PAPER Special Section on Cryptography and Information Security

Fingerprinting Codes for Multimedia Data against Averaging Attack*

Hideki YAGI[†], Toshiyasu MATSUSHIMA[††], *Members*, *and* Shigeichi HIRASAWA[†††], *Fellow*

SUMMARY Code construction for digital fingerprinting, which is a copyright protection technique for multimedia, is considered. Digital fingerprinting should deter collusion attacks, where several fingerprinted copies of the same content are mixed to disturb their fingerprints. In this paper, we consider the averaging attack, which is known to be effective for multimedia fingerprinting with the spread spectrum technique. We propose new methods for constructing fingerprinting codes to increase the coding rate of conventional fingerprinting codes, while they guarantee to identify the same number of colluders. Due to the new fingerprinting codes, the system can deal with a larger number of users to supply digital contents.

key words: *fingerprinting codes, averaging attack, finite geometry, low-density matrix, multimedia*

1. Introduction

With high advances of information technologies, copyright protection of digital contents has become an important problem. As one of solutions, digital fingerprinting has attracted a great deal of attention for protection of content distribution systems. The digital fingerprinting embeds a user's ID, called a fingerprint, into an original content with a watermarking technique. After embedding fingerprints, fingerprinted contents are distributed to respective users.

Digital fingerprinting requires robustness against collusion attacks, in which more than one illicit user colludes to take illegal actions to their distributed contents. Some of well-known collusion attacks are the interleaving attack [1], [4], [9] and the averaging attack [4], [11]–[13], [16], [17]. The interleaving attack is generally considered for fingerprinting of generic digital data. On the other hand, the averaging attack is assumed for multimedia fingerprinting, and it is conducted by the arithmetic averaging operation among all fingerprinted copies of colluders. On the whole, multimedia fingerprinting employs a spread spectrum technique to embed users' fingerprint, where pseudo-random sequences spread a binary value over a wide domain of a host data. Although most of multimedia fingerprinting does not adopt any coding technique, W. Trappe et al. have devised collusion-secure fingerprinting codes against the averaging attack. Their codes are constructed based on incident matrices of block designs [11], which is equivalent to regular low-density (LD) matrices without cycles of length four [5]. The fingerprinting codes devised by Trappe et al. are called anti-collusion (AC) codes [11], [12]. Subsequently, Kang et al. have proposed a method for improving the efficiency of AC codes based on group-divisible design [16], [17]. Although these codes can guarantee to capture colluders whose size is not greater than a pre-determined value, unfortunately, their coding rates rapidly decrease with increasing the code length.

In this paper, we propose methods for improving fingerprinting codes devised by Trappe et al. or Yang et al. [15] by increasing its coding rate, while their resilience is maintained. We first derive some general condition which relaxes restriction of the conventional AC codes, providing flexible design of fingerprinting codes. We then propose two explicit construction methods by using finite field arithmetics based on the derived condition. The proposed method presented first utilizes structure of finite geometries, which allow us to algebraically realize AC codes satisfying the derived condition. The latter method is based on a construction technique of structured LD matrices. The latter construction method is the primary contribution of this paper, and the first one is utilized as its component. The proposed methods also increase the coding rate of AC codes in [16], [17]. Consequently, we can realize content distribution systems which can deal with greater number of users.

This paper is organized as follows: In Sect. 2, we describe a model of fingerprinting system considered in this paper. In Sect. 3, we briefly review conventional AC codes for multimedia. In Sect. 4, we derive some general condition, which relaxes restriction of the conventional AC codes. In Sect. 5, based on the derived condition, we propose a method for improving the conventional AC codes by using finite geometries. In Sect. 6, we propose another method, which combines structured LD matrices and other code matrices of AC codes. In Sect. 7, we compare the effectiveness of the proposed methods with the method in [16], [17]. In Sect. 8, some concluding remarks are stated.

Manuscript received March 27, 2008.
Manuscript revised July 23, 2008.

[†] The author was with Media Network Center, Waseda University, Tokyo, 169-8555 Japan. He is now with Education and Research Center for Frontier Science, the University of Electro-Communications, Chofu-shi, 182-8585 Japan.

[††] The author is with the School of Fundamental Science and Eng., Waseda University, Tokyo, 169-8555 Japan.

[†††] The author is with the School of Creative Science and Eng., Waseda University, Tokyo, 169-8555 Japan.

[*] Some content of this paper was presented at 2007 IEEE Int. Conf. on System, Man, and Cybernetics (SMC2007), Montreal, Canada, Oct. 2007.

DOI: 10.1587/transfun.E92.A.207

2. Model of Fingerprinting System

2.1 Digital Fingerprinting for Multimedia

We describe a model of multimedia fingerprinting considered in this paper. The model in this paper follows [11], [12].

When distributing a digital content to users, a codeword corresponding to each user is embedded into an original content by a watermarking technique. The codeword allocated for each user is called the user's fingerprint, and the distributed contents are called fingerprinted contents. Some illicit users may collude to use their fingerprinted content for an illegal purpose. They may attempt to disturb their fingerprints so that their fingerprints are not revealed from an illegally utilized content. This action is called a collusion attack. A detector of colluders estimates colluders' fingerprints from a disturbed fingerprint.

Let $\Gamma := \{1, 2, \dots, |\Gamma|\}$ be a set of users of a digital content. We denote a codeword to a user $j \in \Gamma$ by $\boldsymbol{b}_j = (b_{1j}, b_{2j}, \dots, b_{Nj})^{\mathrm{T}} \in \{0, 1\}^N$, where T denotes the transposition. In stead of directly embedding $\boldsymbol{b}_j$ into a host content, we create a fingerprint watermark $\boldsymbol{w}_j$ by a spread spectrum technique beforehand. We arrange N mutually orthogonal bases $\boldsymbol{u}_1, \boldsymbol{u}_2, \dots, \boldsymbol{u}_N \in \mathcal{R}^N$ of an equal energy λ, which form an N dimensional real vector space. We denote the set of these N orthogonal bases by $\mathcal{U} := \left\{\boldsymbol{u}_i \middle| \|\boldsymbol{u}_i\|^2 = \lambda, i = 1, 2, \dots, N\right\}$. Then each $\boldsymbol{w}_j \in \mathcal{R}^N, j = 1, 2, \dots, |\Gamma|$, is created by a fingerprint $\boldsymbol{b}_j$ and $\mathcal{U}$ as

$$\boldsymbol{w}_j := \sum_{i=1}^{N} (2b_{ij} - 1)\boldsymbol{u}_i, \qquad (1)$$

where the summation expresses the addition of real numbers.

Next, regarding a distributed content to users as a host signal, the created fingerprint watermark is embedded into it. Denoting embedded parts of a host signal by a vector $\boldsymbol{x} \in \mathcal{R}^N$, the distributed content to a user $j \in \Gamma$ is[†], $\boldsymbol{y}_j := \boldsymbol{x} + \boldsymbol{w}_j$.

Figure 1 illustrates the embedding process of finger prints. Since the fingerprint is embedded by using watermarking and spread spectrum techniques, any users cannot perceive their own fingerprint $\boldsymbol{w}_j$ (and hence $\boldsymbol{b}_j$) from the fingerprinted content $\boldsymbol{y}_j$ without the knowledge of $\boldsymbol{x}$ and $\mathcal{U}$. Therefore illicit users may collude to disturb their fingerprints by creating an illegal content from their distributed contents.

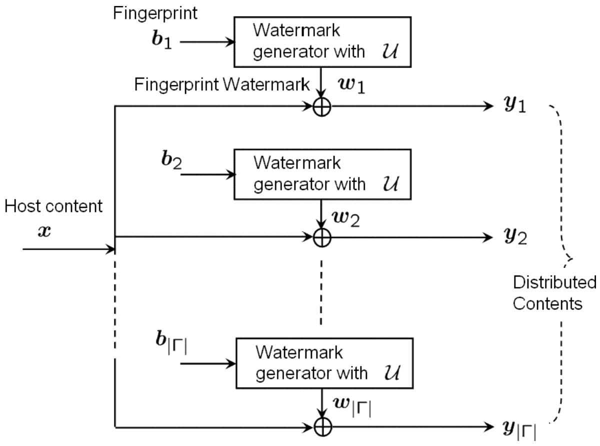

Fig. 1 Illustration of fingerprinting model.

2.2 Assumed Collusion Attack

We consider a set of colluders with size $h \geq 1$, denoted by $\mathcal{S} \subseteq \Gamma$. We assume $\mathcal{S} = \{j_1, j_2, \dots, j_h\}$ such that $1 \leq j_1 < j_2 < \cdots < j_h \leq |\Gamma|$. Assume that a colluder set $\mathcal{S}$ attacks to create an illegal content, denoted by $\boldsymbol{y} \in \mathcal{R}^N$. A detector of colluders estimates $\mathcal{S}$ from an attacked content $\boldsymbol{y}$. In this paper, we assume the following collusion attack.

Definition 1: Assume that an attacked content by a colluder set $\mathcal{S}$ is expressed as

$$\boldsymbol{y} := \frac{1}{h}\sum_{j \in \mathcal{S}} \boldsymbol{y}_j = \boldsymbol{x} + \frac{1}{h}\sum_{j \in \mathcal{S}}\sum_{i=1}^{N}(2b_{ij} - 1)\boldsymbol{u}_i, \qquad (2)$$

where the second equality is obtained from Eq. (1). This attack is called the averaging attack, which is known to be effective for multimedia fingerprinting[††], [4], [11]–[13]. □

3. Fingerprinting Codes against Averaging Attack

3.1 Conventional Anti-Collusion Codes

Trappe et al. have proposed a code construction method of anti-collusion (AC) codes by utilizing block design [11]. We here introduce the definition of AC codes. For any subsets $\mathcal{I} \subseteq \Gamma$, let $Q(\mathcal{I}) := \{i | b_{ij} = 0, \forall j \in \mathcal{I}\}$. In other words, $Q(\mathcal{I})$ represents the set of symbol positions where any fingerprints in $\mathcal{I}$ equally take 0-component.

Definition 2: Assume that the detector of colluders have the complete knowledge about a host signal $\boldsymbol{x}$, the set of orthogonal sequences $\mathcal{U}$. For some positive integer ℓ, a binary code $\mathcal{B} = \{\boldsymbol{b}_j\}$ is referred to as an ℓ-resilient AC code, iff any subsets $\mathcal{I} \subseteq \Gamma$ of size $|\mathcal{I}| \leq \ell$ has unique $Q(\mathcal{I})$. The parameter ℓ is called the resilience of AC codes. □

Hereafter we assume that a host signal $\boldsymbol{x}$, the set of orthogonal sequences $\mathcal{U}$, and an AC code are known to the detector.

[†] In this paper, for simplicity, the fingerprinted content is defined in this manner. More precisely, each $\boldsymbol{w}_j$ is multiplied by some value called Just-Difference Noticeable (JDN) coefficient [7], before it is added to the host signal.

[††] For simplicity, although we only discuss the case of the averaging attack, the argument here can hold for the logical OR attack [15]. The fingerprinting codes devised by J. Yang et al. [15] are also based on block design, and the proposed method in this paper can also improve their performance.

Trappe et al. have proposed some construction of AC codes, which satisfy some condition.

Definition 3: Consider the following condition on AC codes:

(i) any codeword has a constant Hamming weight k;

(ii) two distinct codewords have at most one "1-component" in the same position.

We call this condition the conventional condition.

□

In the next section, the conventional condition will be relaxed to enlarge a class of AC codes.

Resilience of the AC code based on the conventional condition can be guaranteed by the following lemma.

Lemma 1 ([11]): Let $B = [b_{ij}]$ be some binary matrix of N rows which satisfies the conventional condition. If a j-th column vector $\boldsymbol{b}_j = (b_{1j}, b_{2j}, \ldots, b_{Nj})^{\mathrm{T}}$ is a j-th user's fingerprint, a set of column vectors $\mathcal{B} = \{\boldsymbol{b}_j\}$ becomes a $(k-1)$-resilient AC code. i.e., if $|\mathcal{S}| \le k-1$, $Q(\mathcal{S})$ can be uniquely identified. □

We here show a principle for detecting a set of colluders with a $(k-1)$-resilient AC code. We here use an example for simplicity. Suppose $k = 3$ and $\mathcal{S} = \{i, j\}$ whose fingerprints are given by $\boldsymbol{b}_i = (1,1,0,1,0,0,0)^{\mathrm{T}}, \boldsymbol{b}_j = (1,0,1,0,0,1,0)^{\mathrm{T}}$. Then from Eq. (1),

$$\boldsymbol{w}_i = \boldsymbol{u}_1 + \boldsymbol{u}_2 - \boldsymbol{u}_3 + \boldsymbol{u}_4 - \boldsymbol{u}_5 - \boldsymbol{u}_6 - \boldsymbol{u}_7, \tag{3}$$

$$\boldsymbol{w}_j = \boldsymbol{u}_1 - \boldsymbol{u}_2 + \boldsymbol{u}_3 - \boldsymbol{u}_4 - \boldsymbol{u}_5 + \boldsymbol{u}_6 - \boldsymbol{u}_7. \tag{4}$$

As a result of the averaging attack, an illegal content $\boldsymbol{y} = \boldsymbol{x} + (\boldsymbol{w}_i + \boldsymbol{w}_j)/2$ is produced. With a known $\boldsymbol{x}$, we can subtract $\boldsymbol{x}$ from $\boldsymbol{y}$, which gives $\boldsymbol{y} - \boldsymbol{x} = \boldsymbol{u}_1 - \boldsymbol{u}_5 - \boldsymbol{u}_7$ from Eqs. (2)–(4). The detected sequence $\boldsymbol{y} - \boldsymbol{x}$ has a tuple of coefficients $(1,0,0,0,-1,0,-1)$, which can be calculated by taking the inner product between $\boldsymbol{y} - \boldsymbol{x}$ and each orthogonal basis $\boldsymbol{u}_i, i = 1, 2, \ldots, N$. Thus the position set of (-1)-components in this tuple is $\{5, 7\}$, and it coincides with the set $Q(\mathcal{S})$. From Definition 2, since a $(k-1)$-resilient AC code uniquely determines $Q(\mathcal{S})$ for any $\mathcal{S}$ such that $|\mathcal{S}| \le k-1$, the position set of (-1)-components calculated from $\boldsymbol{y} - \boldsymbol{x}$ reveals that the user i and j participate in the collusion. Even for a general case, any $(k-1)$-resilient AC code can identify colluders in this way [11], [12].

3.2 Class of AC Codes Based on Finite Geometries

A subclass of AC codes by Trappe et al. can be algebraically constructed by using finite geometries. We briefly describe two kinds of finite geometries, namely, a Euclidean geometry and a projective geometry. Refer to [5], [10] for more detail.

For a prime p and two positive integers $m \ge 2$ and $s \ge 1$, an m-dimensional Euclidean geometry $\mathrm{EG}(m, p^s)$ over a Galois field $\mathrm{GF}(p^s)$ consists of points, lines, and hyperplanes. Any points in $\mathrm{EG}(m, p^s)$ are p^{ms} m-dimensional vectors over $\mathrm{GF}(p^s)$, and they constitute an m-dimensional vector space V over $\mathrm{GF}(p^s)$. For an integer μ such that $0 \le \mu \le m$, μ-dimensional hyperplanes (generally, called a μ-flat) is a μ-dimensional subspace of V and its cosets. Any μ-flat contains exactly $p^{\mu s}$ points. Points and lines correspond to 0-flats and 1-flats, respectively.

For a given $\mu < m$, let $\boldsymbol{a}_0, \boldsymbol{a}_1, \ldots, \boldsymbol{a}_\mu$ be $\mu + 1$ linear independent points in $\mathrm{EG}(m, p^s)$. Then using μ elements $\beta_1, \beta_2, \ldots, \beta_\mu$ of $\mathrm{GF}(p^s)$, $p^{\mu s}$ points expressed as

$$\boldsymbol{a}_0 + \beta_1 \boldsymbol{a}_1 + \beta_2 \boldsymbol{a}_2 + \cdots + \beta_\mu \boldsymbol{a}_\mu \tag{5}$$

constitute a μ-flat.

It can be easily verified that any pair of two μ-flats, (F_1, F_2), has at most one $(\mu-1)$-flat in common, which implies F_1 and F_2 have at most $p^{(\mu-1)s}$ points in common. In a Euclidean geometry $\mathrm{EG}(m, p^s)$, there are

$$f_{\mathrm{EG}}^{(m)}(\mu) := p^{(m-\mu)s} \prod_{i=1}^{\mu} \frac{p^{(m-i+1)s} - 1}{p^{(\mu-i+1)s} - 1} \tag{6}$$

μ-flats in total.

Denoting an m-dimensional projective geometry over a Galois field $\mathrm{GF}(p^s)$ by $\mathrm{PG}(m, p^s)$, $\mathrm{PG}(m, p^s)$ contains $(p^{(m+1)s} - 1)/(p^s - 1)$ points. We can also consider μ-flats in a projective geometry $\mathrm{PG}(m, p^s)$, and each μ-flat consists of $(p^{(\mu+1)s} - 1)/(p^s - 1)$ points. In $\mathrm{PG}(m, p^s)$, there are

$$f_{\mathrm{PG}}^{(m)}(\mu) := \prod_{i=0}^{\mu} \frac{p^{(m-i+1)s} - 1}{p^{(\mu-i+1)s} - 1} \tag{7}$$

μ-flats in total. Any pair of two μ-flats, (F_1, F_2), has at most one $(\mu-1)$-flat in common, which implies F_1 and F_2 have at most $(p^{\mu s} - 1)/(p^s - 1)$ points in common.

For simplicity, we sometimes use expression $\mathrm{FG}(m, p^s)$ to express either a Euclidean geometry $\mathrm{EG}(m, p^s)$ or a projective geometry $\mathrm{PG}(m, p^s)$. In a similar manner, $f_{\mathrm{FG}}^{(m)}(\mu)$ expresses either $f_{\mathrm{EG}}^{(m)}(\mu)$ or $f_{\mathrm{PG}}^{(m)}(\mu)$. We define $N_0 := f_{\mathrm{FG}}^{(m)}(0)$, which expresses the number of points in $\mathrm{FG}(m, p^s)$.

We number N_0 points in a given $\mathrm{FG}(m, p^s)$ from 1 to N_0 and $f_{\mathrm{FG}}^{(m)}(\mu)$ μ-flats in $\mathrm{FG}(m, p^s)$ from 1 to $f_{\mathrm{FG}}^{(m)}(\mu)$. Suppose an $N_0 \times f_{\mathrm{FG}}^{(m)}(\mu)$ matrix $B_\mu = [b_{ij}]$ and allocate the rows and the columns of B_μ to points and μ-flats in $\mathrm{FG}(m, p^s)$, respectively. A component b_{ij} in a matrix B_μ takes $b_{ij} = 1$ if a point i is contained in a μ-flat j, and takes $b_{ij} = 0$ otherwise. This matrix B_μ is referred to as the incident matrix of μ-flats over points in $\mathrm{FG}(m, p^s)$.

Arranging each column vector of the incident matrix B_1 of 1-flats (lines) over points in $\mathrm{FG}(m, p^s)$ as a codeword, an AC code of Trappe et al. can be obtained. They utilize the two properties; (i) any 1-flat in $\mathrm{FG}(m, p^s)$ has a constant number of points, (ii) any pair of two 1-flats has at most one point in common. That is, AC codes obtained from $\mathrm{FG}(m, p^s)$ satisfy the conventional condition. Any AC code constructed from $\mathrm{EG}(m, p^s)$ becomes a $(p^s - 1)$-resilient AC code from Lemma 1. By using $\mathrm{PG}(m, p^s)$ to construct an AC code, the code becomes a p^s-resilient AC code. We denote this AC code by $\mathcal{B}_1$ and refer to as conventional AC codes based on finite geometries.

We here mention parameters of an AC code $\mathscr{B}_1$. The code length is $N = N_0$, which equals to the number of points in FG(m, p^s). The number of codewords (the number of accommodated users) is $f^{(m)}_{\mathrm{FG}}(1)$, which expresses the number of 1-flats in FG(m, p^s). Thus the coding rate r_1 is given by $r_1 = (\log_2 f^{(m)}_{\mathrm{FG}}(1))/N_0$ [5]. For given resilience and code length, as the number of codewords increases, the system can provide services to more users. Therefore we need to increase the number of codewords as large as possible for given resilience and code length.

For $\mathscr{B}_1$ based on EG(m, s), the number of codewords is $f^{(m)}_{\mathrm{EG}}(1) = p^{(m-1)s}(p^{ms}-1)/(p^s-1) = O(p^{2(m-1)s})$. Therefore, taking notice that $N_0 = p^{ms}$ and $p^s = N_0^{\frac{1}{m}}$, we have

$$f^{(m)}_{\mathrm{EG}}(1) = O(N_0^2 p^{-2s}) = O(N_0^{2-\frac{2}{m}}). \tag{8}$$

Equation (8) indicates that the number of codewords approximately increases in the order of square of code length. This quantity is insufficient if we want to accommodate a large number of users. Increasing the number of codewords in larger order of code length is desired. Note that, for the case with PG(m, s), the number of codewords is also the order of square of code length but it is omitted here.

3.3 Class of AC code Based on Quasi-Cyclic LD Matrices

Another class of ℓ-resilient AC codes by Trappe et al. can be algebraically constructed based on regular low-density (LD) matrices without cycles of length four [5].

Let α be a primitive element over a Galois field GF(p^{ms}) and we denote the zero element over this field by $0 = \alpha^{-\infty}$. Then any non-zero element can be expressed as α^i for $i = 0, 1, \ldots, p^{ms} - 2$. For any element α^i, let $\boldsymbol{z}_i = (z_{i,-\infty}, z_{i,0}, z_{i,1}, \ldots, z_{i,p^{ms}-2})$ be a p^{ms}-tuple over GF(2) such that it takes $z_{i,j} = 1$ if $i = j$, and $z_{i,j} = 0$ otherwise. The vector $\boldsymbol{z}_i$ is called the location vector of α^i. Arrange p^{ms} cyclic-shifted versions of the location vector $\boldsymbol{z}_i$ to form a $p^{ms} \times p^{ms}$ circulant matrix, where the first row is $\boldsymbol{z}_i$ itself and a j-th row is the right-shifted version of $\boldsymbol{z}_i$ by $j-1$ times. We denote this matrix of α^i by $\pi^i(I)$, where I corresponds to the $p^{ms} \times p^{ms}$circulant matrix of $0 = \alpha^{-\infty}$ (i.e., the identity matrix) and π expresses the cyclic permutation.

For two integers $\gamma \geq 1$ and $\rho \geq 1$, a quasi-cyclic (QC) LD matrix defined over a Galois field GF(p^{ms}) is of the form

$$M_0 := \begin{bmatrix} \pi^{a_{1,1}}(I) & \pi^{a_{1,2}}(I) & \cdots & \pi^{a_{1,\rho}}(I) \\ \pi^{a_{2,1}}(I) & \pi^{a_{2,2}}(I) & \cdots & \pi^{a_{2,\rho}}(I) \\ \vdots & \vdots & \ddots & \vdots \\ \pi^{a_{\gamma,1}}(I) & \pi^{a_{\gamma,2}}(I) & \cdots & \pi^{a_{\gamma,\rho}}(I) \end{bmatrix}, \tag{9}$$

where $\pi^{a_{i,j}}(I)$ for $i = 1, 2, \ldots, \gamma$ and $j = 1, 2, \ldots, \rho$ is a $p^{ms} \times p^{ms}$ circulant matrix of $\alpha^{a_{i,j}}$. Thus the size of M_0 is $\gamma p^{ms} \times \rho p^{ms}$. Define

$$\boldsymbol{a}_j := \begin{bmatrix} a_{1,j} \\ a_{2,j} \\ \vdots \\ a_{\gamma,j} \end{bmatrix}, \quad \text{for } j = 1, 2, \ldots, \rho. \tag{10}$$

If any pair of $\boldsymbol{a}_i$ and $\boldsymbol{a}_j\,(i, j = 1, 2, \ldots, \rho)$ satisfies $d_H(\boldsymbol{a}_i, \boldsymbol{a}_j) \geq \gamma - 1$, then any pair of two columns of the matrix M_0 has at most one 1-component in common, and vice versa [2], [3], [14]. Such matrix M_0 is called a (γ, ρ) QC-LD matrix.

The QC-LD matrices are utilized for constructing error-correcting codes [5], and there have been proposed many types of QC-LD matrices. Some examples of QC-LD matrices considered in this paper are constructed based on the structure of the Reed-Solomon code [2], [6] or based on the structure of the Array codes [3], [14]. In [2], it has been shown that (γ, ρ) QC-LD matrices for $2 \leq \gamma \leq p^{ms} - 1, 1 \leq \rho \leq p^{ms}$ can be constructed for a given GF(p^{ms}).

Since a (γ, ρ) QC-LD matrix satisfies (i) each column weight is γ, and (ii) any pair of two columns has at most one 1-component in common, any AC code whose codewords are arranged from column vectors of M_0 is a $(\gamma-1)$-resilient AC code. We denote this AC code by $\mathcal{M}_0$ and call conventional AC codes based on QC-LD matrices.

We here mention the code parameters of conventional AC codes based on QC-LD matrices. The code length, the number of codewords, and the resilience are $N = \gamma p^{ms}$, ρp^{ms}, and $\ell = p^{ms} - 1$, respectively. To the code length N, the number of codewords is expressed as $O(\frac{N^2}{\gamma^2})$ if we choose the value of ρ as $\rho = p^{ms}$ to maximize the number of codewords. Similar to the conventional AC codes based on finite geometries, increasing the number of codewords in larger order of code length is desired.

4. Relaxation of Conditions on AC Codes

In this section, we relax the conventional condition of AC codes in Definition 3, which provides a larger class of AC codes[†]. Our purpose in this paper is to increase the coding rate of AC codes, and enlarging the class of AC codes may unable us to find AC codes with larger coding rates.

4.1 Derivation of Condition

The relaxation proposed in this paper consists of two main ideas; (i) increasing the Hamming weight of codewords, and (ii) allowing more than one 1-component in common between two codewords. For a real number v, let $\lceil v \rceil$ be the minimum integer not less than v.

Lemma 2: Assume that a binary matrix satisfies:

(i) the Hamming weight of each column is at least k;

[†] This extension is not limited to the AC codes using finite geometries or QC-LD matrices but can apply to any AC codes in [11], [15].

(ii) any pair of distinct two column vectors has at most t 1-components in common.

Then, an AC code obtained from this matrix is a $(\lceil k/t \rceil - 1)$-resilient AC code.

(*Proof*) The proof is an extension of that of Theorem 1 in [11] for the case $t \geq 1$.

We denote the row position set (support set) in which a j-th column vector $\boldsymbol{b}_j$ has 1-components by $\mathcal{A}_j$. i.e., $\mathcal{A}_j := \{i | b_{ij} = 1\}$. By using $\mathcal{A}_j, j \in \mathcal{S}$, the set $Q(\mathcal{S})$ is expressed as $Q(\mathcal{S}) = \bigcap_{j \in \mathcal{S}} \overline{\mathcal{A}_j}$, where $\overline{\mathcal{A}_j}$ is the complement of $\mathcal{A}_j$. Suppose $|\mathcal{S}| \leq \lceil k/t \rceil - 1$. If $\bigcap_{j \in \mathcal{S}} \overline{\mathcal{A}_j} \neq \bigcap_{i \in \mathcal{I}} \overline{\mathcal{A}_i}$ for arbitrary subset $\mathcal{I} \subseteq \Gamma$ such that $\mathcal{I} \neq \mathcal{S}$ and $|\mathcal{I}| \leq \lceil k/t \rceil - 1$, a code $\mathcal{B} = \{\boldsymbol{b}_j\}$ is an ℓ-resilient AC code. Furthermore, from De Morgan's low, this condition is equivalent to

$$\bigcup_{j \in \mathcal{S}} \mathcal{A}_j \neq \bigcup_{i \in \mathcal{I}} \mathcal{A}_i, \quad \forall \mathcal{I} \neq \mathcal{S},\ s.t. |\mathcal{I}| \leq \lceil k/t \rceil - 1. \tag{11}$$

Thus it suffices to show Eq. (11) for proving the lemma.

We suppose temporarily that a set $\mathcal{I} \neq \mathcal{S}$ with size $|\mathcal{I}| \leq \lceil k/t \rceil - 1$ satisfies $\bigcup_{j \in \mathcal{S}} \mathcal{A}_j = \bigcup_{i \in \mathcal{I}} \mathcal{A}_i$. Then $\mathcal{A}_j \subseteq \bigcup_{i \in \mathcal{I}} \mathcal{A}_i$ for any $j \in \mathcal{S}$. From the assumption of the lemma, for some $\mathcal{A}_{j^o}, j^o \in \mathcal{S} \setminus (\mathcal{S} \cap \mathcal{I})$, any $\mathcal{A}_i, i \in \mathcal{I}$, has at most t elements in common with $\mathcal{A}_{j^o}$. Therefore it requires $|\mathcal{I}| \geq \lceil k/t \rceil$ to satisfy $\mathcal{A}_{j^o} \subseteq \bigcup_{i \in \mathcal{I}} \mathcal{A}_i$ from the assumption $|\mathcal{A}_{j^o}| \geq k$. Thus it contradicts the assumption $|\mathcal{I}| \leq \lceil k/t \rceil - 1$, and Eq. (11) holds. □

The AC codes assumed in Lemma 2 are reduced to the AC codes in [11] if their codewords have a constant Hamming weight k and $t = 1$. Therefore this extension provides a large class of ℓ-resilient AC codes. More importantly, it is possible to increase the number of codewords by varying the parameters k and t for given N and ℓ. Later, we will give two explicit construction methods by utilizing the relaxed condition.

4.2 Distortion Given by AC Codes with Relaxed Condition

We here mention distortion to an original content given by AC codes with the relaxation of the conditions.

Distortion to a digital content $\boldsymbol{x}$ by a fingerprint $\boldsymbol{b}_j$ can be measured by $||\boldsymbol{w}_j||^2$, where $|| \cdot ||^2$ denotes the square of norm, since $\boldsymbol{y}_j = \boldsymbol{x} + \boldsymbol{w}_j$. Then the average distortion of $\boldsymbol{x}$ by an AC code $\mathcal{B} = \{\boldsymbol{b}_j\}$ is expressed as $E[||\boldsymbol{w}_j||^2]$, where $E[\cdot]$ denotes the expectation by $\mathcal{B}$. It follows from Eq. (1) that the average distortion to the content $\boldsymbol{x}$ can be calculated as

$$\begin{aligned} E\Big[||\boldsymbol{w}_j||^2\Big] &= E\left[\left\| \sum_{i=1}^{N} (2b_{ij} - 1)\boldsymbol{u}_i \right\|^2\right] \\ &= E\Big[\sum_{i=1}^{N} (2b_{ij} - 1)^2 ||\boldsymbol{u}_i||^2\Big], \end{aligned} \tag{12}$$

where the second equality can be obtained by the property of the orthogonal sequences $\{\boldsymbol{u}_i\}$. Taking notice that $(2b_{ij} - 1)^2 = 1$ for any $b_{ij} \in \{0, 1\}$, we have

$$E\Big[||\boldsymbol{w}_j||^2\Big] = E\Big[\sum_{i=1}^{N} ||\boldsymbol{u}_i||^2\Big] = \sum_{i=1}^{N} ||\boldsymbol{u}_i||^2 = N\lambda, \tag{13}$$

where λ represents the equal power of orthogonal bases $\{\boldsymbol{u}_i\}$. It can be seen from Eq. (13) that the distortion takes a constant value $N\lambda$ regardless of distribution of symbols of $\{\boldsymbol{b}_j\}$.

Therefore the distortion to an original content given by the AC codes of Lemma 2 is equal to that in [11] for given code length, even if the Hamming weight of each codeword increases.

5. Improvement of AC Codes Using Finite Geometries

We propose an explicit construction method by using finite geometries based on the relaxed condition. The proposed method increases the coding rate of the conventional AC codes with keeping code length and resilience.

5.1 AC Codes Based on Finite Geometries

As explained in Sect. 3.2, when constructing ℓ-resilient AC codes of [11] by using finite geometries FG(m, p^s), we consider relationship between points and lines (1-flats) in FG(m, p^s). In the new code construction, the relationship between points and μ-flats ($\mu \geq 1$) in FG(m, p^s) is utilized.

Definition 4: For $\mu \geq 1$, let B_μ be the incident matrix of μ-flats over points in a finite geometry FG(m, p^s), and we denote its j-th column vector by $\boldsymbol{b}_j$. Allocating $\boldsymbol{b}_j$ to a j-th user's fingerprint, the obtained code $\mathcal{B}_\mu := \{\boldsymbol{b}_j\}$ is called a μ-th order FG-AC code. In particular, an AC code $\mathcal{B}_\mu$ constructed from a Euclidean geometry and a projective geometry are called a μ-th order EG-AC code and a μ-th order PG-AC code, respectively. □

We then obtain the following theorem.

Theorem 1: For some EG(m, p^s), the μ-th order EG-AC code $\mathcal{B}_\mu$ is a $(p^s - 1)$-resilient AC code. For some PG(m, p^s), the μ-th order PG-AC code $\mathcal{B}_\mu$ is a p^s-resilient AC code.

(*Proof*) As explained in Sect. 3.2, any pair of two μ-flats, F_1 and F_2, in FG(m, p^s) has at most one $(\mu - 1)$-flat in common. It can be seen that the case of EG-AC codes corresponds to that of substituting $k = p^{\mu s}, t = p^{(\mu-1)s}$ in Lemma 2. Therefore the resilience is $\lceil k/t \rceil - 1 = p^s - 1$. As for PG codes, we have the relationship of $k = (p^{(\mu+1)s} - 1)/(p^s - 1), t = (p^{\mu s} - 1)/(p^s - 1)$ in Lemma 2, which leads to the resilience of $\lceil k/t \rceil - 1 = p^s$. □

From Theorem 1, it can be found that the resilience of a μ-th order FG-AC code $\mathcal{B}_\mu$ is independent of the order μ.

We here mention parameters of μ-th order FG-AC codes. For a given FG(m, p^s), we can construct $m - 1$ μ-th order FG-AC codes $\mathcal{B}_\mu$ for $\mu = 1, 2, \ldots, m - 1$ with the same resilience. From the properties of the incident matrix of μ-flats over points, the code lengths of these FG-AC codes $\mathcal{B}_\mu$ are equally $N = N_0$ and the numbers of codewords are $f_{\mathrm{FG}}^{(m)}(\mu)$. The coding rate, denoted by r_μ, is given by $r_\mu = (\log_2 f_{\mathrm{FG}}^{(m)}(\mu))/N_0$. i.e., parameter of $\mathcal{B}_\mu$ depending on the order μ is only the number of codewords, which determines the FG-AC code $\mathcal{B}_{\mu^*}$ with the maximal size for a given FG(m, p^s). Therefore we call such an order μ^* the maximal order of the FG-AC codes for a given FG(m, p^s).

We show the following theorems about the maximal order of FG-AC codes for a given FG(m, p^s). For a real number v, let $\lfloor v \rfloor$ be the maximum integer not greater than v.

Theorem 2: For a given EG(m, p^s), the maximal order μ^* of the EG-AC codes is given by

$$\mu^* = \left\lfloor \frac{m}{2} \right\rfloor. \tag{14}$$

(*Proof*) If $m = 2$, we can easily check that $\mu^* = 1$ takes the maximum value in Eq. (6). When $m \geq 3$, we show that the function $f_{\mathrm{EG}}^{(m)}(\mu)$ is convex upward. We define $g_{\mathrm{EG}}^{(m)}(\mu) := f_{\mathrm{EG}}^{(m)}(\mu) - f_{\mathrm{EG}}^{(m)}(\mu - 1)$ for $2 \leq \mu \leq m$ and we investigate its sign. We can easily verify that

$$\begin{aligned} g_{\mathrm{EG}}^{(m)}(\mu) &= p^{(m-\mu)s} \prod_{i=1}^{\mu-1} \frac{p^{(m-i+1)s}-1}{p^{(\mu-i)s}-1} \left\{ \frac{p^{(m-\mu+1)s}-1}{p^{\mu s}-1} - p^s \right\} \\ &= \frac{p^{(m-\mu)s}}{p^{\mu s}-1} \prod_{i=1}^{\mu-1} \frac{p^{(m-i+1)s}-1}{p^{(\mu-i)s}-1} \\ &\quad \cdot \left\{ p^{(m-\mu+1)s} - p^{(\mu+1)s} + p^s - 1 \right\} \end{aligned} \tag{15}$$

from Eq. (6). In Eq. (15), the first and second terms of r.h.s. are strictly positive for $2 \leq \mu \leq m$. i.e.,

$$\frac{p^{(m-\mu)s}}{p^{\mu s}-1} \prod_{i=1}^{\mu-1} \frac{p^{(m-i+1)s}-1}{p^{(\mu-i)s}-1} > 0. \tag{16}$$

Thus only the last term

$$h_{\mathrm{EG}}^{(m)}(\mu) := p^{(m-\mu+1)s} - p^{(\mu+1)s} + p^s - 1 \tag{17}$$

varies its sign, and the sign of $h_{\mathrm{EG}}^{(m)}(\mu)$ coincides with that of $g_{\mathrm{EG}}^{(m)}(\mu)$.

If m is odd, $h_{\mathrm{EG}}^{(m)}(\mu) > 0$ for $2 \leq \mu \leq \frac{m-1}{2}$, and $h_{\mathrm{EG}}^{(m)}(\mu) < 0$ for $\frac{m+1}{2} \leq \mu \leq m$. This fact implies that the function $f_{\mathrm{EG}}^{(m)}(\mu)$ is convex upward and it takes the maximum value for $\mu = \frac{m-1}{2}$. If m is even, $h_{\mathrm{EG}}^{(m)}(\mu) > 0$ for $2 \leq \mu \leq \frac{m}{2}$, and $h_{\mathrm{EG}}^{(m)}(\mu) < 0$ for $\frac{m}{2} + 1 \leq \mu \leq m$. Then the function $f_{\mathrm{EG}}^{(m)}(\mu)$ is also convex upward and it takes the maximum value for $\mu = \frac{m}{2}$. Thus Eq. (14) holds. □

Theorem 3: For a given PG(m, p^s), the maximal order μ^* of the PG-AC codes is given by

$$\mu^* = \begin{cases} \frac{m-1}{2}, & \text{for odd } m, \\ \frac{m}{2} - 1 \text{ and } \frac{m}{2}, & \text{for even } m. \end{cases} \tag{18}$$

(*Proof*) We can prove the theorem in a similar manner to Theorem 2. If $m = 2$, we can easily check that both $\mu^* = 0$ and $\mu^* = 1$ take the maximum value in Eq. (7). When $m \geq 3$, defining $g_{\mathrm{PG}}^{(m)}(\mu) := f_{\mathrm{PG}}^{(m)}(\mu) - f_{\mathrm{PG}}^{(m)}(\mu - 1)$ for $2 \leq \mu \leq m$, it follows from Eq. (7) that

$$\begin{aligned} g_{\mathrm{PG}}^{(m)}(\mu) &= \frac{1}{p^{(\mu+1)s}-1} \prod_{i=0}^{\mu-1} \frac{p^{(m-i+1)s}-1}{p^{(\mu-i)s}-1} \\ &\quad \cdot \left\{ p^{(m-\mu+1)s} - p^{(\mu+1)s} \right\}. \end{aligned} \tag{19}$$

Again, in Eq. (19), the first and second terms of r.h.s. are strictly positive for $2 \leq \mu \leq m$ and only the last term $h_{\mathrm{PG}}^{(m)}(\mu) := p^{(m-\mu+1)s} - p^{(\mu+1)s}$ varies its sign.

If m is odd, $h_{\mathrm{PG}}^{(m)}(\mu) > 0$ for $2 \leq \mu \leq \frac{m-1}{2}$, and $h_{\mathrm{PG}}^{(m)}(\mu) < 0$ for $\frac{m+1}{2} \leq \mu \leq m$. Then the function $f_{\mathrm{PG}}^{(m)}(\mu)$ is convex upward and it takes the maximum value for $\mu = \frac{m-1}{2}$. If m is even, $h_{\mathrm{PG}}^{(m)}(\mu) \geq 0$ for $2 \leq \mu \leq \frac{m}{2}$, where equality holds iff $m = \frac{m}{2}$, and $h_{\mathrm{PG}}^{(m)}(\mu) < 0$ for $\frac{m}{2} + 1 \leq \mu \leq m$. Then the function $f_{\mathrm{EG}}^{(m)}(\mu)$ is also convex upward and it takes the maximum value for $\mu = \frac{m}{2} - 1$ and $\mu = \frac{m}{2}$. Thus Eq. (18) holds. □

It follows from Theorems 2 and 3 that there always exists a better FG-AC code than the AC code $\mathcal{B}_1$ of Trappe et al. when $m > 3$.

We compare the numbers of codewords for $\mathcal{B}_1$ and $\mathcal{B}_{\mu^*}$ as a function of the code length $N = N_0$. From Eq. (8), the number of codewords of $\mathcal{B}_1$ is $f_{\mathrm{EG}}^{(m)}(1) = O(N^{2-\frac{2}{m}})$ where $N = N_0$. Similarly, for general $1 \leq \mu \leq m$, we can easily verify $f_{\mathrm{EG}}^{(m)}(\mu) = O(p^{(m-\mu)(\mu+1)s})$ from Eq. (6). Define a function $\delta(m)$ of m as

$$\delta(m) := \begin{cases} \frac{1}{m}, & \text{for odd } m, \\ 0, & \text{for even } m. \end{cases} \tag{20}$$

From Theorem 2, $\mu^* = \lfloor \frac{m}{2} \rfloor$ and by substitution, we have

$$\begin{aligned} f_{\mathrm{EG}}^{(m)}(\mu^*) &= O\!\left(p^{\frac{1}{4}ms(m+2+\delta(m))}\right) && (21) \\ &= O\!\left(N^{\frac{1}{4}(m+2+\delta(m))}\right). && (22) \end{aligned}$$

It follows from Eqs. (8) and (22) that the number of codewords of $\mathcal{B}_{\mu^*}$ rapidly increases to the code length N as m increases, while the number of codewords in $\mathcal{B}_1$ does not increases more than the order of N^2.

5.2 Examples of FG-AC Codes with the Maximal Order

For a given EG(m, p^s), we show some examples of EG-AC codes $\mathcal{B}_{\mu^*}$ with the maximal order in Table 1. For $m > 3$, we display codes with the resilience $(p^s - 1)$ greater than one in the increasing order of their resilience. In the table, the columns of "$\log_2 f_{\mathrm{EG}}(1)$" and "$\log_2 f_{\mathrm{EG}}(\mu^*)$" express the logarithm of the number of codewords of $\mathcal{B}_1$ (the Trappe's AC code) and that of the EG-AC code $\mathcal{B}_{\mu^*}$ with the maximal order.

Table 1 Examples of EG-AC codes with maximal order.

ℓ	(m, p^s)	N_0	$\log_2 f_{\mathrm{EG}}(1)$	$\log_2 f_{\mathrm{EG}}(\mu^*)$	μ^*
2	$(4, 3^1)$	81	10.08	10.19	2
	$(5, 3^1)$	243	13.26	15.00	2
	$(6, 3^1)$	729	16.43	19.80	3
	$(7, 3^1)$	2187	19.60	26.16	3
	$(8, 3^1)$	6561	22.77	32.52	4
3	$(4, 2^2)$	256	12.41	12.48	2
	$(5, 2^2)$	1024	16.41	18.50	2
	$(6, 2^2)$	4096	20.41	24.52	3
4	$(4, 5^1)$	625	14.25	14.30	2
	$(5, 5^1)$	3125	18.90	21.28	2

Table 2 Examples of PG-AC codes with maximal order.

ℓ	(m, p^s)	N_0	$\log_2 f_{\mathrm{PG}}(1)$	$\log_2 f_{\mathrm{PG}}(\mu^*)$	μ^*
3	$(4, 3^1)$	121	10.24	10.24	1, 2
	$(5, 3^1)$	364	13.43	15.05	2
	$(6, 3^1)$	1093	16.60	19.82	2, 3
	$(7, 3^1)$	3280	19.77	26.18	3
	$(8, 3^1)$	9841	22.94	32.52	3, 4
4	$(4, 2^2)$	341	12.50	12.50	1, 2
	$(5, 2^2)$	1365	16.51	18.52	2
	$(6, 2^2)$	5461	20.51	24.53	2, 3
5	$(4, 5^1)$	781	14.31	14.31	1, 2
	$(5, 5^1)$	3906	18.96	21.29	2

It follows from the property of the function $f_{\mathrm{EG}}^{(m)}(\mu)$ that the number of codewords of $\mathcal{B}_{\mu^*}$ becomes larger than that of $\mathcal{B}_1$ as the dimension m of EG(m, p^s) increases. In particular, the number of codewords of $\mathcal{B}_{\mu^*}$ is 2^2 times larger than that of $\mathcal{B}_1$ when $m \geq 5$, $2^{6.5}$ times larger for EG$(7, 3)$, and 2^{10} times larger for EG$(8, 3)$.

We also show examples of PG-AC codes $\mathcal{B}_{\mu^*}$ with the maximal order in Table 2 for a given PG(m, p^s). The PG-AC codes with the maximal order behave similar to the EG-AC codes. There exist two maximal orders when m is even.

6. Improvement of AC Code Based on Quasi-Cyclic LD Matrix

In this section, we show how to improve such AC codes by using QC-LD matrices.

6.1 AC Codes Based on Quasi-Cyclic LD Matrix

We propose a method for increasing the number of codewords of AC codes based on QC-LD matrices while maintaining the resilience.

The proposed method combines a QC-LD matrix M_0 given by Eq. (9) and another code matrix of an AC code. Let n and f be integers such that $n \geq p^{ms}$ and $f \geq 2$. Let B be an $n \times f$ code matrix of some ℓ-resilient AC code. We replace an (i, j)-th circulant matrix $\pi^{a_{i,j}}(I)$ $(i = 1, 2, \ldots, \gamma, j = 1, 2, \ldots, \rho)$ of M_0 with a matrix $\pi^{a_{i,j}}(B)$, which can be obtained to right-shift the matrix B $a_{i,j}$ times. Let the resultant matrix be denoted by M'. The following theorem shows that this code matrix gives some AC code.

Theorem 4: Let $\mathcal{M}'$ be a set of all column vectors of M'. A code $\mathcal{M}'$ has (i) the code length γn, (ii) the number of codewords ρf, and (iii) the resilience $\min\{\gamma - 1, \ell\}$.

(*Proof*) Since both (i) the code length and (ii) the number of codewords are obvious, we here mention (iii) the resilience.

We partition γn rows of the matrix M' by n rows into γ groups, and we call a ν-th group (the $(\nu n + 1)$-th row to $(\nu + 1)n$-th row) the ν-th row section. Each column vector $\boldsymbol{m}_j \in \{0, 1\}^{\gamma n}$ of M' can be expressed with γ vectors $\boldsymbol{m}_{\nu,j} \in \{0, 1\}^n$ as

$$\boldsymbol{m}_j := \begin{bmatrix} \boldsymbol{m}_{1,j} \\ \boldsymbol{m}_{2,j} \\ \vdots \\ \boldsymbol{m}_{\gamma,j} \end{bmatrix}. \tag{23}$$

We denote the support set of $\boldsymbol{m}_{\nu,j}$ $(\nu = 1, 2, \ldots, \gamma)$ by $\mathcal{A}_{\nu,j}$.
(i) the case of $\gamma - 1 \leq \ell$:

Consider a colluder set $\mathcal{S}$ of size $|\mathcal{S}| \leq \gamma - 1$. We suppose there exists a set of users, $\mathcal{I}$, of size $|\mathcal{I}| \leq \gamma - 1$ satisfying $\boldsymbol{Q}(\mathcal{S}) = \boldsymbol{Q}(\mathcal{I})$. As in the proof of Lemma 2, this equation can be re-written as[†]

$$\bigcup_{j \in \mathcal{S}} \bigcup_{\nu \in [1,\gamma]} \mathcal{A}_{\nu,j} = \bigcup_{i \in \mathcal{I}} \bigcup_{\nu \in [1,\gamma]} \mathcal{A}_{\nu,i}. \tag{24}$$

In this case, it requires $\mathcal{A}_{\nu,j^o} \subseteq \bigcup_{i \in \mathcal{I}} \mathcal{A}_{\nu,i}$, $\nu = 1, 2, \ldots, \gamma$, for any column vectors $\boldsymbol{m}_{j^o}$, $\forall j^o \in \mathcal{S} \backslash (\mathcal{S} \cap \mathcal{I})$, of M'.

For any $i \in \mathcal{I}$, there are no two row sections (say, ν_1-th and ν_2-th row sections) such that $\boldsymbol{m}_{\nu_1,j^o} = \boldsymbol{m}_{\nu_1,i}$ and $\boldsymbol{m}_{\nu_2,j^o} = \boldsymbol{m}_{\nu_2,i}$. Otherwise, these equations imply that there exists a pair of two column vectors having more than one 1-component in common in the original QC-LD matrix M_0. Since $|\mathcal{I}| \leq \gamma - 1$, there exists at least one row section (say, ν^*-th row section) in which $\boldsymbol{m}_{\nu^*,j^o} \neq \boldsymbol{m}_{\nu^*,i}$ for any $i \in \mathcal{I}$. Since column vectors $\boldsymbol{m}_{\nu^*,j^o}$ and $m_{\nu^*,i}$ for $i \in \mathcal{I}$ are originally codewords of an ℓ-resilient AC code $\mathcal{B}$ and $\gamma - 1 \leq \ell$, we have $\mathcal{A}_{\nu^*,j^o} \nsubseteq \bigcup_{i \in \mathcal{I}} \mathcal{A}_{\nu^*,i}$. Thus it contradicts to Eq. (24), and the code is a $(\gamma - 1)$-resilient AC code when $\gamma - 1 \leq \ell$.
(ii) The case of $\gamma - 1 > \ell$:

Taking similar steps to the case (i), it can be shown that the code should be a ℓ-resilient AC code. Thus the theorem holds. □

Theorem 4 indicates that we can increase the number of codewords of $\mathcal{M}_0$ by using some $\mathcal{B}$ such that $\gamma - 1 \leq \ell$ and $f > p^{ms}$ with keeping the resilience. In particular, if $n = p^{ms}$, the code length of a resultant AC code $\mathcal{M}'$ remains equal as well as the resilience.

By using a Euclidean geometry, we can consider a similar technique in Sect. 5. For a Euclidean geometry EG(m, p^s), let B_μ be a $p^{ms} \times f_{\mathrm{EG}}^{(m)}(\mu)$ incident matrix of the

[†] For integers $i < j$, $[i, j]$ denotes the set of integers from i to j.

μ-flats over points. We replace an (i, j)-th circulant matrix $\pi^{a_{i,j}}(I)$ $(i = 1, 2, \dots, \gamma, j = 1, 2, \dots, \rho)$ of M_0 with a matrix $\pi^{a_{i,j}}(B_\mu)$, which can be obtained to right-shift the matrix B_μ $a_{i,j}$ times. We denote the resultant $\gamma p^{ms} \times \rho f^{(m)}_{\mathrm{EG}}(\mu)$ matrix by M_μ. Let $\mathcal{M}_\mu$ be an AC code whose codewords are column vectors of M_μ, and then we have the following corollary.

Corollary 1: The AC code $\mathcal{M}_\mu$ for a given EG(m, p^s) has (i) the code length γp^{ms}, (ii) the number of codewords $\rho f^{(m)}_{\mathrm{EG}}(\mu)$, and (iii) the resilience $\ell = \min\{\gamma - 1, p^s - 1\}$. □

It follows from Corollary 1 that we can improve AC codes which are constructed based on a QC-LD matrix by using a Euclidean geometry EG(m, p^s). Note that the code length of $\mathcal{M}_0$ is not altered in this case. In particular, if $\gamma - 1 < p^s$, the AC codes $\mathcal{M}_\mu$ are always more efficient than the conventional code $\mathcal{M}_0$ with keeping the code length and resilient. About the obtained AC codes based-on QC-LD matrices, we can assert similar effectiveness mentioned in Sect. 5.2.

We state some relationship between the conventional AC code $\mathcal{M}_0$ and the proposed AC code $\mathcal{M}_\mu$ for a given EG(m, p^s). Denoting the incident matrix of 0-flats (namely, points) over points in EG(m, p^s) by B_0, we can see $I = B_0$. Therefore substituting $\mu = 0$ in the matrix M_μ, we can obtain the matrix M_0. This fact implies that the conventional AC code $\mathcal{M}_0$ based on QC-LD matrix is an instance of the AC codes $\mathcal{M}_\mu$ with $\mu = 0$.

Using projective geometries PG(m, p^s), we can obtain a similar result as in Corollary 1. We only show a result here.

Corollary 2: Assume that we allocate each column vector of the incident matrix of μ-flats over points in PG(m, p^s) to a user's codeword. This AC code has (i) the code length $\gamma(p^{(m+1)s} - 1)/(p^s - 1)$, (ii) the number of codewords $\rho f^{(m)}_{\mathrm{PG}}(\mu)$, and (iii) the resilience

$$\ell = \begin{cases} \gamma - 1, & \text{if } \gamma - 1 < (p^{(\mu+1)s} - 1)/(p^{\mu s} - 1), \\ p^s, & \text{otherwies.} \end{cases}$$

Then, if $\gamma - 1 < (p^{(\mu+1)s} - 1)/(p^{\mu s} - 1)$, this code is also a $(\gamma - 1)$-resilient AC code. □

As for the case of projective geometry, the code length increases from an original QC-LD matrix.

6.2 Example of FG-AC Codes Based on QC-LD Matrices

As illustration of the obtained EG-AC codes based on QC-LD matrices, we show some examples by using QC-LD matrices constructed from Reed-Solomon code [2] in Table 3. For a given GF(p^{ms}), (γ, ρ) QC-LD matrices with $2 \leq \gamma \leq p^{ms} - 1$ and $1 \leq \rho \leq p^{ms}$ can be constructed. We choose some γ and fix values of ρ as $\rho = p^{ms}$ to make the size of AC codes as large as possible. We show the logarithm of the code size for M_0 in the column of "$\log_2 \rho f_{\mathrm{EG}}(0)$" and for M_{μ^*} in that of "$\log_2 \rho f_{\mathrm{EG}}(\mu^*)$."

Table 3 Examples of EG-AC codes based on QC-LD matrices; we set $\gamma = p^s$ and $\rho = p^{ms}$ for given EG(m, p^s).

ℓ	(m, p^s)	N	$\log_2 \rho f_{\mathrm{EG}}(0)$	$\log_2 \rho f_{\mathrm{EG}}(\mu^*)$
2	$(3, 3^1)$	81	9.51	11.63
	$(4, 3^1)$	243	12.68	16.53
	$(5, 3^1)$	729	15.85	22.92
	$(6, 3^1)$	2187	19.02	29.31
	$(7, 3^1)$	6561	22.19	37.25
3	$(3, 2^2)$	256	12.00	14.39
	$(4, 2^2)$	1024	16.00	20.48
	$(5, 2^2)$	4096	20.00	28.50
4	$(3, 5^1)$	625	13.93	16.56
	$(4, 5^1)$	3125	18.58	23.59

Table 4 Examples of PG-AC codes based on QC-LD matrices; we set $\gamma = p^s$ and $\rho = p^{ms}$ for given PG(m, p^s).

ℓ	(m, p^s)	N	$\log_2 \rho f_{\mathrm{PG}}(0)$	$\log_2 \rho f_{\mathrm{PG}}(\mu^*)$
2	$(3, 3^1)$	120	10.08	11.78
	$(4, 3^1)$	363	13.26	16.58
	$(5, 3^1)$	1092	16.43	22.97
	$(6, 3^1)$	3279	19.60	29.33
	$(7, 3^1)$	9840	22.77	37.27
3	$(3, 2^2)$	340	12.41	14.48
	$(4, 2^2)$	1364	16.41	20.50
	$(5, 2^2)$	5460	20.41	28.52
4	$(3, 5^1)$	780	14.25	16.62
	$(4, 5^1)$	3905	18.90	23.60

Similar to the results in Table 1, the constructed AC codes $\mathcal{M}_{\mu^*}$ have greater number of codewords than the base AC codes $\mathcal{M}_0$, and their effectiveness becomes high with increasing the dimension m of EG(m, p^s). Moreover, it should be noted that all the AC codes $\mathcal{M}_{\mu^*}$ in Table 3 have higher coding rate than EG-AC codes $\mathcal{B}_{\mu^*}$ of the same length in Table 1. This reason will be discussed in the next section.

We show some examples of PG-AC codes based on QC-LD matrices in Table 4. As shown in Corollary 2, although the code length increases from an original QC-LD matrix M_0 in this case, we replace the $p^{ms} \times p^{ms}$ identity matrix I in Eq. (9) with the $f^{(m)}_{\mathrm{PG}}(0) \times f^{(m)}_{\mathrm{PG}}(0)$ identity matrix to construct a new QC-LD matrix for reference. We show the logarithm of the code size for $\mathcal{M}_0$ in the column of "$\log_2 \rho f_{\mathrm{PG}}(0)$" and for $\mathcal{M}_{\mu^*}$ in that of "$\log_2 \rho f_{\mathrm{PG}}(\mu^*)$." Note that the code length is $N = \gamma(p^{(m+1)s} - 1)/(p^s - 1)$, where $(p^{(m+1)s} - 1)/(p^s - 1)$ expresses the number of points in PG(m, p^s). In Table 3, the constructed AC codes $\mathcal{M}_{\mu^*}$ have greater number of codewords than the base AC codes $\mathcal{M}_0$, and their effectiveness becomes high with increasing the dimension m of PG(m, p^s).

6.3 Relation between Two Proposed Methods

We have proposed two explicit construction methods in Sects. 5 and 6. Especially, as shown in Sects. 5.1 and 6.2, if we want to construct an AC code of $N = p^{ms}$ and $\ell = p^s - 1$, we have two options. That is, (i) to construct $\mathcal{B}_{\mu^*}$ from EG(m, p^s) and (ii) to combine M_0 and B_{μ^*} which are both defined over GF$(p^{(m-1)s})$.

We here compare these two constructions. From Eq. (22), the number of codewords of an EG-AC code $\mathcal{B}_{\mu^*}$ is $f_{\mathrm{EG}}^{(m)}(\mu^*) = O(N^{\frac{1}{4}(m+2+\delta(m))})$. On the other hand, the number of codewords of $\mathcal{M}_{\mu^*}$ is $\rho f_{\mathrm{EG}}^{(m-1)}(\mu^*)$ such that $\rho = p^{(m-1)s}$ and $\mu^* = \lfloor \frac{m-1}{2} \rfloor$. Let a function $\delta'(m)$ be

$$\delta'(m) := \begin{cases} \frac{5}{m}, & \text{for odd } m, \\ \frac{4}{m}, & \text{for even } m. \end{cases} \tag{25}$$

Then we have

$$\begin{aligned} \rho f_{\mathrm{EG}}^{(m-1)}(\mu^*) &= O\left(p^{\frac{1}{4}ms\{m+4-\delta'(m)\}}\right) \\ &= O\left(N^{\frac{1}{4}\{m+4-\delta'(m)\}}\right), \end{aligned} \tag{26}$$

where the last equality is obtained by $N = p^{ms}$ (see Appendix for the derivation). We see that $\rho f_{\mathrm{EG}}^{(m-1)}(\mu^*) > O(N^{\frac{1}{4}\{m+3\}})$ for $m \geq 4$, which is greater than $f_{\mathrm{EG}}^{(m)}(\mu^*)$. This fact implies that if $m \geq 4$, the AC code $\mathcal{M}_{\mu^*}$ is more effective than EG-AC code $\mathcal{B}_{\mu^*}$. On the other hand, if (and only if) $m \leq 3$, $\mathcal{B}_{\mu^*}$ is more effective than $\mathcal{M}_{\mu^*}$. From Eq. (26), the second construction combining an QC-LD matrix and an EG-AC code becomes highly effective with increasing the value of m.

7. Comparison with a Previous Method

In this section, we compare the effectiveness of proposed methods with the method of Kang et al. based on group-divisible design [16], [17].

In [16], [17], the conventional condition of AC codes has been also relaxed. In their relaxed condition, any codewords have the equal Hamming weight k. The codewords of an AC code based on group-divisible design can be divided into some groups of the same cardinality. A codeword has exactly one 1-component in common with other codewords in the same group, and it has no 1-component in common with codewords in different groups. Since these conditions imply that any codeword has at most one 1-component in common with other codewords, the conditions of AC codes in [16], [17] are a spacial case of our relaxed condition in Sect. 4 with $t = 1$.

In the code construction in [17], the code length is $N = p^m$ where p is a prime and $m \geq 2$ is a positive integer. The number of codewords is $p^{2(m-1)}$ and $k = p$. We can easily check that the number of codewords is expresses as $O(N^{2-\frac{2}{m}})$. On the other hand, the proposed method based on Euclidean geometry gives the number of codewords $O\Big(N^{\frac{1}{4}(m+2+\delta(m))}\Big)$ from Eq. (22). If we use the proposed method based on QC-LD matrices combined with EG(m, s), the number of codewords is $O\Big(N^{\frac{1}{4}(m+4-\delta'(m))}\Big)$ from Eq. (26). Note that both AC codes by the proposed methods can have the same code length and resilience as those of the AC codes by Kang et al. [17]. Therefore, for any $m \geq 3$, we can show that the AC codes by the proposed methods have larger coding late while the code length and the resilience are equal (if $m = 2$, all methods give the AC codes with same parameters).

8. Conclusion and Future Improvements

In this paper, for some class of AC codes proposed by Trappe et al., two methods for increasing their coding rate were proposed based on finite field arithmetics, while their resilience is maintained. We showed examples of the AC codes with the maximal number of codewords for a given finite geometry. In the first method, the obtained AC code can have the greater number of codewords than a conventional AC code by Trappe et al. as the dimension m of a finite geometry FG(m, p^2) increases. Taking a similar approach to this construction method, other methods for constructing efficient AC codes based on QC-LD matrices were proposed. In this method, although all the case does not necessarily guarantee the same resilience, conditions on parameters which provides the same resilience were derived. Consequently, we can construct a fingerprinting system which can provide service of distributing a digital content for more users, while keeping both the resilience and the distortion to original digital contents.

Unfortunately, the codes obtained by the proposed method have comparatively large code lengths, which implies the distortion to the original content by these codes might be large. An effective shortening method of the code while the resilience is maintained should be devised. In this paper, the resilience is guaranteed by assuming no noise sequence. The performance of the AC codes, where there occurs a noise sequence, should be analyzed. An effective detecting algorithm for the AC codes considered in this paper is also needed.

Acknowledgments

The authors wish to thank Prof. T. Kawabata at the University of Electro-Communications for his valuable comments. This work is partly supported by by MEXT, Grant-in-Aid for Young Scientists (B) No. 20760249 and the Telecommunications Advancements Foundation (TAF).

References

[1] D. Boneh and J. Shaw, "Collusion-secure fingerprinting for digital data," IEEE Trans. Inf. Theory, vol.44, no.5, pp.1897–1905, Sept. 1998.

[2] I. Djurdjevic, J. Xu, K. Abdel-Ghaffar, and S. Lin, "A class of low-density parity check codes constructed based on Reed-Solomon codes with two information symbols," IEEE Commun. Lett., vol.7, no.7, pp.317–319, June 2003.

[3] H. Fujita and K. Sakaniwa, "An efficient encoding method for LDPC codes based on cyclic shift," Proc. 2004 IEEE Int. Symp. on Inform. Theory (ISIT2004), pp.275–, Chicago, USA, June-July 2004.

[4] S. He and M. Wu, "Joint coding and embedding techniques for multimedia fingerprinting," IEEE Trans. Information Forensics and Se-

curity, vol.1, pp.231–247, June 2006.

[5] S. Lin and D.J. Costello, Jr., Error Control Coding: Fundamentals and Applications, 2nd ed., Prentice-Hall, Upper Saddle River, NJ, 2004.

[6] T. Mittelholzer, "Efficient encoding and minimum distance bounds of Reed-Solomon-type array codes," Proc. 2002 IEEE Int. Symp. on Inform. Theory (ISIT2002), p.282, Lausanne, Switzerland, June-July 2003.

[7] C. Podilchuk and W. Zeng, "Image adaptive watermarking using visual models," IEEE J. Sel. Areas Commun., vol.16, no.4, pp.525–540, May 1998.

[8] R. Safavi-Naini and Y. Wang, "New results on frame-proof codes and traceability schemes," IEEE Trans. Inf. Theory, vol.47, no.7, pp.3029–3033, Nov. 2001.

[9] J.N. Staddon, D.R. Stinson, and R. Wei, "Combinatorial properties of frameproof and traceability codes," IEEE Trans. Inf. Theory, vol.47, no.3, pp.1042–1049, March 2001.

[10] H. Tang, J. Xu, S. Lin, and K.A.S. Abdel-Ghaffar, "Codes on finite geometries," IEEE Trans. Inf. Theory, vol.51, no.2, pp.572–596, Feb. 2005.

[11] W. Trappe, M. Wu, Z.J. Wang, and K.J.R. Liu, "Anti-collusion fingerprinting for multimedia," IEEE Trans. Signal Process., vol.51, no.4, pp.1069–1087, April 2003.

[12] M. Wu, W. Trappe, Z.J. Wang, and K.J.R. Liu "Collusion-resistant fingerprinting for multimedia," IEEE Signal Process. Mag., vol.21, no.2, pp.15–27, March 2004.

[13] H. Yagi, T. Matsushima, and S. Hirasawa, "New traceability codes against a generalized collusion attack for digital fingerprinting," Proc. 2006 Int. Workshop on Information Security Applications (WISA2006), pp.569–584, Jeju Island, Korea, Aug. 2006.

[14] K. Yang and T. Helleseth, "On the minimum distance of array codes as LDPC codes," IEEE Trans. Inf. Theory, vol.49, no.12, pp.3268–3271, Dec. 2003.

[15] J. Yang, P. Liu, and G.Z. Tan, "The digital fingerprint coding based on LDPC," Proc. 2004 7th Int. Conf. on Signal Processing (ICSP2004), pp.2600–2603, Beijing, China, Aug.-Sept. 2004.

[16] I.K. Kang, C.-H. Lee, H.-Y. Lee, J.-T. Kim, and H.-K. Lee, "Averaging attack resilient video fingerprinting," Proc. IEEE Int. Symp. on Circuits and Systems, pp.5529–5532, Kobe, Japan, May 2005.

[17] I.K. Kang, K. Sinha, and H.-K. Lee, "New digital fingerprint code construction scheme using group-divisible design," IEICE Trans. Fundamentals, vol.E89-A, no.12, pp.3732–3735, Dec. 2006.

Appendix: Derivation of Eq. (26)

If constructing $\mathcal{M}_{\mu^*}$ from $\mathcal{M}_0$ and $\mathcal{B}_{\mu^*}$ defined over GF($p^{(m-1)s}$) with $\gamma = p^s$, the code length is $N = p^{ms}$, and the number of codewords is $\rho f_{\mathrm{EG}}^{(m-1)}(\mu^*)$ such that $\rho = p^{(m-1)s}$.

By substituting m with $m-1$ in Eq. (21), we have

$$\begin{aligned} f_{\mathrm{EG}}^{(m-1)}(\mu^*) &= O\left(p^{\frac{1}{4}\left\{(m-1)^2+2(m-1)+(m-1)\delta(m-1)\right\}s}\right) \\ &= O\left(p^{\frac{1}{4}\left\{m^2-1+(m-1)\delta(m-1)\right\}s}\right) \\ &= O\left(p^{\frac{1}{4}\left\{m^2-m\delta(m)\right\}s}\right), \qquad (\mathrm{A}\cdot 1) \end{aligned}$$

where the last equality can be obtained by the relation $1-(m-1)\delta(m-1) = m\delta(m)$. Therefore we have

$$\begin{aligned} \rho f_{\mathrm{EG}}^{(m)}(\mu^*) &= O\left(p^{(m-1)s}p^{\frac{1}{4}\left\{m^2-m\delta(m)\right\}s}\right) \\ &= O\left(p^{\frac{1}{4}\left\{m^2+4(m-1)-m\delta(m)\right\}s}\right) \\ &= O\left(p^{\frac{1}{4}ms\left\{m+4-\delta(m)-\frac{4}{m}\right\}}\right) \\ &= O\left(p^{\frac{1}{4}ms\{m+4-\delta'(m)\}}\right), \qquad (\mathrm{A}\cdot 2) \end{aligned}$$

where the last equality follows from Eq. (25). Since $N = p^{ms}$, we obtain Eq. (26).

Hideki Yagi was born in Yokohama, Japan, on Oct. 14, 1975. He received the B.E. degree, M.E. degree, and Dr.E. degree in Industrial and Management Systems Engineering from Waseda University, Tokyo, Japan, in 2001, 2003 and 2005, respectively. From 2005 to 2008, he was with Media Network Center, Waseda University as a Research Associate and an Assistant Professor. He is currently an Assistant Professor at the University of Electro-Communications, Tokyo, Japan. His research interests are coding theory and information security. He is a member of the Society of Information Theory and its Applications and IEEE.

Toshiyasu Matsushima was born in Tokyo, Japan, on Nov. 26, 1955. He received the B.E. degree, M.E. degree and Dr.E. degree in Industrial and Management Systems Engineering from Waseda University, Tokyo, Japan, in 1978, 1980 and 1991, respectively. From 1980 to 1986, he was with Nippon Electric Corporation, Kanagawa, Japan. From 1986 to 1992, he was a lecturer at Department of Management Information, Yokohama College of Commerce. From 1993, he was an associate professor and since 1996 has been a professor of School of Science and Engineering, Waseda University, Tokyo, Japan. His research interests are information theory and its application, statistics and artificial intelligence. He is a member of the Society of Information Theory and Its Applications, the Japan Society for Quality Control, the Japan Industrial Management Association, the Japan Society for Artificial Intelligence and IEEE.

Shigeichi Hirasawa was born in Kobe, Japan, on Oct. 2, 1938. He received the B.S. degree in mathematics and the B.E. degree in electrical communication engineering from Waseda University, Tokyo, Japan, in 1961 and 1963, respectively, and the Dr.E. degree in electrical communication engineering from Osaka University, Osaka, Japan, in 1975. From 1963 to 1981, he was with the Mitsubishi Electric Corporation, Hyogo, Japan. Since 1981, he has been a professor of School of Science and Engineering, Waseda University, Tokyo, Japan. In 1979, he was a Visiting Scholar in the Computer Science Department at the University of California, Los Angels (CSD, UCLA), CA. He was a Visiting Researcher at the Hungarian Academy of Science, Hungary, in 1985, and at the University of Trieste, Italy, in 1986. In 2002, he was also a Visiting Faculty at CSD, UCLA. From 1987 to 1989, he was the Chairman of Technical Group on Information Theory of IEICE. He received the 1993 Achievement Award, and the 1993 Kobayashi-Memorial Achievement Award from IEICE. In 1996, he was the President of the Society of Information Theory and Its Applications (Soc. of ITA). His research interests are information theory and its applications, and information processing systems. He is an IEEE Life Fellow and a member of Soc. of ITA, the Operations Research Society of Japan, the Information Processing Society of Japan, the Japan Industrial Management Association, and Informs.