Circuits Syst Signal Process (2008) 27: 195–211
DOI 10.1007/s00034-008-9023-1

Design and Analysis of High-Capacity Anti-Collusion Hiding Codes

Byung-Ho Cha · C.-C. Jay Kuo

Received: 6 June 2007 / Revised: 7 October 2007 / Published online: 4 March 2008

Abstract A new methodology for the design of high-capacity anti-collusion hiding codes in a large-scale fingerprint-based traitor tracing system is proposed in this work. We consider a hiding code of MN bits, where M bits are used as the user ID and N bits are the length of the spreading codes. Since each user is assigned one out of 2^M ID numbers and one out of N spreading codewords, the total number of users is equal to $2^M N$. To accommodate an even larger number of users, we propose a shifted spreading scheme that shifts the spreading codeword circularly by a certain amount. By allowing P shifts (with P mod N), the total number of users increases from $2^M N$ to $2^M NP$. When multiple users perform a weighted collusion attack, we show that the task of identifying colluders and their attack weights can be formulated as a user detection and channel estimation problem in a multiuser wireless communication system with a multipath fading channel. For the latter, there exist code design techniques that choose the spreading codes carefully so as to reduce multiaccess interference of users with different spreading codewords effectively. By exploiting this analogy, we develop an anti-collusion code called OSIFT (Orthogonal Spreading followed by the Inverse Fourier Transform). We compare several hiding codes in a fingerprinting system consisting of hundreds of colluders. It is demonstrated by computer simulation that, when the OSIFT code is adopted, colluders and their attack weights can be found by the proposed fingerprinting system more accurately.

Keywords Watermarking · Fingerprinting · User capacity · Collusion attack · Weighted collusion attack · Shifted spreading · Anti-collusion codes · Hiding codes

B.-H. Cha (✉) · C.-C.J. Kuo
Ming Hsieh Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089-2564, USA
e-mail: byungcha@usc.edu

C.-C.J. Kuo
e-mail: cckuo@sipi.usc.edu

1 Introduction

Multimedia contents can be massively and effectively distributed over wired and wireless networks nowadays. The advance of content distribution networks poses a serious threat to content protection against illegal copying and re-distribution. Although encryption techniques play an important role in content protection, they are no longer useful after contents are decrypted. To address this issue, the idea of *fingerprinting for traitor tracing* has been proposed in recent years. That is, hiding codes are embedded in media contents and used to detect traitors who create and distribute unauthorized copies.

One powerful scheme to break the fingerprint-based traitor tracing system is the collusion attack [10, 11, 13, 31]. Users that have the same content but different hiding codes (or fingerprints) may merge their received copies via linear combination with an attempt to remove their individual codes without much degradation of media quality. When the attack is successful, the traitor tracing system will not be able to identify participating attackers from this newly generated copy. The collusion attack is a true concern since participants do not need any background on multimedia signal processing, and its computational complexity is low.

Although some hiding codes developed in the context of watermarking are robust against noise, compression and various manipulation attacks, they do not resist the collusion attack well so that they are not suitable candidates for fingerprinting applications. The design and analysis of anti-collusion codes has been an important research topic for years. Some of the previous work in this area will be reviewed in Sect. 2. Despite these research efforts, the total number of users (i.e., the numbers of users that can be accommodated in a traitor tracing system) and the possible maximum number of colluders have not yet been thoroughly studied. When a media content is distributed to a larger number of users, there is a higher probability for more colluders to participate in an attack. It is important to design a hiding code that is effective in a large-scale traitor tracing system, which is the main focus of our current research.

We take a new approach to anti-collusion code design by drawing an analogy between the fingerprinting system and the multiuser wireless communication system. There are several ways to allow multiple users to access the shared channel, e.g., TDMA (time-division multiaccess), FDMA (frequency-division multiaccess), CDMA (code-division multiaccess), MC-CDMA (multicarrier code-division multiaccess) and OFDMA (orthogonal frequency division multiaccess) [20]. One goal of a wireless multiaccess communication system is to accommodate as many users as possible in a shared channel of finite capacity while meeting a certain detection performance. The design of our fingerprinting system is highly motivated by the MC-CDMA system in broadband communications. Specifically, we consider a hiding code of MN bits, where M bits are used as the user ID and N bits are the length of spreading codes. Since each user is assigned one out of 2^M ID numbers and one out of N spreading codewords, the user capacity[1] is $2^M N$. To accommodate an even

[1]The term *user capacity* is used for both total number of users and number of colluders. Also, the term *capacity* in this paper has the same meaning as *user capacity*. A general term *user capacity* can be found in [19, 24].

larger number of users, we propose a shifted spreading scheme that shifts the spreading codeword circularly by a certain amount. By allowing P shifts (with P mod N), the user capacity increases from $2^M N$ to $2^M NP$.

We call users with the same spreading codeword but different shifts to be in the same group. It is convenient to classify collusion attacks into two types as explained below. In the first case, all colluders are from the same group. When they perform a weighted collusion attack, the shifted spreading operation and their weights correspond to delayed transmission paths and their channel coefficients, respectively. In wireless communications, this case maps to a single user communication problem. The task of identifying colluders and their attack weights can be formulated as a single user detection and channel estimation problem. There are many existing solutions to address this problem [8, 21].

In the second case, colluders may come from different groups with various shifts. We can map each group of colluders into a single user with a multipath fading channel in wireless communication, and then the multigroup colluder detection problem is equivalent to multiuser detection in multipath fading channels. This is a well-studied problem in multiaccess communication systems. The interference among users is called multiaccess interference (MAI), where the power of the code of a target user is lower than that of other interfering users. The user capacity of a wireless communication system is bounded by MAI and the length of multipath fading channels. For more details, we refer to [23]. One way to alleviate the MAI effect is to design MAI-free spreading codes [17, 18].

In the fingerprint system, we view a host media content as a channel. Each user is assigned a hiding code as his/her fingerprint. As a result of the weighted collusion attack, hiding codes of colluders from different groups experience MAI so that they are more difficult to detect. By following the work in [17, 18], we have developed an anti-collusion code called OSIFT (Orthogonal Spreading followed by the Inverse Fourier Transform) [3–5]. The OSIFT code was initially proposed in [3]. Several variations of the OSIFT code were considered in [4]. The idea of shifted spreading to increase the user capacity was discussed in [5]. We integrate all main results in our previous work in this paper to make the process of high-capacity anti-collusion codes design more transparent and accessible to researchers in this field.

Here, we introduce the OSIFT codes in a more natural way and analyze the performance more thoroughly. We also compare several hiding codes in a fingerprinting system consisting of hundreds of colluders, and show that colluders and their attack weights can be found by a simple correlation-based detector more accurately when OSIFT is adopted as the spreading code. The rest of this paper is organized as follows. Related previous work is reviewed in Sect. 2. The proposed fingerprinting system is described in Sect. 3, and the design of OSIFT spreading codes is discussed in Sect. 4. The detection analysis of OSIFT spreading codes is presented in Sect. 5. Simulation results using an audio watermarking system as an example are reported in Sect. 6. Our concluding remarks are given in Sect. 7.

2 Review of Previous Work

A series of research has been performed in the last decade to deal with the threat of collusion attacks. In the early research stage, Cox et al. [7] proposed a spread spectrum (SS) watermark generation and embedding technique, where codes are generated randomly by an independently identically distributed (i.i.d.) uniform or Gaussian source. It was shown experimentally that the resulting codes are robust against average collusion attacks to a certain degree. However, no quantitative analysis was provided. To design a specific code that resists the collusion attack, Boneh and Shaw [2] proposed a collusion-secure (CS) code based on the principle of marking assumption. Their method can catch one out of c colluders with a high probability if these colluders are not able to change the state of undetectable marks. However, their method has some limitations. First, it works well only for symbol sequences but not multimedia data, since the latter can be manipulated through noise and re-quantization. Second, the code length is very long for a large value of c because it is proportional to $c^2 \log^2 L$ to support a total number of L users. Since the pioneering work of [2] and [7], research has been conducted along two directions. One is to focus on better anti-collusion code design while the other is to provide deeper forensic performance analysis.

To improve the performance of the CS code, Trappe et al. [16] proposed an AND-anti-collusion code (AND-ACC) based on orthogonal code modulation. The marking assumption in the CS code is replaced by a combinatorial design and enhanced by accumulation of pseudo noise (PN) codes called ACC. AND-ACC achieves better performance in colluder detection. Besides, it can be applied to general Gaussian signals and images assisted by correlation detection. More recently, He and Wu [10] proposed another way to extend the CS codes based on ACC and considered a cross-layer design to separate coding and modulation in multimedia contents.

Su, Eggers and Girod [14] analyzed the performance of i.i.d. and CS codes by considering an optimal collusion attack and studying the lower bound of distortion caused by the attack. Recently, Wang et al. [26] investigated the error performance of ACC using maximum and threshold detectors and proposed a scheme to estimate the size of colluders. Zhao et al. [31] extended forensic analysis to unbounded Gaussian (UG) and bounded Gaussian (BG) codes under non-linear collusion attacks, e.g., max, min, median operations. The research in [26] and [31] provided lower and upper bounds on probability error functions with respect to specific collusion attack models and conducted a thorough performance analysis on the relationship among the code length, the user, and the colluder numbers. Zhao and Liu [28, 30] examined more effective collusion attacks, which are achieved by pre-collusion operations of selfish colluders, and tried to understand colluders' behavior in minimizing their risk. Swaminathan, He and Wu [15] extended the analysis to quantization index modulation (QIM) and spread transform dither modulation (STDM). They showed that STDM with orthogonal spreading vectors is more robust than QIM, but the performance of STDM is much worse than that of the SS method.

Chu, Qiao and Nahrstedt [6] proposed a two-layer fingerprinting design and applied CS codes for leakage identification. Although the scheme proposed in [6] is not robust against various collusion attacks, the effort in addressing a large number

of users in a system is interesting and meaningful. Wang et al. [25] showed some disadvantages of the two-layer fingerprinting design, and proposed a group-oriented fingerprinting method to enhance the performance of orthogonal modulation codes under massive content distribution. That is, one may assign correlation fingerprints to potential colluders and independent fingerprints to trusted members. Zhao and Liu [29] extended the group-oriented fingerprinting scheme and applied it to the video multicast application. They considered a scalable video codec and proposed a method that integrates TDMA and CDMA embedding.

3 System Model

Codeword spreading with shifts is shown in Fig. 1. A fingerprinting system for hiding code embedding, extraction and detection is shown in Fig. 2. Two embedding schemes are considered in the embedder; namely, traditional and shifted spreading. The detection scheme in the detector is enhanced by channel estimation and equalization. Finally, a weighted collusion attack is presented. For convenience, the notation used in this paper is summarized in Table 1.

3.1 Hiding Code Embedding and Detection

By following the work of Bassia et al. [1], we employ a time-domain embedding scheme as described below. Let $x_l(i)$, $i = 0, \ldots, T-1$, be the host signal for user l of length $T = MN$. We can divide it into M segments, each of which has N samples as

$$x_{l,m}(i) = x_l(m \cdot N + i), \quad m = 0, \ldots, M-1,\ i = 0, \ldots, N-1. \tag{1}$$

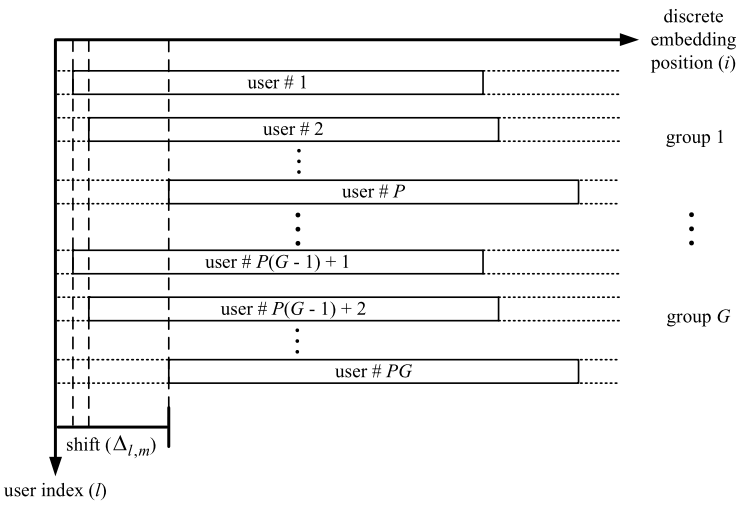

Fig. 1 Codeword spreading with shifts

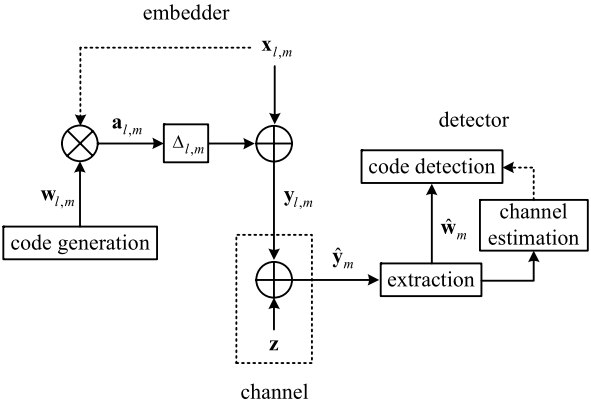

Fig. 2 An overview of the hiding code embedding, extraction and detection system

Table 1 Notation

Terms and definitions			
x, X	Host signal	$\mathbf{u}$, U	User ID
y, Y	Marked signal	$\mathbf{m}$, M	User message
Δ	Shift	s, $\mathbf{S}$	Spreading code
τ	Threshold	w, W	Hiding code
N	Length of spreading code	M	Length of user ID
k, K	Colluded user	l, L	Total user
p, P	User in group	g, G	Group
e, E	Noise	v	Statistics
λ, Λ	DFT response	h, $\mathbf{H}$	Impulse response

Each segment $x_{l,m}(i)$ carries a 1-bit user identification (ID) code, so that the user identification code, denoted by $u_{l,m}$, has a total length of M. This 1-bit user ID is spread by a code $s_{l,m}(i)$. We adopt the following additive embedding method:

$$y_{l,m}(i) = x_{l,m}(i) + a_{l,m}(i) \tag{2}$$

where $a_{l,m}(i)$ is the embedding signal and $y_{l,m}(i)$ is the output signal. They are of length MN. Generally, we can express $a_{l,m}(i)$ as

$$a_{l,m}(i) = \alpha \big| x_{l,m}(i) \big| w_{l,m}(i) \tag{3}$$

where $w_{l,m}(i)$ is the hiding code for user l and α is a variable (or constant) that adjusts the embedded code strength. Hiding code $w_{l,m}(i)$ of MN bits is generated by M-bit user ID $u_{l,m}$, N-bit spreading code $s_{l,m}(i)$, and the inverse Fourier transform (IFT). Since each user can be assigned one out of 2^M possible ID numbers and one out of N spreading codes, the total user capacity of the above fingerprinting system is equal to $2^M N$.

One way to enlarge user capacity is to circularly shift a hiding code by a certain amount as shown in Fig. 1. In this figure, we have G groups of users. Each group shares the same spreading codeword. If there are N spreading codewords in total, we

have $G = N$. Each group can support P users due to the application of a circular shift with amount $\Delta_{l,m}$, where l is the user index. Thus, the user capacity increases from $2^M G$ to $2^M GP$. For an N-bit codeword, it is clear that P mod N. Furthermore, to be robust against the collusion attack, we demand P to be a small integer number. The case of $P = 2$ is reported in the computer simulation section in this work. Results of a larger P value will be examined in our future work.

With shifted spreading, the embedding of hiding codes can be re-written as

$$y_{l,m}(i - \Delta_{l,m}) = x_{l,m}(i - \Delta_{l,m}) + a_{l,m}(i - \Delta_{l,m}) \tag{4}$$

where

$$a_{l,m}(i - \Delta_{l,m}) = \alpha \big| x_{l,m}(i - \Delta_{l,m}) \big| w_{l,m}(i). \tag{5}$$

Under the antipodal model and synchronization assumptions, the output of the correlation detector is given by

$$v_{l,m} = \frac{\sum_{i=0}^{N-1} (\hat{y}_m(i) - x_m(i)) b_m(i) s_{l,m}(i)}{N \tilde{\lambda}_{l,m} \sqrt{\sum_{i=0}^{N-1} |s_{l,m}(i)|^2}} \tag{6}$$

where $s_{l,m}(i)$ is the spreading code of user l, $b_m(i) = 1/(\alpha |x_m(i)|)$, $v_{l,m}$ is the statistics of detection, and the term $\tilde{\lambda}_{l,m}$ will be explained in Sect. 3.3.

3.2 Weighted Collusion Attack

The collusion attack is a cost effective attack to remove hiding codes of attackers without severe degradation of multimedia quality. The weighted collusion attack with shifted spreading can be expressed mathematically as

$$\hat{y}_m(i) = \sum_{k=0}^{K-1} h_{k,m}(i) y_{k,m}(i - \Delta_{k,m}) + e(i), \quad i = 0, \ldots, N-1 \tag{7}$$

where $\hat{y}_m(i)$ is the colluded signal, $y_{k,m}(i)$ is the host signal embedded with user code k (called a colluder), $e(i)$ is the noise term, and $h_{k,m}(i)$ is the weight factor for colluder k and $\Delta_{k,m}$ is the shift amount.

Two types of collusion attacks have been considered in the literature: the average collusion attack [13] and the pre-colluded collusion attack [30]. The weights of all users are equal in the average collusion attack. In the pre-colluded collusion attack, users are divided into multiple groups, the average collusion attack is performed in each individual group, and finally, another average collusion attack is performed on output signals of all groups. These two attack types are actually special cases of the general weighted collusion attacks. In this work, we would like to identify colluders as well as their weights based on the received colluded copy. When the detector performs detection to determine colluders, it does not have the information of $h_{k,m}(i)$. Thus, weights $h_{k,m}(i)$ have to be estimated during the collusion attack detection process. Equation (7) can be viewed as a signal experiencing multipath fading

in a communication channel. Channel estimation and user detection techniques can help recover weights $h_{k,m}(i)$ and hiding codes from (7).

If all colluders in (7) come from the same user group (so that they share the same codeword), then we have a single user detection problem in wireless communication systems, which is relatively easy to solve. Once channel coefficients and the user codewords are determined, we can immediately identify participating colluders and their weights. However, if colluders in (7) come from different user groups, there will be MAI, which degrades the detection performance. We need to reduce MAI for better detection results. In wireless communications, MAI reduction can be achieved by using the multiuser detection (MUD) techniques in the receiver (or the detector) [23] or adopting MAI-free codes in the transmitter (or the embedder) [17, 18]. Here, we follow the latter approach, namely, designing an anti-collusion code that decouples different user groups as much as possible.

3.3 Channel Estimation and Equalization

The received signal for user l in a communication channel can be written in vector form as

$$\mathbf{r}_l = \mathbf{H}_l \mathbf{c}_l + \mathbf{e} \tag{8}$$

where $\mathbf{c}_l = (\ldots, c_l(-i), \ldots, c_l(-1), c_l(0), c_l(1), \ldots, c_l(i), \ldots)$ is the transmitted signal of the vector form, i is the time index, $\mathbf{e}$ is the noise vector and $\mathbf{H}_l$ is the channel impulse response matrix. Thus, weights in the weighted collusion attack play the same role as channel coefficients in a wireless communication system. Various channel estimation methods have been proposed in the literature e.g. [12, 22]. One popular technique for channel estimation is to employ pilot symbols [9]. If $\mathbf{e}$ is an i.i.d. white Gaussian and $\mathbf{H}_l$ is of full column rank, the maximum likelihood (ML) estimation of $\mathbf{c}_l$ with observation $\mathbf{r}_l$ becomes the simple least square (LS) estimation given by

$$\tilde{\mathbf{c}}_l = \left(\mathbf{H}_l^t \mathbf{H}_l\right)^{-1} \mathbf{H}_l^t \mathbf{r}_l. \tag{9}$$

By taking the discrete Fourier transform (DFT) of the channel response vector $\tilde{\mathbf{c}}_l$, we can get $\tilde{\lambda}_l$ for the channel equalization purpose.

4 Design of Anti-Collusion Hiding Codes

A general framework for hiding code construction is illustrated in Fig. 3, which is motivated by PMU-OFDM (precoded multiuser orthogonal frequency division multiaccess) or MC-CDMA (multicarrier code division multiaccess) communication systems [17, 18]. It consists of three modules: (1) message generation, (2) spreading by chip signals, and (3) multiplication by the identity or the IFT matrix.

In the first module, we assign each user a data sequence of sufficient length called the message sequence. For a system with L users, we use $\mathrm{U} = \{\mathbf{u}_l \in \mathbb{Z},\ l = 1, \ldots, L\}$ and $\mathrm{M} = \{\mathbf{m}_l \in \mathbb{Z},\ l = 1, \ldots, L\}$ to denote the set of user IDs and the set of user message sequences, respectively. To identify a user uniquely from the observed message sequence, we can define a one-to-one mapping between U and M. The message

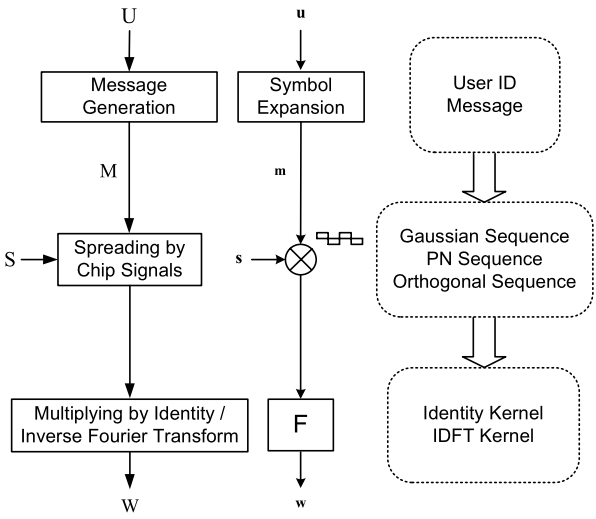

Fig. 3 Proposed framework for hiding code construction

sequence can be a one-bit or a multi-bit sequence [27]. If the message sequence is chosen to be of M bits there are $L = 2^M$ distinct ID numbers.

In the second module, we choose a code (or a chip signal) for a user to modulate each symbol in his/her message sequence. The spreading code may take a binary value (i.e., 1 and -1) or a q-bit value. The Gaussian and PN sequences have been used as the spreading code. The maximal length sequences, Gold sequences and Kasami sequences are examples of PN sequences. The PN sequence has a noise-like spectrum so that code detection can be efficiently achieved by de-spreading if there is no collusion attack. However, under the collusion attack, since PN sequences have weak cross-correlation, different codewords tend to interfere with each other in the de-spreading process. In contrast, orthogonal codes have zero cross-correlation between different codewords so that they are more robust against the collusion attack. The price to pay is that the spike of their self-correlation spectrum is not as sharp as that of PN sequences, which makes the code detection task more challenging. Examples of orthogonal codes include Hadamard–Walsh (HW) codes, Orthogonal Gold codes, Multirate OGold codes, and so on. For the collusion attack, orthogonal codes are a more attractive choice than the Gaussian and PN codes.

In the third module, we select one from the following two choices: multiplied by the identity matrix or the IFT matrix. For the former case, the system is analogous to CDMA, which is a single carrier communication scheme. For the latter case, the system is analogous to a multi-carrier communication scheme, which includes PMU-OFDM and MC-CDMA as special cases. It is well known that the multi-carrier communication system is more robust against frequency selective fading and has been widely used in broadband communication systems such as ADSL, Wi-Fi and WiMax.

If the identity matrix is used, the code design in the first and the second modules is conducted in the time domain. On the other hand, if the IFT matrix is used, the code design in the first and the second modules is actually conducted in the frequency domain. The flexibility of code design in the frequency domain allows the power of

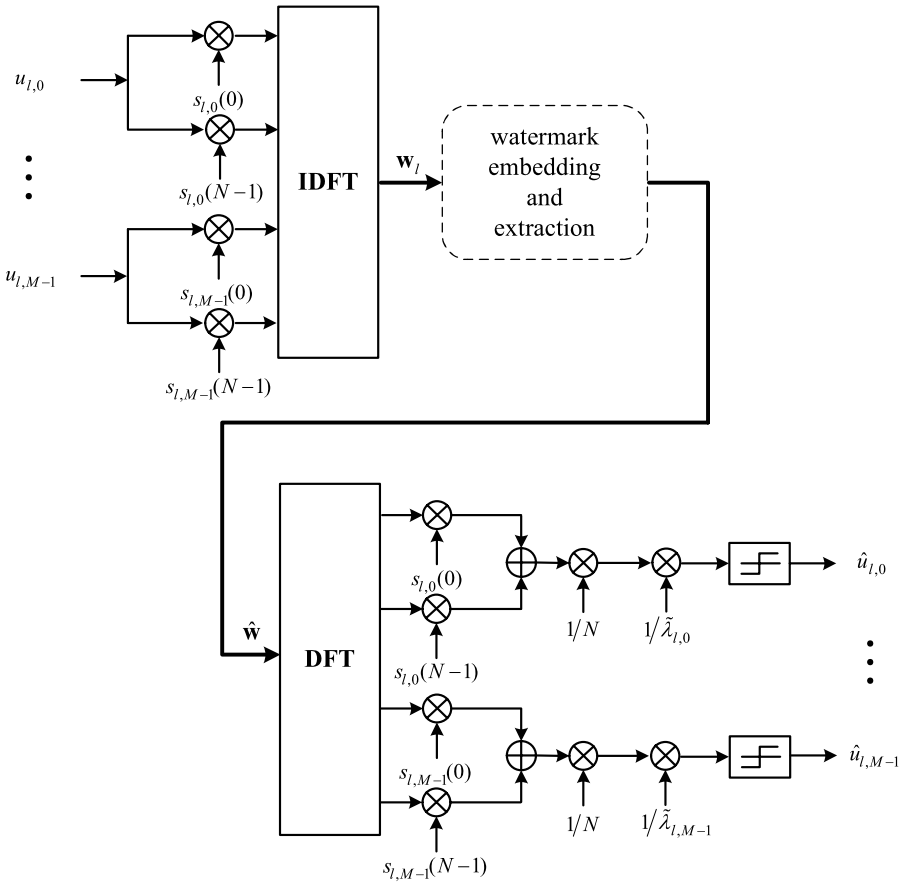

Fig. 4 The OSIFT code generation and detection and its interface with the fingerprint embedding and extraction system

resulting codes to be more uniformly distributed over a broader spectrum so that it is more robust against narrow-band interference [9].

As an example of anti-collusion codes, the OSIFT code is constructed by the following three steps.

- Step 1: Assign L different messages of the same length M to L users.
- Step 2: Each symbol in the message is spread by the orthogonal Hadamard–Walsh (HW) codes of length N in L users.
- Step 3: The IFT is applied.

The OSIFT code generation and detection system and its interface with the hiding code embedding and extraction system are shown in Fig. 4.

For the OSIFT code generation, we follow a procedure similar to that in [17, 18]. It is summarized below. Each bit of user l's ID vector $\mathbf{u}_{l,m}$ is repeated N times in the frequency domain. It is spread by an orthogonal code of length N with the following

property:

$$\sum_{i=0}^{N-1} s_{l,m}(i)s_{k,m}(i) = \begin{cases} N, & l = k, \\ 0, & l \neq k. \end{cases} \tag{10}$$

In particular, we choose the HW codes as the spreading codes due to their computational simplicity. The HW matrices can be recursively defined by

$$\mathbf{S}_N = \mathbf{S}_2 \otimes \mathbf{S}_{N/2} = \begin{pmatrix} \mathbf{S}_{N/2} & \mathbf{S}_{N/2} \\ \mathbf{S}_{N/2} & -\mathbf{S}_{N/2} \end{pmatrix} \tag{11}$$

where $N = 2^n$ ($n \geq 2$), $\otimes$ is the Kronecker product, and

$$\mathbf{S}_2 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}.$$

HW codes of length N are column vectors of HW matrices of dimension $N \times N$. HW codes only take 1 and -1 two values, which simplifies the spreading operation greatly. For a given shift, since the proposed OSIFT codes are formed by N orthogonal HW codes, they can be assigned to G groups of users.

5 Detection of Colluders

As discussed in Sect. 3.2, colluders in a weighted collusion attack may come from the same group sharing the same codeword or from different user groups with different codewords. We analyze them separately below.

If all colluders are from the same user group, their difference lies in shifted spreading. This corresponds to the single user detection problem with multipath fading channels. In detection, the first step is to identify the group indexing by checking the spreading code. The second step is to identify shifted amounts, which is similar to the synchronization process in communication. However, the problem is easier in our current context since the basic shift unit $\Delta_{p,m}$ which is a subset of $\Delta_{l,m}$, is known. Although we do not need to estimate the delay amount, the weights (or the multipath channel coefficients) of each colluder have to be estimated. This can be done using a channel estimation technique [18].

If colluders are from multiple user groups, there exists MAI between different groups. To see this, the output of the correlation detector in (6) can be written as

$$v_{l,m} = \frac{1}{\psi} \sum_{i=0}^{N-1} \big(\hat{y}_m(i) - x_m(i)\big) b_m(i) s_{l,m}(i) \tag{12}$$

where $\hat{y}_m(i)$ is the colluded output, $x_m(i)$ is the host signal, and

$$\psi = N \sqrt{\sum_{i=0}^{N-1} \big|s_{l,m}(i)\big|^2} \quad \text{and} \quad b_m(i) = \frac{1}{\alpha |x_m(i)|}.$$

Since

$$\hat{y}_m(i) - x_m(i) = \sum_{k=0}^{K-1} h_{k,m}(i)\hat{w}_{k,m}(i) + e(i),$$

we can simplify (12) as

$$\begin{aligned} v_{l,m} &= \frac{1}{\psi}\sum_{i=0}^{N-1}\left(\sum_{k=0}^{K-1} h_{k,m}(i)\hat{w}_{k,m}(i) + e(i)\right) b_m(i)s_{l,m}(i) \\ &= \frac{1}{\psi}\sum_{i=0}^{N-1}\sum_{l} h_{l,m}(i)\hat{w}_{l,m}(i)b_m(i)s_{l,m}(i) \\ &\quad + \frac{1}{\psi}\sum_{i=0}^{N-1}\sum_{k\neq l} h_{k,m}(i)\hat{w}_{k,m}(i)b_m(i)s_{l,m}(i) + \frac{1}{\psi}\sum_{i=0}^{N-1} e(i)b_m(i)s_{l,m}(i) \end{aligned} \tag{13}$$

where the first term denotes the detection of users from the same group, the second term is the detection of users from other groups, and the third term is the contribution from noise. The second term is called the MAI term in [17]. It was shown in [17] that some spreading codes are better than others in eliminating MAI. These superior spreading codes are called MAI-free codes. Once MAI is significantly reduced, (13) is reduced to detection of colluders from the same group.

6 Simulation Results

Based on the discussion in Sect. 3, we choose the following system parameters in our computer simulation. The code strength is $\alpha = 0.05$. We randomly select one user ID from 2^M user IDs generated by the i.i.d. random source in the simulation. Three spreading codes of length N are compared: bounded Gaussian (BG), pseudo-noise (PN) and orthogonal HW codes (OS). BG codes are known to have a better performance than that of unbounded Gaussian (UG) codes [31]. In addition, PN codes represent a typical hiding code that allows cross-correlations among codewords. By using IFT in the third stage, we obtain three additional schemes: BGIFT, PNIFT and OSIFT codes. The length of hiding codes is $T = MN$. We adopt 16-bit music signals sampled at 44.1 KHz as the host signal. A violin sound, whose energy is relatively small, is used in the simulation as shown in Fig. 5. We consider a system consisting of L users and select K colluders randomly while the remaining $L - K$ users are innocent.

Example 1 (Spreading without Shifts) The length of the spreading codes is chosen to be $N = 256$ and the length of user ID $u_{l,m}$ is chosen to be $M = 16$ or 32. In this example, we focus on the impact of different spreading codes. Thus, a different user ID is assigned to a different spreading code and the system has a total number of 256 users by direct spreading without shifts. For results with the equal-weight collusion attack, we refer to [4], which shows that OSIFT codes can detect all colluders. For

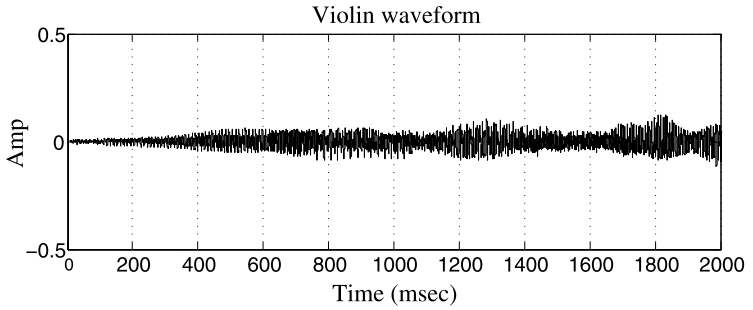

Fig. 5 The waveform of a violin sound used as a host audio signal in simulation

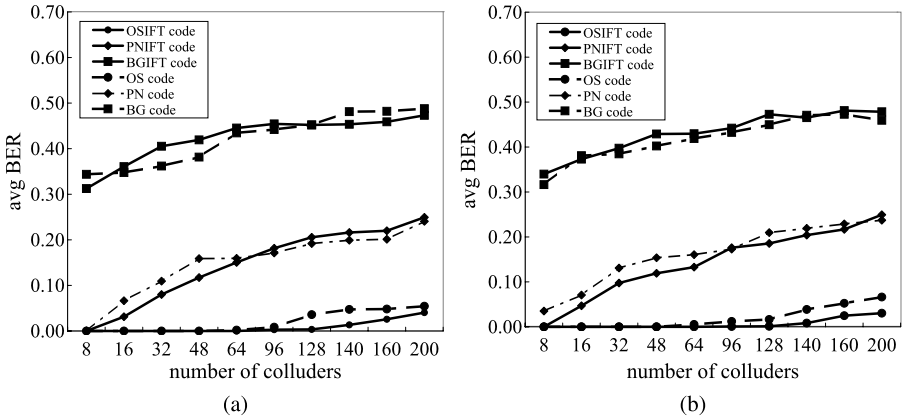

Fig. 6 The average BER results of BG, PN, OS, BGIFT, PNIFT, and OSIFT codes under the unequal-weight collusion attack: **a** the length of user ID $M = 16$, and **b** $M = 32$

the unequal-weight collusion attack, we consider the following case. The number of pre-colluded users, L_p, ranges from $K = 2$ to $K = 50$ to yield one colluded copy in the first stage. Then, this copy is colluded with other $K - L_p$ users that have not participated in the collusion process before. After getting the individual BER for each colluder, we get the average BER among all colluders. The average BERs of message length $M = 16$ and $M = 32$ with 6 hiding codes are compared in Fig. 6. As shown in this figure, the proposed OSIFT codes have the lowest average BERs in $M = 16$ and $M = 32$. The OS codes have fairly good performance, but they are worse than OSIFT codes. PN and PNIFT codes have similar performance. PNIFT codes are a little better than PN codes. BG and BGIFT codes give the worst performance. Next, we show detection rate R_d against the unequal-weight collusion attack with PN, OS, PNIFT, and OSIFT hiding codes in Fig. 7 under the constraints that false alarm rate $R_{fa} \leq 10^{-3}$ and BER $\leq \eta_k$. We do not show results for BG and BGIFT codes in this figure since their performance is too poor to compare. We observe similar BER performance among all 6 hiding codes for $M = 16$ and $M = 32$.

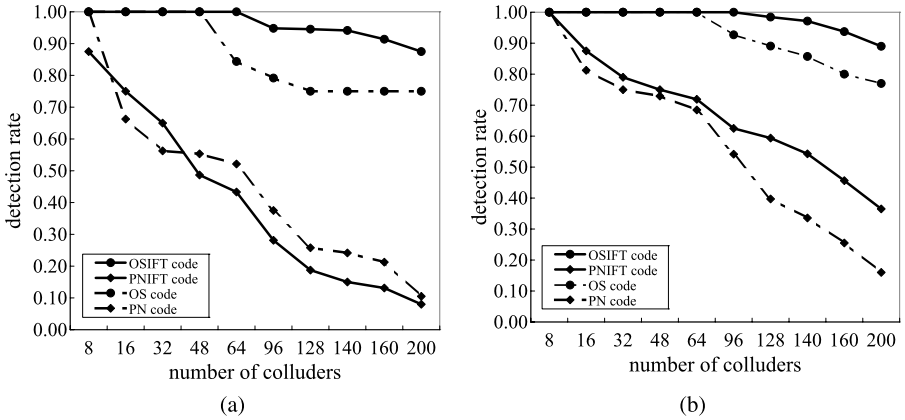

Fig. 7 The detection rate results of PN, OS, PNIFT, and OSIFT codes under the false alarm rate $R_{\text{fa}} < 10^{-3}$ and the unequal-weight collusion attack: **a** $M = 16$ and $\eta_k < 0.06$, and **b** $M = 32$ and $\eta_k < 0.12$

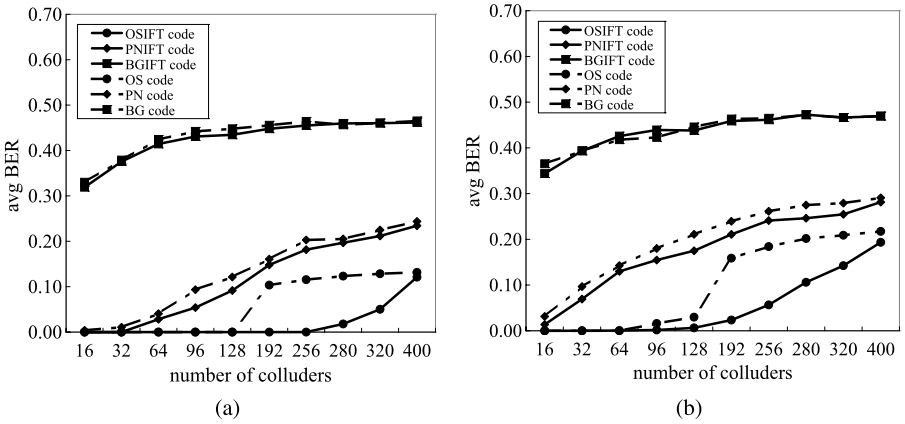

Fig. 8 The average BER results of BG, PN, OS, BGIFT, PNIFT, and OSIFT codes with shifts: **a** the equal-weight collusion attack, and **b** the unequal-weight collusion attack

Example 2 (Spreading with Shifts) By allowing shifted spreading with shift amounts equal to 0 and 64, we increase the user capacity in Example 1 from $N = 256$ to $L = PN = 2 \times 256 = 512$ and, we set $M = 32$ in this example. The average BER results of equal- and unequal-weighted collusion attacks with 6 hiding codes are shown in Fig. 8. The proposed OSIFT codes have the lowest average BER under shifted spreading. OSIFT codes can maintain zero BER up to $K = 256$ colluders. The OS codes have good performance when the number of colluders is less than 128. After that, we see a dramatic increase in the average BER, which is due to the interference of codewords from different groups due to the multipath fading effect. The same problem occurs in PN and PNIFT codes. BG and BGIFT codes give the worst performance. The colluder detection rates for equal- and unequal-weighted collusion attacks with PN, OS, PNIFT, and OSIFT hiding codes are shown in Fig. 9. The pro-

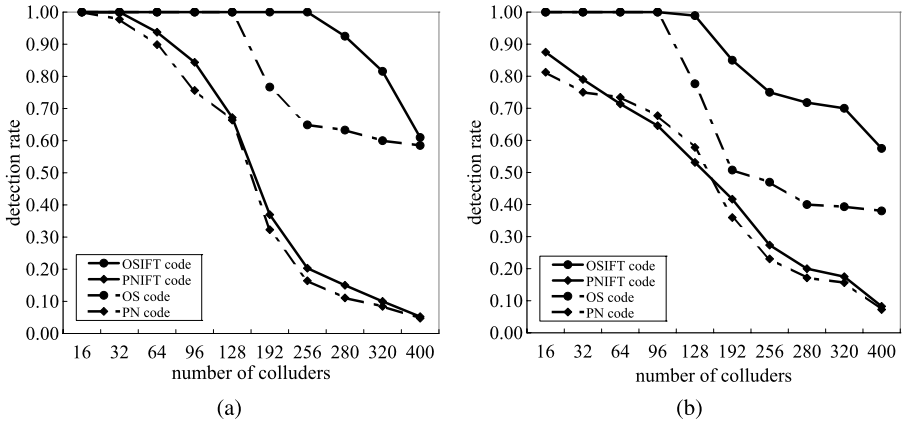

Fig. 9 The detection rate results of PN, OS, PNIFT, and OSIFT codes with shifts under the BER $\eta_k < 0.12$ and the false alarm rate $R_{\text{fa}} < 10^{-3}$: **a** the equal-weight collusion attack, and **b** the unequal-weight collusion attack

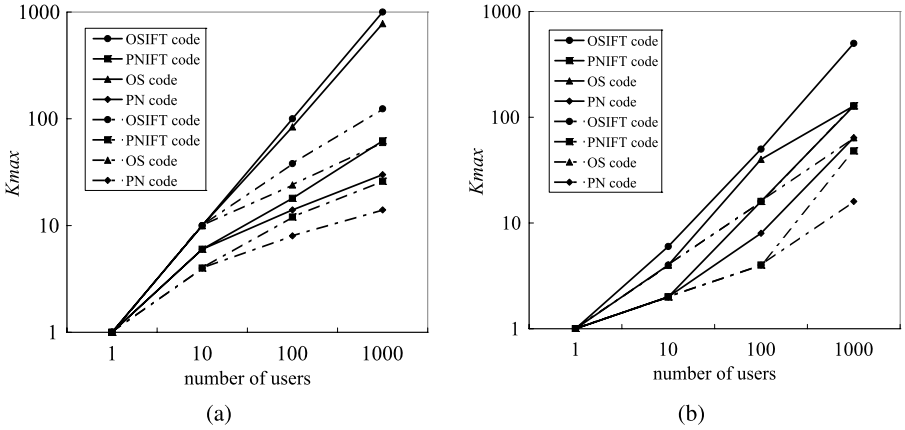

Fig. 10 Maximum number of colluders versus user number with PN, OS, PNIFT, and OSIFT codes against equal-weight collusion attacks (*continuous lines*) and unequal-weight collusion attacks (*dotted lines*) with $M = 32$ and colluder detection rate $R_{\text{d}} > 90\%$: **a** without shifted spreading, and **b** with shifted spreading and $P = 2$

posed OSIFT codes have the best performance. The OS codes behave poorly for the violin sound that has a small amplitude. The PNIFT codes perform better than the PN codes when shifted spreading is used.

Example 3 (Numbers of Colluders) Here, we examine the maximum number of colluders, $K_{\max}$, that can be accommodated in a system when the colluder detection rate is no less than 0.8. The length of message is $M = 32$. The other system parameters are the same as those given in Examples 1 and 2. The results for PN, OS, PNIFT, and OSIFT codes against four collusion attack scenarios are shown in Fig. 10. The performance of four hiding codes without and with shifted spreading is compared in

Fig. 10a and b, respectively. Both plots are given in the log-log scale. Again, the proposed OSIFT gives the best overall performance. We also observe that, when shifted spreading is adopted, we can double the user capacity, but at the price of a lowered maximum number of colluders.

7 Conclusion and Future Work

We proposed a systematic way to design a family of anti-collusion codes called OSIFT and showed its superior performance against the weighted collusion attack. A new methodology to increase the user capacity by shifting the spreading codeword was also investigated. The weighted collusion attack was analyzed by considering its analogy to multi-user wireless communication systems. We observed an interesting trade-off between the total number of users and the maximum number of colluders to be accurately detected (see Example 3 in the simulation section). That is, by introducing codeword shifts, we can increase the user capacity, but the colluder detection rate is lowered. It appears to be possible to develop more advanced detection schemes to improve the colluder detection rate. This is under our current investigation.

References

1. Bassia, P., Pitas, I., Nikolaidis, N.: Robust audio watermarking in the time domain. IEEE Transactions on Multimedia **3**, 232–240 (2001)
2. Boneh, D., Shaw, J.: Collusion-secure fingerprinting for digital data. IEEE Transactions on Information Theory **44**, 1897–1905 (1998)
3. Cha, B.-H., Kuo, C.-C.J.: Design of collusion-free codes based on MAI-free principle. In: Proc. IEEE Int'l Conf. Intelligent Information Hiding and Multimedia Signal Processing, pp. 639–642. Pasadena, CA (2006)
4. Cha, B.-H., Kuo, C.-C.J.: Design of collusion-free hiding codes using MAI-free principle. In: Proc. IEEE Int'l Conf. Acoustics, Speech, and Signal Processing, pp. 145–148. Honolulu, HI (2007)
5. Cha, B.-H., Kuo, C.-C.J.: Design of multiuser collusion-free hiding codes with delayed embedding. In: Proc. IEEE Int'l Conf. Intelligent Information Hiding and Multimedia Signal Processing. Kaohsiung, Taiwan (2007)
6. Chu, H., Qiao, L., Nahrstedt, K.: A secure multicast protocol with copyright protection. In: Proc. ACM SIGCOMM, pp. 42–60. Pittsburgh, PA (2002)
7. Cox, I.J., Kilian, J., Leighton, F.T., Shamoon, T.: Secure spread spectrum watermarking for multimedia. IEEE Transactions on Image Processing **6**, 1673–1687 (1997)
8. Giannakis, G.B., Serpedin, E.: Single-user channel estimation and equalization. IEEE Transactions on Signal Processing **45**, 67–81 (1997)
9. Hanzo, L., Munster, M., Choi, B.J., Keller, T.: OFDM and MC-CDMA for broadband multi-user communications, WLANs and broadcasting. Wiley, New York (2004)
10. He, S., Wu, M.: Joint coding and embedding techniques for multimedia fingerprinting. IEEE Transactions on Information Forensics and Security **1**, 231–247 (2006)
11. Kirovski, D., Mihak, M.K.: Bounded Gaussian fingerprints and the gradient collusion attack. In: Proc. IEEE Int'l Conf. Acoustics, Speech, and Signal Processing, pp. 1037–1040. Philadelphia, PA (2005)
12. Pun, M.O., Morelli, M., Kuo, C.-C.J.: Maximum-likelihood synchronization and channel estimation for OFDMA uplink transmissions. IEEE Transactions on Communications **54**, 726–736 (2006)
13. Stone, H.S.: Analysis of attacks on image watermarks with randomized coefficients. Tech. Rep. 96-045, NEC Res. Inst. Tech., Princeton, NJ (1996)
14. Su, J.K., Eggers, J.J., Girod, B.: Capacity of digital watermarks subjected to an optimal collusion attack. In: Proc. European Signal Processing Conference. Tampere, Finland (2000)

15. Swaminathan, A., He, S., Wu, M.: Exploring QIM based anti-collusion fingerprinting for multimedia. In: Proc. SPIE Conf. Security, Watermarking, and Steganography. San Jose, CA (2006)
16. Trappe, W., Wu, M., Wang, Z.J., Liu, K.J.R.: Anti-collusion fingerprinting for multimedia. IEEE Transactions on Signal Processing **51**, 1069–1087 (2003)
17. Tsai, S.H., Lin, Y.P., Kuo, C.-C.J.: A precoded multiuser OFDM (PMU-OFDM) transceiver for time asynchronous systems. In: Proc. IEEE GLOBECOM, pp. 2214–2218. St. Louis, MO (2005)
18. Tsai, S.H., Lin, Y.P., Kuo, C.-C.J.: MAI-free MC-CDMA based on Hadamard–Walsh codes. IEEE Transactions on Signal Processing **54**, 3166–3179 (2006)
19. Tse, D.N.C., Hanly, S.V.: Linear multiuser receivers: effective interference, effective bandwidth and user capacity. IEEE Transactions on Information Theory **45**, 641–657 (1999)
20. Tse, D.N.C., Viswanath, P.: Fundamentals of wireless communication. Cambridge University Press, Cambridge (2005)
21. Tugnait, J.K., Tong, L., Ding, Z.: Single-user channel estimation and equalization. IEEE Signal Processing Magazine **17**, 16–28 (2000)
22. Tureli, U., Kivanc, D., Liu, H.: Channel estimation for multicarrier CDMA. In: Proc. IEEE Int'l Conf. Acoustics, Speech, and Signal Processing, pp. 2909–2912. Istanbul, Turkey (2000)
23. Verdu, S.: Multiuser detection. Cambridge University Press, Cambridge (1998)
24. Viswanath, P., Anantharam, V., Tse, D.N.C.: Optimal sequences, power control, and user capacity of synchronous CDMA systems with linear MMSE multiuser receivers. IEEE Transactions on Information Theory **45**, 1968–1983 (1999)
25. Wang, Z.J., Wu, M., Trappe, W., Liu, K.J.R.: Group-oriented fingerprinting for multimedia forensics. EURASIP Journal on Applied Signal Processing **14**, 1242–2162 (2004)
26. Wang, Z.J., Wu, M., Zhao, H.V., Liu, K.J.R.: Anti-collusion forensics of multimedia fingerprinting using orthogonal modulation. IEEE Transactions on Image Processing **14**, 804–821 (2005)
27. Wu, M., Liu, B.: Data hiding in image and video: part I—fundamental issues and solutions. IEEE Transactions on Image Processing **12**, 685–695 (2003)
28. Zhao, H.V., Liu, K.J.R.: Behavior forensics for scalable multiuser collusion: fairness versus effectiveness. IEEE Transactions on Information Forensics and Security **1**, 311–329 (2006)
29. Zhao, H.V., Liu, K.J.R.: Fingerprint multicast in secure video streaming. IEEE Transactions on Information Forensics and Security **15**, 12–29 (2006)
30. Zhao, H.V., Liu, K.J.R.: Tritor-within-traitor behavior forensics: strategy and risk minimization. IEEE Transactions on Information Forensics and Security **1**, 440–456 (2006)
31. Zhao, H.V., Wu, M., Wang, Z.J., Liu, K.J.R.: Forensic analysis of nonlinear collusion attacks for multimedia fingerprinting. IEEE Transactions on Image Processing **14**, 646–661 (2005)