國立交通大學

應用數學系

碩士論文

智慧財產權保護碼

On Codes for Copyright Protection

研究生：黎冠成

指導老師：符麥克　教授

中華民國九十五年六月

智慧財產權保護碼

On Codes for Copyright Protection

研 究 生：黎冠成　　Student：Li, Guan-Cheng
指導教授：符麥克　　Advisor：Michael Fuchs

國 立 交 通 大 學
應 用 數 學 系
碩 士 論 文

A Thesis
Submitted to Department of Applied Mathematics
College of Science
National Chiao Tung University
in Partial Fulfillment of the Requirements
for the Degree of
Master
in
Applied Mathematics
June 2006

Hsinchu, Taiwan, Republic of China

中 華 民 國 九十五 年 六 月

LI, GUAN-CHENG

ON CODES FOR COPYRIGHT PROTECTION

Master Thesis

In Partial Fulfillment of Requirement
For the Degree of Master

Advisor:
Professor Michael Fuchs

Submitted to
Institute of Applied Mathematics
College of Science
National Chiao Tung University
Hsinchu, Taiwan, Republic of China
June 2006

Contents

Preface

Unauthorized illegal duplication is a major problem in many areas. For digital media, duplication is especially easy because copying such material is immediate and no information is degraded in the process. In addition, the growth of the Internet makes it possible to distribute the material in a much larger scale than before. Because of both technical and legal issues, it is often difficult to find and prosecute the pirates. Hence, to protect digital copies is a complicated task. Recently, electronic fingerprinting was devised as a method to discourage people from illegally redistributing their legally purchased copy.

Electronic fingerprinting deals with the problem of object identification through the use of electronic marks, unique to each object. We consider fingerprinting for the purpose of protecting innocent users from being framed and tracing of illegitimately copied and distributed data, so called pirate copies.

We examine the possibilities of designing fingerprinting codes that are resistant to tampering. We show that under certain assumptions, we are often able to protect blameless users and even trace back the criminals.

Also, with the model we describe, the result of tracing should be reliable. That is, our tracing may fail in the sense that no pirates are identified, but we should not mistakenly accuse an innocent user. In this thesis, we mainly focus on a number of code constructions, and discuss their mathematical properties against piracy.

Acknowledgement

First of all, I am especially grateful to my supervisor Professor Michael Fuchs for his incessant patience in giving me professional guidance on mathematics research. He is an enthusiastic mentor who taught me not only the essence of the theory but also to articulate it clearly and convincingly. This is particularly valuable for me since I was trained to be a problem solver rather than a seller of my theory under traditional education before.

I want to thank Professor Kar-Kin Zao of Institute of Computer Science for providing inspirations in his course “Internet Security” which gave me related ideas on electronic security.

I would also like to thank Professor Ta-Yuan Huang for his endless encouragement, in particular, regarding the undergraduate training of algebra in his class, which later introduced me into wonderful fields of mathematics and computer science.

Since this is my first time typing in LateX, I owe many thanks to my senior classmates for giving me advice about the typesetting of a thesis. Finally, I would show my appreciation to the support of my beloved girlfriend, Yu-Han and my family, and ascribe the completion of my degree to my family members.

Li, Guan-Cheng
psd23.am93g@nctu.edu.tw
National Chiao Tung University
May, 16th, 2006.

Chapter 1

Introduction

Conventional mechanisms for copyright protection are obviously incapable of treating digital data owing to the essential difference of the documents. This leads to the interest of developing other means for deterring the pirates from illegally redistributing products. Digital fingerprinting, for example, can serve our purpose. A fingerprint is a set of number sequence added to digital data that can be detected or extracted later to make an assertion about the data. The fingerprint can be applied in several areas, including:

- Ownership assertion
- Authentication and integrity verification
- Content labeling
- Digital watermarking
- Access control protocols
- Content protection
- Detection of copyright violations
- Secure on-line multimedia distribution
- Resource usage control

- Trust and trust management

With digital fingerprinting, a publisher embeds a unique fingerprint into each distributed copy of a document, keeping a database of sold copies and their corresponding buyers. If an illegally distributed copy is discovered, the publisher would certainly want to trace back to the unauthorized user by comparing its fingerprint to the database. Because of the uniqueness of the fingerprint, the pirates would introduce some kind of marking distortion upon the documents. In order to redistribute illegal copies anonymously, a pirate may try different types of attacks to disclose the fingerprint. Assuming that the pirate has an access to a single document copy, that has been marked for him, he may try to restore the original document by identifying and removing the fingerprint. However, such an attack may be questionable if the fingerprinting is hidden carefully and scattered all over the document. A stronger attack results if several pirates collude and compare their independently marked copies. They can identify the hidden fingerprint by locating the differences among their copies, replace them with other feasible marks, combine their copies into several new ones whose fingerprint differs from all of the pirates, and resell their pirated products with different fingerprints without ever worrying about being caught. The copies replaced by feasible marks are called the descendence as will be made precise in Chapter 2.

Frameproof codes were introduced by Boneh and Shaw [8] as a method of digital fingerprinting which prevents a coalition of a specified size w[¶] from framing a user not in the coalition. Several constructions of w-frameproof codes were mainly introduced later on by Stinson, Wei, Encheva, and Cohen [12, 14, 30].

Besides the design of frameproof codes against piracy, an efficient traitor tracing algorithm might be necessary in order to identify the offenders. The traitor tracing problem was introduced by Chao, Fiat and Naor for broadcast encryption systems, where the data should be accessible only to authorized users. When an illegal copy produced by a group of authorized users of the copyrighted material is detected, traitor tracing schemes allow to trace back at least one producer of it. In particular, these schemes are suitable for pay-per-view TV applications. We consider, as an example, a pay-per-view movie type scenario introduced by Fiat and Tassa. In this scenario, the content is divided into n segments. Each of this

[¶] w is a predetermined threshold for designing codes.

segments is marked with one of q different symbols. Each user receives a differently marked copy of the content. The ordered set of the marks for each copy can be given as a q-ary vector of length n. A coalition of colluding users can make an illegal copy by combining different segments of their data and broadcast it. After an illegal copy is detected, traitor tracing schemes attempt to reveal at least one traitor. The practical applications require to accommodate as many users as possible when there is a restriction on the number of symbols which can be used for marking the data. On the other hand, some digits of the codes, whatever registered or pirated, might happen to be erased or appear undetectable however accidentally or deliberately. Therefore, there might be a need to distinguish codes in more than one position in order to be fault-tolerant.

Several codes providing some forms of traceability have been designed to be used in these schemes. These codes have been extensively studied in recent years. The weak forms are frameproof (FP) codes and secure frameproof (SFP) codes. A stronger form includes identifiable-parent-property (IPP) codes introduced by Hollmann, van Lint, Linnartz and Tolhuizen [21], and traceability (TA) codes introduced by Chor, Fiat and Naor [10]. Such codes allow the tracing of at least one parent of any illegal copy when the size of the coalition of colluders does not exceed some given number w. Their combinatorial properties and related structures with codes have been studied by Hollmann et al., Staddon, Stinson and Wei, Barg, et al. and Sarkar [28, 30, 31, 21].

As a matter of fact, FP codes turn out to be a subclass of SFP codes, SFP codes are a subclass of IPP codes, and IPP codes are a subclass of TA codes. They will be mathematically formulated in Chapter 2. Their relationship with hash families will be treated in Chapter 3.

The aim of this thesis is to study the above codes under the presence of unreadable marks. In such a situation, Boneh and Shaw [8] pointed out that codes with traitor tracing properties do not exist. This will be made precise in Chapter 4. They provided an alternative, slightly weaker form of traceability codes by using randomness and probabilistic traitor tracing. Their work is important from an application point of view because they trade off some accuracy for a fast traitor-tracing algorithm under the condition that undetectable marks exist. Hence, IPP and TA codes are only interesting from a theoretic point of views and are less applicative owing to the intolerance of undetectable marks. The probabilistic traitor

tracing (PTT) algorithm due to Boneh and Shaw will be presented in the second half of Chapter 4.

However, it should be pointed out that if there are too many unreadable marks then even the probabilistic approach fails. An extreme case would be a codeword filled with unreadable marks which is totally impossible for the distributor to recognize, not mentioning tracing back. However, the pirated products with unreadable marks will soon be detected by the distributor, and in practical situations, the pirates will scatter only a few unreadable marks to the products in order to falsely convince the customers that the pirated products are copyrighted ones.

On the other hand, FP and SFP codes are immune from undetectable marks. Since SFP is stronger than FP, SFP codes find more practical applications such as the distribution of multi-license. In such a scenario a distributor sells his products to an institution instead of an individual. The distributor then gives a couple of codes as a base to generate more codes for the use of employees in the institution. The distributor certainly hopes that the base codes exhibit the secure frameproof property so that codewords authorized to each institution can be treated independently.

We conclude the introduction by giving a sketch of the thesis. In Chapter 2, we will provide the basic definitions which will be more general than the original definitions given by Stinson in [30]. Chapter 3 is then dedicated to the relationship between hash families and codes. In Chapter 4, we study unreadable marks and the probabilistic approach, and prove that IPP and TA codes do not exist. Finally, in Chapter 5, we investigate explicit constructions for SFP codes. Most of the results in Chapter 4 and 5 are taken from the literature. We however tried to increase clarity by adding more details and giving simplified proofs of many results. Moreover, we tried to give a complete picture by incorporating all results presently known concerning codes for copyright protection under the presence of unreadable marks.

Chapter 2

Definitions and Basics

2.1 Some Coding Theory

Throughout the thesis, we denote by N the code length, by n the code size, and by q the number of alphabets over a code C.

2.1.1 Hamming Distance

Definition: The **Hamming distance** d_H between two codewords is the number of positions whose entries are different.

Example 2.1. $d_H(11001, 01101) = 2$

2.1.2 Hamming Weight

Definition: The **Hamming weight** denotes the number of nonzero entries in a codeword.

Example 2.2. *The Hamming weight of* $(1, 0, 1, 1, 0)$ *is usually denoted as* $weight\,(1, 0, 1, 1, 0) = 3$.

2.1.3 Minimum Distance

Definition: The **minimum distance** of a code $C \subseteq \sum^N$ is the least Hamming distance $d_H(x, y)$ between any pair of different codewords $x, y \in C$.

2.1.4 Error Correcting Code

Definition: The $(N, k, d)_q$**-Error Correcting Code (ECC)** is a q-ary linear code with cardinality k, code length N, and minimum Hamming distance between any two codewords d. It follows that the code rate R is k/n and code size is q^k. In some situations we also need to specify by D the maximum Hamming distance between any two codewords. Normally we omit the subscript in the binary case.

In the nonlinear case, (N, n, q) is a q-ary code of length N with code size n. The rate is computed as $N^{-1} \log_q |n|$. The following two nonlinear codes are for practical applications. One is the **constant-weight code** being a binary code whose codewords have a fixed number of $1's$. The other is the **equidistant code** being a code where any two codewords enjoy a fixed Hamming distance.

We further introduce some more terminology for linear ECC as follows:

Theorem 2.1 (Singleton Bound)**.** *For a code* $C : \sum^k \mapsto \sum^N$ *with minimum distance* d, $N \geqslant k + d - 1$.

Codes satisfying the equality of Singleton Bound are called **Maximum Distance Separable (MDS)** code.

A code C with odd d is said to be a **Perfect Code** if for every codeword w of length N not in C, there is an unique codeword w_0 in C such that $d_H(w, w_0) \leqslant (d-1)/2$.

2.1.5 Code Composition

Definition: Let A be an (N_2, n_2, q_2) code over an alphabet Q_2 with $|Q_2| = q_2$ and let B be an (N_1, q_2, q_1) code over an alphabet Q_1 with $|Q_1| = q_1$. Let $Q_2 =$

$\{a_1, \ldots, a_{q_2}\}$ and let $B = \{b_1, \ldots, b_{q_2}\}$. Let $\theta : Q_2 \mapsto B$ be the one-to-one mapping defined by $\theta(a_i) = b_i$ for $1 \leqslant i \leqslant q_2$. For any codeword $a = (a_1, \ldots, a_{N_2}) \in A$ we denote by $\tilde{a} = (\theta(a_1), \ldots, \theta(a_{N_2})) = (b_1, \ldots, b_{N_2})$ the q_1-ary sequence of length $N_1 N_2$ obtained from a by using θ. The set

$$A \bigstar B = \{\tilde{a} = (b_1, \ldots, b_{N_2}) \,|\, (a_1, \ldots, a_{N_2}) \in A\}$$

is called $(N_1 N_2, n_2, q_1)$ concatenation code of A and B, with inner code A and outer code B.

2.2 Descendence

Certain properties of the codes discussed above can be formulated using mathematical notations. Subsequently, let C be a code of length N on an alphabet Q with $|Q| = q$.

We denote by "?" the unreadable mark deliberately or accidently inserted into the pirated codewords. For any subset of codewords $C_0 \subseteq C$, we define the set of descendants of C_0, denoted $desc(C_0)$ by

$$desc\,(C_0) := \left\{ x \in Q^N \;:\; x_i \in \begin{cases} \{a_i \;:\; a \in C_0\}\,, & \text{if } |\,\{a_i \;:\; a \in C_0\}\,| = 1; \\ \{a_i \;:\; a \in C_0\} \cup \{?\}, & \text{otherwise.} \end{cases} \right\}.$$

Namely, the set $desc(C_0)$ consists of the N-tuples plus perhaps some unreadable marks that could be produced by a coalition holding the codewords in the set C_0. If in a certain entry there is only one choice for the coalition, then only that feasible element will be used in that entry. Besides, the coalition could choose more than one elements plus a question mark.

Let $w \in \mathbb{N}$ be the number of codewords a coalition could have. We define the $w - descendant$ code of C, denoted $desc_w(C_0)$[††], as follows:

$$desc_w(C) := \bigcup_{C_0 \subseteq C, |C_0| \leq w} desc\,(C_0)\,.$$

[††] Some papers also call it the *feasible set*.

In other words, the set $desc_w(C)$ consists of the N-tuples that could be produced by comparing the codewords they jointly hold by some coalition of size at most w.

Example 2.3. *Let* $C = \{(1,2,0,1,1),(2,2,0,1,0)\}$.
Then $desc_2(C) = \left\{ \left(\begin{pmatrix} 1 \\ 2 \\ ? \end{pmatrix}, 2, 0, 1, \begin{pmatrix} 0 \\ 1 \\ ? \end{pmatrix} \right) \right\}$. *And,* $|desc_2(C)| = 9$.

Remark 2.1. *Two pirated codewords (1,0,0,?,?) and (1,0,1,?,?) are obviously different because of the third entry. However, when given two codewords (1,0,1,?,?) and (1,0,1,?,?), we still treat them differently although they might become the same codewords.*

Next, we give the definitions concerning the mathematical properties required by FP, SFP, IPP, and TA codes.

2.3 Frameproof code

Definition: C is a w-frameproof (**FP**) code provided that for all $x \in desc_w(C)$, $x \in desc(C_i) \cap C$ implies $x \in C_i$.

Roughly speaking, a code is w-frameproof if no coalition of size at most w can frame another user not in the coalition by producing the codeword held by that user.

2.4 Secure Frameproof code

Definition: C is a w-secure frameproof (**SFP**) code provided that for all $x \in desc_w(C) \cap Q^N$, $x \in desc(C_i) \cap desc(C_j)$ implies that $C_i \cap C_j \neq \emptyset$, where $i \neq j$.

In other words, a code is w-secure frameproof if no coalition of size at most w can frame a disjoint coalition of size at most w by producing an N-tuple that could have been produced by the second coalition. In other words, whenever given two

disjoint coalitions C_1 and C_2 of size at most w, we know that they cannot produce the same false fingerprint, i.e., $desc(C_i) \cap desc(C_j) \cap Q^N = \emptyset$.

Remark 2.2. *Note that FP and SFP codes are resistent from the threats of unreadable marks because if innocent users are safe from being framed by colluded codewords, they are even safer from being framed by those codewords with unreadable marks under the assumption mentioned earlier in Remark 2.1.*

2.4.1 Separating Weights

Here, we do not look at the unreadable marks.

Definition: The **separating weight** λ_w of two coalitions is the least number of positions where the descendences of them are separated. The normalized separating weight is $\tau_w := \lambda_w / N$ where N is the code length.

Obviously, a code is $w - SFP$ if and only if $\lambda_w > 0$.

Sometimes λ_w is incremented by various means such as concatenation method in order to overcome some undetectable marks problem. Namely, if some unreadable marks occurs in a supposedly separating position, other positions can serve as a backup in order to separate codes correctly.

Example 2.4. *The code* $\{1122334, 2112433, 1212343\}$ *is a 2-SFP with* $\lambda_2 = 2$.

Assign 1122334 to user 1 as $u^{(1)}$, 2112433 to user 2 as $u^{(2)}$, and 1212343 to

user 3 as $u^{(3)}$.

$$\begin{aligned}
Coalition\left(\{\text{user } 1 \text{ and } 2\}\right) &= desc_2\left(\left\{u^{(1)}, u^{(2)}\right\}\right) \\
&= \left\{\left(\binom{1}{2}, 1, \binom{1}{2}, 2, \binom{3}{4}, 3, \binom{3}{4}\right)\right\} \\
Coalition\left(\{\text{user } 2 \text{ and } 3\}\right) &= desc_2\left(\left\{u^{(2)}, u^{(3)}\right\}\right) \\
&= \left\{\left(\binom{1}{2}, \binom{1}{2}, 1, 2, \binom{3}{4}, \binom{3}{4}, 3\right)\right\} \\
Coalition\left(\{\text{user } 1 \text{ and } 3\}\right) &= desc_2\left(\left\{u^{(1)}, u^{(3)}\right\}\right) \\
&= \left\{\left(1, \binom{1}{2}, \binom{1}{2}, 2, 3, \binom{3}{4}, \binom{3}{4}\right)\right\}
\end{aligned}$$

Note that the coalition of user 1 and 2 cannot frame user 3 because of the second and sixth entries, the coalition of user 2 and 3 cannot frame user 1 because of the third and seventh entries, and the coalition of user 1 and 3 cannot frame user 2 because of the first and fifth entries.

Note that the separating weight of such code is $\lambda_2 = 2$ because they are differentiated in at least two positions. The normalized separating weight is therefore $\tau_2 = 2/7$.

2.5 Identifiable parent property code

Definition: C is a w-identifiable parent property (**IPP**) code provided that for all $x \in desc_w(C)$, it holds that

$$\bigcap_{i\,:\,x \in desc(C_i)} C_i \neq \emptyset.$$

A code enjoys the w-identifiable parent property if no coalition of size at most w can produce an N-tuple that cannot be traced back to at least one member of the coalition. In such a code, whenever a codeword belongs to the descendance of a coalition of size at most w, at least one of the parents of the coalition can be identified.

2.6 Traceability code

Definition: For $x, y \in Q^N$, define $I(x, y) = \{i : x_i = y_i\}$. C is a w-traceability (**TA**) code provided that, for all $x \in desc_w(C)$, $x \in desc(C_i)$ implies that there is at least one codeword $y \in C_i$ such that $|I(x, y)| > |I(x, z)|$ for any $z \in C \setminus C_i$.

In fact, $I(x, y)$ stands for the closeness of two codewords, which can also be expressed as $N - d_H(x, y)$, where N denotes the length of the codeword, and $d_H(x, y)$ is the hamming distance of two codewords.

A code enjoying the w-traceability property allows an efficient (i.e., linear-time) algorithm to determine an identifiable parent. More precisely, if we compare an illegal codeword to each codeword in C, then the codeword closest to the illegal one will be one of the parent in the coalition. Note that TA property is much stronger than just IPP property which necessitates comparisons with $\binom{n}{w}$ sets, resulting in a nonlinear running time.

Remark 2.3. *It has to be made clear that IPP and TA codes appear vulnerable under the presence of unreadable marks because by definition we can say nothing if there are "?", not mentioning identifying or tracing the parents. This will be justified in the beginning of Chapter 4 where we show that they in fact do not exist.*

Remark 2.4. *If there are no unreadable marks in the pirated codewords, then IPP and TA codes can exist. However, the constructions of IPP and TA codes will not be treated because they are only of theoretical interest owing to intolerance of unreadable marks.*

In the sequel, we point out the relationships of these codes.

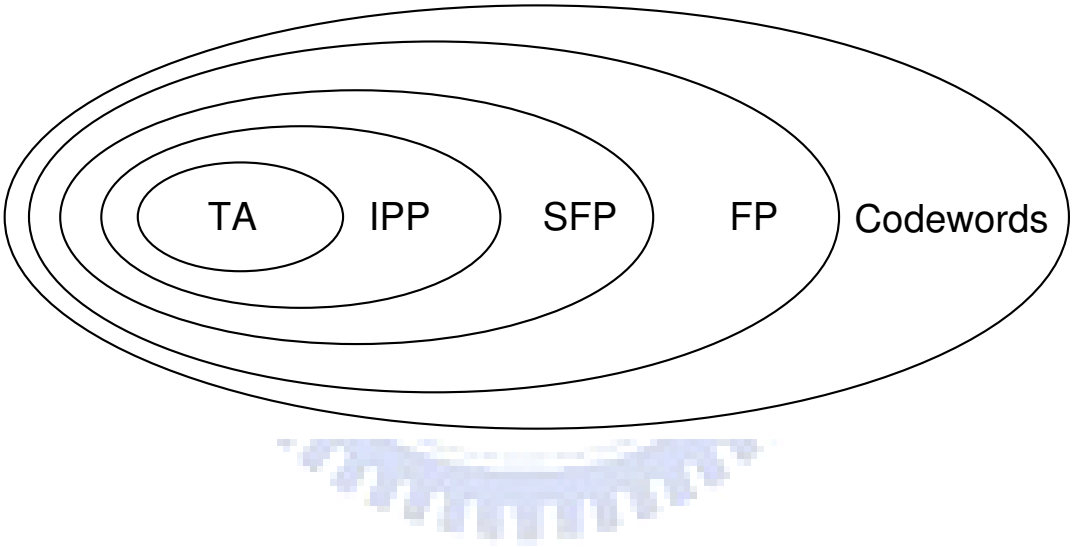
2.7 Relations

1. w**-SFP implies** w**-FP.** This is self-explanatory if we treat an individual as an independent coalition. Let one coalition $\mathcal{A}$ be of size at most w and the other coalition $\mathcal{B}$ be simply one individual. w-SFP assures that two disjoint

coalitions of size at most w cannot produce the same codeword. The coalition $\mathcal{B}$ is a trivial coalition since the descendence of $\mathcal{B}$ is $\mathcal{B}$ itself, which would not be framed by coalition $\mathcal{A}$ by the definition of SFP.

2. **w-IPP implies w-SFP.** This is clear because IPP itself is an intensified version of SFP. Namely, $(C_i \cap C_j) \subseteq \bigcap_{i:\, x \in desc(C_i)} C_i \neq \emptyset$.

3. **w-TA implies w-IPP.** Suppose C is a w-TA code. If $x \in desc_w(C)$, then there is a aubset $C_i \subseteq C$, where $|C_i| = w$, such that $x \in desc(C_i)$. Let $y \in C_i$ such that $|I(x,y)| \geqslant |I(x,z)|$ for all $z \in C_i$. Hence $|I(x,y)| \geqslant |I(x,z)|$ for any $z \in C$ by the definition of a w-TA code. We show that, for any $C_j \subseteq C$ with $|C_j| \leqslant w$, $x \in desc(C_j)$ implies $y \in C_j$. In fact, if $y \notin C_j$, then there is $w \in C_j$ such that $|I(x,w)| > |I(x,y)|$ by the definition of a w-TA code. This contradicts the fact that $|I(x,y)| \geqslant |I(x,z)|$ for any $z \in C$.



Chapter 3

Hash Families and Codes

Before going into explicit constructions of such codes, some preliminaries are needed to reinforce the mathematical structures and serve as basic tools in the construction.

Recently, hash families and related structures have been used to construct codes for copyright protection. Subsequently, we will define them and discuss their inter-relationship with the codes defined in the previous chapter.

3.1 Hash Functions

Let $n \geqslant m$. An (n, m)-hash function is a function $h : A \mapsto B$, where $|A| = n$ and $|B| = m$. An (n, m)-hash family is a finite set $\mathcal{H}$ of (n, m)-hash functions such that $h : A \mapsto B$ for each $h \in \mathcal{H}$, where $|A| = n$ and $|B| = m$. We use the notation $HF(N; n, m)$ to denote an (n, m)-hash family with $|\mathcal{H}| = N$.

3.2 Perfect Hash Families

Let n, m and w be integers such that $n \geqslant m \geqslant w \geqslant 2$. An (n, m, w)-perfect hash family is an (n, m)-hash family, $\mathcal{H}$, such that for any $X \subseteq A$ with $|X| = w$, there exists at least one $h \in \mathcal{H}$ such that $h|_X$ is injective. We use the notation $PHF(N; n, m, w)$ to denote an (n, m, w)-perfect hash family with $|\mathcal{H}| = N$.

3.3 Separating Hash Families

Let n, m, w_1 and w_2 be integers such that $n \geqslant m$. An (n, m, w_1, w_2)-separating hash family is an (n, m)-hash family, $\mathcal{H}$, such that for any $X_1, X_2 \subseteq A$ with $|X_1| = w_1$, $|X_2| = w_2$ and $X_1 \cap X_2 = \emptyset$, there exists at least one $h \in \mathcal{H}$ such that $\{h(x) : x \in X_1\} \cap \{h(x) : x \in X_2\} = \emptyset$. We use the notation $SHF(N; n, m, w_1, w_2)$ to denote an (n, m, w_1, w_2)-separating hash family with $|\mathcal{H}| = N$.

[16] provides a survey on hash families. The following theorem is immediate from the definition of perfect hash families and separating hash families.

Theorem 3.1. *Let $\mathcal{H}$ be an $(N; n, m)$ hash family.*

1. If $\mathcal{H}$ is a $PHF(N; n, m, w)$, then it is a $PHF(N; n, m, w')$ for all $w' \leqslant w$.
2. If $\mathcal{H}$ is a $SHF(N; n, m, w_1, w_2)$, then it is a $SHF(N; n, m, w_1', w_2')$ for all $w_1' \leqslant w_1$ and $w_2' \leqslant w_2$.
3. If $\mathcal{H}$ is a $PHF(N; n, m, w_1 + w_2)$, then it is a $SHF(N; n, m, w_1, w_2)$.

Next, we establish the relationship between hash families and codes, we depict a (N, n, q)-code, C, as an $n \times N$ matrix $M(C)$ on q symbols, where each row of the matrix corresponds to one of the codewords. Similarly, we can represent an $HF(N; n, m)$, $\mathcal{H}$, as an $N \times n$ matrix on m symbols, where each row of the matrix corresponds to one of the functions in $\mathcal{H}$. These two matrices are transpose to each other.

Given an (N, n, q)-code C, we define $\mathcal{H}(C)$ to be the $HF(N; n, q)$ whose matrix representation is $M(C)^\top$. Thus if $C = \{x^1, x^2, \cdots, x^n\}$ and $1 \leqslant j \leqslant N$, then the hash function $h_j \in \mathcal{H}(C)$ is defined by the rule $h_j(i) = x^i_j,\ 1 \leqslant i \leqslant n$.

Obviously, the matrix representation of PHF and SHF should satisfy the following:

Lemma 3.1. *A $PHF(N; n, q, w)$ can be depicted as an $N \times n$ matrix with entries from $\{1, 2, \ldots, q\}$ such that in any w columns, there exists at least one row such that the w entries are distinct.*

Lemma 3.2. *A $SHF(N; n, q, w_1, w_2)$ can be depicted as an $N \times n$ matrix with entries from $\{1, 2, \ldots, q\}$ such that in any two disjoint columns C_1 and C_2 of size w_1 and w_2 respectively, there exists at least one row such that the entries in the columns C_1 are distinct from the entries in the columns C_2.*

Hence the relationship between PHF and FP codes and between SHF and SFP codes follows immediately by definition.

Theorem 3.2. *A (N, n, q)-code, C, is a $w - FP$ code if and only if $\mathcal{H}(C)$ is an $SHF(N; n, q, w, 1)$.*

Theorem 3.3. *A (N, n, q)-code, C, is a $w-SFP$ code if $\mathcal{H}(C)$ is an $PHF(N; n, q, 2w)$, where $n \geqslant 2w$.*

Theorem 3.4. *A (N, n, q)-code, C, is a $w - SFP$ code if and only if $\mathcal{H}(C)$ is an $SHF(N; n, q, w, w)$, where $n \geqslant 2w$.*

The proofs are trivial. Perfect hash families and separating hash families turn out to be just another languages for FP and SFP codes.

3.4 Difference Matrices

Definition: An $(n, k; \lambda)$-**difference matrix** is a $k \times n\lambda$ matrix $D = (d_{i,j})$, with entries from $\mathbb{Z}_n$, in which the multiset

$$\{d_{h,j} - d_{i,j} \bmod n \ :\ 1 \leqslant j \leqslant n\lambda\}$$

contains every element of $\mathbb{Z}_n$ λ times, for all h, i such that $1 \leqslant h < i \leqslant k$.

Example 3.1. *If* $gcd\left((k-1)!, n\right) = 1$*, then the* $k \times n$ *matrix* D *defined by* $d_{i,j} = ij \bmod n$ *is a* $(n, k; 1)$*-difference matrix.*

The concept of difference matrix will serve as a tool later in the recursive construction of perfect hash families in Theorem 5.20.

3.5 Set Systems

Definition: A **set system** is a pair $(X, \mathcal{B})$ where X is a set of elements called points, and $\mathcal{B}$ is a set of subsets of X, the members of which are called blocks. A set system can be described by an incidence matrix. Let $(X, \mathcal{B})$ be a set system where $X = \{x_1, x_2, \ldots, x_n\}$ and $\mathcal{B} = \{B_1, B_2, \ldots, B_N\}$. The incidence matrix of $(X, \mathcal{B})$ is the $N \times n$ matrix $A = (a_{ij})$, where

$$a_{ij} = \begin{cases} 1 & \text{if } x_j \in B_i \\ 0 & \text{if } x_j \notin B_i. \end{cases}$$

Conversely, given an incidence matrix, we can define an associated set system in an obvious way. Here, if C is a $(N, n, 2)$-code, then the matrix $M(C)$ is a $0-1$ matrix, which can therefore be thought of as the incidence matrix of a set system. For any codeword $w \in C$, we will use B_w to denote the associated block in the corresponding set system.

3.6 Sandwich Free Families

A set system $(X, \mathcal{B})$ is an (w_1, w_2)-sandwich free family provided that, for any two disjoint subsets C_1, C_2 of $\mathcal{B}$, where $|C_1| \leqslant w_1$ and $|C_2| \leqslant w_2$, the following property holds:

$$\left(\bigcap_{B \in C_1} B\right) \bigcup \left(\bigcap_{B \in C_2} B\right) \nsubseteq \left(\bigcup_{B \in C_1} B\right) \bigcap \left(\bigcup_{B \in C_2} B\right)$$

An (w_1, w_2)-sandwich free family, $(X, \mathcal{B})$, will be denoted as an $(w_1, w_2)-SFF(N, n)$ if $|X| = n$ and $|\mathcal{B}| = N$.

The connection between SFF and SFP codes is stated as follows.

Theorem 3.5. *A $w - SFP(N, n)$ exists if and only if there exists a $(w, w) - SFF(N, n)$.*

The proof is not so straightforward like the previous one, and will be given in the proof of Theorem 5.16 which focuses on explicit constructions of such codes.

3.7 Secure Codes

A code C is w-secure if there exists a tracing algorithm $\mathcal{A}$ satisfying the following: if a coalition C of size at most w generates a word x then $\mathcal{A}(x) \in C$.

The tracing algorithm $\mathcal{A}$ on input x must output a member of the coalition C that generated the codeword. Hence, an illegal copy can be traced back to at least one member of the guilty coalition. Clearly there is no hope in recovering the entire coalition since some of its members might be passive; they are part of the coalition, but they contribute nothing to the construction of illegal copies.

Actually, the concept of w-secure codes is not new to us since we have the following result.

Proposition 3.1. *C is w-secure if and only if C is an w-IPP code.*

Proof. We firstly derive a necessary condition of a code to be w-secure. Consider the following scenario: let C be some code. Let C_1 and C_2 be two coalition of w colluders such that $C_1 \cap C_2 = \emptyset$. Suppose an unregistered codeword is caught which is marked by a codeword x which belongs to both $desc_w(C_1)$ and $desc_w(C_2)$. As a consequence, both coalitions are suspicious. Since their intersection is empty, it is not possible to determine with certainty who created the unregistered x. It follows that if C is w-secure then when the intersection of C_1 and C_2 is empty, the intersection of $desc_w(C_1)$ and $desc_w(C_2)$ must be empty as

well. Of course, the same is true for j subsets $C_1, \ldots, C_j$. This gives the necessary condition. The sufficient condition is self-explanatory by the definition of identifiable parent property of IPP code. ∎

Hence, TA codes are secure codes as well. However, both IPP and TA codes do not exist under the presence of unreadable marks as will be clarified in the next chapter. Therefore, IPP and TA codes are only interesting from a theoretic point of view, and will not be treated subsequently. In the next chapter we will explain more about unreadable marks and introduce a probabilistic traitor tracing algorithm to construct "almost" secure codes.

Chapter 4

Unreadable Marks and PTT

Unreadable marks or undetectable bits are symbols in an uncertain state. For instance, when the police or distributer recovers an illegal copy of an object, she might find some symbols undefined in the codeword or could hardly determine which state an unreadable mark is in. The only thing she can do is to simply replace them by "?"'s.

On the other hand, unreadable marks can be deliberately created by the coalitions in order to make traitor tracing less feasible and make themselves safer from being prosecuted. As a matter of fact, IPP and TA codes do not exist under the presence of unreadable marks as will be indicated later. However, FP and SFP codes are resistent from the threats of unreadable marks because if innocent users are safe from being framed by colluded codewords, they are even safer from being framed by those codewords with unreadable marks.

Without unreadable marks, IPP and TA codes can exist and have been investigated by several researchers in [21, 35, 10, 6, 2, 36, 37, 19, 29, 20]. However, in the context of fingerprinting, the assumption that marks cannot become unreadable is unrealistic.

Based on the above reasoning and the fact that SFP is an intensified version of FP codes, SFP finds more practical applications in industry. Therefore, the explicit construction of SFP codes will be our main focus in the next chapter.

The remaining of this chapter will explain how unreadable marks contravene the existence of IPP and TA codes. In order to overcome the problem, a probabilistic approach will be proposed.

4.1 Unreadable Marks

Recall in Section 3.7 the idea of secure codes is introduced. We rephrase Proposition 3.1 as the following lemma.

Lemma 4.1. *If C is a w-secure code then*

$$C_1 \cap \cdots \cap C_r = \emptyset \;\Rightarrow\; desc_w(C_1) \cap \cdots \cap desc_w(C_r) = \emptyset$$

for all coalitions $C_1, \cdots, C_r$ of at most w colluders each.

It seems that secure codes provide a good solution to the problem of collusion. Unfortunately, when $w > 1$, w-secure codes do not exist.

Theorem 4.1. *For $w \geqslant 2$ there are no w-secure codes.*

Proof. Obviously, it is sufficient to show that there are no 2-secure codes. Let $c^{(1)}, c^{(2)}, c^{(3)}$ be three distinct legal codewords assigned to users u_1, u_2, u_3, respectively. Define the majority word $M = MAJ\left(c^{(1)}, c^{(2)}, c^{(3)}\right)$ by

$$M_i = \begin{cases} c_i^{(1)}, & \text{if } c_i^{(1)} = c_i^{(2)} \text{ or } c_i^{(1)} = c_i^{(3)} \\ c_i^{(2)}, & \text{if } c_i^{(2)} = c_i^{(3)} \\ ?, & \text{otherwise.} \end{cases}$$

One can readily verify that the majority word M belongs to $desc_2\{u_1, u_2\}$, $desc_2\{u_1, u_3\}$, and $desc_2\{u_2, u_3\}$, simultaneously. However, the intersection of the coalitions is empty. Hence, by Lemma 4.1, the 2-secure code cannot exist. ∎

The proof of the theorem shows that if a coalition employs the "majority" strategy it is guaranteed to defeat all fingerprinting codes. Based on above argument and Proposition 3.1 the existence of IPP and TA codes is denied. This forces us to weaken our requirements for fingerprinting schemes. In the following section,

we intend to allow the distributor to make some random choices when embedding the codewords in the products. The point is that the random choices will be kept hidden from the users. This enables us to construct codes which will capture a member of the guilty coalition with sufficiently high probability.

4.2 Probabilistic Traitor Tracing

Probabilistic traitor tracing (**PTT**) is much more efficient in most of the cases. In this scheme, we need not to identify colluders who have absolutely committed crime.[‖] Instead, we treat a couple of might-be-colluders as suspects, and compute the probability that they might be colluders. This may not deterministically tell us who is guilty for the first time. However, after several times of identification, some pirates will become more and more suspicious by accumulating their probabilities of being guilty. Such a strategy works particularly well for the applications such as pay-per-view movies that call for iterative retrievals of data.

Suppose a coalition C of w users creates an illegal copy of an object. Fingerprinting schemes that enable the capture of a member of the coalition C with probability at least $1-\epsilon$ are called w-secure codes with ϵ error. Namely, Pr $[\mathcal{A}(x) \in C] > 1-\epsilon$. In other words, The traitor tracing algorithm $\mathcal{A}$ on input x outputs a member of the coalition C that generated the codeword x with high probability. To do so, we intend to allow the distributor to make some random choices when embedding the codewords in the objects. Our point is that the random choices will be kept hidden from the users.

We begin by considering an (N, n)-code which is n-secure with ϵ-error for any $\epsilon > 0$. Let c_m be a column of height n in which the first m bits are 1 and the rest are 0. The code $C\,(N = d(n-1), n)$ consists of all columns $c_1, \ldots, c_{n-1}$, each duplicated d times. The amount of duplication determines the error probability ϵ.

Example 4.1. *The code $C(16, 5)$ for five users A, B, C, D, E is*

[‖] More generally speaking, we say they committed crime with probability 1.

	$\overbrace{\quad}^{B_1}$	$\overbrace{\quad}^{B_2}$	$\overbrace{\quad}^{B_3}$	$\overbrace{\quad}^{B_4}$
A :	1111	1111	1111	1111
B :	0000	1111	1111	1111
C :	0000	0000	1111	1111
D :	0000	0000	0000	1111
E :	0000	0000	0000	0000

An intuitive traitor tracing strategy is: if any of the first three positions of a pirated codeword is 1, then we know A must belong to the coalition. If we look at the other direction, we have that if any of the last three positions of a pirated codeword is 0, then we know E must belong to the coalition. If A and B collude, C, D, and E are safe from being framed. However, if A and E collude, the descendance of A and E could jeopardize legal users of B, C, and D. Nevertheless, this is very unlikely because A and E differ in 16 places and the probability for A and E to frame B, C, or D is barely $\left(\frac{1}{2}\right)^{16} \approx 10^{-5}$. This gives a heuristics for probabilistic traitor tracing.

Consider, if B is innocent, then what A, C, D, E could detect in the first eight positions is totally indifferent, namely, either 11111111 or 00000000. If some of A, C, D, or E collude, then the number of $0's$ and $1's$ should be evenly distributed in B_1 and B_2. If the number of $1's$ tends to appear more in B_2 rather than in B_1, then we deduce that B is highly suspicious.

Let $w^{(1)}, \ldots, w^{(n)}$ denote the codewords of $C(N, n)$. Before the distributor embeds the codewords of $C(N, n)$ in an objects he picks a permutation π as random as possible. User u_i's copy of the object will be fingerprinted using the word $\pi w^{(i)}$. Note that the same permutation π is used for all users. The point is that π will be kept hidden from the user. Keeping the permutation hidden from the users is equivalent to hiding the information of which mark in the object encodes which bit in the code. This simple technique will be shown to be effective to overcome the barrier of unreadable marks.

Before going to the construction, we introduce some notation:

1. Let B_m be the set of all bit positions in which the users see columns of type c_m. That is, B_m is the set of all bit positions in which the first m users see a 1 and the rest see a 0. The number of elements in B_m is d.

2. For $2 \leqslant s \leqslant n-1$ define $R_s = B_{s-1} \cup B_s$.

3. For a binary string x, let $weight(x)$ denote the number of $1's$ as a binary case of Hamming weight defined in Section 2.1.2.

Theorem 4.2 (Boneh and Shaw [8]). *For $n \geqslant 3$ and $\epsilon > 0$ let $d = 2n^2 \log \frac{2n}{\epsilon}$. The fingerprinting scheme $C(N, n)$ is n-secure with ϵ-error.*

The argument has been literally treated above, but we formalize the language here. The length of this code is $d(n-1) = \mathcal{O}\left(n^3 \log \frac{n}{\epsilon}\right)$ Intuitively, suppose user s is NOT a member of the coalition C_0 which produced the word x. The hidden permutation π prevents the coalition from knowing which marks represent which bits in the code $C(N, n)$. The only information the coalition has is the value of the marks it can detect. Observe that without user s a coalition sees exactly the same values for all bit positions $i \in R_s$. Hence, for a bit position $i \in R_s$, the coalition C_0 cannot tell if i lies in B_s or in B_{s-1}. This means that whichever strategy they use to set the bits of $x|_{R_s}$, the $1's$ in $x|_{R_s}$ will be roughly evenly distributed between $x|_{B_s}$ and $x|_{B_{s-1}}$ with high probability. As a result, if the $1's$ in $x|_{R_s}$ are not evenly distributed then, with high probability, user s is a member of the coalition that generated x.

Algorithm for probabilistic traitor tracing will be stated accordingly. The input codeword x found in the illegal copy may contain some unreadable marks, call it "?". As a convention these bits are set to "0" before the word x is feed into the algorithm.

INPUT: $x \in \{0,1\}^N$.

AIM: Find a subset of the coalition that produced x.

Algorithm:

1. If $weight\left(x|_{B_1}\right) > 0$ then output "User 1 is guilty."
2. If $weight\left(x|_{B_{n-1}}\right) < d$ then output "User n is guilty."
3. For s from 2 to $n-1$ do:
   Let $k = weight\left(x|_{R_s}\right)$.
   If $weight\left(x|_{B_{s-1}}\right) < \frac{k}{2} - \sqrt{\frac{k}{2} \log \frac{2n}{\epsilon}}$, then output "User s is guilty."

The correctness of algorithm rely on the following theorem.

Theorem 4.3. *Consider the code $C\,(N = d(n-1), n)$ where $d = 2n^2 \log \frac{2n}{\epsilon}$. Let S be the set of users which is declared as guilty on input x. Then with probability at least $1-\epsilon$, the set S is a subset of the coalition C_0 that produced x.*

Before the proof of the theorem we introduce two preliminary lemmas.

Lemma 4.2 (Chernoff Bound)**.** *Let X be a binomial random variable over k experiments with success probability $1/2$. Then,*

$$Pr\left[X - \frac{k}{2} < a\right] \leqslant e^{-2a^2/k}$$

The proof can be found in standard textbooks on probability theory.

Lemma 4.3. *Let Y follows a hyper-geometric distribution:*

$$Pr\left[Y = r\right] = \frac{\binom{d}{r}\binom{d}{k-r}}{\binom{2d}{k}}.$$

Let X follows a binomial distribution with success probability $1/2$:

$$Pr\left[X = r\right] = \binom{k}{r}\left(\frac{1}{2}\right)^k.$$

Then, $Pr\left[Y = r\right] \leqslant 2Pr\left[X = r\right]$

Proof. For the sake of brevity assume k is even. (The case for k odd is similar.)

$$\begin{aligned}
\Pr[Y=r] &= \frac{\binom{d}{r}\binom{d}{k-r}}{\binom{2d}{k}} \\
&= \binom{k}{r}\frac{d(d-1)\cdots(d-r+1)d(d-1)\cdots(d-k+r+1)}{2d(2d-1)\cdots(2d-k+1)} \\
&\leqslant \binom{k}{r}2^{-k}\frac{d^2(d-1)^2\cdots\left(d-\frac{k-2}{2}\right)^2}{d(d-1)\cdots\left(d-\frac{k-1}{2}\right)} \\
&= \binom{k}{r}2^{-k}\frac{d(d-1)\cdots\left(d-\frac{k-2}{2}\right)}{\left(d-\frac{1}{2}\right)\left(d-\frac{3}{2}\right)\cdots\left(d-\frac{k-1}{2}\right)} \\
&= \binom{k}{r}2^{-k}\frac{d(d-1)\cdots\left(d-\frac{k-2}{2}\right)}{\left(d-1+\frac{1}{2}\right)\left(d-2+\frac{1}{2}\right)\cdots\left(d-\frac{k-2}{2}+\frac{1}{2}\right)\left(d-\frac{k-1}{2}\right)} \\
&\leqslant \binom{k}{r}2^{-k}\frac{d}{\left(d-\frac{k-1}{2}\right)} \\
&\leqslant \binom{k}{r}2^{-k}\cdot 2 \\
&= 2\Pr[X=r]
\end{aligned}$$

Note that the last inequality follows since $k \leqslant d$. ∎

The proof of Theorem 4.3 is now as follows:

Proof. Suppose user 1 was declared guilty, i.e., $1 \in S$. Then $weight\,(x|_{B_1}) > 0$. This tells us that user 1 must be a member of C_0 (otherwise, the bits in B_1 would appear undistinguishable for C_0). Similarly, if $n \in S$ then $n \in C_0$.

Suppose the algorithm declared user $1 < s < n$ as guilty. We show that the probability that s is not guilty is at most $\frac{\epsilon}{n}$. This will show that the probability that there exists a user in S which is not guilty is at most ϵ.

Let s be an innocent user, i.e., $s \notin C_0$. As was discussed above, this means that the coalition C_0 cannot distinguish between the bit positions in R_s. Because the permutation π was chosen uniformly at random from the set of all permutations, the $1's$ in $x|_{R_s}$ may be regarded as being randomly placed in $x|_{R_s}$. Let $k = weight\,(x|_{R_s})$. Define Y to be a random variable which counts the number of

$1's$ in $x|_{B_{s-1}}$ given that $x|_{R_s}$ contains k $1's$. For any integer $r,\ 0 \leqslant r \leqslant k$:

$$\Pr[Y = r] = \Pr\left[weight(x|_{B_{s-1}}) = r \mid weight(x|_{R_s}) = k\right] = \frac{\binom{d}{r}\binom{d}{k-r}}{\binom{2d}{k}}$$

follows a hyper-geometric distribution where $d = 2n^2 \log \frac{2n}{\epsilon}$ is the size of the block. The expectation of Y is $\frac{k}{2}$. To bound the probability that s was pronounced guilty we need to bound

$$\Pr\left[Y < \frac{k}{2} - \sqrt{\frac{k}{2}\log\frac{2n}{\epsilon}}\right]$$

from above. This can be done by comparing Y to an appropriate binomial random variable.

Let X be a binomial random variable over k experiments with success probability $\frac{1}{2}$. Lemma 4.3 tells us that for any r we have that $\Pr[Y = r] \leqslant 2\Pr[X = r]$. This means that for any $a > 0$

$$\Pr\left[Y - \frac{k}{2} < a\right] \leqslant 2\Pr\left[X - \frac{k}{2} < a\right] \leqslant 2e^{-2a^2/k}$$

where the last inequality follows from the standard Chernoff bound of Lemma 4.2. Plugging in $a = \sqrt{\frac{k}{2}\log\frac{2n}{\epsilon}}$ leads to

$$\Pr\left[Y < \frac{k}{2} - \sqrt{\frac{k}{2}\log\frac{2n}{\epsilon}}\right] \leqslant 2e^{-\log\frac{2n}{\epsilon}} = \frac{\epsilon}{n}$$

Hence, if user s is innocent then the probability of her being declared guilty is at most $\frac{\epsilon}{n}$. This also means the probability that some innocent user will be declared guilty is at most ϵ, as desired. ∎

Note that the code size is always smaller than the code length by a factor of d here, meaning a poor code size. This problem can be overcome with the concatenation method discussed in [8] in order to increase the code size and hence accommodate more users. We provide the sketch concept here. Recall in Section 2.1.5 the definition of code composition. Let $C'(N', n')$ be an outer code over an alphabet size n, with code size n' and code length N', where the codewords are chosen independently and uniformly at random. The idea is to compose our n-secure inner code $C(N, n)$ with the outer code $C'(N', n')$. Then the concatenated

code will contain n' codewords and has length $N'N = N'd(n-1)$. It is made up of N' copies of $C(N, n)$. The point is that the codewords of the code C' will be kept secret from the users. This is in addition to keeping hidden the N' permutations used when embedding the N' copies of $C(N, n)$ in the products. A traitor tracing algorithm is also provided for this scheme which is similar to the original one. Moreover, N and n can be chosen in such a way that n is exponential in N. For more details we refer the reader to their paper [8]. In the next chapter we will concentrate on the construction of secure frameproof codes.

Chapter 5

Constructions of SFP Codes

This chapter discusses various constructions that meet the requirement of secure-frameproof property. The constructions can be classified into two classes. One of them is called direct construction which will be studied in the first half of this chapter. In such scheme, we construct directly without any help of previous existential results. The other is recursive construction which will be investigated in the second half of this chapter. Given a code[‡‡] satisfying certain properties the recursive construction augments it to longer codewords and larger code size satisfying the original properties as well.

Part I: Direct Construction

5.1 Hadamard Matrices and Jacobsthal Matrices

Definition: A Hadamard matrix is an $n \times n$ real matrix H which satisfies $HH_T = nI$. The name derives from a theorem of Hadamard.

Theorem 5.1. *Let $X = (x_{ij})$ be an $n \times n$ real matrix whose entries satisfy $|x_{ij}| \leqslant 1$ for all i and j. Then $|det(X)| \leqslant n^{n/2}$. Equality holds if and only if X is a Hadamard matrix.*

[‡‡] We call it the initial seed.

Let $x_1, x_2, \ldots, x_n$ be the rows of X. By Euclidean geometry, $|det(X)|$ is the volume of the parallelepiped with sides $x_1, x_2, \ldots, x_n$; namely,

$$|det(X)| \leqslant |x_1| \cdot |x_2| \cdots |x_n|$$

where $|x_i|$ is the Euclidean length of x_i; equality holds if and only if $x_1, x_2, \ldots, x_n$ are mutually perpendicular. By assumption,

$$|x_i| = \left(x_{i1}^2 + x_{i2}^2 + \cdots + x_{in}^2\right)^{1/2} \leqslant n^{1/2},$$

with equality if and only if $|x_{ij}| = 1$ for all j.

Subsequently, we focus on Hadamard matrices with all entries ± 1.

For which orders n do Hadamard matrices exist? There is a well-known necessary condition:

Theorem 5.2. *If a Hadamard matrix of order n exists, then $n = 1$ or 2 or $n \equiv 0 \pmod 4$.*

To see this, we observe first that changing the sign of every entry in a column of a Hadamard matrix gives another Hadamard matrix. So changing the signs of all columns for which the entry in the first row is $-$, we may assume that all entries in the first row are $+$. (We abbreviate $+1$ and -1 to $+$ and $-$ respectively.)

$$\begin{array}{cccc} \overbrace{+\cdots+}^{a} & \overbrace{+\cdots+}^{a} & \overbrace{+\cdots+}^{a} & \overbrace{+\cdots+}^{a} \\ +\cdots+ & +\cdots+ & -\cdots- & -\cdots- \\ +\cdots+ & -\cdots- & +\cdots+ & -\cdots- \end{array}$$

Because every other row is orthogonal to the first, we see that each further row has m entries $+$ and $-$, where $n = 2m$. Moreover, if $n > 2$, the first three rows are displayed in the above figure with $n = 4a$. The most important open question in the theory of Hadamard matrices is that of existence (In other words, whether or not the above necessary condition could serve as a sufficient condition is not known).

Conjecture 5.1. *A Hadamard matrix of order $4n$ exists for every positive integer n.*

The simplest construction comes from James Joseph Sylvester.

Theorem 5.3. *Let H be a Hadamard matrix of order n. Then the partitioned matrix*

$$\begin{bmatrix} H & H \\ H & -H \end{bmatrix}$$

is a Hadamard matrix of order $2n$.

This observation can be applied recursively and leads to the following series of matrices.

$$H_1 = \begin{bmatrix} 1 \end{bmatrix}$$

$$H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$H_4 = \begin{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} & \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \\ \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} & -\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \cdots$$

In this manner, Sylvester constructed Hadamard matrices of order 2^n for every non-negative integer n. Sylvester's matrices have a number of special properties. They are symmetric. The elements in the first column and the first row are all positive. The elements in all the other rows and columns are evenly divided between positive and negative.

Raymond Paley later showed how to construct a Hadamard matrix of order $q + 1$ where q is any prime power which is congruent to 3 modulo 4. He also constructed matrices of order $2(q + 1)$ for prime powers q which are congruent to 1 modulo 4. His method uses finite fields.

Let q be a prime power congruent to 3 modulo 4. Recall that in the field GF(q), half of the nonzero elements are quadratic residues, and half are quadratic non-residues. The quadratic character of GF(q) is defined as:

$$\chi(x) = \begin{cases} 0 & \text{if } x = 0; \\ +1 & \text{if } x \text{ is a quadratic residue}; \\ -1 & \text{if } x \text{ is a quadratic non-residue}. \end{cases}$$

Definition: Let A be a matrix whose rows and columns are indexed by elements of GF(q), and has entries as $a_{xy} = \chi(y - x)$. Then, A is skew-symmetric, with zero diagonal and ± 1 elsewhere. Such matrix A is then called **Jacobsthal matrix**.

Theorem 5.4. *If we replace the diagonal zeros by $-1s$ in the Jacobsthal matrix and augment it by a new row and a new column all of entries 1, we obtain a Hadamard matrix of order $q+1$ called Hadamard matrix of Paley type.*

$$H = \begin{bmatrix} 1 & 1 \\ 1 & A - I \end{bmatrix}$$

Example 5.1. *For $p = 7$, we obtain the following matrix:*

$$A = \begin{pmatrix} 0 & 1 & 1 & -1 & 1 & -1 & -1 \\ -1 & 0 & 1 & 1 & -1 & 1 & -1 \\ -1 & -1 & 0 & 1 & 1 & -1 & 1 \\ 1 & -1 & -1 & 0 & 1 & 1 & -1 \\ -1 & 1 & -1 & -1 & 0 & 1 & 1 \\ 1 & -1 & 1 & -1 & -1 & 0 & 1 \\ 1 & 1 & -1 & 1 & -1 & -1 & 0 \end{pmatrix}$$

A normalized Hadamard matrix H of order $q+1$ of Paley type is now given as follows:

$$H = \begin{pmatrix} 1 & 1 \\ 1 & A - I \end{pmatrix}.$$

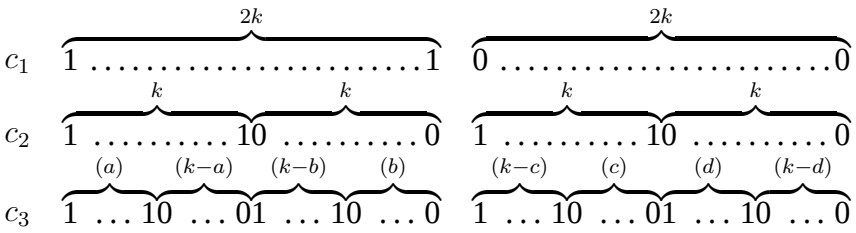
Example 5.2. *For $p = 7$, we obtain the following matrix over GF(3) by replacing -1 by 2 from A:*

$$A' = \begin{pmatrix} 0 & 1 & 1 & 2 & 1 & 2 & 2 \\ 2 & 0 & 1 & 1 & 2 & 1 & 2 \\ 2 & 2 & 0 & 1 & 1 & 2 & 1 \\ 1 & 2 & 2 & 0 & 1 & 1 & 2 \\ 2 & 1 & 2 & 2 & 0 & 1 & 1 \\ 1 & 2 & 1 & 2 & 2 & 0 & 1 \\ 1 & 1 & 2 & 1 & 2 & 2 & 0 \end{pmatrix}$$

Let H_{4k} be any Hadamard matrix of order $4k$ when $+1s$ are replaced by $0s$ and $-1s$ by $1s$.

Theorem 5.5 (Encheva & Cohen [17])**.** *H_{4k} is a binary $2-SFP\,(N, n)$ where $N = n = 4k$.*

Proof. We show that there is a column like $(0011)^\top$ or $(1100)^\top$ for any $c_1, c_2, c_3, c_4 \in H_{4k}$. We consider a normalized Hadamard matrix where the first row is the all $1s$ and firstly assume none of c_1, c_2, c_3, c_4 is the all $1s$ codeword. Suppose the contrary. The supports of c_1, c_2, c_3 may be generalized as



Because none of c_1, c_2, c_3, c_4 is the all $1s$ codeword, they should contain equal number of $0s$ as $1s$, and every two rows should coincide in half of the positions and differ in the other half positions. Therefore, c_3 should contain $2k$ $1s$, yielding:

$$a + (k - b) + (k - c) + d = 2k$$

Since c_3 should coincide with c_2 in exactly $2k$ positions, we have that:

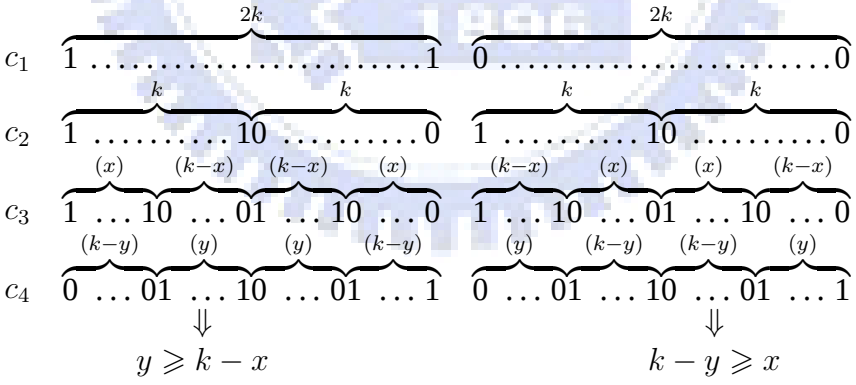
$$a + b + (k - c) + (k - d) = 2k$$

Again since c_3 should coincide with c_1 in exactly $2k$ positions, we have that:

$$a + b + c + d = 2k$$

A routine calculation leads to $a = b = c = d$.

Accordingly, the support of c_1, c_2, c_3, c_4 is given by



We deduce that

$$x + y = k$$

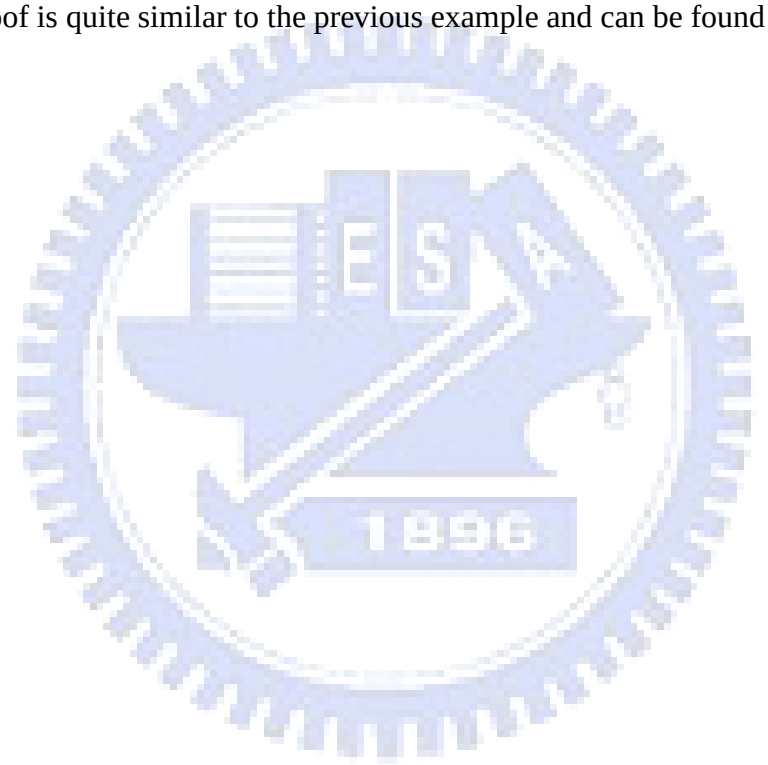
which gives the following catastrophic patterns:

$$
\begin{array}{ll}
 & \overbrace{1\ \ldots\ 10}^{(x)}\ \overbrace{\ldots\ 01}^{(k-x)}\ \overbrace{\ldots\ 10}^{(k-x)}\ \overbrace{\ldots\ 0}^{(x)}\quad \overbrace{1\ \ldots\ 10}^{(k-x)}\ \overbrace{\ldots\ 01}^{(x)}\ \overbrace{\ldots\ 10}^{(x)}\ \overbrace{\ldots\ 0}^{(k-x)} \\
c_3 & \\
 & \overbrace{0\ \ldots\ 01}^{(x)}\ \overbrace{\ldots\ 10}^{(k-x)}\ \overbrace{\ldots\ 01}^{(k-x)}\ \overbrace{\ldots\ 1}^{(x)}\quad \overbrace{0\ \ldots\ 01}^{(k-x)}\ \overbrace{\ldots\ 10}^{(x)}\ \overbrace{\ldots\ 01}^{(x)}\ \overbrace{\ldots\ 1}^{(k-x)} \\
c_4 &
\end{array}
$$

However, this is impossible because c_3 and c_4 should coincide in $2k$ positions. Moreover, if one of c_1, c_2, c_3, c_4 is the all $1s$ codeword then it is even easier for them to exhibit the $2 - SFP$ property. Hence, H_{4k} is $2 - SFP$. ∎

Theorem 5.6. *Jacobsthal matrices generate* $2 - SFP$ *over GF(3).*

The proof is quite similar to the previous example and can be found in [17].



5.2 The Subsets Method

Another direct construction which employs the properties of subsets was proposed by Tonien and Safavi-Naini [34].

Let (k) be the set $\{1, 2, \ldots, k\}$. By $(k)_t$ we denote the set of all subsets of (k) which contain exactly t elements.

With parameters k, t, r, consider the matrix $M_{t,r}(k)$ whose rows are labeled by elements of $(k)_t$ and columns are labeled by elements of $(k)_r$. For $U \in (k)_t, V \in (k)_r$, the entry at the row U and column V of the matrix $M_{t,r}(k)$ is $|U \cap V|$. The code $C_{t,r}(k)$ is composed by the rows of the matrix $M_{t,r}(k)$. Without ambiguity, we identify a codeword of $C_{t,r}(k)$ with a set $U \in (k)_t$ and a position with a set $V \in (k)_r$. By definition, the symbol of the codeword U at the position V is $U_V = |U \cap V|$.

The code $C_{t,\leqslant r}(k)$ can be defined in a similar way. Code $C_{t,\leqslant r}(k)$ is depicted by the matrix $M_{t,\leqslant r}(k)$ whose rows and columns are labeled by elements of the sets $(k)_t$ and $(k)_{\leqslant r}$ respectively. For $U \in (k)_t$ and $V \in (k)_{\leqslant r}$, the symbol of the codeword U at the position V is $U_V = |U \cap V|$.

Codes $C^*_{t,r}(k)$ and $C^*_{t,\leqslant r}(k)$ are binary codes. They are constructed the same as code $C_{t,r}(k)$ and $C_{t,\leqslant r}(k)$ except that the symbol of the codeword U at the position V is $U_V = |U \cap V| \pmod 2$.

Example 5.3. *Codes of $C_{3,2}(5)$, $C^*_{3,2}(5)$, $C_{3,\leqslant 2}(4)$, and $C^*_{3,\leqslant 2}(4)$ are shown below:*

$C_{3,2}(5)$	$\{1,2\}$	$\{1,3\}$	$\{1,4\}$	$\{1,5\}$	$\{2,3\}$	$\{2,4\}$	$\{2,5\}$	$\{3,4\}$	$\{3,5\}$	$\{4,5\}$
$\{1,2,3\}$	2	2	1	1	2	1	1	1	1	0
$\{1,2,4\}$	2	1	2	1	1	2	1	1	0	1
$\{1,2,5\}$	2	1	1	2	1	1	2	0	1	1
$\{1,3,4\}$	1	2	2	1	1	1	0	2	1	1
$\{1,3,5\}$	1	2	1	2	1	0	1	1	2	1
$\{1,4,5\}$	1	1	2	2	0	1	1	1	1	2
$\{2,3,4\}$	1	1	1	0	2	2	1	2	1	1
$\{2,3,5\}$	1	1	0	1	2	1	2	1	2	1
$\{2,4,5\}$	1	0	1	1	1	2	2	1	1	2
$\{3,4,5\}$	0	1	1	1	1	1	1	2	2	2

$C^*_{3,2}(5)$	$\{1,2\}$	$\{1,3\}$	$\{1,4\}$	$\{1,5\}$	$\{2,3\}$	$\{2,4\}$	$\{2,5\}$	$\{3,4\}$	$\{3,5\}$	$\{4,5\}$
$\{1,2,3\}$	0	0	1	1	0	1	1	1	1	0
$\{1,2,4\}$	0	1	0	1	1	0	1	1	0	1
$\{1,2,5\}$	0	1	1	0	1	1	0	0	1	1
$\{1,3,4\}$	1	0	0	1	1	1	0	0	1	1
$\{1,3,5\}$	1	0	1	0	1	0	1	1	0	1
$\{1,4,5\}$	1	1	0	0	0	1	1	1	1	0
$\{2,3,4\}$	1	1	1	0	0	0	1	0	1	1
$\{2,3,5\}$	1	1	0	1	0	1	0	1	0	1
$\{2,4,5\}$	1	0	1	1	1	0	0	1	1	0
$\{3,4,5\}$	0	1	1	1	1	1	1	0	0	0

$C_{3,\leqslant 2}(4)$	$\{1\}$	$\{2\}$	$\{3\}$	$\{4\}$	$\{1,2\}$	$\{1,3\}$	$\{1,4\}$	$\{2,3\}$	$\{2,4\}$	$\{3,4\}$
$\{1,2,3\}$	1	1	1	0	2	2	1	2	1	1
$\{1,2,4\}$	1	1	0	1	2	1	2	1	2	1
$\{1,3,4\}$	1	0	1	1	1	2	2	1	1	2
$\{2,3,4\}$	0	1	1	1	1	1	1	2	2	2

$C^*_{3,\leqslant 2}(4)$	$\{1\}$	$\{2\}$	$\{3\}$	$\{4\}$	$\{1,2\}$	$\{1,3\}$	$\{1,4\}$	$\{2,3\}$	$\{2,4\}$	$\{3,4\}$
$\{1,2,3\}$	1	1	1	0	0	0	1	0	1	1
$\{1,2,4\}$	1	1	0	1	0	1	0	1	0	1
$\{1,3,4\}$	1	0	1	1	1	0	0	1	1	0
$\{2,3,4\}$	0	1	1	1	1	1	1	0	0	0

The following two theorems are used to establish the secure-frameproof property of $C_{t,r}(k)$ and $C_{t,\leqslant r}(k)$.

Theorem 5.7. *If $S_1, S_2, S_3,$ and S_4 are arbitrary subsets of (k) such that*

$$S_i \not\subset S_j \text{ and } S_j \not\subset S_i \text{ for all } i \in \{1,2\} \text{ and } j \in \{3,4\}$$

then there exists an elements $V \in (k)_{\leqslant 3}$ such that the following two sets

$$\{|V \cap S_1| \bmod 2, |V \cap S_2| \bmod 2\} \text{ and } \{|V \cap S_3| \bmod 2, |V \cap S_4| \bmod 2\}$$

are disjoint.

This further implies the following.

Corollary 5.1. *If $S_1, S_2, S_3,$ and S_4 are arbitrary subsets of (k) such that*

$$S_i \not\subset S_j \text{ and } S_j \not\subset S_i \text{ for all } i \in \{1,2\} \text{ and } j \in \{3,4\}$$

then there exists an elements $V \in (k)_{\leqslant 3}$ such that the following two sets

$$\{|V \cap S_1|, |V \cap S_2|\} \text{ and } \{|V \cap S_3|, |V \cap S_4|\}$$

are disjoint.

The proof of Theorem 5.7 and Corollary 5.1 can be found in [34] which exhaustively investigates all of the possibilities of distribution of $0s$ and $1s$. Based on the above fact, we have the following explicit constructions.

Theorem 5.8. *For any $k > 4$, $C_{2,2}(k)$ is a ternary $2-SFP$ with code size $n = \binom{k}{2}$ and code length $N = \binom{k}{2}$.*

Proof. We indicate a proof which is easier than the original found in the paper of Tonien and Safavi-Naini. First it is sufficient to show that if $C_{2,2}(8)$ is $2 - SFP$ then the same is true as well for $C_{2,2}(k)$ for all $k \geqslant 8$.

Therefore, whenever $k \geqslant 8$, the submatrix of dimension $\binom{8}{2} \times \binom{8}{2}$ out of the matrix of dimension $\binom{k}{2} \times \binom{k}{2}$ will always have the $2 - SFP$ property. Thus the conclusion follows. Now in order to finish the proof, we still have to verify the $2 - SFP$ property for $k = 5, 6, 7, 8$, but this can be done either by hand or computers. ∎

Theorem 5.9. *For any $k > t$, $C_{t,\leqslant 3}(k)$ is a quaternary $2 - SFP$ with code size $n = \binom{k}{t}$ and code length $N = \binom{k}{1} + \binom{k}{2} + \binom{k}{3} = \frac{1}{6}k(k^2 + 5)$.*

Theorem 5.10. *For any $k \geqslant 4t + r - 1$ and $r \geqslant 3$, $C_{t,r}(k)$ is a $(min\{t, r\} + 1) - ary\ 2 - SFP$ with code size $n = \binom{k}{t}$ and code length $N = \binom{k}{r}$.*

Proof. For any four distinct elements S_1, S_2, S_3, S_4 of $(k)_t$, by Corollary 5.1, there exists $V \in (k)_{\leqslant 3}$ such that the two sets $\{|V \cap S_1|, |V \cap S_2|\}$ and $\{|V \cap S_3|, |V \cap S_4|\}$ are disjoint. Since $k \geqslant 4t+r-1 = |S_1|+|S_2|+|S_3|+|S_4|+r-1$, we can add more elements from the set $(k) \setminus (S_1 \cup S_2 \cup S_3 \cup S_4)$ to V and obtain a set $V' \in (k)_r$. We have $V \cap S_i = V' \cap S_i$, and thus, the two sets $\{|V' \cap S_1|, |V' \cap S_2|\}$ and $\{|V' \cap S_3|, |V' \cap S_4|\}$ are disjoint. This proves that the code $C_{t,r}(k)$ is a $2-SFP$. ∎

Combining the results and Theorem 5.7, 5.9, and 5.10, we have the following binary codes.

Theorem 5.11. *For any $k > t$, $C^*_{t,\leqslant 3}(k)$ is a binary $2 - SFP$ with code size $n = \binom{k}{t}$ and code length $N = \binom{k}{1} + \binom{k}{2} + \binom{k}{3} = \frac{1}{6}k(k^2 + 5)$.*

Theorem 5.12. *For any $k \geqslant 4t + r - 1$ and $r \geqslant 3$, $C^*_{t,r}(k)$ is a binary $2 - SFP$ with code size $n = \binom{k}{t}$ and code length $N = \binom{k}{r}$.*

Note that the "Subsets Method" which is capable of generating exponential code sized $2 - SFP$ is much better than the "Hadamard Method" which gives

$2-SFP$ codes with code size only the same as the code length. In order to demonstrate this, we take advantage of Stirling's formula

$$k! \sim \sqrt{2\pi k}\left(\frac{k}{e}\right)^k.$$

Consider the binary code derived in Theorem 5.12, the maximum code size is for $t = \lfloor \frac{k}{2} \rfloor$,

$$\begin{aligned} n &= \binom{k}{\frac{k}{2}} \\ &= \frac{k!}{(\frac{k}{2})!\,(\frac{k}{2})!} \\ &\sim \frac{\sqrt{2\pi k}\,(\frac{k}{e})^k}{\sqrt{2\pi\frac{k}{2}}\left(\frac{k/2}{e}\right)^{k/2}\sqrt{2\pi\frac{k}{2}}\left(\frac{k/2}{e}\right)^{k/2}} \\ &= 2^k\sqrt{\frac{2}{\pi k}} \end{aligned}$$

which is exponential with respect to the code length N. Moreover, the minimum code length is for $r = 3$, $N = \frac{1}{6}k(k^2+5)$. Therefore, the maximum code rate can be achieved as:

$$\begin{aligned} R &= N^{-1}\log_q n \\ &\sim \left(\frac{1}{6}k(k^2+5)\right)^{-1}\log_2\left(2^k\sqrt{\frac{2}{\pi k}}\right) \\ &\leqslant \frac{6k}{k(k^2+5)}\log_2\left(\sqrt{\frac{2}{\pi k}}\right) \end{aligned}$$

which tends to zero as k goes to infinity. Nevertheless, we have $N = \mathcal{O}\left((\log n)^3\right)$. Later on, we will introduce better codes with positive code rates where again the code size is exponential with respect to the code length. Also, up to now we have only investigated $2-SFP$ codes, we will show $w-SFP$ codes for $w > 2$ in the sequel.

Part II: Recursive Construction

5.3 Concatenation Method

Recall that in Section 2.1.5 of second chapter the concatenation of two codes is defined. The construction of the section employs concatenation of two codes: normally a longer outer code B and a shorter inner code A. The concatenation is usually used to increase the code length and the separating weights, λ_w, defined in Section 2.4.1.

Theorem 5.13. *Let $u \geqslant v$, C_1 be a $u - SFP$ code with λ_u and C_2 a $v - SFP$ code with λ_v, then the concatenated $C_1 \bigstar C_2$ is a $v - SFP$ with a new separating weight $\lambda' \geqslant \lambda_u \lambda_v$.*

Proof. C_2 is an $v - SFP$ outer code and the symbols of C_2 are replaced by a one-to-one mapping by codewords of C_1, so if any two coalitions of C_2 of size at most v is separable, then any two coalitions of $C_1 \bigstar C_2$ of size at most v is separable as well. The separating weight of outer code is λ_v, and for each separated position of C_2 the inner code C_1 itself separates in at least λ_u positions by definition. Hence, the new separating weight λ' is at least $\lambda_u \lambda_v$. ∎

Example 5.4. *Let A and B be the following code:*

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 2 & 2 \\ 2 & 2 & 2 & 2 & 1 \\ 0 & 1 & 2 & 1 & 0 \end{bmatrix}_{4\times 5} \qquad B = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 2 & 2 & 2 & 2 & 0 \\ 3 & 3 & 3 & 3 & 0 \\ 3 & 2 & 1 & 0 & 1 \\ 2 & 3 & 0 & 1 & 1 \\ 1 & 0 & 3 & 2 & 1 \\ 0 & 1 & 2 & 3 & 1 \\ 1 & 3 & 2 & 0 & 2 \\ 0 & 2 & 3 & 1 & 2 \\ 3 & 1 & 0 & 2 & 2 \\ 2 & 0 & 1 & 3 & 2 \\ 2 & 1 & 3 & 0 & 3 \\ 3 & 0 & 2 & 1 & 3 \\ 0 & 3 & 1 & 2 & 3 \\ 1 & 2 & 0 & 3 & 3 \end{bmatrix}_{16\times 5}$$

It is easy to check that A *is a* $3-SFP$ *code and* B *a* $2-SFP$ *code. Define the following mapping of alphabet symbols of* B *to the rows of* A*:*

$$\theta \ : \ \begin{cases} 0 \ \mapsto \ (00001) \\ 1 \ \mapsto \ (11122) \\ 2 \ \mapsto \ (22221) \\ 3 \ \mapsto \ (01210) \end{cases}$$

Applying this mapping to B *we obtain a code* $A\bigstar B$ *with parameters* $N = 16, n = 25, q = 3$*:*

$$A\bigstar B = \begin{bmatrix} 00001 & 00001 & 00001 & 00001 & 00001 \\ 11122 & 11122 & 11122 & 11122 & 00001 \\ 22221 & 22221 & 22221 & 22221 & 00001 \\ 01210 & 01210 & 01210 & 01210 & 00001 \\ 01210 & 22221 & 11122 & 00001 & 11122 \\ 22221 & 01210 & 00001 & 11122 & 11122 \\ 11122 & 00001 & 01210 & 22221 & 11122 \\ 00001 & 11122 & 22221 & 01210 & 11122 \\ 11122 & 01210 & 22221 & 00001 & 22221 \\ 00001 & 22221 & 01210 & 11122 & 22221 \\ 01210 & 11122 & 00001 & 22221 & 22221 \\ 22221 & 00001 & 11122 & 01210 & 22221 \\ 22221 & 11122 & 01210 & 00001 & 01210 \\ 01210 & 00001 & 22221 & 11122 & 01210 \\ 00001 & 01210 & 11122 & 22221 & 01210 \\ 11122 & 22221 & 00001 & 01210 & 01210 \end{bmatrix}_{16\times 25}$$

A more practical application of the concatenation method will be indicated later in Section 5.5.

5.4 Conversion from Hash Families

Constructions for hash families have been extensively investigated by many researches. Here, we assume the existence of certain hash families and use them to construct secure frameproof codes. We first construct small codes and use them as the initial seed to construct bigger ones.

We will use sandwich free families, perfect hash families, and separating hash families to construct SFP codes. Note that in the construction the unreadable marks are unnecessary for discussion. Before doing so, we present a direct construction and a recursive construction of SFP codes which explains the idea of recursive construction.

Theorem 5.14. *For any integer $w \geqslant 2$, there is a $w - SFP\left(\binom{2w-1}{w-1}, 2w\right)$.*

Proof. Recall the representation of incidence matrix defined in Section 3.5 of set systems. Let the code C be built from an incidence matrix whose first row contains all $1s$ and the remaining columns corresponds to $\mathcal{B}$ which is the set of subsets $B_1, \ldots, B_N$, where B_i contains all possible $(w-1)$ choices out of $(2w-1)$ elements, yielding $N = \binom{2w-1}{w-1}$. We will show that $C = \left\{u^{(1)}, \ldots, u^{(2w)}\right\}$ is a $w - SFP(N, n)$ code where $N = \binom{2w-1}{w-1}$ is the code length and $n = 2w$ is the code size. It suffices to verify that for all $C_1, C_2 \subseteq C$ and $|C_1| = |C_2| = w$, $C_1 \cap C_2 = \emptyset$. Since $n = 2w$, if follows that $C_2 = C \setminus C_1$. Because the code length is $N = \binom{2w-1}{w-1}$, there is a unique bit position i such that $u_i^{(j)} = 1$ for all $u^{(j)} \in C_1$ and $u_i^{(j)} = 0$ for all $u^{(j)} \in C_2$. It follows that $x_i = 1$ if $x \in desc_w(C_1)$ and $x_i = 0$ if $x \in desc_w(C_2)$ or vice versa. Hence, $desc_w(C_1) \cap desc_w(C_2) = \emptyset$. ∎

Example 5.5. *Using the above method, a 3-SFP(10,6) code can be constructed and interpreted as an incidence matrix as follows:*

$$M(C) = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}_{6\times 10}$$

A recursive construction can be provided in a similar way.

Theorem 5.15. *For any integer $w \geqslant 2$, there is a $w - SFP\left(2\binom{2w-1}{w-1}, 2w+1\right)$.*

Proof. Let C be the code defined in Theorem 5.14. Denote by $M(C)$ the incidence matrix of C of dimension $2w \times \binom{2w-1}{w-1}$. Then we can construct a $(2w+1) \times 2\binom{2w-1}{w-1}$ matrix M as follows:

$$M = \left(\begin{array}{c|c} M(C) & M(C) \\ \hline 0 \cdots 0 & 1 \cdots 1 \end{array} \right)$$

Then, it is not hard to say that M can serve as the incidence matrix of a $w - SFP\left(2\binom{2w-1}{w-1}, 2w+1\right)$. ∎

Next, we formulate the SFP codes in terms of hash families. Recall the definitions of set systems and sandwich free families defined earlier in Section 3.5 and Section 3.6:

Lemma 5.1. *Let* $C = \left\{u^{(1)}, \ldots, u^{(w)}\right\} \subseteq \{0,1\}^N$ *and let* $x \in \{0,1\}^N$. *Then* $x \in desc_w(C)$ *if and only if*

$$\bigcap_{i=1}^{w} B_{u^{(i)}} \subseteq B_x \subseteq \bigcup_{i=1}^{w} B_{u^{(i)}}.$$

Proof. Note that $\cap B_{u^{(i)}} \subseteq B_x$ if and only if $x_j = 1$ when all the codewords in C have jth bit equal to 1. Likewise, $B_x \subseteq \cup B_{u^{(i)}}$ if and only if $x_j = 0$ when all the codewords in C have jth bit equal to 0. The conclusion follows. ∎

Based on the lemma, we restate Theorem 3.5 and prove it now.

Theorem 5.16. *A* $w - SFP(N,n)$ *exists if and only if there exists a* $(w,w) - SFF(N,n)$.

Proof. Suppose that $(X, \mathcal{B})$ is a set system. It suffices to say that $(X, \mathcal{B})$ is not a $(w,w) - SFF$ if and only if there is a set $W \subseteq X$ such that

$$\bigcap_{B \in C_1} B \subseteq W \subseteq \bigcup_{B \in C_1} B$$

and

$$\bigcap_{B \in C_2} B \subseteq W \subseteq \bigcup_{B \in C_2} B$$

where $|C_1| = |C_2| = w$. Now, viewing C_1 and C_2 as sets of codewords in the associated (n, N)-code, the two conditions above are equivalent to

$$desc_w(C_1) \cap desc_w(C_2) \neq \emptyset.$$

by Lemma 5.1. ∎

The following two theorems are given earlier in Chapter 3 which will be used now.

Theorem 5.17. *A (N, n, q)-code, C, is a $w-SFP$ code if $\mathcal{H}(C)$ is an $PHF(N; n, q, 2w)$, where $n \geqslant 2w$.*

Theorem 5.18. *A (N, n, q)-code, C, is a $w - SFP$ code if and only if $\mathcal{H}(C)$ is an $SHF(N; n, q, w, w)$, where $n \geqslant 2w$.*

These theorems tell us that if we can find the constructions for PHF or SHF, we have equivalently the SFP codes as well. We now examine the recursive construction.

Theorem 5.19. *If there exists a $(w_1, w_2)-SFF(v, m)$ and a $PHF(N; n, m, w_1+w_2)$, then there exists a $(w_1, w_2) - SFF(vN, n)$.*

Proof. Let (X, B) be n $(w_1, w_2)-SFF(v, m)$, and let F be a $PHF(N; n, m, w_1+w_2)$, where $f : Y \mapsto X$ for any $f \in F$. Define $W = X \times F$, and for every $y \in Y$, define

$$A_y = \left\{ \left(B_{f(y)}, f \right) : f \in F \right\}.$$

Let $A = (A_y : y \in Y)$. We will show that the set system (W, A) is a $(w_1, w_2) - SFF(vN, n)$.

Suppose that (W, A) is not a $(w_1, w_2) - SFF(vN, n)$. Then there exist two disjoint subsets $C_1, C_2 \subseteq Y$ such that $|C_1| = w_1$, $|C_2| = w_2$ and

$$\left(\bigcap_{y \in C_1} A_y \right) \bigcup \left(\bigcap_{y \in C_2} A_y \right) \subseteq \left(\bigcup_{y \in C_1} A_y \right) \bigcap \left(\bigcup_{y \in C_2} A_y \right).$$

Then, for every $f \in F$, it must be the case that

$$\left(\bigcap_{y \in C_1} B_{f(y)} \right) \bigcup \left(\bigcap_{y \in C_2} B_{f(y)} \right) \subseteq \left(\bigcup_{y \in C_1} B_{f(y)} \right) \bigcap \left(\bigcup_{y \in C_2} B_{f(y)} \right).$$

However, since F is a perfect hash family, there is an $f \in F$ such that $f|_{C_1 \cup C_2}$ is one-to-one. For this particular f, $f(C_1)$ and $f(C_2)$ are two disjoint subsets of X, and therefore the last equation contradicts the fact that (X, B) is an $(w_1, w_2) - SFF(v, m)$. ∎

In [4], a recursive construction of perfect hash families is discussed in order to provide an infinite class of prefect hash families. They are stated in the following two theorems.

Theorem 5.20. *Suppose there exists a* $PHF(N_0; n_0, m, w)$*, and suppose that* $gcd(n_0, \binom{w}{2}!) = 1$*. Then there is a* $PHF\left(\left(\binom{w}{2}+1\right)N_0; n_0^2, m, w\right)$.

Proof. Recall the definition of difference matrices mentioned in Section 3.4. Denote $D = (d_{i,j})$ by the rule $d_{i,j} = ij \bmod n_0, 0 \leqslant i \leqslant \binom{w}{2}, 0 \leqslant j \leqslant n_0 - 1$. According to the fact that $gcd(n_0, \binom{w}{2}!) = 1$, the values of $d_{i,j} \bmod n_0$ are pairwise distinct. Therefore, D is an $\left(n_0, \binom{n}{2}+1; 1\right)$-difference matrix which can be embedded into the original $PHF(N_0; n_0, m, w)$ to yield a bigger $PHF\left(\left(\binom{w}{2}+1\right)N_0; n_0^2, m, w\right)$. ∎

One nice thing about Theorem 5.20 is that it can be iterated.

Theorem 5.21. *Suppose there exists a* $PHF(N_0; n_0, m, w)$*, and suppose that* $gcd(n_0, \binom{w}{2}!) = 1$*. Then there is a* $PHF\left(\left(\binom{w}{2}+1\right)^j N_0; n_0^{2^j}, m, w\right)$ *for any integer* $j \geqslant 1$.

In order to iterate, we need two seeds as the following:

Example 5.6. *There exists a* $PHF(7; 7, 4, 4)$ *as follows:*

$$\begin{pmatrix} 1 & 2 & 3 & 4 & 1 & 2 & 3 \\ 1 & 2 & 3 & 4 & 2 & 1 & 4 \\ 1 & 2 & 3 & 4 & 3 & 4 & 1 \\ 1 & 2 & 3 & 4 & 4 & 3 & 2 \\ 2 & 3 & 2 & 3 & 1 & 1 & 4 \\ 2 & 4 & 1 & 2 & 3 & 4 & 3 \\ 1 & 1 & 2 & 2 & 3 & 4 & 3 \end{pmatrix}$$

Example 5.7. *There exists an* $(2,2) - SFF(3,4)$*, or* $2 - SFP(3,4)$ *whose incidence matrix can be depicted as follows:*

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

Then, with Theorem 5.19 and Theorem 5.21 in mind, we have the following infinite classes of 2-SFP codes.

Theorem 5.22. *There exists a* $2 - SFP(3 \cdot 7^{j+1}, 7^{2^j})$ *for all* $j \geqslant 0$.

Proof. We have Example 5.6 as a initial seed, and we iterate by Theorem 5.21 to get an infinite class of $PHF\left(\left(\binom{4}{2}+1\right)^j 7; 7^{2^j}, 4, 4\right) = PHF\left(7^{j+1}; 7^{2^j}, 4, 4\right)$ for all $j \geqslant 0$. On the other hand, since a $2 - SFP(3, 4)$ exists by Example 5.7, we have an infinite class of $2 - SFP(3 \cdot 7^{j+1}, 7^{2^j})$ by Theorem 5.19. ∎

If we use separating hash families instead of perfect hash families, we also have a similar recursive construction.

Theorem 5.23. *If there exists an* $(w_1, w_2)-SFF(v, m)$ *and an* $SHF(N; n, m, w_1, w_2)$, *then there exists an* $(w_1, w_2) - SFF(vN, n)$.

The proof is similar as before. Also in [4], a similar recursive construction for providing infinite class of separating hash families is provided as follows:

Theorem 5.24. *Suppose there exists an* $SHF(N_0; n_0, m, w_1, w_2)$, *where* $gcd(n_0, (w_1w_2)!) = 1$. *Then there exists an* $SHF\left((w_1w_2+1)^j N_0; n_0^{2^j}, m, w_1, w_2\right)$ *for any integer* $j \geqslant 0$.

An initial seed of separating hash families can be provided below:

Example 5.8. *There exists an* $SHF(3; 7, 4, 2, 2)$ *as follows:*

$$\begin{pmatrix} 1 & 1 & 2 & 2 & 3 & 3 & 4 \\ 2 & 1 & 1 & 2 & 4 & 3 & 3 \\ 1 & 2 & 1 & 2 & 3 & 4 & 3 \end{pmatrix}$$

Combining the seeds served by Example 5.7 and Example 5.8 leads to:

Theorem 5.25. *There exists a* $2 - SFP(9 \cdot 5^j, 7^{2^j})$ *for all* $j \geqslant 0$.

Proof. From Theorem 5.24 and Example 5.8, we have an infinite class of $SHF(3 \cdot 5^j; 7^{2^j}, 4, 2, 2)$ for all $j \geqslant 0$. Since a $2 - SFP(3, 4)$ exists by Example 5.7, the conclusion follows by Theorem 5.23. ∎

Here the code rate is

$$\begin{aligned} R &= N^{-1}\log n \\ &= \frac{1}{9\cdot 5^j}\log 7^{2^j} \\ &= \frac{2^j \log 7}{9\cdot 5^j} \\ &= \frac{\log 7}{9}\left(\frac{2}{5}\right)^j \end{aligned}$$

which still tends to zero as j goes to infinity.

Also, the asymptotic behavior of code length is $N = \mathcal{O}\left((\log_7 n)^{\log_2 7}\right)$.

A more general result for $w \geqslant 2$ can be provided in a similar fashion.

Theorem 5.26 (Stinson [31])**.** *Let $w \geqslant 2$. Then there exists an $w - SFP\left(2\binom{2d-1}{d-1}\cdot(w^2+1)^j, (2d+1)^{2^j}\right)$ for all $j \geqslant 0$ and $d > w$ such that $gcd\,(2d+1, (w^2)!) = 1$.*

The proof is similar to the one of Theorem 5.20 combining the existence of $w - SFP\left(2\binom{2w-1}{w-1}, 2w+1\right)$ in Theorem 5.15.

The following result is an immediate corollary of Theorem 5.26.

Corollary 5.2. *For any $w \geqslant 2$, there exists an explicit construction for an infinite class of $w - SFP(N, n)$ where $N = \mathcal{O}\left((\log n)^{\log_2(w^2+1)}\right)$*

It is important that we choose our seeds as best as possible. Moreover, in [5, 32], more new constructions for perfect hash families and separating hash families are established using orthogonal arrays and Latin rectangles as follows.

Theorem 5.27. *For any positive integers m and w such that $w \leqslant m$, there exists an infinite class of $PHF(N; n, m, w)$ for which N is $\mathcal{O}\left((w^2)^{\log^* n}(\log n)\right)$.*

Note that $log^* : \mathbb{N} \mapsto \mathbb{N}$ is a function growing very slowly and is defined recursively as

$$\begin{aligned} &\log^* 1 = 1 \\ &\log^* n = \log^*(\lceil \log n \rceil) + 1, \text{ if } n > 1. \end{aligned}$$

Example 5.9. $\log^* 10^{10} = \log^* 10 + 1 = \log^* 1 + 1 + 1 = 1 + 1 + 1 = 3.$

Theorem 5.28. *For any positive integers m, w_1 and w_2, there exists an infinite class of $SHF(N; n, m, w_1, w_2)$ for which N is $\mathcal{O}\left((w_1 w_2)^{\log^* n} (\log n)\right)$.*

This gives immediately the following consequence.

Corollary 5.3. *For any positive integers m, w, there exists an infinite class of $w - SFP(N, n, q)$ for which N is $\mathcal{O}\left((w^2)^{\log^* n} (\log n)\right)$.*

Here the code rate is

$$R = N^{-1} \log n \sim \frac{\log n}{(w^2)^{\log^* n} (\log n)} = \frac{1}{(w^2)^{\log^* n}}$$

which would tend to zero however much slower than the previous one.

The proofs of Theorem 5.27 and 5.28 are beyond the scope of our thesis and can be found in [32].

5.5 Linear Codes

Recall the notions defined in Section 2.1.4. Error correcting codes with sufficiently large minimum distance can serve the secure-frameproof property. Also, positive code rates which have not been achieved before will be obtained by using error correcting codes here.

In [1, 14, 4], the following condition is given.

Theorem 5.29. *If an error correcting code* $C(N,k,d)_q$ *satisfies* $\frac{d}{N} > 1 - \frac{1}{\binom{w}{2}}$, *then it is also an* $PHF(N;q^k,q,w)$.

Proof. To say that C is a PHF, it suffices to say that any w codewords $\{u^{(1)},\ldots,u^{(w)}\}$ of C contain w distinct values in at least one entry. Otherwise, then for every entry some pair of the u_i's agree and hence the sum of distances between the $\binom{w}{2}$ pairs of u_i's is $\leqslant N\binom{w}{2} - N = \binom{w}{2}N\left(1 - 1/\binom{w}{2}\right)$, a contradiction. ∎

One nice application can be derived from Reed Solomon codes. A Reed Solomon code is a maximum distance separable code having parameters $(q-1,k,q-k)_q$, where $k < q-1$ and q is a prime power. Suppose we take p to be a prime, $p \geqslant \binom{w}{2}$, $q = p^j$ and $k = p^{j-1}$. Then we verify the conditions of Theorem 5.29:

$$\begin{aligned}\frac{d}{N} &= \frac{p^j - p^{j-1}}{p^j - 1} \\ &> 1 - \frac{1}{p} \\ &\geqslant 1 - \frac{1}{\binom{w}{2}}\end{aligned}$$

Hence, we obtain the following result.

Corollary 5.4. *Suppose* p *is a prime,* $p \geqslant \binom{w}{2}$, *and* $j \geqslant 1$. *Then there is a* $PHF\left(p^j - 1; p^{jp^{j-1}}, p^j, w\right)$.

Combining the results of Theorem 5.17 and Corollary 5.4, we have the following SFP codes constructed using Reed Solomon codes.

Theorem 5.30. *Suppose* p *is a prime,* $p \geqslant \binom{w}{2}$, *and* $j \geqslant 1$. *Then there is a* $\lfloor \frac{w}{2} \rfloor - SFP\left(p^j - 1, p^{jp^{j-1}}, p^j\right)$.

A similar condition for separating hash families can be obtained.

Theorem 5.31. *If an error correcting code $C(n,k,d)_q$ satisfies $\frac{d}{n} > 1 - \frac{1}{w_1 w_2}$, then it is also a $SHF(n;k,q,w_1,w_2)$.*

Similar constructions using Reed-Solomon codes can be derived as well.

Theorem 5.32. *Suppose p is a prime, $p \geqslant w^2$, and $j \geqslant 1$. Then there is a $w - SFP\left(p^j - 1, p^{jp^{j-1}}, p^j\right)$.*

This is an advanced construction with a positive code rate.

$$\begin{aligned} R &= N^{-1}\log_q n \\ &= \frac{\log_{(p^j)}\left(p^{jp^{j-1}}\right)}{p^j - 1} \\ &= \frac{\log p^{jp^{j-1}}}{(p^j-1)\log p^j} \\ &= \frac{jp^{j-1}\log p}{j\,(p^j-1)\log p} \\ &= \frac{p^{j-1}}{p^j-1} \\ &\sim \frac{1}{p} \end{aligned}$$

Moreover, note that $N = \mathcal{O}\left(\frac{p}{j}\log_p n\right)$.

Subsequently, we restrict ourselves to the situation $w = 2$. In [15], a necessary condition is provided.

Theorem 5.33. *If C is a linear, binary $2 - SFP$ code of dimension k, then $D < N - 2(k-2)$.*

Proof. If $k \leqslant 1$, the result is trivial. For $k = 2$, it only says that the all-one codeword 1 cannot be in the code C. Suppose not, denote by **0** the all-zero codeword and by **1** the all-one codeword, and choose a codeword $x \in C$, $x \neq$ **0** or **1**. Then,

$desc_2(\{\mathbf{0},\mathbf{1}\}) = desc_2(\{x, x+1\})$. Here $x+\mathbf{1}$ inverts each entry of x by changing 0 to 1 and 1 to 0 in x, which violates the $2-SFP$ property.

We then turn to the case $k \geqslant 3$. We shall prove that if $N - D \leqslant 2(k-2)$, then C cannot be a $2-SFP$. Consider a codeword x of maximum Hamming weight. Since the code is linear, for every set of $k-2$ coordinate positions, there exist at least three nonzero codewords which are zero on these positions, and thus at least on which is not x. In particular, there is a nonzero codeword u which is zero on half the positions not in the support of x, and one v which is zero on the other half. Thus $desc_2(\{\mathbf{0}, c\}) \cap desc_2(\{u, v\}) \neq \emptyset$. ∎

Theorem 5.33 gives the necessary condition for a $2-SFP$ code. The following theorem gives the sufficient condition (compare with Theorem 5.29):

Theorem 5.34. *If a linear code satisfies $4d > 2D + N$, or if $4d > 3D$, then it is $2-SFP$.*

The proof is provided in [15] which exhaustively investigates the possibilities of D and d given four codewords and identifies the condition that they can be separated.

Equidistant codes (see Section 2.1.4) provide more examples or $2-SFP$ codes.

Theorem 5.35. *All linear, equidistant codes are $2-SFP$.*

The proof is easy by simply checking the equidistant property between any two codewords and similar to the one given as for Theorem 5.5.

The nonlinear case can also be addressed as follows.

Theorem 5.36. *All nonlinear, equidistant codes with $2d > n$ are $2-SFP$.*

The proof can be found in [18].

In the sequel we provide an example [15, 14] which combines the construction techniques of concatenation methods and error correcting codes.

Choose the first seed C_1 as a $(4, 2, 3)_3$ tetracode. This code is a MDS code, extended perfect Hamming code, and equidistant with Hamming distance 3 defined

by the generator matrix

$$\begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & 2 & 1 \end{bmatrix}$$

Clearly, it is a $2-SFP$.

The second seed we use to concatenate with the tetracode is the extended Reed-Solomon code $C_2(9,3,7)_{3^2}$. It is a $2-SFP$ by Theorem 5.29. The result is then $C_1 \bigstar C_2(36,6)_3$, which is another $2-SFP$ by Theorem 5.13.

Next, in order to produce infinite families of SFP codes with positive rates, we need the following constructive result of algebraic-geometry codes.

Theorem 5.37 (Tsfasman [38])**.** *For any $\alpha > 0$ there is an infinite family of codes $C\left(N, NR, N\delta\right)_q$ for $N \geqslant N_0(\alpha)$ and*

$$R + \delta \geqslant 1 - \left(\sqrt{q} - 1\right)^{-1} - \alpha$$

where R stands for the code rate.

Now $C_1 \bigstar C_2$ is a large enough seed for the algebraic-geometry codes of [38]. Because of Theorem 5.34, we choose $d = \lceil 3N/4 \rceil$ and hence, concatenate with the $(N,k,d)_{3^6}$ algebraic-geometry code C of rate approximately $R = \frac{1}{4} - (3^3 - 1)^{-1} = 11/52$. We summarize the result in the following theorem.

Theorem 5.38. *$\{C_1 \bigstar C_2 \bigstar C\}_N$ gives an infinite class of linear ternary $2-SFP$ codes with positive rates $R' = R/6 \approx 0.0352$.*

The structure of algebraic geometric is beyond the scope of the thesis. Their construction can be found in [38]. Cohen et al. gives more algebraic geometry codes of various rates used for similar concatenations. For more details, we refer the interested reader to [15].

5.6 Comparisons

We summarize this chapter by providing a table of comparison.

Method	Coalition size (w)	Code length (N)	Code size (n)	Alphabet size (q)	Asymptotic Behavior	Code Rate (R)	Comment
Hadamard Matrix	2	$4k$	$4k$	2	$N = \mathcal{O}(n)$	$\to 0$	Theorem 5.5
Subsets Method	2	$\binom{k}{\lfloor \frac{k}{2} \rfloor}$	$\binom{k}{3}$	2	$N = \mathcal{O}\left((\log n)^3\right)$	$\to 0$	Theorem 5.11
Direct Construction	w	$\binom{2w-1}{w-1}$	$2w$	2	$N = \mathcal{O}(2^{n-1})$	*	Theorem 5.14
PHF and SFF	2	$3 \cdot 7^{j+1}$	7^{2^j}	2	$N = \mathcal{O}\left((\log_7 n)^{\log_2 7}\right)$	$\to 0$	Theorem 5.22
SHF and SFF	2	$9 \cdot 5^j$	7^{2^j}	2	$N = \mathcal{O}\left((\log_5 n)^{\log_2 5}\right)$	$\to 0$	Theorem 5.25
Extended SFP	w	$2\binom{2d-1}{d-1} \cdot \left(w^2+1\right)^j$	$(2d+1)^{2^j}$	2	$N = \mathcal{O}\left((\log n)^{\log_2 \left(w^2+1\right)}\right)$	$\to 0$	Theorem 5.26
Latin Rectangle	2	N	n	m	$N = \mathcal{O}\left(w\right)^{2 \log^* n} (\log n))$	$\to 0$	Corollary 5.3
Reed Solomon Code	w	$p^j - 1$	$p^{jp^{j-1}}$	p^j	$N = \mathcal{O}\left(\frac{p}{j} \log_p n\right)$	$\to \frac{1}{p}$	Theorem 5.32
Concatenation with Algebraic Geometry Code	2	N	n	3	Depending on the structure of algebraic geometry code	$\to 0.0352$	Theorem 5.38

*We do not compute the rate because the rate is a function of w resulting in different classes of codes.

Chapter 6

Summary

We have tried to give a complete picture of codes for copyright protection. Also, we redefined the descendence under the presence of unreadable marks. In particular, we investigated various constructions of secure frameproof codes. Most of the explicit constructions discussed so far treat coalitions of size 2. Few of them handle a general coalition size. Most of the code rates tend to zero except for two constructions that are based on error correcting codes and algebraic geometry codes. However, as was indicated in the beginning of the thesis, it is important to be able to handle coalitions of large size and the code size should be as large as possible in order to accommodate many users. On the other hand, under the presence of unreadable marks, it is impossible for the police to identify the traitors. As an alternative, the probabilistic approach is capable of tracing traitors with a certain successful probability, which might be an interesting direction of future research.

Appendix A

Acronyms

ECC Error Correcting Code

HF Hash Functions

PHF Perfect Hash Families

SHF Separating Hash Families

SS Set Systems

SFF Sandwich Free Families

FP Frameproof Code

SFP Secure Frameproof Code

IPP Identifiable-Parent-Property Code

TA Traceability Code

PTT Probabilistic Traitor Tracing

TTA Traitor Tracing Algorithm

MDS Maximum Distance Separable

RS Reed Solomon Code

Bibliography

[1] N. Alon, "Explicit construction of exponential sized families of k-independent sets," *Discrete Math*, vol. 58, 191 - 193, 1986.

[2] N. Alon, E. Fischer, and M. Szegedy, "Parent-identifying codes," *Journal of Combinatorial Theory Series A*, vol. 95, 349 - 359, 2001.

[3] N. Alon and J. Spencer, "The Probabilistic Method," Wiley, 1992.

[4] M. Atici, S. S. Magliveras, D. R. Stinson, and W.-D. Wei, "Some Recursive Constructions for Perfect Hash Families," *Journal of Combinatorial Designs*, vol. 4, 353 - 363, 1996.

[5] M. Atici, D. R. Stinson, and R. Wei, "A new practical algorithm for the construction of a perfect hash function," *J. Combin. Math. Combin. Comput.*, vol. 35, 127 - 145, 2000.

[6] A. Barg, G. Cohen, S. Encheva, G. Kabatiansky, and G. Zémor, "A hypergraph approach to the identifying parent property: the case of multiple parents," *SIAM J. Discrete. Math*, vol. 14, 423 - 431, 2001.

[7] Th. Beth, D. Jungnickel and H. Lenz, "Design Theory," Wissenschaftsverlag, Berlin, 1985.

[8] D. Boneh and J. Shaw, "Collusion-scrure fingerprinting for digital data," *IEEE Transactions on Information Theory*, vol. 44, no. 5, 1897 - 1905, 1998.

[9] C. J. Colbourn and J. H. Dinitz, "CRC Handbook of Combinatorial Designs," CRC Press, Inc., 1996.

[10] B. Chor, A. Fiat, and M. Naor. "Tracing traitors," *Lecture Notes in Computer Science*, vol. 839, 257 - 270, 1994.

[11] H. Chu, L. Qiao, and K. Nahrstedt, "A secure multicast protocol with copyright protection," *ACM SIGCOMM Computer Communications Review*, vol. 32, no. 2, 42 - 60, 2002.

[12] G. Cohen and S. Encheva, "On some efficient constructions of frameproof codes," *Information Theory, 2001. Proceedings. 2001 IEEE International Symposium*, 118, 2001.

[13] G. Cohen and S. Encheva, "Efficient constructions of frameproof codes," *Electronics Letters*, vol. 36, no. 22, 1840 - 1842, 2000.

[14] G. Cohen, S. Encheva, S. Litsyn, and H. G. Schaathun, "Intersecting codes and separating codes," *Discrete Applied Mathematics*, vol. 128, 75 - 83, 2002.

[15] G. Cohen, S. Encheva, and H. G. Schaathun, "More on (2,2) separating systems," *IEEE Transactions on Information Theory*, vol. 48, no. 9, 2606 - 2609, 2002.

[16] Z. J. Czech, G. Havas and B. S. Majewski. "Perfect hashing," *Theoretical Computer Science*, vol. 182, 1 - 143, 1997.

[17] S. Encheva and G. Cohen, "Some new p-ary two-secure frameproof codes," *Applied Mathematics Letters*, vol. 4, 177 - 182, 2001.

[18] S. Encheva and G. Cohen, "Identifying codes for copyright protection," Dept. INF, CNRS, Paris, France, Tech. Rep., 2001.

[19] M. Fernández and M. Soriano, "Decoding codes wiht the idengifiable parent property," *The Seventh IEEE Symposium on Computers and communications ISCC 2002 Taormina (Italy)*, 2002.

[20] E. Gafni, J. Staddon, and Y. L. Yin, "Efficient methords for integrating traceability and broadcast encryption," *Advances in Cryptology-Crypto '99 (Lecture Notes in Computer Science)*, Springer-Verlag, vol. 1666, 372 - 387, 1999.

[21] H. Hollmann, J. van Lint, J.-P. Linnartz, and L. Tolhuizen, "On codes with identifiable parent property," *Journal of Combinatorial Theory A*, vol. 82, 121 - 133, 1998.

[22] N. F. Johnson, Z. Duric, and S. Jajodia, "A Role of Digital Watermarking in Electronic Commerce," *Special Issue of the ACM on Electronic Commerce*, 1999.

[23] D. Kirovski, H. S. Malvar, and Y. Yacobi, "Multimedia content screening using a dual watermarking and fingerprinting system," *Proc. ACM Multimedia*, 372 - 381, 2002.

[24] P. C. Li, G. H. J. van Rees, and R. Wei, "Constructions of 2-cover-free families and related separatinghash families," *Submitted*, 2005.

[25] J. van Lint, "Introduction to Coding Theory," Springer-Verlag, 1982.

[26] J. van Lint and R. M. Wilson, "A Course in Combinatorics," Combridge University Press, 1992.

[27] F. J. MacWilliams and N. J. A. Sloane, "The Theory of Error-Correcting Codes," North-Holland Mathematical Library, 1983.

[28] P. Sarkar and D. R. Stinson, "Frameproof and IPP Codes," *Lecture Notes in Computer Science*, vol. 2247, 117 - 126, 2001.

[29] A. Silverberg, J. N. Staddon, and J. L. Walker, "Efficient traitor tracing algorithms using list decoding," *ASIACRYPT 2001, Lecture Notes in Computer Science*, vol. 2248, 175 - 192, 2001.

[30] J. N. Staddon, D. R. Stinson, R. Wei, "Combinatorial properties of frameproof and traceability codes," *IEEE Transactions on Information Theory*, vol. 47, no. 3, 1042 - 1049, 2001.

[31] D. R. Stinson, T. van Trung, and R. Wei, "Secure frameproof codes, key distribution patterns, group testing algorihtms and related structures," *Journal of Statistical Planning and Inference*, vol. 86, no. 2, 595 - 617, 2000.

[32] D. R. Stinson, R. Wei, and L. Zhu, "New constructions for perfect hash families and related structures using combinatorial designs and codes," *Journal of Combinatorial Designs*, vol. 8, no. 3, 189 - 200, 2000.

[33] D. Tonien and R. Safavi-Naini, "Recursive constructions of secure codes and hash families using difference function families," *Journal of Combinatorial Theory A*, vol. 133, 664-674, 2006.

[34] D. Tonien and R. Safavi-Naini, "Explicit construction of secure frameproof codes," *International Journal of Pure and Applied Mathematics*, vol. 6, no. 3, 343 - 360, 2003.

[35] T. van Trung and S. Martirosyan, "New constructions for IPP codes," *Designs, Codes, and Cryptography*, vol. 35, 227 - 239, 2005.

[36] T. van Trung and Sosina Martirosyan, “On a class of traceability codes,” *Designs, Codes and Cryptography*, vol. 31, no. 2, 125 - 132, 2004.

[37] T. van Trung and Sosina Martirosyan, “Constructions for efficient IPP code,” *preprint*, 2002.

[38] M. A. Tsfasman, “Algebraic-geometric codes and asymptotic problems,” *Discrete Applied Mathathematics*, vol. 33, 241 - 256, 1991.