Secure Frameproof Codes, Key Distribution Patterns, Group Testing Algorithms and Related Structures

D. R. Stinson
Department of Computer Science and Engineering
University of Nebraska-Lincoln
Lincoln, NE 68588
stinson@bibd.unl.edu

Tran van Trung
Institute for Experimental Mathematics
University of Essen
Ellernstraße 29, 45326 Essen, Germany
trung@exp-math.uni-essen.de

R. Wei
Department of Mathematics and Statistics
University of Nebraska-Lincoln
Lincoln, NE 68588
rwei@cse.unl.edu

November 26, 1997

Abstract

Frameproof codes were introduced by Boneh and Shaw as a method of "digital fingerprinting" which prevents a coalition of a specified size c from framing a user not in the coalition. Stinson and Wei then gave a combinatorial formulation of the problem in terms of certain types of extremal set sytems.

In this paper, we study frameproof codes that provide a certain (weak) form of traceability. We extend our combinatorial formulation to address this stronger requirement, and show that the problem is solved by using (i, j)-separating systems, as defined by Friedman, Graham and Ullman. Using constructions based on perfect hash families, we give the first efficient explicit constructions for these objects for general values of i and j. We also review nonconstructive existence results that are based on probabilistic arguments.

Then we look at two other, related concepts, namely key distribution patterns and non-adaptive group testing algorithms. We again approach these problems from the point of view of extremal set systems, and we describe a natural common setting in which these two problems are complementary special cases. This approach also demonstrates a close relationship between these two problems and frameproof codes. Explicit constructions are given, and some non-constructive existence results are reviewed. In the case of key distribution patterns, our explicit constructions are the most efficient ones known.

1 Introduction

In order to protect a product (such as digital data, computer software, etc.), a distributor marks each copy with some codeword and then ships each user his data "marked" with that codeword. This marking (a "digital fingerprint") allows the distributor to detect any unauthorized copy and trace it back to the user who created it. This will deter users from releasing an unauthorized copy. However, a coalition of users may detect some of the marks, namely the ones where their copies differ. They can then change these marks arbitrarily. To prevent a group of users from "framing" another user, Boneh and Shaw [6] defined the concept of c-frameproof codes. A c-frameproof code has the property that no coalition of at most c users can frame a user not in the coalition. In [38], combinatorial methods are used to further investigate frameproof codes. Several constructions of c-frameproof codes are given in [6, 38, 8].

Frameproof codes prevent a coalition from framing a user not in the coalition. However, the coalition may be able to change some of the marks they have detected so that the distributor cannot trace an illegal copy. To protect against this situation, we might try to construct a code such that, given an illegal copy, at least one user in the coalition that created it can be found. Unfortunately, such a code cannot exist, as pointed out in [6]. In this article, we consider a slightly weaker property, where we require that it is impossible that an illegal copy could be created by two disjoint coalitions. In a sense, our definition is the strongest that can be realized.

We present a combinatorial formulation of this problem that turns out to be equivalent to the (i, j)-separating systems defined almost 30 years ago by Friedman, Graham and Ullman [18]. Using constructions based on perfect hash families, we give the first efficient explicit constructions for these objects for general values of i and j (our constructions produce systems in which the number of blocks is a polynomial in $\log v$ for fixed i and j). We also review nonconstructive existence results that are based on probabilistic arguments, presenting a unified treatment of several bounds that can be found in the literature. These bounds improve on the explicit constructions, showing the existence of systems in which the number of blocks is $O(\log v)$ for fixed i and j.

In the second part of the paper we look at two other, related concepts, namely key distribution patterns and non-adaptive group testing algorithms. We again approach these problems from the point of view of extremal set systems, and we describe a natural common setting which shows a close relationship between these two problems and frameproof codes. We accomplish this by generalizing the definitions of two (equivalent) types of set systems, namely cover-free families and disjunct systems, and show that the two problems are in fact complementary special cases of these set systems. Our explicit constructions based on hash families can be applied here as well, yielding key distribution patterns in which the number of keys is a polynomial in the logarithm of the number of participants, as well as group testing algorithms in which the number of tests is a polynomial in the logarithm number of samples. In the case of key distribution patterns, our explicit constructions are the most efficient ones known. We also review a simple probabilistic method that can be used to obtain good bounds.

Remark 1.1 We will be defining several types of set systems in this paper. To improve readability, we have listed all the abbreviations we use in the Appendix.

2 Definitions and basic results on frameproof codes

In this section, we give definitions of frameproof codes and secure frameproof codes, and prove some basic results and properties of them.

2.1 Frameproof codes

Let v and b be positive integers (b denotes the number of users in the scheme). A set $\Gamma = \{w^{(1)}, w^{(2)}, \ldots, w^{(b)}\} \subseteq \{0,1\}^v$ is called a (v,b)-*code* and each $w^{(i)}$ is called a *codeword*. (For $1 \leq i \leq b$, the codeword $w^{(i)}$ is held by user i.) A binary v-tuple $x \in \{0,1\}^v \backslash \Gamma$ is called an *unregistered word*. Given a code Γ, the *incidence matrix* $M(\Gamma)$ will be the $b \times v$ matrix in which the rows are the b codewords in Γ.

Let Γ be a (v,b)-code. Suppose $C = \{w^{(u_1)}, w^{(u_2)}, \ldots, w^{(u_d)}\} \subseteq \Gamma$. For $i \in \{1, 2, \ldots, v\}$, we say that bit position i is *undetectable* for C if

$$w_i^{(u_1)} = w_i^{(u_2)} = \ldots = w_i^{(u_d)}.$$

Let $U(C)$ be the set of undetectable bit positions for C. Then

$$F(C) = \{x \in \{0,1\}^v : x|_{U(C)} = w^{(u_i)}|_{U(C)} \text{ for all } w^{(u_i)} \in C\}$$

is called the *feasible set* of C. The feasible set $F(C)$ represents the set of all possible v-tuples that could be produced by the coalition C by comparing the d codewords they jointly hold. Observe that $C \subseteq F(C)$ for all C, and $F(C) = C$ if $|C| = 1$.

Now, if there is a codeword $w^{(j)} \in F(C) \backslash C$, then user j could be "framed" if the coalition C produces the v-tuple $w^{(j)}$. The following definition from [6] is motivated by the desire for this situation not to occur.

Definition 2.1 A (v,b)-code Γ is called a *c-frameproof code* if, for every $C \subseteq \Gamma$ such that $|C| \leq c$, we have $F(C) \cap \Gamma = C$. We will say that Γ is a c-FPC(v,b) for short.

Thus, in a c-frameproof code, the only codewords in the feasible set of a coalition of at most c users are the codewords of the members of the coalition. Hence, no coalition of at most c users can frame a user who is not in the coalition.

Example 2.1 ([6]) For any integer b, there exists a b-FPC(b,b), Γ. The incidence matrix $M(\Gamma)$ is a $b \times b$ identity matrix. $\square$

In general, we are interested in constructing c-FPC(v,b) with b as large as possible as a function of c and v. Using the above example and error correcting codes with large minimal distance, Boneh and Shaw [6] gave a construction of frameproof codes. Stinson and Wei [38] investigated the combinatorial properties of frameproof codes and gave some explicit constructions of frameproof codes by using combinatorial designs, such as t-designs, packing designs and perfect hash families. Further existence results, both constructive and nonconstructive, can be found in Chee [8]. Related questions, including generalizations of frameproof codes to the setting of public-key cryptography, have been studied in [3, 9, 28, 29, 30, 31].

2.2 Traceability of frameproof codes

Suppose that Γ is a c-FPC(v, b). For any $x \in \{0,1\}^v$, define

$$F^{-1}(x) = \{C \subseteq \Gamma : |C| \leq c \text{ and } x \in F(C)\}.$$

Evidently, $F^{-1}(x)$ consists of all the coalitions of size at most c that could have produced x.

Suppose that $x \in \{0,1\}^v \backslash \Gamma$ (i.e., x is an unregistered word). If it happened that $|F^{-1}(x)| = 1$, say $F^{-1}(x) = \{C\}$, then we could conclude that C was the coalition that constructed x (assuming, of course, that all coalitions have size at most c). More generally, if $F^{-1}(x) \neq \emptyset$ and there exists a codeword $w^{(j)}$ such that $w^{(j)} \in C$ for all $C \in F^{-1}(x)$, then we would at least be able to identify user j as being guilty. Unfortunately, as shown in [6], this is hoping for too much. The following theorem is a straightforward generalization of [6, Theorem 11], which concerned the case $c = 2$.

Theorem 2.1 *Suppose Γ is a c-FPC(v, b) with $b \geq 2c - 1$. Suppose $D \subseteq \Gamma$, where $|D| = 2c - 1$. Then there exists an unregistered word $\mathsf{maj}(D) \in \{0,1\}^v$ such that $\mathsf{maj}(D) \in F(C)$ for any $C \subseteq D$ with $|C| = c$.*

Proof. Let $D = \{w^{(u_1)}, w^{(u_2)}, \ldots, w^{(u_{2c-1})}\}$. For $1 \leq i \leq v$, define

$$\mathsf{maj}(D)_i = \begin{cases} 1 & \text{if } |\{j : w_i^{(u_j)} = 1\}| \geq c \\ 0 & \text{if } |\{j : w_i^{(u_j)} = 0\}| \geq c. \end{cases}$$

It is easy to see that $\mathsf{maj}(D) \in F(C)$ for any $C \subseteq D$ with $|C| = c$. It remains to show that $\mathsf{maj}(D)$ is an unregistered word. Suppose not; then $\mathsf{maj}(D) = w^{(u)}$ for some u. Let $C \subseteq D \backslash \{w^{(u)}\}$ with $|C| = c$. Then $w^{(u)} \in F(C) \cap \Gamma$, which contradicts the fact that Γ is c-frameproof. $\square$

The above theorem says that we cannot be guaranteed of identifying a guilty user in a c-FPC(v, b). For, if $x = \mathsf{maj}(D)$ for some D where $|D| = 2c - 1$, then

$$\bigcap_{C \in F^{-1}(x)} C = \emptyset.$$

Thus we are forced to consider a weaker condition, as follows.

Definition 2.2 Suppose that Γ is a (v, b)-code. Γ is said to be a *c-secure frameproof code* if for any $C_1, C_2 \subseteq \Gamma$ such that $|C_1| \leq c$, $|C_2| \leq c$ and $C_1 \cap C_2 = \emptyset$, we have that $F(C_1) \cap F(C_2) = \emptyset$. We will say that Γ is a c-SFPC(v, b) for short.

Example 2.2 Let $M(\Gamma)$ denote the following incidence matrix for a $(3, 4)$-code:

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

We will show that Γ is a 2-SFPC$(3,4)$ by computing $F(C)$ for all C such that $|C| \leq 2$:

$$\begin{aligned} F(\{w^{(1)}, w^{(2)}\}) &= \{100, 010, 110, 000\}, \\ F(\{w^{(1)}, w^{(3)}\}) &= \{100, 001, 101, 000\}, \\ F(\{w^{(1)}, w^{(4)}\}) &= \{100, 111, 101, 110\}, \\ F(\{w^{(2)}, w^{(3)}\}) &= \{010, 001, 000, 011\}, \\ F(\{w^{(2)}, w^{(4)}\}) &= \{010, 111, 011, 110\} \qquad \text{and} \\ F(\{w^{(3)}, w^{(4)}\}) &= \{001, 111, 011, 101\}. \end{aligned}$$

From this, it can easily be checked that

$$\begin{aligned} F(\{w^{(1)}, w^{(2)}\}) \cap F(\{w^{(3)}, w^{(4)}\}) &= \emptyset, \\ F(\{w^{(1)}, w^{(3)}\}) \cap F(\{w^{(2)}, w^{(4)}\}) &= \emptyset \qquad \text{and} \\ F(\{w^{(1)}, w^{(4)}\}) \cap F(\{w^{(2)}, w^{(3)}\}) &= \emptyset. \end{aligned}$$

$\square$

The following result is easy.

Theorem 2.2 *A c-SFPC(v,b) is a c-FPC(v,b).*

Proof. Let Γ be a c-SFPC(v,b). Suppose that Γ is not a c-FPC(v,b). Then there exists a set $C \subseteq \Gamma$ and a codeword $w^{(j)}$ such that $|C| \leq c$ and $w^{(j)} \in F(C) \backslash C$. Suppose we define $C_1 = C$ and $C_2 = \{w^{(j)}\}$. Then, we have $|C_1| \leq c$, $|C_2| \leq c$, $C_1 \cap C_2 = \emptyset$ and $F(C_1) \cap F(C_2) = \{w^{(j)}\} \neq \emptyset$. This contradicts the fact that Γ is a c-SFPC(v,b). $\square$

A c-SFPC(v,b) does not permit traceability, but it does afford some security, as follows:

- It is impossible for a coalition C_1 of size at most c to implicate a disjoint coalition C_2 of size at most c by constructing an unregistered word $x \in F(C_1)$.
- If x is an unregistered word that has been constucted by a coalition of size at most c, then any $C \in F^{-1}(x)$ contains at least one guilty user.

We now consider 2-SFPC(v,b) in more detail. Suppose that Γ is a 2-SFPC(v,b), suppose that x is an unregistered word, and suppose that $C \in F^{-1}(x)$. Since x is an unregistered word, $|C| \neq 1$. Since Γ is a 2-SFPC(v,b), $|C| \leq 2$. Therefore, $|C| = 2$.

Since $F^{-1}(x)$ consists of a collection of 2-subsets of Γ, we can view it as the set of edges of a graph on vertex set Γ. Since Γ is a 2-SFPC, it must be the case that any two distinct edges in $F^{-1}(x)$ are incident. From this it is easily seen that one of two possibilies must occur:

1. $F^{-1}(x)$ is a *star graph* (i.e., there exists a vertex that is incident with every edge of $F^{-1}(x)$).
2. $F^{-1}(x)$ is isomorphic to K_3 (the complete graph on three vertices).

As a consequence of this characterization of $F^{-1}(x)$ in the case $c = 2$, we obtain the following result.

Theorem 2.3 *Suppose that Γ is a 2-SFPC(v,b), and suppose that x is an unregistered word that is produced by a coalition of size at most two. Then one of the following two possibilies must occur:*

1. *at least one guilty user can be identified; or*

2. *a set of three participants can be identified, two of which must be guilty.*

3 Combinatorial descriptions of secure frameproof codes

In this section, we give two combinatorial descriptions of c-SFPC(v,b). The first equivalence involves set systems, satisfying certain union and intersection properties, that we call "sandwich-free families". The second concerns separating systems, as defined in [18].

3.1 Sandwich-free families

We first define some terminology concerning set systems. 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 $X=\{x_1,x_2,\dots,x_v\}$ and $\mathcal{B}=\{B_1,B_2,\dots,B_b\}$. The *incidence matrix* of $(X,\mathcal{B})$ is the $b\times v$ 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.

Now, if Γ is a (v,b)-code, then the matrix $M(\Gamma)$ is a 0-1 matrix, which can therefore be thought of as the incidence matrix of a set system. For any codeword $w\in\Gamma$, we will use B_w to denote the associated block in the corresponding set system.

Lemma 3.1 *Let $C=\{w^{(1)},w^{(2)},\cdots,w^{(d)}\}\subseteq\{0,1\}^v$ and let $x\in\{0,1\}^v$. Then $x\in F(C)$ if and only if*

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

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

Lemma 3.1 can be used to prove the following combinatorial description of frameproof codes that was given in [38].

Theorem 3.2 *[38] There exists a c-FPC(v,b) if and only if there exists a set system $(X,\mathcal{B})$ such that $|X|=v,|\mathcal{B}|=b$ and for any subset of $d\le c$ blocks $B_1,B_2,\dots,B_d\in\mathcal{B}$, there does not exist a block $B\in\mathcal{B}\backslash\{B_1,B_2,\dots,B_d\}$ such that*

$$\bigcap_{i=1}^{d}B_i\subseteq B\subseteq\bigcup_{i=1}^{d}B_i.$$

We will present a similar description of secure frameproof codes. First, we need to define a certain type of set system.

Definition 3.1 A set system $(X, \mathcal{B})$ is an (i,j)*-sandwich-free family* provided that, for any two disjoint subsets C_1, C_2 of $\mathcal{B}$, where $|C_1| \leq i$ and $|C_2| \leq j$, the following property holds:

$$\left(\bigcap_{B \in C_1} B\right) \bigcup \left(\bigcap_{B \in C_2} B\right) \not\subseteq \left(\bigcup_{B \in C_1} B\right) \bigcap \left(\bigcup_{B \in C_2} B\right).$$

An (i,j)-sandwich-free family, $(X, \mathcal{B})$, will be denoted as an (i,j)-SFF(v,b) if $|X| = v$ and $|\mathcal{B}| = b$.

Theorem 3.3 *A c-SFPC(v,b) exists if and only if there exists a (c,c)-SFF(v,b).*

Proof. Suppose that $(X, \mathcal{B})$ is a set system. It is easy to see that $(X, \mathcal{B})$ is not a (c,c)-sandwich-free family 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 $|C_1| = |C_2| = c$. Now, viewing C_1 and C_2 as sets of codewords in the associated (v,b)-code, the two conditions above are equivalent to

$$F(C_1) \cap F(C_2) \neq \emptyset,$$

by Lemma 3.1. ∎

Example 3.1 The following $(2,2)$-SFF$(3,4)$ is equivalent to the 2-SFPC$(3,4)$ presented in Example 2.2:

$$\begin{aligned} X &= \{1,2,3\} \\ \mathcal{B} &= \{\{1\},\{2\},\{3\},\{1,2,3\}\}. \end{aligned}$$

∎

We also have the following result concerning frameproof codes, which follows immediately from Theorem 3.2.

Theorem 3.4 *A c-FPC(v,b) exists if and only if there exists a $(1,c)$-SFF(v,b).*

3.2 Separating systems

Friedman, Graham and Ullman [18] defined separating systems as follows.

Definition 3.2 A set system $(X, \mathcal{B})$ is an (i, j)-*separating system* provide that, for any $P, Q \subseteq X$ such that $|P| \leq i$, $|Q| \leq j$ and $P \cap Q = \emptyset$, there exists a $B \in \mathcal{B}$ such that either $P \subseteq B$ and $Q \cap B = \emptyset$; or $Q \subseteq B$ and $P \cap B = \emptyset$. An (i, j)-separating system, $(X, \mathcal{B})$, will be denoted as an (i, j)-SS(v, b) if $|X| = v$ and $|\mathcal{B}| = b$.

Sandwich-free families and separating systems are closely related; they are in fact "dual" incidence structures. This is made precise in the proof of the following theorem.

Theorem 3.5 *There exists an (i, j)-SFF(v, b) if and only if there exists an (i, j)-SS(b, v).*

Proof. Suppose $(X, \mathcal{B})$ is an (i, j)-SFF(v, b). Let $C_1, C_2 \subseteq X$, $|C_1| = i$, $|C_2| = j$ and $C_1 \cap C_2 = \emptyset$. Then there exists a point $x \in X$ such that

$$x \in \left(\bigcap_{B \in C_1} B \right) \bigcup \left(\bigcap_{B \in C_2} B \right) \quad \text{and} \quad x \notin \left(\bigcup_{B \in C_1} B \right) \bigcap \left(\bigcup_{B \in C_2} B \right).$$

It follows that either

$$x \in \bigcap_{B \in C_1} B \quad \text{and} \quad x \notin \bigcup_{B \in C_2} B,$$

or

$$x \in \bigcap_{B \in C_2} B \quad \text{and} \quad x \notin \bigcup_{B \in C_1} B.$$

From this, it is easily seen that if A is the incidence matrix of $(X, \mathcal{B})$, then the transpose, A^T, is the incidence matrix of an (i, j)-SS(b, v).

Conversely, if A is the $v \times b$ incidence matrix of an (i, j)-SS(b, v), then A^T is the incidence matrix of an (i, j)-SFF(v, b). $\square$

We have the following corollary of Theorems 3.5, 3.3 and 3.4.

Corollary 3.6 *A c-FPC(v, b) exists if and only if there exists a $(1, c)$-SS(b, v), and a c-SFPC(v, b) exists if and only if there exists a (c, c)-SS(b, v).*

Example 3.2 The following $(2, 2)$-SS$(4, 3)$ is equivalent to the 2-SFPC$(3, 4)$ presented in Example 2.2 and the 2-SFF$(3, 4)$ presented in Example 3.1:

$$\begin{aligned} X &= \{1, 2, 3, 4\} \\ \mathcal{B} &= \{\{1, 4\}, \{2, 4\}, \{3, 4\}\}. \end{aligned}$$

$\square$

4 Constructions of secure frameproof codes

In this section we present some explicit constructions, both direct and recursive, for secure frameproof codes.

4.1 Two direct constructions

We begin with two direct constructions of secure codes.

Theorem 4.1 *For any integer $c \geq 2$, there is a c-SFPC$\left(\binom{2c-1}{c-1}, 2c\right)$.*

Proof. We define the incidence matrix $M(\Gamma)$. The rows of $M(\Gamma)$ are indexed by the elements in the set $\{1, \ldots, 2c\}$, and the columns are indexed by the c-subsets $S \subseteq \{1, \ldots, 2c\}$ such that $1 \in S$. Denote these subsets as $S_1, \ldots, S_v$, where $v = \binom{2c-1}{c-1}$. Now, the entry in row i and column j of $M(\Gamma)$ is defined to be

$$m_{ij} = \begin{cases} 1 & \text{if } i \in S_j \\ 0 & \text{if } i \notin S_j. \end{cases}$$

We show that $\Gamma = \{w^{(1)}, \ldots, w^{(2c)}\}$ is a c-SFPC$\left(\binom{2c-1}{c-1}, 2c\right)$. It suffices to verify that Definition 2.2 is satisfied for all $C_1, C_2 \subseteq \Gamma$ such that $|C_1| = |C_2| = c$ and $C_1 \cap C_2 = \emptyset$. Since $b = 2c$, it follows that $C_2 = \Gamma \backslash C_1$. Without loss of generality, suppose that $w^{(1)} \in C_1$. Now, there is a unique bit position i such that $w_i^{(j)} = 1$ for all $w^{(j)} \in C_1$ and $w_i^{(j)} = 0$ for all $w^{(j)} \in C_2$. If follows that $x_i = 1$ if $x \in F(C_1)$ and $x_i = 0$ if $x \in F(C_2)$. Hence, $F(C_1) \cap F(C_2) = \emptyset$, as desired. □

Remark 4.1 The c-SFPC constructed in Theorem 4.1 has v as small as possible, given that $b = 2c$.

Example 4.1 The 2-SFPC$(3, 4)$ given in Example 2.2 is constructed by the method of Theorem 4.1. □

Example 4.2 We present a 3-SFPC$(10, 6)$ constructed using the method described in Theorem 4.1. The incidence matrix $M(\Gamma)$ is as follows:

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

□

The following result can be proved in a similar way.

Theorem 4.2 *For any integer $c \geq 2$, there is a c-SFPC$\left(2\binom{2c-1}{c-1}, 2c+1\right)$.*

Proof. Let the $2c \times \binom{2c-1}{c-1}$ matrix $M(\Gamma)$ be defined as in Theorem 4.1. Then construct a $(2c+1) \times 2\binom{2c-1}{c-1}$ matrix M as follows:

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

It is not hard to show that M is the incidence matrix of a c-SFPC$\left(2\binom{2c-1}{c-1}, 2c+1\right)$. □

4.2 A construction using perfect hash families

Perfect hash families have been extensively studied by computer scientists for over 15 years. Results on perfect hash families can be found in numerous textbooks and papers; Mehlhorn [26] is a good textbook source. Recently, several applications in cryptography have been found, including broadcast encryption [17], threshold signature schemes [5, 4] and frameproof codes [38]. We will modify our construction given in [38, Theorem 3.12] so that it can be used for secure frameproof codes.

First, we need to define perfect hash families.

Definition 4.1 An (n, m, w)-*perfect hash family* is a set of functions $\mathcal{F}$, such that $|Y| = n$, $|X| = m$, $f : Y \to X$ for each $f \in \mathcal{F}$, and for any $C \subseteq \{1, 2, \ldots, n\}$ such that $|C| = w$, there exists at least one $f \in \mathcal{F}$ such that $f|_C$ is one-to-one. When $|\mathcal{F}| = N$, an (n, m, w)-perfect hash family will be denoted by $\mathrm{PHF}(N; n, m, w)$.

Remark 4.2 A $\mathrm{PHF}(N; n, m, w)$ can be depicted as an $N \times n$ matrix with entries from $\{1, 2, \ldots, m\}$, having the property that in any w columns there exists at least one row such that the w entries in the given w columns are distinct.

Given a "small" c-SFPC, we can recursively construct a "large" c-SFPC by using perfect hash families. We present our construction, using the more general language of sandwich-free families, in the following theorem.

Theorem 4.3 *If there exists an (i, j)-SFF(v, m) and a PHF$(N; n, m, i + j)$, then there exists an (i, j)-SFF(vN, n).*

Proof. Let $(X, \mathcal{B})$ be an (i, j)-SFF(v, m), and let $\mathcal{F}$ be a $\mathrm{PHF}(N; n, m, i + j)$, where $f : Y \to X$ for any $f \in \mathcal{F}$.

Define $W = X \times \mathcal{F}$, and for every $y \in Y$, define

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

Let $\mathcal{A} = \{A_y : y \in Y\}$. We will show that the set system $(W, \mathcal{A})$ is an (i, j)-SFF(vN, n).

Suppose that $(W, \mathcal{A})$ is not an (i, j)-SFF(vN, n). Then there exist two disjoint subsets $C_1, C_2 \subseteq Y$ such that $|C_1| = i$, $|C_2| = j$ 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 $f \in \mathcal{F}$, it must be the case that

$$\left(\bigcap_{y \in C_1} B_{f(y)}\right) \bigcup \left(\bigcap_{y \in C_2} B_{f(y)}\right) \subseteq \left(\bigcup_{y \in C_1} B_{f(y)}\right) \bigcap \left(\bigcup_{y \in C_2} B_{f(y)}\right). \tag{1}$$

Now, since $\mathcal{F}$ is a perfect hash family, there is an $f \in \mathcal{F}$ 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 (1) contradicts the fact that $(X, \mathcal{B})$ is an (i, j)-SFF(v, m). $\square$

The following recursive construction is useful in obtaining explicit constructions for infinite classes of perfect hash families.

Lemma 4.4 *[2, Theorem 4.4] Suppose there exists a PHF$(N_0; n_0, m, w)$, where* $\gcd\left(n_0, \binom{w}{2}!\right) = 1$. *Then there exists a PHF$\left(\left(\binom{w}{2}+1\right)^j N_0; {n_0}^{2^j}, m, w\right)$ for any integer $j \geq 0$.*

We give an example to show how Lemma 4.4 and Theorem 4.3 can be applied to obtain an infinite family of 2-SFPC that can be constructed efficiently.

Example 4.3 There exists a PHF$(7; 7, 4, 4)$ as follows:

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

□

Theorem 4.5 *There exists a 2-SFPC$(3 \cdot 7^{j+1}, 7^{2^j})$ for all $j \geq 0$.*

Proof. From Lemma 4.4 and Example 4.3, we have a PHF$(7^{j+1}; 7^{2^j}, 4, 4)$ for all $j \geq 0$. Since a 2-SFPC$(3, 4)$ exists by Example 2.2, the conclusion follows from Theorem 4.3. □

4.3 A construction using separating hash families

If we look closely at the proof of Theorem 4.3, we see that the existence of a hash function f, that is one-to-one on the set $C_1 \cup C_2$ of size $i+j$, is stronger than what we require. In fact, it is sufficient that, for any two disjoint subsets C_1 and C_2 of sizes i and j, respectively, there exists a hash function f such that $\{f(y) : y \in C_1\} \cap \{f(y) : y \in C_2\} = \emptyset$. This motivates the following definition.

Definition 4.2 An $(n, m, \{w_1, w_2\})$*-separating hash family* is a set of functions $\mathcal{F}$, such that $|Y| = n$, $|X| = m$, $f : Y \to X$ for each $f \in \mathcal{F}$, and for any $C_1, C_2 \subseteq \{1, 2, \ldots, n\}$ such that $|C_1| = w_1$, $|C_2| = w_2$ and $C_1 \cap C_2 = \emptyset$, there exists at least one $f \in \mathcal{F}$ such that

$$\{f(y) : y \in C_1\} \cap \{f(y) : y \in C_2\} = \emptyset.$$

The notation SHF$(N; n, m, \{w_1, w_2\})$ will be used to denote an $(n, m, \{w_1, w_2\})$-separating hash family with $|\mathcal{F}| = N$.

Remark 4.3 An SHF$(N; n, m, \{w_1, w_2\})$ can be depicted as an $N \times n$ matrix with entries from $\{1, 2, \ldots, m\}$, such that in any two disjoint sets C_1 and C_2 of w_1 and w_2 columns (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.

It is easy to see that any PHF$(N; n, m, w)$ is also an SHF$(N; n, m, \{w_1, w_2\})$ if $w_1 + w_2 = w$. Also, an SHF$(N; n, m, \{1, 1\})$ is equivalent to a PHF$(N; n, m, 2)$. The following result is also very easy; we record it as a lemma for future reference.

Lemma 4.6 *An SHF$(N; n, 2, \{w_1, w_2\})$ is equivalent to a (w_1, w_2)-SS(n, N).*

The following theorem is proved in essentially the same way as Theorem 4.3.

Theorem 4.7 *If there exists an (i, j)-SFF(v, m) and an SHF$(N; n, m, \{i, j\})$, then there exists an (i, j)-SFF(vN, n).*

We state a recursive construction for separating hash families that is similar to Lemma 4.4.

Theorem 4.8 *Suppose there exists an SHF$(N_0; n_0, m, \{w_1, w_2\})$, where $\gcd(n_0, w_1 w_2) = 1$. Then there exists an SHF$\Big((w_1 w_2 + 1)^j N_0; {n_0}^{2^j}, m, \{w_1, w_2\}\Big)$ for any integer $j \geq 0$.*

Proof. The proof is essentially the same as the proof of [2, Theorem 4.4]. □

We illustrate the application of the above results by constructing an infinite family of 2-SFPC. The following separating hash family is easily constructed by hand.

Example 4.4 There exists an SHF$(3; 7, 4, \{2, 2\})$ as follows:

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

□

Theorem 4.9 *There exists a 2-SFPC$(9 \cdot 5^j, 7^{2^j})$ for all $j \geq 0$.*

Proof. From Theorem 4.8 and Example 4.4, we have an SHF$(3 \cdot 5^j; 7^{2^j}, 4, \{2, 2\})$ for all $j \geq 0$. Since a 2-SFPC$(3, 4)$ exists by Example 2.2, the conclusion follows from Theorem 4.7. □

Theorem 4.9 produces an infinite family of 2-SFPC(v, b) in which v is $O\left((\log b)^{\log_2 5}\right)$. This represents a significant improvement over Theorem 4.5, in which v is $O\left((\log b)^{\log_2 7}\right)$.

The following general result can be proved in a similar fashion.

Theorem 4.10 *Let $c \geq 2$. Then there exists a c-SFPC$(2\binom{2c-1}{c-1} \cdot (c^2+1)^j, (2c+1)^{2^j})$ for all $j \geq 0$.*

Proof. Theorem 4.2 shows that a c-SFPC$\left(2\binom{2c-1}{c-1}, 2c+1\right)$ exists. By Corollary 3.6, this is equivalent to a (c, c)-SS$\left(2c+1, 2\binom{2c-1}{c-1}\right)$. Applying Lemma 4.6, this is in turn equivalent to an SHF$\left(2\binom{2c-1}{c-1}, 2c+1, 2, \{c, c\}\right)$. Since $\gcd(2c+1, c^2) = 1$, we can apply Theorem 4.8 to construct an SHF$\left(2\binom{2c-1}{c-1}(c^2+1)^j, (2c+1)^{2^j}, 2, \{c, c\}\right)$ for all $j \geq 0$. By Lemma 4.6 and Corollary 3.6, the resulting SHF are equivalent to the desired c-SFPC. □

The following result is an immediate corollary of Theorem 4.10.

Corollary 4.11 *For any $c \geq 2$, there exists an explicit construction for an infinite class of c-SFPC(v,b) in which v is $O\left((\log b)^{\log_2(c^2+1)}\right)$.*

In view of Theorem 3.5, constructions for c-SFPC(v,b) can equivalently be viewed as constructions for (c,c)-SS(b,v). As far as we know, Theorem 4.10 provides the first efficient explicit constructions for (c,c)-SS(b,v) for general c. (A construction, similar in flavour to Theorem 4.9, was already given in [18] for the case $c=2$.)

5 Nonconstructive bounds

Nonconstructive existence results can often be obtained by probabilistic methods. For some of the structures under discussion in this paper, bounds of this type have been derived. In this section, we present a unified treatment of several of these bounds, most of which are proved in a similar fashion. To illustrate the technique used, we begin with a bound for perfect hash families. This is essentially the bound first proved in [25] (see also [26]).

Given a graph $G=(V,E)$, let $P(G,m)$ be the *chromatic polynomial* of G, which is defined as follows: For a positive integer m, $P(G,m)$ denotes the number of *m-colourings of G* (i.e., the number of ways to colour the vertices of G using a specified set of m colours, such that no two vertices $v_1, v_2 \in V$ having the same colour are joined by an edge $e \in E$). It is well-known that $P(G,m)$ is a polynomial in m of degree $|V|$. If the vertices of G are coloured independently at random using m colours, then the probability that the result is an m-colouring is $P(G,m)/m^{|V|}$.

Now, let A be an $N \times n$ matrix whose entries are elements of a fixed set S of size m and whose columns are labelled $1,\ldots,n$. For a set C of w columns of A, define $X_A(C)=0$ if there exists a row of A such that the entries in the columns in C are distinct, and define $X_A(C)=1$, otherwise. Let $X(C)$ denote the random variable defined by letting A be a random $N \times n$ matrix (i.e., the entries of A are chosen independently at random from S). Clearly, the expected value of $X(C)$ is

$$\begin{aligned} E[X(C)] &= \left(1-\frac{P(K_w,m)}{m^w}\right)^N \\ &= \left(1-\frac{m(m-1)\cdots(m-w+1)}{m^w}\right)^N . \end{aligned}$$

If we define the random variable

$$X = \sum_{\{C \subseteq \{1,\ldots,n\} : |C|=w\}} X(C),$$

then we have the following formula:

$$E[X] = \binom{n}{w}\left(1-\frac{m(m-1)\cdots(m-w+1)}{m^w}\right)^N .$$

It is clear that there exists a PHF$(N;n,m,w)$ if $E[X]<1$. Thus we have proved the following bound of Mehlhorn:

Theorem 5.1 *[25] There exists a PHF($N; n, m, w$) if*

$$N > \frac{\log \binom{n}{w}}{\log (m^w) - \log (m^w - w!\binom{m}{w})}.$$

We can use a similar approach to prove bounds for separating hash families. Suppose that $|C_1| = w_1$, $|C_2| = w_2$ and $C_1 \cap C_2 = \emptyset$. Denote $w = w_1 + w_2$. Given an $N \times n$ array A, define $X_A(C_1, C_2) = 0$ if there exists a row of A such that the entries in the columns in C_1 are distinct from the entries in the columns in C_2, and define $X_A(C_1, C_2) = 1$, otherwise. Then we have

$$E[X(C_1, C_2)] = \left(1 - \frac{P(K_{w_1,w_2}, m)}{m^w}\right)^N.$$

For convenience, denote

$$p = 1 - \frac{P(K_{w_1,w_2}, m)}{m^w}.$$

Then we have

$$E[X] = \begin{cases} \binom{n}{w_1}\binom{n-w_1}{w_2}p^N & \text{if } w_1 \neq w_2 \\ \frac{1}{2}\binom{n}{w_1}\binom{n-w_1}{w_1}p^N & \text{if } w_1 = w_2. \end{cases}$$

As was the case with perfect hash families, there will exist an SHF($N; n, m, \{w_1, w_2\}$) if $E[X] < 1$. A slightly improved bound can be obtained, however, by observing that if $E[X] < n/2$, then an SHF($N; n/2, m, \{w_1, w_2\}$) exists. (This is a standard method in probabilistic combinatorics, and this observation also could have been used in deriving a bound for perfect hash families.) This discussion gives rise to the following two bounds.

Theorem 5.2 *Suppose that n, m, w_1 and w_2 are positive integers and p is defined as above. Then the following hold:*

1. *If*

$$N > \frac{(w_1 + w_2)\log n}{-\log p},$$

then there exists an SHF($N; n, m, \{w_1, w_2\}$).

2. *If*

$$N > \frac{(w_1 + w_2 - 1)\log(2n)}{-\log p},$$

then there exists an SHF($N; n, m, \{w_1, w_2\}$).

5.1 Discussion and applications

We illustrate the application of Theorem 5.2 by deriving some known bounds as corollaries. We will set $m = 2$ since this case corresponds to separating systems and secure frameproof codes. It is easy to see that

$$P(K_{w_1,w_2}, 2) = 2,$$

and hence

$$p = 1 - \frac{1}{2^{w_1+w_2-1}}$$

when $m = 2$.

Here are some applications of Theorem 5.2:

- Suppose that $w_1 = 1$ and $w_2 = 2$. Then $p = 3/4$, and
$$\frac{w_1 + w_2 - 1}{-\log_2 p} = \frac{2}{2 - \log_2 3} \approx 4.819.$$
Hence, a $(1,2)$-SS$(n, \approx 4.819 \log_2 n)$ exists by part 2. of Theorem 5.2. This result was shown independently in Alon and Spencer [1, Theorem 1.1, p. 200], Körner [24, Theorem 1] and Chee [8, Theorem 9.3.1].

- Suppose that $w_1 = w_2 = 2$. Then $p = 7/8$ and
$$\frac{w_1 + w_2 - 1}{-\log_2 p} = \frac{3}{2 - \log_2 7} \approx 15.573.$$
Hence, a $(2,2)$-SS$(n, \approx 15.573 \log_2 n)$ exists by part 2. of Theorem 5.2. This result was shown in Körner and Simonyi [23, Theorem 2]. Earlier, it had been shown by Friedman, Graham and Ullman [18, Theorem 5] that a $(2,2)$-SS$(n, \approx 20.764 \log_2 n)$ exists. This is in fact the result that would be obtained from part 1. of Theorem 5.2.

- Friedman, Graham and Ullman [18, p. 546] showed the existence of a (w_1, w_2)-SS$(n, \gamma \log_2 n)$ where γ is the value given in part 1. of Theorem 5.2. Of course this can be improved slightly by instead using part 2. of the same theorem.

6 Related topics

In this section, we study two topics: key distribution patterns (which are of interest in cryptography) and non-adaptive group testing algorithms. We begin defining the types of set systems that we will use in studying these topics.

6.1 Cover-free families and disjunct systems

Definition 6.1 A set system $(X, \mathcal{B})$ is an (i,j)*-cover-free family* provided that, for any two disjoint subsets C_1, C_2 of $\mathcal{B}$, where $|C_1| \leq i$ and $|C_2| \leq j$, it holds that
$$\bigcap_{B \in C_1} B \not\subseteq \bigcup_{B \in C_2} B.$$
An (i,j)-cover-free family, $(X, \mathcal{B})$, will be denoted as an (i,j)-CFF(v, b) if $|X| = v$ and $|\mathcal{B}| = b$.

Remark 6.1 Our definition is a generalization of term "cover-free family" as it is commonly used in the combinatorial literature (see, e.g., [15]). The standard definition corresponds to the case $i = 1$. Another equivalent concept is "superimposed distance codes"; see [22, 12].

Definition 6.2 A set system $(X, \mathcal{B})$ is an (i,j)*-disjunct system* provide that, for any $P, Q \subseteq X$ such that $|P| \leq i$, $|Q| \leq j$ and $P \cap Q = \emptyset$, there exists a $B \in \mathcal{B}$ such that $P \subseteq B$ and $Q \cap B = \emptyset$. An (i,j)-disjunct system, $(X, \mathcal{B})$, will be denoted as an (i,j)-DS(v, b) if $|X| = v$ and $|\mathcal{B}| = b$.

Remark 6.2 Our definition is a generalization of term "disjunct" as it is defined in [11, p. 62]. The definition in [11] corresponds to the case $i = 1$. When $i = 1$, this property is also known as j-complete; see [7].

Cover-free families and disjunct systems are dual incidence structures. The following is proved in the same way as Theorem 3.5.

Theorem 6.1 *There exists an (i, j)-CFF(v, b) if and only if there exists an (i, j)-DS(b, v).*

The following two lemmas follow immediately from the definitions.

Lemma 6.2 *Any (i, j)-disjunct system is an (i, j)-separating system.*

Lemma 6.3 *Any (i, j)-cover-free family is an (i, j)-sandwich-free family.*

We now give a combinatorial interpretation of the incidence matrix of a disjunct system.

Lemma 6.4 *An (i, j)-DS(n, N) is equivalent to an $N \times n$ $0-1$ matrix, such that in any two disjoint sets C_1 and C_2 of i and j columns (respectively), there exists at least one row such that the entries in the columns C_1 are all "1"s and the entries in the columns C_2 are are all "0"s.*

We noted above that disjunct systems are separating systems and cover-free familes are sandwich-free families. A simple construction allows us to reverse this process at the expense of doubling the number of blocks and points, respectively.

Theorem 6.5 *If there exists an (i, j)-SS(v, b), then there exists an (i, j)-DS$(v, 2b)$.*

Proof. Let $(X, \mathcal{B})$ be an (i, j)-SS(v, b). Define

$$\mathcal{C} = \mathcal{B} \cup \{X \backslash B : B \in \mathcal{B}\}.$$

It is easy to check that $(X, \mathcal{C})$ is an (i, j)-DS(v, b). ∎

A equivalent version of Theorem 6.5 is stated in the following corollary.

Corollary 6.6 *If there exists an (i, j)-SFF(v, b), then there exists an (i, j)-CFF$(2v, b)$.*

Proof. Apply Theorems 6.5, 6.1 and 3.5. ∎

6.2 Key distribution patterns

The elegant idea of a key distribution pattern is due to Mitchell and Piper [27]. Here is an informal definition.

Definition 6.3 Let v and b be positive integers. An (i, j)-*key distribution pattern* is a method of distributing a set of v keys to a set of b users, such that any subset of i users can form a conference keys by combining the keys that they hold in common. Any conference key thus formed should be secure against a (disjoint) coalition of size at most j. The key distribution pattern will be denoted as an (i, j)-*KDP*(b, v).

We now describe how key distribution patterns are constructed from cover-free families. Suppose that $(X, \mathcal{B})$ is an (i,j)-CFF(v,b), where $i \geq 2$. For each $x \in X$, let k_x be a *key*, chosen at random from a specified abelian group, say G. Suppose we have a set of b participants, denoted u_B $(B \in \mathcal{B})$, and each participant u_B is given the keys k_x $(x \in B)$. Let C be a subset of i participants. Then, for any coalition D of size at most j that is disjoint from C, there exists a key that is held by every member of C and by no member of D. Now, suppose the conference key k_C is defined to be

$$k_C = \sum_{\{x : x \in B \text{ for all } B \in C\}} k_x.$$

Then every member of C can compute the conference key k_C, but the value of k_C cannot be computed by any coalition D of size at most j.

Key distribution patterns have been an active area of study in the last ten years. For more information, the reader can consult [13, 20, 27, 32, 33, 34, 35, 36, 37].

6.3 Group testing algorithms

We informally define non-adaptive group testing algorithms.

Definition 6.4 Suppose that X is a set of v *samples* that are to be tested positive or negative. Suppose that $\mathcal{B}$ is a set of subsets of X, where each $B \in \mathcal{B}$ represents a subset of samples (called a *group*) that are to be combined and tested together. The testing procedure has the property that if a group contains at least one positive sample, then the test result for that group is positive. Suppose that the testing procedure allows the identification of the positive samples if the number positive samples is at most d. Then the resulting scheme is called a *non-adaptive group testing algorithm* and is denoted by d-NAGTA(v,b). (The term "non-adaptive" means that the tests performed are fixed ahead of time, and do not depend on the outcome of earlier tests. This is useful in practice due to simplicity, as well as the fact that a non-adaptive algorithm can be parallelized to any desired degree.)

Disjunct systems can be used to construct group testing algorithms. Suppose that $(X, \mathcal{B})$ is a $(1,d)$-DS(v,b), where X is the set of v samples and $\mathcal{B}$ is the set of b groups. It is clear that the samples that occur in no group that tests positive are in fact the negative samples. Thus, the positive samples are identified by this testing procudure, and we have a d-NAGTA(v,b).

There is an extensive literature on group testing algorithms. The interested reader should consult the book by Du and Hwang [11], in which many references are given. Two important papers on the topic are [7] and [21].

7 Constructions

The constructions we have discussed in Sections 4 and 5 can be modified in a straightforward fashion to produce cover-free families or (equivalently) disjunct systems. When $i = 1$, we obtain non-adaptive group testing algorithms, and when $i \geq 2$, we obtain key distribution patterns. (In view of Lemma 6.2, these constructions also yield (i,j)-separating systems.) The constructions for key distribution patterns are the most efficient explicit constructions that are known.

7.1 New explicit constructions

The following recursive construction is essentially the same as Theorem 4.8.

Theorem 7.1 *Suppose there exists an (i,j)-$DS(n_0, N_0)$, where* $\gcd(n_0, ij) = 1$. *Then there exists an (i,j)-$DS\left({n_0}^{2^k}, (ij+1)^k N_0\right)$ for any integer $k \geq 0$.*

In order to apply Theorem 7.1, we start with a "small" disjunct system which we construct by direct methods.

Lemma 7.2 *Suppose that i and j are positive integers and $n \geq i + j$. Then there exists an (i,j)-$DS(n, N)$ where $N = \min\left\{\binom{n}{i}, \binom{n}{j}\right\}$.*

Proof. Let X be an n-set. The set of all i-subsets of X is an (i,j)-disjunct system, as is the set of all $(n-j)$-subsets of X. ∎

Now we can easily obtain the following general result, which provides an explcitly contructed class of (i,j)-disjunct systems for any i and j.

Theorem 7.3 *Suppose that i and j are positive integers. Let*

$$n_0 = \min\{n \geq i + j : \gcd(n, ij) = 1\}$$

and let $N_0 = \min\left\{\binom{n_0}{i}, \binom{n_0}{j}\right\}$. Then there exists an (i,j)-$DS\left({n_0}^{2^k}, (ij+1)^k N_0\right)$ for any integer $k \geq 0$.

Proof. Apply Theorem 7.1 and Lemma 7.2. ∎

The following corollary, which is similar to Corollary 4.11, is an immediate consequence of Theorem 7.3.

Corollary 7.4 *For any positive integers i and j, there exists an explicit construction for an infinite class of (i,j)-$DS(v, b)$ in which b is $O\left((\log v)^{\log_2(ij+1)}\right)$.*

7.2 Review of nonconstructive bounds

Non-constructive bounds for non-adaptive group testing algorithms and key distribution patterns have been previously discussed in the mathematical literature. In Section 5.1 we derived some nonconstructive bounds for (i,j)-SS(v,b) in which $b \approx \gamma \log_2 v$ for a specified constant γ. In view of Theorem 6.5, we immediately obtain a bound for disjunct systems: an (i,j)-DS(v,b) exists in which $b \approx 2\gamma \log_2 v$. However, a more direct approach is used in [13] (in the case $i \geq 2$) and in [11, Theorem 4.3.9] (in the case $i = 1$; see also [12, Corollary 2]). This approach, which is a simple modification of the one we described in Section 5, yields smaller constants. In this section, we briefly review the nonconstructive bounds that can be obtained in this way.

We will actually construct an $N \times n$ matrix which satisfies the conditions of Lemma 6.4. Let A be an $N \times n$ $0-1$ matrix whose columns are labelled $1, \ldots, n$. Suppose that $C_1, C_2 \subseteq \{1, \ldots, n\}$, $|C_1| = i$, $|C_2| = j$ and $C_1 \cap C_2 = \emptyset$. Define $X_A(C_1, C_2) = 0$ if there exists a row of A such that the entries in the columns in C_1 are all "1"s and the entries in the columns in C_2 are all "0"s, and define $X_A(C_1, C_2) = 1$, otherwise.

Let $X(C_1, C_2)$ denote the random variable obtained when A is an $N \times n$ matrix in which each entry is defined to be a "1" with probability ρ. (The optimal value of ρ will be chosen a bit later.) Then, the expected value of $X(C_1, C_2)$ is

$$E[X(C_1, C_2)] = \left(1 - \rho^i (1-\rho)^j\right)^N.$$

Since we want to minimize $E[X(C_1, C_2)]$, elementary calculus shows that we should take $\rho = i/(i+j)$. Now, if we define the random variable

$$X = \sum_{\{C_1, C_2 \subseteq \{1,\ldots,n\} : |C_1| = i, |C_2| = j, C_1 \cap C_2 = \emptyset\}} X(C_1, C_2),$$

then it is easy to see that

$$\begin{aligned} E[X] &= \binom{n}{i}\binom{n-i}{j}\left(1 - \rho^i (1-\rho)^j\right)^N \\ &= \binom{n}{i}\binom{n-i}{j}\left(1 - \frac{i^i j^j}{(i+j)^{i+j}}\right)^N. \end{aligned}$$

We obtain the following theorem which is analogous to Theorem 5.2.

Theorem 7.5 *Suppose that n, i and j are positive integers. Define*

$$p = 1 - \frac{i^i j^j}{(i+j)^{i+j}}.$$

Then the following hold:

1. *If*

$$N > \frac{(i+j)\log n}{-\log p},$$

then there exists an (i,j)-DS(n, N).

2. *If*

$$N > \frac{(i+j-1)\log(2n)}{-\log p},$$

then there exists an (i,j)-DS(n, N).

7.2.1 An example

We illustrate the application of the above bounds in a particular case, namely $i = 1$ and $j = 2$. Part 2. of Theorem 7.5 shows the existence of a $(1,2)$-DS$(n, \approx 8.646 \log_2 n)$. This compares with $(1,2)$-DS$(n, \approx 9.638 \log_2 n)$ that would be obtained by applying Theorem 5.2 and Theorem 6.5. In [13], it is shown that there is a $(1,2)$-DS$(n, \approx 12.97 \log_2 n)$; this corresponds to part 1. of Theorem 7.5. In fact the best known result in this particular case is the existence of a $(1,2)$-DS$(n, \approx 5.481 \log_2 n)$, which is shown in [14, 12] by a more refined argument.

Acknowledgment

The authors' research is supported by NSF grant CCR-9610138.

References

[1] N. Alon and J. H. Spencer. *The Probabilistic Method*, John Wiley & Sons, 1992.

[2] M. Atici, S. S. Magliveras, D. R. Stinson and W.-D. Wei. Some recursive constructions for perfect hash families, *J. Combinatorial Designs* **4** (1996), 353-363.

[3] I. Biehl and B. Meyer. Protocols for collusion-secure asymmetric fingerprinting, *14th Symposium on Theoretical Aspects of Computing* (1997).

[4] S. Blackburn. Combinatorics and threshold cryptography, preprint.

[5] S. Blackburn, M. Burmester, Y. Desmedt and P. R. Wild. Efficient multiplicative sharing schemes, *Lecture Notes in Computer Science* **1070** (1996), 107–118 (Advances in Cryptology – Eurocrypt '96).

[6] D. Boneh and J. Shaw. Collusion-secure fingerprinting for digital data, *Lecture Notes in Computer Science* **963** (1995), 452–465 (Advances in Cryptology – Crypto '95).

[7] K. A. Bush, W. T. Federer, H. Pesotan and D. Raghavarao. New combinatorial designs and their application to group testing, *Journal of Statistical Planning and Inference* **10** (1984), 335-343.

[8] Y. M. Chee. *Turán-type Problems in Group Testing, Coding Theory and Cryptography*, PhD Thesis, University of Waterloo, 1996.

[9] B. Chor, A. Fiat and M. Naor. Tracing traitors, *Lecture Notes in Computer Science* **839** (1994), 257–270 (Advances in Cryptology – Crypto '94).

[10] G. D. Cohen, S. Litsyn and G. Zémor. Greedy algorithms in coding theory, *IEEE Trans. Inform. Theory* **42** (1996), 2053–2057.

[11] D.-Z. Du and F. K. Hwang. *Combinatorial Group Testing and Applications*, World Scientific, 1993.

[12] A. G. Dyachkov, V. V. Rykov and A. M. Rashad. Superimposed distance codes, *Problems of Control and Information Theory* **18** (1989), 237–250.

[13] M. Dyer, T. Fenner, A. Frieze and A. Thomason. On key storage in secure networks, *Journal of Cryptology* **8** (1995), 189–200.

[14] P. Erdös, P. Frankl and Z. Füredi. Families of finite sets in which no set is covered by the union of two others, *Journal of Combinatorial Theory A* **33** (1982), 158–166.

[15] P. Erdös, P. Frankl and Z. Füredi. Families of finite sets in which no set is covered by the union of r others, *Isreal Journal of Mathematics* **51** (1985), 75–89.

[16] P. Erdös and Z. Füredi. The greatest angle among n points in the the d-dimensional Euclidean space, *Annals of Discrete Mathematics* **17** (1983), 275–283.

[17] A. Fiat and M. Naor. Broadcast encryption, *Lecture Notes in Computer Science* **773** (1994), 480–491 (Advances in Cryptology – Crypto '93).

[18] A. D. Friedman, R. Ł. Graham and J. D. Ullman. Universal single transition time asynchronous state assignments, *IEEE Trans. Comput.* **C-18** (1969), 541–547.

[19] M. L. Fredman and J. Komlos. On the size of separating systems and families of perfect hash functions, *SIAM J. Algebriac and Discrete Meth.* **5** (1984), 61-68.

[20] L. Gong and D. L. Wheeler. A matrix key-distribution scheme, *Journal of Cryptology* **2** (1990), 51–59.

[21] F. K. Hwang and V. T. Sós. Non-adaptive hypergeometric group testing, *Studia Sci. Math. Hungar.* **22** (1987), 257–263.

[22] W. H. Kautz and R. G. Singleton. Nonrandom binary superimposed codes, *IEEE Trans. Inform. Theory* **10** (1964), 363–37.

[23] J. Körner and G. Simonyi. Separating partition systems and locally different sequences, *SIAM J. Discrete Math.* **1** (1988), 355–359.

[24] J. Körner. On the extremal combinatorics of the Hamming space, *J. Combin. Theory A* **71** (1995), 112–126.

[25] K. Mehlhorn. On the program size of perfect and universal hash functions, in *Proc. 23rd Annual IEEE Symposium on Foundations of Computer Science*, pp. 170–175.

[26] K. Mehlhorn. *Data Structures and Algorithms*, Vol. 1. Springer-Verlag, 1984.

[27] C. J. Mitchell and F. C. Piper. Key storage in secure networks, *Discrete Applied Mathematics* **21** (1988), 215–228.

[28] B. Pfitzmann. Trials of traced traitors, *Lecture Notes in Computer Science* **1174** (1996), 49-64 (Workshop on Information Hiding).

[29] B. Pfitzmann and M. Schunter. Asymmetric fingerprinting, *Lecture Notes in Computer Science* **1070** (1996), 84-95 (Advances in Cryptology – Eurocrypt '96).

[30] B. Pfitzmann and M. Waidner. Asymmetric fingerprinting for larger collusions, *4th ACM Conference on Computer and Communications Security* (1997).

[31] B. Pfitzmann and M. Waidner. Anonymous fingerprinting, *Lecture Notes in Computer Science* **1233** (1997), 88–102 (Advances in Cryptology – Eurocrypt '97).

[32] C. M. O'Keefe. Key distribution patterns using Minkowski planes, *Designs, Codes and Cryptography* **5** (1995), 261–267.

[33] K. A. S. Quinn. Some constructions for key distribution patterns, *Designs, Codes and Cryptography* **4** (1994), 177–191.

[34] K. A. S. Quinn. Bounds for for key distribution patterns, to appear in *J. Cryptology.*

[35] K. A. S. Quinn. On construction methods for key distribution patterns, preprint.

[36] D. R. Stinson. On some methods for unconditionally secure key distribution and broadcast encryption, *Designs, Codes and Cryptography* **12** (1997), 215–243.

[37] D. R. Stinson and Tran van Trung. Some new results on key distribution patterns and broadcast encryption, to appear in *Designs, Codes and Cryptography.*

[38] D. R. Stinson and R. Wei. Combinatorial properties and constructions of traceability schemes and frameproof codes, to appear in *SIAM J. Discrete Math.*

A Abbreviations used in this paper

FPC	frameproof code	Definition 2.1
SFPC	secure frameproof code	Definition 2.2
SFF	sandwich-free family	Definition 3.1
SS	separating system	Definition 3.2
PHF	perfect hash family	Definition 4.1
SHF	separating hash family	Definition 4.2
CFF	cover-free family	Definition 6.1
DS	disjunct system	Definition 6.2
KDP	key distribution pattern	Definition 6.3
NAGTA	non-adaptive group testing algorithm	Definition 6.4