Covert Voice over Internet Protocol communications based on spatial model

Covert Voice over Internet Protocol Communications Based on Spatial Model

HUANG Yongfeng[1,2], TANG Shanyu[3,4*]

[1] *Department of Electronic Engineering, Tsinghua University, Beijing 100084, China*
[2] *Tsinghua National Laboratory for Information Science and Technology, Tsinghua University, Beijing 100084, China*
[3] *School of Computer Science, China University of Geosciences, Wuhan 430074, China*
[4] *Department of Computing, London Metropolitan University, London N7 8DB, United Kingdom*

This paper presents a new spatial steganography model for covert communications over Voice over Internet Protocol (VoIP), providing a solution to the issue of increasing the capacity of covert VoIP channels without compromising the imperceptibility of the channels. Drawing from Orthogonal Modulation Theory in communications, the model introduced two concepts, orthogonal data hiding features and data hiding vectors, to covert VoIP communications. By taking into account the variation characteristics of VoIP audio streams in the time domain, a hiding vector negotiation mechanism was suggested to achieve dynamic self-adaptive steganography in media streams. Experimental results on VoIP steganography show that the proposed steganographic method effectively depicted the spatial and temporal characteristics of VoIP audio streams, and enhanced robustness against detection of steganalysis tools, thereby improving the security of covert VoIP communications.

covert communication, multimedia networking, VoIP, data security

*Corresponding author (email: shanyu.tang@gmail.com)

1 Introduction

Covert communication is the science of hiding the existence of secret information in innocent cover objects. In this paper, covert communications mean that secret messages are transmitted between the communicating parties by using Voice over Internet Protocol (VoIP) steganography rather than cryptography.

Streaming media have emerged as a new means of communications over the Internet, and the pace of fresh developments is accelerating. It is anticipated that streaming media will dominate the long-distance communications in the next few years. The applications of streaming media technologies include Video on Demand (VOD), Audio on Demand (AOD), Internet Protocol Television (IPTV), and VoIP, etc.

With the upsurge of streaming media applications available for commercial use in recent years, streaming media become one of the most interesting cover objects for information hiding. Unlike static cover objects, such as text, images, and audio files, streaming media as cover objects have the following characteristics:

a) Spatial characteristics of media streams. Media streams are dynamic chunks of a series of packets that consist of IP headers, UDP headers, RTP headers, and payloads *i.e.* audio and/or video frames. These headers and frames have a number of unused fields, providing plausible covert channels and thus giving scope for steganography. Each header or frame could be treated as a dimension, and so it is worth exploring data hiding in headers and frames in view of spatial dimensions.

b) Temporal characteristics of media streams. The temporal behavior of media streams manifests in two aspects. First, the payloads of media streams show a significant temporal characteristic. For example, audio frames can be sorted into two groups, active voice and inactive voice; *I*, *B* and *P* frames appear alternately in video streams. Second, the packet loss rate of media streams is related to time in the course of media streams transmission. Thus, steganographic algorithms for embedding data in media streams

should be ‘self-adaptive’ in the temporal dimension.

c) Real-time media streams. Streaming media communications originate from the sender, and the receiver normally plays the arrived media streams without storing them. Steganography in media streams is a dynamic process, which should not interrupt the real time distribution of media streams. It requires both data embedding and data extraction to be conducted in a real time manner. Such real time data embedding and extraction differentiates steganography in media streams from steganography in static cover objects such as text, image and audio files.

Covert communications over streaming media are resulted from the integration of information hiding and network media stream technologies, and have become the main focus of attention in the field of information security. There has been some effort to study covert communications over streaming media; however, most existing studies focus mainly on the steganographic algorithms based on a single characteristic of streaming media. Such approaches have bottlenecked efforts to improve data hiding capacity. In this study, we seek ways to improve data hiding capacity by taking into account multiple characteristics in the spatial and temporal dimensions of streaming media.

The motivation of this paper is to achieve dynamic self-adaptive covert communications over VoIP by introducing two concepts, orthogonal hiding features and hiding vectors, to covert communications over VoIP, with their orthogonal hiding features being abstracted as a multi-dimensional hiding space model according to the spatial structure of IP packets of media streams, and use of hiding vectors in the model to greatly improves the secret data conveying rate of covert VoIP communications.

2 Related work

Contrary to image and audio steganography, covert communications over streaming media are largely unexplored so far. There have been some attempts to develop network protocols- and/or VoIP-based steganographic algorithms. For example, in [1] [2] [3] researchers studied ways to hide data by means of

TCP/IP protocols, suggesting headers substitution-based algorithms for hiding data in unused header fields.

Bai *et al.* [4] suggested a covert channel over the jitter fields of RTCP headers, where a piece of secret information is modulated into the jitter fields according to the statistical parameters computed. Depending on whether the secret information bit number is '0' or '1', the time intervals between packets are modulated. The advantage of this method lies in high steganographic transparency, but data hiding capacity is relatively low and performance is less robust.

In light of audio/video frames, Huang *et al.* [5] investigated steganography in VoIP streams encoded by G.711 codec, proposing a LSB matching-based steganographic algorithm with the help of the redundant audio data mechanism to avoid loss of hidden information. Their method sustains better concealment level than traditional LSB algorithms.

Aoki [6] worked out a technique of lossless steganography in G.711μ encoded speech. The method adopts audio streams encoded by pulse-code modulation (PCM) as cover objects, in which plenty of least significant bits exist. When using μ-law G.711 in networks, the codec takes a 14-bit signed linear audio sample as an input, increases the magnitude by 32 (binary 100000), and converts it to an 8 bit value; the sign bit '0' is expressed as '-0' and '+0'. With this method, the audio sample '0' is altered to '-0' or '+0' when embedding secret data '0' or '1', respectively. So data hiding capacity is dependent on the number of '0's in the audio sample, and the applications are limited.

Hiding data in low bit rate VoIP streams is more challenging, as there is less redundancy in the audio streams. By analyzing noisy resistance, Su *et al.* [7] suggested a codebook-based steganographic algorithm for hiding secret data in VoIP streams coded by G.729A codec. Liu *et al.* [8] identified the G.729 speech parameters that are suitable for data hiding through analyzing the parameter characteristics of G.729 encoded audio frames.

Xiao *et al.* [9] pioneered a method for hiding data in low bit rate VoIP streams, which is the first ever

effort to improve the codebook partition by using Graph theory along with Quantization Index Modulation. With this approach, random codebook partitions are prevented and speech quality is then guaranteed.

Mazurczyk *et al.* [10][11], Druid [12] and Kratzer *et al.* [13] developed various methods of hiding data in VoIP streams, suggesting synchronization mechanisms between the sending party and the receiving party.

Analysis of the studies above shows the previously proposed steganographic algorithms are based solely on a single hiding feature. Although these algorithms can achieve different levels of data hiding, the corresponding data hiding capacities are not large enough to have practical applications in covert communications systems. To improve usability of the systems, it is necessary to seek a new way to further increase the data hiding capacity. Hence, in this study we suggest a new method capable of hiding large amounts of data in multiple dimensions of media streams so as to greatly improve the data hiding capacity.

To take up the challenges, this study is to build a multi-dimensional data hiding spatial theoretical model for steganography in media streams through analyzing the fundamental characteristics of media streams. The new model is then applied to covert communications over VoIP, which is widely used on the Internet. The experimental results show that the model effectively depicts the spatial and temporal characteristics of media streams, providing a good solution to data hiding capacity and security issues in covert VoIP communications.

3 Proposed covert VoIP communication model

3.1 Structure of media streams

Media streams-based communications work on the same network principles (and often on the same network) as conventional data traffic. The session initiation protocol (SIP) is used to initiate and control

sections between the communicating parties; when the content of the request begins to flow, it is carried over the real-time transport protocol (RTP). As shown in Fig. 1, audio and video media streams are dynamic chunks of a series of packets that consist of IP headers, UDP headers, RTP headers, and numbers of audio and/or video frames, which can be regarded as a four-layer/dimension structure.

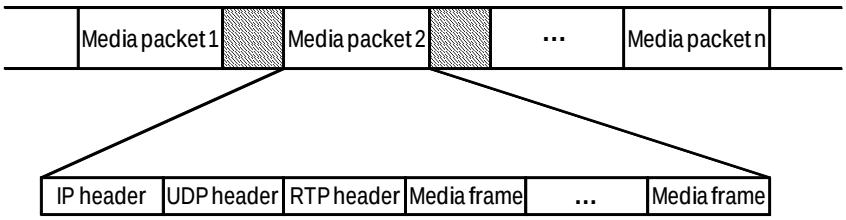


Figure 1 Structure of media streams.

Media streams are essentially a sequence of IP packets, which are the basic units consisting of protocol headers and payloads (frames) with unused fields. As cover objects, media streams can then be characterized from the perspective of temporal and spatial dimensions. The proposed spatial model detailed below describes the macro hiding features of media streams and their dynamic hiding behavior in the temporal domain, with particular focus on designing a self-adaptive hiding algorithm and a mechanism of hiding synchronization. The micro hiding features of individual IP packets are studied in view of multi-dimensional hiding space, so as to seek ways to largely improve data hiding capacity.

3.2 Spatial model for steganography in streaming media

Previous studies have established several algorithms for hiding data in different IP headers [3]. For instance, there are IP protocol- or RTP protocol-based steganographic algorithms, as well as algorithms for hiding data in media frames, which have laid the foundation for this work. On the basis of the research results, we develop a new spatial model for steganography in media streams. The definitions below introduce the proposed model.

According to the spatial structure of IP packets of media streams, their orthogonal hiding features are

abstracted as a multi-dimensional hiding space model; use of hiding vectors in the model greatly improves the secret data conveying rate of covert communications over media streams. By taking into account the variation characteristics of media streams in the time domain, a hiding vector negotiation mechanism is proposed in this study to achieve dynamic self-adaptive covert VoIP communications.

Suppose a media stream is denoted by $P(k)$, $P(k)=\sum_{k=1}^{\infty} p(kT)$, where $p(kT)$ represents the IP packets that comprise the media stream $P(k)$, and T is the time length of the media packet.

Definition 1: Hiding feature. It is the feature of a media packet which is suited to data hiding, denoted by *e*. In general, a cover object has different features suitable for data hiding. From a hiding feature, ones can design a variety of steganographic algorithms. For example, RTP timestamp is a hiding feature of a media packet, on which different steganographic algorithms can be established [4][10].

Definition 2: Hiding feature set. It is a set of all the hiding features of a *media packet*, through which data hiding is achievable, denoted by R_m, $R_m=\{e_1,...,e_m\}$, where e_i is the *i*th hiding feature. For instance, in a *media packet*, codebook field of G.723.1 codec and RTP timestamp field have different hiding features, which comprise the hiding feature set of the media packet.

Definition 3: Orthogonality of two hiding features. If two hiding features (e_i and e_j) of the media packet $p(kT)$ are used for data hiding through two corresponding steganographic algorithms, $f_i(e_i)$ and $f_j(e_j)$, the two algorithms operate independently, and their results do not affect each other, the features (e_i and e_j) are then considered to be orthogonal (statistically unrelated), denoted by $e_i \perp e_j$. The corresponding algorithms, $f_i(e_i)$ and $f_j(e_j)$, are orthogonal too, *i.e.* $\langle f_i(e_i), f_j(e_j)\rangle = 0$.

Definition 4: Orthogonal hiding feature set. It is a group of pairs of orthogonal hiding features of a media packet, denoted by $R_n^{\perp}=\{e_{k\in N} \mid \forall i,j\in N, e_i \perp e_j\}, N=\{1,...,n\}$, and $R_n^{\perp}\subseteq R_m, \forall n\le m$.

Definition 5: *n*-dimensional hiding space of a packet based on *n*-element orthogonal hiding feature set.

If the orthogonal hiding feature set of a media packet has *n* elements, the packet contains *n* dimensions for data hiding, denoted by $\phi_n(p(kT)) = \{e_i \mid i = 1,...n\}$. Among them, the *i*th dimensional hiding feature e_i constitutes the *i*th dimensional axis in *n*-dimensional data hiding space.

Definition 6: Metafunction of hiding feature. In *n*-dimensional data hiding space of a media packet, if m_i steganographic functions are obtained through the *i*th dimensional hiding feature, those functions are regarded as metafunctions, which can be described as $f_{i,k_i}(e_i), \forall i = 1,...,n; k_i = 1,...,m_i$. They satisfy $f_{i,k_i}(e_i) = f_{i,k_i}(p(kT)), \forall i = 1,...,n; k_i = 1,...,m_i$. In particular, no hiding is denoted as the null metafunction $f_{i,0}(e_i), \forall i = 1,...,n$.

Theorem 1: Judgment rule for steganographic function orthogonality.

If the media packet *p*(*kT*) has *n*-dimensional hiding space, the *i*th dimensional steganographic metafunction $f_{i,k_i}(e_i), \forall i = 1,...,n; k_i = 1,...,m_i$ in the space is reciprocally orthogonal to the *j*th dimensional steganographic metafunction $f_{j,k_j}(e_j), \forall j = 1,...,n; k_j = 1,...,m_j$ in terms of the data hiding operation on the packet, *i.e.* $\langle f_{i,k_i}(p(kT)), f_{j,k_j}(p(kT)) \rangle = 0, \forall i, j \in \{1,...,n\}, k_i = 1,...,m_i, k_j = 1,...,m_j$.

The reciprocally orthogonal character of steganographic metafunctions mentioned above means that there are no logical relevance to each other between the steganographic metafunctions in the *i*th and *j*th dimensions, *i.e.*, these steganographic metafunctions corresponding to different data hiding features (algorithms) are not, of course, related reciprocally, leading to any decrease in the total hiding capacity when they are used together in covert communications.

Definition 7: Hiding vector function. The hiding vector function $F_{K_3}(p(kT))$ is an ordered collection of different dimensional hiding metafunctions, which are used for data hiding simultaneously, in *n*-dimensional space of the media packet *p*(*kT*). It is given by

$$F_{\mathbf{k}}(p(kT)) = (f_{1,k_1} \circ \cdots \circ f_{n,k_n})(p(kT)), \forall \mathbf{k} \in \{1,\ldots,m_1\} \times \cdots \times \{1,\ldots,m_n\}$$

Definition 8: Data hiding capacity of hiding vector. It is defined as the sum of numbers of bits of se-

cret information hidden in a media packet by using the hiding vector $F_{K_3}(p(kT)), \forall k_3 = 1,...,m_3$, in bits per second (bps). The data hiding capacity of the hiding vector can be expressed as $\left|F\left(p\left(kT\right)\right)\right|$.

Theorem 2: Data hiding capacity theorem.

If a media packet has an *n*-dimensional hiding vector function, the data hiding capacity of the vector function is equal to the sum of the data hiding capacities of *n* metafunctions that constitute the hiding vector, *i.e.* $\left|F_{\mathbf{k}}\left(p\left(kT\right)\right)\right| = \sum_{i=1}^{n}\left|f_{i,k_i}\left(e_i\right)\right|, \forall \mathbf{k} \in \{1,\ldots,m_1\} \times \cdots \times \{1,\ldots,m_n\} = M\left(R_n^{\perp}\right)$.

Definition 9: Imperceptibility of hiding vector. The imperceptibility of a hiding vector is a measure of the degree of difficulty in distinguishing the difference between the untapped and stego media streams, and it is dependent upon the worst imperceptibility among all the metafunctions that constitute the hiding vector, *i.e.* the Buckets effect of the hiding vector. The imperceptibility of the hiding vector $F_{\mathbf{k}}\left(p\left(kT\right)\right), \mathbf{k} \in M\left(R_n^{\perp}\right)$ can be expressed as $\overline{\overline{F}}_{\mathbf{k}}\left(p\left(kT\right)\right) = \min\left\{\overline{\overline{f}}_{i,k_i} \mid i = 1,...,n\right\}, \mathbf{k} \in M\left(R_n^{\perp}\right)$.

Theorem 3: Imperceptibility theorem.

In *n*-dimensional hiding space of a media packet, the phase of the hiding vector $\overline{\overline{F}}_{\mathbf{k}}\left(p\left(kT\right)\right)$ represents its imperceptibility. If there are two different hiding vectors with the same data hiding capacity in the *n*-dimensional hiding space, the one whose phase direction is close to the space diagonal (*i.e.* the difference in imperceptibility between each pair of dimensions is minimal) will have better imperceptibility.

Definition 10: Hiding capacity set *F*. In media streams-based covert communications systems, different hiding metafunctions in *n*-dimensional hiding space of a media packet can constitute various hiding vectors $F_{K_3}, K_3 = 1,...,\left|M\right|$. The hiding vector set provides the hiding capacity set for covert communication terminals, *i.e.* $F = \left\{F_{K_3} \mid K_3 = 1,...,\left|M\right|\right\}$. In the hiding capacity set, each hiding vector is marked by a unique k_3, which is called hiding vector number.

3.3 Spatial model-based covert communications over media streams

Cachin [14] suggested an information-theoretic model applicable to image and audio steganography, but the model is not suitable for covert communications over media streams because of heavy packet loss during the transmission of media streams. To bridge the gap, we propose the following model based on the spatial model detailed in the previous section.

In the Internet environment, UDP protocol is generally used to transfer streaming media such as VoIP, etc., thus, packet loss is inevitable. And at different periods, because of the different numbers of users on the Internet, there is a great difference in the size of network traffic. Therefore, the packet loss rate varies with time in VoIP communications over the Internet. In our experimental environment, the average packet loss rate was 17% for one week between 1 September 2015 and 6 September 2015; the average annual packet loss rate was about 13% between 1 September 2014 and 1 September 2015, indicating high variable packet loss rates for VoIP communications.

If VoIP packets are used as the carriers to realize covert communication, it needs to solve the problem of the recovery and reliability of the hidden information in the streaming packets in the case of packet loss. In the proposed algorithm, a spatial model for data hiding is used to achieve a high hiding capacity for covert communications over VoIP. The proposed algorithm solves the packet loss problem by using the fast starting retransmission algorithm, *i.e.* re-transmitting the packet once three irregular orders of ACK occur without waiting until the expiry of the timer RTO, so as to achieve a reasonable conveying rate of the hidden message (See Section 4.1 and Fig. 4).

Assuming the communicating parties have established the same hiding capacity set in their terminals $F = \left\{ F_{k_3} \mid k_3 = 1,..., \prod_{i=1}^{n} m_i \right\}$, the key methodology ‘temporal and spatial hiding synchronization’, is introduced to covert VoIP communications.

In the media streams $P(k) = \sum_{k=1}^{N} p(kT)$, part of media packets is used to hide the secret data

$M = \bigcup_{i \in N} \{0,1\}^i$.

$$P^*(k) = F(P(k), M) = \sum_{k=1}^{n_1-1} p(kT) + \sum_{k=n_1}^{n_2} F_{k_3}(p(kT), B_i) + \sum_{k=n_2+1}^{N} p(kT)$$

$$P^*(k) = \sum_{k=1}^{n_1-1} p(kT) + \sum_{k=n_1}^{n_2} p^*(kT) + \sum_{k=n_2+1}^{N} p(kT) \qquad (1)$$

$$\forall k_3 = 1,..., \prod_{i=1}^{n} m_i, B_i \in M$$

where B_i is part of the bits of the secret data, $P^*(k)$ is the stego media streams consisting of media packets without any hidden data, $p(kT)$, and those with hidden data, $p^*(kT)$. The stego media packets $p^*(kT)$ are those packets, selected from the media packets $p(kT)$, in which the secret data are embedded by using some hiding vector functions $F_{k_3}(\ )$ in the hiding capacity set F.

The media streams with hidden data $P^*(k)$, originating from the sender, are transmitted over the Internet and distributed to the receiver. As equation (1) shows, the receiver requires solving the following two main problems before the secret data M can be extracted from the received stego media streams $P^*(k)$.

(a) Identifying media packets with hidden data from those without hidden data in $P^*(k)$, and

(b) Judging which hiding vectors are used by the stego packets $p^*(kT)$ to embed the secret data.

The above problems are regarded as the 'temporal synchronization' and 'spatial synchronization' issues of covert communications.

Supposing a hiding vector $F_0()$ is added to the hiding capacity set of the communicating parties $F = \left\{ F_{k_3} \mid k_3 = 1,..., \prod_{i=1}^{n} m_i \right\}$ to form $F = \left\{ F_{k_3} \mid k_3 = 0,..., \prod_{i=1}^{n} m_i \right\}$, where $F_0()$ is a void function representing no data hiding operation, equation (1) becomes

$$P^*(k) = \sum_{k=1}^{N} F_{k_3}(p(kT), B_i) = \sum_{k=1}^{N} p^*(kT), \forall k_3 = 1,..., \prod_{i=1}^{n} m_i, B_i \in M \vee B_i = \varnothing \qquad (2)$$

Equation (2) shows the temporal and spatial hiding synchronization in covert communications over

media streams requires the communicating parties to set up an updating mechanism for consulting and sharing the information on the hiding vectors having been used by individual media packets for data hiding.

In other words, the temporal and spatial hiding synchronization means a negotiation mechanism allows the communicating parties to share the information about the currently used hiding vector number thorough a covert channel. For a media packet using the hiding vector $F_0()$, there is no hidden information; otherwise, the media packet contains secret data. If the receiving party has known the hiding vector $F_{k_3}(\ )$ used by the current media packet, the receiver can then extract the secret data from the media packets. The data extracting process is described as

$$\hat{B} = F_{k_3}^{-1}\left(P^*\left(k\right)\right) = \sum_{k=1}^{N} B_i = \sum_{k=1}^{N} F_{k_3}^{-1}\left(p^*\left(kT\right)\right), \forall k_3 = 1,...,\prod_{i=1}^{n} m_i \quad (3)$$

where $F_{k_3}^{-1}(\ )$ is the reverse engineering of the hiding vector, *i.e.* the process of extracting secret data. $F_{k_3}^{-1}\left(p^*\left(kT\right)\right)$ refers to the process of extracting secret data from the media packets with hidden data.

To sum up, according to the multi-dimensional data hiding spatial model for steganography in media streams, the temporal and spatial hiding synchronization in covert VoIP communications is essentially a process of negotiating the hiding vectors of individual media packets.

4 Results and discussion

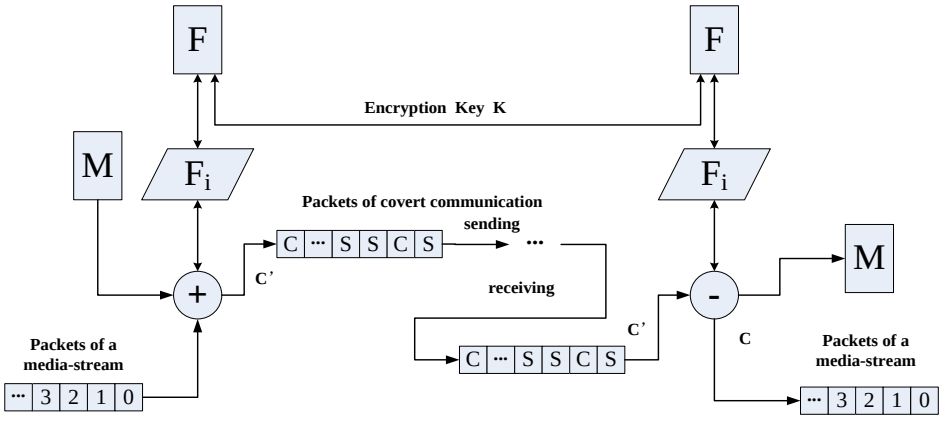
In this section, we detail an application of the proposed multi-dimensional data hiding spatial model to improving covert VoIP communications software called StegTalk. It enables the software to adopt a dynamic self-adaptive method of selecting hiding vectors, and a mechanism for negotiating hiding vectors via a covert channel.

4.1 Application of Spatial model to covert VoIP communications

In the experiments, VoIP streams were chosen as cover objects to conduct steganography experiments so as to verify the proposed multi-dimensional data hiding spatial model.

VoIP is currently the most typical media stream application on the Internet, and relevant technologies have been developed rapidly over the past 10 years. According to the data from IDC, VoIP usage in 2000 amounted to 15% of the long-distance communications, but it reached 70% in 2009. So the significance of exploring steganography in VoIP streams becomes clear in media streams-based covert communications.

On the basis of our previous studies [4] [5] [7] [9] [15] and others [16] [17] on VoIP steganographic algorithms, we first established a multi-dimensional data hiding spatial model and a hiding capacity set for covert VoIP communications, and then developed VoIP-based covert communications software with high data hiding capacity, high imperceptibility and high security. Taking this as an experimental platform, we consequently summarized the advantages of the proposed multi-dimensional data hiding spatial model through practical test and theoretical analysis.



Note: F represents the hiding capacity Set, $F=\{F_0, F_1 \cdots F_{15}\}$
M represents the secret data $M=U_{i \in n}\{0,1\}^i$
F represents the hiding -vector
K represents the key to control the selection of F_i

Figure 2 Implementation of covert VoIP communications using StegTalk.

The related work described in Section 2 shows that there are several algorithms for embedding data in VoIP streams and we have suggested some algorithms for steganography in VoIP [4][5][7][9][15][18]. Based on our previous research and the proposed multi-dimensional data hiding spatial model, we developed software called StegTalk for covert VoIP communications (Fig. 2).

For the VoIP media streams $P(k)=\sum_{k=1}^{N} p(kT)$, the hiding feature set (R) of the packet $p(kT)$ is given by

$$R=\{e_{RTP}, e_{PCM-LSB,} e_{G.723.1-LSB}, e_{G.729A-LSB}, e_{G.723.1-QIM}, e_{G.723.1-sil}, e_{G.729A-sil}\} \tag{4}$$

where e_{RTP} is the hiding feature using RTP header fields as cover objects [4], $e_{PCM-LSB}$ is the hiding feature using the lowest bit position in PCM code as cover objects [5][6][11][19][20], $e_{G.723.1-LSB}$ is the hiding feature using G.723.1 compression parameters as cover objects [7], $e_{G.729A-LSB}$ is the hiding feature using G.729A compression parameters as cover objects [7], $e_{G.723.1-QIM}$ is the hiding feature using G.723.1 codebooks as cover objects [9], $e_{G.723.1-sil}$ is the hiding feature using G.723.1 inactive voice frames as cover objects, and $e_{G.729A-sil}$ is the hiding feature using G.729A inactive voice frames as cover objects [18].

From the hiding feature set R of VoIP packets, ones can build various orthogonal hiding feature sets, $R_1^{\perp}$ to $R_5^{\perp}$, through analysis of the hiding process of the algorithms in the set R, given by

$$
\begin{aligned}
R_1^{\perp} &= \{e_{RTP}, e_{PCM-LSB}\} \\
R_2^{\perp} &= \{e_{RTP}, e_{G.723.1-LSB}, e_{G.723.1-QIM}\} \\
R_3^{\perp} &= \{e_{RTP}, e_{G.723.1-sil}, e_{G.723.1-QIM}\} \\
R_4^{\perp} &= \{e_{RTP}, e_{G.729A-sil}, e_{G.729A-QIM}\} \\
R_5^{\perp} &= \{e_{RTP}, e_{G.729A-LSB}, e_{G.729A-QIM}\}
\end{aligned} \tag{5}
$$

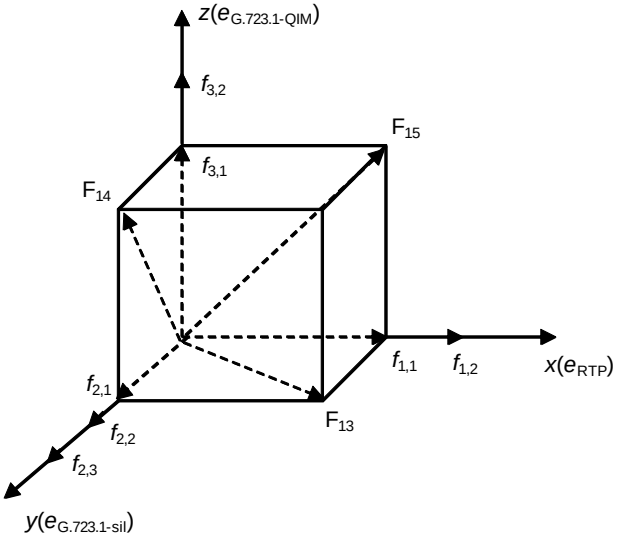


Figure 3 Scheme for $R_3^{\perp}$ 3-dimensional orthogonal hiding space.

The different orthogonal hiding feature sets above can then be used to constitute different multi-dimensional orthogonal hiding space. For example, $R_1^{\perp}$ is 2-dimensional hiding space, $R_2^{\perp}$ to $R_5^{\perp}$ are 3-dimensional orthogonal hiding space. Given that G.723.1 codec is the most commonly used audio encoding protocol in VoIP and in view of our previous research, the following focuses mainly on the $R_3^{\perp}$ 3-dimensional orthogonal hiding space, as shown in Fig. 3.

In Fig. 3, the coordinates $f_{1,1}$ and $f_{1,2}$ on the axis x(e_{RTP}) denote two different algorithms for hiding data in RTP header fields [4]. The former employs LSB substitution of RTP header timestamp with data hiding capacity of 2 bits/packet; the latter embeds data in the PT field of RTP header with data hiding ca-

pacity of 4 bits/packet. Similarly, the coordinates $f_{2,1}$, $f_{2,2}$ and $f_{2,3}$ on the axis y($e_{G.723.1\text{-}sil}$) refer to algorithms for hiding data in G.723.1 (6.3kbps) encoded inactive voice frames at different data hiding capacities of 16, 44 and 105 bits/frame [18]. The coordinates $f_{3,1}$ and $f_{3,2}$ represent two steganographic algorithms using G.723.1 quantization codebook [9], with data hiding capacities of 3 and 6 bits/frame, respectively. These algorithms in $R_3^{\perp}$ have been proved to be able to withstand steganalysis presented in [21][22].

Table 1 Some commonly used hiding vectors

Vector no	Hiding vector	Hiding metafunction	Hiding capacity	Imperceptibility
00	F_0	Reserved	Reserved	Reserved
01	F_1	$f_{1,1}$, $f_{2,1}$, $f_{3,1}$	2+16*m+3*n	f_{21}
02	F_2	$f_{1,2}$, $f_{2,1}$, $f_{3,1}$	9+16*m+3*n	f_{21}
03	F_3	$f_{1,1}$, $f_{2,2}$, $f_{3,1}$	2+44*m+3*n	f_{22}
04	F_4	$f_{1,2}$, $f_{2,2}$, $f_{3,1}$	9+44*m+3*n	f_{22}
05	F_5	$f_{1,1}$, $f_{2,3}$, $f_{3,1}$	2+105*m+3*n	f_{23}
06	F_6	$f_{1,2}$, $f_{2,3}$, $f_{3,1}$	9+105*m+3*n	f_{23}
07	F_7	$f_{1,1}$, $f_{2,1}$,	2+16*m+6*	f_{21}

		$f_{3,2}$	n		
08	F_8	$f_{1,2}$,	$f_{2,1}$,	7+16*m+6*	f_{21}
		$f_{3,2}$	n		
09	F_9	$f_{1,1}$,	$f_{2,2}$,	2+44*m+6*	f_{22}
		$f_{3,2}$	n		
10	F_{10}	$f_{1,2}$,	$f_{2,2}$,	7+44*m+6*	f_{22}
		$f_{3,2}$	n		
11	F_{11}	$f_{1,1}$,	$f_{2,3}$,	2+105*m+6	f_{23}
		$f_{3,2}$	*n		
12	F_{12}	$f_{1,2}$,	$f_{2,3}$,	7+105*m+6	f_{23}
		$f_{3,2}$	*n		
13	F_{13}	$f_{1,1}$, $f_{2,1}$		2+16*m	f_{21}
14	F_{14}	$f_{2,1}$, $f_{3,1}$		16*m+3*n	f_{21}
15	F_{15}	$f_{1,1}$, $f_{3,1}$		2+3*n	f_{31}

According to the theorem for hiding algorithm orthogonality, the coordinates on the three axes are reciprocally orthogonal. So they can be used to constitute different hiding vectors, as listed in Table 1. The value for *n* is the number of audio frames in a RTP packet, *m* is the number of inactive voice frames, and $m <= n$. In the light of the definition of imperceptibility of hiding vectors, the imperceptibility of a hiding vector equals the lowest imperceptibility among its hiding metafunctions.

In this study, secret data are segmented into various bit-strings that are hidden in different dimensions by using different related algorithms, respectively. Such a multi-dimensional hiding method can improve the imperceptibility of hidden data.

The hiding vectors listed in Table 1 can constitute a hiding capacity set for covert VoIP communica-

tions. Considering the vector length affects synchronization efficiency, the hiding capacity set F does not contain the hiding vectors with lower data hiding capacity. The hiding capacity set is composed of 15 commonly used hiding vectors, given by

$$F=\{F_0, F_1, F_2, F_3, F_4, F_5, F_6, F_7, F_8, F_9, F_{10}, F_{11}, F_{12}, F_{13}, F_{14}, F_{15}\} \quad (6)$$

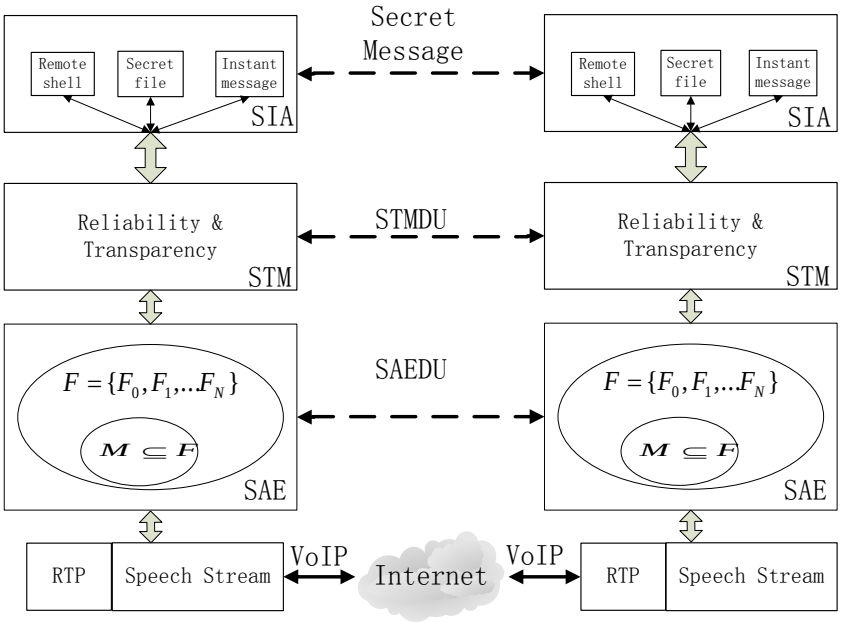


Figure 4 Improved hierarchical model for hiding vectors.

Based on $R_3^{\perp}$ 3-dimensional hiding space and the hiding capacity set F, we developed VoIP-based covert communications software called StegTalk by using our hierarchical model for covert VoIP communications [15]. In the hierarchical model, we suggested a 3-layer covert communications framework to ensure the reliability and adaptability of different algorithms for steganography in VoIP in case of packet loss. As shown in Fig. 4, the improved hierarchical model is suggested to adapt to the hiding vector. The steganographic information application (SIA) layer provides three types of services for end-users, Remote shell, Secret file and Instant message, forwarding streams to the steganographic transmission management (STM) layer, enabling reliable and transparent transmission of streams in an

end-point-to-end-point manner. STM solves the packet loss problem by using the fast starting retransmission algorithm, *i.e.* re-transmitting the packet once three irregular orders of ACK occur without waiting until the expiry of the timer RTO, so as to achieve a reasonable conveying rate of the secret message. The steganographic adaptation and embedding (SAE) layer is responsible for embedding the message, which is performed under the control of a key by using a hiding vector chosen from the hiding vector capacity set. So the hiding vectors can be chosen dynamically and adaptively for covert communications.

StegTalk uses a dynamic mechanism for negotiating hiding vectors between the communicating parties to achieve hiding synchronization. With StegTalk, a special covert channel over the RTCP header field is used to transmit the hiding vector number, accomplishing out-band synchronization. Details of the synchronization algorithm are described below.

(a) For the VoIP media streams $P(k)=\sum_{k=1}^{N} p(kT)$, the VoIP packet $p(kT)$ (k=1,…,n) is chosen as the data hiding unit. The hiding vector number used by each VoIP packet is transmitted by a covert channel over the LSR (Packet loss rate) header field of the corresponding RTCP packet. The detailed description is referred to our previous work [23].

(b) The mapping between the VoIP packet $p(kT)$ and the RTCP packet is carried out through correlating the timestamp field values of their header fields, as shown in Fig. 4. If the timestamp field values are the same, the VoIP packet $p(kT)$ and the RTCP packet are corresponding.

(c) If the hiding vector number transmitted by the covert channel over the RTCP header field is 0, the corresponding VoIP packet contains no hidden information; otherwise, secret information is embedded in the VoIP packet, and the corresponding steganographic algorithm searches the hiding vector by the hiding vector number from the hiding capacity set of the terminal.

(d) In covert VoIP communications, the appearance that the hiding vector number transmitted by the covert channel over a RTCP packet is not 0, indicates the beginning of covert communications; the $p(kT)$

packet and subsequent packets in VoIP streams use the same hiding vector to embed data until the appearing of a different hiding vector number, as shown in Fig. 5.

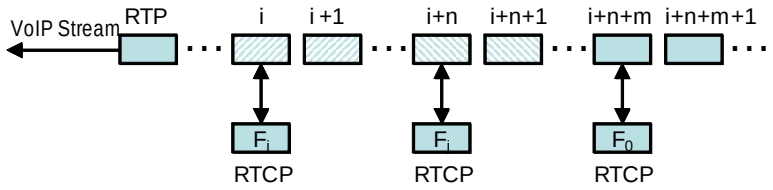


Figure 5 Mechanism for negotiating the hiding vector numbers to achieve hiding synchronization.

In Fig. 5, the first RTCP packet uses the hiding vector F_i; through correlating the timestamp field of the header, the corresponding packet $p(k_iT)$ starts data embedding, and all the packets between $p(k_1T)$ and $p((k_1+n\text{-}1)T)$ use the same hiding vector F_i to embed data. From the $p((k_i+n)T)$ packet, the hiding vector F_j is used for data hiding until the $p((k_i+n+m\text{-}1)T)$ packet. At the beginning of the $p((k_i+n+m)T)$ packet, no data embedding occurs, as the corresponding RTCP sets the hiding vector F_0. The process is called ‘hiding vector negotiation mechanism’.

4.2 Performance and security analysis

We conducted testing on covert VoIP communications, performance and security analysis, thereby summarizing the advantages of the proposed multi-dimensional data hiding spatial model.

4.2.1 Data hiding capacity

According to the proposed data hiding spatial model, the data hiding capacity of a hiding vector is greater than any data hiding capacity of its vector components. The model makes the breakthrough necessary to solve the contradiction problems between data hiding capacity and imperceptibility when the steganographic algorithm is based solely on a single hiding feature. Use of the hiding vector can im-

prove the data hiding capacity of covert communications significantly, *i.e.* the data hiding rate, without compromising imperceptibility.

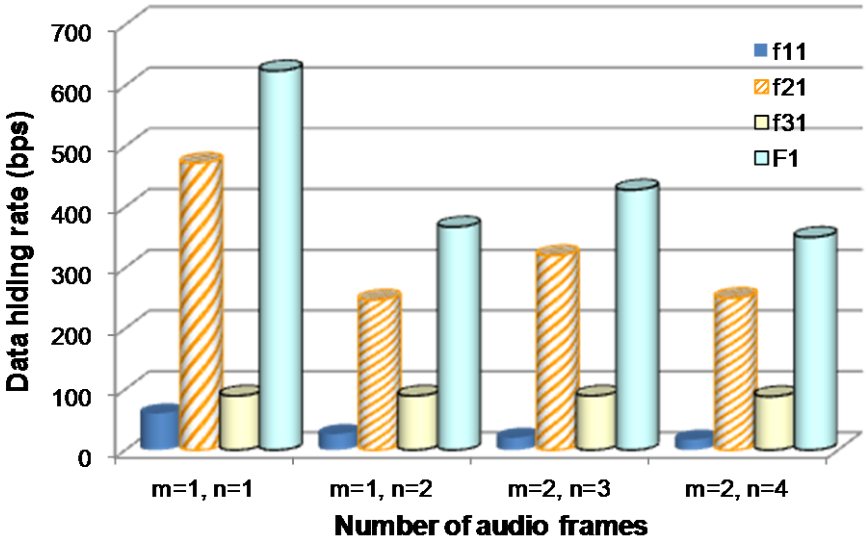


Figure 6 Data hiding rates (bps) when using StegTalk for covert VoIP communications.

Fig. 6 shows the real data hiding rates when different hiding vectors were used in the covert communications software StegTalk to embed data in VoIP streams. The results were obtained by assuming a RTP packet contains *n* audio frames, *m* of which are inactive voice frames, without considering packet loss. As Fig. 6 shows, the data hiding rates were much higher when the hiding vector F_1 was used in StegTalk for covert communications, compared with an individual hiding metafunction f_{11}, f_{21} or f_{31} being used as the steganographic algorithms, respectively.

4.2.2 *Imperceptibility of covert VoIP communications*

The imperceptibility of covert VoIP communications is a measure of evaluating the quality of the stego speech. We adopted the ITU P.862 recommendation to measure the subjective quality of the stego speech. The recommendation describes an objective method for predicting the subjective quality of narrow-band speech codec. It uses the perceptual evaluation speech quality (PESQ) value to assess the sub-

jective quality of the stego speech.

The testing results with the ITU P.862 method is listed in Table 2. Four groups of audio streams (CM, CF, EM, and EF) were chosen as cover objects for covert VoIP communications in the experiments. Table 2 shows comparisons in audio quality between non-covert communications and the covert communications that used the hiding vector F_1 to hide secret information in four different VoIP streams. Comparisons in PESQ values between the covert communications using the hiding vector F_1 and those using the vector components (f_{11}, f_{12}, and f_{13}) are also available in the table.

Table 2 Comparisons in audio quality between covert and non-covert communications

Audio streams	P.862 PESQ Value (Mean)				
	No hiding	f_{11}	f_{21}	f_{31}	F_1
CM	3.49	3.49	3.49	3.43	3.43
CF	3.34	3.34	3.34	3.31	3.31
EM	3.30	3.30	3.30	3.28	3.28
EF	3.30	3.30	3.30	3.25	3.24
Average	3.36	3.36	3.33	3.32	3.31

Analysis of the data listed in Table 2 shows, the imperceptibility (audio quality) of the covert communications using hiding vectors did not decrease, while the data hiding rate increased significantly. In addition, theoretical analysis reveals that segmenting the secret message into various sections and then hiding these sections in different dimensions of the hiding vector can improve the imperceptibility of the secret information. This explains our experimental results perfectly.

4.2.3 *Security against steganalysis*

In the proposed multi-dimensional steganography in media streams, the secret message is first segmented into blocks according to the data hiding capacities of the hiding vectors in corresponding packets; each block is further divided into bit-strings according to the data hiding capacities of the metafunctions that constitute the hiding vector. Each metafunction hides a segment of different size of the secret message. For instance, the secret message M_2 consisting of 10 letters of the alphabet has 80 bits as follows:

M_2=10101101010011110010101010101100111100101011010101010001101001101100111100110001

Suppose the vectors F_1, F_2, F_{13} and F_{14} in Table 1 are used to transmit the secret message M_2. According to formulas for calculating the hiding capacity (Table 1), M_2 can be divided into four segments; each segment is divided into bit strings of different sizes, which are embedded into different metafunctions of the hiding vector, as shown in Table 3.

As Table 3 shows, each metafunction (associated with a hiding feature) contains only a bit string of the secret message. Since most of existing VoIP steganalysis methods are only effective in detecting a specific steganographic algorithm based on a single hiding feature, they are unlikely to detect the whole secret message that are embedded in multiple hiding features. The following experimental results prove that the proposed multi-dimensional steganography greatly improved the security of covert VoIP communications.

Table 3 Multi-dimensional embedding in metafunctions of the hiding vector

Hiding vector	Capacity	Secret data segmenting	Metafunction

F_1	21	10101101	$f_{1,1}$={10}
		01001111	$f_{2,1}$={10110101001
		00101	11100 }
			$f_{3,1}$={101}
F_2	25	01010101	$f_{1,2}$={0101}
		10011110	$f_{2,1}$={01011001111
		01010110	0010101}
		1	$f_{3,1}$={101}
F_{13}	18	01010100	$f_{1,1}$={0`}
		01101001	$f_{2,1}$={01010001101
		10	00110}
F_{14}	19	11001111	$f_{2,1}$={11001111001
		00110001	10001}
		XXX	$f_{3,1}$={XXX}

Four groups of audio streams (CM, CF, EM, and EF) were used as cover objects, each containing three audio pieces of 10, 5 and 3 seconds in length with 333, 166 and 100 frames, respectively. Table 4 lists the hiding algorithms and hiding vectors used in the experiments. CNV denotes the CNV-based data embedding algorithm proposed in [9]; LSB stands for data embedding into the last bits of the parameters of encoded audio frames. The cover objects were categorized into three groups, the first used for data hiding using the CNV-based algorithm, the second using the LSB algorithm [19] to embed data, and in the last group, all the frames of each audio sample were divided into three pieces which were then used to hide data using the three hiding vectors listed in Table 4.

Table 4 Hiding vectors used

Hiding vectors	Payload	$f_{1,1}$	$f_{2,1}$	$f_{3,1}$
F_1	3 bits	CNV：3bits	/	/
F_2	24 bits	CNV：3bits	LSB：21bits	/
F_3	25 bits	/	LSB：22bits	3

We used the Mel frequency cepstral coefficients (MFCC) method, one of the most effective steganalysis methods [24], to perform security analysis, so as to verify the security of the proposed multi-dimensional steganography. Table 5 lists the detection accuracy of using MFCC to detect the hidden message for CNV-based steganography, LSB-substitution based single steganography (LsbOnlyEmbed) and multi-dimensional steganography (MultidEmbed), respectively.

Table 5 Detection accuracy of MFCC steganalysis

Length of audio pieces	CNV	LsbOnlyEmbed	MultidEmbed
10s (333 frames)	54.50%	92.20%	81.26%
5s (166 frames)	51.94%	87.84%	72.15%
3s (100 frames)	51.62%	81.92%	65.24%

Table 5 shows MFCC was unable to detect the hidden data embedded by using the CNV algorithm, it

was very effective in detecting the hidden data embedded with the LsbOnlyEmbed algorithm, but it was unlikely to detect the multi-dimensional steganography. For the 3-second samples, the accuracy of detecting single steganography reached 81.92%, whereas that of detecting multi-dimensional steganography achieved 65.24% only, which is an extremely poor result for steganalysis.

The experimental results shows, for the same cover sample, it was unlikely to detect multi-dimensional steganography compared with single steganography. So the proposed multi-dimensional steganography could improve the security of covert communications. Even if part of the secret message hidden in one dimension had been detected, it would have been unlikely to extract the whole message as other parts of the message were hidden in other dimensions.

4.2.4 *Hiding vector negotiation mechanism*

Synchronization efficiency is defined as the ratio of the number of bits of the secret information *m* to the number of all the bits transmitted over a covert communication section, given by

$$\rho = m / C$$

where ρ is the synchronization efficiency, *m* is the number of bits of the secret information, and *C* is the number of bits of the stego streams consisting of the secret information and the controlling bits such as synchronization controlling bits.

In one of the most influential literatures [12], the authors described their VoIP-based covert communications software, which gained synchronization efficiency of 45% - 91.2%. A hierarchical model for covert VoIP communications was suggested in [15], attaining synchronization efficiency of 78% - 92% by means of an effective method of optimizing communication efficiency. There is still room for further improvement. If the hiding vector synchronization algorithm, suggested by the proposed multi-dimensional data hiding spatial model, were used for data hiding, synchronization efficiency would reach 86% - 97%. The above data analysis indicates covert communications using the proposed hiding

vector negotiation mechanism can achieve the highest synchronization efficiency.

Our study also shows, with the hiding vector negotiation mechanism, synchronization efficiency is related to the updating frequency of hiding vectors of media packets. Fig. 7 shows the relationship between the synchronization efficiency and the updating frequency of hiding vectors using StegTalk. If each RTP packet uses a different hiding vector to hide information, the updating frequency of hiding vectors maximizes, and the overhead of hiding synchronization maximizes as well, thus resulting in the lowest synchronization efficiency; on the contrary, if ten RTP packets share the same hiding vector, it leads to the highest synchronization efficiency.

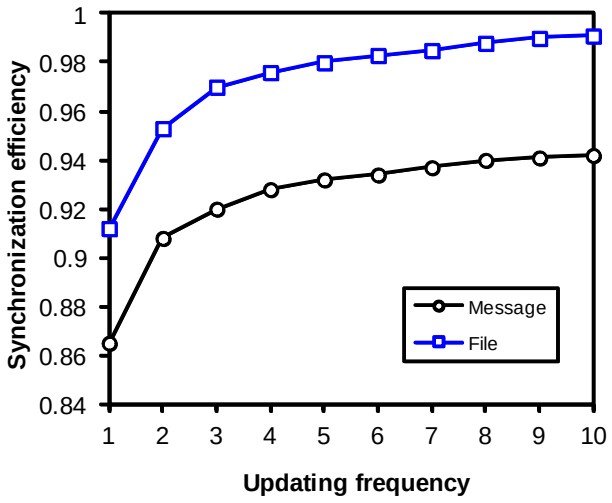


Figure 7 Relationship between synchronization efficiency and the updating frequency of hiding vectors.

The results shown in Fig. 7 are based on the test condition that the percentage of inactive audio frames in a VoIP stream was about 70%. In the experiments, apart from 5 bits being used for implementing hiding vector synchronization, 3 bits were introduced to mark the end position of the secret information. If the secret information is an instant message, the overhead for marking the end position is higher; however, it uses fewer bits to mark a secret file. So the synchronization efficiency in transmitting a secret file

over covert communications using StegTalk software is much higher than those for transmitting an instant message (see Fig. 7).

5 Conclusions

In this study, we have built a multi-dimensional data hiding spatial theoretical model for covert VoIP communications, breaking through the bottleneck problem of low data hiding capacity encountered while implementing a sole conventional steganographic algorithm. A new concept of data hiding vectors has been introduced to improve the synchronization efficiency by reducing overhead. The proposed hiding vector negotiation mechanism for covert VoIP communications has been adopted to withstand against detection of steganalysis. Testing on the application of the spatial model to VoIP streams has shown that the covert VoIP communications can improve data hiding capacity, imperceptibility, as well as synchronization efficiency to a great extent.

Exploring other methods for constructing data hiding vectors, and improving the imperceptibility of the combined hiding vectors through balancing the imperceptibility of hiding vector components for each vector are the subjects of future work to break through the Buckets effect limitation.

This work was supported by the National Natural Science Foundation of China (*Grant Nos. 61271392, U1405254 and 61272469*), *and the British Government under Grant ktp008263.*

1 S. Zander, G. Armitage and P. Branch, “Covert channels and countermeasures in computer network protocols,” IEEE Communications Magazine, vol. 45, no. 12, pp. 136-142, December, 2007.

2 X. Luo, E.W.W. Chan and R.K.C. Chang, “TCP covert timing channels: Design and detection,” in Proc. of IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), pp. 420-429, Anchorage, Alaska, June 24-27, 2008.

3 S.H. Sellke, C. Wang, S. Bagchi and N. Shroff, “TCP/IP timing channels: Theory to implementa-

tion," in Proc. of 28th IEEE International Conference on Computer Communications, pp. 2204-2212,Rio de Janeiro, Brazil, April 19-25, 2009.

4 L.Y. Bai, Y. Huang, G. Hou and B. Xiao, "Covert channels based on Jitter field of the RTCP header," in Proc. of the Fourth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, pp. 1388-1391, Harbin, August 15-17, 2008.

5 Y. Huang, B. Xiao and H. Xiao, "Implementation of covert communication based on steganography," in Proc. of 4th International Conference on Intelligent Information Hiding and Multimedia Signal Processing, pp. 1512-1515, Harbin, China, August 15-17, 2008.

6 N. Aoki, "A technique of lossless steganography for G.711 telephony speech," in Proc. of Fourth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, pp. 608-611, Harbin, August 15-17, 2008.

7 Y. Su, Y. Huang and X. Li, "Steganography-oriented noisy resistance model of G.729a," in Proc. of 2006 IMACS Multi-conference on Computational Engineering in Systems Applications, pp. 11-15, Beijing, China, 2006.

8 L. Liu, M. Li, Q. Li and Y. Liang, "Perceptually transparent information hiding in G.729 bitstream," in Proc. of 4th International Conference on Intelligent Information Hiding and Multimedia Signal Processing, pp. 406-409, Harbin, August 15-17, 2008.

9 B. Xiao, Y. Huang and S. Tang, "An approach to information hiding in low bit-rate speech stream," in Proc. of IEEE Global Telecommunications, pp. 371-375, USA, 2008.

10 W. Mazurczyk and J. Lubacz, "LACK - a VoIP steganographic method," Telecommunication Systems Journal, vol. 45, no. 2-3, pp 153-163, October, 2010.

11 Józef Lubacz, Wojciech Mazurczyk, Krzysztof Szczypiorski, "Principles and overview of network steganography," IEEE Communications Magazine, vol. 52, no. 5, pp. 225-229, 2014.

12 Druid, "Real-time steganography with RTP," Available: http://druid.caughq.org, 01 December 2011.

13 C. Kratzer, J. Dittmann, T. Vogel and R. Hillert, “Design and evaluation of steganography for Voice-over-IP,” in Proc. of 2006 IEEE International Symposium on Circuits and Systems, pp. 2397-2340, Island of Kos, Greece, 2006.

14 C. Cachin, “An information-theoretic model for steganography,” Information and Computation, vol. 192, no. 1, pp. 41-56, 2004.

15 X. Bo and Huang Yongfeng, “Reliable transmission of information hiding communication over stream-media,” Journal of Xidian University, vol. 35, no. 3, pp. 554-558, 2008.

16 W. Mazurczyk and Z. Kotulski, “Covert channel for improving VoIP security,” in Proc. of 2006 Multi-conference on Advanced Computer Systems, pp. 311-320, Międzyzdroje, Poland, 2006.

17 C. Wang and Q. Wu, “Information hiding in real-time VoIP streams,” in Proc. of 9th IEEE International Symposium on Multimedia, pp. 255-262, Taichung, Taiwan, 2007.

18 Y.F. Huang, S. Tang, and J. Yuan, “Steganography in inactive frames of VoIP streams encoded by source codec,” IEEE Transactions on Information Forensics and Security, vol. 6, no. 2, pp. 296-306, 2011.

19 X. Li, B. Yang, D. Cheng, and T. Zeng, “A generalization of LSB matching,” IEEE Signal Processing Letters, vol. 16, no. 2, pp. 69-72, 2009.

20 N. Aoki, “A band extension technique for G.711 speech using steganography,” IEICE Transactions on Communications, vol. 89, no. 6, pp. 1896-1898, 2006.

21 Y. Huang, S. Tang, C. Bao, and Y.J. Yip, “Steganalysis of compressed speech to detect covert voice over Internet protocol channels,” IET Information Security, vol. 5, no. 1, pp. 26-32, 2011.

22 Y. Huang, S. Tang, and Y. Zhang, “Detection of covert voice over Internet protocol communications using sliding window-based steganalysis,” IET Communications, vol. 5, no. 7, pp. 929-936, 2011.

23 Huang Yong-feng, Yuan Jian, and Chen Mingchao, “Key distribution in the covert communication based on VoIP,” Chinese Journal of Electronics, vol. 20, no. 2, pp. 357-361, 2011.

[24] Q.Z. Liu, A.H. Sung, and M.Y. Qiao, “Temporal derivative-based spectrum and mel-cepstrum audio steganalysis,” IEEE Transactions on Information Forensics and Security, vol. 4, no. 3, pp. 359-368, 2009.