On the Connections Between Universal Hashing, Combinatorial Designs and Error-Correcting Codes

D. R. Stinson
Department of Computer Science and Engineering
and Center for Communication and Information Science
University of Nebraska-Lincoln
Lincoln NE 68588, USA
stinson@bibd.unl.edu
http://bibd.unl.edu/~stinson/

October 4, 1995

Abstract

In this primarily expository paper, we discuss the connections between two popular and useful tools in theoretical computer science, namely, universal hashing and pairwise independent random variables; and classical combinatorial stuctures such as error-correcting codes, balanced incomplete block designs, difference matrices and orthogonal arrays.

1 Introduction

The concept known as "universal hashing" was invented by Carter and Wegman [5] in 1979. In [29, p. 18], Avi Wigderson characterizes universal hashing as being a tool which "should belong to the fundamental bag of tricks of every computer scientist". This is no exaggeration, as there are probably well in excess of fifty papers in theoretical computer science that employ universal hashing as an important tool. Several of the most attractive applications are outlined in the the lecture notes [29].

A closely related topic goes by several names: "strongly universal hashing" [27], "two-point based sampling" [6], and "pairwise independent random variables" [11]. A recent paper by Wigderson [28], presented at the *26th Symposium on the Theory of Computing* Conference held in Montréal in 1994, is entitled "The Amazing Power of Pairwise Independence." This paper contains a bibliography of 36 papers in computer science concerning this topic.

The applications of these topics in computer science are numerous. Here are a few examples:

- the static and dynamic dictionary problems
- partial and complete derandomization of randomized algorithms
- public vs. private coins in interactive proofs
- authentication codes

2 Definitions of Hash Families

We begin with the relevant definitions of various types of hash families.

DEFINITION An $(N; n, m)$-*hash family* is a set of N functions $\mathcal{F}$ such that

$$f : X \to Y$$

for each $f \in \mathcal{F}$, where $|X| = n$ and $|Y| = m$. There will be no loss in generality in assuming $n \geq m$.

DEFINITION An $(N; n, m)$-hash family is ϵ-*universal* provided that for any two distinct elements $x_1, x_2 \in X$, there exist at most ϵN functions $f \in \mathcal{F}$

such that $f(x_1) = f(x_2)$. We will use the notation ϵ-U as an abbreviation for ϵ-universal.

The relevance of this definition to hashing is clear: If a function f is chosen at random from a given ϵ-U $(N; n, m)$-hash family, then the probability that any two distinct inputs collide under f is at most ϵ.

Two special cases of ϵ-U hash families are of particular interest.

- The case $\epsilon = 1/m$ is known as *universal hashing.* This is the definition originally given in 1979 by Carter and Wegman [5].
- The case $\epsilon = (n - m)/(m(n - 1))$ is known as *optimally-universal hashing.* (The reason for this terminology is that $\epsilon \geq (n-m)/(m(n-1))$ in any ϵ-U $(N; n, m)$ hash family; see Corollary 3.7.) This definition was first given in 1980 by Sarwate [21].

Here are some more definitions.

DEFINITION Suppose that the functions in an $(N; n, m)$ hash family, $\mathcal{F}$, have range $Y = G$, where G is an additive abelian group (of order m). $\mathcal{F}$ is called ϵ-Δ *universal* provided that for any two distinct elements $x_1, x_2 \in X$ and for any element $y \in G$, there exist at most ϵN functions $f \in \mathcal{F}$ such that $f(x_1) - f(x_2) = y$. We will use the notation ϵ-ΔU as an abbreviation for ϵ-Δ universal.

In the case where $G = (\mathbb{Z}_2)^\ell$ for some integer ℓ, the definition reduces to what Krawczyk [15, 16] calls "ϵ-opt secure" and Rogaway [20] terms "almost XOR universal" (in this situation, the difference of two ℓ-tuples is the same thing as the bitwise exclusive-or).

Here are yet more definitions.

DEFINITION An $(N; n, m)$ hash family is *ϵ-strongly universal* [25] provided that the following two conditions are satisfied:

1. for any element $x \in X$ and any element $y \in Y$, there exist exactly N/m functions $f \in \mathcal{F}$ such that $f(x) = y$.
2. for any two distinct elements $x_1, x_2 \in X$ and for any two (not necessarily distinct) elements $y_1, y_2 \in Y$, there exist at most $\epsilon N/m$ functions $f \in \mathcal{F}$ such that $f(x_i) = y_i$, $i = 1, 2$.

We will use the notation ϵ-SU as an abbreviation for ϵ-strongly universal.

It is not difficult to see that $\epsilon \geq 1/m$ in any ϵ-SU $(N; n, m)$ hash family; see Theorem 5.1. The case when $\epsilon = 1/m$ was termed *strongly universal* (or SU) by Wegman and Carter [27]. An SU hash family is the same thing as *pairwise independent random variables.* That is, if a hash function $f \in \mathcal{F}$ is chosen uniformly at random, then it is easy to see that

$$p(x_1 = y_1, x_2 = y_2) = p(x_1 = y_1) \times p(x_2 = y_2) = \frac{1}{m^2},$$

for all distinct $x_1, x_2 \in X$ and all $y_1, y_2 \in Y$.

Warning: Some authors refer to what we have defined as "strongly universal" as just "universal". It is true that strongly universal imples universal (see Theorem 2.1), but the converse is not true. Thus we use the terms "strongly universal" and "universal" in accordance with the original Wegman-Carter definitions in order to differentiate between the two classes.

For all the classes we have defined, we will be particularly interested in cases where n and m are powers of a given prime power q.

We will typically depict an $(N; n, m)$ hash family in the form of an $N \times n$ array of m symbols, where each row of the array corresponds to one of the functions in the family. In the case of an ϵ-U $(N; n, m)$ hash family, this array has the property that, for any two columns, there exist at most ϵN rows such that the entries in the two given columns are equal. The array corresponding to an ϵ-ΔU $(N; n, m)$ hash family has the property that, for any two columns, there exist at most ϵN rows such that the entries in the two given columns have a specified difference. Finally, in the case of an ϵ-SU $(N; n, m)$ hash family, each element occurs the same number of times in each column, and if we inspect any two columns, we find every possible ordered pair of elements occuring at most $\epsilon N/m$ times.

In later sections, we will discuss in detail the relation between these hash families and various types of classical combinatorial designs. For now we observe the following relations between the classes we have already defined.

Theorem 2.1 *Suppose $\mathcal{F}$ is an $(N; n, m)$ hash family. Then the following implications hold:*

1. *If $\mathcal{F}$ is ϵ-SU, then $\mathcal{F}$ is ϵ-U.*

2. If $\mathcal{F}$ is ϵ-SU and Y is an abelian group, then $\mathcal{F}$ is ϵ-ΔU.

3. If $\mathcal{F}$ is ϵ-ΔU, then $\mathcal{F}$ is ϵ-U.

Proof.

1. Suppose $\mathcal{F}$ is an ϵ-SU hash family, and x_1, x_2 are distinct elements in x. For each $y \in Y$, there are at most $\epsilon N/m$ functions $f \in \mathcal{F}$ such that $f(x_1) = f(x_2) = y$. Since there are m choices for y, there are at most ϵN functions $f \in \mathcal{F}$ such that $f(x_1) = f(x_2)$.

2. Suppose $\mathcal{F}$ is an ϵ-SU hash family, and Y is an abelian group. Let y be any element in Y. For each $y_1 \in Y$, there are at most $\epsilon N/m$ functions $f \in \mathcal{F}$ such that $f(x_1) = y_1 + y$ and $f(x_2) = y_1$. Note that, if $f(x_1) - f(x_2) = y$, then $f(x_1) = y_1 + y$ and $f(x_2) = y_1$ for a uniquely defined value $y_1 \in Y$. Since there are m choices for y_1, there are at most ϵN functions $f \in \mathcal{F}$ such that $f(x_1) - f(x_2) = y$.

3. The ϵ-ΔU condition with $y = 0$ is the same as the ϵ-U condition.

□

3 Universal Families and Codes

Universal hash families turn out to be equivalent to certain (error-correcting) codes, which we now define.

DEFINITION Let Y be an alphabet of N symbols. An (N, K, D, q) *code* is a set $\mathcal{C}$ of K vectors in Y^N such that the Hamming distance between any two distinct vectors in $\mathcal{C}$ is at least D. If the code is *linear* (i.e., if q is a prime power, $Y = \mathbb{F}_q$, and $\mathcal{C}$ is a subspace of $(\mathbb{F}_q)^N$), then we will denote it by an $[N, k, D, q]$ code, where $k = \log_q K$ is the *dimension* of the code.

The following equivalence was first observed by Bierbrauer, Johansson, Kabatianskii and Smeets [2].

Theorem 3.1 *If there exists an (N, K, D, q) code, then there exists a $(1-\frac{D}{N})$-U $(N; K, q)$ hash family. Conversely, if there exists an ϵ-U $(N; n, m)$ hash family, then there exists an $(N, n, N(1-\epsilon), m)$ code.*

Proof. Suppose $\mathcal{C} = \{C_1, \ldots, C_K\}$ is the hypothesized code. Construct an $N \times K$ array, A, in which the columns are the codewords in $\mathcal{C}$. If we look at any two columns of A, we see that they contain different entries in at least D rows. Setting $D = (1 - \epsilon)N$, the associated hash family has $\epsilon = 1 - D/N$.

The process can be reversed: by taking the columns of the array associated with an ϵ-U $(N; n, m)$ hash family as codewords of a code, we obtain an (N, n, d, m) code with $d \geq N(1 - \epsilon)$. □

One nice application, mentioned in [2], uses Reed-Solomon codes. An *extended Reed-Solomon* code is a linear code having parameters $[q, k, q - k + 1, q]$, where $k \leq q$ and q is a prime power (see, for example, [18]). Applying Theorem 3.1, the following is obtained.

Theorem 3.2 *Suppose q is prime and $1 \leq k \leq q$. Then there is a $\frac{k-1}{q}$-U $(q; q^k, q)$ hash family.*

EXAMPLE Suppose we take $q = 5$, $k = 3$. We will construct a $\frac{2}{5}$-U $(5; 125, 5)$ hash family, $\mathcal{F}$.

We need a $[5, 3, 3, 5]$ Reed-Solomon code, $\mathcal{C}$. Such a code has generator matrix

$$G = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 3 & 0 \\ 1 & 4 & 1 & 4 & 0 \end{pmatrix}.$$

Saying that G is a generator matrix for $\mathcal{C}$ is equivalent to saying that the rows of G form a basis for $\mathcal{C}$.

There are five functions in $\mathcal{F}$, which we denote by f_i, $i \in \mathbb{Z}_5$. Each $f_i : (\mathbb{Z}_5)^3 \to \mathbb{Z}_5$. Now, a typical codeword in $\mathcal{C}$ is obtained by computing $(a, b, c)G$, $a, b, c \in \mathbb{Z}_5$. $f_i(a, b, c)$ is computed by extracting the value of the ith co-ordinate from the codeword $(a, b, c)G$. If we write $G = (g_{ij})$, then $g_{ij} = 2^{ij} \bmod 5$ if $0 \leq j \leq 3$. Thus it is not difficult to compute the following formula for the hash functions f_i, $i \in \mathbb{Z}_5$:

$$f_i(a, b, c) = \begin{cases} a + b2^i + c4^i \bmod 5 & \text{if } 0 \leq i \leq 3 \\ a & \text{if } i = 4. \end{cases}$$

Codes from algebraic geometry can profitably be applied here; see Bierbrauer [1], for example.

We now look at bounds on ϵ-U hash families. Not surprisingly, we will employ well-known bounds from coding theory. The *Plotkin bound* (see, for example [17, p. 58]) implies that

$$\frac{D}{N} \leq \frac{K(q-1)}{(K-1)q}$$

in any (N, K, D, q) code. It can be used to derive the following lower bound on ϵ that was first proved by Sarwate [21].

Theorem 3.3 *If there exists an ϵ-U $(N; n, m)$ hash family, then*

$$\epsilon \geq \frac{n-m}{m(n-1)}.$$

Proof. Using Theorem 3.1, construct an $(N, n, N(1-\epsilon), m)$ code from the hash family. This code must satisfy the Plotkin bound, so we obtain

$$\frac{N(1-\epsilon)}{N} \leq \frac{n(m-1)}{(n-1)m}.$$

This simplifies to yield the desired result. □

We can also employ the Plotkin bound to give a quick derivation of the following lower bound on N which was proved from first principles by Stinson [25, Theorem 4.1].

Theorem 3.4 *[25] If there exists an ϵ-U $(N; n, m)$ hash family, then*

$$N \geq \frac{n(m-1)}{n(\epsilon m-1)+m^2(1-\epsilon)}.$$

Proof. Using Theorem 3.1, construct an $(N, n, N(1-\epsilon), m)$ code from the hash family, and then construct the shortened code (see [17, p. 45]), with parameters $(N-1, n/m, N(1-\epsilon), m)$. Since this shortened code must also satisfy the Plotkin bound, we obtain

$$\frac{N(1-\epsilon)}{N-1} \leq \frac{\frac{n}{m}(m-1)}{\left(\frac{n}{m}-1\right)m}.$$

This simplifies to yield the desired result. □

3.1 U Families

As mentioned above, a $\frac{1}{m}$-U $(N; n, m)$ hash family is often called "universal", and denoted as a U $(N; n, m)$ hash family. Setting $\epsilon = 1/m$ in Theorem 3.4, we get the following result.

Corollary 3.5 *[24] If there exists a U $(N; n, m)$ hash family, then $N \geq n/m$.*

Here is an infinite class of optimal U hash families that can be produced from first order Reed-Muller codes. A *first order Reed-Muller code* is a linear code having parameters $[q^{a-1}, a, q^{a-1} - q^{a-2}, q]$, where $a \geq 2$ and q is a prime power (see, for example, [26, p. 44]).

Applying Theorem 3.1, we have the following.

Theorem 3.6 *Suppose q is prime and $a \geq 2$. Then there is a $\frac{1}{q}$-U $(q^{a-1}; q^a, q)$ hash family.*

EXAMPLE From a $[4, 3, 2, 2]$ Reed-Muller code, we can construct a $\frac{1}{2}$-U $(4; 8, 2)$ hash family. The code has generator matrix

$$G = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{pmatrix}.$$

This gives rise to the following hash family:

0	1	1	0	1	0	0	1
0	0	1	1	1	1	0	0
0	1	0	1	1	0	1	0
0	0	0	0	1	1	1	1

EXAMPLE From a $[q, 2, q-1, q]$ Reed-Muller code (which is the same as a $[q, 2, q-1, q]$ Reed-Solomon code), we can construct a $\frac{1}{q}$-U $(q; q^2, q)$ hash family, $\mathcal{F}$, as follows. For each $x \in \mathbb{F}_q$, let $f_x : \mathbb{F}_q \times \mathbb{F}_q \to \mathbb{F}_q$ be defined as $f_x(y, z) = xy + z$.

3.2 OU Families

We stated earlier that an $\frac{n-m}{m(n-1)}$-U $(N; n, m)$ hash family is known as optimally universal, and denoted as an OU $(N; n, m)$ hash family.

Setting $\epsilon = (n-m)/(m(n-1))$ in Theorem 3.4, we get the following result.

Corollary 3.7 *[24] If there exists an OU $(N; n, m)$ hash family, then $N \geq (n-1)/(m-1)$.*

It is possible to characterize the OU hash families in terms of resolvable balanced incomplete block designs. We require some definitions before stating our main theorems.

DEFINITION A (v, k, λ)-*BIBD* (or balanced incomplete block design) is a pair $(Y, \mathcal{B})$ where Y is a set of v elements called *points* and $\mathcal{B}$ is a set of k-subsets of Y called *blocks*, such that every pair of points occurs in exactly λ blocks.

By elementary counting it can be shown that every point occurs in exactly

$$r = \frac{\lambda(v-1)}{k-1}$$

blocks, and

$$|\mathcal{B}| = b = \frac{vr}{k} = \frac{\lambda(v^2-v)}{k^2-k}.$$

DEFINITION A (v, k, λ)-BIBD, $(Y, \mathcal{B})$, is said to be *resolvable* if $\mathcal{B}$ can be partitioned into r *parallel classes*, each of which consists of v/k blocks that partition Y.

A famous inequality of Bose [3] states that $b \geq v + r - 1$ in a resolvable (v, k, λ)-BIBD (equivalently, $r \geq k + \lambda$).

DEFINITION If a resolvable (v, k, λ)-BIBD has $b = v + r - 1$ (equivalently, $r = k + \lambda$), then it is termed *affine resolvable*.

The following was first shown by Stinson in [24].

Theorem 3.8 *An OU $(N; n, m)$ hash family with is equivalent to a resolvable (v, k, λ)-BIBD, where $v = n$, $k = n/m$ and $\lambda = N(n-m)/(m(n-1))$. The BIBD is affine resolvable if and only if $N = (n-1)/(m-1)$.*

Affine resolvable BIBDs have been studied extensively. A survey was published by Shrikhande [23]. On interesting property of affine resolvable BIBDs is that any two blocks from diferent parallel classes have precisely k^2/v points in common [3]. It is also known that the parameters of an affine resolvable BIBD have the form $k = s\mu$, $v = s^2\mu$ and $\lambda = (s\mu - 1)/(s-1)$ for positive integers s and μ (see [3]). However, the only parameters for which affine resolvable BIBDs are known to exist are as follows (see [23]):

1. An affine resolvable $(q^n, q^{n-1}, (q^{n-1} - 1)/(q-1))$-BIBD exists whenever q is a prime power and $n \geq 2$. The blocks of the design are the hyperplanes of $AG(n,q)$, the n-dimensional affine geometry over $\mathbb{F}_q$.

2. An affine resolvable $(4t, 2t, 2t-1)$-BIBD exists whenever a Hadamard matrix of order $4t$ exists. (It is widely believed that Hadamard matrices exist for all orders divisible by four. For a recent survey, see [22].)

We obtain the following optimal OU hash families as a consequence:

Corollary 3.9 *[24]*

1. *Let q be a prime power and let $a \geq 2$ be an integer. Then there exists an OU $((q^a - 1)/(q-1); q^a, q)$ hash family.*

2. *Suppose there is a Hadamard matrix of order $n \equiv 0 \bmod 4$. Then there is an OU $(n-1; n, 2)$ hash family.*

EXAMPLE The following is an affine resolvable $(9,3,1)$-BIBD:

$\{1,2,3\}$	$\{4,5,6\}$	$\{7,8,9\}$
$\{1,4,7\}$	$\{2,5,8\}$	$\{3,6,9\}$
$\{1,5,9\}$	$\{2,6,7\}$	$\{3,4,8\}$
$\{1,6,8\}$	$\{2,4,9\}$	$\{3,5,7\}$

It gives rise to the following hash family:

1	1	1	2	2	2	3	3	3
1	2	3	1	2	3	1	2	3
1	2	3	3	1	2	2	3	1
1	2	3	2	3	1	3	1	2

4 ΔUniversal Families and Difference Matrices

Although ϵ-ΔU hash families were defined only recently, many constructions for strongly universal hash families use them either implicitly or explicitly. Examples can be found in [9, 15, 16, 20] as well as in other papers. Thus we feel that ϵ-ΔU hash families are an important concept in their own right.

ϵ-ΔU hash families are closely related to difference matrices. Here is a definition of these combinatorial structures.

DEFINITION Let G be an additive abelian group. A $(g, k; \lambda)$ *difference matrix* defined over G is a $k \times g\lambda$ matrix $D = (d_{ij})$, such that, for all h, i such that $1 \leq h < i \leq k$, and for all elements $x \in G$, there exist exactly λ columns j $(1 \leq j \leq g\lambda)$ such that $d_{hj} - d_{ij} = x$.

Difference matrices are important structures in combinatorial design theory. For a summary of the main results in this area, see [7].

We now begin our discussion of ϵ-ΔU hash families. The following theorem has an easy counting proof.

Theorem 4.1 *If there exists an ϵ-ΔU $(N; n, m)$ hash family, then $\epsilon \geq 1/m$.*

Proof. Let $x_1, x_2 \in X$ be distinct. For any $y \in Y$, let

$$N_y = |\{f \in \mathcal{F} : f(x_1) - f(x_2) = y\}|.$$

Then $N_y \leq \epsilon N$ for all $y \in Y$, and

$$\sum_{y \in Y} N_y = N.$$

Hence, $\epsilon \geq 1/m$. □

The next theorem provides the connection between ϵ-ΔU hash families with smallest possible ϵ and difference matrices.

Theorem 4.2 *A $\frac{1}{m}$-ΔU $(N; n, m)$ hash family defined over an abelian group G of order m is equivalent to an $(m, n, N/m)$ difference matrix defined over G.*

Proof. If we transpose a difference matrix, we obtain a matrix corresponding to a hash family of the desired type (and conversely). □

It was shown by Jungnickel in 1979 [14] that $k \leq \lambda g$ in any $(g, k; \lambda)$ difference matrix. (Difference matrices in which equality holds are known as *generalized Hadamard matrices*; see [8] for a survey.) Recasting Jungnickel's bound in terms of ΔU hash families, we obtain the following result.

Theorem 4.3 *If there exists a* $\frac{1}{m}$*-*ΔU $(N; n, m)$ *hash family, then* $N \geq n$.

We will now prove a bound for ϵ-ΔU $(N; n, m)$ hash families that contains Theorem 4.3 as a special case. Our bound makes use of the well-known second Johnson bound for constant-weight binary codes, which we define now. Consider a $(N, K, D, 2)$ code, $\mathcal{C}$, over a binary alphabet $Y = \{0, 1\}$. The code $\mathcal{C}$ is said to be a *constant-weight* code if every codeword contains exactly w 1's, for some integer w which we call the *weight* of $\mathcal{C}$.

The *second Johnson bound* states that

$$K \leq \frac{\delta N}{w^2 - wN + \delta N}$$

in a constant-weight $(N, K, 2\delta, 2)$ code having weight w.

Here is our new bound.

Theorem 4.4 *If there exists an* ϵ*-*ΔU $(N; n, m)$ *hash family, then*

$$N \geq \frac{n(m-1)}{m - n + m\epsilon(n-1)}.$$

Proof. Without loss of generality, we can assume that there is one function $h \in \mathcal{F}$ such that $h(x) = 0$ for all $x \in X$. (If $\mathcal{F}$ does not contain such a function h, then it can easily be altered so that it does.) For any $f \in \mathcal{F}$ and any $y \in G$, define a function $f_y : X \to G$ by the rule

$$f_y(x) = f(x) + y.$$

Then for any $f \in \mathcal{F}$ and any $y \in G$, define a function $\hat{f}_y : X \to \{0, 1\}$ by the rule

$$\hat{f}_y(x) = \begin{cases} 1 & \text{if } f_y(x) = 0 \\ 0 & \text{otherwise.} \end{cases}$$

Now, consider the $(N-1)m$ functions $\hat{f}_y$, $f \neq h$. Write these functions as an array, and take the columns of this array as a code. This code is seen to be a constant-weight $((N-1)m, n, 2N(1-\epsilon), 2)$ code with weight $w = N-1$. Applying the second Johnson bound, we obtain the following:

$$n \leq \frac{N(1-\epsilon)(N-1)m}{(N-1)^2 - (N-1)^2 m + N(1-\epsilon)(N-1)m}.$$

Simplifying, we get the desired bound. □

Let us now look at constructions of optimal $\frac{1}{m}$-ΔU $(N; n, m)$ hash families. As mentioned above, these are equivalent to generalized Hadamard matrices. The following result on generalized Hadamard matrices is often attributed to Drake [10], but it was in fact first proved by Bose and Bush in 1952 [4].

Theorem 4.5 *Let q be a prime power. For any positive integers a, b such that $a \geq b$, there exists a $(q^b, q^a; q^{a-b})$ difference matrix defined over $(\mathbb{F}_q)^a$.*

The following class of optimal ΔU hash families is produced.

Corollary 4.6 *Let q be a prime power. For any positive integers a, b such that $a \geq b$, there exists a $\frac{1}{q^b}$-ΔU $(q^a; q^a, q^b)$ hash family defined over $(\mathbb{F}_q)^b$.*

Here is a description of the hash family produced by this construction. Let $X = \mathbb{F}_{q^a}$ and let $G = (\mathbb{F}_q)^b$. X is a vector space over $\mathbb{F}_q$ of dimension a. Let $\phi : X \rightarrow G$ be any surjective linear transformation; then $|\phi^{-1}(y)| = q^{a-b}$ for every $y \in G$. (For example, if elements of X are represented as a-dimensional vectors over $\mathbb{F}_q$, then $\phi(x)$ could be defined to be the last b co-ordinates of x.) Then $\mathcal{F} = \{f_x : x \in A\}$, where $f_x(z) = \phi(xz)$. (Observe that the array representation of the hash family is also a difference matrix as defined above; it is not necessary to transpose since $f_x(z) = f_z(x)$.)

EXAMPLE We construct a $\frac{1}{4}$-ΔU $(8; 8, 4)$ hash family that is also a $(8, 8; 2)$ difference matrix over $\mathbb{Z}_2 \times \mathbb{Z}_2$. We begin with the multiplication table of $\mathbb{F}_8 = \mathbb{Z}_2[x]/(x^3 + x + 1)$, where the polynomial $ax^2 + bx + c$ is represented by the triple abc:

	000	001	010	011	100	101	110	111
000	000	000	000	000	000	000	000	000
001	000	001	010	011	100	101	110	111
010	000	010	100	110	011	001	111	101
011	000	011	110	101	111	100	001	010
100	000	100	011	111	110	010	101	001
101	000	101	001	100	010	111	011	110
110	000	110	111	001	101	011	010	100
111	000	111	101	010	001	110	100	011

Then we take the last two co-ordinates of each entry in the table to construct the hash family:

00	00	00	00	00	00	00	00
00	01	10	11	00	01	10	11
00	10	00	10	11	01	11	01
00	11	10	01	11	00	01	10
00	00	11	11	10	10	01	01
00	01	01	00	10	11	11	10
00	10	11	01	01	11	10	00
00	11	01	10	01	10	00	11

Let us compare the construction above with a recent construction given by Mansour, Nisan and Tiwari [19] that uses Toeplitz matrices. An $a \times b$ matrix $S = (s_{i,j})$ is called a *Toeplitz matrix* if $s_{i+1,j+1} = s_{i,j}$ for all i, j such that $1 \leq i \leq a-1$ and $1 \leq j \leq b-1$. Thus a Toeplitz matrix is defined by the $a+b-1$ entries in its first row and column. Let $\mathcal{S}(a,b)$ denote the set of all $a \times b$ Toeplitz matrices over $\mathbb{F}_q$. For any $S \in \mathcal{S}(a,b)$, define a function $f_S : (\mathbb{F}_q)^a \to (\mathbb{F}_q)^b$, where $f_S(x) = xS$, $x \in (\mathbb{F}_q)^a$. Then it can be shown that $\{f_S : S \in \mathcal{S}(a,b)\}$ comprises a $\frac{1}{q^b}$-ΔU $(q^{a+b-1}; q^a, q^b)$ hash family defined over $(\mathbb{F}_q)^b$. Thus the hash families from Theorem 4.6 would be preferred since they are smaller.

It is also possible to use certain error-correcting codes to construct ϵ-ΔU hash families for various values of ϵ. Here is a very useful construction, which is essentially the same as the *q-twisted construction* of Johansson, Kabatianskii and Smeets [12, 13].

Theorem 4.7 *If there exists an $[N, k, D, q]$ code $\mathcal{C}$ with the property that $\mathbf{e} = (1, \ldots, 1) \in \mathcal{C}$, then there exists a $(1 - \frac{D}{N})$-ΔU $(N; q^{k-1}, q)$ hash family defined over $\mathbb{F}_q$.*

Proof. Let $C_1, \ldots, C_{q^{k-1}}$ be a set of representatives of the quotient space $\mathcal{C}/\langle \mathbf{e} \rangle$. Construct a $N \times q^{k-1}$ array, A, in which the columns are the codewords $C_1, \ldots, C_{q^{k-1}}$. As usual, the rows of this array will represent the hash functions in our family $\mathcal{F}$. We will now determine ϵ such that $\mathcal{F}$ is an ϵ-ΔU $(N; q^{k-1}, q)$ hash family.

Let us consider two columns of A, say C_i, C_j, and let $x \in \mathbb{F}_q$. Let δ denote the number of co-ordinates h such that $C_i(h) - C_j(h) = x$. Now it is easy to see that $C_i - C_j$, which is a codeword in $\mathcal{C}$, has exactly δ co-ordinates containing the symbol x. Hence $d(C_i - C_j, x\mathbf{e}) = N - \delta$.

On the other hand, the vector $x\mathbf{e}$ is a codeword in $\mathcal{C}$, and $C_i - C_j \neq x\mathbf{e}$ since the C_i's were chosen from the quotient space. Hence, $d(C_i - C_j, x\mathbf{e}) \geq D$.

Combining the two inequalities, we see that $\delta \leq D - N$, from which it follows that $\epsilon \leq 1 - D/N$. □

We can use (extended) Reed-Solomon codes here in a similar fashion as was done in Theorem 3.2. Note that, in the usual presentation of Reed-Solomon codes, $(1, \ldots, 1)$ is a codeword. From a code of this type having parameters $[q, k, q-k+1, q]$, where $k \leq q$ and q is a prime power, the following is obtained by application of Theorem 4.7.

Theorem 4.8 *Suppose q is prime and $1 \leq k \leq q$. Then there is a $\frac{k-1}{q}$-ΔU $(q; q^{k-1}, q)$ hash family.*

EXAMPLE Earlier, we constructed a $\frac{2}{5}$-U $(5; 125, 5)$ hash family from a $[5,3,3,5]$ Reed-Solomon code, $\mathcal{C}$. We now use the same code to produce a $\frac{2}{5}$-ΔU $(5; 25, 5)$ hash family, $\mathcal{F}$. Recall that $\mathcal{C}$ can be constructed from the generator matrix

$$G = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 3 & 0 \\ 1 & 4 & 1 & 4 & 0 \end{pmatrix}.$$

Hence, it is clear that $(1,1,1,1,1) \in \mathcal{C}$.

There are five functions in $\mathcal{F}$, which we denote by f_i, $i \in \mathbb{Z}_5$. Each $f_i : (\mathbb{Z}_5)^2 \to \mathbb{Z}_5$, where

$$f_i(b,c) = \begin{cases} b2^i + c4^i \bmod 5 & \text{if } 0 \leq i \leq 3 \\ 0 & \text{if } i = 4. \end{cases}$$

5 Strongly Universal Families and Orthogonal Arrays

The following bound has a trivial counting proof.

Theorem 5.1 *If there exists an ϵ-SU $(N; n, m)$ hash family, then $\epsilon \geq 1/m$.*

Proof. Let $x_1, x_2 \in X$ be distinct elements. For any $y_1, y_2 \in Y$, let

$$N_{y_1,y_2} = |\{f \in \mathcal{F} : f(x_1) = y_1, f(x_2) = y_2\}|.$$

Then $N_{y_1,y_2} \leq \epsilon N/m$ for all $y_1, y_2 \in Y$, and

$$\sum_{y_1,y_2 \in Y} N_{y_1,y_2} = N.$$

Hence, $\epsilon \geq 1/m$. □

ϵ-SU hash families with smallest possible ϵ are equivalent to orthogonal arrays, which are defined as follows.

DEFINITION Let Y be a set of v symbols. An *orthogonal array* $OA_\lambda(k, v)$ is a $\lambda v^2 \times k$ array A of elements from Y, such that, within any two columns of A, every possible ordered pair of symbols occurs in exactly λ rows.

The following theorem is easy to prove.

Theorem 5.2 *[24] A $\frac{1}{m}$-SU $(N; n, m)$ hash family is equivalent to an orthogonal array $OA_\lambda(n, m)$ in which $\lambda = N/m^2$.*

Proof. If we write the functions in a $\frac{1}{m}$-SU $(N; n, m)$ hash family as the rows of an $N \times n$ array, then we obtain an $OA_{N/m^2}(n, m)$, and conversely. □

We now turn to bounds on the size of ϵ-SU hash families. The following bound, proved in [25] from first principles, can also be obtained by an appropriate application of the second Johnson bound for constant weight binary codes. The proof is similar to that of Theorem 4.4.

Theorem 5.3 *If there exists an ϵ-SU $(N; n, m)$ hash family, then*

$$N \geq 1 + \frac{n(m-1)^2}{m\epsilon(n-1) + m - n}.$$

Proof. Let $y \in Y$ be any symbol. Without loss of generality, we can assume that there is one function $h \in \mathcal{F}$ such that $h(x) = y$ for all $x \in X$. (If $\mathcal{F}$ does not contain such a function h, then it can easily be altered so that it does.) For any $f \in \mathcal{F}$, define a function $\hat{f} : X \rightarrow \{0, 1\}$ by the rule

$$\hat{f}(x) = \begin{cases} 1 & \text{if } f(x) = y \\ 0 & \text{otherwise.} \end{cases}$$

Now, consider the $N - 1$ functions $\hat{f}$, $f \neq h$. Write these functions as the rows of an array, and take the columns of this array as a code. This code is seen to be a constant-weight $(N-1, n, 2N(1-\epsilon)/M, 2)$ code with weight $w = N/m - 1$. Applying the second Johnson bound, we obtain the following:

$$n \leq \frac{\frac{N}{m}(1-\epsilon)(N-1)}{\left(\frac{N}{m}-1\right)^2 - \left(\frac{N}{m}-1\right)(N-1) + \frac{N}{m}(1-\epsilon)(N-1)}.$$

Simplifying, we get the desired bound. □

Let's look at some constructions for ϵ-SU hash families. One easy way to construct these families uses ϵ-ΔU hash families.

Theorem 5.4 *If there exists an ϵ-ΔU $(N; n, m)$ hash family, then there exists an ϵ-SU $(Nm; n, m)$ hash family,*

Proof. Let $\mathcal{F}$ be the hypothesized ϵ-ΔU $(N; n, m)$ hash family, defined over the abelian group G. For any $f \in \mathcal{F}$ and any $y \in G$, define a function $f_y : X \rightarrow G$ by the rule

$$f_y(x) = f(x) + y.$$

Then define $\mathcal{F}'$ to consist of all the functions f_y, $f \in \mathcal{F}$, $y \in G$. It is easily seen that $\mathcal{F}'$ is an ϵ-SU $(Nm; n, m)$ hash family. □

The following theorem is an immediate corollary of Theorem 5.4 and Corollary 4.6. The equivalent class of orthogonal arrays was first constructed by Bose and Bush [4].

Theorem 5.5 *Let q be a prime power. For any positive integers a, b such that $a \geq b$, there exists $\frac{1}{q^b}$-SU $(q^{a+b}; q^a, q^b)$ hash family.*

The special case $a = b = 1$ is of particular interest: a $\frac{1}{q}$-SU $(q^2; q, q)$ hash family is called "two-point sampling" by Chor and Goldreich [6]. The reason for this terminology is that two elements of $\mathbb{F}_q$ suffice to select a hash function f from this family, from which a sequence of q elements of $\mathbb{F}_q$ is obtained by evaluating f at the q points in $\mathbb{F}_q$. (This is the basis for a particularly simple derandomization technique.)

A $\frac{1}{q}$-SU $(q^2; q, q)$ hash family is of course equivalent to an orthogonal array $\mathrm{OA}_1(q, q)$, which is easily obtained from the Desarguesian affine plane of order q, $\mathrm{AG}(2, q)$. The associated hash family $\mathcal{F} = \{f_{ab}, a, b \in \mathbb{F}_q\}$, where

$$f_{ab}(x) = ax + b.$$

It is interesting to note that Chor and Goldreich attribute this construction to Joffe [11], who presented it as a construction for pairwise independent random variables in 1971. Of course this structure has been known for many years in the context of finite geometries, statistical and combinatorial designs.

We now look at a composition construction from [25] that is based on an idea of Wegman and Carter [27]. The next theorem shows how to compose an ϵ_1-U hash family with an ϵ_2-SU hash family and obtain an $(\epsilon_2 + \epsilon_2)$-SU hash family.

Theorem 5.6 *[25] Suppose there exists an ϵ_1-U $(N_1; n, m_1)$ hash family and an ϵ_2-SU $(N_2; m_1, m_2)$ hash family. Then there is an $(\epsilon_2+\epsilon_2)$-SU $(N_1N_2; n, m_2)$ hash family.*

Proof. Let $\mathcal{F}$ and $\mathcal{G}$ be the ϵ_1-U and ϵ_2-SU hash families, resepctively. We can assume that $f : X \to Y_1$ for every $f \in \mathcal{F}$ and $g : Y_1 \to Y_2$ for every $g \in \mathcal{G}$, where $|X| = n$ and $|Y_i| = m_i$, $i = 1, 2$. For every $f \in \mathcal{F}$ and every $g \in \mathcal{G}$, define a hash function $f \circ g : X \to Y_2$ by the rule

$$(f \circ g)(x) = g(f(x)),$$

and let

$$\mathcal{H} = \{f \circ g : f \in \mathcal{F}, g \in \mathcal{G}\}.$$

We will show that $\mathcal{H}$ is an $(\epsilon_2 + \epsilon_2)$-SU $(N_1N_2; n, m_2)$ hash family.

Let $x_1, x_2 \in X$ $(x_1 \neq x_2)$ and let $y_1, y_2 \in Y_2$. We need to compute an upper bound on the number of functions $h \in \mathcal{H}$ such that $h(x_i) = y_i$, $i = 1, 2$.

We first consider the case when $y_1 = y_2 = y$, say. Let

$$\mathcal{E} = \{f \in \mathcal{F} : f(x_1) = f(x_2)\}$$

and let $\alpha = |\mathcal{E}|$. Since $\mathcal{F}$ is an ϵ_1-U hash family, we have that $\alpha \leq \epsilon_1 N_1$. For any $f \in \mathcal{E}$, there are exactly N_2/m_2 functions $g \in \mathcal{G}$ such that $g(f(x_1)) = g(f(x_2)) = y$. On the other hand, for any $f \in \mathcal{F}\backslash\mathcal{E}$, there are at most $\epsilon_2 N_2/m_2$ functions $g \in \mathcal{G}$ such that $g(f(x_1)) = g(f(x_2)) = y$.

Hence, the number of functions $h \in \mathcal{H}$ such that $h(x_1) = h(x_2) = y$ is at most

$$\begin{aligned}
\alpha \times \frac{N_2}{m_2} + (N_1 - \alpha) \times \frac{\epsilon_2 N_2}{m_2} &= \frac{N_2(\alpha + (N_1 - \alpha)\epsilon_2)}{m_2} \\
&\leq \frac{N_2(\alpha + N_1\epsilon_2)}{m_2} \\
&\leq \frac{N_2(N_1\epsilon_1 + N_1\epsilon_2)}{m_2} \\
&\leq \frac{N_1 N_2(\epsilon_1 + \epsilon_2)}{m_2}.
\end{aligned}$$

If $y_1 \neq y_2$ then the number of functions h is less. Hence, property 2 of an *SU* hash family is satisfied.

It is trivial to prove property 1 of an SU hash family, and thus it follows that we have an ϵ-SU hash family with $\epsilon \leq \epsilon_1 + \epsilon_2$. □

Here is a very nice application of this technique that was presented in [2].

Theorem 5.7 *[2] Suppose r and s are integers. Then there is a $\frac{1}{q^{r-1}}$-SU $(q^{3r+2s}; q^{(r+s)(q^s(q-1)+1)}, q^r)$ hash family.*

Proof. First, apply Theorem 3.2 with q replaced by q^{r+s} and k replaced by $q^s(q-1)+1$. This produces a $\frac{q-1}{q^r}$-U $(q^{r+s}; q^{(r+s)(q^s(q-1)+1)}, q^{r+s})$ hash family. Next, from Theorem 5.5, we obtain a $\frac{1}{q^r}$-SU $(q^{2r+s}; q^{r+s}, q^r)$ hash family. Now, applying Theorem 5.6, we obtain the desired hash family. □

6 Summary

The constructions, bounds and equivalences for hash families that we have presented are summarized in tabular form.

Table 1: Constructions for Hash Families

hash family	ϵ	N	n	m	source
U	$\frac{a-1}{q}$	q	q^a	q	Theorem 3.2
U	$\frac{1}{q}$	q^{a-1}	q^a	q	Theorem 3.6
OU	$\frac{q^{a-1}-1}{q^a-1}$	$\frac{q^{a-1}-1}{q-1}$	q^a	q	Theorem 3.9
ΔU	$\frac{a-1}{q}$	q	q^{a-1}	q	Theorem 4.8
ΔU	$\frac{1}{q^b}$	q^a	q^a	q^b	Theorem 4.6
SU	$\frac{1}{q^{b-1}}$	q^{3b+2a}	$q^{(a+b)(q^a(q-1)+1)}$	q^b	Theorem 5.7
SU	$\frac{1}{q^b}$	q^{a+b}	q^a	q^b	Theorem 5.5

Table 2: Bounds for Hash Families

hash family	bound	source
ϵ-U	$\epsilon \geq \frac{n-m}{m(n-1)}$	Theorem 3.3
ϵ-U	$N \geq \frac{n(m-1)}{n(\epsilon m-1)+m^2(1-\epsilon)}$	Theorem 3.4
ϵ-ΔU	$\epsilon \geq \frac{1}{m}$	Theorem 4.1
ϵ-ΔU	$N \geq \frac{n(m-1)}{m-n+m\epsilon(n-1)}$	Theorem 4.4
ϵ-SU	$\epsilon \geq \frac{1}{m}$	Theorem 5.1
ϵ-SU	$N \geq 1 + \frac{n(m-1)^2}{m\epsilon(n-1)+m-n}$	Theorem 5.3

Table 3: Equivalent Formulations of Hash Families

hash family	equivalent formulation	source
ϵ-U $(N; n, m)$	$(N, n, N(1-\epsilon), m)$ code	Theorem 3.1
OU $(N; n, m)$	resolvable $\left(n, \frac{n}{m}, \frac{N(n-m)}{m(n-1)}\right)$-BIBD	Theorem 3.8
$\frac{1}{m}$-ΔU $(N; n, m)$	$\left(m, n, \frac{N}{m}\right)$ difference matrix	Theorem 4.2
$\frac{1}{m}$-SU $(N; n, m)$	$\mathrm{OA}_{N/m^2}(n, m)$	Theorem 5.2

Acknowledgements

Research of D. R. Stinson is supported by NSF grant CCR–9402141 and NSA grant MDA 904-93-H-3049.

References

[1] J. Bierbrauer, Universal Hashing and Geometric Codes. To appear in *Designs, Codes and Cryptography.*

[2] J. Bierbrauer, T. Johansson, G. Kabatianskii and B. Smeets, On Families of Hash Functions via Geometric Codes and Concatenation. In "Advances in Cryptology – CRYPTO '93", D. R. Stinson, ed., *Lecture Notes in Computer Science* **773** (1994), 331–342.

[3] R. C. Bose, A Note on the Resolvability of Balanced Incomplete Block Designs. *Sankhya* **6** (1942), 105–110.

[4] R. C. Bose and K. A. Bush, Orthogonal Arrays of Strength Two and Three. *Annals Math. Statistics* **23** (1952), 508–524.

[5] J. L. Carter and M. N. Wegman, Universal Classes of Hash Functions. *J. Computer and System Sciences* **18** (1979), 143–154.

[6] B. Chor and O. Goldreich, On the Power of Two-point Based Sampling. *J. Complexity* **5** (1989), 96–106.

[7] C. J. Colbourn and W. de Launey, Difference Matrices. In *The CRC Handbook of Combinatorial Designs* (C. J. Colbourn and J. H. Dinitz, eds.), CRC Press, Inc., to be published.

[8] W. de Launey, A Survey of Generalised Hadamard Matrices and Difference Matrices $D(k, \lambda; g)$ with Large k. *Utilitas Mathematica* **30** (1986), 5–29.

[9] B. den Boer, A Simple and Key-Economical Unconditional Authentication Scheme. *Journal of Computer Security* **2** (1993), 65–71.

[10] D. A. Drake, Partial λ-Geometries and Generalized Hadamard Matrices Over Groups. *Canadian Journal of Mathematics* **31** (1979), 617–627.

[11] A. Joffe, On a Sequence of Almost Deterministic Pairwise Independent Random Variables. *Proc. Amer. Math. Soc.* **29** (1971), 381–382.

[12] T. Johansson, G. Kabatianskii and B. Smeets, On the Relation Between A-Codes and Codes Correcting Independent Errors. In "Advances in Cryptology – EUROCRYPT '93", T. Helleseth, ed., *Lecture Notes in Computer Science* **765** (1994), 1–11.

[13] T. Johansson, B. Smeets and G. Kabatianskii, On the Cardinality of Systematic Authentication Codes via Error Correcting Codes. To appear in *IEEE Transactions on Information Theory.*

[14] D. Jungnickel, On Difference Matrices, Resolvable Transversal Designs and Generalized Hadamard Matrices. *Math. Z.* **167** (1979), 49–60.

[15] H. Krawczyk, LFSR-Based Hashing and Authentication. In "Advances in Cryptology – CRYPTO '94", Y. Desmedt, ed., *Lecture Notes in Computer Science* **839** (1994), 129–139.

[16] H. Krawczyk, New Hash Functions for Message Authentication. In "Advances in Cryptology – EUROCRYPT '95", L. C. Guillou and J.-J. Quisquater, eds., *Lecture Notes in Computer Science* **9219** (1995), 301–310.

[17] J. H. van Lint, *Introduction to Coding Theory.* Springer-Verlag, 1982.

[18] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes.* North-Holland, 1977.

[19] Y. Mansour, N. Nisan and P. Tiwari, The Computational Complexity of Universal Hashing. *Theoretical Computer Science* **107** (1993), 121-133.

[20] P. Rogaway, Bucket Hashing and Its Application to Fast Message Authentication. In "Advances in Cryptology – CRYPTO '95", D. Coppersmith, ed., *Lecture Notes in Computer Science* **963** (1995), 29–42.

[21] D. V. Sarwate, A Note on Universal Classes of Hash Functions. *Information Processing Letters* **10** (1980), 41–45.

[22] J. Seberry and M. Yamada, Hadamard Matrices, Sequences, and Block Designs. In "Contemporary Design Theory, A Collection of Surveys", J. H. Dinitz and D. R. Stinson, eds., John Wiley & Sons, 1992, pp. 431–560.

[23] S. S. Shrikhande, Affine Resolvable Balanced Incomplete Block Designs: A Survey. *Aequationes Math.* **14** (1976), 251-269.

[24] D. R. Stinson, Combinatorial Techniques for Universal Hashing. *J. Computer and System Sciences* **48** (1994), 337–346.

[25] D. R. Stinson, Universal Hashing and Authentication Codes. *Designs, Codes and Cryptography* **4** (1994), 369–380. [Preliminary version appeared in "Advances in Cryptology – CRYPTO '91", J. Feigenbaum, ed., *Lecture Notes in Computer Science* **576** (1992), 74–85.]

[26] M. A. Tsfasman and S. G. Vladut, *Algebraic-Geometric Codes.* Kluwer Academic Publishers, 1991.

[27] M. N. Wegman and J. L. Carter, New Hash Functions and their Use in Authentication and Set Equality. *J. Computer and System Sciences* **22** (1981), 265–279.

[28] A. Wigderson, The Amazing Power of Pairwise Independence. *Proc. 26th Annual ACM Symposium on the Theory of Computing*, ACM Press, 1994, pp. 645–647.

[29] A. Wigderson, Lectures on the Fusion Method and Derandomization. Technical Report SOCS-95.2, School of Computer Science, McGill University (file `/pub/tech-reports/library/reports/95/TR95.2.ps.gz` at the anonymous ftp site `ftp.cs.mcgill.ca`).