Multimed Tools Appl (2016) 75:11493–11512
DOI 10.1007/s11042-015-2865-1

Incorporating data hiding into G.729 speech codec

Shufan Yan[1] · **Guangming Tang**[2] · **Yanling Chen**[1]

Received: 14 December 2014 / Revised: 23 June 2015 / Accepted: 5 August 2015 /
Published online: 14 August 2015

Abstract The rapid development of speech communication technology has made it possible for low bit-rate speech to become appropriate steganographic cover media. To incorporate data hiding into the low bit-rate speech codec, a novel steganography algorithm is proposed in this paper. By analyzing the encoding rule of fixed codebook vector, the way of transposing encoding locations of adjacent pulses is found to be suitable for data embedding with good imperceptibility. Based on encoding location transposition of adjacent pulses, the relationship between adjacent pulse locations is used to embed secret data while the fixed codebook search is being conducted during the encoding process of G.729 codec, which can maintain synchronization between data embedding and speech encoding. The experimental results demonstrate that the proposed steganography algorithm performs well in imperceptibility with a hiding capacity of 550 bits/s. Furthermore, the real-time and anti-detection performances are also satisfactory.

Keywords G.729 codec · Low bit-rate speech · Steganography · Fixed codebook search

1 Introduction

Nowdays increasingly great importance has been attached to the information security problem. To make the information transmission more secure and reliable, digital steganography, a technique of covert communication, has emerged and developed quickly. By embedding secret information into public digital media, steganography conceals both the information content and the information transmission behavior.

With the development of network and multimedia technology, low bit-rate codecs are widely applied to speech communication systems such as voice over Internet protocol (VoIP) service. As a result, steganography in low bit-rate speech streams has attracted interests

✉ Shufan Yan
yansfluk@163.com

[1] Zhengzhou Information Science and Technology Institute, Zhengzhou, People's Republic of China

[2] Department of Information Security, Zhengzhou Information Science and Technology Institute, Zhengzhou, People's Republic of China

of researchers [7, 9, 15]. Compared to the traditional steganography, the low bit-rate speech steganography has the following advantages. First, speech data is generated and transmitted in real time during the communication process, which improves steganography security by providing insufficient time to perform steganalysis. Second, the data volume of continuous speech communication is much greater than individual image or text, hence it is possible for large-scale secret information transmission. Third, speech data is hard to recognize from the enormous data over the Internet, let alone those with secret data embedded.

Due to the advantages of taking low bit-rate speech streams as covers to perform steganography, there have been some attempts in this area. For optionally regulating hiding capacity and imperceptibility, Tian et al. [13] proposed a dynamic ME method for VoIP steganography. By using Compounded Pseudorandom Sequence, a VoIP-based covert communication scheme was proposed in [14]. Liu et al. [6] improved the traditional LSB method and proposed the LSD (least significant digit) method. Inspired by the "Hamming+1" scheme, Yan et al. [17] proposed a triple-layer steganography scheme for low bit-rate speech streams. To improve the security of the QIM (quantization index modulation) steganography in low bit-rate speech streams, a modified QIM steganography algorithm which selected the hiding positions randomly to adjust the embedding rate was proposed in [11]. Huang et al. [2] proposed a steganography algorithm based on the pitch period prediction process, which incorporated data hiding into G.723.1 codec. It is really an excellent work to combine data hiding with speech encoding, but the proposed algorithm can only be used in G.723.1 codec. To extend the application range of speech steganography in other low bit-rate codecs, we focus on incorporating data hiding into G.729 codec.

G.729 codec is one of the most representative low bit-rate codecs. Owing to the wide application of G.729 codec in a variety of speech communication systems, steganography aiming at G.729 codec has been explored. The abilities of parameters in the G.729 speech frame for carrying secret data were analyzed using a noisy resistance model in [8]. The study shows that the fixed codebook parameter is appropriate for data embedding. On this basis, Tian et al. took the fixed codebook parameters as cover data to design steganography algorithms [10, 12]. In [12], in order to solve the synchronization problem, some fields in the IP header were selected to embed the synchronization parameters. In [10], by introducing the notion of partial similarity value (PSV), an adaptive partial-matching steganography for VoIP was presented. Wu et al. [16] proposed a steganography algorithm to embed secret data into G.729 coding speech by adapting the techniques of covering code and the interleaving, which achieved high a hiding capacity. The fixed codebook parameter takes relatively more bits in G.729 speech frame, so it affords an approach to achieve high hiding capacity when being used to embed data. However, most of the steganography algorithms based on fixed codebook search process perform embedding by substituting the least significant bits (LSBs) of fixed codebook parameters. The way of direct substitution will cause large pulse displacements in the fixed codebook vector. Therefore, the imperceptibility remains to be improved.

In order to improve the imperceptibility of speech steganography in G.729 codec, a steganography algorithm for G.729 codec is presented in this paper. First, by analyzing the encoding rule, we prove that transposing encoding locations of adjacent pulses can reduce pulse displacements. Second, based on transposing encoding locations of adjacent pulses, a steganography algorithm is proposed by modifying the relationship between pulse locations, which incorporates data embedding into the fixed codebook search process in G.729 codec.

The rest of the paper is organized as follows. Section 2 analyzes the fixed codebook search process in G.729 codec. A novel speech steganography algorithm is presented in Section 3. Section 4 shows the experimental results. The conclusions are summarized in Section 5.

2 Analysis of fixed codebook search process in G.729 codec

ITU-T G.729 is a kind of speech compression technology based on conjugate structure algebraic code excited linear prediction (CS-ACELP) algorithm. For each speech frame of 10 milliseconds, G.729 codec extracts CELP parameters (linear prediction parameters, adaptive and fixed codebook parameters, adaptive and fixed codebook gains) and then encodes them for transmission. The fixed codebook vector is an important part of the exciting signal. It serves as an approximation to the residual signal after short-time and long-time predictions, which can provide a precise error compensation to help improving the speech quality.

2.1 Fixed codebook search process in G.729 codec

During G.729 encoding process, the fixed codebook vector is searched every subframe (a frame is divided into two subframes). The fixed codebook structure is shown in Table 1.

The fixed codebook vector $c(n)$ contains four unit pulses.

$$c(n) = s_0\delta(n-i_0) + s_1\delta(n-i_1) + s_2\delta(n-i_2) + s_3\delta(n-i_3) \tag{1}$$

Where $\delta(0)$ denotes the unit pulse.

When using the full search algorithm to search the fixed codebook vector, the search process is composed of four nested loops, with each nested loop searching for one pulse location. In order to get the optimal pulse location, the computation times needed are $2^3\times2^3\times2^3\times2^4=8192$. To decrease the computational complexity, G.729 adopts the centralized search algorithm. It simplifies the search process by restricting the entering times into the fourth nested loop. Let $\max_3$ and av_3 be the largest absolute correlation degree and the average correlation degree, respectively. The threshold thr_3 is calculated first:

$$thr_3 = av_3 + k(\max_3 - av_3) \tag{2}$$

$k$equals to 0.4. The last nested loop is effective only if the absolute correlation degree of the first three pulses c_3 is greater than thr_3. The block diagram of the centralized search algorithm is shown in Fig. 1.

Table 1 G.729 fixed codebook structure

Pulse	Sign	Location								
τ_0	s_0±1	i_0	0	5	10	15	20	25	30	35
τ_1	s_1±1	i_1	1	6	11	16	21	26	31	36
τ_2	s_2±1	i_2	2	7	12	17	22	27	32	37
τ_3	s_3±1	i_3	3	8	13	18	23	28	33	38
			4	9	14	19	24	29	34	39

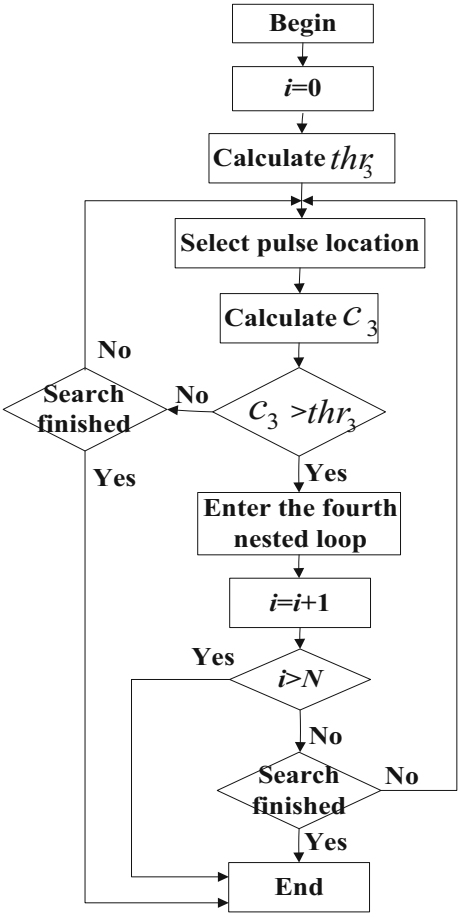


Fig. 1 The block diagram of the centralized search algorithm

N is the maximum search time of each subframe allowed. In the worst situation where N= 90, the total computation times in each subframe are 90×16=1440. Therefore, only 1440/8192=17.6% of the codebook space is searched.

Though the centralized search algorithm decreases the computation complexity by non-exhaustive way, the fixed codebook vector obtained is just local optimal because of the restriction to the search range. Consequently, there are some redundancies in the fixed codebook parameters, to which minor modifications won't cause significant loss of speech quality.

2.2 Fixed codebook vector encoding

The locations of the first three pulses τ_0, τ_1 and τ_2 are encoded with 3 bits respectively, with signs encoded to 1 bit respectively. The location of the fourth pulse τ_3 is encoded with 4 bits, with signs encoded with1 bit. Let C denote the fixed codebook parameter.

$$C = \left\lfloor \frac{i_0}{5} \right\rfloor + 8\left\lfloor \frac{i_1}{5} \right\rfloor + 64\left\lfloor \frac{i_2}{5} \right\rfloor + 512\left(2\left\lfloor \frac{i_3}{5} \right\rfloor + jx\right)$$
$$jx = \begin{cases} 0 & if\ i_3 \bmod 5 = 3 \\ 1 & if\ i_3 \bmod 5 = 4 \end{cases} \tag{3}$$

Where $i_j(j=0,1,2,3)$ denotes the pulse location. Let S denote the sign parameter.

$$S = s_0 + 2s_1 + 4s_2 + 8s_3 \tag{4}$$

Where $s_j=1$ denotes the sign is positive and $s_j=0$ denotes the sign is negative. Finally, transform C and S to binary format.

Two important theorems about transposing encoding locations of adjacent pulses in the fixed codebook vector are given below.

Theorem 1 Let τ_a and τ_b be any two adjacent pulses of the first three pulses in the fixed codebook vector. Transpose the encoding locations of τ_a and τ_b, the fixed codebook vector will remain unchanged or the two pulses τ_a and τ_b shift one unit respectively.

Proof Take τ_0 and τ_1 for example. Assume that C' and S' are parameters obtained after transposing their encoding locations.

$$C' = \left\lfloor \frac{i_1}{5} \right\rfloor + 8\left\lfloor \frac{i_0}{5} \right\rfloor + 64\left\lfloor \frac{i_2}{5} \right\rfloor + 512\left(2\left\lfloor \frac{i_3}{5} \right\rfloor + jx\right) \tag{5}$$

$$S' = s_1 + 2s_0 + 3s_2 + 4s_3 \tag{6}$$

In the decoder, the locations and signs of τ_0 and τ_1 are obtained by decoding C' and S'.

$$i_0{}' = \left[\sum_{i=4}^{6} LSB^{(i)}(C')\cdot 2^{i-4}\right] \times 5 + 1 \tag{7}$$

$$s_0{}' = LSB^{(2)}(S') \tag{8}$$

$$i_1{}' = \left[\sum_{i=1}^{3} LSB^{(i)}(C')\cdot 2^{i-1}\right] \times 5 \tag{9}$$

$$s_1{}' = LSB^{(1)}(S') \tag{10}$$

$LSB^{(i)}(C')$ denotes the *ith* lowest bit of C'.

Then discuss the following two cases.

(1)

$$\lfloor i_0/5 \rfloor \neq \lfloor i_1/5 \rfloor$$

$$|i_0-i_0'| = \left| \left[\sum_{i=1}^{3} LSB^{(i)}(C)\cdot 2^{i-1} \right] \times 5 - \left[\sum_{j=4}^{6} LSB^{(i)}(C')\cdot 2^{i-4} \right] \times 5 - 1 \right| = 1 \tag{11}$$

$$|s_0-s_0'| = \left| LSB^{(1)}(S) - LSB^{(2)}(S') \right| = 0 \tag{12}$$

$$|i_1-i_1'| = \left| \left[\sum_{i=4}^{6} LSB^{(i)}(C)\cdot 2^{i-4} \right] \times 5 + 1 - \left[\sum_{i=1}^{3} LSB^{(i)}(C')\cdot 2^{i-1} \right] \times 5 \right| = 1 \tag{13}$$

$$|s_1-s_1'| = \left| LSB^{(2)}(S) - LSB^{(1)}(S') \right| = 0 \tag{14}$$

Let $c'(n)$ be the fixed codebook vector decoded.

$$c'(n) = s_0\delta(n-i_0-1) + s_1\delta(n-i_1+1) + s_2\delta(n-i_2) + s_3\delta(n-i_3) \tag{15}$$

Known from Table 1, in this case i_0+1 doesn't equal to i_1. Hence, compared to the original vector $c(n)$, $c'(n)$ shifts one unit right at pulse τ_0 and one unit left at pulse τ_1. That is the two pulses shift one unit respectively.

(2)

$$\lfloor i_0/5 \rfloor = \lfloor i_1/5 \rfloor$$

It is same to case (1):

$$|i_0-i_0'| = |i_1-i_1'| = 1 \tag{16}$$

$$|s_0-s_0'| = |s_1-s_1'| = 0 \tag{17}$$

But in this case i_0+1 equals to i_1. Hence, $c'(n)$ equals to $c(n)$. That is the fixed codebook vector remains unchanged.

The cases of transposing the encoding locations of τ_1 and τ_2 are same to the cases above. □

Theorem 2 Transpose the encoding locations of τ_2 and τ_3 will make the total pulse displacements in the fixed codebook vector be three units at most.

Proof Except for adding the *jx* bit while encoding i_3, it is same to the situation in Theorem 1. So the conclusions are directly given below.

(1)

$$\lfloor i_2/5 \rfloor \neq \lfloor i_3/5 \rfloor$$

After transposing the encoding locations, let jx=0. There are two cases:

a)

$$i_3 \bmod 5 = 3$$

$$c'(n) = s_0\delta(n-i_0) + s_1\delta(n-i_1) + s_2\delta(n-i_2-1) + s_3\delta(n-i_3+1) \tag{18}$$

Compared to $c(n)$, $c'(n)$ shifts one unit right at pulse τ_2 and one unit left at pulse τ_3.

b)

$$i_3 \bmod 5 = 4$$

$$c'(n) = s_0\delta(n-i_0) + s_1\delta(n-i_1) + s_2\delta(n-i_2-1) + s_3\delta(n-i_3+2) \tag{19}$$

Compared to $c(n)$, $c'(n)$ shifts one unit right at pulse τ_2 and two units left at pulse τ_3.

(2)

$$\lfloor i_2/5 \rfloor = \lfloor i_3/5 \rfloor$$

After transposing the encoding locations, let jx stay the original value. Then the fixed codebook vector remains unchanged. □

Known from the two theorems above, the pulse displacements caused by transposing encoding locations of adjacent pulses are no more than three units. If the LSB of fixed codebook parameter is directly substituted, the pulse displacements will be five units. Therefore, transposing encoding locations of adjacent pulses in the fixed codebook vector introduces smaller displacements, which can be used to design speech steganography algorithm with better imperceptibility.

3 Steganography algorithm based on fixed codebook search process

With the guidance of incorporating data hiding into low bit-rate speech codec, in this section we propose a speech steganography algorithm based on the fixed codebook search process in G.729 codec.

In the fixed codebook vector, different pulses have different locations. The location relationship between adjacent pulses can be used to embed secret data. Let m_1 and m_2 be two secret bits. m_1 is embedded according to the location relationship between pulses τ_0 and τ_1, while m_2 is embedded according to the location relationship between pulses τ_2 and τ_3. The embedding rules are described below.

$$\begin{cases} i_0 > i_1 & if\ m_1 = 0 \\ i_0 < i_1 & if\ m_1 = 1 \end{cases} \tag{20}$$

$$\begin{cases} i_2 > i_3 & if\ m_2 = 0 \\ i_2 < i_3 & if\ m_2 = 1 \end{cases} \tag{21}$$

Pulse τ_3 is different from the first three pulses, which has 16 locations to choose and the location values are continuous between adjacent ones. Taking advantage of the parity of i_3, another one secret bit m_3 can be embedded.

$$\begin{cases} i_3 \bmod 2 = 0 & if\ m_3 = 0 \\ i_3 \bmod 2 = 1 & if\ m_3 = 1 \end{cases} \tag{22}$$

Based on the embedding rules above, data embedding can be performed while the fixed codebook vector is encoded during the fixed codebook search process. The key problem is how to change the location relationship to satisfy embedding requirement.

Let τ_j' and τ_{j+1}' (j=0,2) be the pulses decoded after transposing the encoding locations of τ_j and τ_{j+1}. Known from the two theorems described in Section 2.2, the new pulse locations i_j' and i_{j+1}' decoded with τ_j' and τ_{j+1}' are different from the original pulse locations i_j and i_{j+1}. The influence of transposing encoding locations of adjacent pulses on the relationship of pulse locations is analyzed in two cases below.

(1)

$$\lfloor i_j/5 \rfloor \neq \lfloor i_{j+1}/5 \rfloor$$

Table 1 shows that pulses τ_j and τ_{j+1} have different locations, that is $i_j \neq i_{j+1}$. Let's assume $i_j < i_{j+1}$. Because $\lfloor i_j/5 \rfloor \neq \lfloor i_{j+1}/5 \rfloor$, in this case $i_{j+1} - i_j \geq 6$. According to theorems 1 and 2, the pulse locations i_j' and i_{j+1}' decoded with τ_j' and τ_{j+1}' satisfy:

$$i_j' = \begin{cases} i_{j+1} - 2 & if\ j = 2\ and\ i_{j+1} \bmod 5 = 4 \\ i_{j+1} - 1 & else \end{cases} \tag{23}$$

$$i_{j+1}' = i_j + 1 \tag{24}$$

According to Eqs. (23) and (24), $i_j' - i_{j+1}' \geq i_{j+1} - i_j - 3 \geq 3$, that is $i_j' > i_{j+1}'$. The case of $i_j > i_{j+1}$ is same to the case of $i_j < i_{j+1}$, except for $i_j - i_{j+1} \geq 3$. So in this case $i_{j+1}' - i_j' \geq i_j - i_{j+1} + 2 \geq 5$, that is $i_j' < i_{j+1}'$. As a result, after transposing encoding locations of τ_j and τ_{j+1}, the relationship between i_j' and i_{j+1}' satisfies:

$$\begin{cases} i_j' > i_{j+1}' & if\ i_j < i_{j+1} \\ i_j' < i_{j+1}' & if\ i_j > i_{j+1} \end{cases} \tag{25}$$

Hence we can draw the conclusion that transposing encoding locations of adjacent pulses can change the location relationship between adjacent pulses.

(2)

$$\lfloor i_j/5 \rfloor = \lfloor i_{j+1}/5 \rfloor$$

According to theorems 1 and 2, in this case transposing encoding locations can't change the location relationship between adjacent pulses. Hence only the parity of i_3 can be used to embed secret data.

3.1 Embedding procedure

The embedding procedure is described below.

Step 1: Let $M=(m_1,m_2,\cdots,m_n)$ be the secret bit stream to be embedded.

Step 2: During the fixed codebook search process in each subframe, get the four pulses τ_0, τ_1, τ_2 and τ_3 in the fixed codebook vector. Let i_0, i_1, i_2 and i_3 be the pulse locations, s_0, s_1, s_2 and s_3 be the pulse signs.

Step 3: According to the values of i_0 and i_1, there are two cases.

(1) $\lfloor i_0/5 \rfloor \neq \lfloor i_1/5 \rfloor$

Let $m_a(a\in[0,n])$ be the secret bit to be embedded, i_0', i_1' and s_0', s_1' be the pulse locations and signs after m_l being embedded. m_a is embedded according to Eq. (26).

$$\begin{cases} i_0' = i_0, s_0' = s_0 \\ i_1' = i_1, s_1' = s_1 \end{cases} \quad if\ m_a = 0, i_0 > i_1\ or\ m_a = 1, i_0 < i_1$$
$$\begin{cases} i_0' = i_1 - 1, s_0' = s_1 \\ i_1' = i_0 + 1, s_1' = s_0 \end{cases} \quad if\ m_a = 0, i_0 < i_1\ or\ m_a = 1, i_0 > i_1 \tag{26}$$

(2) $\lfloor i_0/5 \rfloor = \lfloor i_1/5 \rfloor$

No embedding is performed.

Step 4: According to the values of i_2 and i_3, there are two cases.

(1)

$$\lfloor i_2/5 \rfloor \neq \lfloor i_3/5 \rfloor$$

Let m_b, m_c $(b,c\in[0,n])$ be the secret bits to be embedded, i_2', i_3' and s_2', s_3' be the pulse locations and signs after m_b and m_c being embedded. m_b and m_c are embedded according to Eq. (27).

$$\begin{cases} i_2' = i_2, s_2' = s_2 \\ i_3' = i_3 + [7-2\cdot(i_3 \bmod 5)]\cdot\left|i_3 \bmod 2 - m_c\right|, s_3' = s_3 \end{cases} \quad if\ m_b = 0, i_2 > i_3\ or\ m_b = 1, i_2 < i_3$$

$$\begin{cases} i_2' = i_3 - i_3 \bmod 5 + 2, \ s_2' = s_3 \\ i_3' = i_2 + 2 - \left|i_2 \bmod 2 - m_c\right|, s_3' = s_2 \end{cases} \quad if\ m_b = 0, i_2 < i_3\ or\ m_b = 1, i_2 > i_3 \tag{27}$$

(2)

$$\lfloor i_2/5 \rfloor = \lfloor i_3/5 \rfloor$$

No embedding is performed.

Step 5: Repeat Step 2 to Step 4 until all the secret data is embedded.

3.2 Extracting procedure

The extracting procedure is described below.

Step 1: During G.729 decoder operates on each subframe, get the four pulses τ_0', τ_1', τ_2' and τ_3' after the fixed codebook vector being decoded. Let i_0', i_1', i_2' and i_3' be the pulse locations.

Step 2: According to the values of i_0' and i_1', there are two cases.

(1)

$$\lfloor i_0'/5 \rfloor \neq \lfloor i_1'/5 \rfloor$$

m_a is extracted according to Eq. (28).

$$m_a = \begin{cases} 0 & if\ i_0' > i_1' \\ 1 & if\ i_0' < i_1' \end{cases} \tag{28}$$

(2)

$$\lfloor i_0/5 \rfloor = \lfloor i_1/5 \rfloor$$

No extracting is performed.

Step 3: According to the values of i_2' and i_3', there are two cases.

(1)

$$\lfloor i_2/5 \rfloor \neq \lfloor i_3/5 \rfloor$$

m_b is extracted according to Eq. (29).

$$m_b = \begin{cases} 0 & if\ i_2' > i_3' \\ 1 & if\ i_2' < i_3' \end{cases} \tag{29}$$

(2)

$$\lfloor i_2/5 \rfloor = \lfloor i_3/5 \rfloor$$

No extracting is performed.

Step 4: m_c is extracted according to Eq. (30).

$$m_c = \begin{cases} 0 & if\ i_3' \bmod 2 = 0 \\ 1 & if\ i_3' \bmod 2 = 1 \end{cases} \tag{30}$$

Step 5: Repeat Step 1 to Step 4 until all the secret data is extracted.

Let event A_1 denote that transposing operation can be used to perform embedding. Let event A_2 denote transposing operation is necessary for performing embedding. Let event A_3 denote that the parity of i_3 needs to be modified to perform embedding. The probabilities of the events are: $P(A_1)=7/8$, $P(A_2|A_1)=1/2$, $P(A_3)=1/2$. Let C denote the average number of secret bits can be embedded per subframe.

$$\begin{aligned} C &= P(A_1) \times 2 + 1 \\ &= 2.75 \end{aligned} \tag{31}$$

The hiding capacity of the proposed algorithm is $2.75\times2\times100=550$bits/s. According to the two theorems in Section 2.2, the average pulse displacements of the fixed codebook vector in each subframe can be calculated.

$$\begin{aligned} S = {} & P(A_1) \times P\left(A_2 \middle| A_1\right) \times 2 + P(A_1) \times \left[P\left(A_2 \middle| A_1\right) \times P(A_3) \times 3.5 \right. \\ & \left. + P\left(A_2 \middle| A_1\right) \times (1-P(A_3)) \times 2.5\right] + (1-P(A_1)) \times P(A_3) \\ = {} & 2.25 \end{aligned} \tag{32}$$

Define embedding rate as the average number of bits embedded per pulse displacement. The embedding rate of the proposed algorithm is $C/S=11/9$. The embedding rate of directly using LSB substitution method to the fixed codebook parameters is just 0.4. Therefore, under the same embedding rate condition, the proposed algorithm has a better performance in imperceptibility by decreasing pulse displacements.

4 Experiments

To evaluate the effectiveness of the proposed steganography algorithm, we selected speech files of different lengths from the AN4 database [1] as steganography covers to conduct experiments. The speech files were divided into three groups which were defined as Sample-1, Sample-2 and Sample-3 in terms of their lengths. Sample-1

contained speech files with lengths no more than 2 s; sample-2 contained speech files with lengths between 2 and 4 s; sample-3 contained speech files with lengths more than 4 s. Each group contained 300 speech files, with half male and half female speakers. Each speech file was sampled at 8 kHz and quantized to 16 bits using the linear pulse code modulation (PCM).

In the experiments, G.729 codec operated at 8 kbps. To maximize the hiding capacity, each subframe is used to embed secret data. The proposed algorithm is evaluated in terms of three aspects: imperceptibility, real-time performance and security.

4.1 Imperceptibility

The waveforms and spectrograms of normal speech files (synthetic speech files with nothing embedded) and stego speech files (synthetic speech files with secret data embedded) are shown in Figs. 2 and 3. In both two figures, graphs (a)–(c) denote normal speech files and graphs (d)–(f) denote the corresponding stego speech files. Known from the two figures, there is nearly no difference between the normal speech files and the stego speech files in time domain and frequency domain. Therefore, the proposed algorithm has little influence on speech quality.

When evaluating speech quality, objective evaluation methods are widely used for their advantages of convenience, flexibility and reliability. Objective evaluation methods can be divided into two groups: evaluation methods based on output only and evaluation methods based on input-output. A typical representative of the former is the evaluation method based on E-Model [4], which is a non-intrusive method. Because just using the output signal to evaluate speech quality, it has a good real-time performance. A typical representative of the latter is the perceptual evaluation speech quality (PESQ) method described in ITU-T P.862 Recommendation [3], which offers a high evaluation precision

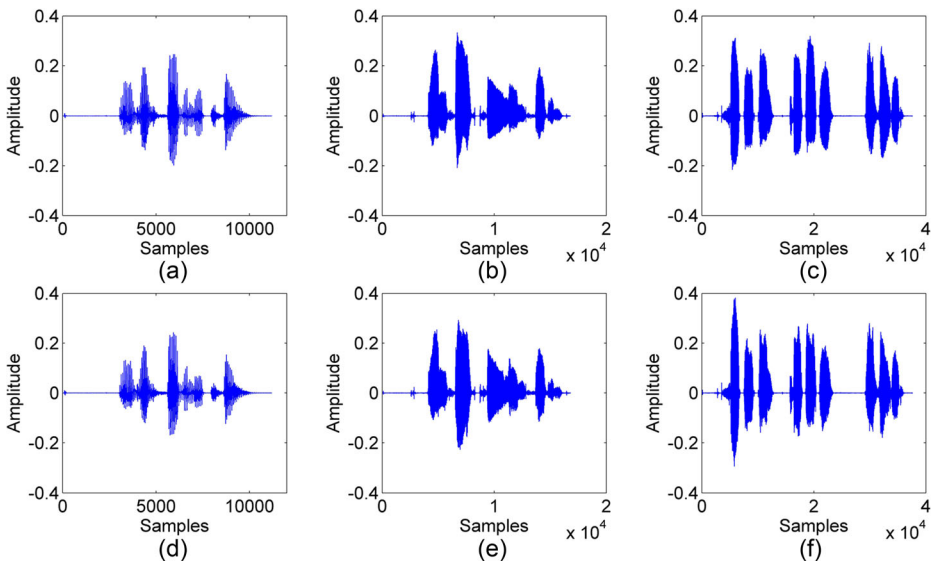


Fig. 2 Waveforms of normal speech files and stego speech files

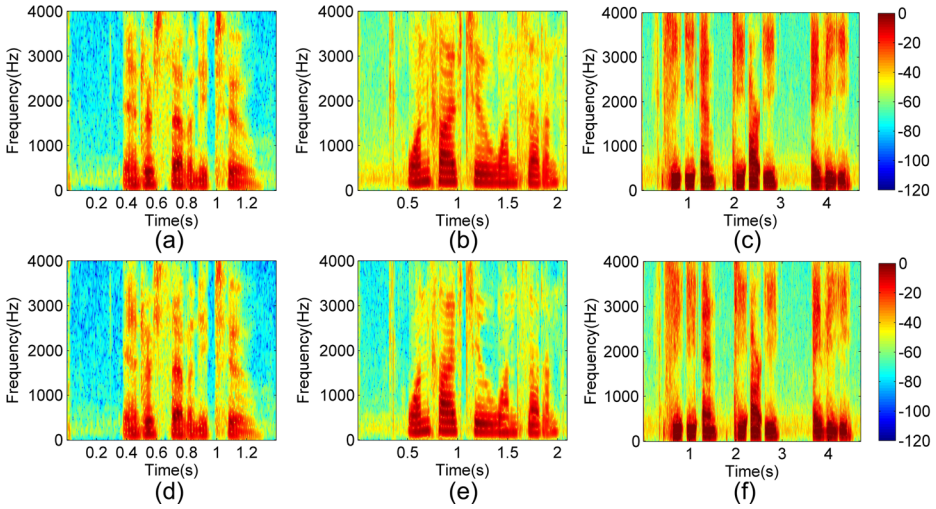
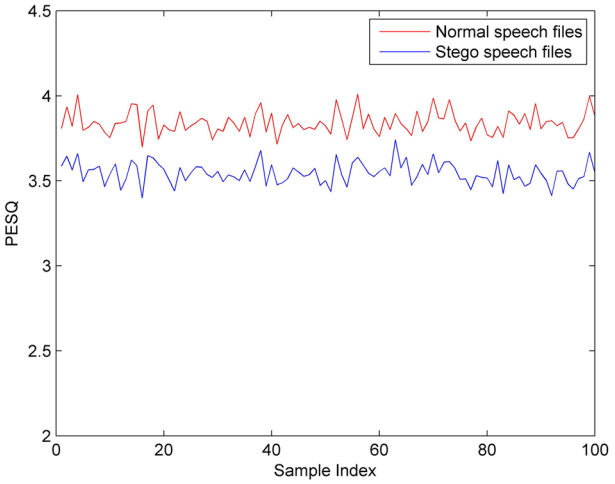


Fig. 3 Spectrograms of normal speech files and stego speech files

by comparing an original signal with a degraded signal and outputs a PESQ value as a prediction of the perceived quality. What concerned most in this paper is the quality of the stego speech file compared to the normal speech file, so we select the PESQ method to precisely reflect the difference in perceived qualities. The range of the PESQ value is −0.5 (worst) to 4.5 (best).

Figures 4, 5 and 6 show the PESQ values of 100 normal and stego speech files in three groups respectively. Table 2 shows the statistical results of the PESQ values for the three speech groups, where the negative change rate denotes deterioration in PESQ value and the positive one denotes amelioration. From the table, the average worsening change rates of PESQ values are 6.55, 6.47 and 6.60 %, respectively. The total average

Fig. 4 PESQ values for Sample-1



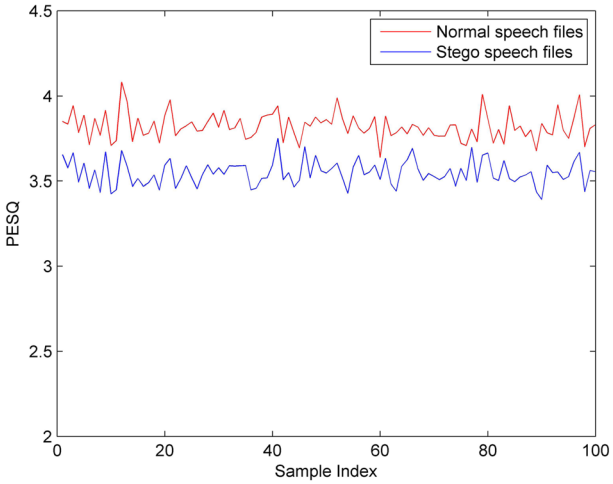


Fig. 5 PESQ values for Sample-2

worsening change rate is 6.54 %. Though the PESQ values of stego speech files are slightly lower than that of normal speech files, it still remains at a normal level. Hence the stego speech files obtained by using the proposed algorithm have good perceived qualities.

The steganography algorithms proposed in [10] and [16] also perform data embedding in G.729 codec, they are representative research products published in recent years. In [10], fixed codebook vector is also taken as the cover data to embed secret data. In [16], parameters in fixed codebook vector and LSP (Line Spectrum Pair) are used to embed secret data together. Figure 7 shows the PESQ values for randomly selected 100 speech files of different lengths after being embedded using the proposed algorithm, the algorithm in [10] and the algorithm in [16]. The embedding rates of the proposed algorithm, the algorithm in [10] and the algorithm in [16] are 550 bits/s, 398 bits/s and 2.4kbits/s,

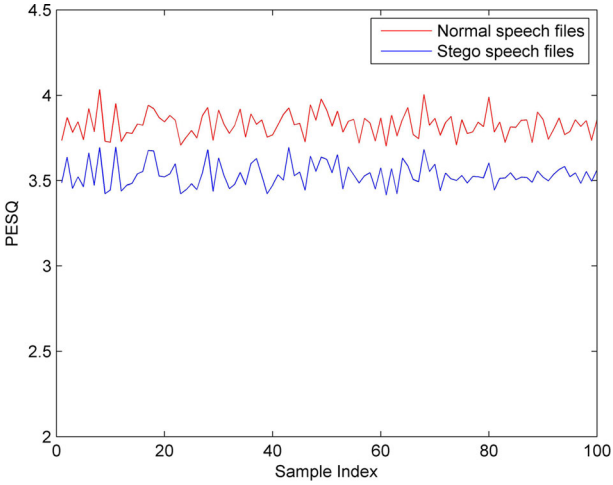


Fig. 6 PESQ values for Sample-3

Table 2 PESQ statistical results

Speech file	Proposed method			Nothing embedded			Chang in PESQ(%)		
	Average	Max	Min	Average	Max	Min	Average	Max	Min
Sample-1	3.55	3.79	3.38	3.81	3.95	3.49	−6.55	−2.46	−11.38
Sample-2	3.54	3.75	3.40	3.83	4.01	3.53	−6.47	−2.38	−11.61
Sample-3	3.54	3.81	3.37	3.79	4.03	3.51	−6.60	−2.54	−11.51

respectively. Known from Fig. 7, the PESQ values of the proposed algorithm are around 3.5, indicating the perceived qualities of stego speeches are good. The PESQ values of the algorithm in [10] are around 3.2, so the proposed algorithm performs better than the algorithm in [10] in imperceptibility while achieving a higher hiding capacity. Though the embedding rate of the algorithm in [16] is much bigger than the proposed algorithm, the PESQ values of the algorithm in [16] is around 2.9, indicating a bad perceived qualities of stego speeches.

4.2 Real-time performance

To satisfy the real-time requirement, the latency caused by secret data embedding should be controlled in a small range. In order to test the influence of the proposed algorithm on the real-time performance of G.729 codec, we recorded the processing time of G.729 codec operating on each speech frame with and without secret data embedded on Intel Core 3.20 GHz computers with 4G DDR3 SD RAM.

The encoding time and decoding time of 100 speech frames are shown in Figs. 8 and 9. As is shown, the two curves in either figure are very close to each other. The average encoding and decoding time delays caused by the proposed algorithm are 0.185 and 0.093 ms respectively, which is negligible compared to the 15 ms time delay allowed in

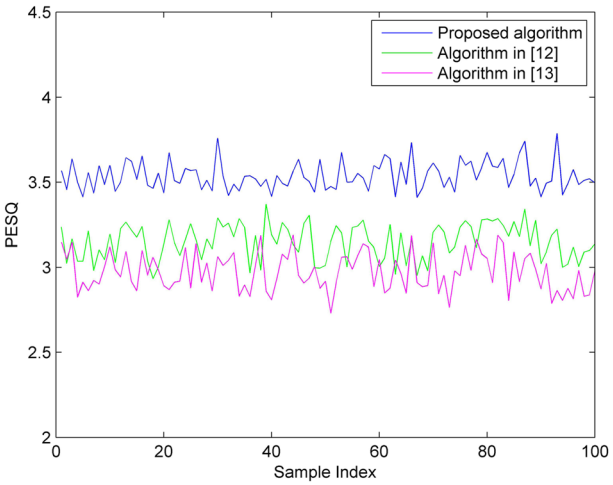


Fig. 7 Comparisons of PESQ values of the proposed algorithm, the algorithm in [10] and the algorithm in [16]

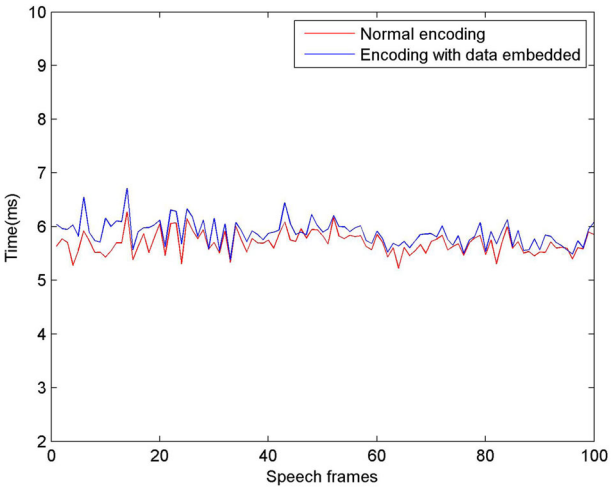


Fig. 8 Encoding time of speech frames with and without data embedded

G.729 codec. Therefore, the proposed algorithm can satisfy the real-time requirement of G.729 codec well.

4.3 Security

The anti-detection performance is of great importance when considering the steganography security. We used the derivative mel-Frequency cepstral coefficients (DMFCC)-based steganalysis algorithm [5], an advanced algorithm based on Fourier spectrum statistics and mel-cepstrum coefficients, to detect the proposed algorithm. The LIBSVM Version 3.0 was used in our test. In the SVM-scale of LIBSVM, the lower and the upper is −1 and 1 respectively. The other parameters used are default values. In the SVM-train of LIBSVM, the svm_type is C-SVC, the kernel_type is RBF (radial basis

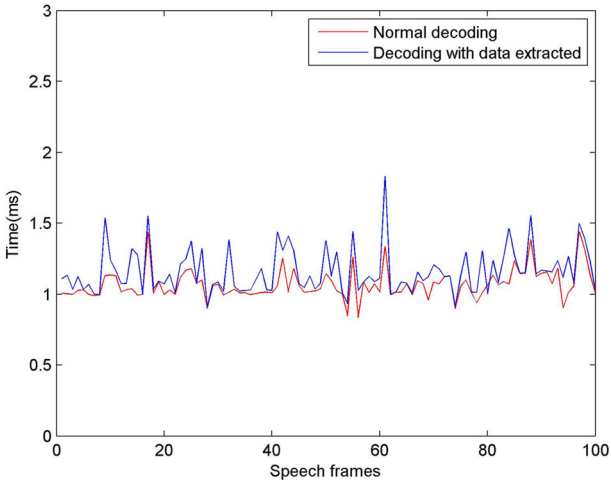


Fig. 9 Decoding time of speech frames with and without data extracted

Table 3 Steganalysis results using the DMFCC-based algorithm

Window length(frames)	Sample-1(%)	Sample-2(%)	Sample-3(%)
1	48.47	48.24	48.95
10	48.81	48.63	49.07
30	48.90	49.09	50.49
50	50.58	50.75	50.69
100	51.12	51.27	52.15
200	52.70	52.85	52.76
300	52.91	52.54	53.06
500	55.34	54.96	54.97
Average	51.10	51.04	51.52
Max	55.34	54.96	54.97
Min	48.47	48.24	48.95

function), the cost is 1000, the epsilon is 0.00001, and the other parameters used are default values.

The detection results are presented in Table 3. The average detection accuracies for the three groups of speech files are 51.10, 51.04 and 51.52 % respectively. This indicates the proposed algorithm can avoid being detected by the DMFCC-based algorithm.

Figure 10 shows the steganalysis results of the proposed algorithm and the algorithm in [10] using DMFCC-based algorithm at different detection window lengths. Know from Fig. 10, as the detection window becomes long, the detection accuracies of the two algorithms increase. This is because longer detection window means more speech frames are analyzed. But the average detection accuracies are around 51 % and the maximum accuracies are under 55 %, indicating the steganalysis algorithm is useless for the above two algorithms. The main reason is the two algorithms are performed in compressed domain, which have little impact on the time or transform domain.

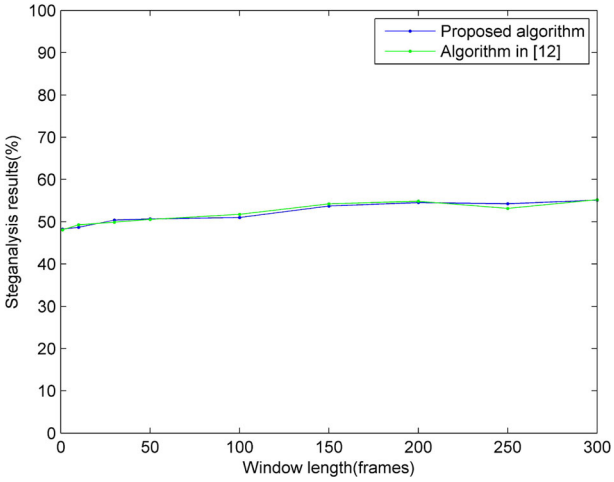


Fig. 10 Comparison of steganalysis results between the proposed algorithm and the algorithm in [10]

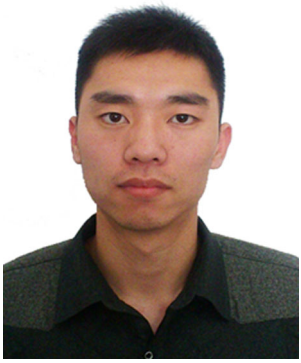
5 Conclusion

In this paper, a novel steganography algorithm for low bit-rate speech is proposed. Based on analyzing the fixed codebook search process in G.729 codec, we prove that transposing encoding locations of adjacent pulses in the fixed codebook vector causes small displacements. Then, the location relationship between adjacent pulses and the parity of the location value in the fourth pulse are used to perform embedding. The proposed steganography algorithm can attain a hiding capacity of 550 bits/s. Experimental results show that the proposed steganography algorithm outperforms the traditional steganography algorithms based on the fixed codebook in terms of imperceptibility. Meanwhile, the proposed steganography algorithm satisfies the real-time requirement well and can avoid being detected by the DMFCC-based steganalysis algorithm. The emphasis of future work will be put on how to apply the proposed algorithm to other codecs considering their characteristics.

References

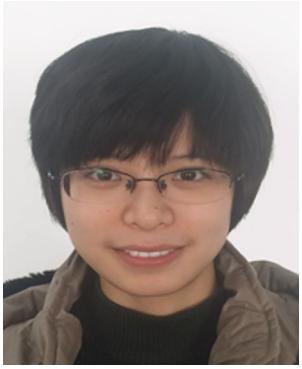
1. An4 database (1991) http://www.speech.cs.cmu.edu/databases/an4/
2. Huang Y, Liu C, Tang S, Bai S (2012) Steganography integration into a low-bit rate speech codec. Inf Forensics Secur 7:1865–1875,
   IEEE Transactions on
3. ITU-T, Recommendation P (2001) 862-perceptual evaluation of speech quality (PESQ): an objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs. International telecommunication union-telecommunication standardization sector (ITU-T)
4. ITU-T, Recommendation G107 (2002) The E-Model, a computational model for use in transmission planning
5. Liu Q, Sung AH, Qiao M (2009) Temporal derivative-based spectrum and mel-cepstrum audio steganalysis. Inf Forensics Secur 4:359–368
6. Liu J, Zhou K, Tian H (2012) Least-significant-digit steganography in low bitrate speech. In: Proceedings of IEEE international conference on communications:1133–1137
7. Mazurczyk W (2013) VoIP steganography and its detection-a survey. ACM Comput Surv (CSUR) 46(2):20
8. Su Y, Huang Y, Li X (2006) Steganography-oriented Noisy Resistance Model of G.729a. In: Proceedings of IMACS multiconference:11–15
9. Tang S, Chen Q, Zhang W et al (2015) Universal steganography model for low bit-rate speech codec. Secur Commun Netw. doi:10.1002/sec.1183
10. Tian H, Jiang H, Zhou K, Feng D (2011) Adaptive partial-matching steganography for voice over IP using triple M sequences. Comput Commun 34:2236–2247
11. Tian H, Liu J, Li S (2014) Improving security of quantization-index-modulation steganography in low bit-rate speech streams. Multimedia Systems 20:143–154
12. Tian H, Zhou K, Jiang H, Huang Y, Liu J, Feng D (2009) An M-sequence based steganography model for voice over IP. In: Proceedings of the 44th IEEE international conference on communications, Dresden, Germany:1–5
13. Tian H, Zhou K, Feng D (2010) Dynamic matrix encoding strategy for voice-over-IP steganography. J Cent S Univ Technol 17:1285–1292
14. Tian H, Zhou K, Lu J (2012) A VoIP-based covert communication scheme using compounded pseudorandom sequence. Int J Adv Comput Technol 4(1):223–230
15. Wei Z, Zhao B, Liu B et al (2014) A novel steganography approach for voice over IP. J Ambient Intell Humaniz Comput 5(4):601–610

16. Wu Z, Cao H, Li D (2015) An M-sequence based steganography model for voice over IP. Chin J Electron 24(1):157–165
17. Yan S, Tang G, Sun Y, Gao X, Shen L (2014) A triple-layer steganography scheme for low bit-rate speech streams. Multimed Tools Appl 1–20. doi:10.1007/s11042-014-2265-y



Shufan Yan received the B.S. degree in electronic science and technology from Zhengzhou information science and technology institute, Henan, China, in 2012. He is pursuing the M.S. degree in information security at Zhengzhou information science and technology institute. His research interests include information hiding and information security.

Guangming Tang received the B.S., M.S. and Ph.D. degrees in information security from Zhengzhou information science and technology institute, Henan, China, in 1983, 1990, and 2008, respectively. She is now a professor at the Department of Information Security, Zhengzhou information science and technology institute. Her research interests include information hiding, watermarking and software reliability. She has published 60 research articles and 3 books in these areas.



Yanling Chen received the B.S. degree in electronic science and technology from Zhengzhou information science and technology institute, Henan, China, in 2012. She is pursuing the M.S. degree in information security at Zhengzhou information science and technology institute. Her research interests include information hiding and information security.