Orthogonal Arrays, Resilient Functions, Error Correcting Codes and Linear Programming Bounds*

Jürgen Bierbrauer
Mathematisches Institut der Universität
Im Neuenheimer Feld 288
69120 Heidelberg
Germany

K. Gopalakrishnan
Department of Computer Science and Engineering
University of Nebraska - Lincoln
Lincoln, NE 68588

D. R. Stinson
Department of Computer Science and Engineering and
Center for Communication and Information Science
University of Nebraska - Lincoln
Lincoln, NE 68588

April 20, 1999

Abstract

Orthogonal arrays (OAs) are basic combinatorial structures, which appear under various disguises in cryptology and the theory of algorithms. Among their applications are **universal hashing, authentication codes, resilient and correlation-immune functions, derandomization of algorithms,** and **perfect local randomizers.** In this paper, we give new explicit bounds on the size of orthogonal arrays using Delsarte's linear programming method.

*This paper is an expanded and revised version of the extended abstract "Bounds for Resilient Functions and Orthogonal Arrays" by Jürgen Bierbrauer, K. Gopalakrishnan and D. R. Stinson, which appeared in *Advances in Cryptology, Proceedings of CRYPTO'94,* Lecture Notes in Computer Science, Vol. xxx, pp. yyy-zzz, Springer-Verlag, 1994.

Specifically we prove that the minimum number of rows in a binary orthogonal array of length n and strength t is at least $2^n - (n2^{n-1}/t+1)$ and also at least $2^n - (2^{n-2}(n+1)/\lceil \frac{t+1}{2} \rceil)$. We also prove that these bounds are as powerful as the linear programming bound itself for many parametric situations.

An (n, m, t)- resilient function is a function $f : \{0,1\}^n \longrightarrow \{0,1\}^m$ such that every possible output m-tuple is equally likely to occur when the values of t arbitrary inputs are fixed by an opponent and the remaining $n-t$ input bits are chosen independently at random. A basic problem is to maximize t given m and n, i.e., to determine the largest value of t such that an (n, m, t)-resilient function exists. In this paper, we obtain upper and lower bounds for the optimal values of t where $1 \leq n \leq 25$ and $1 \leq m < n$. The upper bounds are derived from Delsarte's linear programming bound, and the lower bounds come from constructions based on error correcting codes. We also obtain new explicit upper bounds for the optimal values of t.

It was proved by Chor et al in [8] that an $(n, 2, t)$ - resilient function exists if and only if $t < \lfloor \frac{2n}{3} \rfloor$. The corresponding question for $m = 3$ or higher remained open. As a consequence of the new upper bounds, we have completed the determination of the optimal resiliency of resilient functions with $m = 3$, and we have also done most of the cases for $m = 4$. Several other infinite classes of resilient functions are also constructed using the theory of anticodes.

keywords: Resilient functions, Orthogonal arrays, Error correcting codes, Linear programming bound, Anticodes.

1 Introduction

Orthogonal arrays (OAs) are basic combinatorial structures. They and some natural generalizations appear under various disguises in cryptology and the theory of algorithms. Among the applications we mention **universal hashing** and **authentication codes**, **resilient** and **correlation-immune functions**, **derandomization of algorithms** and **perfect local randomizers**.

Here, we concentrate on resilient functions, two possible applications of which are mentioned in [3] and [8]. The first application concerns the generation of shared random strings in the presence of faulty processors. The second involves renewing a partially leaked cryptographic key (one setting in which this would be relevant is quantum cryptography [2]). Correlation-immune functions are used in stream ciphers as combining functions for running-key generators that are resistant to a correlation attack (see, for example, Rueppel [22]).

The concept of binary resilient functions was introduced and studied in the papers Chor *et al* [8], Bennett *et al* [3] and Stinson [24]. Here is the definition. Let $n \geq m \geq 1$ be integers and suppose

$$f : \{0,1\}^n \longrightarrow \{0,1\}^m.$$

We will think of f as being a function that accepts n input bits and produces m output bits. Let $t \leq n$ be an integer. Suppose $(x_1, \ldots, x_n) \in \{0,1\}^n$, where the values of t arbitrary inputs are fixed by an opponent, and the remaining $n-t$ input bits are chosen independently at random. Then f is said to be *t-resilient* provided that every possible output m-tuple is equally likely to occur. More formally, the property can be stated as follows: For every t-subset $\{i_1, \ldots, i_t\} \subseteq \{1, \ldots, n\}$, for every choice of $z_j \in \{0,1\}(1 \leq j \leq t)$, and for every $(y_1, \ldots, y_m) \in \{0,1\}^m$, we have

$$Pr(f(x_1, \ldots, x_n) = (y_1, \ldots, y_m) | x_{i_j} = z_j, 1 \leq j \leq t) = \frac{1}{2^m}.$$

We will refer to such a function f as an (n, m, t) - resilient function. Here are some examples from [8] (all addition is modulo 2):

(1) $m = 1$, $t = n-1$. Define $f(x_1, \ldots, x_n) = x_1 + \ldots + x_n$.

(2) $m = n-1$, $t = 1$. Define $f(x_1, \ldots, x_n) = (x_1 + x_2, x_2 + x_3, \ldots, x_{n-1} + x_n)$.

(3) $m = 2$, $n = 3h$, $t = 2h-1$. Define

$$f(x_1, \ldots, x_{3h}) = (x_1 + \ldots + x_{2h}, x_{h+1} + \ldots + x_{3h}).$$

Many interesting results on resilient functions can be found in [3], [8], [24], [13] and [12]. The basic problem is to maximize t given m and n; or equivalently, to maximize m given n and t. An (n, m, t) resilient function is said to be *optimal* if $(n, m, t+1)$ resilient function does not exist.

The paper is organized as follows. In Section 2 the basic definition of orthogonal arrays and the associated terminology are introduced. After a brief survey of the classical bounds on the size of an orthogonal array in Section 2.1, stronger bounds based on linear programming are developed in Section 2.2. In Section 3 the connections between orthogonal arrays and resilient functions are explored and as a consequence, the lower bounds on the size of an orthogonal array are translated into upper bounds on the optimal resiliency of resilient functions. In Section 4 lower bounds on the optimal resiliency of resilient functions are developed. The lower bounds come from various constructions of resilient functions. In Section 4.1 constructions using linear codes are presented and in Section 4.2 constructions using non-linear codes are presented. Some constructions of new resilient functions from old are discussed in Section 4.3. The development up to this stage culminates in Section 4.4 where a table of upper and lower bounds for the optimal value of t when $1 \leq n \leq 25$ and $1 \leq m < n$ is presented.

Several properties of Krawtchouk polynomials are used extensively in the rest of the paper and because of the great importance of them they are dealt with separately in Section 5. In Section 6, we consider a recent result of Levenshtein and discuss the implication of the result to resilient functions. The result is that the minimum size

$N(n,t)$ of a binary orthogonal array of strength t and length n satisfies the equality $2N(n,t) = N(n+1,t+1)$, when t is even.

New explicit bounds on the size of orthogonal arrays are derived in Section 7 using Delsarte's linear programming method. Specifically we prove that the minimum number of rows in a binary orthogonal array of length n and strength t is at least $2^n - (n2^{n-1}/t+1)$ and also at least $2^n - (2^{n-2}(n+1)/\lceil \frac{t+1}{2} \rceil)$. We prove in Section 8 that these two bounds are as powerful as the linear programming bound itself for many parametric situations.

It was proved by Chor et al in [8] that an $(n,2,t)$ - resilient function exists if and only if $t < \lfloor \frac{2n}{3} \rfloor$. The corresponding question for $m = 3$ or higher remained open. As a consequence of our new explicit bounds, we have completed the determination of the optimal resiliency of resilient functions with $m = 3$, and we have also done most of the cases for $m = 4$. This is the main theme of Section 9. The theory of anticodes provides a method for constructing good linear codes. In many parametric situations, the resilient functions derived from these linear codes are optimal. In Section 9.1, anticodes are studied and such parametric situations are characterized. We conclude the paper with several lingering open problems in Section 10.

2 Orthogonal Arrays

An *orthogonal array* $OA_\lambda(t,k,v)$ is a $\lambda v^t \times k$ array of v symbols, such that in any t columns of the array every one of the possible v^t ordered t-tuples of symbols occurs in exactly λ rows. Usually t is referred to as the *strength* of the orthogonal array, k is called the number of *factors*, v is called the number of *levels* and λ is called the *index* of the orthogonal array. If $\lambda = 1$, then we write $OA(t,k,v)$. An orthogonal array is said to be *simple* if no two rows are identical. Of course, an array with $\lambda = 1$ is simple. In this paper, we consider only simple arrays.

A *large set of orthogonal arrays* $LOA_\lambda(t,k,v)$ is a set of v^{k-t}/λ simple arrays $OA_\lambda(t,k,v)$ such that every possible k-tuple of symbols occurs in exactly one of the orthogonal arrays in the set.

2.1 Classical Bounds

One classical bound for orthogonal arrays is the Rao bound [21], proved in 1947. We record the Rao bound as the following theorem.

Theorem 2.1 *Suppose there exists an $OA_\lambda(t,k,v)$. Then*

$$\lambda v^t \geq 1 + \sum_{i=1}^{t/2} \binom{k}{i} (v-1)^i$$

if t is even; and

$$\lambda v^t \geq 1 + \begin{pmatrix} k-1 \\ (t-1)/2 \end{pmatrix} (v-1)^{(t+1)/2} + \sum_{i=1}^{(t-1)/2} \begin{pmatrix} k \\ i \end{pmatrix} (v-1)^i$$

if t is odd.

Another classical bound which provides necessary conditions for existence of orthogonal arrays of index unity ($\lambda = 1$) is the Bush bound [6], proved in 1952. This bound is as follows:

Theorem 2.2 *Suppose there exists an $OA(t, k, v)$, where $t > 1$. Then*

$$\begin{array}{cl} k \leq v + t - 1 & \textit{if } v \geq t,\ v \textit{ even} \\ k \leq v + t - 2 & \textit{if } v \geq t \geq 3,\ v \textit{ odd} \\ k \leq t + 1 & \textit{if } v \leq t. \end{array}$$

2.2 Bounds based on Linear Programming

We can often find better lower bounds on the size of orthogonal arrays by using Delsarte's linear programming bound; this is the main theme of this section.

While developing the bounds based on linear programming techniques, we will be using several standard results from coding theory without proof; the reader is referred to MacWilliams and Sloane [19] for background information on error-correcting codes.

An (n, M, d) binary error correcting code $\mathcal{C}$ is a set of M vectors of length n such that the Hamming distance between any two vectors in $\mathcal{C}$ is at least d. A code is said to be *linear* if the codewords form an m-dimensional subspace of $[GF(2)]^n$, where $M = 2^m$. Let A_i be the number of codewords having Hamming weight i. The sequence $(A_0, A_1, \ldots, A_n)$ is called the *weight distribution* of the code. The homogeneous polynomial

$$\sum_{i=0}^{n} A_i x^{n-i} y^i$$

is called the *weight enumerator* of $\mathcal{C}$ and is denoted by $W_{\mathcal{C}}(x, y)$. Clearly

$$A_0 + A_1 + \ldots + A_n = M.$$

The *distance distribution* of the code is defined to be the sequence $(B_0, B_1, \ldots, B_n)$, where

$$B_i = \frac{1}{M} |\{(u, v) : u, v \in \mathcal{C}, d(u, v) = i\}|.$$

Note that $B_0 = 1$ and $B_0 + B_1 + \ldots + B_n = M$. It is easy to observe that, for a linear code, the distance distribution is identical to the weight distribution.

If $\mathcal{C}$ is a linear code, the *dual code* $\mathcal{C}^{\perp}$ is defined to be the set of all vectors u having inner product zero with every codeword of $\mathcal{C}$. Let $A_i^{'}$ denote the number of codewords having weight i in $\mathcal{C}^{\perp}$. The weight enumerators of $\mathcal{C}$ and $\mathcal{C}^{\perp}$ are related by the MacWilliams identity shown below. For a proof of it see for example [19, p. 127].

$$W_{\mathcal{C}^{\perp}}(x,y) = \frac{1}{M} W_{\mathcal{C}}(x+y, x-y).$$

Equivalently,

$$\sum_{k=0}^{n} A_k{}' x^{n-k} y^k = \frac{1}{M} \sum_{i=0}^{n} A_i (x+y)^{n-i} (x-y)^i.$$

If we write

$$(x+y)^{n-i}(x-y)^i = \sum_{k=0}^{n} P_k(i) x^{n-k} y^k,$$

then we get

$$A_k^{'} = \frac{1}{M} \sum_{i=0}^{n} A_i P_k(i).$$

$P_k(i)$ is the value of the *Krawtchouk polynomial* $P_k(x)$ at integer i and can be explicitly described as

$$P_k(i) = \sum_{j=0}^{k} (-1)^j \binom{i}{j} \binom{n-i}{k-j}, k = 0, 1, 2, \ldots$$

In the case of non-linear codes, we cannot talk of the dual code $\mathcal{C}^{\perp}$ as the code $\mathcal{C}$ is not a vector space. However, we can still transform the distance distribution $(B_0, B_1, \ldots, B_n)$ of the code $\mathcal{C}$ in a similar way using the Krawtchouk polynomials to obtain the *dual distance distribution* $(B_0^{'}, B_1^{'}, \ldots, B_n^{'})$, which is defined as follows:

$$B_k^{'} = \frac{1}{M} \sum_{i=0}^{n} B_i P_k(i).$$

If $B_i^{'} = 0$ for $1 \leq i \leq d^{'} - 1$ and $B_{d^{'}}^{'} \neq 0$, then $d^{'}$ is called the *dual distance* of the code $\mathcal{C}$. This concept was defined by Delsarte [10]. If $\mathcal{C}$ is linear then $(B_0^{'}, B_1^{'}, \ldots, B_n^{'})$ is indeed the distance distribution as well as the weight distribution of the dual code $\mathcal{C}^{\perp}$, and $d^{'}$ is the minimum distance of $\mathcal{C}^{\perp}$. It turns out that even when $\mathcal{C}$ is not a linear code, the dual distance $d^{'}$ has a combinatorial significance. The following Theorem describes the combinatorial significance of the dual distance $d^{'}$ and the proof, due to Delsarte, can be found in [19, p. 139] and [10].

Theorem 2.3 *If we write the vectors in $\mathcal{C}$ as rows of an $M \times n$ array, then any set of $r \leq d^{'} - 1$ columns contains each r-tuple exactly $M/2^r$ times, and $d^{'}$ is the largest number with this property. In other words $\mathcal{C}$ is an orthogonal array $OA_{\lambda}(d^{'} - 1, n, 2)$ where $\lambda = M/2^{d^{'}-1}$.*

It is clear that for any code $\mathcal{C}$, we have $B_i \geq 0$ for $i = 0, 1, \ldots, n$. On the other hand, it is a non-trivial theorem (see for example [19, p. 139],[10]) that $B_i^{'} \geq 0$ for $i = 0, 1, \ldots, n$.

Suppose a simple $OA_\lambda(t, n, 2)$ exists. Let M be the number of rows in this orthogonal array. A lower bound on M can be obtained by solving a suitable linear programming problem, if we view the rows of this orthogonal array as codewords of a code $\mathcal{C}$ (see [19, §4 of Ch.17], [9] for similar approaches to a different problem). Let $(B_0, B_1, \ldots, B_n)$ be the distance distribution of $\mathcal{C}$ and let $(B_0^{'}, B_1^{'}, \ldots, B_n^{'})$ be its transform. Then

$$M = B_0 + B_1 + \ldots + B_n.$$

Also

$$\begin{aligned} B_k^{'} &= \frac{1}{M} \sum_{i=0}^{n} B_i P_k(i) \\ &= \frac{1}{M} \left(\sum_{i=1}^{n} B_i P_k(i) + \binom{n}{k} \right). \end{aligned}$$

Further

$$\sum_{k=0}^{n} B_k' = \frac{1}{M} \sum_{i=0}^{n} B_i \sum_{k=0}^{n} P_k(i) = \frac{1}{M} B_0 2^n = \frac{2^n}{M}.$$

Here we have used

$$\sum_{k=0}^{n} P_k(i) = \delta_{i,0} 2^n.$$

This is a property of the Krawtchouk polynomials and can be found in Section 5 as equation (7). Moreover the Krawtchouk transform is an involutorial operation on the set of real $(n+1)$-tuples with non-vanishing sum and constant term $= 1$. Hence, if we compute

$$B_k'' = \frac{M}{2^n} \left(\sum_{i=1}^{n} B_i' P_k(i) + \binom{n}{k} \right), \qquad k = 0, 1, \ldots, n,$$

then $B_k'' = B_k$.

In view of Theorem 2.3, the dual distance of $\mathcal{C}$ is at least $t + 1$ since it is an orthogonal array of strength t. Hence we have $B_k^{'} = 0$ for $1 \leq k \leq t$. So we formulate the following linear programming problem which we will refer to as **LP1**:

Maximize $w_{t+1} + w_{t+2} + \cdots + w_n$
subject to

$$\sum_{i=t+1}^{n} w_i P_k(i) \geq -\binom{n}{k} \text{ for } 0 \leq k \leq n$$

$w_i \geq 0$ for $t + 1 \leq i \leq n$

Let $W = W(n,t)$ be the optimal solution to the above linear programming problem. Then we have

$$\frac{2^n}{M} = \sum_{i=0}^{n} B'_i \leq 1 + W \tag{1}$$

Thus by solving a linear program we get a lower bound for the size of an orthogonal array.

3 Upper Bounds for Resilient Functions

In this section, we will be concerned with developing upper bounds for the optimum value of t for a given n and m. It is easy to see that $n \geq m + t$ and so we have the upper bound

$$t \leq n - m. \tag{2}$$

However, the above bound is usually very weak and better upper bounds can be obtained by exploring the relation between resilient functions and orthogonal arrays. These connections were explored and several results were proved in [24], [8] and [7]. We will briefly survey them and then use the results for our purposes.

The proof of the following theorem which elucidates the connection between resilient functions and orthogonal arrays can be found in [24].

Theorem 3.1 *An (n,m,t) - resilient function is equivalent to a large set of orthogonal arrays $LOA_{2^{n-m-t}}(t,n,2)$.*

In view of Theorem 3.1, any necessary condition for the existence of an orthogonal array $OA_{2^{n-m-t}}(t,n,2)$ is also a necessary condition for the existence of an (n,m,t) - resilient function. We obtain the following Corollary which gives a necessary condition for existence of an (n,m,t) - resilient function from the classical Rao bound for orthogonal arrays.

Corollary 3.2 *[8, 24] Suppose there exists an (n,m,t) - resilient function. Then*

$$m \leq n - \log_2 \left[\sum_{i=0}^{t/2} \binom{n}{i} \right]$$

if t is even; and

$$m \leq n - \log_2 \left[\binom{n-1}{(t-1)/2} + \sum_{i=0}^{(t-1)/2} \binom{n}{i} \right]$$

if t is odd.

Proof: Set $v = 2$, $k = n$ in Theorem 2.1 and apply Theorem 3.1. $\square$

Similarly from the classical Bush bound, we can obtain the following Corollary that was proved in [3] from first principles.

Corollary 3.3 *[3, 24] There exists an (n, m, t) - resilient function with $n = m + t$ if and only if $m = 1$ or $t = 1$.*

Proof: The cases $t = 1$ and $m = 1$ were given earlier in examples. So, suppose $n = m + t$ and $2 \leq t \leq n - 2$. Apply Theorem 2.2 with $v = 2$ to get $m + t \leq t + 1$, or $m \leq 1$, a contradiction. $\square$

The upper bounds based on the Rao bound for orthogonal arrays are stronger than the ones obtained using the trivial bound of equation (2). However, we can often do better by using Delsarte's linear programming bound. In view of Theorem 3.1, an (n, m, t) resilient function exists if only if $LOA_{2^{n-m-t}}(t, n, 2)$ exists. Clearly $LOA_{2^{n-m-t}}(t, n, 2)$ exists only if an $OA_{2^{n-m-t}}(t, n, 2)$ exists. The number of rows in this orthogonal array is given by

$$M = 2^{n-m-t}2^t = 2^{n-m}. \tag{3}$$

In view of the bound of inequality (1) of Section 2.2, this immediately implies that

$$m \leq \log_2(1 + W).$$

Thus we have an upper bound for the optimal value of m, given n and t. The upper bound for the optimal value of t for a given n and m can be trivially computed once we have a table of upper bounds for the optimal value of m. The upper bounds depicted in Table 2 are obtained in this manner using MAPLE V to do the necessary computation.

In a forthcoming book [15], Hedayat, Sloane and Stufken provide a compilation of a table of minimal possible index of orthogonal arrays with 2, 3 and 4 levels, k factors and strength t. Upper bounds for the optimal value of t can also be obtained from these tables.

We now have three upper bounds on the optimal value of t, viz., the trivial bound of equation (2), the bound based on the Rao bound for orthogonal arrays and the bound based on Delsarte's linear programming bound. In Table 1, a comparison of these three bounds for some values of n and m is provided. This table illustrates the relative strengths of the various bounds. The improvements provided by the stronger bounds are more marked for larger values of n.

4 Lower Bounds for Resilient Functions

Obviously, any method of construction of resilient functions yields a lower bound on the optimal value of t. The most important construction method for resilient

Table 1: A comparison of the bounds.

n	m	Trivial bound	Rao bound	L.P. bound
18	2	16	14	11
19	2	17	15	11
20	2	18	15	12
18	3	15	12	9
19	3	16	13	9
20	3	17	13	10
18	4	14	10	8
19	4	15	11	9
20	4	16	12	9

functions uses linear error correcting codes and these are discussed in Section 4.1. Recently, Stinson and Massey [25] constructed resilient functions from non-linear codes also. This construction is discussed in Section 4.2. Some constructions of new resilient functions from old are developed in Section 4.3. Finally, a table of bounds for resilient functions is presented in Section 4.4

4.1 Constructions using linear codes

The most basic and well-known construction method for resilient functions uses linear error-correcting codes and therefore we will explore the connection between t-resilient functions and error-correcting codes in this section. A resilient function constructed in this way is said to be a linear resilient function.

An (n, m, d) linear code is an m-dimensional subspace C of $[GF(2)]^n$ such that the Hamming distance between any two vectors in C is at least d. Let G be an $m \times n$ matrix whose rows form a basis for C; G is called a *generating matrix* for C. The following construction for resilient functions was given in [8, 3]:

Theorem 4.1 *Let G be a generating matrix for an (n, m, d) linear code C. Define a function $f : [GF(2)]^n \longrightarrow [GF(2)]^m$ by the rule $f(x) = xG^T$. Then f is an $(n, m, d-1)$ - resilient function.*

This result can easily be seen to be true using the orthogonal array characterization of Theorem 3.1 The inverse image $f^{-1}(0, 0, \ldots, 0)$ is in fact the dual code $C^{\perp}$. Any $d-1$ columns of the generating matrix for $C^{\perp}$ (= the parity check matrix for C) are linearly independent, since the minimum distance of C is d. This implies that any set of $d-1$ columns of $C^{\perp}$ contain each $d-1$ tuple exactly $|C^{\perp}|/2^{d-1} = 2^{n-m-d+1}$ times and hence it follows that $C^{\perp}$ is an orthogonal array $OA_{2^{n-m-d+1}}(d-1, n, 2)$.

Now, any other inverse image $f^{-1}(y)$ is an additive coset of $C^{\perp}$, and thus is also an $OA_{2^{n-m-d+1}}(d-1, n, 2)$. Hence we obtain 2^m OA's that form a large set. By Theorem 3.1, it follows that f is an $(n, m, d-1)$ - resilient function.

Thus, whenever a linear (n, m, d) linear code exists, so does an $(n, m, d-1)$ resilient function. Brouwer and Verhoeff [5] provide a compilation of best known linear binary codes. These provide lower bounds for the optimum value of t for a given n and m.

4.2 Constructions using non-linear codes

Often the lower bounds derived from the constructions based on linear codes do not match the upper bounds. In some such cases, constructions from non-linear codes might be of help.

An (n, M, d) code is said to be a *systematic code* if there exist m co-ordinates such that every possible m-tuple occurs in exactly one codeword within the m specified co-ordinates (of course, this implies $M = 2^m$). These m co-ordinates are called *information bits* and the remaining $n - m$ co-ordinates are called *parity-check bits*. Clearly any linear code is a systematic code. However, there are several well-known classes of non-linear codes (for example Preparata codes and Kerdock codes) which are systematic codes. The following theorem was proved by Stinson and Massey in [25].

Theorem 4.2 *If there exists a systematic (n, M, d) code, $\mathcal{C}$, having a dual distance $d^{'}$, then there is an $(n, n-m, d^{'}-1)$-resilient function, where $M = 2^m$.*

Note that this result subsumes Theorem 4.1 as any linear code is a systematic code.

It was conjectured in [8] and [3] that if there exists a resilient function with certain parameters, then there exists a linear function with the same parameters. However this conjecture was disproved by Stinson and Massey [25]. They constructed a class of non-linear resilient functions from the Kerdock codes; further there are no linear resilient functions with the same parameters. The proof of the following theorem can be found in [25].

Theorem 4.3 *For any odd integer $r \geq 3$, a $(2^{r+1}, 2^{r+1} - 2r - 2, 5)$- resilient function exists, but no linear resilient function with these parameters exists.*

The first member of the above infinite class will be of special interest to us. It is the $(16, 8, 5)$ resilient function constructed from the Kerdock code $\mathcal{K}(4)$, which is also known as the Nordstrom-Robinson code. This resilient function is optimal as the upper bound for t is 5 when $n = 16$ and $m = 8$.

4.3 New Resilient Functions from Old

We now proceed to describe some constructions of new resilient functions from old. When we apply these constructions to the $(16, 8, 5)$ resilient functions, we obtain new resilient functions which are optimal.

Theorem 4.4 *Suppose there is a linear (n, m, t)-resilient function. Then there exist $(n-1, m, t-1)$- and $(n-1, m-1, t)$-resilient functions.*

Proof: Since the (n, m, t)-resilient function is linear, it must have been derived from an $(n, m, t+1)$ linear code $\mathcal{C}$. The *punctured code* is obtained by deleting a fixed coordinate from every codeword in $\mathcal{C}$, thus producing a linear $(n-1, m, t)$ code. From this punctured code, we obtain an $(n-1, m, t-1)$-resilient function.

If we choose a co-ordinate, say i, of $\mathcal{C}$ which is not always zero, then there will be 2^{m-1} codewords in $\mathcal{C}$ with a "0" in the ith co-ordinate, and 2^{m-1} codewords with a "1" in the ith co-ordinate. If we delete the codewords with a "1" in the ith co-ordinate, and then delete the ith co-ordinate, then we obtain the *shortened code*, which is an $(n-1, m-1, t+1)$ linear code. From this shortened code, we obtain an $(n-1, m-1, t)$-resilient function. □

Unfortunately, Theorem 4.4 is not of much use for us as the only $(16, 8, 5)$-resilient functions are non-linear. Let us see if we can generalize these constructions to the case of an arbitrary (non-linear) resilient function. In one case, we can always do so.

Theorem 4.5 *Suppose there is an (n, m, t)-resilient function. Then there is an $(n-1, m, t-1)$-resilient function.*

Proof: Let $f : \{0,1\}^n \longrightarrow \{0,1\}^m$ be an (n, m, t)-resilient function. Define $g : \{0,1\}^{n-1} \longrightarrow \{0,1\}^m$ by the rule $g(x_1, \ldots, x_{n-1}) = f(x_1, \ldots, x_{n-1}, 0)$. It is easy to see that g is $(t-1)$-resilient. □

Let us recast this procedure in terms of large sets of orthogonal arrays. Suppose A is an $OA_\lambda(t, n, v)$. Let x be a symbol. Delete all rows in which x does not occur in the last column, and then delete the last column. In this way we obtain an $OA_\lambda(t-1, n-1, v)$ (this operation is called "derivation" [26]). If we have a large set of OA's, then the set of derived arrays also forms a large set. Hence, we obtain the following general result (which holds for arbitrary "v"):

Theorem 4.6 *Suppose there is a large set of $OA_\lambda(t, n, v)$. Then there is a large set of $OA_\lambda(t-1, n-1, v)$.*

On the other hand, it does not appear that the existence of an (n, m, t)-resilient function automatically implies the existence of an $(n-1, m-1, t)$-resilient function. Again, it is useful to think of this in terms of orthogonal arrays. Suppose A is an $OA_\lambda(t, n, v)$, and we delete the last column. Then it is clear that we get an

$OA_\lambda(t, n-1, v)$ (this operation is called "restriction" [26]). Unfortunately, the set of restricted arrays of a large set of OA's is *never* a large set, since every possible row occurs v times in the set of restricted arrays.

However, in certain special cases we can obtain an $(n-1, m-1, t)$- resilient function from an (n, m, t)-resilient function. One such case is when the original (n, m, t)-resilient function is obtained from a systematic code.

Theorem 4.7 *Suppose there is an (n, m, t)-resilient function derived from a systematic code. Then there is an $(n-1, m-1, t)$-resilient function (which is also derived from a systematic code).*

Proof: Let $\mathcal{C}$ be the $(n, 2^{n-m}, d)$ systematic code from which the (n, m, t)-resilient function is derived. Then the dual distance of the code is $t+1$. If we choose a parity-check co-ordinate, say i, of $\mathcal{C}$ and delete the ith coordinate from every codeword in $\mathcal{C}$, then we obtain the punctured code, which is an $(n-1, 2^{n-m}, d-1)$ code. It is easy to see that the punctured code is also a systematic code. The deletion of a column of an orthogonal array does not change its strength and so the dual distance of the punctured code is still $t+1$. Hence we can obtain an $(n-1, m-1, t)$-resilient function from the punctured code. $\square$

A $(16, 8, 5)$-resilient function exists by virtue of Theorem 4.3. Applying Theorem 4.7 repeatedly, starting with the $(16, 8, 5)$- resilient function, we get resilient functions with parameters $(15, 7, 5), (14, 6, 5)$ and $(13, 5, 5)$. Applying Theorem 4.5 to the $(16, 8, 5)$-resilient function, we get a $(15, 8, 4)$-resilient function. Applying Theorem 4.7 repeatedly, starting with the $(15, 8, 4)$- resilient function, we get resilient functions with parameters $(14, 7, 4), (13, 6, 4)$ and $(12, 5, 4)$. All these resilient functions are optimal as they match the upper bounds derived from linear programming bound.

4.4 A Table of Bounds

In Table 2 upper and lower bounds for the optimal value of t when $1 \leq n \leq 25$ and $1 \leq m < n$ are given. The rows are indexed by values of n and the columns are indexed by values of m. Suppose that the ordered pair (u, l) is the entry in the mth column of the nth row. This indicates that there is no $(n, m, u+1)$ resilient binary function (Specifically, such a function is ruled out by the linear programming bound of section 2.2). It also means that there exists an (n, m, l) resilient binary function. Specifically, there exists an $(n, m, l+1)$ binary linear code from which we can obtain the (n, m, l) resilient binary function unless the entry is marked with a *. The (n, m, l) resilient functions corresponding to the entries marked with a * are obtained from constructions using non-linear codes discussed in this section. If there is a single entry in the mth column of the nth row, then it indicates that $u = l$. In other words, the entry is the optimal value of t. It is clear from the table that the optimal value of t is determined exactly for all cases where $1 \leq m < n \leq 17$.

5 Properties of Krawtchouk Polynomials

The Krawtchouk polynomials were introduced and defined in Section 2.2. In the rest of the paper, we will be using several properties of Krawtchouk polynomials extensively and so we list them together in this section. This is not intended to be an exhaustive list of their properties; we list only those properties which we need in the sequel. All these properties can be found either as theorems or as problems in Section 7 of Chapter 5 in [19].

Recall that for any positive integer n, the Krawtchouk polynomial $P_k^n(x) = P_k(x)$ is defined by

$$P_k(x) = \sum_{j=0}^{k} (-1)^j \binom{x}{j} \binom{n-x}{k-j}, k = 0, 1, 2, \ldots$$

The first few Krawtchouk polynomials are shown below.

$$\begin{aligned} P_0(x) &= 1 \\ P_1(x) &= n - 2x \\ P_2(x) &= \binom{n}{2} - 2nx + 2x^2 \end{aligned}$$

The following two simple properties deal with the values of Krawtchouk polynomials under certain special situations.

$$P_k(0) = \binom{n}{k} \tag{4}$$

$$P_0(i) = 1 \tag{5}$$

From the definition of the Krawtchouk polynomials, it immediately follows that

$$\sum_{i=0}^{n} P_i(k) z^i = (1+z)^{n-k}(1-z)^k. \tag{6}$$

Setting $z = 1$ in the identity (6) gives

$$\sum_{i=0}^{n} P_i(k) = 2^n \delta_{k,0}, \tag{7}$$

where $\delta_{r,s}$ is the Kronecker symbol defined by $\delta_{r,s} = 1$ if $r = s$ and $\delta_{r,s} = 0$ if $r \neq s$.

There is also some sort of symmetry among the values of the Krawtchouk polynomial which are captured in the following identities.

$$P_k(i) = (-1)^k P_k(n-i) \tag{8}$$

$$P_k(i) = (-1)^i P_{n-k}(i) \tag{9}$$

Actually identity (9) can be inferred from identity (8) using the following more general identity which we refer to as the inversion identity.

$$\binom{n}{i} P_s(i) = \binom{n}{s} P_i(s) \tag{10}$$

Finally, we will need the following simple identity which follows at once from the identity (6):

$$P_k^{n+1}(i) = P_k^n(i) + P_{k-1}^n(i). \tag{11}$$

6 Self-Complementary Orthogonal Arrays

In this section we shall consider the following Theorem of Levenshtein about orthogonal arrays and discuss its implications to resilient functions. For the sake of completeness we include a proof of the Theorem which is similar to Levenshtein's proof [17, Lemma 1].

Theorem 6.1 *Let t be even. Then $OA_\lambda(t, n, 2)$ exists if and only if $OA_\lambda(t+1, n+1, 2)$ exists and hence the minimum size $N(n, t)$ of an OA of strength t and length n satisfies the equality*

$$2N(n, t) = N(n + 1, t + 1).$$

Proof: We will first show that an $OA_\lambda(t + 1, n + 1, 2)$ exists if an $OA_\lambda(t, n, 2)$ exists. Suppose an $OA_\lambda(t, n, 2)$ exists. Let A be such an OA. Let M denote the number of rows in A. Let P be an array formed by adding a column of zeroes to the array A. Now define the array B to consist of all the rows of the array P and their complements. More formally, a binary vector x of length $n + 1$ is a row of B if and only if either x or $1 + x$ is a row of P, where 1 is the all one's vector of length $n + 1$. We shall prove that the array B is indeed an $OA_\lambda(t + 1, n + 1, 2)$.

Let $(A_0, A_1, \ldots, A_n)$ be the distance distribution of the OA A considered as a code and similarly let $(B_0, B_1, \ldots, B_n, B_{n+1})$ be the distance distribution of the OA B. As we saw earlier, it is always the case that $B_0 = 1$. Also

$$\begin{aligned} B_{n+1} &= \frac{1}{2M}|\{(u, v) : u, v \in B, d(u, v) = n + 1\}|. \\ &= 1, \end{aligned}$$

since for every row u of B, there is exactly one row v of B such that $d(u, v) = n + 1$ (v is simply the complement of u), and B has $2M$ rows. Let us now express B_i in terms of the A_i's. By definition

$$B_i = \frac{1}{2M}|\{(u, v) : u, v \in B, d(u, v) = i\}|.$$

The number of ordered pairs (u, v) such that $d(u, v) = i$ and both u and v are rows of P is simply $A_i M$ since the last element is zero for all the rows of P. We will denote by P^c the complements of the rows of P. As P^c is merely a translate of P, the number of ordered pairs (u, v) such that $d(u, v) = i$ and both u and v are rows of P^c is once again $A_i M$.

Now suppose $u \in P$ and $v \in P^c$. Then clearly $1 + v \in P$ and $d(u, v) = i$ if and only if $d(u, 1 + v) = n + 1 - i$. So it follows that the number of ordered pairs (u, v) such that $u \in P$, $v \in P^c$ and $d(u, v) = i$ is simply $A_{n+1-i}M$. By symmetry we also have that the number of ordered pairs (u, v) such that $u \in P^c$, $v \in P$ and $d(u, v) = i$ is also $A_{n+1-i}M$.

Altogether, the number of ordered pairs (u, v) such that $d(u, v) = i$ and both u and v are rows of B is simply $2M(A_i + A_{n+1-i})$ and so by definition, we have for $1 \leq i \leq n$

$$B_i = A_i + A_{n+1-i}.$$

Thus the distance distribution of B is completely determined by that of A.

Let $(A_0^{'}, A_1^{'}, \dots A_n^{'})$ be the transform of the distance distribution of A and let $(B_0^{'}, B_1^{'}, \dots B_n^{'})$ be the transform of the distance distribution of B. It is always the case that $A_0^{'} = B_0^{'} = 1$. As A is an orthogonal array of strength t, we also have that $A_i^{'} = 0$ for $1 \leq i \leq t$ by Theorem 2.3. We now compute $B_k^{'}$ in terms of the values of $A_i^{'}$ using the properties of the Krawtchouk polynomials developed in Section 5. The computation proceeds as follows:

$$\begin{aligned}
B_k^{'} &= \sum_{i=0}^{n+1} B_i P_k^{n+1}(i) \\
&= B_0 P_k^{n+1}(0) + B_{n+1} P_k^{n+1}(n+1) + \sum_{i=1}^{n} B_i P_k^{n+1}(i) \\
&= \binom{n+1}{k}[1 + (-1)^k] + \sum_{i=1}^{n} (A_i + A_{n+1-i}) P_k^{n+1}(i) \\
&\quad \text{using identities (4) and (8)} \\
&= \binom{n+1}{k}[1 + (-1)^k] + \sum_{i=1}^{n} A_i P_k^{n+1}(i) + \sum_{i=1}^{n} A_{n+1-i} P_k^{n+1}(i) \\
&= \binom{n+1}{k}[1 + (-1)^k] + \sum_{i=1}^{n} A_i P_k^{n+1}(i) + (-1)^k \sum_{i=1}^{n} A_{n+1-i} P_k^{n+1}(n+1-i) \\
&\quad \text{using identity (8)} \\
&= \binom{n+1}{k}[1 + (-1)^k] + \sum_{i=1}^{n} A_i P_k^{n+1}(i) + (-1)^k \sum_{i=1}^{n} A_i P_k^{n+1}(i) \\
&= [1 + (-1)^k] \left(\binom{n+1}{k} + \sum_{i=1}^{n} A_i P_k^{n+1}(i) \right)
\end{aligned}$$

$$
\begin{aligned}
&= [1+(-1)^k]\sum_{i=0}^{n} A_i P_k^{n+1}(i) \\
&\quad \text{using identity (4)} \\
&= [1+(-1)^k]\left(\sum_{i=0}^{n} A_i P_k^{n}(i) + \sum_{i=0}^{n} A_i P_{k-1}^{n}(i)\right) \\
&\quad \text{using identity (11)} \\
&= [1+(-1)^k](A_k^{'} + A_{k-1}^{'})
\end{aligned}
$$

As A is an orthogonal array of strength t, its dual distance is $t+1$ and so $A_i^{'} = 0$ for $1 \le i \le t$. This fact implies, in view of the above expression for $B_k^{'}$, that $B_k^{'} = 0$ for $2 \le k \le t$. Also $B_1^{'} = 0$, as $[1+(-1)^k] = 0$ when $k = 1$. Finally $B_{t+1}^{'} = 0$, as $[1+(-1)^{t+1}] = 0$ when t is even (which is a hypothesis in the Theorem). Thus $B_k^{'} = 0$ for $1 \le k \le t+1$ and hence it follows that the strength of B as an orthogonal array is at least $t+1$ by Theorem 2.3. Thus when t is even, $OA_\lambda(t+1, n+1, 2)$ exists if $OA_\lambda(t, n, 2)$ exists.

The fact that $OA_\lambda(t, n, 2)$ exists if $OA_\lambda(t+1, n+1, 2)$ is easy and follows immediately from Theorem 4.6. Note that we do not need the assumption that t is even for this part. A simple consequence is that the minimum size $N(n, t)$ of an OA of strength t and length n satisfies the equality

$$2N(n, t) = N(n+1, t+1).$$

□

The above constructions easily extend to large sets of orthogonal arrays as well and hence an equivalent statement can be made about resilient functions.

Theorem 6.2 *Let t be even. Then an (n, m, t)-resilient function exists if and only if an $(n+1, m, t+1)$-resilient function exists. Hence, if the optimal resiliency t is even for parameters (n, m), then the optimal resiliency is $t+1$ for parameters $(n+1, m)$.*

Note further that an (n, m, t)-resilient function always exists if an $(n+1, m, t+1)$-resilient function exists as stated in Theorem 4.5, i.e. the assumption that t is even is not required.

Using Theorems 4.6 and 6.2, we can make the following comments about Table 2. For each fixed m, the optimal resiliency increases by at most 1 when n is increased by 1. So all the integers appear in each column in their natural order. An integer may appear several times in a column, but any even integer can appear exactly once in each column as dictated by Theorem 6.2. Stated in another way, for each m the probability that the optimal resiliency t is odd for a randomly picked n is at least half.

7 Explicit Bounds for Orthogonal Arrays and Resilient Functions

The orthogonal array bounds obtained by the linear programming technique are usually much better than the Rao bound. The disadvantage of the linear programming bound is that one needs to solve a different linear program for every parameter situation. Of course, this can be done in polynomial time by using Karmarkar's algorithm among others. Nonetheless, the linear programming bound is fairly difficult to compute as compared to the Rao bound. Thus it is of interest to try to derive explicit bounds as corollaries of the linear programming bound. We will pursue this idea in this section.

Let us first form the dual of the linear programming problem **LP1**. We will refer to the dual as **DLP1**.

Minimize $\sum_{k=1}^{n} x_k \binom{n}{k}$
subject to
$\sum_{k=1}^{n} x_k P_k(i) \leq -1$ for $i = t+1$ to n
$x_1, x_2, \ldots x_n \geq 0$

It is a standard theorem in the theory of linear programming (see for example [20]) that in a pair of primal-dual linear programs the optimal value of the maximization problem will always be greater than or equal to the value attained by the objective function of the minimization problem at any feasible solution vector. So any feasible solution to the dual linear program **DLP1** would yield a lower bound on the size of the orthogonal array of strength t and length n and consequently an upper bound on m of t-resilient functions on n-tuples. In fact, similar techniques were previously used by Delsarte to obtain explicit bounds on the size of a code with a given minimum distance [19, §4 of Ch.17].

Assume $2(t+1) > n$. Consider the solution vector

$$x_k = \begin{cases} \frac{1}{2(t+1)-n} & \text{if } k = 1 \\ 0 & \text{otherwise} \end{cases}$$

The non-negativity conditions are certainly satisfied. As $P_1(x) = n - 2x$, we have

$$\sum_{k=1}^{n} x_k P_k(i) = x_1(n - 2i) \leq -1$$

for $i \geq t+1$. So this is indeed a feasible solution. The value of the objective function at this solution vector is

$$x_1 n = \frac{n}{2(t+1)-n}.$$

Hence it follows that

$$\frac{2^n}{M} \leq 1 + W \leq 1 + \frac{n}{2(t+1)-n} = \frac{2(t+1)}{2(t+1)-n} \tag{12}$$

As a consequence, we get the following Theorem, which was first proved by Friedman [11] using very different methods.

Theorem 7.1 *Suppose an $OA_\lambda(t, n, 2)$ exists. Let $M = \lambda 2^t$ be the number of rows in the OA. Then*

$$M \geq 2^n - \frac{n 2^{n-1}}{t+1}. \tag{13}$$

Note: The assumption that $2(t+1) > n$ is not needed for the final statement of the Theorem as otherwise the right hand side of the above equation is not positve and hence the theorem will be trivially true.

This Theorem along with equation (3) immediately yields an explicit bound on resilient functions as well. We state this bound as the following Corollary.

Corollary 7.2 *If an (n, m, t)-resilient function exists, then*

$$t \leq \left\lfloor \frac{2^{m-1} n}{2^m - 1} \right\rfloor - 1.$$

Now we derive a second bound. Assume $2t > n-3$, t even. Consider the solution vector

$$x_k = \begin{cases} \frac{1}{2t+3-n} & \text{if } k = 1 \text{ or } n \\ 0 & \text{otherwise} \end{cases}$$

Again the nonnegativity is obvious. The main condition is

$$\frac{1}{2t+3-n}(P_1(i) + P_n(i)) = \frac{n - 2i + (-1)^i}{2t+3-n} \leq -1$$

for $i \geq t+1$. This is trivially satisfied for $i > t+1$. For $i = t+1$ it is satisfied with equality as $i = t+1$ is odd. The value attained by the objective function at this feasible solution is

$$x_1 \binom{n}{1} + x_n \binom{n}{n} = \frac{n+1}{2t+3-n}.$$

Hence it follows that

$$\frac{2^n}{M} \leq 1 + W \leq \frac{2(t+2)}{2t+3-n} \tag{14}$$

As a consequence, we get the following Theorem.

Theorem 7.3 *Suppose an $OA_\lambda(t,n,2)$ exists, where t is even. Let $M=\lambda 2^t$ be the number of rows in the OA. Then*

$$M \geq 2^n - \frac{2^{n-1}(n+1)}{t+2}. \tag{15}$$

Note: The assumption that $2t > n-3$ is not needed for the final statement of the theorem as otherwise the right hand side of the above equation is not positive and so the theorem will be trivially true.

The bound of Theorem 7.3 is exactly half of the bound given by Theorem 7.1 if (n,t) is replaced by $(n+1,t+1)$. So when t is even, the above bound is also implied by Theorem 7.1 in light of Theorem 6.1. This observation is made by Levenshtein (private communication).

We have the following Corollary obtained by combining Theorem 7.1 and Theorem 7.3.

Corollary 7.4 *Suppose an $OA_\lambda(t,n,2)$ exists. Let $M=\lambda 2^t$ be the number of rows in the OA. Then*

$$M \geq 2^n - \frac{2^{n-2}(n+1)}{\lceil \frac{t+1}{2} \rceil}. \tag{16}$$

Proof: Note that when t is odd,

$$2^n - \frac{2^{n-2}(n+1)}{\lceil \frac{t+1}{2} \rceil} = 2^n - \frac{2^{n-1}(n+1)}{t+1}.$$

The above bound is inferior to the bound given by Theorem 7.1 and hence is valid. On the other hand, when t is even

$$2^n - \frac{2^{n-2}(n+1)}{\lceil \frac{t+1}{2} \rceil} \quad = \quad 2^n - \frac{2^{n-1}(n+1)}{t+2}.$$

This bound is valid in view of Theorem 7.3.

Corollary 7.4, together with equation (3), yields a second explicit bound on resilient functions. We state this bound as the following Corollary.

Corollary 7.5 *If an (n,m,t)-resilient function exists, then*

$$t \leq 2\left\lfloor \frac{2^{m-2}(n+1)}{2^m-1} \right\rfloor - 1.$$

Several remarks are now in order. The statement in Corollary 7.2 was conjectured in [8]. It was shown to be true for $m=1,2$ (and in these cases, the bound is tight). The conjecture was also proved for arbitrary m in the special case of linear resilient functions, i.e. one where every output bit is a linear function of the input bits.

Corollary 7.2 establishes that the conjecture is true for arbitrary m and for arbitrary resilient functions.

As stated earlier, Theorem 7.1 and Corollary 7.2 were previously proved by Friedman [11] in a different setting. His proof involved a study of colorings of the n-dimensional boolean cube and can be interpreted as bounds for orthogonal arrays as well. As shown in this section, these bounds can be derived in a straightforward way from the well-known linear programming bounds of coding theory.

Further the classical Rao bound can also be obtained as the objective function of an appropriate feasible solution to the dual linear programming problem **DLP1**. Thus the linear programming technique gives a unified approach to prove the known general bounds on orthogonal arrays and resilient functions as well as new better bounds.

8 The power of explicit bounds

It is clear that the explicit bounds of inequalities (13) and (15) are trivial to compute. One might ask if these explicit bounds are weaker than the bounds computed by the linear programming technique. It turns out that for many parametric situations, these two bounds are as powerful as the linear programming bound itself. Levenshtein calls a bound of this type "universal" and studies universal bounds for codes in [16]. We will pursue this issue in this section.

Theorem 8.1 *When t is odd and $t+1 \leq n \leq 2t+1$, the bound of equation (13) is as powerful as the linear programming bound.*

Proof: It has already been shown in equation (12) that

$$1 + W \leq \frac{2(t+1)}{2(t+1)-n}.$$

If we can also show that

$$1 + W \geq \frac{2(t+1)}{2(t+1)-n}$$

when t is odd and $t+1 \leq n \leq 2t+1$, then the proof is completed. We can indeed do so by exhibiting a feasible solution for the primal linear programming problem. Observe first that because of equation (10) the main condition in **LP1** may equivalently be written as follows:

$$\sum_{i=t+1}^{n} \frac{w_i P_i(k)}{\binom{n}{i}} \geq -1 \text{ for } 0 \leq k \leq n.$$

Consider the solution vector given by

$$w_i = \begin{cases} \frac{n}{2(t+1)-n} & \text{if } i = t+1 \\ 0 & \text{otherwise} \end{cases}$$

As we are assuming $2(t+1) > n$, the nonnegativity is certainly satisfied. The main condition has been reformulated above. It says

$$w_{t+1} P_{t+1}(k) \geq -\binom{n}{t+1}$$

or equivalently

$$P_{t+1}(k) \geq \binom{n}{t+1} \left[1 - \frac{2(t+1)}{n}\right].$$

That this is true is the content of the following Lemma:

Lemma 8.2 *When t is odd and $t+1 \leq n \leq 2t+1, 0 \leq i \leq n$ it is the case that*

$$P_{t+1}(i) \geq \binom{n}{t+1} \left[1 - \frac{2(t+1)}{n}\right].$$

Clearly the value of the objective function is as claimed. Although this Lemma can be viewed as yet another property of the values of Krawtchouk polynomials at certain points, it does not seem to be known before. The proof of this Lemma is fairly technical and is not relevant to the main theme of this paper and hence is omitted here. However, the proof is made available in APPENDIX A for the interested reader.

Note that we are using the assumption that t is odd only in the proof of this Lemma and not anywhere else.

A similar statement is made about the bound of equation (15) in the following theorem.

Theorem 8.3 *When t is even and $t+1 \leq n \leq 2t+2$, the bound of equation (15) is as powerful as the linear programming bound.*

Proof: The proof will be along the same lines as that of Theorem 8.1. It has already been shown in equation (14) that

$$1 + W \leq \frac{2(t+2)}{2t+3-n}$$

If we can also show the reverse inequality when t is even and $t+1 \leq n \leq 2t+2$, then the proof is completed. Once again, we can do so by exhibiting a feasible solution for the primal linear programming problem. Consider the solution vector given by

$$w_i = \begin{cases} \frac{t+2}{2t+3-n} & \text{if } i = t+1 \\ \frac{n-t-1}{2t+3-n} & \text{if } i = t+2 \\ 0 & \text{otherwise} \end{cases}$$

Again nonnegativity is obvious by our assumptions. The value of the objective function is just right as

$$1 + w_{t+1} + w_{t+2} = 1 + \frac{n+1}{2t+3-n} = \frac{2(t+2)}{2t+3-n}.$$

We have to check the main condition of the dual problem, in the reformulation given above. As

$$\frac{w_{t+1}}{\binom{n}{t+1}} = \frac{w_{t+2}}{\binom{n}{t+2}} = \frac{t+2}{(2t+3-n)\binom{n}{t+1}},$$

this condition is

$$\frac{t+2}{(2t+3-n)\binom{n}{t+1}}(P_{t+1}(k) + P_{t+2}(k)) \geq -1.$$

As $P_{t+1}(k) + P_{t+2}(k) = P_{t+2}^{n+1}(k)$ by equation (11) and

$$\binom{n}{t+1} = \binom{n+1}{t+2}\frac{t+2}{n+1}$$

this is equivalent to

$$P_{t+2}^{n+1}(k) \geq \binom{n+1}{t+2}\left[1 - \frac{2(t+2)}{n+1}\right].$$

This is just another instance of Lemma 8.2 proved in Appendix A.

We end this Section with the following Corollary which is a simple consequence of Theorem 8.3.

Corollary 8.4 *When t is even and $n = t+1$, the bound given by equation (13) is the same as the one given by (15) and hence is also as powerful as the linear programming bound.*

9 Optimal Resilient Functions for fixed m

Let us now focus on the special cases where m is a small fixed integer. When $m = 1$, the trivial upper bound of equation (2) says that $t \leq n-1$. The function $f : \{0,1\}^n \longrightarrow \{0,1\}$ defined by the rule $f(x_1, \ldots, x_n) = x_1 + \cdots + x_n$ is clearly an $(n, 1, n-1)$-resilient function. Thus, the optimal value of t is indeed $n-1$, when $m = 1$ (see [8]). When $m = 2$, it was proved in [8] that an (n, m, t) - resilient function exists if and only if $t < \lfloor \frac{2n}{3} \rfloor$. The corresponding question for $m = 3$ or higher remained open. As a consequence of our new bounds, we have completed the determination of the optimal resiliency of resilient functions with $m = 3$, and we have also done most of the cases for $m = 4$.

The *simplex code* $\mathcal{S}_m$ is the dual of the Hamming code $\mathcal{H}_m$. The generator matrix of $\mathcal{S}_m$ is of size $m \times 2^m - 1$ and has all non-zero binary m-tuples as its columns. It is not too difficult to see that each non-zero codeword of the simplex code has weight 2^{m-1}. Thus $\mathcal{S}_m$ is a linear constant-weight code with parameters $(2^m - 1, m, 2^{m-1})$. We shall make use of the simplex code in the proof of the following theorem.

Theorem 9.1 *Suppose there exists an (n, m, t) linear resilient function which is optimal by virtue of meeting one of the two bounds of Corollary 7.2 and Corollary 7.5. Then there exists an $(n + 2^m - 1, m, t + 2^{m-1})$ linear resilient function which is also optimal by virtue of meeting the same bound.*

Proof: Since the (n, m, t) resilient function is linear, it must have been constructed from an $(n, m, t+1)$ linear code $\mathcal{C}$. Now consider the linear code $\mathcal{C}'$ whose generating matrix is obtained by pasting the generating matrices of the code $\mathcal{C}$ and the simplex code $\mathcal{S}_m$, i.e.,

$$G_{\mathcal{C}'} = [G_{\mathcal{C}} \mid G_{\mathcal{S}_m}]$$

Clearly $\mathcal{C}'$ is a linear $(n + 2^m - 1, m, t + 1 + 2^{m-1})$ code and from this code we can construct a linear resilient function with parameters $(n + 2^m - 1, m, t + 2^{m-1})$.

By assumption the (n, m, t) resilient function is optimal as the parameters meet one of the two bounds derived in section 7. If we increase n by $2^m - 1$, both the bounds go up by 2^{m-1} and so it immediately follows that the linear resilient function with parameters $(n + 2^m - 1, m, t + 2^{m-1})$ is also optimal by virtue of meeting the same bound. $\square$

The simplex code itself yields a $(2^m - 1, m, 2^{m-1} - 1)$ linear resilient function, which meets the bound of Theorem 7.2 and hence is optimal. Applying Theorem 9.1, we see that optimal linear resilient functions exist whenever $n \equiv 0 \bmod (2^m - 1)$, a result first shown by Friedman [11].

Let us now consider resilient functions with $m = 3$. The bound of Corollary 7.2 gives $t \leq \lfloor 4n/7 \rfloor - 1$ and the bound of Corollary 7.5 gives $t \leq 2\lfloor (2n+2)/7 \rfloor - 1$. The two bounds are tabulated in Table 3.

We see from the Table that Corollary 7.2 gives the strongest bound for $n \equiv 3, 6 \bmod 7$, Corollary 7.5 gives the strongest bound for $n \equiv 2 \bmod 7$, and the two bounds are equal for $n \equiv 0, 1, 4, 5 \bmod 7$.

Of course these are only upper bounds on t. But matching lower bounds could be obtained as follows. In view of Theorem 4.1, whenever a linear (n, m, d) code exists, so does an $(n, m, d - 1)$ resilient function. When $m = 3$ and $4 \leq n \leq 10$, it so happens that the linear resilient functions constructed from the best known linear codes are optimal by virtue of meeting one of the two bounds of Corollary 7.2 and Corollary 7.5. In fact the bound of Corollary 7.2 is always met except when $n = 9$, in which case the bound of Corollary 7.5 is met. This fact coupled with Theorem 9.1 completely determines the optimal resiliency of resilient functions with $m = 3$. We record the result as the following theorem.

Table 3: A comparison of the two explicit bounds when $m = 3$

n	$\lfloor 4n/7 \rfloor - 1$	$2\lfloor (2n+2)/7 \rfloor - 1$
$7h$	$4h-1$	$4h-1$
$7h+1$	$4h-1$	$4h-1$
$7h+2$	$4h$	$4h-1$
$7h+3$	$4h$	$4h+1$
$7h+4$	$4h+1$	$4h+1$
$7h+5$	$4h+1$	$4h+1$
$7h+6$	$4h+2$	$4h+3$

Theorem 9.2 *The optimal resiliency t of resilient functions with $m = 3$ is*

$$\left\lfloor \frac{4n}{7} \right\rfloor - 1 \text{ if } n \not\equiv 2 \bmod 7 \text{ and}$$

$$\left\lfloor \frac{4n}{7} \right\rfloor - 2 \text{ if } n \equiv 2 \bmod 7.$$

Similar analysis settles the question for $m = 4$ in all cases except for two congruence classes modulo 15, as described in the following theorem.

Theorem 9.3 *The optimal resiliency t of resilient functions with $m = 4$ is*

$$\left\lfloor \frac{8n}{15} \right\rfloor - 1 \text{ if } n \not\equiv 2, 3, 4, 6 \text{ or } 10 \bmod 15 \text{ and}$$

$$\left\lfloor \frac{8n}{15} \right\rfloor - 2 \text{ if } n \equiv 2, 6 \text{ or } 10 \bmod 15.$$

Here, it turns out that the bound of Corollary 7.2 is met in those cases when $n \not\equiv 2, 3, 4, 6, 10 \bmod 15$, and the bound of Corollary 7.5 is relevant for $n \equiv 2, 6, 10 \bmod 15$. The two congruence classes $n \equiv 3, 4 \bmod 15$ are unsolved at present.

9.1 Optimal resilient functions from anticodes

It is also possible to determine other infinite classes of optimal resilient functions, by using the method of *anticodes* [19, Ch. 17, §6] and we do so in the following.

A well-known technique for constructing good linear codes is to take several copies of the generator matrix of the simplex code $\mathcal{S}_m$ and delete certain columns from it. The columns to be deleted constitute the generator matrix of an *anticode*. An anticode is a code which has an *upper bound* on the distance between its codewords. Note that an anticode may contain repeated codewords as even a distance of zero between the codewords is allowed.

Consider an anticode whose generator matrix is of size $m \times k$ and whose *maximum distance* is δ. Suppose s is the maximum number of times any column occurs in the generating matrix of the anticode. Then we can form a new code whose generator matrix can be obtained by deleting the columns of the generator matrix of the anticode from s copies of the generator matrix of $\mathcal{S}_m$ placed side-by-side. The new code has length $s(2^m - 1) - k$, dimension $\leq m$, and minimum distance $s2^{m-1} - \delta$.

Using the above general technique and anticodes formed from subspaces Solomon and Stiffler [23, 1] constructed a large class of good linear codes. Let V_m denote a m-dimensional vector space over GF(2). Let $B_1, B_2, \ldots, B_p$ be subspaces of V_m of dimensions $u_1, u_2, \ldots, u_p$ such that no non-zero element of V_m is contained in more than s of the B_i. Let $m > u_1 \geq u_2 \geq \ldots \geq u_p \geq 1$. Then they showed that there exists an (n, m, d) binary linear code, where

$$\begin{aligned} n &= s(2^m - 1) - \sum_{i=1}^{p}(2^{u_i} - 1), \\ d &\geq s2^{m-1} - \sum_{i=1}^{p} 2^{u_i - 1}. \end{aligned}$$

Belov *et al* [1] showed that a necessary and sufficient condition for the existence of such subspaces is

$$\sum_{i=1}^{\min(s+1,p)} u_i \leq sm. \tag{17}$$

We now proceed to determine the parametric situations where the resilient functions constructed from the linear codes obtained by the above construction meet the bound of Corollary 7.2 or Corollary 7.5 and so are optimal.

The resilient functions so constructed will meet the bound of Corollary 7.2 if and only if

$$\begin{aligned} s2^{m-1} - \sum_{i=1}^{p} 2^{u_i - 1} &= \left\lfloor \frac{2^{m-1} n}{2^m - 1} \right\rfloor \\ &= \left\lfloor \frac{2^{m-1}\left[s(2^m - 1) - \sum_{i=1}^{p}(2^{u_i} - 1)\right]}{2^m - 1} \right\rfloor \\ &= \left\lfloor s2^{m-1} - \frac{2^{m-1}\sum_{i=1}^{p}(2^{u_i} - 1)}{2^m - 1} \right\rfloor \end{aligned}$$

$$= \quad s2^{m-1} - \left\lceil \frac{2^{m-1}\sum_{i=1}^{p}(2^{u_i}-1)}{2^m-1} \right\rceil$$

So, the bound of Corollary 7.2 will be met if and only if

$$\left\lceil \frac{2^{m-1}\sum_{i=1}^{p}(2^{u_i}-1)}{2^m-1} \right\rceil = \sum_{i=1}^{p} 2^{u_i-1}.$$

This could be equivalently expressed as

$$\sum_{i=1}^{p} 2^{u_i-1} - 1 < \frac{2^{m-1}\sum_{i=1}^{p}(2^{u_i}-1)}{2^m-1} \leq \sum_{i=1}^{p} 2^{u_i-1}.$$

The right inequality holds if and only if

$$\begin{aligned} 2^{m-1}\sum_{i=1}^{p}(2^{u_i}-1) &\leq (2^m-1)\sum_{i=1}^{p}2^{u_i-1} \\ \iff \quad \sum_{i=1}^{p}2^{m-1+u_i} - p2^{m-1} &\leq \sum_{i=1}^{p}2^{m-1+u_i} - \sum_{i=1}^{p}2^{u_i-1} \\ \iff \quad \sum_{i=1}^{p}2^{u_i-1} &\leq p2^{m-1}. \end{aligned}$$

But the last statement indeed holds as $u_i < m$ for $1 \leq i \leq p$.
The left inequality holds if and only if

$$\begin{aligned} \left(\sum_{i=1}^{p}2^{u_i-1} - 1\right)(2^m-1) &< 2^{m-1}\sum_{i=1}^{p}(2^{u_i}-1) \\ \iff \quad \sum_{i=1}^{p}2^{m-1+u_i} - \sum_{i=1}^{p}2^{u_i-1} - 2^m + 1 &< \sum_{i=1}^{p}2^{m-1+u_i} - p2^{m-1} \\ \iff \quad 2^{m-1}(p-2) &< \sum_{i=1}^{p}2^{u_i-1} - 1 \end{aligned}$$

The last statement clearly holds if $p = 1$ or $p = 2$ and so we will focus on them. When $p = 1$, the feasibility condition (17) reduces to $u_1 \leq m$ which is indeed the case. When $p = 2$, the feasibility condition (17) reduces to $u_1 + u_2 \leq sm$. This is always satisfied for $s \geq 2$ and satisfied only if $u_1 + u_2 < m$ for $s = 1$. Thus for each fixed m, we get many infinite classes of optimal resilient functions. We summarize the above discussion as the following Theorem.

Theorem 9.4 *Let m be a fixed integer. Let $m > u_1 \geq u_2 \geq 1$. Then for the following values of n, the upper bound on resiliency given in Corollary 7.2 is tight and it can be attained by linear resilient functions.*

$$\begin{aligned} n &= s(2^m - 1) - 2^{u_1} + 1 \textit{ for } s \geq 1 \\ n &= s(2^m - 1) - 2^{u_1} - 2^{u_2} + 2 \\ &\quad \textit{for } s = 1 \textit{ if } u_1 + u_2 \leq m \textit{ and for } s \geq 2 \end{aligned}$$

A similar analysis yields the parametric situations where the bound of Corollary 7.5 is met by the resilient functions constructed using the above technique. We omit the details and record the final result as the following Theorem.

Theorem 9.5 *Let m be a fixed integer. Let $m > u_1 \geq u_2 \geq u_3 \geq 2$. Then for the following values of n, the upper bound on resiliency given in Corollary 7.5 is tight and it can be attained by linear resilient functions.*

$$\begin{aligned} n &= s(2^m - 1) - 2^{u_1} + 1 \textit{ for } s \geq 1 \\ n &= s(2^m - 1) - 2^{u_1} - 2^{u_2} + 2 \\ &\quad \textit{for } s = 1 \textit{ if } u_1 + u_2 \leq m \textit{ and for } s \geq 2 \\ n &= s(2^m - 1) - 2^{u_1} - 2^{u_2} - 2^{u_3} + 3 \\ &\quad \textit{for } s = 1 \textit{ if } u_1 + u_2 \leq m, \textit{ for } s = 2 \textit{ if } u_1 + u_2 + u_3 \leq 2m \\ &\quad \textit{and for } s \geq 3 \end{aligned}$$

As an example, consider $m = 4$. Then the dimensions of the subspaces must be $1, 2$ or 3. The number of non-zero vectors in these subspaces are respectively $1, 3$ and 7. Taking one at a time, we get optimal resilient functions when $n \equiv 8, 12, 14 \bmod 15$. Taking two at a time (with possible repetition), we get optimal resilient functions when $n \equiv 1, 5, 7, 9, 11, 13 \bmod 15$. These resilient functions meet the bound of Corollary 7.2. Taking three at a time (with possible repetition), we get the additional optimal resilient functions corresponding to $n \equiv 2, 6 \bmod 15$ which meet the bound of Corollary 7.5. Note that in this case, we are not allowed to use subspaces of dimension one.

It should be noted that originally the theory of anticodes was developed to construct optimal linear codes which meet the Griesmer bound [14]. But, we used the theory here to construct linear optimal resilient functions.

Analogous results can be found in [11, Proposition 2.2] and [4, Theorem 5]. However they were obtained from first principles without using the theory of anticodes. The above discussion is intended to illustrate the application of the existing theory of anticodes in the construction of linear optimal resilient functions and is not meant to be exhaustive.

10 Concluding Remarks and Open Problems

In this paper, upper and lower bounds on the largest value of t such that an (n, m, t)-resilient function exists are derived. The upper bounds are derived from linear programming techniques and the lower bounds come from constructions based on error correcting codes. Although we have determined exactly the optimal value of t for $1 \leq m < n \leq 17$, it is evident from the table that there are still many parameter situations where the upper bounds and the lower bounds differ. It would be interesting to determine the optimal values in those cases as well. It is possible that constructions from some other non-linear codes, augmented with our constructions of new resilient functions from old of Section 4.3, might do the trick. For example, if we could construct a $(21, 12, 5)$-resilient function from a systematic code, it would imply that $(20, 11, 5)$, $(19, 10, 5)$, $(20, 12, 4)$, $(19, 11, 4)$ and $(18, 10, 4)$ resilient functions exist and these resilient functions are optimal.

We derived two explicit bounds on the size of the orthogonal arrays using linear programming technique in Section 7. It was also proved in Section 8 that these bounds are as powerful as the LP bounds itself for certain parameter situations. But we still need to use the original LP bound to get tighter bounds for other parametric situations. It would be interesting to determine whether a set of explicit bounds could be developed using the LP bounds such that for any parametric situation at least one of them is as powerful as the LP bound itself. If this can be done, then the original LP bound could be completely eliminated.

By applying our new bounds, we have completed the determination of the optimal resiliency of resilient functions with $m = 3$ in Section 9. Most of the cases for $m = 4$ were also done. One question pertinent to the remaining cases is whether $(18, 4, 8)$ and $(19, 4, 9)$ resilient functions exist or not. Actually $(18, 4, 8)$-resilient function exists if and only if $(19, 4, 9)$-resilient function exists in view of Theorem 6.2 and so we will restrict our attention to one of them. Even if $(18, 4, 8)$-resilient function exists, we cannot immediately extend it to an infinite class of optimal resilient functions using Theorem 9.1 as it has to be a non-linear resilient function. Note that $(18, 4, 7)$-resilient function is the best possible linear resilient function for this parameter situation. On the other hand if $(18, 4, 8)$-resilient function does not exist, then the linear $(18, 4, 7)$-resilient function would be an optimal one. Unfortunately, once again we cannot extend it to an infinite class of optimal resilient functions using Theorem 9.1 as it is not optimal by virtue of meeting one of the two bounds of Corollary 7.2 and Corollary 7.5 (which happens to be 8 and 9 respectively in this case).

Although we have confined our discussion throughout this paper to binary orthogonal arrays and resilient functions for the sake of simplicity, most of the techniques used in this paper are applicable mutatis mutandis to non-binary orthogonal arrays and resilient functions. As an example, the following bound on non-binary orthogonal arrays was proved by Bierbrauer [4] by an algebraic method based on characters of homocyclic groups.

Theorem 10.1 *If there exists an $OA_\lambda(t, n, v)$, then*

$$M \geq v^n \left(1 - \frac{(v-1)n}{v(t+1)}\right).$$

Several other results about non-binary resilient functions can be found in [13].

Acknowledgments

We would like to thank V. Levenshtein for several helpful discussions. The research of K. Gopalakrishnan and D. R. Stinson is supported by NSF grant CCR-9121051.

References

[1] B. I. Belov, V. N. Logachev, and V. P. Sandimirov. Construction of a class of linear binary codes achieving the Varshamov-Griesmer bound. *Problems of Info. Transmission*, **10(3)** (1974), 211–217.

[2] C. H. Bennett, G. Brassard, and A. K. Ekert. Quantum cryptography. *Scientific American*, **267(4)** (1992), 26–33.

[3] C. H. Bennett, G. Brassard, and J. M. Robert. Privacy amplification by public discussion. *SIAM J. Computing*, **17** (1988), 210–229.

[4] J. Bierbrauer. Bounds on orthogonal arrays and codes. Preprint, January 1994.

[5] A. E. Brouwer and Tom Verhoeff. An updated table of minimum-distance bounds for binary linear codes. *IEEE Transactions on Information Theory*, **39** (1993), 662–677.

[6] K. A. Bush. Orthogonal arrays of index unity. *Ann. Math. Stat.*, **23** (1952), 426–434.

[7] P. Camion, C. Carlet, P. Charpin, and N. Sendrier. On correlation-immune functions. In *Advances in Cryptology - CRYPTO '91*, pages 86–100. Springer Verlag, 1992.

[8] Benny Chor, Oded Goldreich, Johan Håstad, Joel Friedman, Steven Rudich, and Roman Smolensky. The Bit Extraction Problem or t-Resilient Functions. *IEEE Symposium on Foundations of Computer Science*, **26** (1985), 396–407.

[9] P. Delsarte. Bounds for unrestricted codes, by linear programming. *Philips Research Reports*, **27** (1972), 272–289.

[10] P. Delsarte. Four fundamental parameters of a code and their combinatorial significance. *Information and Control*, **23** (1973), 407–438.

[11] J. Friedman. On the bit extraction problem. *Proc. 33rd IEEE Symp. on Foundations of Computer Science*, pages 314–319, 1992.

[12] K. Gopalakrishnan, D. G. Hoffman, and D. R. Stinson. A Note on a Conjecture concerning Symmetric Resilient Functions. *Information Processing Letters*, **47** (1993), 139–143.

[13] K. Gopalakrishnan and D. R. Stinson. Characterizations of Non-binary Correlation-Immune and Resilient Functions. Technical Report UNL - CSE - 93 - 010, University of Nebraska - Lincoln, Lincoln, Nebraska - 68588, March 1993. Submitted to *Designs, Codes and Crytography.*

[14] J. H. Griesmer. A bound for error correcting codes. *IBM J. Res. Develop.*, **4** (1960), 532–542.

[15] A. S. Hedayat, N. J. A. Sloane, and J. Stufken. *Orthogonal Arrays: Theory and Practice*. To be published.

[16] V. Levenshtein. Krawtchouk Polynomials and Universal Bounds for Codes and Designs in Hamming Spaces. Submitted to *IEEE Trans. Inform. Theory.*

[17] V. Levenshtein. Bounds for self-complementary codes and their applications. In *CISM Courses and Lectures - Eurocode 92*, pages 159–171. Springer Verlag, 1993.

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

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

[20] Christos H. Papadimitriou and Kenneth Steiglitz. *Combinatorial Optimization - Algorithms and Complexity.* Prentice Hall Inc., 1982.

[21] C. R. Rao. Factorial experiments derivable from combinatorial arrangements of arrays. *J. Royal Stat. Soc.*, **9** (1947), 128–139.

[22] Rainer A. Rueppel. *Analysis and Design of Stream Ciphers*. Springer-Verlag, 1986.

[23] G. Solomon and J. J. Stiffler. Algebraically punctured cyclic codes. *Inform. Contr.*, **8** (1965), 170–179.

[24] D. R. Stinson. Resilient functions and large sets of orthogonal arrays. *Congressus Numerantium*, **92** (1993), 105–110.

[25] D. R. Stinson and J. L. Massey. An infinite class of counterexamples to a conjecture concerning non-linear resilient functions. To appear in *Journal of Cryptology*.

[26] L. Teirlinck. Large sets of disjoint designs and related structures. In Jeffrey H. Dinitz and Douglas R. Stinson, editors, *Contemporary Design Theory - A Collection of Surveys*, chapter 12, pages 564–567. John Wiley & Sons, New York, 1992.

APPENDIX A

A Property of Krawtchouk Polynomials

In this Appendix we shall prove Lemma 8.2, which was used in Section 8 to prove the power of the explicit bounds derived in the paper. First we will state it in a slightly different form.

Theorem A.1 *Let k be an even integer, let $k \leq n \leq 2k-1$ and let $0 \leq i \leq n$. Then*

$$P_k(i) \geq \binom{n}{k}\left[1 - \frac{2k}{n}\right].$$

The right-hand side can be written in a handier combinatorial form:

$$\binom{n}{k}\left[1 - \frac{2k}{n}\right] = \binom{n}{k} - 2\binom{n-1}{k-1} = \binom{n-1}{k} - \binom{n-1}{k-1}.$$

The number $P_k(i)$ can also be expressed in a combinatorial fashion in the following way [18, Ch. 5, page 64]:

$$P_k(i) = \sum_y (-1)^{<x,y>}.$$

Here x is a fixed binary vector of length n and weight i and the sum is over all the binary vectors of length n and weight k. This invites the following combinatorial interpretation.

Let S be an n-element set. Let X be a fixed subset of S of size i. Denote by $g = g(n, k, i)$, the number of subsets Y of S of size k such that $|X \cap Y|$ is even. Similarly denote by $u = u(n, k, i)$, the number of subsets Y of S of size k such that $|X \cap Y|$ is odd. Then clearly

$$g(n,k,i) + u(n,k,i) = \binom{n}{k}$$
$$g(n,k,i) - u(n,k,i) = P_k(i)$$

Using these relations the claim of the Theorem A.1 can be equivalently expressed in terms of the function g:

Theorem A.2 *Let k be an even integer, let $k \leq n \leq 2k-1$ and let $0 \leq i \leq n$. Then*

$$g(n,k,i) \geq \binom{n-1}{k}$$

Lemma A.3 *The statement of Theorem A.2 is true when $i = n$ or $i = 0$.*

Proof: For $i = 0$ this is trivial. When $i = n$, any k-subset intersects X in k elements. As k is even, the number of k-subsets intersecting X in even number of elements is simply $\binom{n}{k}$ which is clearly greater than or equal to $\binom{n-1}{k}$. $\square$

So we may assume $1 \leq i \leq n-1$ in the sequel. For these cases, instead of proving the claim of Theorem A.2, we shall prove the following stronger claim.

Theorem A.4 *Let $k \leq n \leq 2k-1$ and let $1 \leq i \leq n-1$. Then*

$$g(n, k, i) \geq \binom{n-1}{k}.$$

Proof: The proof is by induction on all the three parameters n, k and i. As $i \leq n-1$, the set $S - X$ is non-empty. Let $x \in S - X$. Considering those subsets Y of size k which exclude and include x, we can form the following recurrence relation.

$$g(n, k, i) = g(n-1, k, i) + g(n-1, k-1, i)$$

Then using the induction hypothesis we get

$$\begin{aligned} g(n, k, i) &\geq \binom{n-2}{k} + \binom{n-2}{k-1} \\ &= \binom{n-1}{k}. \end{aligned}$$

As long as the recurrence relation is applicable, it gives exactly what we want. It is also easy to see that the recurrence relation is not applicable exactly when either $n = k$ or $n = 2k-1$. So we have to take care of these two limiting cases.

Lemma A.5 *The statement of Theorem A.4 is true when $k = n$.*

Proof: When $k = n$, $g(n, k, i) = g(n, n, i)$ is 1 if i is even and 0 if i is odd. But $\binom{n-1}{n} = 0$. Hence the statement of Theorem A.4 is true when $k = n$. $\square$

The case of $n = 2k-1$ is little bit more laborious and will be done through the following sequence of lemmas.

Let $n = 2k$. If i is odd, then of each complementary pair of k-subsets, precisely one will intersect X in even cardinality. Thus we get the following Lemma.

Lemma A.6

$$g(2k, k, i) = \frac{1}{2}\binom{2k}{k} = \binom{2k-1}{k} \text{ if } i \text{ is odd.}$$

Using the same trick in the case of odd n ($n = 2k-1$), we get

Lemma A.7

$$g(2k-1,k-1,i)+g(2k-1,k,i)=\binom{2k-1}{k} \textit{ for odd } i.$$

As $i \geq 1$, the fixed subset X is non-empty. Let $x \in X$. Considering those subsets Y of size k which exclude and include x, we can form the following recurrence relation.

$$\begin{aligned} g(n,k,i) &= g(n-1,k,i-1)+u(n-1,k-1,i-1) \\ &= \binom{n-1}{k-1} - g(n-1,k-1,i-1)+g(n-1,k,i-1) \end{aligned}$$

Lemma A.8 *If i is even, then*

$$g(2k,k,i)=2g(2k-1,k,i-1).$$

Proof: By the second recurrence relation, we have

$$g(2k,k,i)=\binom{2k-1}{k-1}+g(2k-1,k,i-1)-g(2k-1,k-1,i-1).$$

As $i-1$ is odd, we can use Lemma A.7 to get

$$g(2k-1,k-1,i-1)=\binom{2k-1}{k}-g(2k-1,k,i-1).$$

Further simplification establishes the Lemma. □

Lemma A.9 *If i is even, then $g(2k,k,i) \geq 2\binom{2k-2}{k}$.*

Proof: By Lemma A.8 we have

$$g(2k,k,i)=2g(2k-1,k,i-1).$$

Using the first recursion, we get

$$g(2k,k,i)=2\{g(2k-2,k,i-1)+g(2k-2,k-1,i-1)\}.$$

As $i-1$ is odd, we can use Lemma A.6 to get

$$2g(2k-1,k-1,i-1)=\binom{2k-2}{k-1}.$$

It follows that

$$g(2k,k,i)=\binom{2k-2}{k-1}+2g(2k-2,k,i-1).$$

We can now use induction on i to assume the validity of the claim of Theorem A.4 in case of the last term (note that the parameter k also satisfies the desired conditions) and get

$$g(2k,k,i) \geq \binom{2k-2}{k-1} + 2\binom{2k-3}{k} = 2\binom{2k-2}{k}.$$

This is exactly what was claimed. □.

The above Lemma immediately yields the following Corollary in view of Lemma A.8.

Corollary A.10 *The statement in Theorem A.4 is true in the case $n = 2k-1$ when i is odd.*

Lemma A.11 *The statement in Theorem A.4 is true in the case $n = 2k-1$ when i is even.*

Proof: By the second recurrence relation, we have

$$g(2k-1,k,i) = \binom{2k-2}{k-1} + g(2k-2,k,i-1) - g(2k-2,k-1,i-1).$$

As $i-1$ is odd, we can use Lemma A.6 to get

$$g(2k-2,k-1,i-1) = \binom{2k-3}{k-1}.$$

It follows that

$$g(2k-1,k,i) = \binom{2k-3}{k-1} + g(2k-2,k,i-1).$$

Once again, we can use induction on i to assume the validity of the claim of Theorem A.4 in case of the last term and get

$$g(2k-1,k,i) \geq \binom{2k-3}{k-1} + \binom{2k-3}{k} = \binom{2k-2}{k}.$$

□.

Finally, we need to take care of the case $i = 1$. This is to assure the basis of induction as it is done over i also.

Lemma A.12 *The statement of Theorem A.4 is true when $i = 1$.*

Proof: When $i = 1$, clearly

$$g(n,k,i) = g(n,k,1) = \binom{n-1}{k}.$$

Thus the claim of Theorem A.4 is trivially satisfied. □

It would be nice to have a combinatorial proof of Theorem A.4 instead of the above lengthy proof by induction through multiple parameters. We leave this as an open problem.