of r. From these data we can learn that both the average BCR and the average EBCR decrease with the ER; that is, the ME strategy enhances the embedding transparency at the cost of a decreased ER. Furthermore, the size of the involved matrix increases exponentially with the decrease of the ER, which means that the improvement of the embedding transparency will also induce a significant increase in the computing complexity.

To further analyze the steganographic performance of the ME strategy in the whole embedding process, let us assume that M is divided into Q groups, such as $M'_1, M'_2, \ldots, M'_Q$, and r_i denotes the length of M'_i, and so

$$L = \sum_{i=1}^{Q} r_i. \quad (31)$$

In order to embed the entire M into C, the length of C must meet the following requirement:

$$N \geq \sum_{i=1}^{Q} 2^{r_i} - 1. \quad (32)$$

Accordingly, the practical ER must meet the following relation:

$$\frac{L}{N} \leq \text{ER} \leq \frac{\sum_{i=1}^{Q} r_i}{\sum_{i=1}^{Q} 2^{r_i} - 1}. \quad (33)$$

Further, the mathematical expectation of the whole BCR is

$$E(\text{BCR}_{\text{whole}}) = \frac{\sum_{i=1}^{Q}(1 - 2^{-r_i})}{N}. \quad (34)$$

And the mathematical expectation of the whole EBCR is

$$E(\mathrm{EBCR}_{\mathrm{whole}}) = \frac{\sum_{i=1}^{Q}(1-2^{-r_i})}{\sum_{i=1}^{Q} r_i}. \quad (35)$$

Clearly, if $r_1 = r_2 = \cdots = r_Q$, then $E(\mathrm{BCR}_{\mathrm{whole}}) = E(\mathrm{BCR}_1) = E(\mathrm{BCR}_2) = \cdots = E(\mathrm{BCR}_Q)$ and $E(\mathrm{EBCR}_{\mathrm{whole}}) = E(\mathrm{EBCR}_1) = E(\mathrm{EBCR}_2) = \cdots = E(\mathrm{EBCR}_Q)$. In addition, we can strike a good balance between the embedding capacity and the induced embedding distortion by properly setting the value of each r_i.

4. ENCODING STRATEGIES BASED ON DIGITAL LOGIC

4.1. Motivation and principle

As mentioned above, previous approaches popularly employ the strategy of trading the embedding capacity for the embedding transparency. However, due to the significance of the embedding capacity for VoIP-based steganography, their applications in VoIP-based steganography scenarios are limited. Therefore, we are motivated to study some new approaches that can enhance the embedding transparency without sacrificing the embedding capacity for VoIP-based steganography. Our main idea is to increase the similarity between the cover and the embedding message (embedding similarity) by virtue of a given set of transforms, denoted by $F = \{f_1, f_2, \ldots, f_s\}$, where s is the number of the transforms. Let us assume that the covert message M is to be embedded into the cover C. In order to obtain the best similarity, we perform all transforms in F on M and get the set $F(M) = \{f_1(M), f_2(M), \ldots, f_s(M)\}$. Then we evaluate the similarity between C and each element in $F(M)$, and choose the optimal element $f(M)$ with the best similarity. Accordingly, we can embed $f(M)$ into C to obtain the best transparency. Obviously, we should extract M by performing the inverse transform of transform f at the receiver side, which suggests that each adopted transform $f_i \in F$ must have the inverse transform f_i^{-1} and consequently $f_i^{-1}(f_i(M)) = M$. In addition, the computing complexities of the transforms must be small enough to meet the requirement of real-time services, which is why we employ the digital logic as the encoding transforms. In a real-time application, we often adopt the 'divide and rule' strategy, because M is often too large to be considered as a group. In other words, we divide C and M into N parts, respectively, i.e. $C = \{C'_1, C'_2, \ldots, C'_N\}$ and $M = \{M'_1, M'_2, \ldots, M'_N\}$, where $C'_i = \{c'_{i1}, c'_{i2}, \ldots, c'_{il(i)}\}$, $M'_i = \{m'_{i1}, m'_{i2}, \ldots, m'_{il(i)}\}$, and $l(i)$ is the length of the ith part, $i = 1, 2, \ldots, N$. Generally, the length of C is greater than or equal to the length of M. Here, we assume that their lengths are equal for the convenience of the description; that is, the lengths satisfy the following equation:

$$L_M = L_C = \sum_{i=1}^{N} l(i), \quad (36)$$

where L_M and L_C are the lengths of M and C, respectively. Accordingly, the transforms will be performed in parts and PSV (see Section 2) will be employed to evaluate the embedding similarity. In what follows, we present our digital logic-based encoding strategies in detail.

4.2. Strategy 1: Encoding based on logical operations

As is well known, the common logical operations include AND, OR, XOR and NOT. However, we can only choose XOR and NOT, because the other two have no inverse operations. Moreover, since XOR is performed between two elements, we need to introduce a PRS as the reference sequence (RS) when using it. Typical PRS candidates include the m-sequence [16, 27], chaotic sequence [28], PRS generated by toral automorphisms [29], etc. We denote the RS as $S = \{S'_1, S'_2, \ldots, S'_N\}$ and still assume that the length of S (denoted by L_S) is equal to the length of M and C, i.e. $L_S = L_M = L_C$. In addition, we define a unary operation ORI for completeness, of which the result is the operated element itself, namely, $\mathrm{ORI}(X) = X$, where X is a binary sequence. For $M'_i \in M$ and $S'_i \in S$, we can obtain a value set (denoted by A) of M'_i and S'_i under the operations of set $O = \{\mathrm{ORI}, \mathrm{NOT}, \mathrm{XOR}\}$. Clearly, $A = \{\mathbb{O}, I, M'_i, S'_i, \mathrm{NOT}(M'_i), \mathrm{NOT}(S'_i), \mathrm{XOR}(M'_i, S'_i), \mathrm{XNOR}(M'_i, S'_i)\}$, where $\mathbb{O} = \{x_j = 0 | j = 1, 2, \ldots, l(i)\}$, $I = \mathrm{NOT}(\mathbb{O}) = \{x_j = 1 | j = 1, 2, \ldots, l(i)\}$ and $\mathrm{XNOR}(M'_i, S'_i) = \mathrm{NOT}(\mathrm{XOR}(M'_i, S'_i))$. In fact, A includes all transform candidates for M'_i and S'_i based on operations of set O, which will be elaborated by the following lemmas.

LEMMA 1. *$< A, O >$ is a typical Algebraic System. In other words, set A is closed under all operations of set O.*

Proof. A is clearly closed under ORI and NOT. Thus, we shall focus on the proof that A is also closed under XOR. We adopt the exhaustive approach as follows:

- For $X \in A$, $\mathrm{XOR}(\mathbb{O}, X) = X \in A$.
- For $X \in A$, $\mathrm{XOR}(I, X) = \mathrm{NOT}(X) \in A$.
- (1) $\mathrm{XOR}(M'_i, S'_i) \in A$;
  (2) $\mathrm{XOR}(M'_i, \mathrm{NOT}(M'_i)) = I \in A$;
  (3) $\mathrm{XOR}(M'_i, \mathrm{NOT}(S'_i)) = \mathrm{NOT}(\mathrm{XOR}(M'_i, S'_i)) = \mathrm{XNOR}(M'_i, S'_i) \in A$;
  (4) $\mathrm{XOR}(M'_i, \mathrm{XOR}(M'_i, S'_i)) = \mathrm{XOR}(\mathbb{O}, S'_i) = S'_i \in A$;
  (5) $\mathrm{XOR}(M'_i, \mathrm{XNOR}(M'_i, S'_i)) = \mathrm{XOR}(I, S'_i) = \mathrm{NOT}(S'_i) \in A$.
- (1) $\mathrm{XOR}(S'_i, \mathrm{NOT}(M'_i)) = \mathrm{XOR}(\mathrm{NOT}(M'_i), S'_i) = \mathrm{NOT}(\mathrm{XOR}(M'_i, S'_i)) = \mathrm{XNOR}(M'_i, S'_i) \in A$;
  (2) $\mathrm{XOR}(S'_i, \mathrm{NOT}(S'_i)) = I \in A$;
  (3) $\mathrm{XOR}(S'_i, \mathrm{XOR}(M'_i, S'_i)) = M'_i \in A$;

(4) $\mathrm{XOR}(S_i', \mathrm{XNOR}(M_i', S_i')) = \mathrm{NOT}(M_i') \in A$.

- (1) $\mathrm{XOR}(\mathrm{NOT}(M_i'), \mathrm{NOT}(S_i',)) = \mathrm{XOR}(M_i', S_i') \in A$;
  (2) $\mathrm{XOR}(\mathrm{NOT}(M_i'), \mathrm{XOR}(M_i', S_i')) = \mathrm{XOR}(I, S_i') = \mathrm{NOT}(S_i') \in A$;
  (3) $\mathrm{XOR}(\mathrm{NOT}(M_i'), \mathrm{XNOR}(M_i', S_i')) = \mathrm{XOR}(\mathbb{O}, S_i') = S_i' \in A$.
- (1) $\mathrm{XOR}(\mathrm{NOT}(S_i'), \mathrm{XOR}(M_i', S_i')) = \mathrm{XOR}(I, M_i') = \mathrm{NOT}(M_i') \in A$;
  (2) $\mathrm{XOR}(\mathrm{NOT}(S_i'), \mathrm{XNOR}(M_i', S_i')) = \mathrm{XOR}(\mathbb{O}, M_i') = M_i' \in A$.
- $\mathrm{XOR}(\mathrm{XOR}(M_i', S_i'), \mathrm{XNOR}(M_i', S_i')) = I \in A$.

To summarize, for $\forall X \in A$ and $Y \in A$, $\mathrm{XOR}(X, Y) \in A$, that is, A is closed under XOR. Therefore, we can safely conclude that A is closed under all operations of set O, i.e. $< A, O >$ is a typical Algebraic System. □

According to the encoding requirement, we can further determine four transform candidates, i.e. $\mathrm{ORI}(M_i')$, $\mathrm{NOT}(M_i')$, $\mathrm{XOR}(M_i', S_i')$ and $\mathrm{XNOR}(M_i', S_i')$, which have the following characteristics:

LEMMA 2. *For* $\forall X, Y \in T = (ORI(M_i'), NOT(M_i'), XOR(M_i', S_i'), XNOR(M_i', S_i'))$ *and* $X \neq Y$, $ORI(X) = X \in T$; $NOT(X) \in T$; $XOR(X, Y) \in T$; $XNOR(X, Y) \in T$.

This lemma can be easily proved using the same approach used for Lemma 1. Moreover, this lemma suggests again that we cannot obtain more available transforms by combining some transforms in T. Of course, the more RSs we introduce, the more transforms we can obtain. Generally speaking, if we choose n RSs, we can obtain 2^{n+1} available transforms. In order to inform the receiver of the employed transform, we set a binary code for each transform. The length of each code (denoted by L_{code}) can be determined by the following equation:

$$L_{\mathrm{code}} = \lceil \log_2(G) \rceil, \tag{37}$$

where G is the number of the adopted transforms. If we choose all the transforms in set T, then $L_{\mathrm{code}} = 2$. Figure 1 illustrates the embedding process in this case. However, it is not a good idea to introduce too many transforms. We shall explain the reason shortly in the following text.

Another key problem is how to transmit these indicating codes to the receiver. In our previous study [16], we presented a synchronization mechanism using techniques of the protocol steganography, in which synchronization patterns are hidden in the unused and/or optional fields of the header of a certain packet. In the IP header, there are a total of 64 bits that can be used to embed messages [30]. Moreover, the headers of upper-level protocols (e.g. RTP, etc.) also have many unused or optional fields. Therefore, we can distribute the indicating codes (often being encrypted) among those fields in a predetermined manner. The codes are very small in their quantity and their manners and embedded locations can be altered continually, and so such a type of covert transmission is potentially hard

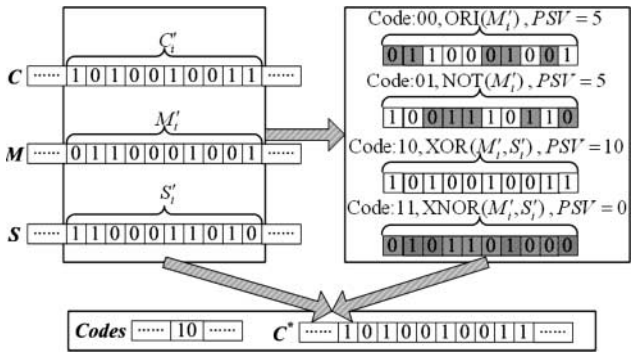

FIGURE 1. The embedding process using Strategy 1.

to discover. The receiver knows exactly how and where the indicating codes are embedded. The receiver first checks the indicating codes and further extracts the corresponding hidden parts when receiving an IP packet. After collecting all the parts, the receiver can successfully restitute the whole secret message. We give a specific example in Section 5 to illustrate this approach.

Further, multi-level steganography (MLS), recently proposed by Frączek *et al.* [31], can provide a more sophisticated alternative for this problem. MLS is a deep hiding technique, which consists of two or more steganographic methods. In MLS, one method called the upper-level method employs overt traffic as a carrier, and serves as a carrier for another method called the lower-level method. Benefiting from the relationship between the involved methods, MLS can further increase the undetectability of covert communication. To build a MLS scheme for this problem, IP steganography, RTP steganography and both of them can be used as the upper-level method. The main problem is to design a suitable lower-level method that can cooperate well with the upper-level one. This is beyond the scope of this paper, but a problem deserving our future study.

In addition, although introducing more transforms can further increase the embedding similarity, 'plenty is no plague' cannot be applied to this case, because more transforms will induce more delay for VoIP service and longer binary flag code, which may degrade the steganographic performance. Therefore, we need to determine the number of transforms according to the practical requirement of covert communication.

4.3. Strategy 2: Encoding based on shifting operations

The shifting operations, which allow bits to be moved to the left or right in a word, are often used in serial transfers of data. There are three types of shifting operations: logical, arithmetic and circular. The first two types cannot be adopted in our strategy, for they have no inverse operations that can be used to get the original data. However, as its name implies, the circular shift circulates the bits in a given word around the two ends without any loss of information, and so we can recover the

Code	Description	Code	Description	Code	Description	Code	Description
000	CSL (0)	001	CSL (1)	010	CSL (2)	011	CSL (3)
100	CSR (4)	101	CSR (3)	110	CSR (2)	111	CSR (1)

FIGURE 2. The circular shift-based transforms for 8-bit words.

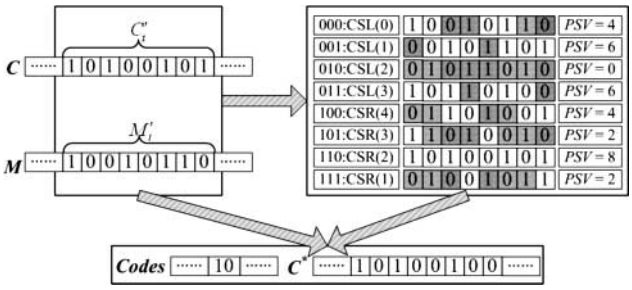

FIGURE 3. The embedding process using Strategy 2.

original data by the opposite shift. The circular shift involves the circular shift left (CSL) and the circular shift right (CSR) operations. For an x-bit word, a CSL by x–y bits (denoted by CSL(x–y), $0 \leq y \leq x$) is equivalent to a CSR by y bits (denoted by CSL(y)). Therefore, we define the circular shift-based transforms that consist of the operation codes (Op. code) and the shift numbers. Figure 2 depicts a set of transforms for 8-bit words. As the figure shows, the first 1 bit marked with underline in each code is used to represent the Op. code, namely Op. $=$ 1 represents CSR; and Op. $=$ 0 represents CSL. In this strategy, the length of each code (L_{code}) depends on the operation word (one part of the message) and can be determined as follows:

$$L_{code} = \lceil \log_2(n) \rceil, \tag{38}$$

where n is the length of the given part. The synchronization of these codes is similar to Strategy 1.

Further, we can perform the above transforms on each part, and choose their optimal transforms for the embedding process. Figure 3 illustrates a typical embedding process, in which the length of each part is 8 bit and the coding method in Fig. 2 is employed.

4.4. Strategy 3: Hybrid strategy

The above two strategies can be merged for better performance. A straightforward method may be to merge these transforms directly. Due to the difference between these transforms' characteristics, the codes of unequal length have to be adopted; that is, the codes for the transforms in Strategy 1 are tantamount to the Op. codes, and the codes for the transforms in Strategy 2 include the Op. codes and the shift numbers. However, this will incur more cost in bits to represent the code and induce an intractable synchronization problem. Therefore, the value of this method is limited. Another method is to first apply the two strategies in each packet and choose the one with the larger total PSV as the final strategy. This method needs an extra 1-bit flag to indicate which strategy is finally adopted. However, due to the option of more transforms, we can further enhance the embedding transparency.

5. IMPLEMENTATION AND EVALUATION

This section presents specific examples illustrating the proposed strategies and evaluates them in StegVoIP, which is a prototypical covert communication system based on VoIP [14–16]. StegVoIP supports typical coders, such as G.711, G.723.1, G.729a, etc. In the prototype implementations, we typically adopt G.729a as the codec of the cover speech, while the strategies can also effectively work in conjunction with other coders. Moreover, we consider the embedding process in each packet. Let the length of codes in each packet be X, and the number of LSBs be Y. If the Y LSBs are divided into n parts and the length of the ith part is l_i, for Strategy 1, we can obtain the following equations:

$$\begin{cases} 2 \cdot n = X, \\ \sum_{i=1}^{n} l_i = Y. \end{cases} \tag{39}$$

For Strategy 2, we can obtain the following equations:

$$\begin{cases} \sum_{i=1}^{n} \log_2 l_i = X, \\ \sum_{i=1}^{n} l_i = Y. \end{cases} \tag{40}$$

A typical implementation can be described as follows: the identification field of the IPv4 header can be employed to hide the codes of transforms. As proposed in [32], the high 8 bits of the identification field can be used for data hiding, so we can set $X = 8$. If we choose 8 LSBs (the bits with the least impact on the speech quality) in each G.729a frame based on the observation that the parameters of the fixed codebook have the best transparency for data hiding [15, 16], and set the payload of each packet at four frames (40 octets) in order to not induce the fragment strategy of Internet protocol, then 32 bits are available for data hiding in each packet ($Y = 32$). There are many partition methods. In this paper, for Strategy 1, LSBs are divided into four parts that contain 8 bits each and each code can be represented by 2 bits; for Strategy 2, LSBs are divided into two parts that contain 16 bits each and each code can be represented by 4 bits. For Strategy 3, we employ the second bit (DF) in the flag field, which represents 'do not fragment' [30, 32]. Because the packets are no larger than the maximum fragment size, the setting of the DF flag has no effect on the packets' behaviors. Here, DF $= 0$ indicates that Strategy 1 is adopted, and DF $= 1$ indicates that Strategy 2 is adopted. Further, if we choose 4

TABLE 2. Modes definition of Group I.

Mode	Approach	Parameters setting
LSB(8)	LSB	8 LSBs/frame
RIM(2)	RIM (Section 3.1)	8 LSBs/frame; Max interval = 2
RPM(0.5)	RPM (Section 3.2)	8 LSBs/frame; Substitution prob. = 0.5
TSENG(3)	TSENG (Section 3.3)	8 LSBs/frame; Cover vector size = 3
ME(2)	ME (Section 3.4)	8 LSBs/frame; Message vector size = 2
S1(8)	Strategy 1 (Section 4.1)	8 LSBs/frame
S2(8)	Strategy 2 (Section 4.2)	8 LSBs/frame
S3(8)	Strategy 3 (Section 4.3)	8 LSBs/frame

TABLE 3. Modes definition of Group II.

Mode	Approach	Parameters setting
RIM(4)	RIM (Section 3.1)	8 LSBs/frame; Max interval = 4
RPM(0.3)	RPM (Section 3.2)	8 LSBs/frame; Substitution prob. = 0.3
TSENG(7)	TSENG (Section 3.3)	8 LSBs/frame; Cover vector size = 7
ME(3)	ME (Section 3.4)	8 LSBs/frame; Message vector size = 3
S1(4)	Strategy 1 (Section 4.1)	4 LSBs/frame
S2(4)	Strategy 2 (Section 4.2)	4 LSBs/frame
S3(4)	Strategy 3 (Section 4.3)	4 LSBs/frame

LSBs in each G.729a, then $Y = 16$; for Strategy 1 we can also divide LSBs into four parts that contain 4 bits each and each code can be represented by 2 bits; for Strategy 2, we can also divide LSBs into two parts that contain 8 bits each and each code can be represented by 3 bits; that is, we can only use 6 bits of the high 8 bits of the identification field; and Strategy 3 can be implemented as well as the former examples. In this paper, we employ randomly generated m sequences as the PRSs used in our strategies.

In order to compare the proposed strategies and the approaches introduced in Section 3, we carry out two groups of contrasting experiments. The test modes of Group I are shown in Table 2, and that of Group II in Table 3. Each mode in Group I can achieve the maximum embedding capacity of the corresponding approach; and the modes in Group II can render better embedding transparency than corresponding modes in Group I .For all these modes, the statistical analyses are made on ER, BCR and EBCR. Further, to compare their embedding transparency, we employ the Perceptual Evaluation of Speech Quality (PESQ) method [33] to evaluate the speech quality of cover speeches and their steganographic versions. PESQ compares an original signal with a degraded signal and outputs a PESQ score as a prediction of the perceived quality. The range of the PESQ score is -0.5 to 4.5. Moreover, the PESQ score can be converted to Mean Opinion Score-Listening Quality Objective (MOS-LQO). The range of MOS-LQO is 1.017 (worst) to 4.549 (best), which more closely matches the range of the subjective Mean Opinion Score (MOS) [34].

For the experiments, we employ speech samples provided in ITU-T Rec. P. 501 Annex B [35], which are popularly used in conjunction with objective speech quality evaluation procedures. These samples consist of 10 categories, i.e. Chinese speech, English (UK) speech, English (US) speech, Finnish speech, French speech, German speech, Italian speech, Japanese speech, Polish speech and Spanish speech. Each category includes female speech and male speech. All samples are first encoded in G.729a. For each sample, the corresponding steganographic experiments are performed on its G.729a encoded file. The secret message (choosing from the introduction of Xiamen [36], possibly only some forward parts) can be successfully embedded and retrieved in any case. In the PESQ experiments, the reference signals are PCM encoded files converted from original G.729a encoded files; the degraded signals are PCM files converted from steganographic G.729a files. All the PCM files are formatted with 8000 HZ sampling rate, 16 bits quantization and mono.

Figures 4–7 show the statistical results of the mean ER, mean BCR, mean EBCR and mean MOS-LQO, respectively, for all modes in Group I. From these figures, we can observe that: (i) the speech quality of steganographic samples largely depends on the BCR; that is, the smaller the BCR value, the higher is the speech quality. Thus, all transparency-oriented encoding strategies aim to decrease the BCR value. Accordingly, they render higher speech quality than the traditional LSB method. (ii) However, the methods they use to decrease the BCR are different. The RIM and the RPM reduce the BCR by decreasing the ER. In the

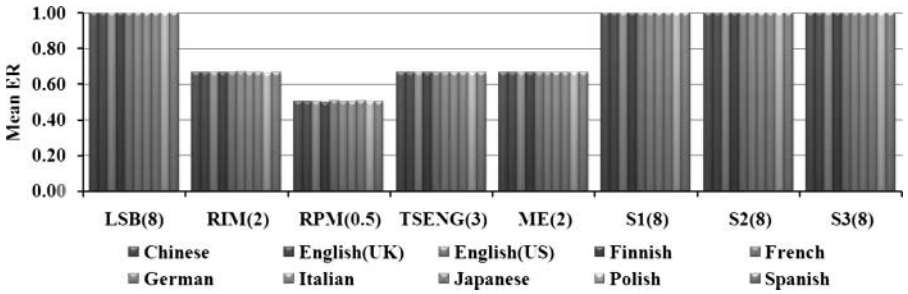

FIGURE 4. Test results of ER for Group I.

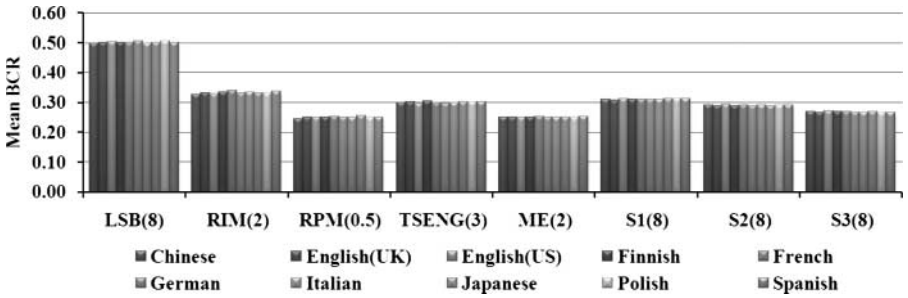

FIGURE 5. Test results of BCR for Group I.

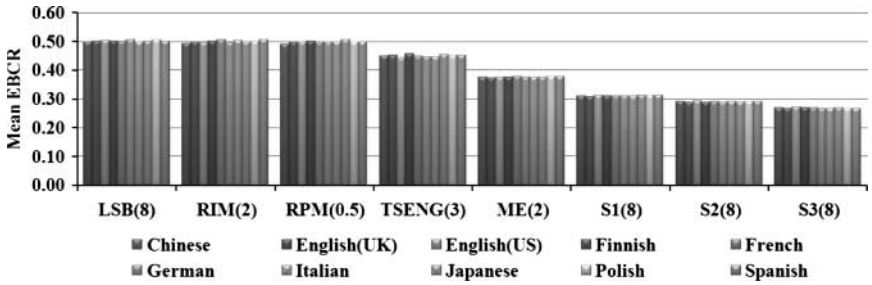

FIGURE 6. Test results of EBCR for Group I.

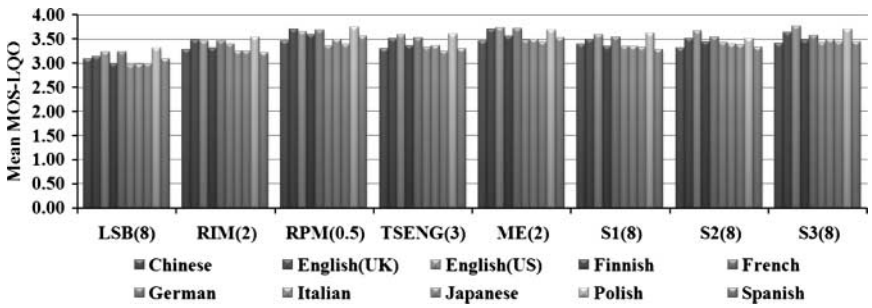

FIGURE 7. Test results of MOS-LQO for Group I.

experiments, RPM(0.5) can achieve better MOS-LQO scores than RIM(2), because RPM(0.5) sacrifices more the ER than RIM(2). TSENG and ME reduce the BCR by decreasing both the ER and the EBCR, and ME(2) can achieve better MOS-LQO scores than TSENG(3), because of its comparatively smaller EBCR. In contrast, our strategies aim to reduce the BCR by

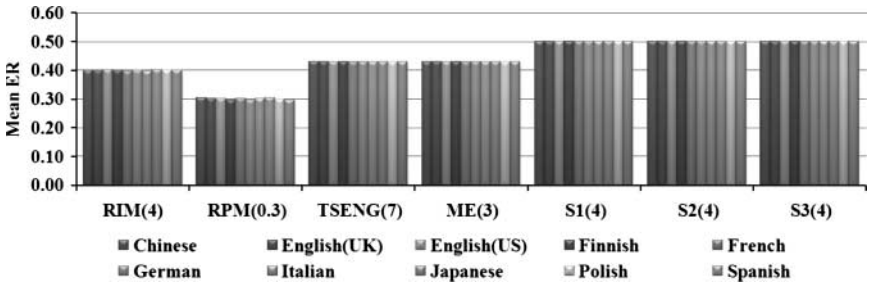

FIGURE 8. Test results of ER for Group II.

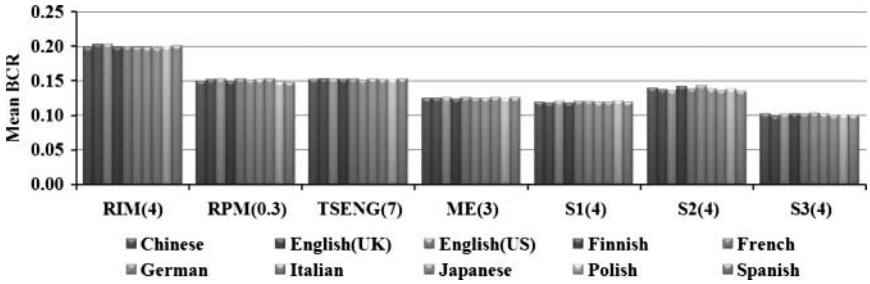

FIGURE 9. Test results of BCR for Group II.

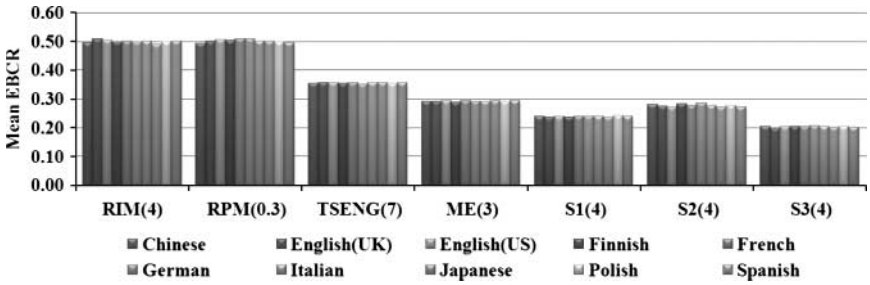

FIGURE 10. Test results of EBCR for Group II.

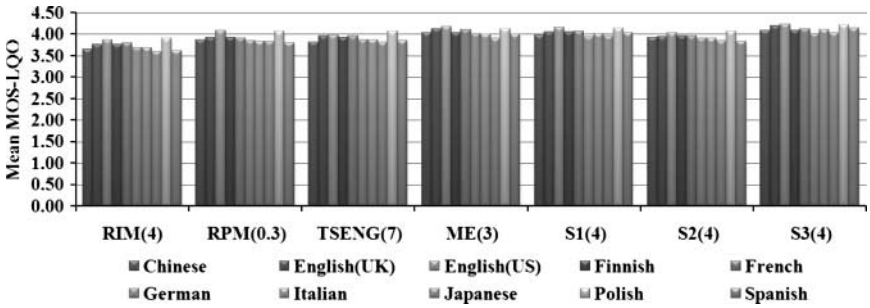

FIGURE 11. Test results of MOS-LQO for Group II.

only decreasing the EBCR. Therefore, our strategies support the maximum ER (namely, ER $=$ 1.0). However, they can also achieve comparable MOS-LQO scores with the previous methods.

Figures 8–11 show the statistical results of group II. In the group, we intentionally halve the ER of our proposed strategies to achieve better MOS-LQO scores. However, they still provide larger ER than the previous methods. Moreover,

Strategy 1 and Strategy 3 provide higher speech quality than all the previous methods; Strategy 2 achieves better MOS-LQO scores than the previous methods except for ME. However, the gap of speech quality between Strategy 2 and ME is small. Therefore, we can safely conclude that the proposed strategies can provide consistently better performances on both the ER and the embedding transparency than the previous strategies. Moreover, the reader may find that the mean MOS-LQO score of Strategy 2 is smaller than that of Strategy 1 in Group II, but the opposite is true in Group I. That difference mainly stems from difference in the adopted part size. This also suggests that the part size is an important factor, which impacts on the embedding transparency. We determine it experientially here, but we will analyze it in more depth in our future work.

In summary, our strategies reduce the embedding distortion by decreasing the EBCR instead of the ER. Therefore, compared with previous methods, our strategies can effectively enhance the embedding transparency while maintaining the maximum ER. Furthermore, by properly adjusting the ER, they can also achieve the desired steganographic transparency.

6. CONCLUSIONS AND FUTURE WORK

Transparency-orientated strategies have been proved crucial through years of practice in steganography over storage media. However, in VoIP-based scenarios, the application of the previous transparency-orientated encoding strategies proposed for steganography on storage media is limited, because they enhance embedding transparency at the expense of decreasing the ER (embedding capacity). Therefore, we are motivated to present digital-logic-based encoding strategies. Differing from the previous approaches, the proposed strategies employ digital logic operations to increase the similarity between the cover and the covert message, and thereby effectively reduce the distortion (enhance the transparency) while maintaining the maximum ER. From the experimental results, we learn that the proposed strategies can provide consistently better performances on both the ER and the embedding transparency than the previous strategies. In addition, it is easy to find that the embedding transparency can be further enhanced if we introduce more digital logic transforms. However, more transforms will likely induce higher complexity of the algorithm and more indicating codes. Therefore, one must strike a sensible trade-off among the embedding transparency, the number of required indicating codes and the complexity of the algorithm.

For our future work, we shall study the embedding impact of each bit in the cover stream. If the impact for each bit on speech quality can be exactly quantified, we may design better encoding strategies with impact weights that can further improve the steganographic transparency. Moreover, other encoding strategies aiming at improving the steganographic performance (e.g. capacity, security, undetectability and reliability) are important subjects worthy of future study.

ACKNOWLEDGEMENTS

The authors wish to thank anonymous reviewers for their valuable comments and suggestions which have largely improved this paper.

FUNDING

This work was supported in part by Natural Science Foundation of Fujian Province of China [2011J05151]; Scientific Research Foundation of National Huaqiao University [11BS210]; Fundamental Research Funds for the Central Universities [JB-ZR1131]; National High Technology Research and Development Program (863 Program) of China [2009AA01A402]; and National Basic Research Program (973 Program) of China under [2011CB302305].

REFERENCES

[1] Provos, N. and Honeyman, P. (2003) Hide and seek: an introduction to steganography. *IEEE Secur. Priv.*, **1**, 32–44.

[2] Cheddad, A., Condell, J., Curran, K. and Kevitt, P.M. (2010) Digital image steganography: survey and analysis of current methods. *Signal Process.*, **90**, 727–752.

[3] Cetin, O. and Ozcerit, A. (2009) A new steganography algorithm based on color histograms for data embedding into raw video streams. *Comput. Secur.*, **28**, 670–682.

[4] Delforouzi, A. and Pooyan, M. (2008) Adaptive digital audio steganography based on integer wavelet transform. *Circuits Syst. Signal Process.*, **27**, 247–259.

[5] Liu, T. and Tsai, W. (2007) A new steganographic method for data hiding in microsoft word documents by a change tracking technique. *IEEE Trans. Inf. Forensics Sec.*, **2**, 24–30.

[6] Goode, B. (2002) Voice over Internet protocol (VoIP). *Proc. IEEE*, **90**, 1495–1517.

[7] Lubacz, J., Mazurczyk, W. and Szczypiorski, K. (2010) Vice over IP. *IEEE Spectr.*, **47**, 42–47.

[8] Wang, C. and Wu, Q. (2007) Information Hiding in Real-Time VoIP Streams. *Proc. 9th IEEE Int. Symp. Multimedia*, Taichung, Taiwan, December 10–12, pp. 255–262. IEEE Computer Society, Piscataway, NJ, USA.

[9] Dittmann, J., Hesse, D. and Hillert, R. (2005) Steganography and Steganalysis in Voice over IP Scenarios: Operational Aspects and First Experiences with a New Steganalysis Tool Set. *Proc. SPIE*, Vol. 5681, San Jose, CA, USA, January 17–20, pp. 607–618. SPIE Press, Bellingham, WA, USA.

[10] Kratzer, C., Dittmann, J., Vogel, T. and Hillert R. (2006) Design and Evaluation of Steganography for Voice-over-IP. *Proc. IEEE Int. Symp. Circuits and Systems*, Island of Kos, Greece, May 21–24, pp. 2397–2340. IEEE Circuits and Systems Society, Piscataway, NJ, USA.

[11] Huang, Y., Xiao, B. and Xiao, H. (2008) Implementation of Covert Communication Based on Steganography. *Proc. Int. Conf. Intelligent Information Hiding and Multimedia Signal Processing*, Harbin, China, August 15–17, pp. 1512–1515. IEEE Computer Society, Piscataway, NJ, USA.

[12] Mazurczyk, W. and Lubacz, J. (2010) LACK—a VoIP steganographic method. *Telecommun. Syst. J.*, **45**, 153–163.

[13] Huang, Y., Tang, S. and Yuan, J. (2011) Steganography in inactive frames of VoIP streams encoded by source codec. *IEEE Trans. Inf. Forensics Sec.*, **6**, 296–306.

[14] Tian, H., Zhou, K., Huang, Y., Liu, J. and Feng, D. (2008) A Covert Communication Model Based on Least Significant Bits Steganography in Voice over IP. *Proc. 9th IEEE Int. Conf. for Young Computer Scientists*, Zhangjiajie, China, November 18–21, pp. 647–652. IEEE Computer Society, Piscataway, NJ, USA.

[15] Tian, H., Zhou, K., Jiang, H., Huang, Y., Liu, J. and Feng, D. (2009) An Adaptive Steganography Scheme for Voice over IP. *Proc. IEEE Int. Symp. Circuits and Systems*, Taipei, Taiwan, May 24–27, pp. 2922–2925. IEEE Circuits and Systems Society, Piscataway, NJ, USA.

[16] Tian, H., Zhou, K., Jiang, H., Huang, Y., Liu, J. and Feng D. (2009) An m-Sequence Based Steganography Model for Voice over IP. *Proc. 44th IEEE Int. Conf. Communications*, Dresden, Germany, June 14–18, pp. 1–5. IEEE Communications Society, Piscataway, NJ, USA.

[17] Fridrich, J. (2006) Minimizing the Embedding Impact in Steganography. *Proc. 8th ACM Workshop on Multimedia & Security*, Geneva, Switzerland, September 26–27, pp. 2–10. ACM Press, New York, NY, USA.

[18] Fridrich, J., Tevný, T. and Kodovský, J. (2007) Statistically Undetectable JPEG Steganography: Dead Ends Challenges, and Opportunities. *Proc. 9th ACM Workshop on Multimedia & Security*, Dallas, TX, USA, September 20–21, pp. 3–14. ACM Press, New York, NY, USA.

[19] Möller, S., Pfitzmann, A. and Stirand, I. (1996) Computer Based Steganography: How it Works and Why Therefore Any Restrictions on Cryptography are Nonsense, at Best. *Proc. 1st Int. Workshop Information Hiding*, Lecture Notes in Computer Science 1174, Cambridge, UK, May 30–June 1, pp. 7–21. Springer, Berlin.

[20] Aura, T. (1996) Practical Invisibility in Digital Communication. *Proc. 1st Int. Workshop Information Hiding*, Lecture Notes in Computer Science 1174, Cambridge, UK, May 30–June 1, pp. 265–278. Springer, Berlin.

[21] Katzenbeisser, S. and Petitcolas, F.A.P. (1999) *Information Hiding Techniques for Steganography and Digital Watermarking*. Artech House Press, Boston, USA.

[22] Bo, S., Hu, Z., Wu, L. and Zhou, D. (2005) *Steganography of Telecommunication Information*. National defense industry Press, Beijing, China.

[23] Tseng, Y., Chen, Y. and Pan, H. (2002) A secure data hiding scheme for binary images. *IEEE Trans. Commun.*, **50**, 1227–1231.

[24] Westfeld, A. (2001) F5—a Steganographic Algorithm. *Proc. 4st Int. Workshop Information Hiding*, Lecture Notes in Computer Science 2137, Pittsburgh, PA, USA, April 25–27, pp. 289–302. Springer, Berlin.

[25] Fridrich, J. and Soukal, D. (2006) Matrix embedding for large payloads. *IEEE Trans. Inf. Forensics Sec.*, **1**, 390–394.

[26] Khatirinejad, M. and Lison¡ek, P. (2009) Linear codes for high payload steganography. *Discrete Appl. Math.*, **157**, 971–981.

[27] Engelberg, S. and Benjamin, H. (2005) Pseudorandom sequences and the measurement of the frequency response. *IEEE Instrum. Meas. Mag.*, **8**, 54–59.

[28] Tefas, A., Nikolaidis, A., Nikolaidis, N., Solachidis, V., Tsekeridou, S. and Pitas, I. (2001) Statistical Analysis of Markov Chaotic Sequences for Watermarking Applications. *Proc. IEEE Int. Symp. Circuits and Systems*, Sydney, Australia, May 6–9, pp.57–60. IEEE Circuits and Systems Society, Piscataway, NJ, USA.

[29] Voyatzis, G. and Pitas, I. (1996) Applications of Toral Automorphisms in Image Watermarking. *Proc. IEEE Int. Conf. Image Processing*, Lausanne, Switzerland, September 16–19, pp. 237–240. IEEE Signal Processing Society, Piscataway, NJ, USA.

[30] Murdoch, S.J. and Lewis, S. (2005) Embedding Covert Channels into TCP/IP. *Proc. 7th Int. Workshop Information Hiding*, Lecture Notes in Computer Science 3727, Barcelona, Spain, June 6–8, pp. 247–262. Springer, Berlin.

[31] Frączek, W., Mazurczyk, W. and Szczypiorski K. (2011) Multi-level Steganography Applied to Networks. *Proc. of 3rd Int. Workshop on Network Steganography co-located with 10th Int. Conf. Telecommunication Systems, Modeling and Analysis*, Prague, Czech Republic, May 26–28, pp. 90–96. ATSMA, Dallas, TX, USA.

[32] Ahsan, K. and Kundur, D. (2002) Practical Data Hiding in TCP/IP. *Proc. 5th ACM Workshop on Multimedia & Security*, Juan-les-Pins, France, December 6, pp. 63–70. ACM Press, New York, NY, USA.

[33] ITU-T. P.862 (2001) *Perceptual Evaluation of Speech Quality (PESQ): An Objective Method for End-to-end Speech Quality Assessment of Narrow-Band Telephone Networks and Speech Codecs*. International Telecommunications Union, Geneva, Switzerland.

[34] ITU-T. P.800 (1996) *Methods for Subjective Determination of Transmission Quality*. International Telecommunications Union, Geneva, Switzerland.

[35] ITU-T. P.501 (2007) *Test Signals for use in Telephonometry*. International Telecommunications Union, Geneva, Switzerland.

[36] DRWI (2010) *Xiamen overview*. http://www.chinatravel.com/fujian/xiamen/overview.htm (accessed June 20, 2011).