Perfect Hash Families, Identifiable Parent Property Codes and Covering Arrays

Dissertation
zur Erlangung des Grades
eines Doktors der Naturwissenschaften

dem Fachbereich 6 (Mathematik)
der Universität Duisburg-Essen
vorgelegt von

Sosina Martirosyan
aus Eriwan (Republik Armenien)

im Juli 2003

Die Disputation fand am 15. October 2003 statt.

Vorsitzender:
Prof. Dr. Jürgen Herzog

Gutachter:
Prof. Dr. Trung van Tran
Prof. Dr. Wolfgang Lempken

Zusammenfassung

In letzter Zeit haben einige kombinatorische Strukturen und Codes eine Vielzahl verschiedener Anwendungen in der Kommunikationstechnik, Kryptographie, Netzwerktechnik und der Informatik gefunden.

Der Zweck dieser Dissertation ist, offene Probleme im Zusammenhang mit verschiedenen kombinatorischen Objekten zu lösen, welche durch praktische Anwendungen im Bereich der Informatik und Kryptographie motiviert sind. Genauer gesagt, untersuchen wir perfect hash families, identifiable parent property codes und covering arrays.

Perfect hash families sind kombinatorische Strukturen, die verschiedene praktische Anwendungen haben, so wie Compilerbau, Probleme der Komplexität von Schaltkreisen, Datenbank-Verwaltung, Betriebssysteme, derandomization probabilistischer Algorithmen und broadcast encryption.

Wir konzentrieren uns auf explizite Konstruktionsverfahren für perfect hash families. Erstens liefern wir eine explizite rekursive Konstruktion einer unendlichen Klasse von perfect hash families mit dem besten bekannten asymptotischen Verhalten unter allen ähnlichen, bekannten Klassen. Zum zweiten stellen wir ein neues rekursives Konstruktionsverfahren vor, mit dessen Hilfe man ‘gute’ perfect hash families für kleine Parameter erzeugen kann. Durch diese Methode erhalten wir eine unendliche Klasse von perfect hash families, die eine sehr große Menge von Parameter-Werten abdeckt. Weiterhin leiten wir eine neue untere Schranke für die minimale Anzahl von Hash-Funktionen her. Ein Vergleich der existierenden Schranken zeigt, dass unsere Schranke für einige Parameter-Bereiche schärfer ist als andere bekannte Schranken.

Identifiable parent property codes (IPP) wurden entwickelt für die Anwendung in Verfahren, die urheberrechtlich geschützte digitale Daten gegen unerlaubte Kopien schützen, die gemeinsam von mehreren berechtigten Nutzern hergestellt werden. TA codes sind eine gut erforschte Teilmenge der IPP-Codes. Wir stellen zwei neue Konstruktionen für IPP-Codes vor. Unsere erste Konstruktion bietet eine unendlichen Klasse von IPP-Codes mit dem besten bekannten asymptotischen Verhalten unter allen ähnlichen Klassen in der Literatur. Weiterhin beweisen wir, dass diese Codes ein Verfahren zum Finden von Verrätern mit im Allge-

meinen Laufzeit $O(M)$ erlauben, wobei M die Code-Größe ist. Man beachte, dass vorher außer den TA-Codes keine IPP-Codes mit dieser Eigenschaft bekannt waren. Für einige unendliche Unterklassen dieser Codes kann man sogar noch schnellere Verfahren zum Aufspüren von Verrätern finden, mit Laufzeit $poly(logM)$. Außerdem wird eine neue unendliche Klasse von IPP-Codes konstruiert, die 'gute' IPP-Codes für nicht zu große Werte von n liefert, wobei n die Code-Länge bezeichnet. Diese Klasse von IPP-Codes deckt einen großen Bereich von Parameter-Werten ab. Weiterhin konstruieren wir eine große Klasse von w-TA-Codes, die eine positive Antwort auf ein offenes Existenzproblem geben.

Covering arrays sind von vielen Wissenschaftlern intensiv untersucht worden, aufgrund ihrer zahlreichen Anwendungen in der Informatik, so wie Software- oder Schaltkreis-Testen, switching networks, Datenkompressions-Probleme, und etliche mathematische Anwendungen, so wie Differenz-Matrizen, Such-Theorie und Wahrheits-Funktionen.

Wir untersuchen explizite Konstruktions-Methoden für t-covering arrays. Zuerst benutzen wir den Zusammenhang zwischen perfect hash families und covering arrays, um unendliche Familien von t-covering arrays zu finden, für die wir beweisen, dass sie besser sind als die augenblicklich bekannten probabilistischen Schranken für covering arrays. Diese Familien haben ein sehr gutes asymptotisches Verhalten. Zum zweiten liefern wir, angeregt durch ein Ergebnis von Roux und auch von einem kürzlich erzielten Ergebnis von Chateauneuf und Kreher für 3-covering arrays, verschiedene neue Konstruktionen für t-covering arrays, $t \geq 4$, die als eine Verallgemeinerung dieser Ergebnisse gesehen werden können.

Ich habe diese Arbeit selbständlig verfasst und dabei keine anderen als die in der Literatureliste aufgeführten Hilfsmittel benutzt.

Sosina Martirosyan
Essen, July 2003

Acknowledgment

First of all, I would like to thank my supervisor Professor Tran van Trung for his endless patience in giving me professional advice on mathematics research and writing a dissertation. I am grateful for his support and all I have learned from him during my studies at the Institute for Experimental Mathematics (IEM), University of Essen (now University Duisburg-Essen).

I would further like to thank Professor Han Vinck for all his support and guidance, in particular, regarding my research on optical orthogonal codes. As head of the Digital Communication Group, he has created a pleasant and enjoyable working atmosphere, for which I am especially thankful.

My research has been funded by the German Research foundation DFG as part of the graduate program ”Mathematical and Engineering Methods for Secure Data Transmission and Information Exchange.” I am grateful to DFG and the head of the graduate college, Professor Gerhard Frey, for giving me the wonderful opportunity to concentrate my full effort on doing research and writing a dissertation.

I thank my father Samvel Martirosyan for his many contributions towards my development as a researcher. His encouragement was the impetus that drove me to become a mathematician. I am grateful to my mother Susanna Hovakimyan whose help made possible to write this thesis.

Finally, special thanks go to my colleagues in the Digital Communication Group at IEM for their support, friendship, and contribution to the pleasant working environment: Lejla Batina, Valér Čanda, Lenka Fibíková, Jürgen Häring, Yuan Luo, Oliver Meili, Chaichana Mitrpant, Birgit Rieth and Tadashi Wadayama.

Contents

Chapter 1

Introduction

Recently, several combinatorial structures and codes have found vast range of applications to communications, cryptography, networking and computer sciences. Connections of error correcting codes and cryptography are surveyed in [1]. The paper [3] describes some applications of coding theory to communication combinatorial problems. Many practical problems where combinatorial designs have played a substantial role are discussed in some survey papers such as applications of combinatorial designs in computer science [2], combinatorial designs and cryptography [4], applications of combinatorial designs to communications, cryptography, and networking [5].

The purpose of this dissertation is to solve open problems related to several combinatorial objects motivated by practical applications in the area of computer sciences and cryptography. Precisely, we study perfect hash families, identifiable parent property codes and covering arrays.

Perfect hash families were introduced by Mehlhorn in compiler design to prove lower bounds on the size of a computer program. In the last few years, perfect hash families have been applied to circuit complexity problems, database management, operating systems, derandomization of probabilistic algorithms and broadcast encryption. More recently, they found applications in secret sharing, key distribution patterns, non-adaptive group testing algorithms, in constructing cryptographic codes, covering arrays and efficient multicast re-keying schemes.

An (n, M, m)-*hash family* is a set $\mathcal{H}$ of functions $\{h : A \to B\}$, where $|A| = M$, $|B| = m$ and $|\mathcal{H}| = n$. An (n, M, m, w)-*perfect hash family* is an (n, M, m)-hash family such that for any $X \subseteq A$ with $|X| = w$, there is at least one function $h \in \mathcal{H}$ such that h is injective when restricted on X.

The main problem is to minimize the number of hash functions. Numerous upper and lower bounds have been derived on the minimal number n for which an (n, M, m, w)-perfect hash family exists. It is proved that n is $\Theta(\log M)$ for

any fixed m and w. Such existence results are of probabilistic nature and it turns out to be a difficult problem to give explicit constructions, which are as good asymptotically. Several explicit constructions of perfect hash families from error correcting codes and incomplete block designs are derived which yield perfect hash families with $n = O(\log M)$. However these constructions are restricted since they do not provide perfect hash families for large n in respect to the alphabet size m. The known explicit recursive constructions give perfect hash families where n is a polynomial function of $\log M$.

We focus on explicit construction techniques for perfect hash families. Firstly, we provide an explicit recursive construction of an infinite class of perfect hash families with the best asymptotic behavior among similar known classes. Secondly, we present a new recursive construction technique which allows to construct 'good' perfect hash families for small sizes of n. Applying this construction we obtain an infinite class of perfect hash families covering a very large set of parameter values. Further, using a rather simple method we obtain a new lower bound on the minimal number of hash functions. A comparison of the existing bounds shows that our bound is stronger than other known bounds for some parameter sets. It is better than Fredman-Komlós and Körner-Marton bounds almost everywhere.

Identifiable parent property (IPP) codes have been introduced by Hollmann, van Lint, Linnartz and Tolhuizen in 1997. These codes are designed to be used in the schemes that protect copyrighted digital data against illegal reproduction or redistribution. 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 goal of such schemes is to handle as many colluders as possible. The practical applications require to accommodate many users when there is a restriction on the number of symbols which can be used for marking the data.

We say a code has 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. TA codes are well studied subsets of IPP codes. The TA property ease the parent identification process allowing efficient traitor tracing algorithms. Combinatorial properties of IPP codes and TA codes have been studied by several authors. Relationships of IPP codes with other known combinatorial structures and codes lead to several sufficient and necessary conditions on the existence of IPP codes. Probabilistic techniques are used to prove the existence of w-IPP codes with $n = O(\log M)$, where n is the length of the codes and M is the size, for any alphabet of size $q > w$. During the last few years several explicit constructions of IPP codes have been derived. Certain classes of TA codes are shown to have a fast traitor tracing algorithm by using the list decoding techniques.

We present two new explicit construction methods for IPP codes using recursion techniques. Our first construction provides an infinite class of IPP codes with the best asymptotic behavior among explicitly constructed classes of IPP codes known in the literature. In fact, for any fixed $q > w$ we are able to construct an infinite class of w-IPP codes in which the length n of the codewords is $O((w^2)^{\log^*(M)}(\log(M))$, where M is the number of codewords and log^* is a very slow growing function. Moreover, we prove that these codes allow a traitor tracing algorithm with a runtime of $O(M)$ in general. It should be noted that no IPP codes other than the TA codes with this property were known before. For some infinite subclasses of these codes, even faster traitor tracing algorithms with runtime $poly(\log M)$ can be obtained. Also, another new infinite class of IPP codes is constructed which provides 'good' IPP codes for small values of n. This class of IPP codes covers a wide range of parameter values. The known construction methods and probabilistic existence results do not prove the existence of w-TA codes with $q < w^2$, and $b > q$, where b is the size of the code and q is the size of the alphabet. Thus existence of such w-TA codes is stated as an open problem by Staddon, Stinson, and Wei. We provide a positive answer to the problem by constructing a large class of w-TA codes with $q < w^2$ and $b > q$.

Covering arrays have undergone an intensive survey by many researchers due to their numerous applications in computer science such as software or circuit testing, switching networks, data compression problem, and also several mathematical applications such as difference matrices, search theory and truth functions.

The application of covering arrays to software system testing is discussed in many papers. One of the approaches to reduce costs for testing a software system is to use combinatorial designs to generate an efficient test set. Software system faults are often caused by interactions among components. The goal of a software developer is to test all combinations of potential interactions with small number of tests. For the system where most errors occur because of interactions of its maximum t components, a test plan can be designed using t-covering arrays.

We present new explicit construction methods for t-covering arrays. Firstly, using the relationships between perfect hash families and covering arrays we can construct infinite families of t-covering arrays which are proved to be better than currently known probabilistic bounds for covering arrays. These families have very good asymptotic behavior. Secondly, inspired from a result of Roux and also from a recent result of Chateauneuf and Kreher for 3-covering arrays, we show several constructions for t-covering arrays, which can be viewed as generalizations of their results for t-covering arrays, $t \geq 4$. These constructions are more efficient than the other known constructions when the size of array is not very large.

Thesis Overview

Chapter 2 is a brief survey on q-ary code and introduces some significant known results which are needed in the subsequent chapters. In Section 2.7 we present a new construction for q-ary codes with large Hamming distance which is used in next chapters to give new classes of traceability codes and perfect hash families. It is also shown that the construction produces a class of optimal q-ary codes in the sense that parameters of the codes achieve the Plotkin bound.

Chapter 3 studies perfect hash families. A new necessary condition on the existence of perfect hash families is derived in Section 3.4, which is expressed in form of an upper bound on M. A comparison of some known bounds is provided which shows that our bound is stronger than other known bounds for some parameters sets. Two new explicit recursive constructions are described in Section 3.8. Firstly, we provide an explicit recursive construction of an infinite class of perfect hash families with very good asymptotic behavior. Secondly, we present a new construction technique which provides an infinite class of perfect hash families covering very large parameter ranges.

Chapter 4 introduces identifiable parent property codes (IPP). We describe two new explicit construction techniques for IPP codes. Our first construction given in Section 4.5 shows an infinite class of IPP codes with very good asymptotic behavior of parameters. We also prove that for an infinite subclass of these codes a traitor tracing algorithm with a runtime $poly(\log M)$ exists. Using our second construction, given in Section 4.6, we obtain a new infinite class of IPP codes covering a very large set of parameters. Further, in Section 4.7, we construct a large class of w-TA codes and thus give a positive answer to an open problem on existence of w-TA codes.

Chapter 5 presents covering arrays. In Section 5.3 using the relationships between perfect hash families and covering arrays we obtain infinite families of t-covering arrays which are proved to be better than currently known probabilistic bounds. These families have a very good asymptotic behavior. Section 5.4 includes several new constructions for t-covering arrays, $t \geq 4$, which result in "good" covering arrays of small sizes.

Chapter 2

q-ary Codes

This chapter is a survey on q-ary codes. Codes over an alphabet of size q, called q-ary codes, are well studied objects in the theory of error-correcting codes [9, 10, 13, 26]. We recall some known results and definitions of the theory of error-correcting codes. Our goal is to present some conceptions and techniques which are used in the other chapters. In Section 2.1 we present preliminaries. Some know necessary and sufficient conditions on existence of q-ary codes are surveyed in Sections 2.2 and 2.3. Several known codes and combinatorial structures are presented in Sections 2.4, 2.5 and 2.6.

We present a new construction for q-ary codes with large Hamming distance in Section 2.7 (see also [71]). These codes are used later to obtain new classes of perfect hash families and TA codes in Chapters 3 and 4. We show that the construction produces a class of optimal q-ary codes in the sense that parameters of the codes achieve the Plotkin bound. Finally, the decoding problem is discussed in Section 2.8.

2.1 Definitions

In this section we present some basic definitions on error correcting codes.

Let Q be an alphabet of size q and let $\mathcal{C} \subseteq Q^n$. Then $\mathcal{C}$ is called a q-ary code of length n. If $|\mathcal{C}| = M$, then we call $\mathcal{C}$ an (n, M, q) code. M is called the size of the code and the elements of $\mathcal{C}$ are called *codewords* and each codeword will have the form $x = (x_1, \ldots, x_n)$, where $x_i \in Q,\ 1 \leq i \leq n$.

A (n, M, q) code $\mathcal{C}$ can be depicted as an $M \times n$ matrix C on q symbols, where each row of the matrix corresponds to one of the codewords.

When Q is a field, $\mathcal{C}$ is called a *linear code* if it is a vector subspace of Q^n.

The dimension k of a linear code $\mathcal{C}$ is defined to be the dimension of $\mathcal{C}$ as a vector space over Q. Notice that if Q is a finite field with q elements, then for an

(n, M, q) linear code $k = \log_q M$. For a linear (n, M, q) code of dimension k we will use the notation $[n, k, q]$.

Let $\mathcal{C}$ be a linear code of length n and dimension k over Q. Denote by G the $k \times n$ matrix whose rows are k basis vectors of $\mathcal{C}$. Then G is called a *generator matrix* for $\mathcal{C}$ and $\mathcal{C} = \{uG | u \in Q^k\}$.

Example 2.1.1 *A generator matrix of a [13, 7, 3] code.*

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

Let $x = (x_1, x_2, \ldots, x_n)$ and $y = (y_1, y_2, \ldots, y_n)$ be any q-ary vectors of the length n. The *Hamming distance* between x and y is

$$d(x, y) := |\{i | x_i \neq y_i\}|.$$

Definition 2.1.2 *The minimum distance of $\mathcal{C}$ is*

$$d := d_{min}(\mathcal{C}) = \min \{d(x, y) \,|x, y \in \mathcal{C} \text{ and } x \neq y\}$$

The code given in Example 2.1.1 has minimum distance $d = 8$.

We use the notation $(n, M, q; d)$ for an (n, M, q) code with minimum distance d and for a linear code of dimension k we use the notation $[n, k, q; d]$.

2.2 Bounds

A code $\mathcal{C}$ is "good" if both M and d are large with respect to n.

The question is that for fixed n, q and d how large the size of the code M can be. We present here necessary conditions for the existence of an $(n, M, q; d)$ code.

Theorem 2.2.1 *(Singleton bound) Let $\mathcal{C}$ be an $(n, M, q; d)$ code. Then $M \leq q^{n-d+1}$.*

In particular, the bound for linear codes is given by $k \leq n - d + 1$. Any code having parameters which meet the Singleton bound is called an *maximum distance separable (MDS)* code. The main conjecture on *MDS* codes asserts that all *MDS* codes are short in respect to the alphabet size. In practice we are interested in codes which are long in respect to the alphabet size. The Singleton bound is not sharp for long codes.

An other bound called the Plotkin bound is given in this form in [29].

Theorem 2.2.2 *(Plotkin bound) If there exists an* $(n, M, q; d)$ *code, then*

$$M(M-1)d \leq 2n \sum_{i=0}^{q-2} \sum_{j=i+1}^{q-1} M_i M_j,$$

where $M_i = \lfloor (M+i)/q \rfloor$.

A weaker form of this bound (see for example [10], p. 58) performed as an upper bound on the minimum distance d is useful for our discussion in the next chapters.

Corollary 2.2.3 *If there exists a n* $(n, M, q; d)$ *code, then*

$$d \leq n\left(1 - \frac{1}{q}\right)\frac{M}{M-1}.$$

In Section 2.7 we present a new construction technique for $(n, M, q; d)$ codes, which produces examples of codes parameters of which meet the Plotkin bound.

2.3 Existence Result

Now we state a lower bound on the code size presenting an existence result.

Theorem 2.3.1 *(Gilbert-Varshamov bound)*

There exists an $(n, M, q; d)$ *code, where*

$$M \geq \frac{q^n}{\sum_{i=0}^{d-1} \binom{n}{i} (q-1)^i}.$$

The existence result in the last theorem is nonconstructive. It is difficult to give explicit constructions which produce codes having as good parameters as in the Gilbert-Varshamov bound.

In next sections we introduce some important classes of q-ary error correcting codes.

2.4 Reed-Solomon Codes

The Reed-Solomon codes are very important and well-studied family of linear codes. These codes employ finite algebra concepts and are suitable for a practical implementation. A spacial case $n = q - 1$ had been introduced by Reed and Solomon in 1960.

Here we give a definition of the Reed-Solomon codes.

Definition 2.4.1 *(Reed-Solomon code(RS))*

Let $\alpha = \{\alpha_1, \alpha_2, \ldots, \alpha_n\}$ where the α_i are distinct elements of F_q, and let $v = \{v_1, v_2, \ldots, v_n\}$ where the v_i are nonzero (but not necessarily distinct) elements of F_q. Then the Reed-Solomon code, denoted by $RS_{n,k}(\alpha, v)$, consists of all vectors

$$(v_1 f(\alpha_1), v_2 f(\alpha_2), \ldots, v_n f(\alpha_n))$$

where $f(x)$ ranges over all polynomials of degree less than k with coefficients from F_q.

Theorem 2.4.2 *Let $\mathcal{C}$ be a Reed-Solomon code defined above, then $\mathcal{C}$ is a linear code over F_q with length $n \leq q$ and $d = n - k + 1$ provided $k \neq 0$ denoted $[n, k, q, d] - RS$. In particular, RS codes are MDS codes.*

Choose the following particular polynomial basis $1, x, \cdots, x^i, \cdots, x^{k-1}$. In this basis the generator matrix of $RS_{n,k}(\alpha, v)$ is the following:

$$\begin{bmatrix} v_1 & v_2 & \cdots & v_j & \cdots & v_n \\ v_1\alpha_1 & v_2\alpha_2 & \cdots & v_j\alpha_j & \cdots & v_n\alpha_n \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ v_1\alpha_1^i & v_2\alpha_2^i & \cdots & v_j\alpha_j^i & \cdots & v_n\alpha_n^i \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ v_1\alpha_1^{k-1} & v_2\alpha_2^{k-1} & \cdots & v_j\alpha_j^{k-1} & \cdots & v_n\alpha_n^{k-1} \end{bmatrix}.$$

The length of a Reed-Solomon code unfortunately cannot exceed q. It is known that these codes can be naturally extended to codes on the projective line with $k + d = n + 1$, $n \leq q + 1$.

Now consider the code $\mathcal{C}$ whose generator matrix results in adding a new column to the generator matrix for $RS_{n,k}(\alpha, v)$ given above:

$$\begin{bmatrix} v_1 & v_2 & \cdots & v_j & \cdots & v_n & 0 \\ v_1\alpha_1 & v_2\alpha_2 & \cdots & v_j\alpha_j & \cdots & v_n\alpha_n & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\ v_1\alpha_1^i & v_2\alpha_2^i & \cdots & v_j\alpha_j^i & \cdots & v_n\alpha_n^i & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\ v_1\alpha_1^{k-2} & v_2\alpha_2^{k-2} & \cdots & v_j\alpha_j^{k-2} & \cdots & v_n\alpha_n^{k-2} & 0 \\ v_1\alpha_1^{k-1} & v_2\alpha_2^{k-1} & \cdots & v_j\alpha_j^{k-1} & \cdots & v_n\alpha_n^{k-1} & 1 \end{bmatrix}. \tag{2.1}$$

Definition 2.4.3 *(Extended Reed-Solomon Code (ERS)). The code $\mathcal{C}$ with generator matrix (2.1) is called extended Reed-Solomon code.*

Theorem 2.4.4 *Let $\mathcal{C}$ be an Extended Reed-Solomon code defined above, then $\mathcal{C}$ is a [n, k, q; d] linear code over F_q with length $n \leq q+1$ and $d = n-k+1$ provided $k \neq 0$. In particular, ERS codes are MDS codes.*

The Reed-Solomon codes have found a wide range of applications in digital communications and storage. These codes are used to correct errors in many systems including storage devices (tape, Compact Disk, DVD, barcodes, etc.), wireless or mobile communications (cellular telephones, microwave links, etc.), satellite communications digital television, high-speed modems etc. [25]

2.5 Connections of MDS Codes and Orthogonal Arrays

Orthogonal arrays are well studied combinatorial structures which are in fact generalization of MDS codes. [17, 23]

Definition 2.5.1 *(t-orthogonal array) Let C be a $n \times M$ matrix on v symbols such that each $t \times M$-subarray contains each ordered t-tuple of symbols in exactly λ times as a column. Then C is called a t-orthogonal array, denoted $\mathsf{OA}_\lambda(t, n, v)$. In this case we have $M = \lambda v^t$.*

Example 2.5.2 *The array* $\mathsf{C}^\top$ *is an* $\mathsf{OA}_2(2,7,3)$

$$\mathsf{C} = \left[\begin{array}{ccccccc}
0&0&0&0&0&0&0\\
1&1&1&1&1&1&0\\
2&2&2&2&2&2&0\\
0&0&1&2&1&2&0\\
1&1&2&0&2&0&0\\
2&2&0&1&0&1&0\\
0&1&0&2&2&1&1\\
1&2&1&0&0&2&1\\
2&0&2&1&1&0&1\\
0&2&2&0&1&1&1\\
1&0&0&1&2&2&1\\
2&1&1&2&0&0&1\\
0&1&2&1&0&2&2\\
1&2&0&2&1&0&2\\
2&0&1&0&2&1&2\\
0&2&1&1&2&0&2\\
1&0&2&2&0&1&2\\
2&1&0&0&1&2&2
\end{array}\right]$$

(C *is obtained by Addelman-Kempthorne construction).*

Note that if $\lambda = 1$ then $\mathsf{OA}_1(t,n,v)$ is an $(n, v^t, v; d)$ code where $d = n-t+1$ (consider the transpose of the orthogonal array and the fact that any two rows of the transposition matrix agree in at most $t-1$ positions). Thus it is a MDS code.

Now let C be the corresponding matrix of an MDS $(n, v^t, v; d)$ code. Then transpose matrix of C denote $\mathsf{C}^\top$ is an $\mathsf{OA}_1(t,n,v)$ as each t rows of the matrix contain each ordered t-tuple of symbols in exactly one time as a column (otherwise $d < n-t+1$ which would give a contradiction to the definition of a MDS code). Thus $\mathsf{OA}_1(t,n,v)$ and MDS codes defined above are identical.

The alphabet size q of MDS codes given in Theorem 2.4.4 is a prime power. The following result is proved for orthogonal arrays with $\lambda = 1$ (see also [17] p.181).

Theorem 2.5.3 *[16] For any* t *and* n *with* $2 \le t \le n$*, there is a number* $e_0 = e_0(t,n)$ *such that for any positive number* v *and any prime power* q *there is an orthogonal array* $\mathsf{OA}(t,n,v.q^e)$ *for all* $e \ge e_0$*.*

We describe an explicit family of MDS codes with an alphabet size not necessarily a prime power.

We first describe a simple construction for q-ary codes which has been presented by Bush (1952) [6] for orthogonal arrays. Let $A \subseteq Q_1^n$ be an (n, M_1, q_1) code with minimum distance d_1 and $|Q_1| = q_1$, and let $B \subseteq Q_2^n$ be an (n, M_2, q_2) code with minimum distance d_2 and $|Q_2| = q_2$. Let $Q = Q_1 \times Q_2$. We define a code $\mathcal{C}$ over alphabet Q as follows: For any pair of codewords $\mathbf{a} = (a_1, \ldots, a_n) \in A$ and $\mathbf{b} = (b_1, \ldots, b_n) \in B$ we construct a vector

$$\mathbf{c}(\mathbf{a}, \mathbf{b}) = ((a_1, b_1), \ldots, (a_n, b_n)) \in Q^n.$$

Then it is easy to verify that $C = \{\mathbf{c}(\mathbf{a}, \mathbf{b}) : \mathbf{a} \in A, \mathbf{b} \in B\} \subseteq Q^n$ is an (n, M_1M_2, q_1q_2) code with minimum distance $d = \min\{d_1, d_2\}$.

Thus we have the following result:

Theorem 2.5.4 *Suppose there exist (n, M_1, q_1) code and (n, M_2, q_2) code with minimum distance d_1 and d_2, respectively. Then there exists an (n, M_1M_2, q_1q_2) code with minimum distance $d = \min\{d_1, d_2\}$.*

Theorem 2.5.4 can be used to construct MDS codes for which q is not a prime power. In fact, in the language of orthogonal arrays an (n, M, q) MDS code with minimum distance d is an $\mathsf{OA}_1(n - d + 1, n, q)$; here we have $M = q^{n-d+1}$. We record this special case of the Bush construction in the following theorem:

Theorem 2.5.5 *The existence of (n, q_1^t, q_1) and (n, q_2^t, q_2) MDS codes having the same minimum distance $d = n - t + 1$ implies the existence of an $(n, (q_1q_2)^t, q_1q_2)$ MDS code with minimum distance d.*

As a consequence of Theorem 2.5.5, we have the following corollary.

Corollary 2.5.6 *For any integer $n \geq 2$ and s with a prime factorization $s = p_1^{e_1} \ldots p_r^{e_r}$ such that $n \leq p_i^{e_i} + 1$, $i = 1, \ldots, r$, there is an (n, s^t, s) MDS code, for all $2 \leq t \leq n$.*

Proof: The corollary follows from the existence of $(n, (p_i^{e_i})^t, (p_i^{e_i}))$ MDS (Reed -Solomon) codes for $i = 1, \ldots, r$ (see Theorem 2.4.4). ■

2.6 Algebraic Geometry Codes

In 1977, using algebraic curves over finite fields, V. D. Goppa defined a large class of codes, called *Goppa codes or Algebraic geometry codes (AG)*. The asymptotic performance of these codes exceeds the Gilbert-Varshamov bound for $q \geq 49$. Definitions and basic properties of these codes can be found in [12, 14, 22].

Later in 1982 Tsfasman, Vladut, and Zink, using sequences of modular curves, construct algebraic geometry codes with asymptotic performance giving improvement upon the Gilbert-Varshamov bound for the case q is a perfect square and bigger than 25. In [11] an algorithm for constructing these codes is given, which has complexity $O(n^{30})$. The complexity is of course too high for the practical purpose.

In [15, 19] Garcia and Stichtenoth give an explicit description for sequences of algebraic curves. The AG codes constructed on these curves have better performance than Gilbert-Varshamov bound.

The known low-complexity algorithm for constructing "one-point" AG codes on G-S curves has a runtime upper-bounded by $(n \log_q n)^3$, where n the length of the code and the complexity is measured in terms of multiplications and divisions over the finite field $\mathbb{F}_{q^2}$ [28].

We describe a class of linear AG codes defined on the Garcia-Stichtenoth (G-S) curves [15, 19] below. The lth curve $\mathcal{X}_l$ over $\mathbb{F}_{q^2}$ in the sequence of Garcia-Stichtenoth curves is defined by the equations

$$x_i^q + x_i = \frac{x_{i-1}^q}{x_{i-1}^{q-1} + 1}, \quad i = 1, 2, \ldots, l.$$

The number of rational points of $\mathcal{X}_l$ is more than $q^l(q^2 - q)$ and the genus g_l of $\mathcal{X}_l$ is less than q^{l+1}. The "one-point" AG codes constructed on the G-S curve is as follows: Let $\mathcal{P} = \{P_1, \ldots, P_n, P\}$ be $n + 1$ distinct $\mathbb{F}_{q^2}$-rational points and let $L(mP)$ be the $\mathbb{F}_{q^2}$-vector space consisting of all functions defined on the curve such that the only pole of any $f \in L(mP)$ is P and the pole order is at most m. Define an evaluation map

$$\theta : L(mP) \longrightarrow \mathbb{F}_{q^2}^n$$
$$f \mapsto (f(P_1), \ldots, f(P_n)).$$

Then, the image $\mathcal{C} = Im\theta$ is referred to as a "one-point" AG code. Now, take

$$n = q^l(q^2 - q),$$

$$2g_l - 2 < m < n.$$

Then $\mathcal{C}$ is a linear code with parameters $(n, q^{2k}, q^2; d)$, where $k = m - g_l + 1$ and $d \geq q^l(q^2 - q) - m$. Thus, $q^{l+1} \leq k \leq q^{l+2} - 2q^{l+1} + 1$. We will write $k = \lceil uq^{l+1} \rceil$, where u is a real number satisfying $1 \leq u \leq q - 2$. So, $d \geq q^l(q^2 - q) - \lceil (u + 1)q^{l+1} \rceil + 2$.

The parameters of $\mathcal{C}$ are then

$$(q^l(q^2 - q), q^{2\lceil uq^{l+1} \rceil}, q^2; d).$$

We rewrite it in the next theorem.

Theorem 2.6.1 *[27] For any prime power q there exists a linear code $[q^l(q^2-q), q^{2\lceil uq^{l+1}\rceil}, q^2; d]$ where u is a real number satisfying $1 \leq u \leq q-2$ and $d \geq q^l(q^2-q)-\lceil(u+1)q^{l+1}\rceil+2$. Furthermore, the runtime of a construction algorithm of a generator matrix for such a code is $O(n^3)$.*

2.7 New Construction of $(n, M, q; d)$ Codes

We depict an $(n, M, q; d)$ code $\mathcal{C}$ as an $M \times n$ array $\mathcal{A}(\mathcal{C})$ on q symbols, where each row of the array corresponds to one of the codewords of $\mathcal{C}$. For any $a \in Q$, define

$$m_j(a) = |\{i : \mathcal{A}(\mathcal{C})(i, j) = a\}|,$$

i.e., $m_j(a)$ is the frequency of a on the j^{th} column of $\mathcal{A}(\mathcal{C})$. Define

$$m(\mathcal{C}) = \max_{1\leq j\leq n, a\in Q}(m_j(a)).$$

Definition 2.7.1 *Let $\mathcal{C}$ be an $(n, M, q; d)$ code. We say that $\mathcal{C}$ has an σ-resolution if the codewords of $\mathcal{C}$ can be partitioned into s subsets $A_1, \ldots, A_s$, where $|A_i| = \sigma$, for $i = 1, \ldots, s$, in such a way that each A_i is a code of minimum distance equal to n, i.e., any two codewords of A_i agree in no position.*

2.7.1 Construction

Let $\mathcal{C}_1$ be an $(n_1, M_1, q_1; d_1)$ code over an alphabet Q_1. Let $\mathcal{C}_2$ be an $(n_2, M_2, q_2; d_2)$ code with a σ-resolution $A_1, \ldots, A_s$. Suppose $s \geq m(\mathcal{C}_1)$. For each $a \in Q_1$ denote by $\mathcal{C}_2(a)$ a copy of $\mathcal{C}_2$ defined over an alphabet $Q(a)$ such that $Q(a_1) \cap Q(a_2) = \emptyset$ if $a_1 \neq a_2$. Denote by $A_1(a), \ldots, A_s(a)$ a σ-resolution of $\mathcal{C}_2(a)$.

Let $col_j = (a_{1,j}, a_{2,j}, \ldots, a_{M_1,j})^T$ be the j^{th} column of $\mathcal{A}(\mathcal{C}_1)$, $1 \leq j \leq n_1$. Let $a(1), \ldots, a(t)$, say, be t positions of col_j at which symbol $a \in Q_1$ appears. Note that $t \leq m(\mathcal{C}_1)$. Now replace a at position $a(1)$ by $A_1(a)$, a at position $a(2)$ by $A_2(a)$, etc., and a at position $a(t)$ by $A_t(a)$. Perform this process for every symbol of Q_1 and for every column of $\mathcal{A}(\mathcal{C}_1)$. The resulting code $\mathcal{C}$ obtained by this replacement has parameters $(n_1n_2, \sigma M_1, q_1q_2; n_1n_2 - (n_1 - d_1)(n_2 - d_2))$.

Obviously, the length and the number of codewords of $\mathcal{C}$ is n_1n_2 and σM_1 respectively. Further, any two codewords $c_1, c_2 \in \mathcal{C}$ agree in at most $(n_1 - d_1)$ positions. After replacement c_1 and c_2 correspond to two subsets R_1 and R_2 of σ codewords each. Any two codewords in R_1 (resp. R_2) agree in no position,

whereas a codeword from R_1 and a codeword from R_2 agree in at most $(n_1 - d_1)(n_2 - d_2)$ positions. Hence the minimum distance of $\mathcal{C}$ is $n_1n_2 - (n_1 - d_1)(n_2 - d_2)$, as stated.

Further, if $q_1q_2 \geq M_1$ then $\mathcal{C}$ can be extended to a code $\mathcal{C}^*$ having parameters $(n_1n_2 + 1, \sigma M_1, q_1q_2; d)$, where $d = \min\{n_1n_2, n_1n_2 + 1 - (n_1 - d_1)(n_2 - d_2)\}$. Let $Q = \{a_1, a_2, \ldots, a_{q_1q_2}\}$ be the alphabet of $\mathcal{C}$ and let $\mathcal{C}_1 = \{c_1, c_2, \ldots, c_{M_1}\}$.

By construction, any codeword $c_i \in \mathcal{C}_1$ corresponds to a subset R_i of σ codewords. For any $i = 1, \ldots, M_1$, we add symbol a_i to the $(n_1n_2 + 1)^{th}$ column of each codeword of R_i. This forms a set R_i^*. The collection of all R_i^* forms an $(n_1n_2 + 1, \sigma M_1, q_1q_2; d)$ code $\mathcal{C}^*$ with $d = \min\{n_1n_2, n_1n_2 + 1 - (n_1 - d_1)(n_2 - d_2)\}$. This can be seen as follows: Any two codewords $x*$ and $y*$ of $\mathcal{C}^*$ belong either to some R_i^* or to two different R_i^* and R_j^*. In the first case their distance is n_1n_2 because their components agree only at the $(n_1n_2 + 1)^{th}$ column, and in the second case their distance is at least $n_1n_2 + 1 - (n_1 - d_1)(n_2 - d_2)$ because their components at the $(n_1n_2 + 1)^{th}$ column are distinct.

We record the result of the construction in the following theorem:

Theorem 2.7.2 *Suppose there is an $(n_1, M_1, q_1; d_1)$ code $\mathcal{C}_1$ and there is an $(n_2, M_2, q_2; d_2)$ code $\mathcal{C}_2$ with a σ-resolution $A_1, \ldots, A_s$ such that $s \geq m(\mathcal{C}_1)$. Then the following hold.*

(i) There is an $(n_1n_2, \sigma M_1, q_1q_2; n_1n_2 - (n_1 - d_1)(n_2 - d_2))$ code $\mathcal{C}$.

(ii) Further, if $q_1q_2 \geq M_1$, then $\mathcal{C}$ can be extended to a code $\mathcal{C}^*$ having parameters $(n_1n_2 + 1, \sigma M_1, q_1q_2; d)$, where $d = \min\{n_1n_2,\ n_1n_2 + 1 - (n_1 - d_1)(n_2 - d_2)\}$.

2.7.2 Example

We illustrate the construction in Theorem 2.7.2 by the following example.

Example 2.7.3 Let $\mathcal{C}_1$ be a $(3, 4, 2; 2)$ code over the alphabet $Q_1 = \{\mathbf{0}, \mathbf{1}\}$ given by

$$\mathcal{C}_1 = \begin{matrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} & \mathbf{1} \\ \mathbf{1} & \mathbf{0} & \mathbf{1} \\ \mathbf{1} & \mathbf{1} & \mathbf{0} \end{matrix}$$

Let $\mathcal{C}_2(\mathbf{0})$ be a $(3, 6, 3; 2)$ code on the alphabet $\{1, 2, 3\}$ having a 3-resolution $A_1(\mathbf{0})$ and $A_2(\mathbf{0})$:

$$A_1(\mathbf{0}) = \begin{matrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{matrix} \qquad A_2(\mathbf{0}) = \begin{matrix} 1 & 3 & 2 \\ 2 & 1 & 3 \\ 3 & 2 & 1 \end{matrix}$$

Let $\mathcal{C}_2(\mathbf{1})$ be a copy of $\mathcal{C}_2(\mathbf{0})$ on the alphabet $\{4, 5, 6\}$ with the corresponding 3-resolution

$$A_1(\mathbf{1}) = \begin{matrix} 4 & 5 & 6 \\ 5 & 6 & 4 \\ 6 & 4 & 5 \end{matrix} \qquad A_2(\mathbf{1}) = \begin{matrix} 4 & 6 & 5 \\ 5 & 4 & 6 \\ 6 & 5 & 4 \end{matrix}$$

Replacing entries of $\mathcal{A}(\mathcal{C}_1)$ by $A_i(\mathbf{j})$ gives

$$\begin{matrix} A_1(\mathbf{0}) & A_1(\mathbf{0}) & A_1(\mathbf{0}) \\ A_2(\mathbf{0}) & A_1(\mathbf{1}) & A_1(\mathbf{1}) \\ A_1(\mathbf{1}) & A_2(\mathbf{0}) & A_2(\mathbf{1}) \\ A_2(\mathbf{1}) & A_2(\mathbf{1}) & A_2(\mathbf{0}) \end{matrix}$$

Thus, we obtain a $(9, 12, 6; 8)$ code $\mathcal{C}$. Now, since the condition $q_1 q_2 > M_1$ is satisfied, $\mathcal{C}$ can be extended to a $(10, 12, 6; 9)$ code $\mathcal{C}^*$.

$$\mathcal{C} = \begin{matrix} 1 & 2 & 3 & 1 & 2 & 3 & 1 & 2 & 3 \\ 2 & 3 & 1 & 2 & 3 & 1 & 2 & 3 & 1 \\ 3 & 1 & 2 & 3 & 1 & 2 & 3 & 1 & 2 \end{matrix} \quad \begin{matrix} 1 & 3 & 2 & 4 & 5 & 6 & 4 & 5 & 6 \\ 2 & 1 & 3 & 5 & 6 & 4 & 5 & 6 & 4 \\ 3 & 2 & 1 & 6 & 4 & 5 & 6 & 4 & 5 \end{matrix} \quad \begin{matrix} 4 & 5 & 6 & 1 & 3 & 2 & 4 & 6 & 5 \\ 5 & 6 & 4 & 2 & 1 & 3 & 5 & 4 & 6 \\ 6 & 4 & 5 & 3 & 2 & 1 & 6 & 5 & 4 \end{matrix} \quad \begin{matrix} 4 & 6 & 5 & 4 & 6 & 5 & 1 & 3 & 2 \\ 5 & 4 & 6 & 5 & 4 & 6 & 2 & 1 & 3 \\ 6 & 5 & 4 & 6 & 5 & 4 & 3 & 2 & 1 \end{matrix}$$

$$\mathcal{C}^* = \begin{matrix} 1 & 2 & 3 & 1 & 2 & 3 & 1 & 2 & 3 & 1 \\ 2 & 3 & 1 & 2 & 3 & 1 & 2 & 3 & 1 & 1 \\ 3 & 1 & 2 & 3 & 1 & 2 & 3 & 1 & 2 & 1 \end{matrix} \quad \begin{matrix} 1 & 3 & 2 & 4 & 5 & 6 & 4 & 5 & 6 & 2 \\ 2 & 1 & 3 & 5 & 6 & 4 & 5 & 6 & 4 & 2 \\ 3 & 2 & 1 & 6 & 4 & 5 & 6 & 4 & 5 & 2 \end{matrix} \quad \begin{matrix} 4 & 5 & 6 & 1 & 3 & 2 & 4 & 6 & 5 & 3 \\ 5 & 6 & 4 & 2 & 1 & 3 & 5 & 4 & 6 & 3 \\ 6 & 4 & 5 & 3 & 2 & 1 & 6 & 5 & 4 & 3 \end{matrix} \quad \begin{matrix} 4 & 6 & 5 & 4 & 6 & 5 & 1 & 3 & 2 & 4 \\ 5 & 4 & 6 & 5 & 4 & 6 & 2 & 1 & 3 & 4 \\ 6 & 5 & 4 & 6 & 5 & 4 & 3 & 2 & 1 & 4 \end{matrix}$$

2.7.3 A New Class of $(n, M, q; d)$ Codes

In this section we discuss a concrete application of the construction above. We see that the method is suitable for constructing q-ary codes with large distance, and therefore, by Theorem 4.3.13, for constructing w-TA codes with large w.

We need following definitions:

Definition 2.7.4 *(Latin Square) A Latin square on q symbols is an $q \times q$ array such that each of the q symbols occurs once in each row and in each column. The number q is called the order of the square.*

If $A = (a_{ij})$ and $B = (b_{ij})$ are any two $q \times q$ arrays, the *join* (A, B) of A and B is the $n \times n$ array whose $(i, j)th$ entry is the pair (a_{ij}, b_{ij}).

Definition 2.7.5 *Two Latin squares* A, B *of order q are said to be orthogonal if all the entries in the join of* A *and* B *are distinct.*

Definition 2.7.6 *A set of Latin squares $A_1, \ldots, A_r$ are called mutually orthogonal or a set of MOLS, if A_i and A_j are orthogonal for all $1 \leq i < j \leq r$.*

Theorem 2.7.7 *For any prime power q there exists a set of $(q-1)$ MOLS of order q.*

For other basic facts on MOLS, we refer to [17].

Now we are ready to prove the following theorem:

Theorem 2.7.8 *(i) Let q_0 be a prime power. If there is a set of at least $(q_0 - 1)$ mutually orthogonal Latin squares (MOLS) of order σ, then there is an $(n, M, q; d)$ code with*

$$\begin{aligned} n &= (q_0 + 1)\sigma^m \\ M &= q_0^2\sigma^m \\ q &= q_0\sigma^m \\ d &= (q_0 + 1)\sigma^m - 1, \end{aligned}$$

for any positive integer m.

(ii) There is an $(n, M, q; d)$ code with

$$\begin{aligned} n &= (\ldots(((q_0 + 1)\underbrace{q_1 + 1)q_1 + 1)\ldots q_1 + 1)}_{m} \\ M &= q_0^2 q_1^m \\ q &= q_0 q_1^m \\ d &= n - 1, \end{aligned}$$

where $q_1 \geq q_0$ are prime powers and $m \geq 1$ is an integer.

Proof: Take $\mathcal{C}_0$ to be an $\mathsf{OA}_1(2, q_0 + 1, q_0)$ orthogonal array $\mathcal{A}$, see e.g., [17], i.e., $\mathcal{C}_0$ is a $(q_0 + 1, q_0^2, q_0; q_0)$ extended Reed-Solomon code. The array $\mathcal{A}$ has the property that any symbol appears exactly q_0 times in each column. A remark upon MOLS, which are used here, needs to be made. It is known that any given set of u MOLS $M_1, \ldots, M_u$ can be transformed in such a way that any two rows from different M_i and M_j agree in at most one column. Here, we assume that our MOLS have this property.

(i) Now suppose we have a set of q_0 MOLS $M_1, \ldots, M_{q_0}$ of order σ. In the case that we only have $(q_0 - 1)$ MOLS $M_1, \ldots, M_{q_0-1}$, we will take M_0 to be the $\sigma \times \sigma$ matrix with entries from the σ symbols of the latin squares such that each symbol appears σ times in exactly one row. In either cases, $M_0, M_1, \ldots, M_{q_0-1}$ together form a σ resolution of a $(\sigma, q_0\sigma, \sigma; \sigma - 1)$ code $\mathcal{C}$. Applying Theorem 2.7.2 gives a $((q_0 + 1)\sigma, q_0^2\sigma, q_0\sigma; (q_0 + 1)\sigma - 1)$ code $\mathcal{C}_1$. As each symbol of the alphabet appears in each column of $\mathcal{A}(\mathcal{C}_1)$ q_0 times, Theorem 2.7.2 can be applied to $\mathcal{C}_1$ and $\mathcal{C}$ again. This recursive procedure gives rise to codes in (i).

(ii) If $\sigma = q_1$ $(\geq q_0)$ is a prime power, then there are $q_1 - 1$ MOLS $M_1, \ldots, M_{q_1-1}$ of order q_1. $M_1, \ldots, M_{q_1-1}$ and M_0 together form a code $\mathcal{C}$ with a q_1 resolution. Extend $\mathcal{C}_1$ in (i) to a code $\mathcal{C}_1^*$ by adding one more column, as shown in Theorem 2.7.2. Observe that in $\mathcal{C}_1^*$ a symbol appears q_1 or q_0 times in each column. Thus, we can apply Theorem 2.7.2 to $\mathcal{C}_1^*$ and $\mathcal{C}$. Therefore, if at each step the obtained code is extended before applying Theorem 2.7.2, the resulting code after m steps will have parameters given in (ii). ■

It is worth noting that the construction method in Theorem 2.7.2 can produce good q-ary codes.

Consider, for example, the codes in Theorem 2.7.8 (ii). It is easy to check that if $q_0 = q_1$, the parameters of these codes meet the Plotkin bound presented in Theorem 2.2.2 with equality.

In the case when $q_0 \neq q_1$ we study the following example.

Example 2.7.9 *From Theorem 2.7.8 (ii) we obtain:*

$$\begin{array}{llll} q_0 = 2, & q_1 = 3, & i = 1 & (10, 12, 6; 9) \\ q_0 = 3, & q_1 = 4, & i = 1 & (17, 36, 12; 16) \\ q_0 = 4, & q_1 = 5, & i = 1 & (26, 80, 20; 25) \end{array}$$

The (10,12,6;9) code in the example is optimal. The codes (17,36,12;16) and (26,80,20;25) are 'quasi' optimal because the maximum value for M derived from

the Plotkin bound given in Theorem 2.2.2 is 37 in the first case and 81 in the second case.

2.8 The Decoding Problem

The decoding problem for an $(n, M, q; d)$ code is as follows: For a given vector $x \in Q^n$, find a codeword in $\mathcal{C}$ which has a fixed given Hamming distance to x. The problem of finding efficient decoding algorithm for certain classes of codes is one of main important problems studied in algebraic coding theory. Several efficient decoding algorithms have been derived for certain classes of q-ary codes.

The list decoding problem was introduced independently by Elias [7] and Wozencraft [8] in the late 50's. The notation of list decoding is used for the decoding algorithm for which the goal is to output the list of all codewords within a specified distance from the given arbitrary vector.

The list decoding problem can be described as follows:

Given an $(n, M, q; d)$ code $\mathcal{C}$ and an arbitrary vector $x \in Q^n$. Find all codewords in $\mathcal{C}$ within a specified Hamming distance from x.

In [20, 21], Sudan develops a first efficient list decoding algorithm for Reed-Solomon codes, which has polynomial runtime in the length of the code, $poly(n)$. The method has been improved since then.

In [24, 27] Guruswami and Sudan present efficient list decoding algorithms for Reed-Solomon codes, algebraic-geometric and certain concatenated codes. Their algorithms discover all codewords that lie within some multiple of half the minimum distance from the given vector.

List decoding techniques have found applications for the traceability codes for digital fingerprinting presented in Chapter 4.

Chapter 3

Perfect Hash Families

3.1 Introduction

Perfect hash families (PHF), due to their significant applications in information retrieval, have undergone considerable investigation, see, e.g., [40] and [44] for extensive surveys.

Recently, perfect hash families have found numerous interesting applications to computer sciences and cryptography. Perfect hash families were introduced by Mehlhorn in compiler design to prove lower bounds on the size of a computer program. In the last few years, perfect hash families have been applied to circuit complexity problems, database management, operating systems, derandomization of probabilistic algorithms and broadcast encryption. More recently, they found applications in secret sharing, key distribution patterns, non-adaptive group testing algorithms, in constructing cryptographic codes, covering arrays and efficient multicast re-keying schemes.

Two application examples of perfect hash families, namely codes with traceability property and covering arrays, are studied in the next chapters.

One of the fundamental and most studied problems in computer science is the *dictionary problem*. For a given set X of w keys belonging to a universe $\mathsf{A} = \{1, 2, ..., M\}$, $X \subseteq \mathsf{A}$ it is required to store the key $x \in X$ in some data structure so that the membership queries of the form *'is x in X?'* can be answered quickly.

Perfect hashing is one of the best methods to solve this problem in the static case, when no deletion or insertion of elements in X occurs. An overview of the perfect hashing is given in [40].

A *hash function* is a function $h : \mathsf{A} \longrightarrow \mathsf{B}$, where $|\mathsf{A}| = M \geq |\mathsf{B}| = m$, that map the keys from A into set of integers B. Given a key $x \in \mathsf{A}$, the hash function computes an address, i.e., an integer from B, for the storage of x. The storage

area used to store keys is known as a *hash table.*

The function h is called a perfect hash function for $X \subseteq \mathsf{A}$ if it is injective on X. Thus a perfect hash function transforms each key of X into an unique address in the hash table.

The family of hash functions is called a *perfect hash family* for the universe A if for any subset $X \subseteq \mathsf{A}$ such that $|X| \leq w$, there exists at least one hash function which is perfect on X.

Perfect hash families are used in constructing hash tables see, for example, [33]. When perfect hash families are used, there is no need to know the subset X beforehand as the existence of a perfect hash function for each subset is guaranteed from the definition.

Combinatorial structures of perfect hash families have been studied by several researchers.

Necessary conditions for the existence of a perfect hash family can be found in [32, 35, 37, 43]. We provide a new necessary condition for the existence of a perfect hash family as upper bound on the size M of universe A. This result has been presented in the papers [41], [42] and [45]. A comparison of the existing bounds shows that our bound is stronger than other known bounds for some parameter sets. It is better than Fredman-Komlós and Körner-Marton bounds almost everywhere.

Probabilistic methods are used to obtain sufficient conditions on existence of perfect hash families in [30, 32, 35, 43, 47]. These results together with necessary conditions theoretically state that for any fixed m and w, $m \geq w$ there exists an infinite class of perfect hash families with $n = \Theta(\log M)$, where n is the number of hash functions. However, these existence results are not constructive and it is believed to be a difficult problem to give explicit constructions, which are as good asymptotically. Several explicit constructions have been presented in [34, 37, 38, 43, 46, 48, 49].

We focus on explicit construction techniques for perfect hash families. Firstly, we provide an explicit recursive construction of an infinite class of perfect hash families with the best known asymptotic behavior among similar known classes. Secondly, we present a new construction technique for perfect hash families using mutually orthogonal Latin squares, orthogonal arrays and recursive techniques. As result we obtain an infinite class of perfect hash families covering very large parameter ranges. The first construction has been presented also in the papers [42] and [45] and the second construction in the [72], [82] and [83].

In Section 3.2 we define perfect hash families and present notation and examples. Section 3.3 is a survey on known necessary conditions, upper bounds on the size of universe A. A new upper bound is presented in Section 3.4. Section 3.5 includes comparison of bounds. Several known existence results are given in Sec-

tions 3.6 and 3.7. Some open research problems are discussed. Two new classes of perfect hash families are emphasized in Theorems 3.7.7 and 3.7.8. Our new recursive constructions are described in Section 3.8. As result infinite classes of perfect hash families are obtained. Finally, Section 3.9 summarizes the chapter.

3.2 Definitions

A finite set $\mathcal{H}$ of n functions $h : A \longrightarrow B$, where $|A| = M \geq |B| = m$, is called an (n, M, m)-*hash family*, denoted by $(n, M, m) - HF$.

Definition 3.2.1 *Let M, m, w be integers such that $M \geq m \geq w \geq 2$. An (n, M, m)-hash family $\mathcal{H}$ is called an (n, M, m, w)-perfect hash family , denoted $(n, M, m, w) - PHF$, if for any subset $X \subseteq A$ with $|X| = w$, there is at least one function $h \in \mathcal{H}$ such that h is injective on X.*

Instead of m we will also use the notation q to emphasize the fact that q is a power of a prime number.

An (n, M, q) code $\mathcal{C}$ can be depicted as an $M \times n$ matrix C on q symbols, where each row of the matrix corresponds to one of the codewords. Similarly, an $(n, M, m) - HF$, $\mathcal{H}$, can be presented as an $M \times n$ matrix on m symbols, where each column of the matrix corresponds to one of the functions in $\mathcal{H}$.

Definition 3.2.2 *Let C be an $M \times n$ matrix on m symbols corresponding to an (n, M, m, w)-perfect hash family. Then for any fixed w' rows of C, $w' \leq w$, there is at least one column of C with all distinct symbols on that fixed rows. Matrix C satisfying this property is called a w-separate (M, n, m) matrix.*

It is clear that if there exists a w-separate (M, n, m) matrix then there exists also an (n, M, m, w)-perfect hash family.

Definition 3.2.3 *For any fixed n, m and w denote the maximal value of M for which an $(n, M, m, w) - PHF$ exists by $M(n, m, w)$. For any fixed M, m and w denote the minimal number of hash functions n for which an $(n, M, m, w) - PHF$ exists by $n(M, m, w)$. An (n, M, m, w) perfect hash family is called optimal if $M = M(n, m, w)$.*

For any fixed w and m we are interested in studying the behavior of $M(n, m, w)$ as a function on n.

We present two examples of optimal perfect hash families. In the first example we have an optimal $(4, 9, 3, 3) - PHF$. It is not difficult to check that the matrix given in the example is a 3-separate $(9, 4, 3)$ matrix, i.e., an $(4, 9, 3, 3) - PHF$. The optimality will be proved in Section 3.4.

Example 3.2.4 *An optimal* $(4, 9, 3, 3) - PHF$*, i.e., 3-separate* $(9, 4, 3)$ *matrix.*

$$C = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 2 & 2 \\ 1 & 3 & 3 & 3 \\ 2 & 1 & 2 & 3 \\ 2 & 2 & 3 & 1 \\ 2 & 3 & 1 & 2 \\ 3 & 1 & 3 & 2 \\ 3 & 2 & 1 & 3 \\ 3 & 3 & 2 & 1 \end{bmatrix}$$

The second example presented here was found by computer (Tran van Trung). The optimality of this family also will be proved in Section 3.4.

Example 3.2.5 *An optimal* $(6, 8, 4, 4) - PHF$*, i.e., 4-separate* $(8, 6, 4)$ *matrix.*

$$C = \begin{bmatrix} 1 & 1 & 1 & 2 & 4 & 1 \\ 2 & 1 & 4 & 1 & 3 & 2 \\ 1 & 2 & 3 & 3 & 2 & 3 \\ 2 & 2 & 2 & 4 & 1 & 4 \\ 3 & 3 & 3 & 2 & 1 & 2 \\ 4 & 3 & 1 & 4 & 3 & 3 \\ 3 & 4 & 2 & 1 & 2 & 1 \\ 4 & 4 & 4 & 3 & 4 & 4 \end{bmatrix}$$

3.3 Necessary Conditions

The problem of finding a lower bound on $n(M, m, w)$, i.e, an upper bound on $M(n, m, w)$, has been studied by many authors.

We study here an upper bound on $M(n, m, w)$. We will rewrite some of known lower bounds on $n(M, m, w)$ here as upper bounds on $M(n, m, w)$.

Let C be a 2-separate (M, n, m) matrix. Then C has all distinct rows. On the other hand a matrix with all pairwise different rows is 2-separate. The total number of all distinct vectors of length n with symbols from an alphabet of size m is m^n. Taking the m^n such distinct vectors as rows of a matrix we obtain a 2-separate (M, n, m) matrix, i.e, an $(n, m^n, m, 2) - PHF$. Thus

$$M(n, m, 2) = m^n.$$

So hereafter we study the upper bound on $M(n, m, w)$ for $w \geq 3$.

For any integer a and b, $a \geq b$ define

$$a^{\underline{b}} = \prod_{i=0}^{b-1} (a-i).$$

$f(n)$ and $g(n)$ are functions of positive integers n, which take positive (but not necessary integer) values for all n. We say that $f(n) \preceq g(n)$ if $f(n) \leq (1+0(1))g(n)$, where 0(1) tends to zero when n tends to infinity.

Fredman and Komlòs in [32] have obtained upper bounds on $M(n,m,w)$ which can be expressed in the following form in [35].

Theorem 3.3.1 *[32]*

$$\frac{1}{n}\log M(n,m,w) \preceq \frac{m^{\underline{w-1}}}{m^{w-1}}\log(m-w+2) \tag{3.1}$$

Körner and Marton have established a bound which is stronger than Fredman-Komlòs bound (3.1) for many parameter sets.

Theorem 3.3.2 *[35]*

$$\frac{1}{n}\log M(n,m,w) \preceq \min_{0\leq j\leq w-2} \frac{m^{\underline{j+1}}}{m^{j+1}}\log\left(\frac{m-j}{w-j-1}\right) \tag{3.2}$$

for $j = 0, 1, \cdots, w-2$.

Bound (3.2) matches with bound (3.1) for $j = w-2$. Both bounds are asymptotic in nature. Bound (3.2) as given in [35], is in a different form. After correction of a minor mistake the bound looks as follows in [37]:

Theorem 3.3.3 *[37]*

$$\frac{M(n,m,w)^{\underline{j+1}}}{M(n,m,w)^{j+1}}\log\frac{M(n,m,w)-j}{w-j-1} \leq n \min_{0\leq j\leq w-2} \frac{m^{\underline{j+1}}}{m^{j+1}}\log\frac{m-j}{w-j-1} \tag{3.3}$$

for $j = 0, 1, \cdots, w-2$.

The extreme case $j = 0$ is interesting. In this case the bound looks as follows:

Lemma 3.3.4

$$M(n,m,w) \leq (w-1)\left(\frac{m}{w-1}\right)^n. \tag{3.4}$$

In [43] Blackburn and Wild give an upper bound on $M(n, m, w)$ which is better than the Fredman-Komlòs and the Körner-Marton bounds for many parameter sets. The Blackburn-Wild bound is given in the following theorem (see Theorem *1* in [43]):

Theorem 3.3.5 *[43]*

$$M(n, m, w) \leq (w-1)(m^{\lceil \frac{n}{w-1} \rceil} - 1). \tag{3.5}$$

In particular, when $w > n$ we have:

Corollary 3.3.6 *For* $w > n$

$$M(n, m, w) \leq (w-1)(m-1). \tag{3.6}$$

Corollary 3.3.7 *[43]*
For all sufficiently large integers m *and any real number d with* $d > \lceil \frac{n}{w-1} \rceil$ *we have*

$$M(n, m, w) \leq \lceil m^d \rceil - 1. \tag{3.7}$$

In Theorem *2* in [43] the authors present another bound which is stronger than bound (3.5) for some parameter sets.

Theorem 3.3.8 *[43]*

$$M(n, m, w) \leq \frac{m^{\lceil \frac{n-1}{w-1} \rceil + 1}}{w-1} + w(m^{\lceil \frac{n-1}{w-1} \rceil} - 1) + m - 1. \tag{3.8}$$

Theorems (3.3.5) and (3.3.8) give the following result for $w = 3$:

Corollary 3.3.9 *[43]*

$$M(n, m, 3) \leq 2(m^{\lceil \frac{n}{2} \rceil} - 1) \tag{3.9}$$

$$M(n, m, 3) \leq \frac{m^{\lceil \frac{n+1}{2} \rceil}}{2} + 2m^{\lceil \frac{n-1}{2} \rceil - 1} + m - 4. \tag{3.10}$$

It states in [43] that Fredman-Komlòs bound (3.1) and Körner-Marton bound (3.2) are better than Blackburn-Wild bound (3.5) when w is close to m. But when $n \to \infty$ and m and w are fixed, bound (3.5) is stronger than bounds (3.1) and (3.2) for many values of w and m.

3.4 New Upper Bound

Our main result of this section is given in view of Theorem 3.4.11. To obtain the upper bound, we first prove two lemmas, Lemma 3.4.1 and Lemma 3.4.4. The following lemmas are useful for the discussions in the sequel.[45]

Lemma 3.4.1 *Let* C *be* w*-separate* (n, M, m)*-matrix with* $M = M(n, m, w)$. *Then for an arbitrary* m, $m \geq 3$ *the following relation holds:*

(i) if $w \geq 2n$, *then*

$$M(n, m, w) = m$$

(ii) if $w \geq 2n - 1$, *then*

$$\begin{array}{lllll} M(n,m,w) \leq \lfloor \frac{w+1}{w-1}(m-1) \rfloor & \textit{if} & n-1 & \textit{does not divide} & m-1 \\ M(n,m,w) = \frac{w+1}{w-1}(m-1) & \textit{if} & n-1 & \textit{divides} & m-1 \end{array}$$

(iii) if $w \geq 2n - 2$, *then*

$$M(n, m, w) \leq \begin{cases} \lfloor \frac{w-2}{w-4}(m-1) \rfloor & \textit{if} \quad n > 3 \\ \leq 3m - 6 & \textit{if} \quad n = 3 \end{cases}$$

Proof:

For convenience, we call an element x of the symbol set B, $|B| = m$, a *special element*, if x occurs more than once in a column of the matrix C.

(i): Let $w \geq 2n$. Suppose that $M(n, m, w) > m$. Then, evidently, each column in the matrix C contains at least one special symbol. Choose a pair of such special elements from each column and mark out those rows in matrix C where the chosen elements stand. Clearly, the number of such rows in the matrix is $2n$. Then any w rowed submatrix of C with marked out rows will contain no column whose all elements are distinct. Thus we obtain that the matrix C is not w-separate. This is a contradiction which proves (i).

(ii): Let $w = 2n - 1$. On one hand, the number of special elements in each column of the matrix is not less than $M(n, m, w) - m + 1$. (This number is equal to $M(n, m, w) - m + 1$, if in any column of the matrix there exists only a single element of such kind). On the other hand, any row of the matrix may contain at most one special element, otherwise there will be $2n - 1$

rows of the matrix C which do not contain a column whose all elements are distinct.

Hence, we may write

$$(M(n,m,w)-m+1)n \leq M(n,m,w)$$

or

$$M(n,m,w) \leq \lfloor \frac{n(m-1)}{n-1} \rfloor = \lfloor \frac{w+1}{w-1}(m-1) \rfloor.$$

Now we need to give a construction which achieves this bound to complete the proof of (ii).

Construction: Let $w = 2n-1$ and $(n-1)$ divides $(m-1)$. Then we will take a vector of length $\frac{n(m-1)}{n-1}$ with a single special element in its $(i-1)\frac{m-1}{n-1}+1$ to $i\frac{m-1}{n-1}$ position as i-th $(i = 1, \ldots, n)$ column of the matrix. Naturally, the matrix is w-separate.

Example 3.4.2 *An optimal* $(5, 15, 13, 9)-PHF$*, i.e.,* 9*-separate* $(5, 15, 13)$ *matrix:*

$$C = \begin{bmatrix} 1 & 2 & 2 & 2 & 2 \\ 1 & 3 & 3 & 3 & 3 \\ 1 & 4 & 4 & 4 & 4 \\ 2 & 1 & 5 & 5 & 5 \\ 3 & 1 & 6 & 6 & 6 \\ 4 & 1 & 7 & 7 & 7 \\ 5 & 5 & 1 & 8 & 8 \\ 6 & 6 & 1 & 9 & 9 \\ 7 & 7 & 1 & 10 & 10 \\ 8 & 8 & 8 & 1 & 11 \\ 9 & 9 & 9 & 1 & 12 \\ 10 & 10 & 10 & 1 & 13 \\ 11 & 11 & 11 & 11 & 1 \\ 12 & 12 & 12 & 12 & 1 \\ 13 & 13 & 13 & 13 & 1 \end{bmatrix}.$$

(iii):

Let $w = 2n-2$, $n > 3$.

First, suppose that for any two columns of matrix C, there exists no row in the matrix which incudes a special element chosen from both columns. Then we have the following inequality:

$$n(M(n,m,w)-m+1) \leq M(n,m,w)$$

or

$$M(n,m,w) \leq \lfloor \frac{n(m-1)}{n-1} \rfloor = \lfloor \frac{w+2}{w}(m-1) \rfloor. \tag{3.11}$$

Now suppose that in the matrix C there exist two columns such that they both simultaneously have a special element at least in one row. Then any other two columns among the remaining $n-2$ columns in the matrix have no special element on any row, otherwise the matrix C will not be $(2n-2)$-separating. Hence, we have

$$(n-2)(M(n,m,w)-m+1) \leq M(n,m,w)$$

or

$$M(n,m,w) \leq \lfloor \frac{n-2}{n-3}(m-1) \rfloor = \lfloor \frac{w-2}{w-4}(m-1) \rfloor. \tag{3.12}$$

Comparing (3.11) and (3.12) we have the first case of (iii).

Now consider the case for $n=3$, $w=4$.

First, suppose that there exist two columns in the matrix C for which there is no row in C on which both columns have a special element simultaneously. Then we have the inequality

$$2(M(3,m,4)-m+1) \leq M(3,m,4)$$

or

$$M(3,m,4) \leq 2(m-1). \tag{3.13}$$

Now let two columns in each of three pairs of columns simultaneously have a special element on any row of the matrix. In this case the following two conditions are necessary for C to be 4-separate:

1. Each column has at least two distinct special elements.
2. There are no more than two special elements on each row.

From condition 1.) it follows that the number of special elements in each column is no less than $M(3,m,4) - m + 2$. And from condition 2.) it follows that there are at most $2M(3,m,4)$ special elements in the matrix. Combining these facts gives

$$3(M(3,m,4) - m + 2) \leq 2M(3,m,4)$$

or

$$M(3,m,4) \leq 3m - 6 \tag{3.14}$$

From (3.13) and (3.14) the second case in (iii) follows. This completes the proof of the lemma. ■

We claim that, in the manner exactly analogous to the one given in Lemma 3.4.1, one may obtain similar bounds also for other values of w.

In a recent paper Blackburn proves similar results to the ones in Lemma 3.4.1 using the linear programming terminology (see *Propositions 2,3 and 4* in [49]). We present here these results.

Proposition 2 in [49] with *Theorem 2* in [49] covers the case (i) of Lemma 3.4.1.

Proposition 3 in [49] with *Theorem 2* in [49] gives the following bound:

$$M(n,m,2n-1) \leq \frac{w+1}{w-1}m.$$

This is weaker than the bound of case (ii) in Lemma 3.4.1.

Proposition 4 in [49] with *Theorem 2* in [49] gives the following bound:

$$M(n,m,2n-2) \leq \frac{w-1}{w-3}m. \tag{3.15}$$

This is stronger than case (iii) in Lemma 3.4.1 when $2m \geq w^2 - 5w + 6$, and is weaker otherwise.

Proposition 5 in [49] with *Theorem 2* in [49] gives the following bound:

$$M(n,m,w \geq 2n-3) \leq \begin{cases} 4m & \text{if } n = 4, \\ 9/5m & \text{if } n = 5, \\ \frac{w-3}{w-5}m & \text{if } n \geq 6. \end{cases} \tag{3.16}$$

Proposition 6 in [49] with *Theorem 2* in [49] gives the following bound:

$$M(6,m,8) \leq 2m. \tag{3.17}$$

In the general case (see *proof of Theorem 1* in [49]) it is shown that:

Lemma 3.4.3 *[49] For any $w \geq 2$ and positive integer m*

$$M(n, m, w) \leq nm$$

if $w > n$.

This bound is weaker than the bounds from Lemma 3.4.1, or bounds (3.15) and (3.16) when $w \geq 2n - 2$. It is stronger than bound (3.6) if $w - 1 > n$.

Lemma 3.4.4 *[45] For any integers n, m and w, $(w \leq m)$, the following bound holds:*

$$M(n, m, w) \leq \lfloor M(n-1, m, w)\frac{m}{w-1} \rfloor.$$

Proof: Let C be the w-separate (M, n, m)-matrix, where $M = M(n, m, w)$. Consider any column of C. Without loss of generality consider the 1st column, and let x_i, $1 \leq i \leq m$, be the number of symbol i in that column. Then

$$\sum_{i=1}^{m} x_i = M(n, m, w).$$

Choose $w - 1$ greatest numbers of $x_1, x_2, \ldots, x_m$. By rearranging the rows we can assume that they are first $w - 1$ numbers, $x_1, x_2, \ldots, x_{w-1}$. Let $\sum_{i=1}^{w-1} x_i = A$ for some A. Then we have

$$A \geq \frac{M(n, m, w)}{m} \cdot (w - 1), \tag{3.18}$$

since $M(n, m, w)/m$ is the average multiplicity of an element in that column.

Now consider a $A \times n$ matrix C' which is a submatrix of C, such that its rows are the rows of C, for which the symbol on the first position belongs to the set $\{1, 2, ..., w - 1\}$. C' is w-separate as C is w-separate. Let C'' be a $A \times (n - 1)$ matrix obtained from C' after deleting the first column. This matrix C'' is w-separate since there are no w pairwise distinct symbols in the first column of C'. Hence

$$A \leq M(n - 1, m, w). \tag{3.19}$$

From (3.18) and (3.19) we obtain

$$\frac{M(n, m, w)}{m} \cdot (w - 1) \leq M(n, m, w)$$

which proves the lemma.

■

For some small cases the following facts are known.

Lemma 3.4.5 *[37]*

$$M(3,4,4) = M(4,4,4) = 5.$$

Example 3.4.6 *[37] An* $(3,5,4,4) - PHF$

$$C = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 2 \\ 2 & 3 & 3 \\ 3 & 3 & 4 \\ 4 & 4 & 1 \end{bmatrix}.$$

Lemma 3.4.7 *[37]*

$$M(5,4,4) = 6$$

Example 3.4.8 *[37] An* $(5,6,4,4) - PHF$

$$C = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 1 & 1 & 2 \\ 1 & 3 & 2 & 2 & 3 \\ 2 & 4 & 2 & 3 & 3 \\ 3 & 3 & 3 & 4 & 4 \\ 4 & 4 & 4 & 4 & 4 \end{bmatrix}.$$

Here we prove the following:

Lemma 3.4.9 $M(6,4,4) = 8$.

Proof: Applying the result in Lemma 3.4.4 to Lemma 3.4.7 we get

$$M(6,4,4) \leq 8.$$

Thus, from Example 3.2.5 it follows that $M(6,4,4) = 8$. ■

This proves our claim that the family in Example 3.2.5 is optimal.

Lemma 3.4.10 *[37]*

$$M(3,5,4) \leq 9.$$

From Lemmas 3.4.1 and 3.4.4 we obtain a new bound on $M(n,m,w)$.

Theorem 3.4.11 *[45]*
The upper bound on $M(n,m,w)$ is given by:

$$M(n,m,w) \leq \underbrace{\lfloor\lfloor\lfloor A \times \frac{m}{w-1}\rfloor\frac{m}{w-1}\rfloor \times \cdots \times \frac{m}{w-1}\rfloor}_{n-n_1} \tag{3.20}$$

where

$$A = \begin{cases} \lfloor \frac{w+1}{w-1}(m-1)\rfloor & \text{for} \quad w = 2n_1-1 & \\ \lfloor \frac{w-2}{w-4}(m-1)\rfloor & \text{for} \quad w = 2n_1-2 & n_1 > 3 \\ 3m-6 & \text{for} \quad w = 4 & n_1 = 3 \end{cases} \tag{3.21}$$

This bound together with Example 3.2.4 shows that $M(4,3,3) = 9$ and proves the claim that the perfect hash family given in Example 3.2.4 is optimal.

We remark that bound (3.20) can be improved for other choices of n_1 and $A \geq M(n_1,m,w)$. For example if n_1 and $M(n_1,m,w)$ are chosen from Lemma 3.4.3 or from bound (3.16) then bound (3.20) will be stronger for some parameter sets. In the examples presented in the next section bound (3.20) is the stronger when A is given by (3.21).

3.5 Comparison of Bounds

First we compare bounds (3.1) and (3.2). To do this we prove the following lemma:

Lemma 3.5.1 *[45] For any fixed w, there exists a number $m_0(w)$ such that for all $m > m_0(w)$, the minimum of the right side in the Körner-Marton bound (3.2)*

$$\frac{1}{n}\log M(n,m,w) \preceq \min_{0\leq j\leq w-2} \frac{m^{\underline{j+1}}}{m^{j+1}} \log \frac{m-j}{w-j-1} \tag{3.22}$$

achieves for $j = 0$.

Proof: Denote by

$$A_j(m,w) = \frac{m^{\underline{j+1}}}{m^{j+1}} \log \frac{m-j}{w-j-1}.$$

We compute $A_{j+1}(m,w) - A_j(m,w)$, $0 \leq j \leq w-3$. $A_{j+1}(m,w) - A_j(m,w) = \frac{m^{\underline{j+2}}}{m^{j+2}} \log \frac{m-j-1}{w-j-2} - \frac{m^{\underline{j+1}}}{m^{j+1}} \log \frac{m-j}{w-j-1} =$

$= \frac{m^{j+1}}{m^{j+2}}[(m-j-1)\log\frac{m-j-1}{w-j-2} - m\log\frac{m-j}{w-j-1}] =$
$= \frac{m^{j+1}}{m^{j+2}}[\log[\frac{(m-j-1)(w-j-1)}{(m-j)(w-j-2)}]^m - \log(\frac{m-j-1}{w-j-2})^{j+1}].$

Since $w \geq k$ and $\frac{(m-j-1)(w-j-1)}{(m-j)(w-j-2)} > 1$, the value $[\frac{(m-j-1)(w-j-1)}{(m-j)(w-j-2)}]^m$ increases faster than $(\frac{m-j-1}{w-j-2})^{j+1}$ when m increases. Further, it is clear that there exists a number $m_0(w,j)$ such that for $m > m_0(w,j)$ we have $A_{j+1}(m,w) - A_j(m,w) > 0$.

Take

$$m_0(w) = \min m_0(w,j).$$

Then for all $m > m_0(w)$ the value $A_j(m,w)$ increases monotonely with j, which proves the lemma. ■

Lemma 3.5.1 in particular shows that except for a finite values of m the Körner-Marton bound (3.2) is better than the Fredman-Komlòs bound (3.1). We remark that Lemma 3.5.1 compares the Körner-Marton bound and the Fredman-Komlòs bound in the general case. Some attempts to compare these bounds for a few particular cases have been done in [39]. The results we present here in view of Lemmas 3.5.3 and 3.5.4.

Lemma 3.5.2 *[39] For $m \geq 4$ and $w = m$, the Körner-Marton bound (3.2) is the strongest when $j = m-2$ for $n \geq n_0$.*

Note that for $j = w-2$ bound (3.2) is originally given by Fredman-Komlòs (bound (3.1)).

Lemma 3.5.3 *[39] For $m \geq 3$ and $w = 3$ the Körner-Marton bound (3.2) is the strongest when $j = 0$ for $n \geq n_0$.*

Lemma 3.5.4 *[39] For $m \geq 5$ and $w = 4$ the Körner-Marton bound (3.2) is the strongest when $j = 0$ for $n \geq n_0$.*

Comparing bound (3.4) with bound (3.20) we come to the following conclusion:

For a fixed w there exists a $m_0(w)$ such that $m > m_0(w)$ the Körner-Marton bound is stronger than the Fredman-Komlòs bound. In the case when the Körner-Marton bound is stronger than the Fredman-Komlòs bound, then our bound (3.20) is better than both these bounds. Bounds (3.1) and (3.2) and (3.20) are better than bound (3.5) when w is close to m. If $n \to \infty$ with fixed m and w, the Blackburn-Wild bound is stronger for many values of w and m.

Our investigation shows that for some values of w and m our bound is stronger than other known bounds. We illustrate this by an example.

Example 3.5.5 *Let $m = 9$, $w = 5$, for $n > 3$*

bound	3.3	*case*	$j = 3$	$M(n, 9, 5) \preceq (2.2837)^n$
bound	3.3	*case*	$j = 2$	$M(n, 9, 5) \preceq 2(2, 3776)^n$
bound	3.3	*case*	$j = 1$	$M(n, 9, 5) \preceq 3(2.3913)^n$
bound	3.3	*case*	$j = 0$	$M(n, 9, 5) \leq 4.(2.25)^n$
bound	3.5			$M(n, 9, 5) \leq 4(9^{\lceil \frac{n}{4} \rceil} - 1)$
bound	3.20			$M(n, 9, 5) \leq \underbrace{\lfloor \lfloor \lfloor 12 \cdot 2.25 \rfloor 2.25 \rfloor \times \cdots \times 2.25 \rfloor}_{n-3}$

Taking $n = 9$ we get

bound	3.3	*case*	$j = 3$	$M(9, 9, 5) \preceq 1689$
bound	3.3	*case*	$j = 2$	$M(9, 9, 5) \preceq 4856$
bound	3.3	*case*	$j = 1$	$M(9, 9, 5) \preceq 7671$
bound	3.3	*case*	$j = 0$	$M(9, 9, 5) \leq 5911$
bound	3.5			$M(n, 9, 5) \leq 2912$
bound	3.20			$M(9, 9, 5) \leq 1532$

Taking $n = 5$ and calculating the possible exact value (in oppose to asymptotic value) for bound (3.3) in case $j = 3$ we get

bound	3.3	*case*	$j = 3$	$M(5, 9, 5) \leq 87$
bound	3.3	*case*	$j = 2$	$M(5, 9, 5) \preceq 151$
bound	3.3	*case*	$j = 1$	$M(5, 9, 5) \preceq 234$
bound	3.3	*case*	$j = 0$	$M(5, 9, 5) \leq 230$
bound	3.5			$M(5, 9, 5) \leq 320$
bound	3.20			$M(5, 9, 5) \leq 60$

3.6 Existence Results

The following theorem is obtained by a simple probabilistic argument.

Theorem 3.6.1 *[30] A (n, M, m, w)-perfect hash family exists where*

$$n \geq \frac{\log \binom{M}{m}}{\log m^w - \log\left(m^w - w!\binom{m}{w}\right)}. \tag{3.23}$$

A weaker result is derived in [31] by performing some simple approximations.

Theorem 3.6.2 *[30] There exists an (n, M, m, w)-perfect hash family with*

$$n \geq we^{\frac{w^2}{m}} \log M. \tag{3.24}$$

The result of Theorem 3.6.1 has been improved in [47] for some parameter sets.

Theorem 3.6.3 *[47] There exits an (n, M, m, w)-perfect hash family with*

$$n \geq \frac{\log 4(\binom{M}{m} - \binom{M-w}{w})}{\log m^w - \log\left(m^w - w!\binom{m}{w}\right)}. \tag{3.25}$$

It is stated in [47] that bound (3.25) is better that the classical bound (3.23) whenever $\frac{3}{4}\binom{M}{w} < \binom{M-w}{w}$; this inequality holds when w is small compared with M. It is also shown in [47] that bound (3.25) is tight for sufficiently large m. The upper and lower bounds on $M(n, m, w)$ are given in the following form in [35].

Theorem 3.6.4 *[32, 35]*

$$\frac{1}{w-1}\log\frac{1}{1-\frac{m^w}{m^w}} \preceq \frac{1}{n}\log M(n,m,w) \preceq \frac{m^{\overline{w-1}}}{m^{w-1}}\log(m-w+2)$$

Thus, the theorem states that

$$n = \Theta(\log M).$$

However, the existence results discussed above are non-constructive.

Linear perfect hash families are considered in [43]. Let C be the matrix representing a perfect hash family. A perfect hash family is called linear if C corresponds to a linear code. In [43], a probabilistic argumentation shows that there exists a linear $(n, q^i, q, w) - PHF$, where $n = i(w-1)$ for sufficiently large prime power q. It is also shown that no linear $(n, q^i, q, w) - PHF$ exists if $n < i(w-1)$. Thus the authors call the linear perfect hash family *optimal* if $n = i(w-1)$.

Theorem 3.6.5 *[43] Let i and w be integer such that $i \geq 2$ and $w \geq 2$ and let q be a prime power, $M = q^i$. If a linear $(n, M, q, w) - PHF$ exits , then $n \geq i(w-1)$. Furthermore, if $q \geq (\frac{1}{2}w(w-1))^{i(w-1)}$ then a linear $(n, M, q, w) - PHF$ exists with $n = i(w-1)$.*

The proof of the theorem is probabilistic, and produces no explicit classes of perfect hash families.

Hereafter, we consider explicit constructions for perfect hash families in oppose to existence results.

3.7 Direct Constructions

In [43] it is shown that the techniques of Theorem 3.6.5 suffice to construct explicit classes of linear perfect hash families in certain cases. It is shown for instance, that there exist explicit constructions for $(n, M, q, w) - PHF$ where $n = (w-1)\frac{\log M}{\log q}$. These constructions require the prime power q to be very large compared to w and n.

A direct connection between error-correcting codes and perfect hash families, due to Alon is as follows:

Theorem 3.7.1 [34] *Suppose there is an (n, M, m) code $\mathcal{C}$ with minimum Hamming distance d. Then there is an $(n, M, m, w) - PHF$, where*

$$(n-d)\binom{w}{2} < n.$$

Proof: Let C be the matrix representing $\mathcal{C}$. Then C is an $M \times n$ matrix, whose entries are from a set of m symbols. The condition $(n-d)\binom{w}{2} < n$ asserts that for any given w rows, say $i_1, \ldots, i_w$, of C there is at least one column whose w entries in the rows $i_1, \ldots, i_w$ are pairwise distinct. Thus C is an $(n, M, m, w) - PHF$, as desired. ■

This theorem together with some known results on error-correcting codes (see for example [13]) leads to several explicit constructions for perfect hash families. The theorem shows that an error correcting code with large minimum Hamming distance provides a perfect hash family.

We present some construction examples derived as corollaries of Theorem 3.7.1:

Corollary 3.7.2 *Let $\mathcal{C}$ be a q-ary MDS code $[n, k, d]$, $k + d = n + 1$. If $k = \lceil \frac{n}{\binom{w}{2}} \rceil$ for an integer w, then C is an $(n, q^k, q, w) - PHF$.*

From Reed-Solomon codes we have:

Corollary 3.7.3 *Suppose n and q are given, with q a prime power and $n \leq q+1$. Then there exists an $(n, q^{\lceil \frac{n}{\binom{w}{2}} \rceil}, q, w) - PHF$.*

This construction is explicit and gives perfect hash families with

$$n = O(\log M).$$

However, this class of perfect hash families is quite restricted with condition $n \leq q+1$. Also, we have $M > q$ for this class only if $n > \binom{w}{2}$, i.e., only if $q \geq \binom{w}{2}$.

In the case of the extended Reed-Solomon code over $GF(p^j)$ we may rewrite the previous corollary as follows:

Corollary 3.7.4 *Let q be a prime power with $q \geq \binom{w}{2}$, and $j \geq 2$ be an integer. Then there exists a $(q^j, q^{j q^{j-1}}, q^j, w) - PHF$.*

Several other constructions derived as corollaries of Theorem 3.7.1 are given in [38] and in [46].

Lemma 3.7.5 *[46] Suppose there are t MOLS of order m. Then there exists a $(t+2, m^2, m, w) - PHF$ provided $t > \binom{w}{2} - 1$.*

Lemma 3.7.6 *[38] Suppose there is an $(M_0, \binom{w}{2}+1; 1)$ -difference matrix and an $(n_0; M_0, m, w) - PHF$. Then there exists a $((\binom{w}{2}+1)n_0, {M_0}^2, m, w) - PHF$.*

Let $\mathcal{C}$ be a linear AG code defined on the Garcia-Stichtenoth (G-S) curves with parameters given in Theorem (2.6.1).

Applying Theorem 3.7.1 to $\mathcal{C}$ we obtain the following result:

Theorem 3.7.7 *For every prime power q and any integer $l \geq 1$, there exists an $(n; M, q^2, w) - PHF$, where*

$n = q^{l+1}(q-1)$,
$M = q^{2\lfloor uq^{l+1} \rfloor}$,
u is a real number with $1 \leq u \leq q-2$, and
$w = \lfloor \frac{1}{2}(1 + \sqrt{1 + \frac{8}{u+1}(q-1)}) \rfloor$.

This construction gives infinite classes of perfect hash families where $n = O(\log M)$. However, this class is restricted since q has to be a prime power, $q > \binom{w}{2}$, and also the construction algorithm complexity is polynomial in n.

All constructions as direct application of Theorem 3.7.1 are restrictive, in general, in the sense that they can produce perfect hash families with large M only if $q > \binom{w}{2}$. In fact, the previous construction examples provide an evidence of this fact, which follows from the Plotkin bound. The Plotkin bound given in Corollary 2.2.3 together with Theorem 3.7.1 implies:

$$n(1 - \frac{1}{\binom{w}{2}}) < d \leq n\left(1 - \frac{1}{q}\right)\frac{M}{M-1}.$$

Thus

$$1 - \frac{1}{\binom{w}{2}} < \left(1 - \frac{1}{q}\right)\frac{M}{M-1}.$$

So when $M \rightarrow \infty$ for fixed q and w, we have

$$\binom{w}{2} < q.$$

For small M, however, we can construct perfect hash families with $q \leq \binom{w}{2}$ when $\binom{w}{2} < \frac{n}{n-d}$. Perfect hash families in Corollary 3.7.3 can have $q = \binom{w}{2}$. Note however that $q \geq n - 1$.

We present a class of perfect hash families, derived from codes constructed in Section 2.7 by applying Theorem 3.7.1. This class provides examples of perfect hash families with $q < \binom{w}{2}$ and $M > q$.

Theorem 3.7.8 *Let q_0 and q_1 be prime powers such that $q_1 \geq q_0$, and $i \geq 1$ is an integer. Then for any integer n with $n \leq q_0 q_1^i + q_1^i + q_1^{i-1} + \cdots + q_1 + 1$ there exists an $(n, M, q, w) - PHF$ with*

$$\begin{aligned} M &= q_0^2 q_1^i \\ q &= q_0 q_1^i \\ w &= \lceil \frac{\sqrt{8n+1} - 1}{2} \rceil. \end{aligned}$$

where $q_1 \geq q_0$ are prime powers and $m \geq 1$ is an integer.

Proof: First, recall that the parameters $(N, M, q; d)$ of a code $\mathcal{C}^*$ in Theorem 2.7.8 (ii) are $N = q_0 q_1^i + q_1^i + q_1^{i-1} + \cdots + q_1 + 1$, $M = q_0^2 q_1^i$, $q = q_0 q_1^i$, and $d = N - 1$, where $m \geq 1$ is an integer. We remark that if $\mathcal{C}^*$ is shortened, the resulting code with length $n \leq N$ always has minimum distance $d = n - 1$.

Let $(n, M, q; n-1)$ be the parameters of a shortened code $\mathcal{C}$ of $\mathcal{C}^*$ (the case $\mathcal{C} = \mathcal{C}^*$ is also included). So, $n \leq N$. Let $w = \lceil \frac{\sqrt{8n+1}-1}{2} \rceil$. By Theorem 3.7.1, $\mathcal{C}$ is a w-perfect hash family. The proof is complete. ■

Example 3.7.9 *From Theorem 3.7.8 we obtain:*

$$\begin{array}{llll} q_0 = 3, & q_1 = 4, & i = 1 & (17, 36, 12, 6) - PHF, \quad \binom{w}{2} = 15 > q = 12. \\ q_0 = 4, & q_1 = 5, & i = 1 & (26, 80, 20, 7) - PHF, \quad \binom{w}{2} = 21 > q = 20. \\ q_0 = 3, & q_1 = 5, & i = 3 & (531, 1125, 375, 33) - PHF, \quad \binom{w}{2} = 528 > q = 375. \end{array}$$

Note that the condition in Theorem 3.7.1 is sufficient for a code to be a perfect hash family but it is not necessary.

For example, when $n = 6$ and $w = 4$ Theorem 3.7.1 provides no perfect hash families with $M > 4$. However, a $(6, 8, 4, 4)$-perfect hash family exists (see Example 3.2.5).

More generally, the explicit constructions as described in the rest of this section provide classes of perfect hash families, for which the condition of Theorem 3.7.1 does not hold.

Theorem 3.7.10 *[37] For every integer $f > 3$ there exists an explicitly constructed $(5, 2^f, 2^{f-1}, 4) - PHF$.*

Theorem 3.7.11 *[47]*

For every prime number p, such that $p = 11$ or $p \geq 17$ there exists an explicitly constructed $(6, p^2, p, 4) - PHF$.

Theorem 3.7.12 *[49]*

For every integer $a \geq 2$ there exists an explicitly constructed (w, a^w, a^{w-1}, w) $-PHF$.

This improves the result in Theorem 3.7.10 when $w = 4$. Taking $a = 2^i$ from Theorem 3.7.12 we have a $(4, 2^{4i}, 2^{3i}, 4) - PHF$, while taking $f - 1 = 3i$ in Theorem 3.7.10 we get $(5, 2^{3i+1}, 2^{3i}, 4) - PHF$.

An other interesting result is given in [49] as follows:

Theorem 3.7.13 *[49]*

$M(n, m, w) = O(m)$ *if and only if* $w > n$.

In the case when $n > w$ and $m \to \infty$ the problem of computing the exact constant c in $M(n, m, w) = cm$ by providing an explicit construction and a tight bound is reduced to linear programming problems [49]. Some examples given in [49] achieve the bounds in Lemma 3.4.1, or bounds (3.15), (3.16) and (3.17), when $m \to \infty$.

An optimal class of perfect hash families for $w = 2n - 1$ and $(n-1)|(m-1)$ can be found in the proof of Lemma 3.4.1.

3.8 Recursive Constructions

Some recursive constructions given by several authors produce perfect hash families with small fixed m and w and $n \to \infty$.

Two recursive constructions are given in [38]. Using an easily constructed specific family of difference matrices and Lemma 3.7.6 the following result is obtained iteratively.

Theorem 3.8.1 *[38] Suppose there exists an* $(n_0, M_0, m, w) - PHF$ *and suppose that* $\gcd(n_0, \binom{w}{2}!) = 1$. *Then there exists a* $((\binom{w}{2} + 1)^j n_0, {M_0}^{2^j}, m, w) - PHF$ *for any integer* $j \geq 1$.

Theorem 3.8.1 states that for fixed m and w we can construct an infinite class of (n, M, m, w) perfect hash families, where n is $O((\log M)^{log(\binom{w}{2}+1)})$.

As immediate result of theorem we have:

Corollary 3.8.2 *[38] There exists a* $(3 \times 4^j, 5^{2^j}, 3, 3) - PHF$ *for any integer* $j \geq 1$.

With the above parameters, we have

$$n \approx 0.556(\log M)^2.$$

The second recursive construction in [38] is given in the following theorem:

Theorem 3.8.3 *[38] Suppose the following exist:*

- *an* $(n_1, M_0 M_1, m, w) - PHF$,
- *an* $(n_2, M_2, M_1, w-1) - PHF$,
- *an* $(n_3, M_2, m, w) - PHF$.

Then there exists an $(n_1 n_2 + n_3, M_0 M_2, m, w) - PHF$.

The following corollary is an immediate consequence of Theorem 3.8.3.

Corollary 3.8.4 *[38] For any integer $j \geq 2$, there exists a $(2j^2 - 2j, 3^j, 3, 3) - PHF$.*

These parameters give

$$n \approx 0.796(\log M)^2.$$

Further general results are given in the following theorems:

Theorem 3.8.5 *[38] Suppose there exists an $(a_3, m^2, m, 3) - PHF$. Then, for any $j \geq 2$, there exists an $(a_3\binom{j}{2}, m^j, m, 3) - PHF$.*

By using Theorem 3.8.5 for $w = 4$ we have:

Theorem 3.8.6 *[38] Suppose there exists an $(a_3, m^2, m, 3) - PHF$ and an $(a_4, m^2, m, 4) - PHF$. Then, for any $j \geq 2$, there exists an $(a_4(a_3\binom{j}{3} + 1), m^j, m, 4) - PHF$.*

These constructions have been generalized and improved in [37], [45], [46] and [48].

The next lemma shows a simple product type construction of perfect hash families, also called composition or concatenation.

Lemma 3.8.7 *Suppose there exist an $(n_0, M_0, m_0, w) - PHF$ and an $(n_1, M_1, m_1, w) - PHF$, where $m_1 \leq M_0$. Then there exists an $(n_0 n_1, M_1, m_0, w) - PHF$.*

From Lemma 3.8.7 and from a particular case of Corollary 3.7.3 when $n = \binom{w}{2} + 1$ we obtain the following class of perfect hash families:

Lemma 3.8.8 *[37] For any prime power q and any integer w with $\binom{w}{2} < q$ there exists a $((\binom{w}{2} + 1)^j, m^{2^j}, q, w) - PHF$ where $j \geq 1$ is any integer.*

In case $q = w = 3$ this lemma gives a $(4^j, 3^{2^j}, 3, 3) - PHF$ for any integer $j \geq 1$. These perfect hash families have

$$n \approx 0.398(\log M)^2.$$

From Lemmas 3.8.7 and 3.7.5 it follows:

Lemma 3.8.9 *[46] Suppose there exist $\binom{w}{2} - 1$ MOLS of order M_0 and an $(n_0, M_0, m, w) - PHF$. Then there exists a $((\binom{w}{2} + 1)n_0, (M_0)^2, m, w) - PHF$.*

3.8.1 The First New Infinite Class

By combining Lemma 3.8.7 and Theorem 3.7.7 we can prove the following result:

Theorem 3.8.10 *For every given integers w, m, where $m \geq w \geq 2$, and for any integer $l \geq 1$, there exists an (n, M, m, w)-perfect hash family with*

$n = n_0.(q-1)q^{l+1}$,
$M = q^{2\lfloor uq^{l+1} \rfloor}$,
where n_0 is a constant, q is a prime power such that $q \geq \frac{w(w-1)(u+1)}{2} + 1$, and u is a real number with $1 \leq u \leq q-2$.

Moreover, we have $n = O(\log M)$.

Proof: Let w, m, $m \geq w \geq 2$, be given integers. Let q be the smallest prime power such that $w = \lfloor \frac{1}{2}(1 + \sqrt{1 + \frac{8}{u+1}(q-1)}) \rfloor$, with $1 \leq u \leq q-2$, as shown in Theorem 3.7.7. A simple observation shows that we can always construct an $(n_0, q^2, m, w) - PHF$ explicitly for a certain value n_0. Applying Lemma 3.8.7 and Theorem 3.7.7 yields the perfect hash families with parameters as claimed. ■

We remark that the idea of using Garcia-Stichtenoth (G-S) curves and the simple product construction to derive infinite class of perfect hash families with $n = O(\log M)$, for fixed m and w is first given in [48] with a slightly different interpretation.

It should be noticed that the first low-complexity algorithm for constructing "one-point" AG codes on G-S curves has a runtime upper-bounded by $(n \log_q n)^3$, where n the length of the code and the complexity is measured in terms of multiplications and divisions over the finite field $\mathbb{F}_{q^2}$ [28]. The complexity of constructing w-perfect hash families in Theorem 3.8.10 is, therefore, polynomial in n.

3.8.2 The Second New Infinite Class

Next we will describe an explicit new construction of an infinite class of perfect hash families with the best known asymptotic behavior among similar classes. This result has been presented in the papers [42], [45]. We should remark that this result is independently obtained also in [46]. Here we will present also the interpretations and notation in [46] since it is useful for our discussions in the next two chapters.

From Lemma 3.8.7 together with Corollary 3.7.4 we obtain an infinite class of perfect hash families for any integers m and w with very good asymptotic behavior.

The construction algorithm can be described step by step as follows:

Construction algorithm:

Step 0:

Using some method to construct $(n_0, q^j, m, w) - PHF$ for any $i \geq 2$ and prime $q \geq \binom{w}{2}$.

Step 1:

We have:	$(n_0, M_0 = q^j, m, w) - PHF.$
Corollary 3.7.4 provides:	$(n_1{}' = q^j, q^{jq^{j-1}}, q^j, w) - PHF.$
Applying Lemma 3.8.7 gives:	$(n_1 = n_0 n_1{}', M_1 = q^{jq^{j-1}}, m, w) - PHF.$

Thus, in Step 1 we obtain a perfect hash family with

$$n_1 = \frac{n_0}{j \log q} \cdot q \log M_1.$$

$$\vdots$$

Step i:

From Step $i-1$ we have:	$(n_{i-1}, M_{i-1}, m, w) - PHF.$
Corollary 3.7.4 provides:	$(n_i{}' = M_{i-1}, M_{i-1}^{\frac{M_{i-1}}{q}}, M_{i-1}, w) - PHF.$
Applying Lemma 3.8.7 gives:	$(n_i = n_{i-1} n_i{}', M_i = M_{i-1}^{\frac{M_{i-1}}{q}}, m, w) - PHF.$

Thus, in Step i we obtain a perfect hash family with

$$n_i = \frac{n_0}{j \log q} q^i \log M_i.$$

In order to show that the last equation has the form

$$n = \Theta(\varphi(\log M))$$

we perform the following approximations:

From the recurrent relation $M_i = M_{i-1}^{\frac{M_{i-1}}{q}}$ and $M_0 = q^j$, we obtain

$$M_i = q^{\alpha_{i-1} q^{\alpha_{i-2} \cdots \alpha_1 q^{\alpha_0 q^{j-1}}}} \tag{3.26}$$

where $\alpha_0 = j$, and $\alpha_l = \frac{M_{l-1}}{q}$ for $1 \leq l \leq i-1$.
Taking $\alpha_l = 1$ for all $1 \leq l \leq i-1$ in (3.26), we obtain:

$$M_i > q^{q^{\cdots^{q^{q^{j-1}}}}}. \tag{3.27}$$

If $1 \leq i \leq j-1$, i.e., $q^i < jq^{j-1}$, then (3.27) gives

$$n_i = \Theta\left(\underbrace{\log_q \log_q \cdots \log_q}_{i-1} M_i \times \log M_i\right)$$

and if $j-1 < i \leq jq^{j-1}$, then we get

$$n_i = \Theta\left(\underbrace{\log_q \log_q \cdots \log_q}_{i-2} M_i \times \log M_i\right).$$

In general, we have

$$n = \Theta\left(\underbrace{\log_q \log_q \cdots \log_q}_{r} M \times \log M\right)$$

where $r \to \infty$ when $n \to \infty$.

Example 3.8.11 *Case* $w = 3$ *and* $q = 3$.

In this case the algorithm described above suggests taking the Reed-Solomon code with the parameters $(n = q - 1, k, d)$, *where* $q = 3^j$, $k = 3^{j-1}$ *and* $d = 2 \cdot 3^{j-1} > \frac{2}{3}(3^{j-1} - 1)$ *as an* MDS *code in corollary 3.7.3. Thus we have:*

Step 0:

Take: $(4, 9, 3, 3) - PHF$.

Step 1:

We have:	$(4, 3^2, 3, 3) - PHF$.
Corollary 3.7.3 provides:	$(8, 3^6, 9, 3) - PHF$.
Applying Lemma 3.8.7 gives:	$(32, 3^6, 3, 3) - PHF$.

Step 2:

From the Step 1 we have:	$(32, 3^6, 3, 3) - PHF$.
Corollary 3.7.3 provides:	$(3^6 - 1, 3^{6 \cdot 3^5}, 3^6, 3) - PHF$.
Applying Lemma 3.8.7 gives:	$(23296, 3^{1458}, 3, 3) - PHF$.

A similar construction is given with the following notation in [46].

Theorem 3.8.12 *Suppose there exists an* $(n_0, q^{s_0}, m, w) - PHF$ *where* q *is a prime power and* $q^{s_0} \geq \frac{w(w-1)}{2} + 1$. *Then there exists an* $(n_0 R_i; q^{s_i}, m, w) - PHF$ *for all* $i \geq 0$, *where* $R_0 = 1$, *and*

$$\begin{aligned} R_i &= q^{s_{i-1}} R_{i-1}, \\ s_i &= s_{i-1} \lceil \frac{q^{s_{i-1}}}{\binom{t}{2}} \rceil \end{aligned}$$

for all $i \geq 1$.

Proof: We proceed by induction on i. For $i = 0$, the assertion is correct. Now assume $i \geq 1$. Corollary 3.7.3 gives an $(q^{s_{i-1}}; q^{s_i}, q^{s_{i-1}}, w) - PHF$ when $n = q$ and q is replaced by $q^{s_{i-1}}$.

By induction, there exists an $(n_0 R_{i-1}, q^{s_{i-1}}, m, w) - PHF$. Now applying Lemma 3.8.7 yields an $(n_0 R_i, q^{s_i}, m, w) - PHF$. The proof is complete. ■

The asymptotic behavior of the parameters of the perfect hash families produced by Theorem 3.8.12 is calculated as follows in [46], pp.196-197.

The function $\log^* : \mathbb{Z}^+ \longrightarrow \mathbb{Z}^+$ is defined recursively by

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

Note that the function $\log^*$ grows very slowly, e.g., $\log^*(a) \leq 7$ for $a \leq 2^{2^{65536}}$.

Define $n_i = n_0 R_i$ and $M_i = q^{s_i}$ for all $i \geq 0$.

First, we have that

$$s_i \geq \frac{s_{i-1} q^{s_{i-1}}}{w^2} = \frac{s_{i-1} R_i}{w^2 R_{i-1}}.$$

Iterating this inequality, we see that

$$s_i \geq \frac{s_0 R_i}{w^{2i}}.$$

Since

$$s_i = \frac{\log M_i}{\log q}$$

and

$$R_i \frac{n_i}{n_0},$$

we have that

$$\frac{\log M_i}{\log q} \geq \frac{s_0 n_i}{n_0 w^{2i}}.$$

From this, we get the following inequality:

$$n_i \leq \frac{n_0}{s_0 \log q} w^{2i} (\log M_i). \tag{3.28}$$

For i sufficiently large, say $i \geq i_0$, we have

$$q^{s_{i-1}/w^2} > 2.$$

Now, for $i > i_0$, we have

$$M_i = q^{s_i} \geq q^{s_{i-1} q^{s_{i-1}}/w^2} > 2^{q^{s_{i-1}}} = 2^{M_{i-1}}.$$

Hence, it follows that

$$\log^* (M_i) > i - i_0 \tag{3.29}$$

for all $i > i_0$. Substituting (3.28) into (3.29), we get

$$n_i \leq \frac{n_0 w^{2i_0}}{s_0 \log q} (w^2)^{\log^*(M_i)} (\log M_i) \tag{3.30}$$

for all $i > i_0$.

For any given values of q^{s_0}, m and w we can always construct an $(n_0, q^{s_0}, m, w - PHF$ for some n_0 (as a trivial and naive solution we may take $n_0 = \binom{q^{s_0}}{w}$, so that each subset of size w is covered by one function). Therefore, from (3.30) we have the following theorem:

Theorem 3.8.13 *For any positive integers m and w with $w \leq m$, there is an infinite family of $(n, M, m, w) - PHF$ such that n is $O((w^2)^{\log^*(M)}(\log M)$.*

3.8.3 The Third New Infinite Class

Now we give our next new explicit construction of an infinite class of perfect hash families by means of a double recursive method. The main result is given in Theorem 3.8.15.

The construction appears to be rather complex, even though we have attempted to give a clear concise explanation. The result in the general form is presented in the paper [82]. The case $q = w = 3$ is first proved in [45].

We first prove Lemma 3.8.14 below, which is essential for our purpose.

From now on let q be a prime power. We begin with a description of a collection of matrices derived from mutually orthogonal Latin squares (MOLS) whose symbols are elements in the finite field $F_q = \{0, 1, \ldots, q-1\}$.

Let $M_1, \ldots, M_{q-1}$ be a set of $q-1$ MOLS, of which the first column is the vector $(0, 1, \ldots, q-1)^T$. Let M_0 be the $q \times q$ matrix whose all q columns are equal to the vector $(0, 1, \ldots, q-1)^T$ (i.e. each row of M_0 consists of a q time repeating of a symbol). The collection of $M_0, \ldots, M_{q-1}$ is equivalent to an orthogonal array $\mathsf{OA}_1(2, q, q)$ (see, for example [17, p. 130]) and hence to a Reed-Solomon code $\mathcal{RS}$ with parameters $(q, q^2, q, d = q-1)$.

For $2 \leq m \leq q$, set

$$\mathcal{A} = \{A_{0,m}, \ldots, A_{q-1,m}\},$$

where each matrix $A_{h,m}$ is obtained from M_h by deleting its $q-m$ rightmost columns.

Consider the $q^2 \times (m+1)$ array $\mathcal{A}^E$ obtained from $\mathcal{A}$ by extending each matrix $A_{i,m}$ with the $(m+1)^{th}$ column $(i, i, \ldots, i)^T$. Then $\mathcal{A}^E$ is equivalent to the Reed-Solomon code $(m+1, q^2, q, d = m) - \mathcal{RS}$. By Theorem 3.7.1 $\mathcal{A}^E$ is an $(m+1, q^2, q, w) - PHF$ with $\binom{w}{2} < m+1$.

Conversely, if w is given, we set $m = \binom{w}{2}$. Then the collection $\mathcal{A}$ has the following crucial property: every subset $\mathcal{B}$ of w' distinct matrices $A_{i_1,m}, \ldots, A_{i_{w'},m}$ of $\mathcal{A}$, where $1 \leq w' \leq w-1$, forms an $(m, qw', q, w) - PHF$.

This can be easily seen as follows:

Consider $\mathcal{B}$ as part of $\mathcal{A}^E$. Note that $\mathcal{A}^E$ has exactly one column more than $\mathcal{B}$, the $(m+1)^{th}$ column. For any given set W of w rows of $\mathcal{B}$, there is a column $\mathbf{c}$ in $\mathcal{A}^E$, such that the symbols of $\mathbf{c}$ at the given w rows are pairwise distinct, because $\mathcal{A}^E$ is an $(m+1, q^2, q, w) - PHF$. Further, since $\mathcal{B}$ is a collection of w' matrices $A_{h,m}$, there are at least two rows of W belonging to the same matrix in $\mathcal{B}$. This implies that the column $\mathbf{c}$ is not the $(m+1)^{th}$ column of $\mathcal{A}^E$, hence $\mathbf{c}$ must be a column of $\mathcal{B}$, as desired.

Thus, we have proved the following result:

Lemma 3.8.14 *[82] Let $\mathcal{A}$ be the collection of q matrices $\{A_{0,m}, \ldots, A_{q-1,m}\}$ just described above, where each $A_{h,m}$ is a $q \times m$ matrix, whose entries are elements of F_q. Let $m = \binom{w}{2}$. Then, any subset $\mathcal{B}$ of w' distinct matrices $A_{i_1,m}, \ldots, A_{i_{w'},m}$ of $\mathcal{A}$, where $1 \leq w' \leq w-1$, forms an $(m, q.w', q, w) - PHF$.*

We are now ready to prove the following theorem:

Theorem 3.8.15 *[82] Let $w \geq 2$ be any integer and q be any prime power such that $q \geq \binom{w}{2}$. Then there exists an $(O((i+1)^{w-1}), q^{i+1}, q, w) - PHF$ for any integer $i \geq 1$.*

Proof: The proof is by induction on w and i.

In the following we use $n_i(w)$ as an abbreviation for $O((i+1)^{w-1})$ and C_i^w for $(n_i(w), q^{i+1}, q, w) - PHF$.

Note that the vector space F_q^{i+1} is an $(n_i(2), q^{i+1}, q, 2) - PHF$, where $n_i(2) = i+1$. Thus C_i^2 exists for all $i \geq 1$. In other words the statement is valid for $w = 2$.

Assume that the statement is valid for $w - 1 > 2$. That means that for every $2 \leq u \leq w-1$ there exists an $C_i^u = (n_i(u), q^{i+1}, q, u) - PHF$ for all i. We prove that the statement is true for w, i.e., there is an $C_i^w = (n_i(w), q^{i+1}, q, w) - PHF$ for every i.

This is done by induction on i.

For $i = 1$ there is a $C_1^w = (n_1(w), q^2, q, w) - PHF$, where $n_1(w) = \binom{w}{2} + 1$ and $q \geq n_1(w) - 1$. In fact, C_1^w is obtained from the Reed-Solomon code $(n_1(w), q^2, q) - \mathcal{RS}$ by using Theorem 3.7.1. Assume that C_j^w exists for all $j \leq i-1$.

Let

$$\tilde{C_i^w} = (D_{i-1}^w, E_{i-1}^{w-1})$$

denote the concatenation of D_{i-1}^w and E_{i-1}^{w-1}, which are defined as follows:

D_{i-1}^w is obtained from C_{i-1}^w by repeating each of its rows q times.

E_{i-1}^{w-1} is obtained from C_{i-1}^{w-1} by replacing each symbol j by matrix $A_{j,w}$, described in Lemma 3.8.14.

We depict $\tilde{C_i^w}$ as an $q.q^i \times (n_{i-1}(w)+n_{i-1}(w-1).\binom{w}{2})$ array, where the first $n_{i-1}(w)$ columns correspond to D_{i-1}^w and the remaining $n_{i-1}(w-1).\binom{w}{2}$ columns correspond to E_{i-1}^{w-1}. And we partition the rows of the array $\tilde{C_i^w}$ into q^i consecutive blocks, say $B_1, \ldots, B_{q^i}$, each block B_i has q rows.

	D_{i-1}^w	E_{i-1}^{w-1}			
B_1	1st row of C_{i-1}^w repeated q times	$A_{(1,1),w}$	$A_{(1,2),w}$	$\ldots$	$A_{(1,n_{i-1}(w)),w}$
B_2	2nd row of C_{i-1}^w repeated q times	$A_{(2,1),w}$	$A_{(2,2),w}$	$\ldots$	$A_{(2,n_{i-1}(w)),w}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
B_{q^i}	q^ith row of C_{i-1}^w repeated q times	$A_{(q^i,1),w}$	$A_{(q^i,2),w}$	$\ldots$	$A_{(q^i,n_{i-1}(w)),w}$

Array $\tilde{C_i^w}$

Remark that the matrix $A_{(j,k),w}$ in the table corresponds to the symbol at the entry (j,k) of the array C_{i-1}^{w-1}.

Next, we prove that $\tilde{C_i^w}$ is a $w - PHF$.

Let $r_1, \ldots, r_w$ be any given w rows of $\tilde{C_i^w}$. If $r_1, \ldots, r_w$ belong to w different blocks, say $B_{i_1}, \ldots, B_{i_w}$, then from the definition of D_{i-1}^w there is at least one column in D_{i-1}^w containing pairwise distinct symbols in the rows $r_1, \ldots, r_w$. Assume that $r_1, \ldots, r_w$ belong to w' blocks, say $B_{i_1}, \ldots, B_{i_{w'}}$, where $w' \leq w-1$. As C_{i-1}^{w-1} is an $(w-1) - PHF$, there exists a column, say $\mathbf{c}$, whose symbols, say $j_1, \ldots, j_{w'}$, in the rows $i_1, \ldots, i_{w'}$ are pairwise distinct. From the definition of E_{i-1}^{w-1}, the symbols $j_1, \ldots, j_{w'}$ are replaced by matrices $A_{j_1,w}, \ldots, A_{j_{w'},w}$ (notice that $A_{j_1,w}, \ldots, A_{j_{w'},w}$ together form a set of $\binom{w}{2}$ consecutive columns of the blocks $B_{i_1}, \ldots, B_{i_{w'}}$ in E_{i-1}^{w-1}). By Lemma 3.8.14 $A_{j_1,w}, \ldots, A_{j_{w'},w}$ is an $(\binom{w}{2}, q.w', q, w) - PHF$, so there is

a column in E_{i-1}^{w-1} having different symbols in the rows $r_1, \ldots, r_w$. Thus $\tilde{C_i^w}$ is a $w - PHF$.

Now recall that C_{i-1}^{w-1} is an $q^i \times n_{i-1}(w-1)$ array and that E_{i-1}^{w-1} is obtained from C_{i-1}^{w-1} by replacing each entry $j \in \{0, \ldots, q-1\}$ of C_{i-1}^{w-1} by the $(q \times \binom{w}{2})$-matrix $A_{j,w}$.

Since the first column of each matrix $A_{j,w}$ is always the vector $(0, \ldots, q-1)^T$, there are $n_{i-1}(w-1)$ identical columns in $\tilde{C_i^w}$.

Now let C_i^w denote the array obtained from $\tilde{C_i^w}$ by deleting $n_{i-1}(w-1)-1$ of these identical columns. Then C_i^w is an $q^{i+1} \times n_i(w)$ array, where

$$\begin{aligned} n_i(w) &= n_{i-1}(w) + n_{i-1}(w-1) \times \binom{w}{2} - (n_{i-1}(w-1) - 1) \\ &= n_{i-1}(w) + n_{i-1}(w-1)(\binom{w}{2} - 1) + 1. \end{aligned}$$

It is obvious that C_i^w is an $w - PHF$, just as $\tilde{C_i^w}$.

As $n_{i-1}(w) = O(i^{w-1})$ and $n_{i-1}(w-1) = O(i^{w-2})$, we have

$$n_i(w) = O(i^{w-1}) + O(i^{w-2})[\binom{w}{2} - 1].$$

Consequently

$$n_i(w) = O((i+1)^{w-1}).$$

Hence C_i^w is an $(O((i+1)^{w-1}, q^{i+1}, q, w) - PHF$, as desired. ■

In the case $w = 3$ and $q = 3$ Theorem 3.8.15 gives us $(i^2, 3^i, 3, 3) - PHF$, for every integer $i \geq 1$.

Thus

$$n \approx 0.398(\log M)^2.$$

To compare the construction in Theorem 3.8.15 with other known constructions we recall that Corollary 3.8.2 gives perfect hash families with

$$n \approx 0.556(\log M)^2$$

and Corollary 3.8.4 provides

$$n \approx 0.796(\log M)^2.$$

Lemma 3.8.8 yields an $(i^2, 3^i, 3, 3) - PHF$ only if $i = 2^j$, where $j \geq 1$, whereas Theorem 3.8.15 gives an $(i^2, 3^i, 3, 3) - PHF$ for any integer $i \geq 1$.

Theorem 3.8.10 gives a better asymptotic performance, but in this case we are restricted with construction algorithm whose complexity is polynomial in n and we should also note that the smallest M we can get by this construction is larger or equal to 7^{98}.

It is worth noting that at each recursion step the size of the constructed perfect hash family in Theorem 3.8.15 increases much slower than that in Theorem 3.8.12. For example in second step of the construction algorithm in Theorem 3.8.12 we already obtain $M = 3^{1458}$ for $q = 3$ and $w = 3$.

Actually, Theorem 3.8.15 roughly states that a perfect hash family for a certain n can be constructed for any given w and any given code size q^i, where q is a prime power $q \geq \binom{w}{2}$ and $i \geq 1$ is any integer. Thus, Theorem 3.8.15 gives an explicit construction of perfect hash families for a very large set of parameter values.

As an application of Theorem 3.8.15 for the cases $w = 3$ and $w = 4$ we have:

Corollary 3.8.16 *For any prime power $q \geq 3$ there exists a $((i+1)^2, q^{i+1}, q, 3) - PHF$.*

Corollary 3.8.17 *For any prime power $q \geq 7$ there exists a $(\frac{5}{6}i^3 + \frac{5}{2}i^2 + \frac{11}{6}i + 1, q^{i+1}, q, 4) - PHF$.*

3.9 Summary

In this chapter we have studied combinatorial properties of perfect hash families. An existence result proved by a probabilistic methods states that for any $q \geq w$ there exists an infinite class of (n, M, q, w) perfect hash families with $n = O(\log M)$. However, it is a difficult problem to construct such an infinite class explicitly. Theorem 3.8.10 presents an infinite class of perfect hash families for any $w \leq q$ with $n = O(\log M)$. The known low-complexity algorithm for constructing these perfect hash families has a runtime upper-bounded by $(log_q M)^3$. This is considered still to be inefficient for practical purposes.

We present two new recursive explicit constructions for perfect hash families which provide infinite classes of perfect hash families satisfying different requirements. Firstly, for any $w \leq q$ we construct an infinite class of perfect hash families which have a very good asymptotic behavior. Secondly, we present a new construction technique which provides an infinite class of perfect hash families for very large parameter sets. This family is not as good asymptotically as the family obtained by our first construction. However, it provides 'good' perfect hash families of small sizes. We also prove a necessary condition for the exitance of a

perfect hash family in form of an upper bound on M. A comparison of bounds shows that our bound is stronger than other known bounds for many parameter sets. A survey of some significant known results, comparison of new results with known ones and several open problems have been provided.

Chapter 4

Identifiable Parent Property Codes

4.1 Introduction

This chapter deals with the problem of protection copyrighted digital data against piracy. The traitor tracing problem was introduced by Chor, Fiat and Naor in [51] 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 it back to at least one producer (parent) 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 [58]. In this scenario the content is divided into n segments. Each of this 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 goal of such schemes is to handle as many colluders as possible. The practical applications require to accommodate many users when there is a restriction on the number of symbols which can be used for marking the data.

Several codes providing some forms of traceability are designed to be used in these schemes. These codes have been extensively studied in the recent years. The weak forms are frameproof codes introduced by Boneh and Shaw [53], and secure frameproof codes [62]. We study strong forms of codes which 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, called the traceability constant. The strong form of codes studied in this chapter are identifiable parent property (IPP) codes which have been introduced by Hollmann, van Lint, Linnartz and Tolhuizen [55].

Other strong versions of such codes are TA schemes and TA codes introduced by Chor, Fiat and Naor in [51, 58, 60]. In fact, TA codes turn out to be a subclass of IPP codes [63].

Combinatorial properties of IPP codes and TA codes have been studied by several authors. Relationships of IPP codes with several other combinatorial structures and codes have been studied by Hollmann et al. [55], Staddon, Stinson and Wei [63], Barg, et al. [64] and Sarkar, Stinson [67]. Based on these connections several sufficient conditions on the existence of IPP codes are derived in [51, 55, 64, 68, 70, 76, 77]. Necessary conditions for the existence of IPP codes given in the form of an upper bound on the size of codes are obtained in [55, 63, 68, 75, 77]. Probabilistic techniques are used to prove the existence of w-IPP codes with $n = O(\log M)$, where n is the length of the codes and M is the size, for any alphabet of size $q > w$. Using the connections between IPP codes and other known combinatorial structures several explicit constructions are derived in [51, 63, 66, 67, 76]. The question of complexity of traitor tracing algorithms for IPP and TA codes are treated in [66, 73, 78]. Certain classes of TA codes are shown to have a fast traitor tracing algorithm by using the list decoding techniques.

In this dissertation we focus on explicit construction methods for IPP codes using recursion techniques.

Our first construction provides an infinite class of IPP codes with the best asymptotic behavior among explicitly constructed classes of IPP codes known in the literature. In fact, for any fixed $q > w$ we are able to construct an infinite class of w-IPP codes in which the length n of the codewords is $O((w^2)^{\log^*(M)}(\log(M))$, where M is the number of codewords and log^* is a very slow growing function. Moreover, we prove that these codes allow a traitor tracing algorithm with a runtime of $O(M)$ in general. It should be noted that no IPP codes other than TA codes with this property were known before [63]. For some infinite subclasses of these codes even faster, in time $poly(\log M)$ traitor tracing algorithms can be achieved.

Also, another new class of IPP codes is derived. We use perfect hash families and recursive techniques to derive an infinite class of IPP codes. This class of IPP codes is not as good asymptotically as the class of IPP codes constructed by our first construction method. However, the method provides 'good' IPP codes for certain parameter ranges.

The known construction methods and probabilistic existence results do not prove the existence of w-TA codes with $q < w^2$, then $b > q$, where b is the size of the code and q is the size of the alphabet. Thus, as an open problem Staddon, Stinson, and Wei [63] ask the following question: Can we construct w-TA codes with $q < w^2$ and $b > q$? We give an affirmative answer to the Staddon-Stinson-Wei's problem. Precisely, using the new general construction method for q-ary

codes with large Hamming distance given in Section 2.7, we are able to construct a large class of w-TA codes with $q < w^2$ and $b > q$.

Our results in this chapter have been presented in the papers [71], [72], [82], and [83].

In Section 4.2 we present some preliminaries. Section 4.3 summarizes the known results. Some open problems are discussed.

In Section 4.4 we prove that the concatenation of two IPP codes gives an IPP code. The parent identification process for the concatenated code is described. Using the concatenation technique we present a good infinite class of IPP codes.

In Section 4.5 we describe our first construction, in which the concatenation technique and the recursive method are combined. The construction yields an infinite class of IPP codes. We show the asymptotic behavior of the codes and study the complexity of a traitor tracing algorithm.

In Section 4.6 the new class of perfect hash families given in Theorem 3.8.15 is used to construct a new infinite class of IPP codes in view of Theorem 4.6.1

In Section 4.7 a new class of TA codes is derived based on the codes constructed in Section 2.7, which gives an answer to an open problem of the existence of TA codes for certain parameter classes. Finally, Section 4.8 summarizes new results.

4.2 Definitions

In this section we give basic definitions and notation used in this chapter.

Let $\mathcal{C}$ be a q-ary codes of length n. For any subset of codewords $C_0 \subseteq C$, the set of *descendants* of C_0, denoted **desc**(C_0), is defined by

$$\textbf{desc}(C_0) = \{x \in Q^n : x_i \in \{a_i : a \in C_0\},\ 1 \leq i \leq n\}.$$

Thus **desc**(C_0) consists of all n-tuples that could be produced by a coalition holding the codewords in C_0. If $x \in$ **desc**(C_0), then we say that C_0 produces x.

Let w be an integer. Define the *w-descendant code*, denoted **desc**$_w(C)$, as follows:

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

Thus **desc**$_w(C)$ consists of all n-tuples that could be produced by some coalition of size at most w.

Definition 4.2.1 *(Identifiable Parent Property code)*

Let $\mathcal{C}$ be an (n, M, q) code and let $w \geq 2$ be an integer. $\mathcal{C}$ is called an $(n, M, q, w) - IPP$ (Identifiable Parent Property) code provided that, for all $x \in \mathbf{desc}_w(C)$, it holds that

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

In other words, a code has 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.

Definition 4.2.2 *(TA code) Let define $I(x, y) = \{i : x_i = y_i\}$ for any $x, y \in Q^n$. Suppose $C \subseteq Q^n$ is an (n, b, q) code and $w \geq 2$ is an integer. $\mathcal{C}$ is called a w-TA code provided that, for all i and all $x \in \mathbf{desc}(\mathcal{C}_i)$, there is at least one codeword $y \in C_i$ such that $|I(x, y)| > |I(x, z)|$ for any $z \in C \setminus C_i$.*

The w-TA property eases the parent identification process allowing efficient traitor tracing algorithms (linear in the code size in the general case).

Example 4.2.3 *An $(5, 16, 4; 2)$-IPP code (2-TA).*

$$\begin{bmatrix}
1 & 1 & 1 & 1 & 1 \\
1 & 2 & 2 & 2 & 2 \\
1 & 3 & 3 & 3 & 3 \\
1 & 4 & 4 & 4 & 4 \\
2 & 1 & 2 & 3 & 4 \\
2 & 2 & 1 & 4 & 3 \\
2 & 3 & 4 & 1 & 2 \\
2 & 4 & 3 & 2 & 1 \\
3 & 1 & 3 & 4 & 2 \\
3 & 2 & 4 & 3 & 1 \\
3 & 3 & 1 & 2 & 4 \\
3 & 4 & 2 & 1 & 3 \\
4 & 1 & 4 & 2 & 3 \\
4 & 2 & 3 & 1 & 4 \\
4 & 3 & 2 & 4 & 1 \\
4 & 4 & 1 & 3 & 2
\end{bmatrix}.$$

Note that this is an $(5, 16, 4; 4)$-Reed-Solomon code. More generally, the relationship between Reed-Solomon codes and TA codes is given in Section 4.3.

4.3 Known Results

This section is a brief summary of basic known results and developments of the subject.

TA codes form a subclass of IPP codes. This fact is pointed out in the following lemma.

Lemma 4.3.1 ([63], Lemma 1.3) *An (n, M, q, w)-TA code is an (n, M, q, w)-IPP code.*

Proof: Suppose $\mathcal{C}$ is an (n, M, q, w)-TA code. If $x \in \mathbf{desc}_w(C)$, then there is a subset $C_i \subseteq C$, where $|C_i| = w$, such that $x \in \mathbf{desc}(C_i)$. Let $y \in C_i$ such that $|I(x, y)| \geq |I(x, z)|$ for any $z \in C_i$. Thus $|I(x, y)| \geq |I(x, z)|$ for any $z \in C$ by the definition of a w-TA code. We will show that, for any $C_j \subseteq C$ with $|C_j| \leq w$, $x \in \mathbf{desc}(C_j)$ implies $y \in C_j$. In fact, if $y \notin C_j$, then there is $l \in C_j$ such that $|I(x, l)| > |I(x, y)|$ by the definition of a w-TA code. This contradicts the fact that $|I(x, y)| \geq |I(x, z)|$ for any $z \in C$. ■

The converse of Lemma 4.3.1 is not true, as it can be easily checked with a small example.

Example 4.3.2 *A (2,4,4;4)-IPP code which is not a (2,4,4;2)-TA code*

$$\begin{bmatrix} 1 & 1 \\ 2 & 1 \\ 3 & 2 \\ 4 & 1 \end{bmatrix}.$$

It is easy to check that this code is an 4-IPP but it is not a 2-TA. It is a 4-IPP code, since the symbols in the first position of all the codewords are different. Now, take $x = 3\,1$, then x is a descendant of $\{3\,2, 4\,1\}$. However, $|I(x \cap 3\,2)| = |I(x \cap 4\,1)| = |I(x \cap 1\,1)| = 1$. Thus it is not 2-TA code.

More examples showing that a code having the IPP property do not necessarily have the TA property can be found in [55], [63] and [66].

The next result shows that w-IPP codes cannot exist for certain parameter situations.

Lemma 4.3.3 *[63] Suppose C is any (n, M, q) code, and $M - 1 \geq w \geq q$. Then C is not a w-IPP code.*

To avoid the triviality, hereafter we study IPP code with assumption that $q > w$ and $M > q$.

In addition to perfect hash families, which have been discussed in Chapter 3, we define here some other families of hash functions, namely separating hash families [62], partially hashing families [64] and strong separating hash families [67]. These structures are useful for our discussion in the sequel.

Definition 4.3.4 *(Separating hash family (SHF)) An (n, M, q)-hash family $\mathcal{H}$ is called an (n, M, q, w_1, w_2) separating hash family denoted (n, M, q, w_1, w_2)-SHF if for any two disjoint subsets A, B of $\{1, \cdots, M\}$ with $|A| = w_1$ and $|B| = w_2$, there is a function h in $\mathcal{H}$ such that $h(A)$ and $h(B)$ are disjoint.*

Definition 4.3.5 *(Partially hashing family (PAHF)) An (n, M, q)-hash family $\mathcal{H}$ is called an (n, M, q, t, u) partially hashing family denoted $(n, M, q, t, u) - PAHF$ if for any two subsets T, U of $\{1, \cdots, M\}$ such that $T \subset U$, $|T| = t$, and $|U| = u$, there is a function h in $\mathcal{H}$ such that for any $x \in T$ and any $y \in U$, with $y \neq x$ we have $h(x) \neq h(y)$.*

Definition 4.3.6 *(Strong separating hash family (SSHF)) An (n, M, q)-hash family $\mathcal{H}$ is called an (n, M, q, w_1, w_2)-strong separating hash family denoted $(n, M, q, w_1, w_2) - SSHF$ if for any two disjoint subsets A, B of $\{1, \cdots, M\}$ such that $|\mathsf{A}| = w_1$, and $|\mathsf{B}| = w_2$, there is a function h in $\mathcal{H}$ such that h injective on A and $h(\mathsf{A}) \cap h(\mathsf{B}) = \emptyset$.*

The connection of strong separating hash families and partially hashing families is as follows:

Theorem 4.3.7 *[67] A hash family $\mathcal{H}$ is an $(n, M, q, w_1, w_2) - SSHF$ if and only if it is an $(n, M, q, w_1, w_1 + w_2) - PAHF$.*

Recently, hash families have found many applications in cryptography. These applications are discussed for example in [31, 40, 39, 44, 62]. The relationships with IPP codes are presented below.

4.3.1 Connections Between IPP Codes and Other Combinatorial Structures

Connections between hash families and identifiable parent property codes have been studied in [55, 63, 64, 67]. We recall some of the results here.

For $w = 2$ necessary and sufficient conditions for the existence of 2-IPP codes using hash families are obtained in [55].

Theorem 4.3.8 *[55] Let C be the matrix representing an (n, M, q) code $\mathcal{C}$. Then $\mathcal{C}$ is a 2-IPP code if and only if C is simultaneously an $(n, M, q, 3) - PHF$ and an $(n, M, q, 2, 2) - SHF$.*

A relationship between perfect and separating hash families and w-IPP codes for any $w \geq 2$ is given in the following theorem:

Theorem 4.3.9 *[63] Let C be the matrix representing an (n, M, q) code $\mathcal{C}$. Suppose $\mathcal{C}$ is a w-IPP code. Then we have the following*

1. *C is an $(n, M, q, w + 1) - PHF$ if $M \geq w + 1$.*
2. *C is an $(n, M, q, w, w) - SHF$ if $M \geq 2w$.*

It is an open problem [63] whether the converse of the theorem is true for $w > 2$. A characterization of an $(n, M, q, 3) - IPP$ code is given in [64]. The cases $w = 4, 5$ have been studied in [81]. Results show that the converse of Theorem 4.3.9 is not true in these cases.

However, w-IPP codes can be obtained from certain perfect hash families. The following theorem, due to Staddon, Stinson and Wei [63], provides a sufficient condition for existence of IPP codes.

Theorem 4.3.10 *[63](Theorem 2.8) Let $\mathcal{C}$ be an (n, M, q) code whose matrix representation is C. If C is an $(n, M, q, \lfloor (w + 2)^2/4 \rfloor) - PHF$, then $\mathcal{C}$ is an $(n, M, q, w) - IPP$ code.*

This leaves the question of the existence of IPP codes for $w < q < \lfloor (w + 2)^2/4 \rfloor$ open, since a w-PHF exists if and only if $q \geq w$.

The connection of partially hashing families and IPP codes is shown in the next result.

Theorem 4.3.11 *[64] Let $\mathcal{C}$ be an (n, M, q) code whose matrix representation is C. If C is an $(n, M, q, w, \lfloor (w+2)^2/4 \rfloor) - PAHF$, then $\mathcal{C}$ is an $(n, M, q, w) - IPP$ code.*

Based on this result a probabilistic method is used to obtain a lower bound on the size of the codes in [64], showing that w-IPP codes exist for any $q \geq w + 1$.

A similar result in terms of strong separating hash families is given in [67] which is used later for constructing an infinite class of IPP codes.

Theorem 4.3.12 *[67] Let $\mathcal{C}$ be an (n, M, q) code whose matrix representation is C. If C is an $(n, M, q, w, \lfloor (w+2)^2/4 \rfloor - w) - SSHF$, then $\mathcal{C}$ is an $(n, M, q, w) - IPP$ code.*

The relationship of error correcting codes and IPP codes have been studied in several papers.

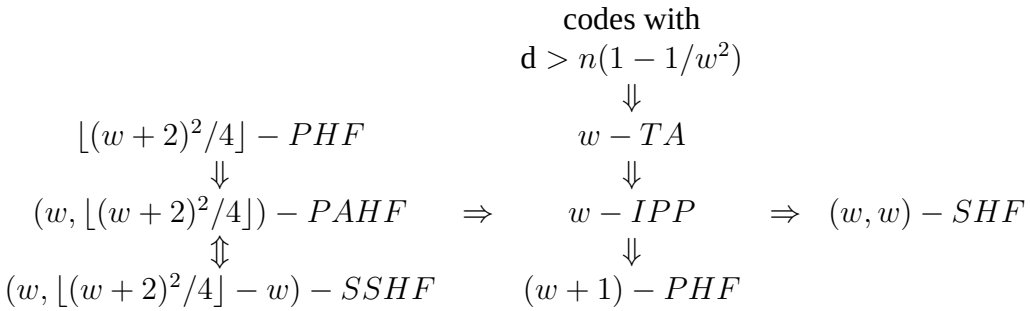
The following result stated in [51], [60], [63] is useful. We present it here with a simple proof.

Theorem 4.3.13 *Any $(n, b, q; d)$ code with $d > n(1 - 1/w^2)$ is an (n, b, q) w-TA code.*

Proof: Let $\mathcal{C}$ be an $(n, b, q; d)$ code with $d > n(1 - 1/w^2)$. Set $\alpha = n(1 - 1/w^2)$. Any two codewords $c_1, c_2 \in \mathcal{C}$ agree in at most $\beta = n - (\alpha + 1) = n/w^2 - 1$ positions. Let $\mathcal{C}' = \{c'_1, \ldots, c'_v\} \subseteq \mathcal{C}$ be a subset of size v. For any $u \in$ **desc**$(\mathcal{C}')$, define $M(u) = \max\{|I(u, c'_i)| : \ i = 1, \ldots, v\}$ and $M = \min_{u \in \textbf{desc}(\mathcal{C}')} M(u)$. Then $n/v \leq M$. On the other hand, for any $c \in \mathcal{C} \setminus \mathcal{C}'$ we have $\sum_{c'_i \in \mathcal{C}'} |I(c, c'_i)| \leq v\beta$. Now $\mathcal{C}$ will be a v-TA code if $v\beta < n/v$. Thus $\beta < n/v^2$, equivalently $n/w^2 - 1 < n/v^2$. Hence $v \leq w$, as desired. ■

The relationships between TA and IPP codes and hash families described in this section are depicted in the following diagram.

Figure 4.1: Connections among different types of codes and hash families



Further relationships between IPP codes and some other combinatorial structures not discussed here can be found in [63] (see Figure 1, [63]).

4.3.2 Necessary Conditions

Theorem 4.3.9 (case 1) asserts that w-IPP codes are subsets of $(w+1)$-PHF. Thus the necessary conditions for the existence of perfect hash families presented in

Section 3.3 also provide necessary conditions for the existence of w-IPP codes, which can be given in form of an upper bound on the size of the codes.

In particular, from bound (3.3.5) it follows:

Theorem 4.3.14 *Let $\mathcal{C}$ be an $(n, M, q; w) - IPP$ code, then*

$$M \leq w(q^{\lceil \frac{n}{w} \rceil} - 1).$$

A stronger upper bound for the size of IPP codes is given in the next theorem. Using an upper bound for separating hash families [63] (Theorem 3.9) together with Theorem 4.3.9 (case 2) we obtain:

Theorem 4.3.15 *[63] Let $\mathcal{C}$ be an $(n, M, q; w) - IPP$ code, then*

$$M \leq q^{\lceil \frac{n}{w} \rceil} + 2w - 2.$$

A stronger bound is established [55] for $w = 2$.

Theorem 4.3.16 *[55] Let $\mathcal{C}$ be an $(n, M, q; 2) - IPP$ code. Then*

$$M \leq 3q^{\lceil \frac{n}{3} \rceil}.$$

The bound in Theorem 4.3.15 has been improved in [75, 77].

Theorem 4.3.17 *[75] Let $\mathcal{C}$ be an $(n, M, q; w) - IPP$ code. Then*

$$M \leq \frac{1}{2} u(u-1) q^{\lceil \frac{n}{u-1} \rceil}$$

where $u = \lfloor (\frac{w}{2} + 1)^2 \rfloor$.

A similar upper bound with a somewhat better constant derived in [77] will be we presented in next section together with the lower bound.

4.3.3 Nonconstructive Existence Results

Probabilistic methods were used to prove existence results for IPP codes in [51, 55, 60, 64, 70, 76, 77]. The results are summarized below.

Theorem 4.3.18 *[51, 60] There exists an (n, b, q, w)-TA code, where $q = 2w^2$ and $n = 4w^2 \log b$.*

Note that this result does not prove the existence of w-TA codes for small q.

In the case of 2-IPP codes, the probabilistic method was used in [55] to prove the following result:

Theorem 4.3.19 *[55] There exists an $(n, M, q, 2)$-IPP code with $M \geq c(\frac{q}{4})^{\frac{n}{3}}$, where $c = (\frac{27}{32})^{\frac{1}{3}}$.*

The existence of w-IPP codes for any $q > w > 2$ was first proved in [64]. Theorem 4.3.11 together with an existence result for partially hashing families obtained from the probabilistic method provides next theorem:

Theorem 4.3.20 *[64]*

For any fixed q and w, $q > w$ there exists an infinite class of (n, M, q, w)-IPP code with

$$\lim_{n\to\infty} \frac{\log_q M}{n} \geq \frac{1}{u-1} \log_q \frac{(q-w)!q^u}{(q-w)!q^u - q!(q-w)^{u-w}}$$

where $u = \lfloor (\frac{w}{2}+1)^2 \rfloor$.

This result has been improved for all w except $w = 2, 3$.

Theorem 4.3.21 *[70]*

For any fixed $q = w+1$, there exists an infinite class of $(n, M, w+1, w)$-IPP code with

$$\lim_{n\to\infty} \frac{\log_{w+1} M}{n} \geq \frac{w!(u-w)^{u-w}}{u^u(u-1)\ln(w+1)}$$

where $u = \lfloor (\frac{w}{2}+1)^2 \rfloor$.

Performing simple asymptotic manipulations this result can be given as follows:

Theorem 4.3.22 *[76]*

There exists an absolute constant $c > 0$ such that for any fixed $q = w+1$, there exists an infinite class of $(n, M, w+1, w)$-IPP code with

$$\lim_{n\to\infty} \frac{\log_{w+1} M}{n} \geq \frac{cw!2^{2w}}{w^2(ew^2)^w} = w^{-w(1+o(1))}.$$

For fixed n, q, w; $q > w$ denote by $M(n, q, w)$ the maximum value of M for which an (n, M, q, w)-IPP code exists. It is proved in [77] that

Theorem 4.3.23 *[77] For every n, q, w; $q > w$ there exist two functions $c_1(w)$ and $c_2(w)$, such that*

$$(c_1(w)q)^{\frac{n}{s(w)}} \leq M(n, q, w) \leq c_2(w)q^{\lceil \frac{n}{s(w)} \rceil}$$

where

$$s(w) = \begin{cases} \frac{w^2}{4} + w & \text{when } w \text{ is even} \\ \frac{w^2}{4} + w - \frac{1}{4} & \text{when } w \text{ is odd} \end{cases}$$

These existence results together with the upper bounds on the size of the code theoretically state that for any fixed q and w such that $q > w$

$$n = O(\log M).$$

However, the existence results surveyed in this section are not constructive. Several explicit constructions of IPP codes will be presented in next sections.

4.3.4 Direct Constructions

Using the relationships between IPP codes and other combinatorial structures discussed in Section 4.3 several explicit classes of IPP codes can be derived.

Applying Theorem 4.3.13 to Reed-Solomon codes we obtain the following theorem:

Theorem 4.3.24 *[63] Suppose n, q and w are given, with q a prime power and $n \leq q + 1$. Then there exists an (n, b, q) w-TA code with $b = q^{\lceil n/w^2 \rceil}$.*

This construction is explicit and gives w-TA codes with

$$n = O(\log b).$$

It gives better parameters than the probabilistic method in Theorem 4.3.18. However, this class is quite restricted because of condition $n \leq q + 1$. The codes in this construction only have $b > q$ only if $n > w^2$, i.e., if $q \geq w^2$.

By combining Corollary 2.5.6 and Theorem 4.3.13 we obtain the following more general theorem:

Theorem 4.3.25 *[82] Let $w \geq 2$ be any given integer. For any integer $n > w^2$ and s having $s = p_1^{e_1} \dots p_k^{e_k}$ as its prime factorization with $n \leq p_i^{e_i}$ for all $i = 1, \dots, k$ there exists an $(n, M, s, w) - IPP$ code, where $M = s^{\lceil n/w^2 \rceil}$.*

Here we describe another nice class of w-TA codes which can be derived as an application of Theorem 4.3.13.

Let $\mathcal{C}$ be a linear AG code defined on the Garcia-Stichtenoth (G-S) curves with parameters given in Theorem 2.6.1.

Applying Theorem 4.3.13 to $\mathcal{C}$ we obtain the following result:

Theorem 4.3.26 *For every prime power q and any integer $l \geq 1$, there exists an $(n; b, q^2, w)$-IPP code, where*

$n = q^{l+1}(q-1)$,
$b = q^{2\lfloor uq^{l+1} \rfloor}$,
u is a real number with $1 \leq u \leq q-2$, *and*
$w = \lfloor \sqrt{\frac{q-1}{u+1}} \rfloor$.

This construction gives an infinite class of w-TA codes, where $n = O(\log b)$. However, this class is restricted to the condition that q is a prime power and is very large compared to w. Also the known construction algorithm complexity of these codes is polynomial in n as discussed in Section 2.6.

The constructions which are direct applications of Theorem 4.3.13 are restrictive, in general, in the sense that they can produce w-TA codes only if $q > w^2$ when b is large. This fact follows from the Plotkin bound. The Plotkin bound given in Corollary 2.2.3 together with Theorem 4.3.13 implies:

$$n(1 - \frac{1}{w^2}) < d \leq n\left(1 - \frac{1}{q}\right)\frac{b}{b-1}.$$

Thus

$$1 - \frac{1}{w^2} < \left(1 - \frac{1}{q}\right)\frac{b}{b-1}.$$

So when $b \to \infty$ for fixed q and w, we have

$$w^2 < q.$$

For small b, however, we can construct w-TA codes with $q \leq w^2$ and $w^2 < \frac{n}{n-d}$.

In Section 4.7 a new class of w-TA codes is obtained by applying Theorem 4.3.13 to the q-ary codes constructed in Section 2.7. This provides examples of w-TA codes with $q < w^2$ and $b > q$.

4.3.5 Efficient Traitor Tracing

In this section we discuss briefly the problems concerning the complexity of the traitor tracing algorithms(TTA) of IPP codes.

It is clear that for any given IPP code the traitor tracing can be carried out in time $O(\binom{M}{w})$, where M is the number of users (code size). This is to say if we have no other better idea than to check all coalitions of size w to find at least one of the dishonest users (the parent).

To find the parent in case of w-TA codes it is enough to check the distance of the descendent vector (recognized illegal copy) to each codeword of the code and find the "nearest" codeword to it.

Thus, in the general case, the runtime of a TTA for a TA code is $O(b)$, where b is the number of the users. This is still inefficient for large populations.

Efficient traitor tracing algorithms can be applied to TA codes, when error correcting codes are used to construct these codes. This problem is discussed first by Silverberg, Staddon and Walker in [66]. They show that powerful new techniques for the list decoding of error correcting codes discussed in Section 2.8 enable to construct a very fast TTA for some TA codes. When algebraic geometry codes, Reed Solomon codes or some concatenated codes are used as TA codes, then traitor tracing can be done in time polynomial in $(w(\log M))$. These traitor tracing algorithms produce a list of all coalitions capable of creating a given pirate. The results are summarized in following theorem:

Theorem 4.3.27 *[66]*

(i) Let $\mathcal{C}$ be a Reed-Solomon code of length n and dimension k over a finite field F_q of size at most 2^n. If w is an integer, $w \geq 2$, and $n > w^2(k-1)$, then $\mathcal{C}$ is a w-TA code and there is a traitor tracing algorithm that runs in time $O(n^{15})$. If $n = (1+\delta)w^2(k-1)$ then the algorithm runs in time $O(\frac{n^3}{\delta^6})$. For $n = \theta(w^2 k)$, the runtime is $O(w^{30}\log_q{}^{15} M)$.

(ii) Let X be a nonsingular plane curve of genus g defined over a finite field F_q, $\mathcal{P}$ a set of n distinct F_q-rational point on X, P_0 an F_q-rational point on X which is not in $\mathcal{P}$, and k an integer such that $k > g-1$. Let w be an integer such that $w \geq 2$ and $n > w^2(k+g-1)$, assume that $q \leq 2^n$, and assume the pre-processing described in [24] has occurred. Then the one-point AG code $C_X(\mathcal{P}, (k+g-1)P_0)$ is a w-TA code with a traitor tracing algorithm that runs in time polynomial in n.

(iii) If k and w are positive integers, q is a prime power, $q > w^2 \geq 4$, and δ is a real number such that $0 < \delta \leq \frac{q/w^2-1}{q-1}$, then there exists an explicit linear w-TA code over the field F_q of length $n = O(\frac{k^2}{\delta^3 \log(1/\delta)})$ (or length $n = O(\frac{k}{\delta^2 \log^2(1/\delta)})$) and dimension k with a polynomial (in n) traitor tracing algorithm.

It should be noticed that the IPP codes given by this theorem are derived from direct applications of Theorem 4.3.13. Thus Theorem 4.3.27 does not provide IPP codes for fixed small q, when $M \to \infty$. Roughly speaking, the alphabet size of these codes is $O(M^{\frac{w^2}{n}})$. In [66], the authors discus potential applications of other decoding methods to the problem of tracing traitors and suggest alternative approaches when additional information is known about the way the traitors are operating. Further developments in this direction can be found in [78, 79, 80].

4.4 Concatenation Construction of IPP Codes

In this section , in view of Theorem 4.4.1, we prove that the concatenation of two IPP codes gives an IPP code. The proof of Theorem 4.4.1 describes, at the same time, a traitor tracing algorithm for the resulting codes by using the traitor tracing algorithm initial codes. Finally, an infinite class of IPP codes is presented in Theorem 4.4.3 for any fixed q and w, $q > w \geq 2$.

Let $\mathcal{A}$ be an (n_2, M_2, q_2) code over an alphabet Q_2 with $|Q_2| = q_2$ and let $\mathcal{B}$ 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 $B = \{\mathbf{b_1}, \ldots, \mathbf{b_{q_2}}\}$. Let $\theta : Q_2 \longrightarrow B$ be the one-to-one mapping defined by $\theta(a_i) = \mathbf{b_i}$ for $1 \leq i \leq q_2$. For any codeword $\mathbf{a} = (a_1, \ldots, a_{n_2}) \in A$ we denote by $\tilde{\mathbf{a}} = (\theta(a_1), \ldots, \theta(a_{n_2})) = (\mathbf{b_1}, \ldots \mathbf{b_{n_2}})$ the q_1-ary sequence of length $n_1 n_2$ obtained from $\mathbf{a}$ by using θ. The set $\mathcal{C} = \{\tilde{\mathbf{a}} = (\mathbf{b_1}, \ldots, \mathbf{b_{n_2}}) / \ \mathbf{a} = (a_1, \ldots, a_{n_2}) \in A\}$ is an $(n_1 n_2, M_2, q_1)$ code, called the concatenated code of $\mathcal{A}$ and $\mathcal{B}$.

Our next important theorem shows that the concatenation technique works for IPP codes.

Theorem 4.4.1 *[82] Let $\mathcal{A}$ be an $(n_2, M_2, q_2, w) - IPP$ code and let $\mathcal{B}$ be an $(n_1, q_2, q_1, w) - IPP$ code. Then the concatenated code $\mathcal{C}$ of $\mathcal{A}$ and $\mathcal{B}$ is an $(n_1 n_2, M_2, q_1, w) - IPP$ code.*

Proof: Let $\mathbf{x} = (x_1, \ldots, x_{n_1 n_2}) \in Q_1^{n_1 n_2}$. We partition $\mathbf{x}$ into n_2 blocks $\mathbf{x}_1, \ldots, \mathbf{x}_{n_2}$ with $\mathbf{x}_i = (x_{(i-1)n_1+1}, \ldots, x_{in_1}) \in Q_1^{n_1}$, $1 \leq i \leq n_2$. We will write $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_{n_2})$. Specially, if $\mathbf{x} = \mathbf{c} = (\mathbf{b}_1, \ldots, \mathbf{b}_{n_2}) \in \mathcal{C}$, then $\mathbf{b}_i$'s are themselves blocks of the partition of $\mathbf{c}$.

Suppose $\mathbf{x} \in \mathbf{desc}(\mathcal{C}_i)$, $1 \leq i \leq r$, where $\mathcal{C}_i \subseteq \mathcal{C}$ with $|\mathcal{C}_i| = \alpha_i \leq w$. We prove that $\bigcap_{1 \leq i \leq r}(\mathcal{C}_i) \neq \emptyset$, i.e., $\mathcal{C}$ is a $w - IPP$ code.

Let $\mathcal{C}_i = \{\mathbf{c}_1^{(i)}, \ldots, \mathbf{c}_{\alpha_i}^{(i)}\} \subseteq \mathcal{C}$, where $\mathbf{c}_j^{(i)} = (\mathbf{b}_{j1}^{(i)}, \ldots, \mathbf{b}_{jn_2}^{(i)})$. For any $1 \leq i \leq r$ and any $1 \leq \ell \leq n_2$ define $D_\ell^{(i)} = \{\mathbf{b}_{1\ell}^{(i)}, \ldots, \mathbf{b}_{\alpha_i \ell}^{(i)}\}$, i.e., $D_\ell^{(i)}$ is the collection of all ℓ^{th} blocks of the codewords of $\mathcal{C}_i$. In other words $D_\ell^{(i)} \subseteq B$ is a subset of α_i codewords. As $\mathbf{x} \in \mathbf{desc}(\mathcal{C}_i)$ by the assumption, we have $\mathbf{x}_\ell \in \mathbf{desc}(D_\ell^{(i)})$ for $1 \leq i \leq r$ and $1 \leq \ell \leq n_2$. Since B is a $w - IPP$ code, we have

$$\bigcap_{1 \leq i \leq r} D_\ell^{(i)} \neq \emptyset.$$

Let $\mathbf{b}_\ell \in \bigcap_{1 \leq i \leq r} D_\ell^{(i)}$ be an arbitrary but fixed codeword, i.e., $\mathbf{b}_\ell$ is a parent of $\mathbf{x}_\ell$ in code B. Set $\mathbf{y} = (\mathbf{b}_1, \ldots, \mathbf{b}_{n_2})$. Let $\bar{\mathbf{y}} = (a_1, \ldots, a_{n_2}) \in Q^{n_2}$ be

the corresponding sequence obtained from $\mathbf{y}$ using θ, i.e., $a_i = \theta^{-1}(\mathbf{b}_i)$. In the same way let $\bar{\mathcal{C}}_i = \{\bar{\mathbf{c}}_1^{(i)}, \ldots, \bar{\mathbf{c}}_{\alpha_i}^{(i)}\} \subseteq A$ denote the corresponding subset of $\mathcal{C}_i$.

Since $\mathbf{y} \in \mathbf{desc}(\mathcal{C}_i)$ by the construction, we have $\bar{\mathbf{y}} \in \mathbf{desc}(\bar{\mathcal{C}}_i)$ for $1 \leq i \leq r$. Hence

$$\bar{\mathbf{y}} \in \bigcap_{1 \leq i \leq r} \mathbf{desc}(\bar{\mathcal{C}}_i).$$

Since $\mathcal{A}$ is a $w - IPP$ code, we have

$$\bigcap_{1 \leq i \leq r} \bar{\mathcal{C}}_i \neq \emptyset.$$

Let $\bar{\mathbf{z}}' = (a'_1, \ldots, a'_{n_2}) \in \bigcap_{1 \leq i \leq r}(\bar{\mathcal{C}}_i)$ be a parent of $\bar{\mathbf{y}}$ in $\mathcal{A}$. Then $\mathbf{z}' = (\mathbf{b}'_1, \ldots, \mathbf{b}'_{n_2}) \in C_i$ for $1 \leq i \leq r$, where $\mathbf{z}'$ the codeword of $\mathcal{C}$ corresponding to $\bar{\mathbf{z}}'$. Therefore

$$\bigcap_{1 \leq i \leq r} \mathcal{C}_i \neq \emptyset.$$

Thus $\mathcal{C}$ is an $w - IPP$ code. ■

Remark: *Note that the proof of Theorem 4.4.1 describes how to identify a traitor. This fact is used for the proof of Theorem 4.5.4.*

■

We demonstrate an IPP code obtained from concatenating two IPP codes in the following example.

Example 4.4.2 *Let* A *be the following code*

$$\mathsf{A} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 2 & 2 \\ 2 & 2 & 2 & 2 & 1 \\ 0 & 1 & 2 & 1 & 0 \end{bmatrix}.$$

It is easy to check that A *is a 2-TA code. And let* B *be the following 2-TA code.*

$$\mathsf{B} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 2 & 2 & 2 & 2 & 0 \\ 3 & 3 & 3 & 3 & 0 \\ 3 & 2 & 1 & 0 & 1 \\ 2 & 3 & 0 & 1 & 1 \\ 1 & 0 & 3 & 2 & 1 \\ 0 & 1 & 2 & 3 & 1 \\ 1 & 3 & 2 & 0 & 2 \\ 0 & 2 & 3 & 1 & 2 \\ 3 & 1 & 0 & 2 & 2 \\ 2 & 0 & 1 & 3 & 2 \\ 2 & 1 & 3 & 0 & 3 \\ 3 & 0 & 2 & 1 & 3 \\ 0 & 3 & 1 & 2 & 3 \\ 1 & 2 & 0 & 3 & 3 \end{bmatrix}.$$

Now we define the following mapping of alphabet symbols of B *to the rows of* A.

$$\theta : \begin{cases} 0 & \mapsto & (0 & 0 & 0 & 0 & 1) \\ 1 & \mapsto & (1 & 1 & 1 & 2 & 2) \\ 2 & \mapsto & (2 & 2 & 2 & 2 & 1) \\ 3 & \mapsto & (0 & 1 & 2 & 1 & 0) \end{cases}$$

Applying this mapping to B *we obtain a code* $\mathcal{C}$ *with parameters* $n = 25$, $b = 16$, $q = 3$*:*

$$\left[\begin{array}{ccccc|ccccc|ccccc|ccccc|ccccc}
0&0&0&0&1&0&0&0&0&1&0&0&0&0&1&0&0&0&0&1&0&0&0&0&1\\
1&1&1&2&2&1&1&1&2&2&1&1&1&2&2&1&1&1&2&2&0&0&0&0&1\\
2&2&2&2&1&2&2&2&2&1&2&2&2&2&1&2&2&2&2&1&0&0&0&0&1\\
0&1&2&1&0&0&1&2&1&0&0&1&2&1&0&0&1&2&1&0&0&0&0&0&1\\
0&1&2&1&0&2&2&2&2&1&1&1&1&2&2&0&0&0&0&1&1&1&1&2&2\\
2&2&2&2&1&0&1&2&1&0&0&0&0&0&1&1&1&1&2&2&1&1&1&2&2\\
1&1&1&2&2&0&0&0&0&1&0&1&2&1&0&2&2&2&2&1&1&1&1&2&2\\
0&0&0&0&1&1&1&1&2&2&2&2&2&2&1&0&1&2&1&0&1&1&1&2&2\\
1&1&1&2&2&0&1&2&1&0&2&2&2&2&1&0&0&0&0&1&2&2&2&2&1\\
0&0&0&0&1&2&2&2&2&1&0&1&2&1&0&1&1&1&2&2&2&2&2&2&1\\
0&1&2&1&0&1&1&1&2&2&0&0&0&0&1&2&2&2&2&1&2&2&2&2&1\\
2&2&2&2&1&0&0&0&0&1&1&1&1&2&2&0&1&2&1&0&2&2&2&2&1\\
2&2&2&2&1&1&1&1&2&2&0&1&2&1&0&0&0&0&0&1&0&1&2&1&0\\
0&1&2&1&0&0&0&0&0&1&2&2&2&2&1&1&1&1&2&2&0&1&2&1&0\\
0&0&0&0&1&0&1&2&1&0&1&1&1&2&2&2&2&2&2&1&0&1&2&1&0\\
1&1&1&2&2&2&2&2&2&1&0&0&0&0&1&0&1&2&1&0&0&1&2&1&0
\end{array}\right]$$

From Theorem 4.4.1 it follows that $\mathcal{C}$ *is a 2-IPP code.*

We remark that the concatenation of two w-TA codes does not necessarily give a w-TA code. In particular the code $\mathcal{C}$ from Example 4.4.2 is not a 2-TA code.

We show that the code $\mathcal{C}$ is not a 2-TA code as follows:

Let the vector

$$x = \{1112111122010010000100001\}$$

be a descendent vector of the set $\mathcal{C}_0 = \{a, b\}$, where

$$a = \{0000100001000010000100001\}$$

and

$$b = \{1112211122111221112200001\}$$

are the first two rows of $\mathcal{C}$.

On one hand it can be computed that $I(x, a) = I(x, b) = 15$. On the other hand we have $I(x, c) = 15$ for

$$c = \{2222111122012100000101210\}.$$

Thus $\mathcal{C}$ is not a 2-TA code.

Now we demonstrate how the traitor tracing algorithm given in the proof of Theorem 4.4.1 works.

First we partition x into 5 blocks as follows:

$$x = \{11121|11122|01001|00001|00001\}$$

For each block we find a parent in the code A. Since A is a 2-TA code we have

$$\begin{array}{lll} (1\ 1\ 1\ 2\ 1) & \mapsto & (1\ 1\ 1\ 2\ 2) \\ (1\ 1\ 1\ 2\ 2) & \mapsto & (1\ 1\ 1\ 2\ 2) \\ (0\ 1\ 0\ 0\ 1) & \mapsto & (0\ 0\ 0\ 0\ 1) \\ (0\ 0\ 0\ 0\ 1) & \mapsto & (0\ 0\ 0\ 0\ 1) \\ (0\ 0\ 0\ 0\ 1) & \mapsto & (0\ 0\ 0\ 0\ 1) \end{array}$$

Thus we obtain

$$y = \{11122|11122|00001|00001|00001\}.$$

Now applying the mapping θ^{-1} to each block we get

$$\bar{y} = \{1|1|0|0|0\}.$$

Next we need to find a parent of $\bar{y}$ in the code $\mathcal{B}$. It is easy to check that $I(\bar{y}, \bar{a}) = I(\bar{y}, \bar{b}) = 3$ and $I(\bar{y}, \bar{c}) < 3$, for any $\bar{c} \in \mathsf{B} \backslash \{\bar{a}; \bar{b}\}$. This shows that $\bar{a}$ and $\bar{b}$ are parents of $\bar{y}$ in $\mathcal{B}$. Thus a and b are parents of x in the code $\mathcal{C}$.

As a first application of Theorem 4.4.1 we obtain the following:

Theorem 4.4.3 *For every given integers w and m; where $m > w \geq 2$, and for any integer $l \geq 1$, there exists an (n, M, m, w)-IPP code, with*

$n = n_0.(q-1)q^{l+1}$,
$M = q^{2\lfloor uq^{l+1} \rfloor}$,
where n_0 is a constant, q is a prime power such that $q \geq w^2(u+1)+1$,
and u is a real number with $1 \leq u \leq q-2$.

Moreover, we have $n = O(\log M)$.

Proof: Let w and m be given integers with $m \geq w \geq 2$. Let q be the smallest prime power such that $w = \lfloor \sqrt{\frac{q-1}{u+1}} \rfloor$, with $1 \leq u \leq q-2$, as shown in Theorem 4.3.26. The existence of an (n_0, q^2, m, w)-IPP code for a certain value n_0 is shown for in Theorem 4.3.20. Applying Theorem 4.4.1 and Theorem 4.3.26 yields the IPP codes with parameters as claimed. ■

From the proof of Theorem 4.4.1 it follows that the runtime of the traitor tracing algorithm of a code in Theorem 4.4.3 differs from the traitor tracing algorithm runtime of a corresponding code from Theorem 4.3.26 by a constant factor. This together with Theorem 4.3.27 (case (ii)) implies that the codes in Theorem 4.4.3 have a traitor tracing algorithm that runs in time polynomial in $\log M$. So, the complexity of the construction algorithm and the traitor tracing algorithm runtime of the codes in Theorem 4.4.3 grow polynomially with $\log M$.

It can be easily checked that the minimum distance of $\mathcal{C}$ in Theorem 4.4.1 does not satisfy the condition of Theorem 4.3.13 whereas the minimum distance of the codes $\mathcal{A}$ and $\mathcal{B}$ do. This proves that the converse of Theorem 4.3.13 is not true. It can be observed that with the traitor tracing algorithm described in the proof of Theorem 4.4.1 a descendent of code $\mathcal{C}$ is not necessarily traced back to its nearest vector even when $\mathcal{A}$ and $\mathcal{B}$ are w-TA codes. Thus code $\mathcal{C}$ does not necessarily have a w-TA property. In the next section, using the concatenation construction together with some recursive techniques we show the existence of a very good class of IPP codes allowing an efficient traitor tracing algorithm for the range of parameters, for which the existence of TA codes is not known. This emphasizes the advantage of considering codes with w-IPP property than only codes with w-TA property.

We remark that Barg and Kabatiansky in a recent paper [73] also prove a similar result by concatenating IPP codes with error correcting codes.

In the next section we present an explicit construction of IPP codes which provides an infinite class of IPP codes for any fixed $q > w$ with an efficient traitor tracing algorithm and with a best known asymptotic behavior among explicitly constructed codes.

4.5 Infinite Class of w-IPP Codes with Efficient TTA

We are now in a position to describe our first construction. First, we describe the construction by making use of Theorem 4.3.25 and 4.4.1. The result is presented in Theorem 4.5.1. The asymptotic behavior of these codes is shown in Theorem 4.5.2. Using the same method a more general result is obtained, which is formulated in Theorem 4.5.3. Theorem 4.5.4 shows that the codes of Theorem 4.5.1 have a traitor tracing algorithm with a runtime of $O(M)$. Theorem 4.5.5 summarizes our main results. Finally, an infinite subclass of these codes having a traitor tracing algorithm with a runtime $poly(\log M)$ is given in Theorem 4.5.6.

4.5.1 Recursive Construction

The construction is carried out by induction on the number of iterations.

Let $w \geq 2$ be any integer. Let $n_0 > w^2$ be an integer and let s_0 be an integer with the prime factorization $s_0 = p_1^{e_1} \dots p_k^{e_k}$ such that $n_0 \leq p_i^{e_i}$ for all $i = 1, \dots, k$.

For the 1^{st} iteration we choose two codes $\mathcal{C}_0$ and $\mathcal{C}_1^*$ using Theorem 4.3.25:
C_0 is an $(n_0, M_0, s_0, w) - IPP$ code with $M_0 = s_0^{\lceil \frac{n_0}{w^2} \rceil}$;
C_1^* is an $(n_0^*, M_1, M_0, w) - IPP$ code with $n_0^* = n_0^{\lceil \frac{n_0}{w^2} \rceil}$ and $M_1 = M_0^{\lceil n_0^*/w^2 \rceil}$.

Applying Theorem 4.4.1 with $\mathcal{A}$ replaced by C_1^* and $\mathcal{B}$ by $\mathcal{C}_0$ we obtain an $(n_1, M_1, s_0, w) - IPP$ code C_1 with $n_1 = n_0 * n_0^* = n_0 * n_0^{\lceil \frac{n_0}{w^2} \rceil}$.

Now an $(n_{i-1}, M_{i-1}, s_0, w) - IPP$ code C_{i-1} exists by induction for the $(i-1)^{th}$ iteration. Choose an $(n_{i-1}^*, M_i, M_{i-1}, w) - IPP$ code C_i^* from Theorem 4.3.25 with

$$n_{i-1}^* = n_{i-2}^{*\lceil \frac{n_{i-2}^*}{w^2} \rceil} \quad \text{and} \quad M_i = M_{i-1}^{\lceil \frac{n_{i-1}^*}{w^2} \rceil}.$$

Applying Theorem 4.4.1 with $A = C_i^*$ and $B = C_{i-1}$, we get an $(n_i, M_i, s_0, w) - IPP$ code C_i with

$$n_i = n_{i-1} * n_{i-2}^{*\lceil \frac{n_{i-2}^*}{w^2} \rceil}.$$

Thus we obtain the following result:

Theorem 4.5.1 *[82] Let $w \geq 2$ be any integer. Let $n_0 > w^2$ be integer and let s_0 be an integer with the prime factorization $s_0 = p_1^{e_1} \dots p_k^{e_k}$ such that $n_0 \leq p_i^{e_i}$ for all $i = 1, \dots, k$. Then, for all $h \geq 0$ there exists an $(n_h, M_h, s_0, w) - IPP$ code, where*

$$n_h = n_{h-1} * n_{h-1}^*, \quad M_h = M_{h-1}^{\lceil \frac{n_{h-1}^*}{w^2} \rceil}, \quad n_{h-1}^* = n_{h-2}^{*\lceil \frac{n_{h-2}^*}{w^2} \rceil},$$
$$M_0 = s_0^{\lceil \frac{n_0}{w^2} \rceil}, \quad \text{and} \quad n_0^* = n_0^{\lceil \frac{n_0}{w^2} \rceil}.$$

4.5.2 Asymptotic Behavior

The asymptotic behavior of the parameters of the codes produced by Theorem 4.5.1 can be examined by a similar argument, which is demonstrated in Section 3.8 (see also [46], pp. 196-197.) In fact, we can show that

$$n_h \leq \alpha.(w^2)^{\log^*(M_h)}(\log M_h),$$

for all sufficiently large h, where α is some constant and the function $\log^* : \mathbb{Z}^+ \longrightarrow \mathbb{Z}^+$ is defined recursively by

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

Note that the function $\log^*(n)$ grows very slowly, e.g., $\log^*(n) \leq 7$ for $n \leq 2^{2^{65536}}$.

We have the following result:

Theorem 4.5.2 *[82] For any integer $w \geq 2$ and any integer s having the prime factorization $s = p_1^{e_1} \dots p_k^{e_k}$ with $w^2 < p_i^{e_i}$ for all $i = 1, \dots, k$, there exists an infinite class of $(n, M, s, w) - IPP$ codes for which n is $O((w^2)^{log^*(M)}(log(M))$.*

As we want to show that the constructed codes in Theorem 4.5.1 having an efficient tracing algorithm, we have chosen the starter code as an MDS code. In fact, the construction works for any starter code. For instance, for given $M, q, w \geq 2$, the probabilistic method in [64] shows the existence of $(n', M, q, w) - IPP$ codes with $q > w$ and some n'. Thus, if we take this $(n', M, q, w) - IPP$ code as a starter code and carry out the same recursive construction, then we get a more general result as follows:

Theorem 4.5.3 *[82] For any integer $w \geq 2$ and any integer $q \geq w$, there exists an infinite class of $(n, M, q, w) - IPP$ codes for which n is $O((w^2)^{log^*(M)}(log(M))$.*

To our knowledge Theorem 4.5.2 and 4.5.3 yield a class of explicit constructed codes with the best known asymptotic behavior among similar known classes. In fact, from Theorem 4.3.10 it follows that an $(n, M, q, w) - IPP$ code is an $(n, M, q, w+1) - PHF$, and therefore an $(n, M, q, w) - PHF$. But the converse is not true: an $(n, M, q, w+1) - PHF$ is not an $(n, M, q, w) - IPP$ code in general. This is to say that an $(n, M, q, w) - IPP$ code is a much stronger structure than an $(n, M, q, w) - PHF$. Even though, our constructed IPP codes have the same asymptotic size as that of the best known explicitly constructed classes of PHF (see Section 3.8).

Remark: *It is worth noting that in a recent paper [67], Sarkar and Stinson construct an infinite class of (n, M, q, w)-IPP codes for which n is $O((w^3)^{log^*(M)}(log(M))$, for integers $q > w \geq 2$ in terms of strong separating hash families.* ■

4.5.3 An Efficient Traitor Tracing Algorithm

The discussions in Section 4.3 show that for w-IPP codes, a traitor tracing algorithm (TTA) will have a runtime complexity of size $O(\binom{M}{w})$, in general. For w-TA codes, however, the runtime of a TTA will be $O(M)$. Therefore, the question of the existence of w-IPP codes in general with an improved runtime for a TTA was raised in [63].

Here, we show that our constructed w-IPP codes have a TTA with a runtime $O(M)$, thereby answering the above question affirmatively.

The recursive process of concatenation used to construct $w - IPP$ codes in Theorem 4.5.1 provides a way to build a TTA for code C_i based on the TTA's of codes C_{i-1} and C_i^*. In fact, the proof of Theorem 4.4.1 describes precisely how a traitor can be traced back for the code C_i. In doing so we assume that the TTA's for codes C_{i-1} and C_i^* are known. Let L_{i-1} and L_i^* be the runtime complexity of such a TTA for C_{i-1} and C_i^*, respectively. Let assume $\mathbf{x} \in \mathbf{desc}(K_j)$, for $j = 1, \ldots, r$, and $K_j \subseteq C_i$ with $|K_j| \leq w$, i.e., $\mathbf{x}$ is a pirate word of length $n_i = n_{i-1} * n_{i-1}^*$ created by r possible coalitions K_j. From the proof of Theorem 4.4.1 we see that the runtime L_i of a TTA for code C_i is given by

$$L_i = L_{i-1} * n_{i-1}^* + L_i^*. \tag{4.1}$$

If we start with C_0 and C_1^* as w-TA codes, for which the runtime of their TTAs are $O(M_0)$ and $O(M_1)$, then we have $L_1 = O(M_1)$, as $|M_0| << |M_1|$. Therefore, if C_i^* is a w-TA code for each step of the recursion, then we have $L_i = O(M_i)$. Now the codes C_0 and C_i^* in Theorem 4.5.1 are in fact w-TA codes, so we have the following result:

Theorem 4.5.4 *[82] For any integer $w \geq 2$ and any integer s having the prime factorization $s = p_1^{e_1} \ldots p_k^{e_k}$ with $w^2 < p_i^{e_i}$ for all $i = 1, \ldots, k$, there exists an infinite class of $(n, M, s, w) - IPP$ codes with n is $O((w^2)^{log^*(M)}(log(M))$, which have a traitor tracing algorithm of linear runtime $O(M)$.*

Proof: Let C_i be the code obtained in the i-th recursion step of the Theorem 4.5.1. Then L_i, the runtime of a TTA for it , is given by (4.1).

Since $n_0 \leq s_0$ we have $n_0^* \leq M_0$. Thus, from $n_{i-1}^* = {n_{i-2}^*}^{\lceil \frac{n_{i-2}^*}{w^2} \rceil}$ and $M_{i-1} = {M_{i-2}}^{\lceil \frac{n_{i-2}^*}{w^2} \rceil}$ we obtain that $n_{i-1}^* \leq M_{i-1}$ for any $i \geq 0$.

Now, as C_{i-1} is a w-IPP code of size M_{i-1} we get $L_{i-1} \leq O(\binom{M_{i-1}}{w}) = O({M_{i-1}}^w)$. Hence $L_{i-1} * n_{i-1}^* \leq O({M_{i-1}}^{w+1})$.

This together with the fact that $M_i = M_{i-1}^{\lceil \frac{n_{i-1}^*}{w^2} \rceil}$ implies that

$$L_{i-1} * n^*_{i-1} \leq O(M_i) \tag{4.2}$$

when $n^*_{i-1} \geq (w+1)w^2$.

Note that C^*_i is a w-TA code for each step of the recursion, thus $L^*_i \leq O(M_i)$. This together with (4.2) and (4.1) gives $L_i \leq O(M_i)$.

So we have proved that $L_i \leq O(M_i)$ without assuming that $L_{i-1} \leq O(M_{i-1})$. Thus for the codes from Theorem 4.5.3 there exists a TTA with a linear in the size of the code runtime. This completes the proof of the theorem. ■

More generally, taking any starter code we obtain the following result:

Theorem 4.5.5 *For any integer $w \geq 2$ and any integer $q > w$ there exists an infinite class of $(n, M, q, w) - IPP$ codes with n is $O((w^2)^{log^*(M)}(log(M))$, which have a traitor tracing algorithm of linear runtime $O(M)$.*

It turns out that the method of list decoding discussed in Section 2.8 can be applied to traitor tracing algorithms, when the mentioned codes are used as TA-codes. This fact is discussed in Section 4.3. For instance, from Theorem 4.3.27 (case (i)) we see that TA codes based on Reed-Solomon codes will have traitor tracing algorithms of runtime $poly(\log M)$, where M is the size of the codes. This, in turn, implies that the method can be applied to our constructed IPP codes. Consequently, if $s = q$ is a prime power and the ingredients of the recursion are Reed-Solomon codes, then the IPP codes of Theorem 4.5.4 allow a traitor tracing algorithm which can run in $poly(\log M)$ time. We present this class in the following theorem:

Theorem 4.5.6 *For any integer $w \geq 2$ and any integer $q > w$ there exists an infinite class of $(n, M, q, w) - IPP$ codes with n is $O((w^2)^{log^*(M)}(log(M))$, which have a traitor tracing algorithm of runtime $poly(\log M)$.*

4.6 Construction of a New Class of w-IPP Codes Using PHF

In this section we use the new class of perfect hash families given by Theorem 3.8.15 to derive IPP codes in view of Theorem 4.6.1.

Using Theorem 4.3.10 and Theorem 3.8.15 we immediately obtain the following new class of IPP codes:

Theorem 4.6.1 *[82] Let $w \geq 2$ be any integer and q be any prime power such that $q \geq \binom{\lfloor (w+2)^2/4 \rfloor}{2}$. Then there exists an $(O((i+1)^{\lfloor (w+2)^2/4 \rfloor - 1}), q^{i+1}, q, w) - IPP$ code for any integer $i \geq 1$.*

Proof: By Theorem 3.8.15 there exists an $(O((i+1)^{\lfloor (w+2)^2/4 \rfloor - 1}), q^{i+1}, q, \lfloor (w+2)^2/4 \rfloor) - PHF$. The theorem then follows from Theorem 4.3.10. ■

It is worth noting that at each recursion step the size of the constructed code in Theorem 4.6.1 increases much slower than that in Theorem 4.5.1. Actually, Theorem 4.6.1 roughly states that $w - IPP$ codes of certain codeword length can be constructed for any given w and any given code size q^i. Thus, Theorem 4.6.1 gives an explicit construction of IPP codes for a very large set of parameter values.

4.7 On a Class of TA Codes

Staddon, Stinson, and Wei [63], ask the following question: Can we construct w-TA codes with $q < w^2$ and $b > q$?

Our aim is to give an answer to the Staddon-Stinson-Wei's problem. In Section 2.7 we have presented a new general construction method for q-ary codes with large Hamming distance. Using this method we are able to construct a large class of w-TA codes with $q < w^2$ and $b > q$, and thus obtain a positive answer to the problem.

4.7.1 Construction of w-TA Codes with $q < w^2$ and $b > q$

The following theorem shows that codes constructed in Theorem 2.7.8, in fact, provide a large class of w-TA codes with $q < w^2$ and $b > q$.

Theorem 4.7.1 *[71]*

Let q_0 and q_1 be prime powers such that $q_1 \geq q_0$.

(i) Suppose $\sqrt{q_0 q_1} + 1 < \lceil \sqrt{q_0 q_1 + q_1 + 1} \rceil$. Then for any integer n with

$$\sqrt{q_0 q_1} + 1 < \lceil \sqrt{n} \rceil \leq \lceil \sqrt{q_0 q_1 + q_1 + 1} \rceil$$

there exists an (n, b, q) w-TA code with $q < w^2$ and $b > q$, where

$$\begin{aligned} b &= q_0^2 q_1 \\ q &= q_0 q_1 \\ w &= \lceil \sqrt{n} \rceil - 1. \end{aligned}$$

(ii) For any integer $m \geq 2$ and for any integer n with

$$\sqrt{q_0 q_1^m} + 1 < \lceil \sqrt{n} \rceil \leq \lceil \sqrt{q_0 q_1^m + q_1^m + \cdots + q_1 + 1} \rceil$$

there exists an (n, b, q) w-TA code with $q < w^2$ and $b > q$, where

$$\begin{aligned} b &= q_0^2 q_1^m \\ q &= q_0 q_1^m \\ w &= \lceil \sqrt{n} \rceil - 1. \end{aligned}$$

Proof: First, recall that the parameters $(N, b, q; d)$ of a code $\mathcal{C}^*$ in Theorem 2.7.8 (ii) are $N = q_0 q_1^m + q_1^m + q_1^{m-1} + \cdots + q_1 + 1$, $b = q_0^2 q_1^m$, $q = q_0 q_1^m$, and $d = N - 1$, where $m \geq 1$ is an integer. We remark that if $\mathcal{C}^*$ is shortened, the resulting code with length $n \leq N$ always have minimum distance $d = n - 1$.

Let $(n, b, q; n-1)$ be the parameters of a shortened code $\mathcal{C}$ of $\mathcal{C}^*$ (the case $\mathcal{C} = \mathcal{C}^*$ is also included). So, $n \leq N$. Let $w = \lceil \sqrt{n} \rceil - 1$. By Theorem 4.3.13, $\mathcal{C}$ is a w-TA code. The condition $q < w^2$, i.e., $\sqrt{q} < w$, thus becomes $\sqrt{q} < \lceil \sqrt{n} \rceil - 1$, equivalently $\sqrt{q} + 1 < \lceil \sqrt{n} \rceil$. As $n \leq N$, we have $\sqrt{q} + 1 < \lceil \sqrt{n} \rceil \leq \lceil \sqrt{N} \rceil$. Now $q = q_0 q_1^m$, so if $m = 1$, we have the condition $\sqrt{q_0 q_1} + 1 < \lceil \sqrt{n} \rceil \leq \lceil \sqrt{q_0 q_1 + q_1 + 1} \rceil$. Thus (i) follows. If $m \geq 2$, we see that the condition $\sqrt{q} + 1 < \lceil \sqrt{N} \rceil$ is always satisfied. In fact, we only need to verify that $\sqrt{q} + 1 < \sqrt{N}$, i.e., $(\sqrt{q_0 q_1^m} + 1)^2 < q_0 q_1^m + q_1^m + q_1^{m-1} + \cdots + q_1 + 1$. Simplifying the last inequality yields $4 q_0 q_1^{m-2} < (q_1^{m-1} + \cdots + q_1 + 1)^2$, which is satisfied for all integers $q_1 \geq q_0 \geq 2$ and $m \geq 2$. Thus we have (ii). The proof is complete. ■

Remark: *In the proof of Theorem 4.7.1 above, we do not use the approximation $\sqrt{q} + 1 < \sqrt{N}$ to show $\sqrt{q} + 1 < \lceil \sqrt{N} \rceil$ for the case $m = 1$. If we used it, we would get an inequality $4q_0 < q_1$. And therefore, we would miss a large number of w-TA codes. In fact, the condition $\sqrt{q_0 q_1} + 1 < \lceil \sqrt{q_0 q_1 + q_1 + 1} \rceil$, as stated in the theorem, is much stronger.* ■

Example 4.7.2 Some small w-TA codes of Theorem 4.7.1 (i) are as follows: A $(10, 12, 6)$ 3-TA code corresponds to $q_0 = 2$ and $q_1 = 3$. This code is also displayed in Example 2.7.3. For $q_0 = 3$ and $q_1 = 4$ we have a (17, 36, 12) 4-TA code, and for $q_0 = 4$ and $q_1 = 5$ we have a (26, 80, 20) 5-TA code.

The discussions in Section 4.3 show that we cannot construct w-TA codes as application of Theorem 4.3.13 for fixes $q < w^2$ when $M \to \infty$. The known existence results on w-TA codes studied above require $q \geq w^2$. We have seen that w-IPP codes in general exist for any fixed $q > w$ with $n = \Theta(\log M)$. However, the existence of w-TA codes with $q < w^2$ when $M \to \infty$ remains open.

4.8 Summary

In this chapter we have investigated identifiable parent property codes. These codes are designed to be used in the schemes that protect copyrights of digital data against a colluding coalition of authorized users of the data. From the known sufficient conditions for the existence of IPP codes we have that for any fixed $q > w$ there exists an infinite class of IPP codes with $n = O(\log M)$.

We focus our attention on explicit constructions of IPP codes. We prove that the concatenation of two IPP codes gives an IPP code. This result together with some recursive techniques allows us to construct an infinite class of explicit IPP codes for any alphabet size which is larger than the coalition size that the code is able to handle. We observe that our constructed class has the best known asymptotic behavior of the parameters among explicit classes. We also study the complexity of the traitor tracing algorithms of these codes. We show the existence of a traitor tracing algorithm with a linear runtime in the size of the code.

It is shown in [66] that powerful list decoding techniques from coding theory can be applied to some classes of TA codes to give a traitor tracing algorithm with runtime polynomial in the length of the code. The classes of codes discussed in [66] are restricted with the large sizes of alphabet compared with the coalition size that the code can handle. We show that list decoding techniques can be applied also to some infinite subclasses of the IPP codes derived by our construction. Thus for any fixed $q > w$ we obtain an infinite class of explicit w-IPP codes with very good asymptotic behavior of parameters which has a traitor tracing algorithm with runtime growing polynomially with the code length.

The connections between IPP codes and other combinatorial structures such as hash families and error correcting codes have been established by several authors. These results yield several explicit construction for IPP codes. We also present two new classes of IPP codes based on these relationships.

First, using a new class of perfect hash families constructed in the Subsection 3.8.3 we derive an infinite class of IPP codes which covers a wide range of parameters.

Secondly, we give an affirmative answer to an open problem of Staddon, Stinson, and Wei about existence of TA codes with $q < w^2$ and $b > q$. In fact, using a new class of q-ary codes with large Hamming distance, constructed in Section 2.7, and using the connections between TA codes and error correcting codes, we obtain a new class of w- TA codes with desired parameters. The existence of w-TA codes remains open in the case when b and $n \to \infty$ for fixed q and w such that $w < q \leq w^2$.

Chapter 5

Covering Arrays

This chapter concerns t-covering arrays which are known also as a qualitatively t-independent family of vectors or t-surjective arrays. Covering arrays have undergone an intensive survey by many researchers due to their numerous applications in computer science such as software or circuit testing, switching networks, data compression problem, and also several mathematical applications such as difference matrices, search theory and truth functions.

The application of covering arrays to software system testing is discussed in many papers e.g., see [101]. One of the approaches to reduce costs for testing a software system is to use combinatorial designs to generate an efficient test set. Software system faults are often caused by interactions among components. The goal of a software developer is to test all combinations of potential interactions with not very large number of tests. For the system where most errors occur because of interactions of its maximum t components, a test plan can be designed using t-covering arrays. As example in [101] covering arrays have been used to design efficient test plans for a telephone switch system and a network performance monitoring system.

Other applications related to covering arrays are authentication, block ciphers, intersecting codes, obvious transfer, pseudorandomness, span programs, universal hashing, resilient functions and zero-knowledge. [108]

We focus on explicit construction methods for t-covering arrays. Firstly, using the relationships between perfect hash families and covering arrays one can construct infinite families of t-covering arrays with very good asymptotic behavior. We obtain an upper bound on the covering array number, which is shown to be better than the known probabilistic upper bound.

Secondly, inspired from a result of Roux and also from a recent result of Chateauneuf and Kreher for 3-covering arrays, several direct constructions for t-covering arrays are presented, which can be viewed as generalizations of their results for t-covering arrays, $t \geq 4$. These constructions yield good upper bounds

on the covering array number when the size of arrays is small. Our main results in this chapter are contained in Theorem 5.3.2, Theorem 5.3.7, Theorem 5.3.8, Theorem 5.4.1, Theorem 5.4.8 and Theorem 5.4.9 (see also [109]).

An introduction to covering arrays is given in Section 5.1. Some known results are presented in Section 5.2. New infinite families of covering arrays are derived in Section 5.3 using recursive techniques. A comparison of parameters of the constructed arrays with a known probabilistic bound is provided. Section 5.4 includes several new constructions for t-covering arrays with $t \geq 4$. Finally, in Section 5.5 we give a summary of new results.

5.1 Introduction

Definition 5.1.1 *A t-covering array, denoted $\mathsf{CA}(N; t, k, v)$, is a $k \times N$-array with entries from a set of $v \geq 2$ symbols such that each $t \times N$-subarray contains each ordered t-tuple of symbols at least once as a column.*

Example 5.1.2 *Example of a $\mathsf{CA}(33; 3, 6, 3)$*

```
012211200220110102200211221001012
122102002101102022012112010012012
221010021211020220101120200121012
210120212010201201021202101210012
101222120002011010222021112100012
000001111122222111110000022222012
```

Let $\mathsf{CAN}(t, k, v)$ denote the minimum number N such that a $\mathsf{CA}(N; t, k, v)$ exists, i.e.,

$$\mathsf{CAN}(t, k, v) = \min\{N : \exists\ \mathsf{CA}(N; t, k, v)\}.$$

Then $\mathsf{CAN}(t, k, v)$ is called the *covering array number*.

A $\mathsf{CA}(N; t, k, v)$ is minimal if $N = \mathsf{CAN}(t, k, v)$. It is shown in [105] that the covering array given in the Example 5.1.2 is minimal, $\mathsf{CAN}(3, 6, 3) = 33$.

Covering arrays can be viewed as a generalization of orthogonal arrays. In fact, if we require that each $t \times N$-subarray contains each ordered t-tuple of symbols in exactly λ times as a column, then we have an *t-orthogonal array*, denoted $\mathsf{OA}_\lambda(t, k, v)$ (see Section 2.5 for details). In this case we have $N = \lambda v^t$. Thus,

an $\mathsf{OA}_\lambda(t,k,v)$ is a $\mathsf{CA}(\lambda v^t;t,k,v)$. In particular, if there is an $\mathsf{OA}_1(t,k,v)$, then $\mathsf{CAN}(t,k,v) = v^t$. For instance, an $\mathsf{OA}_1(t,t+1,v)$ exists for all t and v, see e.g., [23] ; also, for any prime power q and any $t < q$, $\mathsf{OA}_1(t,q+1,q)$ exists [6]. Therefore, $\mathsf{CAN}(t,t+1,v) = v^t$ and $\mathsf{CAN}(t,q+1,q) = q^t$.

A main problem of covering arrays is to minimize N for given values $t,\ k,\ v$, or equivalently to maximize k for given values $t,\ v,\ N$. The case $t = 2$ has been studied by several authors, see for instance [87], [88], [89], [99], [102]. For the case $v = 2$ and $t = 2$ the problem has been completely solved by Katona [88], Réni [87], and Kleiman and Spencer [89] using the Sperner lemma and Erdös-Ko-Rado theorem [85].

For $v > 2$ and $t = 2$ the problem is much harder. Gargano, Körner and Vaccaro [97, 99] have studied the asymptotic behavior for this case, showing that

$$\lim_{k\to\infty} \frac{\mathsf{CAN}(2,k,v)}{\log_2 k} = \frac{v}{2}. \tag{5.1}$$

However, this result is non constructive and is of theoretical nature. For $v > 2$ there are no explicit constructions known which achieve bound (5.1), and not much is known about the covering array number for small k. Sloane in [100] gives a survey on known bounds for the case $q = 3$ and $t = 2$. Improved tables of upper bounds on covering array number for $q \leq 7$ and $k \leq 50$ have been generated in [104] and [107]. Several lower bounds are derived in [102]. Some construction techniques are given in [96, 97, 103, 104, 107].

The case $t = 3$ can be found in [92, 94, 100, 105, 106, 108]. Probabilistic upper bounds on the number of columns N for t-covering arrays are given in [86]. George Sherwood has a database for covering arrays constructed using various computer search techniques [110]. However, very little are known for t-covering arrays with $t \geq 4$.

This chapter is concerned with t-covering arrays for an arbitrary value t. Our interest is in constructing t-covering arrays using combinatorial techniques and in establishing bounds on the covering array numbers $\mathsf{CAN}(t,k,v)$. In particular, we present constructions of good classes of t-covering arrays using recursive methods and perfect hash families. We then show combinatorial methods of how to construct new covering arrays from other covering arrays and thus obtain several bounds for t-covering arrays in the spirit of [94], [108].

5.2 Preliminaries

The following basic facts on $\mathsf{CAN}(t,k,v)$ can be found in [108]. Let A be a $\mathsf{CA}(N;t,k,v)$ with entries $a_{ij} \in V = \{0,\ldots,v-1\}$.

Symbol-fusing. If a symbol x is replaced with any symbol in $V \setminus \{x\}$, wherever x occurs in the array A, then the resulting array is a $\mathsf{CA}(N; t, k, v-1)$. Thus

$$\mathsf{CAN}(t, k, v-1) \leq \mathsf{CAN}(t, k, v).$$

Row-deleting. If any row of A is deleted, then the remaining rows form a $\mathsf{CA}(N; t, k-1, v)$. Hence

$$\mathsf{CAN}(t, k-1, v) \leq \mathsf{CAN}(t, k, v).$$

Derived array. Note that if $x \in V$ appears M times in row i of A, then $M \geq v^{t-1}$. Removing all columns of A not having x on row i and then deleting row i form a $\mathsf{CA}(M; t-1, k-1, v)$. Therefore

$$\mathsf{CAN}(t, k, v) \geq v.\mathsf{CAN}(t-1, k-1, v).$$

Product. Let B be a $\mathsf{CA}(M; t, k, w)$ with entries $b_{ij} \in W = \{0, \ldots, w-1\}$. From the $k \times N$ array C_l with entries $(a_{ij}, b_{il}) \in V \times W$ for all $i = 1, \ldots, k$ and $j = 1, \ldots, N$. Then $[\mathsf{C}_1, \ldots, \mathsf{C}_M]$ is a $\mathsf{CA}(NM; t, k, vw)$ on symbol set $V \times W$. Therefore,

$$\mathsf{CAN}(t, k, vw) \leq \mathsf{CAN}(t, k, v)\mathsf{CAN}(t, k, w).$$

Squaring k[105]

If $(\binom{t}{2}!, k) = 1$, and there is a $\mathsf{CA}(N; t, k, v)$, then for $j \geq 0$

$$\mathsf{CAN}(t, k^{2^j}, v) \leq N(\binom{t}{2} + 1)^j. \tag{5.2}$$

We prove a simple lemma which shows rough lower and upper bounds for $\mathsf{CAN}(t, k, v)$ for certain values of k.

Lemma 5.2.1 *For any $v \geq 2$, $t \geq 2$ we have*

$$v^t \leq \mathsf{CAN}(t, k, v) \leq 2^t.v^t - 1,$$

where $k \leq 2^n$ and n is the smallest integer such that $v \leq 2^n$.

Proof: An obvious lower bound is

$$v^t \leq \mathsf{CAN}(t, k, v),$$

and this bound is reached if $v = q$ is a prime power and $k \leq q + 1$ because an orthogonal array $\mathsf{OA}_1(t, q + 1, q)$ exists [6]. If v is not a prime power, then $2^{n-1} < v < 2^n$ for a certain integer n. Now take $\mathsf{CA}(N; t, 2^n, 2^n) = \mathsf{OA}_1(t, 2^n, 2^n)$. Then $N = 2^{2nt}$. Using the symbol-fusing method one gets a $\mathsf{CA}(N; t, 2^n, v)$. Since $N = 2^t.2^{(n-1)t} < 2^t.v^t$, we have $N \leq (2^t v^t - 1)$. ■

Non trivial lower bounds can be derived using the derived array technique. We discus nontrivial lower bounds useful for large values of k in the next section.

In [94], a Ph.D. dissertation, Roux shows the following theorem, see also [100].

Theorem 5.2.2 (Roux [94])

$$\mathsf{CAN}(3, 2k, 2) \leq \mathsf{CAN}(3, k, 2) + \mathsf{CAN}(2, k, 2).$$

Thus, Roux's theorem gives an upper bound for 3-covering array for $v = 2$.

Recently, Chateauneuf and Kreher [108] generalized Roux's theorem for any $v \geq 2$.

Theorem 5.2.3 (Chateauneuf and Kreher [108])

$$\mathsf{CAN}(3, 2k, v) \leq \mathsf{CAN}(3, k, v) + (v - 1)\mathsf{CAN}(2, k, v).$$

5.3 Recursive Construction of CA Using PHF

Perfect hash families discussed in the Chapter 3 can be used to derive bounds on the covering array number. We describe a relationship between covering arrays and perfect hash families. Let $\mathsf{A} = (a_{i,j})$ denote the $k \times N$-matrix of a $\mathsf{CA}(N; t, k, v)$. For any two column j_1 and j_2 of A, define

$$I(j_1, j_2) = |\{i : \ a_{i,j_1} = a_{i,j_2}\}|$$

and

$$I(\mathsf{A}) = \max\{I(j_1, j_2) : \ j_1 \neq j_2\}.$$

Theorem 5.3.1 *Suppose there exists a* $\mathsf{CA}(N; t, k, v)$.

(i) Then there exists a $(N', k, v, t) - PHF$ *when* $t \leq v$, $N' = N - v \times (v - 1) \times (v - 2) \times \cdots \times (v - t + 1) + 1$

(ii) If $k/I(\mathsf{A}) > \binom{w}{2}$, *then there is a* $(k, N, v, w) - PHF$.

Proof: Let A denote the $k \times N$-array presenting the $\mathsf{CA}(N; t, k, v)$.

(i) It is obvious that any t rows of A contain at least $\frac{v!}{(v-t)!}$ columns with all different symbols if $t \leq v$. Thus if we delete any $\frac{v!}{(v-t)!} - 1$ columns of A we obtain an array A' with $N' = N - v \times (v-1) \times (v-2) \times \cdots \times (v-t+1) + 1$ columns. Any t rows of the array A' include at least one column with all different elements. Thus A' is a $(N', k, v, t) - PHF$ if $t \leq v$.

(ii) Taking the columns of A as codewords, we have a $(k, N, v; d = k - I(\mathsf{A}))$ code. Then apply Theorem 3.7.1. ■

A construction of covering arrays using perfect hash families is as follows:

Theorem 5.3.2 *Suppose there exists a* $(s, k, m, t) - PHF$ *and a* $\mathsf{CA}(N; t, m, v)$. *Then there is a* $\mathsf{CA}(sN; t, k, v)$.

Let $n(k, v, t) = \min\{n : \exists\, (n, k, v, t) - PHF\}$. Some characterizations of covering array number we give in next lemmas.

Lemma 5.3.3 *For any* $v \geq t$ *we have*

$$n(k, v, t) + \frac{v!}{(v-t)!} - 1 \leq \mathsf{CAN}(t, k, v) \leq (2^t v^t - 1) n(k, v, t).$$

Proof: The left hand side of this inequality follows from Theorem 5.3.1 (case (i)), and the right hand side follows from Theorem 5.3.2 and Lemma 5.2.1. ■

Lemma 5.3.4 *For any* $v \geq t$, v *is a prime power we have*

$$n(k, v, t) + \frac{v!}{(v-t)!} - 1 \leq \mathsf{CAN}(t, k, v) \leq v^t n(k, v, t).$$

Proof: The left hand side of this inequality follows from Theorem 5.3.1 (case (i)), and the right hand side follows from Theorem 5.3.2 and the fact that $\mathsf{CAN}(t, v, v) = v^t$ when v is a prime power. ■

Lemma 5.3.5 *For any* $v < t$, *we have*

$$n(k, v, v) + v! - 1 \leq \mathsf{CAN}(t, k, v) \leq v^t n(k, t, t).$$

Proof: The left hand side of this inequality follows from Theorem 5.3.1 (case (i)) and the fact that $\mathsf{CAN}(v,k,v) \leq \mathsf{CAN}(t,k,v)$ when $v < t$, and the right hand side follows from Theorem 5.3.2 and the fact that $\mathsf{CAN}(t,t,v) = v^t$. ■

In particular these lemmas show that nontrivial lower bounds on the covering array numbers can be obtained from the lower bounds of minimal number of hash functions of a perfect hash family having corresponding parameters. It would be interesting to derive tighter nontrivial lower bounds on covering array number using similar techniques as for perfect hash families. A survey on the lower bound for $n(k,v,t)$ is provided in Sections 3.4, 3.3, 3.5. From the lemmas given above it can be also observed that an asymptotic bound of covering array number when $k \to \infty$ with fixed v and t will differ at most by a constant factor from an asymptotic bound of minimal number of hash functions of perfect hash family having corresponding parameters.

We now use Corollary 3.7.3 and Theorem 5.3.2 to construct an infinite class of t-covering arrays with good asymptotic behavior.

Theorem 5.3.6 *[109] Suppose there exists a* $\mathsf{CA}(N_0;t,q^{s_0},v)$*, where* q *is a prime power and* $q^{s_0} > \frac{t(t-1)}{2}$*. Then there exists a* $\mathsf{CA}(N_0R_i;t,q^{s_i},v)$ *for all* $i \geq 0$*, where* $R_0 = 1$*, and*

$$\begin{aligned} R_i &= q^{s_{i-1}}R_{i-1}, \\ s_i &= s_{i-1}\lceil \frac{q^{s_{i-1}}}{\binom{t}{2}} \rceil \end{aligned}$$

for all $i \geq 1$.

Proof: We proceed by induction on i. For $i = 0$, the assertion is correct. Now assume $i \geq 1$. From Corollary 3.7.3 we have a $(q^{s_{i-1}}, q^{s_i}, q^{s_{i-1}}, t) - PHF$.

By induction, there exists a $\mathsf{CA}(N_0R_{i-1};t,q^{s_{i-1}},v)$. Now applying Theorem 5.3.2 yields a $\mathsf{CA}(N_0R_i;t,q^{s_i},v)$. The proof is complete. ■

Let $N_i = N_0R_i$ and $k_i = q^{s_i}$. Then, by a similar argumentation as in Section 3.8 ([46] pp.196-197) it can be proved that

$$N_i \leq \frac{N_0t^{2i_0}}{s_0 \log q}(t^2)^{\log^*(k_i)}(\log k_i)$$

for all $i > i_0$.

For any given values of k_0, v and t we can always construct a $\mathsf{CA}(N_0;t,k_0,v)$ for some N_0. Therefore, we have the following theorem.

Theorem 5.3.7 *[109] For any positive integers v and t, there is an infinite explicit constructive family of covering array $\mathsf{CA}(N; t, k, v)$ such that*

$$N = O((t^2)^{\log^*(k)}(\log k).$$

Theorem 5.3.2 becomes powerful when algebraic-geometric (AG) codes are used. The idea is to derive good classes of perfect hash families from AG codes by Theorem 3.7.1, and then apply Theorem 5.3.2.

Now, combining Theorem 5.3.2 and Theorem 3.7.7 we can prove the following result:

Theorem 5.3.8 *[109] For every given integers $t, v \geq 2$, and for any integer $n \geq 1$, there exists a covering array $\mathsf{CA}(N; t, k, v)$, where*

$N = N_0.(q-1)q^{n+1}$, N_0 is a constant,
$k = q^{2\lfloor uq^{n+1} \rfloor}$, q is a prime power such that $q \geq \frac{t(t-1)(u+1)}{2} + 1$,
and u is a real number with $1 \leq u \leq q-2$.

Moreover, we have $N = O(\log k)$.

Proof: Let $t, v \geq 2$ be given integers. Let q be the smallest prime power such that $t = \lfloor \frac{1}{2}(1 + \sqrt{1 + \frac{8}{u+1}(q-1)}) \rfloor$, with $1 \leq u \leq q-2$, as shown in Theorem 3.7.7. A simple observation shows that we can always construct a $\mathsf{CA}(N_0; t, q^2, v)$ explicitly for a certain value N_0. Applying Theorem 5.3.2 and Theorem 3.7.7 yields the covering arrays with parameters as claimed. ■

It should be noted (see also Section 2.6) that the first low-complexity algorithm for constructing "one-point" AG codes on G-S curves has a runtime upper-bounded by $(N \log_q N)^3$, where N the length of the code and the complexity is measured in terms of multiplications and divisions over the finite field $\mathbb{F}_{q^2}$ [28]. The complexity of constructing t-covering arrays in Theorem 5.3.8 is, therefore, polynomial in N. The covering arrays in Theorem 3.8.13, however, can be viewed as an explicitly constructed family.

The following probabilistic upper bound for $\mathsf{CAN}(t, k, v)$ is due to Godbole *et al* [86].

Theorem 5.3.9 (Godbole, Skipper, Sunley [86])

$$\mathsf{CAN}(t, k, v) \leq \frac{(t-1)\log k}{\log\left(\frac{v^t}{v^t - 1}\right)}\{1 + o(1)\},$$

as $k \to \infty$.

It turns out that the covering arrays in Theorem 5.3.8 yield much better results compared to Godbole-Skipper-Sunley bound.

To see it we consider e.g., the case with a square prime power $v = q^2$. For any given $t \geq 2$ and any prime power q satisfying the condition of Theorem 5.3.8 choose a real number $1 \leq u \leq q-2$ such that $\frac{(q-1)}{(u+1)} = \binom{t}{2}$. By taking a $\mathsf{CA}(N_0; t, q^2, q^2)$ with $N_0 = q^{2t}$, Theorem 5.3.8 gives a $\mathsf{CA}(N; t, k, q^2)$ with $N = q^{2t}(q-1)q^{n+1}$ and $k = q^{2\lfloor uq^{n+1} \rfloor}$. Thus $N \approx \frac{q^{2t}(q-1)}{2u \ln q} \ln k$. For these t and k, the Godbole-Skipper-Sunley bound gives $\mathsf{CAN}(t, k, v) \leq \frac{(t-1)}{\ln \frac{q^{2t}}{q^{2t}-1}} \ln k\{1 + o(1)\}$. Let $\alpha = \frac{q^{2t}(q-1)}{2u \ln q}$ and $\beta = \frac{(t-1)}{\ln \frac{q^{2t}}{q^{2t}-1}}$. Then

$$\begin{aligned} \frac{\alpha}{\beta} &= \frac{(q-1)q^{2t} \ln \frac{q^{2t}}{q^{2t}-1}}{2u(t-1) \ln q} \\ &\approx \frac{(u+1)t}{4u \ln q}, \quad \text{or} \\ &\leq \frac{t}{4 \ln q} \end{aligned}$$

by taking into account $q^{2t} \ln \frac{q^{2t}}{q^{2t}-1} \approx 1$. Thus $\frac{\alpha}{\beta} < 1$ for $q \geq 3$. This shows that sizes of arrays from Theorem 5.3.8 with $v = q^2$ are better than Godbole-Skipper-Sunley bounds.

As examples we consider several values of v.

For $v = 3^2$, $t = 2$ $u = 1$ we have $\alpha = 73.729$ and $\beta = 80.498$

For $v = 7^2$, $t = 3$ and $u = 1$ we have $\alpha = 181378.878$ and $\beta = 235296.999$.

For $v = 13^2$ and $t = 4$ and $u = 1$ we have $\alpha = 1908179711.915$ and $\beta = 2447192161.523$.

Since $\frac{\alpha}{\beta} \to 0$ as $q \to \infty$, the Godbole-Skipper-Sunley bound becomes weak. For instance, if $v = 2^{32}$, $t = 2$, $u = 2^{16} - 2$ we have $\alpha \approx 8,3 * 10^{17}$ whereas $\beta = 25 * 10^{18}$. Thus, α is about 30 times smaller than β.

5.4 Constructions of Roux's Type for t-CA

The constructions from the last section provide asymptotically good classes of covering arrays, when $k \to \infty$ with fixed v, t. In this section we focus on construction techniques which can be used to improve the results for small values of k.

With Theorem 5.2.2 Roux shows an interesting bound for binary 3-covering array, i.e., $v = 2$. This bound is recently generalized by Chateauneuf and Kre-

her to any $v \geq 2$, as presented in Theorem 5.2.3. The idea is to construct a $\mathsf{CA}(3, 2k, v)$ using a $\mathsf{CA}(3, k, v)$ and a $\mathsf{CA}(2, k, v)$.

Remark: *We want to make a remark that Theorem 4.6. of Chateauneuf and Kreher [108] p.231 is incorrect. Theorem 4.6. [108] states that one obtains*

$$\lim_{k\to\infty} \frac{\mathsf{CAN}(3, k, v)}{\log k} = \binom{v}{2}$$

from

$$\mathsf{CAN}(3, 2k, v) \leq \mathsf{CAN}(3, k, v) + (v-1)\mathsf{CAN}(2, k, v), \qquad (*)$$

and

$$\lim_{k\to\infty} \frac{\mathsf{CAN}(2, k, v)}{\log_2 k} = \frac{v}{2} \qquad (**)$$

In fact, it can be shown from $(*)$ *and* $(**)$ *that*

$$\lim_{k\to\infty} \frac{\mathsf{CAN}(3, k, v)}{\log k} = \infty.$$

■

In this section, in the spirit of Roux, Chateauneuf and Kreher, we discuss several constructions of $\mathsf{CA}(t, 2k, v)$ using $\mathsf{CA}(s, k, v)$ for $s \leq t$.

5.4.1 4-Covering Arrays

The structure of covering array becomes more involved when its strength grows. This might be one of the reasons that very little is known about t-covering arrays for $t \geq 4$ in comparison with 2-, 3- covering arrays. In this section, we present a recursive construction of 4-covering arrays based on 2-, 3- covering arrays.

Let D be a $\mathsf{CA}(N_1; 2, v, v)$ with entries $d_{j,i} \in V = \{1, \ldots, v\}$. Let $\mathcal{F}_D = \{f_1, \ldots, f_{N_1}\}$ be a set of mappings derived from D as follows: For each $i = 1, \ldots, N_1$ define

$$f_i : V \longrightarrow V$$

by

$$f_i(j) = d_{j,i}.$$

Thus f_i maps the vector $(1, \ldots, v)^T$ to the $i-$th column of D, i.e., $f_i(j) = d_{j,i}$.

Remark: *The family $\mathcal{F}_D$ has the following property. For any given two pairs (x, y) and (z, w) with $x, y, z, w \in V$ and $x \neq y$, there is at least an $f_i \in \mathcal{F}_D$ such that $f_i(x) = z$ and $f_i(y) = w$. This is because* D *is a* $\mathsf{CA}(N_0; 2, v, v)$. ■

In the following theorem we give a bound for 4-covering arrays by means of a direct construction.

Theorem 5.4.1 *[109] For any $v \geq 2$ we have*

$$\mathsf{CAN}(4, 2k, v) \leq \mathsf{CAN}(4, k, v) + (v-1)\mathsf{CAN}(3, k, v) + 2\mathsf{CAN}(2, v, v)\mathsf{CAN}(2, k, v).$$

Proof: Let A be a $\mathsf{CA}(N_4; 4, k, v)$, B be a $\mathsf{CA}(N_3; 3, k, v)$, C be a $\mathsf{CA}(N_2; 2, k, v)$, and D be a $\mathsf{CA}(N_1; 2, v, v)$, all on the symbol set $V = \{1, 2, \ldots, v\}$. Let $\mathcal{F}_D = \{f_1, f_2, \ldots, f_{N_1}\}$ be the set of mappings derived from D as defined above. Finally, let $\pi = (1, 2, \ldots, v)$ be a cyclic permutation on the symbol set V. Define

$$\mathsf{E}_1 = \begin{array}{|c|}\hline \mathsf{A} \\ \hline \mathsf{A} \\ \hline \end{array}$$

$$\mathsf{E}_2 = \begin{array}{|c|c|c|c|}\hline \mathsf{B} & \mathsf{B} & \ldots & \mathsf{B} \\ \hline \mathsf{B}^{\pi^1} & \mathsf{B}^{\pi^2} & \ldots & \mathsf{B}^{\pi^{v-1}} \\ \hline \end{array}$$

$$\mathsf{E}_3 = \begin{array}{|c|c|c|c|}\hline \mathsf{C} & \mathsf{C} & \ldots & \mathsf{C} \\ \hline \mathsf{C}^{f_1} & \mathsf{C}^{f_2} & \ldots & \mathsf{C}^{f_{N_1}} \\ \hline \end{array}$$

$$\mathsf{E}_4 = \begin{array}{|c|c|c|c|}\hline \mathsf{C}^{f_1} & \mathsf{C}^{f_2} & \ldots & \mathsf{C}^{f_{N_1}} \\ \hline \mathsf{C} & \mathsf{C} & \ldots & \mathsf{C} \\ \hline \end{array}$$

where B^{π^i} and C^{f_j} are the arrays obtained by applying π^i and f_j to the symbols of B and C, respectively.

Construct an array E as follows:

$$\mathsf{E} = \boxed{\mathsf{E}_1 \mid \mathsf{E}_2 \mid \mathsf{E}_3 \mid \mathsf{E}_4}$$

E is therefore an $2k \times N$-array, where $N = N_4 + (v-1)N_3 + 2N_2N_1$.

Consider 4 rows $r_1,\ r_2,\ r_3,\ r_4$ of E.

1. If $r_1,\ r_2,\ r_3,\ r_4$ include 4 distinct rows of A, then all quadruples occur on these rows among the columns of E_1.
2. If $r_1 < r_2 < r_3 \leq k < r_4 = r_1 + k$ or $r_1 \leq k < r_2 = r_1 + k < r_3 < r_4$, then all quadruples of the form $(x, y, w, x)^T$ for any x, y, w occur on these rows among the columns of E_1 and quadruples $(x, y, w, z)^T$ with $x \neq z$ occur in E_2.
3. If $r_1 < r_2 \leq k < r_3 = r_1 + k < r_4$, then we have two subcases.
   - 3.1. $r_4 \neq r_2 + k$. Quadruples of the form $(x, y, x, z)^T$ for any x, y, z occur among the columns of E_1. Let $r_4' = r_4 - k$. Then $r_1, r_2, r_4' \leq k < r_3 = r_1 + k$. For any quadruple of the form $(x, y, x', z)^T$ with $x' \neq x$, then $x' = x^{\pi^i}$ for some i. Hence there is a column in E_2 containing x in row r_1, y in row r_2, $z^{(\pi^i)^{-1}}$ in row r_4', and $x' = x^{\pi^i}$ in row r_3. Therefore, $(x, y, x', z)^T$ appears in that column on the rows r_1, r_2, r_3, r_4.
   - 3.2. $r_4 = r_2 + k$. Quadruples of the form $(x, y, w, z)^T$ with $x \neq y$ for any w, z occur on the rows r_1, r_2, r_3, r_4 among the columns of E_3, because there exists an f_i such that $x^{f_i} = w$ and $y^{f_i} = z$; similarly quadruples $(x, y, w, z)^T$ with $w \neq z$ is covered by E_4; quadruples of the form $(x, x, y, y)^T$ for every x and y occur among the columns of E_3 and E_4.

Therefore, E is a covering array $\mathsf{CA}(N; 4, 2k, v)$ with $N = N_4 + (v-1)N_3 + 2N_2N_1$, as required.

■

From the proof of the theorem it can be observed that shorter covering arrays can be constructed in several cases by choosing the arrays $\mathsf{A}, \mathsf{C}, \mathsf{D}$ more carefully. These cases are listed in the following lemma.

Lemma 5.4.2 *The construction in Theorem 5.4.1 still works if any of arrays* A, C *and* D *is chosen as follows:*

1. *C is a $k \times N_2$-array with entries from a set of v symbols such that each $2 \times N$-subarray contains each ordered 2-tuple of not equal symbols at least once as a column.*

2. *In the binary alphabet case, D is a 2×2 array where both rows are equal to $\{0, 1\}$.*

3. *In the case $k < 4$, A is the same as the array B.*

From Lemma 5.4.2 (case 2) and Theorem 5.4.1 we obtain the following corollary.

Corollary 5.4.3

$$\mathsf{CAN}(4, 2k, 2) \leq \mathsf{CAN}(4, k, 2) + \mathsf{CAN}(3, k, 2) + 4\mathsf{CAN}(2, k, 2).$$

Theorem 5.4.1 together with Lemma 5.4.2 gives the following example.

Example 5.4.4 $\mathsf{CA}(28; 4, 6, 2)$

```
00011101000111010001001111 00
00101011001010110000101110 10
01000111010001110000011110 01
00011101111000101000001001 11
00101011110101000100000101 11
01000111101110000010000011 11
```

It should be noted that the parameters of the covering array in this example matches with the parameter of the corresponding array generated by the Constrained Array Text System [98] when an "expand" computer search program has been used, see [110].

Example 5.4.5 $\mathsf{CA}(40; 4, 8, 2)$

We take the arrays A, B, C and D as follows:

$\mathsf{A} =$

```
0001110100011101
0010101100101011
0100011101000111
0001110111100010
```

A *is a 4-covering array.*

$$\mathsf{B} = \begin{bmatrix} 00011101 \\ 00101011 \\ 01000111 \\ 01110001 \end{bmatrix}$$

B *is a 3-covering array.*

$$\mathsf{C} = \begin{bmatrix} 1000 \\ 0100 \\ 0010 \\ 0001 \end{bmatrix}$$

C *is an array defined in Lemma 5.4.2 (case 1).*

$$\mathsf{D} = \begin{bmatrix} 01 \\ 01 \end{bmatrix}$$

D *is the array from Lemma 5.4.2 (case 2).*

Now applying Theorem 5.4.1 to A, B, C *and* D *we obtain the array* E *as follows:*

0001110100011101	00011101	1000	1000	0000	1111
0010101100101011	00101011	0100	0100	0000	1111
0100011101000111	01000111	0010	0010	0000	1111
0001110111100010	01110001	0001	0001	0000	1111
0001110100011101	11100010	0000	1111	1000	1000
0010101100101011	11010100	0000	1111	0100	0100
0100011101000111	10111000	0000	1111	0010	0010
0001110111100010	10001110	0000	1111	0001	0001

A covering array with this parameters is also constructed in [110]. However, there are no construction methods given in general case. For the binary case, the idea in [110] is to use a computer search program to check all combinations of rows of a specified $2^j \times 2^{j+1}$ array, where $j > 2$ is an integer. The goal of this search is to find covering arrays for some $3 < k < 2^j$. In the last step, a computer program is used to remove redundant rows. This technique yields a $\mathsf{CA}(30; 4, 6, 2)$ covering array while we obtain a $\mathsf{CA}(28; 4, 6, 2)$ as shown in Example 5.4.4. It should be noted that a $\mathsf{CA}(31; 4, 8, 2)$ has been found by a computer search. [110]

If $v = q$ is a prime power, then a $\mathsf{CA}(q^2; 2, q, q)$ exists. Hence, the bound in Theorem 5.4.1 can be strengthened and we obtain:

Corollary 5.4.6 *For any prime power $q \geq 2$ we have*

$$\mathsf{CAN}(4, 2k, q) \leq \mathsf{CAN}(4, k, q) + (q-1)\mathsf{CAN}(3, k, q) + 2q^2\mathsf{CAN}(2, k, q).$$

5.4.2 5-Covering Arrays

We present a construction for 5-covering arrays similar to the contraction for 4-covering arrays described above. We prove the following theorem:

Theorem 5.4.7 *[109] For any $v \geq 3$ we have*

$$\begin{aligned}\mathsf{CAN}(5, 2k, v) \leq \mathsf{CAN}(5, k, v) + (v-1)\mathsf{CAN}(4, k, v)\\ +[6v(v-1) + 2\mathsf{CAN}(2, v, v)]\mathsf{CAN}(3, k, v).\end{aligned}$$

Proof: Let A be a $\mathsf{CA}(N_5; 5, k, v)$, B be a $\mathsf{CA}(N_4; 4, k, v)$, C be a $\mathsf{CA}(N_3; 3, k, v)$, and D be a $\mathsf{CA}(N_1; 2, v, v)$, all on the symbol set $V = \{1, 2, \ldots, v\}$. Again let $\mathcal{F}_D = \{f_1, f_2, \ldots, f_{N_1}\}$ be the set of mappings defined from a $\mathsf{CA}(N_1; 2, v, v)$ as in Section 4. Also, let $\pi = (1, 2, \ldots, v)$ be a cyclic permutation on the symbol set V.

We define three families of mappings from V into V as follows:

(i). Let $\mathcal{G} = \{g_{a,b} : V \longrightarrow V : a, b \in V, a \neq b\}$, where

$$g_{a,b}(x) = \begin{cases} a & \text{if } x = a \\ b & \text{if } x \neq a \end{cases}$$

(ii). Let $\bar{\mathcal{G}} = \{\bar{g}_{a,b} : V \longrightarrow V : a, b \in V, a \neq b\}$, where

$$\bar{g}_{a,b}(x) = \begin{cases} a & \text{if } x = b \\ b & \text{if } x \neq b \end{cases}$$

(iii). Let $\mathcal{H} = \{h_{a,b} : V \longrightarrow V : a, b \in V, a \neq b\}$, where

$$h_{a,b}(x) = \begin{cases} a & \text{if } x \neq a \text{ or } x \neq b \\ b & \text{if } x = a \text{ or } x = b \end{cases}$$

Define

$$\mathsf{E}_1 = \begin{array}{|c|}\hline \mathsf{A} \\ \hline \mathsf{A} \\ \hline \end{array}$$

$$\mathsf{E}_2 = \begin{array}{|c|c|c|c|}\hline \mathsf{B} & \mathsf{B} & \ldots & \mathsf{B} \\ \hline \mathsf{B}^{\pi^1} & \mathsf{B}^{\pi^2} & \ldots & \mathsf{B}^{\pi^{v-1}} \\ \hline \end{array}$$

$$\mathsf{E}_3 = \begin{array}{|c|c|c|c||c|c|c|c|}\hline \mathsf{C} & \mathsf{C} & \ldots & \mathsf{C} & \mathsf{C}^{f_1} & \mathsf{C}^{f_2} & \ldots & \mathsf{C}^{f_{N_1}} \\ \hline \mathsf{C}^{f_1} & \mathsf{C}^{f_2} & \ldots & \mathsf{C}^{f_{N_1}} & \mathsf{C} & \mathsf{C} & \ldots & \mathsf{C} \\ \hline \end{array}$$

$$\mathsf{E}_4 = \begin{array}{|c|c|c|c||c|c|c|c|}\hline \mathsf{C} & \mathsf{C} & \ldots & \mathsf{C} & \mathsf{C}^{g_{1,2}} & \mathsf{C}^{g_{1,3}} & \ldots & \mathsf{C}^{g_{v,v-1}} \\ \hline \mathsf{C}^{g_{1,2}} & \mathsf{C}^{g_{1,3}} & \ldots & \mathsf{C}^{g_{v,v-1}} & \mathsf{C} & \mathsf{C} & \ldots & \mathsf{C} \\ \hline \end{array}$$

$$\mathsf{E}_5 = \begin{array}{|c|c|c|c||c|c|c|c|}\hline \mathsf{C} & \mathsf{C} & \ldots & \mathsf{C} & \mathsf{C}^{\bar{g}_{1,2}} & \mathsf{C}^{\bar{g}_{1,3}} & \ldots & \mathsf{C}^{\bar{g}_{v,v-1}} \\ \hline \mathsf{C}^{\bar{g}_{1,2}} & \mathsf{C}^{\bar{g}_{1,3}} & \ldots & \mathsf{C}^{\bar{g}_{v,v-1}} & \mathsf{C} & \mathsf{C} & \ldots & \mathsf{C} \\ \hline \end{array}$$

$$\mathsf{E}_6 = \begin{array}{|c|c|c|c||c|c|c|c|}\hline \mathsf{C} & \mathsf{C} & \ldots & \mathsf{C} & \mathsf{C}^{h_{1,2}} & \mathsf{C}^{h_{1,3}} & \ldots & \mathsf{C}^{h_{v,v-1}} \\ \hline \mathsf{C}^{h_{1,2}} & \mathsf{C}^{h_{1,3}} & \ldots & \mathsf{C}^{h_{v,v-1}} & \mathsf{C} & \mathsf{C} & \ldots & \mathsf{C} \\ \hline \end{array}$$

Construct an array E as follows:

$$\mathsf{E} = \begin{array}{|c|c|c|c|c|c|}\hline \mathsf{E}_1 & \mathsf{E}_2 & \mathsf{E}_3 & \mathsf{E}_4 & \mathsf{E}_5 & \mathsf{E}_6 \\ \hline \end{array}$$

Let r_1, r_2, r_3, r_4, r_5 be 5 rows of E. Because of the symmetry of E we need to consider the following cases.

1. If $r_1,\ r_2,\ r_3,\ r_4,\ r_5$ satisfy $r_i \neq r_j + k$, $i \neq j$ and $i, j = 1, 2, 3, 4, 5$, then all 5-tuples occur on these rows among the columns of E_1.
2. If $r_1 < r_2 < r_3 < r_4 \leq k < r_5 = r_1 + k$, then 5-tuples of the form $(a, b, c, d, a)^T$ occur on these rows among the columns of E_1, and all 5-tuples $(a, b, c, d, a')^T$ with $a' \neq a$ appear in the columns of E_2.

3. Assume $r_1 < r_2 < r_3 \leq k < r_4 < r_5$, $r_4 = r_1 + k$ and $r_5 \neq r_i + k$ for all $i = 1, 2, 3$. Consider a 5-tuple $X = (a, b, c, a', e)^T$. If $a = a'$, then X is covered by E_1. Now assume $a \neq a'$. As B is a 4-covering array, all $(v-1)$ quadruples $(a, b, c, e_1)^T, \ldots, (a, b, c, e_{v-1})^T$ with $e \neq e_i$, appear on the rows $r_1,\ r_2,\ r_3,\ r_5 - k$ among the columns. Thus, for each π^i, there is a e_j such that $\pi^i(e_j) = e$. Further, $\pi^i(a) = a_i$ with $a \neq a_i$. It follows that all 5-tuples $(a, b, c, a_1, c), (a, b, c, a_2, c), \ldots,$ (a, b, c, a_{v-1}, c), where and $a_i \neq a_j$ for $i \neq j$, appear in the columns corresponding to rows $r_1,\ r_2,\ r_3,\ r_4,\ r_5$ in E_2.

4. Assume $r_1 < r_2 < r_3 \leq k < r_4 < r_5$, $r_4 = r_1 + k$ and $r_5 = r_2 + k$. We need to consider different types of 5-tuples.

   (i) A 5-tuple of the form $(a, b, x, a, b)^T$ for any a, b, x is covered by E_1.

   (ii) A 5-tuple of the form $(a, a, x, b, b)^T$ for any a, b, x is covered by E_2.

   (iii) A 5-tuple of the form $(a, b, x, c, d)^T$ for any a, b, x, c, d and $a \neq b$ is covered by E_3. This is because C is a 3-covering array, there is at least one column of C containing the triple $(a, b, x)^T$ in the rows $r_1,\ r_2,\ r_3$ and there is and f_i such that $f_i(a) = c$ and $f_i(b) = d$.

   From now on we can assume $c \neq d$.

   (iv) Consider a 5-tuple of the form $(a, a, x, c, d)^T$ for any a, x, c, d and $c \neq d$. We have the following subcases:

   (α) $x \neq a, c, d$. There is a column j of C containing the triple $(x, c, d)^T$ in the rows $r_3 + k$, $r_1 + k$, $r_2 + k$ of E_4. Therefore the column j of the block

   $$\begin{array}{|c|}\hline \mathsf{C}^{g_{x,a}} \\ \hline \mathsf{C} \\ \hline \end{array}$$

   contains the 5-tuple $(a, a, x, c, d)^T$ with $x \neq a, c, d$ in the rows $r_1,\ r_2,\ r_3,\ r_1 + k,\ r_2 + k$, because $g_{x,a}(x) = x$, $g_{x,a}(c) = a$, and $g_{x,a}(d) = a$.

   (β) $x = a$. As C is a 3-covering array, there is column j containing the triple $(c, d, c)^T$ in the row $r_1 + k,\ r_2 + k,\ r_3 + k$. Also there is a mapping f_i, $1 \leq i \leq N_1$, such that $f_i(c) = a$ and $f_i(d) = a$, by Remark 5.4.1. Therefore the column j of the block

   $$\begin{array}{|c|}\hline \mathsf{C}^{f_i} \\ \hline \mathsf{C} \\ \hline \end{array}$$

in E_3 contains the 5-tuple $(a, a, a, c, d)^T$ in the rows $r_1,\ r_2,\ r_3,\ r_1 + k,\ r_2 + k$.

(γ) $x \neq a$ and $x = c$. Again there is a column j of C containing the triple $(c, d, a)^T$ in the row $r_1+k,\ r_2+k,\ r_3+k$. Therefore the column j of the block

in E_5 contains the 5-tuple $(a, a, c, c, d)^T$ in the rows $r_1,\ r_2,\ r_3,\ r_1 + k,\ r_2 + k$.

(δ) $x = a = c$ (i.e. $a \neq d$). The 5-tuple $(a, a, a, a, d)^T$ is covered by a column of the block

with $f_i(a) = a$ and $f_i(d) = a$ in part E_3.

(θ) $x = c$ and $a = d$. Consider a column j of C containing the triple $(c, a, b)^T$ with $b \neq c, a$ in the rows $r_1 + k,\ r_2 + k,\ r_3 + k$. The 5-tuple $(a, a, c, c, a)^T$ is contained in a column j corresponding to the rows $r_1,\ r_2,\ r_3,\ r_1 + k,\ r_2 + k$ of the block

of E_6. This is because $h_{c,a}(b) = c$, $h_{c,a}(c) = a$ and $h_{c,a}(a) = a$.

Hence E is a 5-covering array. The proof is complete by using $|\mathcal{G}| = |\bar{\mathcal{G}}| = |\mathcal{H}| = v(v - 1)$. ■

If $v = q$ is a prime power, then $N_1 = v^2$ by Lemma 5.2.1. Therefore we have

Corollary 5.4.8 *For any prime power $q \geq 3$ we have*

$$\mathsf{CAN}(5, 2k, q) \leq \mathsf{CAN}(5, k, q) + (q - 1)\mathsf{CAN}(4, k, q) + (8q^2 - 6q)CAN(3, k, q).$$

5.4.3 t-Covering Arrays for $t \geq 4$

Theorem 5.4.9 *[109] For any integers $t \geq 4$ and $v \geq 2$ we have*

$$\mathsf{CAN}(t, 2k, v) \leq \mathsf{CAN}(t, k, v) + (v-1)\mathsf{CAN}(t-1, k, v) + \sum_{i=2}^{t-2} 2\mathsf{CAN}(i, k, v)\mathsf{CAN}(t-i, k, v).$$

Proof: Let $A_t, A_{t-1}, \ldots, A_2$ be

$$\mathsf{CA}(n_t; t, k, v), \mathsf{CA}(n_{t-1}; t-1, k, v), \ldots, \mathsf{CA}(n_2; 2, k, v),$$

respectively.

Let $\mathsf{B}_i^{n_j}$ be the $k \times n_i.n_j$ array obtained from A_i by repeating each column n_j times, where $i, j = t-2, \ldots, 2$ and $i + j = t$.

Let $C_j^{n_i}$ be the $k \times n_i.n_j$ array obtained by concatenating n_i copies of A_j, where $i, j = t-2, \ldots, 2$ and $i + j = t$.

Define

$$\mathsf{E}_t = \begin{array}{|c|}\hline \mathsf{A}_t \\ \hline \mathsf{A}_t \\ \hline \end{array}$$

$$\mathsf{E}_{t-1} = \begin{array}{|c|c|c|c|}\hline \mathsf{A}_{t-1} & \mathsf{A}_{t-1} & \ldots & \mathsf{A}_{t-1} \\ \hline \mathsf{A}_{t-1}^{\pi} & \mathsf{A}_{t-1}^{\pi^2} & \ldots & \mathsf{A}_{t-1}^{\pi^{v-1}} \\ \hline \end{array}$$

For $i = t-2, \ldots, 2$, define

$$\mathsf{E}_i = \begin{array}{|c|c|}\hline \mathsf{B}_i^{n_{t-i}} & \mathsf{B}_{t-i}^{n_i} \\ \hline \mathsf{C}_{t-i}^{n_i} & \mathsf{C}_i^{n_{t-i}} \\ \hline \end{array}$$

Construct an array E as follows

$$\mathsf{E} = \begin{array}{|c|c|c|c|c|}\hline \mathsf{E}_t & \mathsf{E}_{t-1} & \mathsf{E}_{t-2} & \ldots & \mathsf{E}_2 \\ \hline \end{array}$$

Let $r_1, \ldots, r_t$ be t rows of E. Because of the symmetry of E we need to consider the following cases.

1. If $r_1, \ldots, r_t$ include t distinct rows of A_t, then all t-tuples occur on these rows among the columns of E_1.
2. If $r_1 < \ldots < r_{t-1} \leq k < r_t = r_1 + k$, then t-tuples of the form $(a_1, \ldots, a_{t-1}, a_1)^T$ is covered by E_1, and all t-tuples $(a_1, \ldots, a_{t-1}, a')^T$ with $a' \neq a_1$ appear in the columns of E_2.
3. For the remaining cases we can assume $r_1 < \ldots < r_i \leq k$ and $k < r_{i+1} < \ldots < r_t$, where $i = t-2, t-3, \ldots, \lceil \frac{t}{2} \rceil$. Then for each $i = t-2, t-3, \ldots, \lceil \frac{t}{2} \rceil$ and for any t-tuple $(a_1, a_2, \ldots, a_t)^T$ of symbols there is a column in E_i containing this t-tuple in the rows $r_1 < \ldots < r_i \leq k < r_{i+1} < \ldots < r_t$.

The proof is complete. ■

For $t = 4, 5$ and large k the construction in Theorem 5.4.9 yields a weaker upper bound on covering array number than the previous constructions for 4-, 5-covering arrays discussed in this section. But the generalization of the idea of previous constructions seems to become more involved and difficult to describe with growing t. For 5-covering arrays with not very large k, however, the construction given in Theorem 5.4.9 provides a tighter upper bound. We demonstrate an application of the Theorem 5.4.9 by an example.

Example 5.4.10 *Let* A_5 *be a* $\mathsf{CA}(4^5; 5, 5, 4)$. A_4 *be a* $\mathsf{CA}(4^4; 4, 5, 4)$, A_3 *be a* $\mathsf{CA}(4^3; 3, 4, 4)$ *and a* A_2 *be* $\mathsf{CA}(16; 2, 4, 4)$. *Then applying Theorem 5.4.9 we obtain a* $\mathsf{CA}(5888; 5, 10, 4)$.

Note that from the Corollary 5.4.8 we have a $\mathsf{CA}(8448; 5, 10, 4)$ and from the Equation 5.2 we obtain a $\mathsf{CA}(11264; 5, 10, 4)$.

The probabilistic bound (Theorem 5.3.9) gives a $\mathsf{CAN}(5, 10, 4) \leq 9426$.

5.5 Summary

In this chapter we have studied covering arrays, which are generalizations of orthogonal arrays. These combinatorial structures have undergone an intensive survey during last few years due to their numerous practical applications. One of main problems is to construct covering arrays with a small number of columns. We have developed a number of explicit constructions for covering arrays. Firstly, we show some asymptotically good classes of covering arrays based on perfect hash families. The techniques are recursive and make use Reed-Solomon or "one-point" AG codes to construct infinite families of t-covering arrays. We obtain

an upper bound on covering array number, which is shown to be better than the known probabilistic upper bound.

Secondly, we give some constructions of t-covering arrays with $t \geq 4$. The structure of covering arrays becomes more involved when its strength grows. This might be one of the reasons that very little is known about t-covering arrays for $t \geq 4$ in comparison with 2-, 3- covering arrays. Inspired from a result of Roux and also from a recent result of Chateauneuf and Kreher for 3-covering arrays, several constructions were provided for the arrays of strength $t \geq 4$ which make use covering arrays with lower strength and recursive techniques. These constructions provide better covering arrays than the other known in the literature constructions for certain parameter ranges.

Appendix A

Notation

$a^{\underline{b}}$	$\prod_{i=0}^{b-1}(a-i)$.
(A,B)	*join* of arrays A and B [page 16]
$\mathcal{A}(\mathcal{C})$	$M \times n$ array on q symbols corresponding to an $(n, M, q; d)$ code $\mathcal{C}$ [page 13].
$\mathcal{C}$	a q-ary code. [page 5].
$C^\top$	transpose matrix of C.
$\lvert\mathcal{C}\rvert$	size of $\mathcal{C}$.
C	matrix representation of the code $\mathcal{C}$ [page 5].
$\mathsf{CA}(N; t, k, v)$	*covering array* with k constrains (or of degree k, or with k rows), of v levels (or of degree v, alphabet size), strength t and N columns [page 80].
$\mathsf{CAN}(t, k, v)$	for fixed t, k and v the minimum number N such that a $\mathsf{CA}(N; t, k, v)$ exists.
$\mathbf{desc}(\mathcal{C}_0)$	the set of descendants of $\mathcal{C}_0$ [page 55].
$\mathbf{desc}_w(\mathcal{C})$	w-descendant code of $\mathcal{C}$ [page 55].
$d(x, y)$	Hamming distance between the codewords x and y [page 6].
F_q	finite field of q elements [page 8].
$RS_{n,k}(\alpha, v)$	Reed-Solomon code of length n and dimension k on α and v.
$\mathcal{H}$	hash family [page 21].
$I(x, y)$	$\{i : x_i = y_i\}$, where $x = \{x_1, x_2, \cdots x_n\}$, $y = \{y_1, y_2, \cdots y_n\}$ and $i = 1, 2, \cdots, n$.
$I(\mathsf{A})$	maximal intersection of any two columns of array A [page 83].
$\log^*$	see definition [page 45].

$L(mP)$	$\mathbb{F}_{q^2}$-vector space consisting of all functions defined on the curve such that the only pole of any $f \in L(mP)$ is P and the pole order is at most m [page 12].
$m_j(a)$	frequency of a on the j^{th} column of $\mathcal{A}(\mathcal{C})$ [page 13].
$M(n, m, w)$	for fixed n, m and w the maximal value of M for which an $(n, M, m, w) - PHF$ exists.
$n(M, m, w)$	for fixed n, m and w the minimal number of hash functions n for which an $(n, M, m, w) - PHF$ exists.
$(n, M, q, w_1, w_2) - SSHF$	strong separating (n, M, q) hash family of strengths w_1 and w_2 [page 58].
$(n, M, m) - HF$	family of n hash functions $h : A \longrightarrow B$, where $\|A\| = M \geq \|B\| = m$.
$(n, M, m, w) - PHF$	perfect (n, M, m) hash family of strength w [page 21].
(n, b, q, w)-TA	traceability (n, b, q) code of strength w [page 56]
$[n, k, q]$	linear (n, M, q) code of dimension k, $k = \log M$ [page 6].
$[n, k, q; d]$	$[n, M, q]$ linear code with minimum distance d [page 6].
(n, M, q) code	code of length n, size M over an alphabet of size q [page 5].
$(n, M, q) - \mathcal{RS}$	Reed Solomon (n, M, q) code.
$(n, M, q, t, u) - PAHF$	partially (n, M, q) hashing family of strength t, and u [page 58].
$(n, M, q, w_1, w_2) - SHF$	separating (n, M, q) hash family of strengths w_1 and w_2 [page 58].
$(n, M, q, w) - IPP$	identifiable parent property (n, M, q) code of strength w [page 55].
$(n, M, q; d)$	(n, M, q) code with minimum distance d.
$\mathsf{OA}_\lambda(t, n, v)$	*orthogonal array* with n constrains (or of degree n, or with n rows), of v levels (or of degree v, alphabet size), strength t and index λ [page 9].
$\mathsf{OA}(t, n, v)$	$\mathsf{OA}_\lambda(t, n, v)$ where $\lambda = 1$.
$\mathcal{P} = \{P_1, \ldots, P_n, P\}$	$n + 1$ distinct $\mathbb{F}_{q^2}$-rational points [page 12].
$\preceq$	for any functions $f(x)$ and $g(x)$, $f(x) \preceq g(x)$ denotes that $f(x) \leq (1 + 0(1))g(x)$, where 0(1) tends to zero when x tends to infinite.

Appendix B

Acronyms

AG	algebraic geometry code
CA	covering array
CAN	covering array number
ERS	extended Reed-Solomon code
G-S	Garcia-Stichtenoth
RS	Reed-Solomon code
HF	hash family
IPP	identifiable parent property code
MDS	maximum distance separable
MOLS	mutually orthogonal Latin squares
OA	orthogonal arrays
PAHF	partially hashing family
PHF	perfect hash function
R-S	Reed-Solomon code
SHF	separating hash family
SSHF	strong separating hashing family
TA	traceability code
TTA	traitor tracing algorithm

Bibliography

[1] N. J. A. SLOANE, Error-Correcting Codes and Cryptography, *The Mathematical Gardner, D. A. Klarner (editor), Prindle, Weber and Schmidt, Boston, MA,* (1981), 346–382.

[2] C.J. COLBOURN AND P.C. VAN OORSCHOT, Applications of combinatorial designs in computer science, *ACM Computing Surveys* **21** (1989), 223–250.

[3] G. D. COHEN, Applications of coding theory to communication combinatorial problems, *Discr. Math.* **83** (1990), 237–248.

[4] D. R. STINSON, Combinatorial designs and cryptography, *Surveys in Combinatorics, 1993, K. Walker, ed., Cambridge University Press*, (1993), 257–287.

[5] C. J. COLBOURN, J. H. DINITZ AND D. R. STINSON, Applications of combinatorial designs to communications, cryptography, and networking, *Surveys in Combinatorics (J.D. Lamb and D.A. Preece, eds.), Cambridge University Press*, (1999) 37–100.

———————————-

[6] K. A. BUSH, A generalization of a theorem due to MacNeish, *Ann. Math. Stat.,* **23** (1952), 293–295.

[7] P. ELIAS, List decoding for noisy channels, *Wescon Convention Record, Institute of Radio Engineers (now IEEE)***2**, (1957), 94–104.

[8] J. M. WOZENCRAFT, List decoding, *Quarterly Progress report, Research laboratory of Electronics, MIT***48** , (1958), 90–95.

[9] W. W. PETERSON AND E. J. JR. WELDON, Error correcting codes *MIT Press, Cambridge, MA, 1972.*

[10] J. H. VAN LINT Introduction to Coding Theory, *Springer-Verlag*, (1982).

[11] Y.I. MANIN AND G. VLĂDUT, Linear codes and modular curves. *J. Soviet. Math.,* **30** , (1985), 2611–2643.

[12] M. A. TSFASMAN AND S. G. VLĂDUŢ Algebraic-Geometry codes, *Kluwer Academic Publishers*, Norwell, MA, (1991).

[13] F. J. MACWILLIAMS AND N. J. A SLOANE, The Theory of Error-Correcting Codes, *North-Holland* (1992).

[14] H. STICHTENOTH Algebraic function fields and codes, *Berlin, Germany: Springer-Verlag*, (1993).

[15] A. GARCIA, H. STICHTENOTH, A tower of Artin-Schreier extensions of function fields attaining the Drinfeld-Vlădut bound, *Invent. Math.* **121** (1995), 211–222.

[16] J. L. BLANCHARD, A construction for orthogonal arrays with strength $t \geq 3$, *Discrete Math* **137**,(1995), 35–44.

[17] C. J. COLBOURN AND J. H. DINITZ, *CRC Handbook of Combinatorial Designs*, CRC Press, Inc., 1996.

[18] J. BIERBRAUER AND C. J. COLBOURN, Orthogonal arrays of strength more than two *The CRC Handbook of Combinatorial Designs, C. J. Colbourn and J. H. Dinitz, eds.*, CRC Press, (1996).

[19] A. GARCIA, H. STICHTENOTH, On asymptotic behavior of some towers of function fields over finite fields, *J. Number Theory* **61** (1996), 248–273.

[20] M. SUDAN, Decoding of Reed-Solomon codes beyond the error-correction bound. *Journal of Complexity* **13** (1997), 180–193.

[21] M. SUDAN, Decoding of Reed-Solomon codes beyond the error-correction diameter. *Proc. 35th Annual Allerton Conference on Communication, Control and Computing* (1997), 215–224.

[22] T. HØHOLDT, R. PELLIKAAN, J. H VAN LINT, Algebraic geometry codes , *Handbook of Coding Theory, V. S. Pless, W. C. Huffamm and R.A. Brualdi, EDds. Amsterdam, The Netherlands: Elsevier*(1998).

[23] A. S. HEDAYAT, N. J. A. SLOANE AND JOHN STUFKEN, *Orthogonal Arrays: Theory and Applications, Springer-Verlag, New York,* (1999).

[24] V. GURUSWAMI AND M. SUDAN, Improved decoding of Reed-Solomon and algebraic-geometry codes, *IEEE Trans. Inform. Theory* **45** (1999), 1757–1767.

[25] S. B. WICKER (EDITOR), V. K. BHARGAVA (EDITOR), Reed-Solomon Codes and Their Applications, *Wiley-IEEE Press* (September 1999).

[26] J. WALKER, Codes and Curves, *American Mathematical Society, Student Mathematical Library,* **7**, (2000).

[27] V. GURUSWAMI AND M. SUDAN, List decoding algorithms for certain concatenated codes, *Proc. 32nd ACM Symposium on Theory of Computing (STOC 2000)*, 181–190.

[28] K. W. SHUM, I. ALESHNIKOV, P. V. KUMAR, H. STICHTENOTH, AND V. DEOLALIKAR A Low-Complexity Algorithm for the Construction of Algebraic-Geometric Codes Better Than the Gilbert-Varshamov Bound, *IEEE Transactions on Information Theory* **47** (2001), 2225–2241.

[29] G. T. BOGDANOVA, A. E. BROUWER, S. N. KAPRALOV, AND P. R. J. ÖSTERGRÅRD, Error-Correcting Codes over an Alphabet of Four Elements, *Designs, Codes and Cryptography* **23** (2001), 333–342.

———————————-

[30] K. MEHLHORN, On the program size of perfect and universal hash functions. *Proceedings of the 23rd IEEE Symposium on Foundations of Computer Science (FOCS'82)*, (1982), 170–175.

[31] K. MEHLHORN, *Data Structures and Algorithms,* **1**. *Sorting and Searching, Springer-Verlag, Berlin* (1984).

[32] M. FREDMAN AND J. KOMLÒS, On the size of separating systems and perfect hash functions, *SIAM J. Algebraic and Discrete Methods,* **5** (1984), 61–68.

[33] M. FREDMAN, J. KOMLÒS AND E. SZEMEREDI Storing a sparse table with O(1) worst case access time, *Journal of the Association for Computing Machinery,* **31** (1984), 538–544.

[34] N. ALON, Explicit construction of exponential sized families of k-independent sets, *Discrete Math.* **58** (1986), 191–193.

[35] J. KÖRNER AND K. MARTON, New bounds for perfect hashing via information theory, *Euro. J. Comb.inatorics,* **9** (1988), 523-530.

[36] E. F. BRICKELL, A Problem in Broadcast Encryption, *Presented at the Fifth Vermont Summer Workshop on Combinatorics and Graph Theory*, June, (1991).

[37] JÜRGEN BIERBRAUER *Perfect hashing, Material* (1995).

[38] M. ATICI, S. S. MAGLIVERAS, D. R. STINSON AND W. D. WEI, Some recursive constructions for perfect hash families, *Journal of Combinatorial Designs* **4** (1996), 353–363.

[39] MUSTAFA ATICI, Hash Families: Recursive Constructions and Applications to Cryptography *Phd thesis Lincoln, Nebraska* (1996).

[40] Z. J. CZECH, G. HAVAS, AND B. S. MAJEWSKI, Perfect hashing, *Theor. Comp. Sci.* **182** (1997), 1–143.

[41] SOSINA MARTIROSYAN Lower and Upper Bounds on Maximal Cardinality of a K-separating Set, *Proceedings of the 2nd INTAS Meeting on Information Theory and Combinatorics, Essen, Germany, April 7-9*, (1997), ISBN 90-74249-15-9, 27–36.

[42] SAMVEL MARTIROSYAN AND SOSINA MARTIROSYAN, New Upper Bound on the Cardinality of a K-separated Set or Perfect Hash Family and a Near Optimal Construction for It. *Proceedings of GMIT International conference, Yerevan, Armenia, September*, (1997), 25–29.

[43] S. R. BLACKBURN AND P. R. WILD Optimal linear perfect hash families, *J. Comb. Theory - Series A*, **83** (1998), 223–250.

[44] R. BLACKBURN, Combinatorics and Threshold Cryptography, *Combinatorial Designs and their Applications (CRC Research Notes in Mathematics* **403**, CRC Press, London, (1999), 49–70.

[45] SAMVEL MARTIROSYAN AND SOSINA MARTIROSYAN, New Upper Bound on the Cardinality of a K-separated Set or Perfect Hash Family and a Near Optimal Construction for it, *Transactions of IPIA of NAN RA & YSU "Mathematical Problems of Computer Science"*, **XXI** (2000), 104–115.

[46] D. R. STINSON AND R. WEI AND L. ZHU, New constructions for perfect hash families and related structures using combinatorial designs and codes, *J. Combin. Designs* **8** (2000), 189–200.

[47] S. R. BLACKBURN , Perfect hash families: probabilistic methods and explicit constructions, *J. Comb. Theory - Series A* **92** (2000), 54–60.

[48] HUAXIONG WANG , CHAOPING XING, Explicit constructions of perfect hash families from algebraic curves over finite fields, *Journal of Combinatorial Theory Series A*, **93**, (2001), 112–124.

[49] S. R. BLACKBURN, Perfect Hash Families with Few Functions, *preprint*.

---

[50] D. BONEH AND M. FRANKLIN, An efficient public keys traitor tracing schemes, *Advances in Cryptology - Crypto'94 (Lecture Notes in Computer Science), Springer-Verlag* **839** (1994), 257–270.

[51] B. CHOR, A. FIAT AND M. NAOR, Tracing traitors, *Advances in Cryptology - Crypto'94 (Lecture Notes in Computer Science)*, Springer-Verlag, **839** (1994), 257–270.

[52] R. BLACKBURN, M. BURMESTER, Y. DESMEDT AND P.R. WILD, Efficient Multiplicative Sharing Schemes, *Advances in Cryptology – EUROCRYPT '96, Lecture Notes in Computer Science , U. Maurer (Ed.) Springer, Berlin*, **1070**, (1996), 107–118.

[53] D. BONEH AND J. SHAW, Collusion-secure fingerprinting for digital data, *IEEE Trans. Inform. Theory* **44** (1998), 1897–1905.

[54] D. R. STINSON AND R. WEI, Combinatorial properties and constructions of traceability schemes and frameproof codes, *SIAM J. Discrete Math.* **11** (1998), 41–53.

[55] H. D. L. HOLLMANN, J. H. VAN LINT, J. P. LINNARTZ AND L. M. G. M. TOLHUIZEN, On codes with identifiable parent property, *J. Comb. Theory A*, **82** (1998), 121–133.

[56] D. R STINSON AND TRAN VAN TRUNG, Some new results on key distribution patterns and broadcast encryption, *Designs, Codes and Cryptography* **14** (1998), 261-279.

[57] E. GAFNI, J. STADDON, AND Y. L. YIN, Efficient methods for integrating traceability and broadcast encryption, *Advances in Cryptology–Crypto'99 (Lecture Notes in Computer Science)*, Springer-Verlag, **1666** (1999), 372–387.

[58] A. FIAT AND T. TASSA, Dynamic traitor tracing, *Advances in Cryptology Crypto'99 (Lecture Notes in Computer Science)*, Springer-Verlag, **1666** (1999), 354–371.

[59] R. KUMAR, S. RAJAGOPALAN, AND A. SAHAI, Coding constructions for blacklisting problems without computational assumptions, *Advances in Cryptology–Crypto'99 (Lecture Notes in Computer Science)*, Springer-Verlag, **1666** (1999), 609–623.

[60] B. CHOR, A. FIAT, M. NAOR, AND B. PINKAS, Tracing traitors, *IEEE Trans. Inform. Theory* **46** (2000), 480–491.

[61] K. KUROSAWA, T. YOSHIDA, AND Y. DESMEDT, Inherently large traceability and asymmetric schemes with arbiter, *Proc. IEEE Int. Symp. Inform. Theory (ISIT'2000)* Sorrento, Italy, June (2000).

[62] D. R. STINSON, TRAN VAN TRUNG AND R. WEI, Secure frameproof codes, key distribution patterns, group testing algorithms and related structures, *J. Statist. Planning Inference* **86** (2000), 595–617.

[63] J. N. STADDON, D. R. STINSON AND R. WEI, Combinatorial properties of frameproof and traceability codes, *IEEE Trans. Inform. Theory* **47** (2001), 1042–1049.

[64] 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.,* **14** (2001), 423–431.

[65] R. SAFAVI-NAINI AND YEJING WANG, New results on frameproof codes and traceability schemes, *IEEE Trans. Inform. Theory* **47** (2001), 3029–3033.

[66] A. SILVERBERG, J. N. STADDON, AND J. L. WALKER, Efficient Traitor Tracing Algorithms using List Decoding, *ASIACRYPT 2001, Lect. Notes Comput. Sci.* **2248** (2001), 175–192.

[67] P. SARKAR, D. R. STINSON, Frameproof and IPP codes, *Preprint*.

[68] N. ALON AND E. FISCHER AND M. SZEGEDY, Parent-identifying codes, *Journal of Combinatorial Theory Series A,* **95**, (2001), 349–359.

[69] A. FIAT AND T. TASSA, Dynamic traitor tracing, *J. Cryptology* **14** (2001), 211–223.

[70] N. ALON, G. COHEN, M. KRIVELEVICH, S. LITSYN, Generalized hashing and applications to digital fingerprinting, *manuscript.6*.

[71] TRAN VAN TRUNG AND SOSINA MARTIROSYAN, On a Class of Traceability Codes, *Designs, Codes and Cryptography*, to appear.

[72] TRAN VAN TRUNG AND SOSINA MARTIROSYAN, Constructions for efficient IPP Codes, *preprint (2002)* .

[73] A. BARG AND G. KABATIANSKY, A class of i.p.p. codes with efficient identification , *DIMACS Report-36*, (2002).

[74] Y. YEMANE, Codes with the k-identifable parent property, *PhD Thesis, Royal Holloway, University of London*, (2002).

[75] S. BLACKBURN, An upper bound on the size of a code with the k-identifiable parent property, *J. Combin. Theory Ser. A, Vol 102 (2003), pp. 179-185*.

[76] N. ALON, G. COHEN, M. KRIVELEVICH AND S. LITSYN, Generalized hashing and parent-identifying codes, *submitted*.

[77] N. ALON AND U. STAV, New bounds on parent-identifying codes: the case of multiple parents, *submitted*.

[78] M. FERNÁNDEZ, M. SORIANO, Decoding Codes with the Identifiable Parent Property, *The Seventh IEEE Symposium on Computers and Communications ISCC 2002 Taormina (Italy)*, (2002).

[79] M. FERNÁNDEZ, M. SORIANO , Efficient identification of traitors in fingerprinted multimedia contents, *Accepted to Workshop on Trust and Privacy in Digital Business, Aix en Provence, (France),* (2002).

[80] M. FERNÁNDEZ, M. SORIANO, Soft-Decision Decoding of Traceability Codes, *Accepted to IEEE International Conference on Multimedia and Expo (ICME2002), Lausanne, (Switzerland),* (2002).

[81] A. BARDECK Kombinatorische Untersuchung von IPP-Codes einschliesslich einer Charakterisierung von 4- und 5-IPP-Codes, *Diplomarbeit, Univerisität GH Essen* (2002).

[82] TRAN VAN TRUNG AND SOSINA MARTIROSYAN, New Constructions for IPP codes, *Designs, Codes and Cryptography*, to appear.

[83] TRAN VAN TRUNG AND SOSINA MARTIROSYAN, New Constructions for IPP codes, *2003 Proceedings IEEE International Symposium on Information Theory, Pacifico YOKOHAMA, Yokohama, JAPAN June 29- July 4,* (2003), 255.

———————————-

[84] C. R. RAO, Factorial experiments derivable from combinatorial arrangements of arrays, *J. Royal Statist. Soc.* **9** (1947), 218-139.

[85] P. ERDÖS, C. KO, R. RADO, Intersection theorems for systems of finite sets, *Quart. J. Math. Oxford Ser. 2* **12** (1961), 313–318.

[86] A. P. GODBOLE, D. E. SKIPPER, AND R. A. SUNLEY, t-covering arrays: upper bounds and Poisson approximations, *Combinatorics, Probability and Computing,* **5** (1966), 105–117.

[87] A. RÉNYI, *Foundation of probability*, Wiley, New York, 1971.

[88] G. O. H. KATONA, Two applications (for search theory and truth functions) of Spencer type theorems, *Periodica Math. Hung.,* **3** (1973), 19–26.

[89] D. J. KLEITMAN, J. SPENCER, Families of k-independent sets, *Discr. Math.,* **6** (1973), 255–262.

[90] S. POLJAK, A. PULTR, V. RODL, On qualitatively independent partitions and related problems, *Discr. Appl. Math.,* **6**, (1983), 193–205.

[91] D. T. TANG, L. S. WOO, Exhaustive Test Pattern Generation with Constant Weight Vectors, *IEEE Transactions on Computers* **32(12)**(1983), 1145–1150.

[92] P. BUSSCHBACH, Constructive methods to solve the problems of s-surjectivity, conflict resolution, coding in defective memories, *Report 84D005, École Nationale Supér. Télécomm., Paris,* (1984).

[93] P. ERDÖS, P. FRANKL AND Z. FÜREDI, Families of finite sets in which no set is covered by the union of r other, *Israel Journal of Mathematics* **51** (1985), 75–89.

[94] G. ROUX, k-propriétés dans des tableaux de n colonnes; cas particulier de la k-surjectivité et de la k-permutivité, *Ph.D. Dissertation, University of Paris 6, March* (1987).

[95] G. SEROUSSI AND N. BSHOUTI, Vector sets for exhaustive testing of logic circuits, *IEEE Transactions on Information Theory*, **34**, (1988), 513-522 .

[96] S. POLJAK, Z. TUZA, On the maximum number of qualitatively independent partitions, *J. Comb. Theor, Ser. A* **51** (1989), 11–116.

[97] L. GARGANO, J. KÖRNER, AND U. VACCAR Qualitative Independence and Sperner Problems for Directed Graphs, *J. Combinatorial Theory, Series A*, **61** (1992), 173–192.

[98] G. SHERWOOD, Constrained Array Test System (CATS), *AT&T Bell Laboratories, Murray Hill, NJ, User's Manual* (1992).

[99] L. GARGANO, J. KÖRNER, AND U. VACCARO Sperner capacities, *Graphs and Combinatorics*, **9** (1993), 31–46.

[100] N. J. A. SLOANE, Covering arrays and intersection codes, *J. Combin Designs*, **1** (1993), 51–63.

[101] D. M. COHEN, S. R. DALAL, M. L. FREDMAN AND G. C. PATTON, The AETG system: an approach to testing software based on combinatorial design, *IEEE Trans. Software Engineering*, **23** (1997), 437–444.

[102] B. STEVENS, L. MOURA, AND E. MENDELSOHN, Lower bounds for transversal covers, *Designs, Codes and Cryptography* **15** (1998) 279–299.

[103] D. M. COHEN AND M. L. FREDMAN, New techniques for designing qualitatively independent sets, *J. Combinat. Designs***6**, (1998), 411–416.

[104] B. STEVENS AND E. MENDELSOHN, New recursive methods for transversal covers, *J. Combin. Des.,* **7(3)** (1999) 185–203.

[105] M. CHATEAUNEUF, C. J. COLBOURN, AND D. L. KREHER, Covering arrays of strength 3, *Designs, Codes and Cryptography* **16** (1999), 235–242.

[106] M. CHATEAUNEUF, *Covering arrays*, PhD thesis, Michigan Technological University, (2000).

[107] B. STEVENS, A. LING, AND E. MENDELSOHN, A direct construction of transversal covers using group divisible designs, *Ars Combin.*, **63** (2002) 145-159.

[108] M. CHATEAUNEUF AND D. L. KREHER, On the State of Strength-Three Covering Arrays, *J. Combin. Designs* **10** (2002), 217–238.

[109] TRAN VAN TRUNG AND SOSINA MARTIROSYAN, On Covering Arrays, *preprint.*

[110] G. SHERWOOD, Construction of orthogonal arrays and covering arrays using permutation groups,
http://home.att.net/ gsherwood/cover.htm.

Index