Weakly biased arrays, weakly dependent arrays and error-correcting codes

Jürgen Bierbrauer
Department of Mathematical Sciences
Michigan Technological University
Houghton, Michigan 49931 (USA)

Holger Schellwat
Department of Technology and Science
University of Örebro (Sweden)

October 28, 1999

Abstract

We exhibit the essentially coding-theoretic nature of the basic notions of limited bias and limited dependence. A new efficient construction sheds new light on the construction from [4] and leads to far-reaching improvements. The framework is Fourier analysis, ingredients are algebraic-geometric codes and Ramanujan graphs. The applications comprise universal hashing, authentication, resilient functions and pseudorandomness.

Key Words

$\epsilon-$biased and $\epsilon-$dependent arrays (functions, random variables), (algebraic-geometric) codes, Hermitian codes, expander graphs, Fourier transform.

1 Introduction

The concepts of limited dependence and limited bias have manifold applications in cryptography and complexity theory. We mention universal hashing, authentication, resiliency against correlation attacks, pseudorandomness, block ciphers, derandomization, two-point based sampling, zero-knowledge, span programs, testing of combinatorial circuits, intersecting codes and oblivious transfer (see [31, 28, 30, 4, 12, 35, 19, 7, 14, 9]). A basic notion underlying these concepts are families of $\epsilon-$biased random variables. The heart of the influential papers by Naor and Naor [30] and by Alon, Goldreich, Håstad and Peralta [4] are constructions of families of binary $\epsilon-$biased random variables. In this paper we analyze and simplify these constructions, and we obtain far-reaching improvements. Our language is coding-theoretic and it will turn out that virtually all basic constructions are coding-theoretic.

Definition 1. *An $(n,k)_q-$array $\mathcal{A}$ is an array with n rows and k columns, where the entries are taken from a set with q elements.*

Typically in complexity-theoretic applications one interprets the rows of the array as the points of a probability space with uniform distribution, where each column describes a random variable. One seeks to construct small probability spaces such that the random variables are not far from being independent. When used for the purposes of authentication of messages, the columns of the array are identified with the source, each row is a key and the entries are authenticators. In order to simplify description we chose to use neutral notation. Also, from the point of view of applications it is advisable not to restrict to the binary case. We consider the case that p is a prime.

Definition 2. *Let p be a prime, $v=(v_1,v_2,\dots,v_n)\in\mathbb{F}_p^n$. For every $i\in\mathbb{F}_p=\mathbb{Z}/p\mathbb{Z}$ let $\nu_i(v)$ be the frequency of i as an entry of v and let $\delta_i(v)$ be defined by $\nu_i(v)=\frac{n}{p}+\delta_i(v)$. Let ζ be a primitive complex $p-$th root of unity. The* **bias** *of v is defined as*

$$bias(v)=\frac{1}{n}\mid\sum_{i\in F_p}\delta_i(v)\zeta^i\mid=\frac{1}{n}\mid\sum_{i\in F_p}\nu_i(v)\zeta^i\mid$$

Observe that $\sum_{i\in F_p}\delta_i(v)=0$. We see that $0\leq bias(v)\leq 1$ for every vector v, and $bias(v)=1$ if and only if v is a constant vector. The notion of $bias(v)$ depends on the choice of the root of unity ζ. However, we will always consider the bias of $\mathbb{F}_p-$linear spaces. As the change of root of unity corresponds to the consideration of a multiple of the vector, we see that the bias of an $\mathbb{F}_p-$vector space, defined as the maximum bias of its nonzero members, is defined with no ambiguity, and the result is the same independent of the choice of the p-th root of unity.

Definition 3. *Let $0 \leq \epsilon < 1$. An $(n,k)_p-$array is $\epsilon-$**biased** if every nontrivial linear combination of its columns has bias $\leq \epsilon$.*

Definition 3 shows that we should consider the linear code $\mathcal{C}$ generated by the columns of our $(n,k)_p-$array $\mathcal{A}$. If $dim(\mathcal{C}) < k$, then the 0-word is a nontrivial linear combination of the columns of $\mathcal{A}$. This shows that $\mathcal{A}$ cannot have a bias < 1. We can therefore assume that $\mathcal{C}$ is an $[n,k]_p-$code. Array $\mathcal{A}$ is the transpose of a generator matrix of $\mathcal{C}$ and the bias of $\mathcal{A}$ is the maximum bias of code-words of $\mathcal{C}$. In the binary case it is easy to understand the relationship between these two parameters, the **bias** and the **minimum distance.** An $(n,k)_2$-array $\mathcal{A}$ of rank k is $\epsilon-$biased if and only if the nonzero words of the linear code $\mathcal{C}$ generated by the columns of $\mathcal{A}$ have weights w such that

$$\frac{1-\epsilon}{2} \leq \frac{w}{n} \leq \frac{1+\epsilon}{2}$$

In other words, $\mathcal{C}$ has to be a good code (large minimum distance) and also a good anticode (small maximum distance). It is easy to see that the theory of binary weakly biased arrays is equivalent to the theory of binary linear codes containing the all-1-word. When $p > 2$ the link between minimum distance and bias is less obvious. Here is a generalization of the binary construction for general p :

Theorem 1. *Let p be a prime and $\mathcal{C}'$ a linear code $[n,k+1]_p$, which contains a word m of weight n. Let D be the maximum weight among the words not contained in $\langle m\rangle$. Let $\mathcal{C} \subset \mathcal{C}'$ be a hyperplane avoiding $\langle m\rangle$ (and hence a code $[n,k]_p$ with maximum weight D). The transpose of a generator matrix of $\mathcal{C}$ is an $(n,k)_p-$ array with bias $\epsilon \leq p\frac{D}{n} - (p-1)$.*

Proof. Let $0 \neq v \in \mathcal{C}$. The definition of D and the presence of m show that $\nu_i(v) \geq n - D$ for all $i \in \mathbb{F}_p$. Define $\mu_i(v) \geq 0$ by $\nu_i(v) = n - D + \mu_i(v)$. As the sum over all powers of ζ vanishes we have

$$\sum_i \delta_i(v)\zeta^i = \sum_i \nu_i(v)\zeta^i = \sum_i \mu_i(v)\zeta^i$$

and

$$|\sum_i \delta_i(v)\zeta^i| \leq \sum_i \mu_i(v) = \sum_i \nu_i(v) - p(n-D) = n - p(n-D) = pD - (p-1)n.$$

We conclude $bias(v) \leq p\frac{D}{n} - (p-1)$. ■

Observe the trivial fact that the set of k-tuples form the rows of an unbiased ($= 0-$biased) $(p^k, k)_p-$array.

2 Weak bias and weak dependency

We define the notions that we are most interested in.

Definition 4. *Let $0 \leq \epsilon < 1$. An $(n, k)_p-$array is t-**wise** $\epsilon-$**biased** if every nontrivial linear combination of at most t of its columns has bias $\leq \epsilon$.*

Definition 5. *Let $0 \leq \epsilon < 1$. An $(n, k)_p-$array $\mathcal{A}$ is t-**wise** ϵ **-dependent** if for every set U of $s \leq t$ columns and every $a \in \mathbb{F}_p^s$ the frequency $\nu_U(a)$ of rows of $\mathcal{A}$, whose projection onto U equals a satisfies*

$$|\frac{\nu_U(a)}{n} - 1/p^s| \leq \epsilon.$$

The notion of a t-wise ϵ -dependent array generalizes the notion of an orthogonal array of strength t. An array is t-wise independent ($= t-$ wise 0-dependent) if and only if it is an orthogonal array of strength t.

The standard construction of t-wise $\epsilon-$biased arrays is essentially from [30].

Theorem 2. *Let the following be given:*

- *An $(n, k)_p-$array B, which is $\epsilon-$biased.*
- *A linear code $[N, N-k, t+1]_p$.*

Then we can construct an $(n, N)_p-$array, which is t-wise $\epsilon-$biased.

Proof. Let H be a check matrix of the code, hence an $(k, N)_p$-array each t columns of which are linearly independent. Our array is the product $\mathcal{A} = BH$. Denote by r_i the rows of B, by c_j the columns of $H, i = 1, \ldots, n; j = 1, \ldots, N$. The entries of BH are the dot products $r_i \cdot c_j$. Let $0 \neq \lambda =$

$(\lambda_1, \dots, \lambda_N)$ of weight $\leq t$. The corresponding linear combination of columns of $\mathcal{A}$ has entry $v_i = r_i \cdot (\sum_j \lambda_j c_j)$ in row i. The defining property of H shows that $\sum_j \lambda_j c_j \neq 0$. Vector $v = (v_i)$ is therefore a nontrivial linear combination of the columns of B, hence has bias $\leq \epsilon$. ■

The most important structures from the point of view of applications are $t-$wise $\epsilon-$dependent arrays. The aim is to obtain effective constructions. The main idea in [30] was that the notion of bias is easier to handle than dependency. In the next section we will introduce a very general notion of bias and use Fourier transform to show that $t-$wise $\epsilon-$biased arrays are also $t-$wise $\epsilon-$dependent. This shows that it suffices to construct $t-$wise $\epsilon-$biased arrays. Theorem 2 gives another reduction (via arror-correcting codes): it suffices to construct $\epsilon-$biased arrays.

3 A general notion of bias

Let $G = \mathbb{F}_p^k$ be the elementary abelian group of order $g = p^k$. Consider the space $L^2(G)$ of complex-valued functions on G, with scalar product $\langle , \rangle$, where

$$\langle f, g \rangle = \frac{1}{g} \sum_{x \in G} f(x) \cdot \overline{g(x)}.$$

Here $x = (x_1, x_2, \dots, x_k) \in G$. The characters of G form an orthonormal basis of $L^2(G)$. Let ζ be a primitive complex $p-$th root of unity. Every element $z = (z_1, z_2, \dots, z_k)$ yields a character ϕ_z, where

$$\phi_z(x) = \zeta^{\sum x_i \cdot z_i} = \zeta^{x \cdot z}.$$

Definition 6. *Let $f \in L^2(G)$. Express $f(x)$ in terms of the characters in the form*

$$f = \sum_{z \in G} \hat{f}(z) \phi_z.$$

The complex numbers $\hat{f}(z)$ are known as the **Fourier coefficients** *of f. The* **mass** *of f is defined as*

$$mass(f) = g\langle f, 1 \rangle = g\hat{f}(0) = \sum_{x \in G} f(x).$$

Finally we define the **bias** *of $z \in G$ with respect to function f by*

$$bias_f(z) = \frac{g}{mass(|f|)} |\hat{f}(z)|$$

Observe that because of the orthonormality of the characters the Fourier coefficients are given by

$$\hat{f}(z) = \langle f, \phi_z \rangle = \frac{1}{g} \sum_{x \in G} f(x) \zeta^{-x \cdot z}.$$

Definition 7. *Let $\mathcal{A}$ be an $(n,k)_p-$array with entries in $\mathbb{F}_p$. The characteristic function $f(x) = \chi_{\mathcal{A}}(x)$ counts with which multiplicity x occurs as a row of $\mathcal{A}$. We define the Fourier transform $\hat{\mathcal{A}}(z)$, the mass and the bias of $\mathcal{A}$ by identifying $\mathcal{A}$ with its characteristic function and applying Definition 6. Observe $mass(\mathcal{A}) = n$.*

Let us compare the bias in Definition 7 with the combinatorial notion from Definition 3. So let $f = \chi_{\mathcal{A}}$ be the characteristic function of the $(n,k)_p-$array $\mathcal{A}$. As $f = |f|$ and $mass(f) = n$ we see that the bias of z in the sense of Definition 6 coincides with the bias of vector v, the linear combination of columns of $\mathcal{A}$ with z as vector of coefficients, in the sense of Definition 2.

Lemma 1. *Let $f = \chi_{\mathcal{A}}$ be the characteristic function of an $(n,k)_p-$array $\mathcal{A}$ and $z \in G = \mathbb{F}_p^k$. Then $bias_f(z)$ in the sense of Definition 6 is the same as $bias(v)$ in the sense of Definition 3, where v is the linear combination of the columns of $\mathcal{A}$ with z as vector of coefficients.*

The following characterization of t-wise ϵ -biased arrays in terms of their Fourier transform follows directly from Definition 6 and Lemma 1.

Theorem 3. *An $(n,k)_p-$array $\mathcal{A}$ is t-wise ϵ -biased if and only if its Fourier coefficients $\hat{\mathcal{A}}(z)$ satisfy*

$$|\hat{\mathcal{A}}(z)| \leq \frac{n}{g}\epsilon \text{ for } 0 < wt(z) \leq t.$$

4 The relation between weak dependency and weak bias

Definition 8. *Let $\mathcal{A}$ be an $(n,k)_p-$array, $f \in L^2(G)$. The* **expectation** *$E_{\mathcal{A}}(f)$ with respect to $\mathcal{A}$ is defined as the statistical expected value of f when the rows of $\mathcal{A}$ are used as a probability space with uniform distribution:*

$$E_{\mathcal{A}}(f) = \sum_{x \in G} \frac{\chi_{\mathcal{A}}(x)}{n} f(x)$$

The following lemma was inspired by [13]:

Lemma 2. *Let $\mathcal{A}$ be an $(n,k)_p-$array. Then the following are equivalent:*

- *$\mathcal{A}$ is t-wise $\epsilon-$dependent.*
- *for every subset U of $l \leq t$ columns and every $f \in L^2((\mathbb{F}_p)^U)$ the following holds:*

$$|E_U(f) - mass(f)/p^l| \leq \epsilon \cdot mass(|f|).$$

Here the expectation $E_U(f)$ is with respect to the restriction of $\mathcal{A}$ to the columns in U.

Proof. If f runs through the characteristic functions $\chi_a, a \in (\mathbb{F}_p)^U$, then the second condition expresses the definition of t-wise $\epsilon-$dependent arrays. So one direction of the equivalence is clear. Assume therefore $\mathcal{A}$ is t-wise $\epsilon-$dependent, fix U and denote by $\nu_U(a)$ the multiplicity of a as a row in the restriction to U. Consider a general function $f = \sum_a c_a \chi_a$. Because of the linearity of the expectation and the mass, we get $|E_U(f) - \frac{mass(f)}{p^l}| = |\sum_a c_a(\frac{\nu_U(a)}{n} - \frac{1}{p^l})| \leq \sum_a |c_a| \cdot \epsilon = \epsilon \cdot mass(|f|)$. ∎

Lemma 3. *Let $\mathcal{A}$ be an $(n,k)_p-$array. Then the following are equivalent:*

- *$\mathcal{A}$ is t-wise $\epsilon-$biased.*
- *For every set U of $l \leq t$ columns and every $0 \neq z \in G$ with support in U, the restriction g_z of the character ϕ_z to $\mathbb{F}_p^U$ satisfies*

$$|E_U(g_z)| \leq \epsilon$$

Proof. We have $|E(g_z)| = |\sum_{x \in F_p^U} \frac{\nu_U(x)}{n} \zeta^{-x \cdot z}|$. Comparison with Definition 2 shows that $| E(g_z) |$ is the bias of the linear combination of columns of $\mathcal{A}$ with $-z$ as set of coefficients. ∎

After these preparations we can start comparing bias and dependency.

Theorem 4. *Let $\mathcal{A}$ be a $p-$ary array, which is t-wise ϵ -dependent. Then $\mathcal{A}$ is i-wise $p^i\epsilon$-biased for all* $0 < i \leq t$

Proof. Lemma 2 gives a characterization of weak dependency in terms of a property, which all complex-valued functions defined on $(\mathbb{F}_p)^U$ have to satisfy, where U is a set of $i \leq t$ columns. Let U be given. Let g_z be the restriction to $(\mathbb{F}_p)^U$ of the character ϕ_z, where $0 \neq z \in (\mathbb{F}_p)^U$. Apply Lemma 2, observing $mass(g_z) = 0, mass(|g_z|) = p^i$. We get $|E(g_z)| \leq p^i\epsilon$. Apply Lemma 3. ■

Theorem 5. *Let $\mathcal{A}$ be a $p-$ary array, which is i-wise δ_i-biased* $(i = 1, 2, \ldots, t)$. *Then $\mathcal{A}$ is t-wise ϵ -dependent, where $\epsilon = (\sum_{i=1}^{t} \binom{t}{i}(p-1)^i\delta_i)/p^t$.*

Proof. Let $|U| = l \leq t, g_z$ as above, $h = p^l, a \in \mathbb{F}_p^U$. It is easy to check that $\chi_a = \frac{1}{h}\sum_z \zeta^{-a\cdot z}g_z$. By assumption $|E(g_z)| \leq \delta_{wt(z)}, z \neq 0$. Thus $|E(\chi_a) - \frac{1}{h}| = \frac{1}{h}|\sum_{z\neq 0}\zeta^{-a\cdot z}E(g_z)| \leq \frac{1}{h}\sum_{i=1}^{l}\binom{l}{i}(p-1)^i\delta_i$. The δ_i form by definition a non-decreasing series. It is easily seen that the last expression increases with l. This leads to the statement of the theorem. ■

Two special cases deserve being mentioned.

Corollary 1. *Let $\mathcal{A}$ be a p-ary array. If $\mathcal{A}$ is $(t-1)$-wise 0-biased (equivalently: is an orthogonal array of strength $t-1$) and t-wise ϵ -biased, then $\mathcal{A}$ is t-wise $(\frac{p-1}{p})^t\epsilon$-dependent.*

Theorem 6. *Let $\mathcal{A}$ be a p-ary array. If $\mathcal{A}$ is t-wise ϵ -biased, $\epsilon > 0$, then $\mathcal{A}$ is t-wise ϵ'-dependent for some $\epsilon' < \epsilon$.*

Proof. If $\mathcal{A}$ is t-wise $\epsilon-$biased, then it is also i-wise $\epsilon-$biased for all $1 \leq i < t$. We can apply Theorem 5 with $\delta_i = \epsilon, i = 1, 2, \ldots, t$. It follows that $\mathcal{A}$ is t-wise ϵ'-dependent, where $\epsilon' = \epsilon(\sum_{i=1}^{t}\binom{t}{i}(p-1)^i)/p^t = \epsilon(p^t - 1)/p^t < \epsilon$. ■

5 The basic construction

We have seen that the construction problem of $t-$wise ϵ-dependent arrays can be reduced to ϵ-biased arrays. We concentrate on the following versions of the construction problem:

Problem 1. *For which $a > 0$ can we find asymptotic constructions for $\epsilon-$biased $(n, k)_p-$arrays such that $n \leq (\frac{k}{\epsilon})^a$?*

Problem 2. *For which $b > 0$ can we find asymptotic constructions for $\epsilon-$biased $(n,k)_p-$arrays such that $n = O(\frac{k}{\epsilon^b})$?*

Moreover we are interested in efficient constructions, not just proofs of existence.

It has been proved in the final version of [4] that $b \geq 2$. A nonconstructive argument in [6] shows that $b = 2$. This yields the complete answer to Problem 2: $b = 2$. Our main result will be that there are efficient constructions for every $a > 1$ in Problem 1 and for every $b > 2$ in Problem 2.

Our basic construction is a generalization of what was called the **Powering construction** in [4].

Theorem 7. *Let $\mathcal{C}$ be a code $[n,k,d]_q$, where $q = p^m$. Let $v(1),\dots,v(k)$ be a basis of $\mathcal{C}$. Construct a $(qn,k)_p$-array $A(\mathcal{C})$ as follows: the columns are indexed by the basis elements $v(j)$, the rows are indexed by pairs (i,u), where i is a coordinate of the ambient space $\mathbb{F}_q^n$ containing the code $\mathcal{C}$ and $u \in \mathbb{F}_q$. The entry in row (i,u) and column j is*

$$tr(uv(j)_i).$$

Here $v(j)_i$ is the entry of $v(j)$ in coordinate i and $tr : \mathbb{F}_q \longrightarrow \mathbb{F}_p$ is the trace. Then $A(\mathcal{C})$ has bias $\leq 1 - \frac{d}{n}$.

Proof. Let $\alpha = (\alpha_1,\dots,\alpha_k) \neq (0,\dots,0)$. Let c be the corresponding linear combination of the columns of $A(\mathcal{C})$. The entry of c in row (i,u) is

$$\sum_{j=1}^{k} \alpha_j tr(uv(j)_i) = tr((\sum_j \alpha_j v(j)_i)u) = tr(v_i u).$$

Here $0 \neq v = \sum_j \alpha_j v(j) \in \mathcal{C}$. We imagine the rows of the array as being subdivided in n segments according to the value of i. If $v_i \neq 0$, then the corresponding segment of c is unbiased. If $v_i = 0$, then the corresponding segment is 0. We obtain

$$bias(c) \leq \frac{(n-d)q}{nq} = 1 - \frac{d}{n}.$$

■

6 Construction of weakly biased arrays

In order to make the most of Theorem 7 we need efficiently constructed families of extremly good codes. These are provided by algebraic-geometric codes.

Theorem 8. *Let q be a prime-power and $e \geq 1$ a rational number. Then there is an infinite set $\mathcal{M}$ of natural numbers such that em is a natural number for all $m \in \mathcal{M}$ and for every $m \in \mathcal{M}$ there is an algebraic curve $\mathcal{K}_m$ defined over $\mathbb{F}_{q^m}$ with more than q^{em} rational points and genus g_m, such that $lim_{m \in \mathcal{M}} \frac{g_m}{q^{em}} = 0$.*

A proof can be found in [20]. This proof is "elementary" in the sense that it uses the theory of algebraic curves (equivalently: function fields in one variable) over finite fields. The celebrated first proof in [33] relied on the theory of modular functions.

The basic mechanism of algebraic-geometric codes yields the following:

Theorem 9. *Let an algebraic curve (equivalently: a function field) with genus g be defined with $\mathbb{F}_q$ as field of constants, and with N rational points. Then for every $k < N - 1$ we can construct a linear code with parameters $[N - 1, k, N - k - g]_q$ (the minimal distance can actually be even higher).*

Theorem 8 yields codes $\mathcal{C}_m$ with parameters

$$[q^{em}, Rq^{em}, (1 - R)q^{em} - g_m]_{q^m}.$$

Using these codes in Theorem 7 yields the following:

Theorem 10. *For every prime p, every rational number $e \geq 1$ and $0 < \epsilon < 1$ there is a family of $(n_i, k_i)_p-$arrays, which are ϵ_i- biased such that $lim_i \epsilon_i = \epsilon$ and*

$$lim_i \left(\frac{n_i}{(k_i/\epsilon_i)^{(e+1)/e}} \right) = 1.$$

This produces families of $\epsilon-$biased $(n, k)_p-$arrays, where asymptotically $n = (\frac{k}{\epsilon})^a$, for every $a > 1$. Here $a = (e + 1)/e$ in the terminology of Theorem 10. Each family from Theorem 10 for a fixed value of e yields an exponent $a = (e + 1)/e$ in Problem 1. Moreover these constructions are explicit. The usefulness of this construction is determined by how efficient the AG-codes are, which are used as ingredients. The Reed-Solomon codes are obtained from the projective line ($g = 0$). They yield $e = 1$, hence $a = 2$. This corresponds to the original powering construction from [4].

The first layer in the construction of [20] are the Hermitian curves. These satisfy the assumptions of Theorem 8 for $e = 1.5$ and therefore yield $a = 5/3$ in Problem 1.

We remark that Theorem 10 is best possible in the sense, that it is impossible to replace the exponent a, which is arbitrarily close to 1, but fixed, by $a = 1$. In fact, consider the binary case. A family of binary codes such that $n \sim k/\epsilon$ would give as point in $(\delta, R)-$space, which is asymptotically reachable by binary linear codes and meets the asymptotic Plotkin bound with equality. This is impossible as it contradicts the JPL-bound. Still it is unsatisfactory that in Theorem 10 the dimension of the code appears with an exponent > 1. We can further generalize Theorem 7.

Theorem 11. *Let $\mathcal{C}$ be a code $[n, k, d]_q$, where $q = p^m$ and $\mathcal{B}$ an $(n_0, m)_p$-array of bias ϵ_0. We can construct an $(nn_0, km)_p-$array with bias $\epsilon = 1 - \delta + \delta\epsilon_0 < 1 - \delta + \epsilon_0$, where $\delta = d/n$ is the relative distance of code $\mathcal{C}$.*

Proof. Fix a basis $\{a_r \mid r = 1, \dots, m\}$ of $I\!\!F_q \mid I\!\!F_p$ and denote the columns of $\mathcal{B}$ by $b_r, r = 1, \dots, m$. Construct an $(nn_0, km)_p$-array $A(\mathcal{C}, \mathcal{B})$ as follows: the columns $s(j, r)$ are indexed by pairs of basis elements $v(j)$ of the code and basis elements a_r of $I\!\!F_q \mid I\!\!F_p$. Let $v_i(j) = \sum_{r=1}^m \lambda_r^{(i,j)} a_r$. The rows come in sections of length n_0, the sections being indexed by the coordinates $i = 1, 2 \dots, n$. Section i of column $s(j, r)$ is $\lambda_r^{(i,j)} b_r$. We want to show that this array $A(\mathcal{C}, \mathcal{B})$ has bias $\leq 1 - \delta + \delta\epsilon_0$.

Let $s = \sum_{j,r} \mu(j, r)s(j, r)$ be a linear combination of the columns, where $\mu(j, r) \in I\!\!F_p$, not all zero. Every section of s is a linear combination of columns of $\mathcal{B}$, and hence either is 0 or has bias $\leq \epsilon_0$. It will suffice to show that there are at most $n - d$ indices i such that $s_i = 0$. We have

$$s_i = \sum_{j,r} \mu(j, r)\lambda_r^{(i,j)} b_r = \sum_r (\sum_j \mu(j, r)\lambda_r^{(i,j)}) b_r.$$

It follows that $s_i = 0$ if and only if $\sum_{j=1}^k \mu(j, r)\lambda_r^{(i,j)} = 0$ for all r.

Consider the codeword $w^{(r)} = \sum_{j=1}^k \mu(j, r)v(j) \in \mathcal{C}$. As not all the coefficients $\mu(j, r) \in I\!\!F_p$ vanish we must have $w^{(r)} \neq 0$ for some r. If we expand the entry $w_i^{(r)}$ with respect to the basis a_r we obtain

$$w_i^{(r)} = \sum_r (\sum_{j=1}^k \mu(j, r)\lambda_r^{(i,j)}) a_r.$$

Comparison shows that $s_i = 0$ if and only if $w_i^{(r)} = 0$. Choose r such that $w^{(r)} \neq 0$. It follows that at least d segments have $s_i \neq 0$. The bound on the bias follows from the following innocent Lemma 4, which is trivial. ∎

Lemma 4. *Let $v = (v_1, v_2, \ldots, v_n)$, where $v_i \in \mathbb{F}_p^m$ and $bias(v_i) \leq \epsilon_i$. Then*

$$bias(v) \leq \frac{\sum_{i=1}^{n} \epsilon_i}{n}$$

Observe that Theorem 11 is both a generalization and a strengthening of Theorem 7. In fact, let $\epsilon_0 = 0$ in Theorem 11. We obtain an $(nn_0, km)_p-$array of bias $1-\delta$, whereas the basic construction Theorem 7 yields only an array with k columns and the same bias.

Apply Theorem 11 with Reed-Solomon codes $[p^m, Rp^m, (1-R)p^m]$ as outer codes. A first application of the basic construction to Reed-Solomon codes has provided us with arrays solving Problem 1 with $a = 2$. There are therefore $(n_0, m)_p-$ arrays such that $n_0 \leq (m/\epsilon_0)^2$. Choose $R = \epsilon_0 = \epsilon/2$. Theorem 11 yields an $\epsilon-$biased $(N, K)_p-$array, where $N = n_0 p^m, K = km = Rmp^m$. For which exponent a' will it be true that

$$N = n_0 p^m = \frac{4m^2}{\epsilon^2} p^m \leq (K/\epsilon)^{a'} = (mp^m/2)^{a'}?$$

This will be true for every $a' > 1$ when m is large enough. We have a second construction showing that every exponent $a > 1$ in Problem 1, and the second construction does not involve complicated AG-codes.

7 Block covering systems

An analysis of the constructions in [30] and [2] suggests the introduction of the following concept.

Definition 9. *A* **block covering system** $BCS(n, l, b, \gamma, \epsilon)$ *is a family $\mathcal{B}$ of l-element subsets of a fixed $n-$set Ω, where $|\mathcal{B}| = b$ and the following holds: whenever $W \subset \Omega, \mid W \mid = w$ such that $1 - \frac{w}{n} \leq \gamma$, then*

$$\frac{\textit{number of blocks avoiding } W}{b} \leq \epsilon.$$

Here it is permitted that a point occurs in a block with multiplicity more than 1. It is also permitted that different blocks are made of the same set of points, even with the same multiplicities. Clearly, we are interested in $BCS(n, l, b, \gamma, \epsilon)$ with small l, small b and small ϵ.

In the language of graphs we can describe a BCS as a bipartite graph with n vertices on the left, $b \geq n$ vertices on the right, where each right vertex has precisely l neighbours. The defining property says that every sufficiently large set W of vertices on the left has as set of neighbours almost all vertices on the right. Here the sizes are measured with respect to the size of the underlying set. Set W must satisfy that a random choice of a left vertex lands in W with probability at least $1 - \gamma$. Likewise a random choice among reight vertices lands in $\Gamma(W)$ with probability $\geq 1 - \epsilon$. This notion is clearly closely related to concepts like **expanders** and **dispersers,** which are often used in theoretical computer science, see the survey [32]. In fact, a BCS can be seen as some kind of a disperser. The main differences are the following: in a disperser the dispersion is from the large set of vertices to the small set. Typically dispersers are very unbalanced. In BCS the dispersion happens the other way around, from the small set to the large set. Also, the assumption of regularity is on starting end for dispersers, on the receiving end for BCS. More importantly, in BCS the sizes of the subsets are measured relative to the size of the underlying set.

One source for BCS are orthogonal arrays. The result of Gopalakrishnan and Stinson [24] says in our terminology that the rows of an $OA_1(2, l, q)$ form a

$$BCS(q, l, q^2, \gamma, \frac{\gamma}{1 + (1 - \gamma)(l - 1)}).$$

Observe that $OA_1(2, l, q)$ can be obtained by projecting from 2-dimensional Reed-Solomon codes.

An often used construction idea for BCS (and for dispersers) are walks in expander graphs and neighbourhoods of expander graphs. A study of Eigenvalues (see [26, 27]) leads to the following:

Theorem 12. *Let Γ be a graph on n vertices, which is connected and regular of valency Δ. Let λ_1 be the second-largest Eigenvalue of Γ. The family $T_l(\Gamma)$ of walks of length l leads after identification of inverse walks to a $BCS(n, l, \frac{1}{2} n\Delta^{l-1}, \gamma, \gamma(\gamma + (1 - \gamma)\frac{\lambda_1}{\Delta})^{l-1})$.*

Theorem 12 can be found in [3].

A different and in a way dual method of obtaining BCS from graphs is to use the family of neighbours of vertices.

Bounds from [5] and [2] lead to the following result:

Theorem 13. *Let* Γ *be a graph on* n *vertices, which is connected and regular of valency* Δ. *Let* λ_1 *be the second-largest Eigenvalue of* Γ. *The family* $\mathcal{B} = \{i^\perp \mid i \in V\}$ *is a* $BCS(n, \Delta, n, \gamma, Min\{\frac{\gamma}{1+4(1-\gamma)(1-\lambda_1/\Delta)}, \frac{\gamma}{1-\gamma}(\frac{\lambda_1}{\Delta})^2\})$.

7.1 BCS, codes and $\epsilon-$biased arrays

Theorem 14. *Assume the following exist:*

- *A linear code* $\mathcal{C}$ *with parameters* $[n, k, d]_q$
- *A* $BCS(n, l, b, 1 - \frac{d}{n}, \epsilon)$ *for some* l.

Then we can construct a q^l- *ary* $\mathbb{F}_q-$*linear code* $\tilde{\mathcal{C}}$ *of length* b, *with just as many codewords as* $\mathcal{C}$ *and minimum distance (equal to minimum weight)* $\geq (1 - \epsilon)b$.

Proof. Let $v \in \mathcal{C}$. The corresponding codeword of $\tilde{\mathcal{C}}$ is $\tilde{v} = (\tilde{v}_B)$, where B varies through the blocks. Here $\tilde{v}_B = (v_i)_{i \in B}$. Observe that we interpret $\tilde{\mathcal{C}}$ as defined over $V(l, q)$, the vector space of dimension l over $\mathbb{F}_q$. Clearly $\tilde{\mathcal{C}}$ is linear over the ground field $\mathbb{F}_q$. This shows that minimum weight and minimum distance agree. Let $v \neq 0$. Then $\tilde{v}_B = 0$ if and only if $B \cap supp(v) = \emptyset$. The definition of a BCS shows that the number of such blocks is at most $\epsilon \cdot b$. ■

The output of Theorem 14 is a code, which is q^l-ary and $\mathbb{F}_q-$linear. Concatenation with any linear $q-$ary code of dimension l as inner code produces a q-ary linear code. This approach was used in [2] in an attempt to construct asymptotically nontrivial families of linear codes. We remark that codes, which are q^l-ary but only $q-$linear (also known as **additive codes** or **subgroup codes**)) have recently found renewed interest. In [16, 25] the mechanism of BCH-codes is extended to cover also additive codes, in [17] use is made of this generalization of BCH-codes to construct quantum codes.

Here we use concatenation for the construction of weakly biased arrays. All we need to know is the bias of the inner code.

Theorem 15. *Assume the following exist:*

- *A binary linear code $\mathcal{C}$ with parameters $[n, k, d]_2$.*
- *A $BCS(n, l, b, 1 - \frac{d}{n}, \epsilon)$ for some l.*
- *An ϵ_0-biased $(n_0, l)_2-$array.*

Then we can construct an $(n_0 b, k)_2-$array, which is $\{\epsilon+\epsilon_0(1-\epsilon)\}-$biased.

Proof. Apply Theorem 14 and use concatenation in an obvious way. Denote the code generated by the columns of the resulting array by $\mathcal{C}'$. We need to control the bias of an arbitrary codeword v'. If $B \cap supp(v) = \emptyset$, then v'_B is a 0-tuple of length n_0. By definition of a BCS there are at least $(1-\epsilon)b$ blocks B such that $B \cap supp(v) \neq \emptyset$. Each corresponding section v'_B is a nonzero codeword of the inner code, hence of bias $\leq \epsilon_0$. It follows that v'_B can have at most $\epsilon_0 n_0$ more zeroes than ones. The total surplus of zeroes in v' is therefore limited by $\epsilon b n_0 + (1-\epsilon) b \epsilon_0 n_0$. Division by the length $b n_0$ yields the desired upper bound on the bias. ∎

The construction of weakly biased binary arrays as given in [30] is an application of Theorem 15 in the binary case, where the inner code is the Simplex code $[2^l, l, 2^{l-1}]$ of bias $\epsilon_0 = 0$ and the BCS is the family of walks (see Theorem 12). We remark that the family of codes constructed in [30] and used in the role of code $\mathcal{C}$ in Theorem 15 is not interesting. In fact, the use of 2-dimensional Reed-Solomon codes makes sure that the family has to be asymptotically trivial.

Apply Theorem 15 with the following ingredients: Code $\mathcal{C}$ belongs to a family of asymptotically nontrivial codes with relative minimum distance δ. The BCS is derived from Ramanujan graphs (see [29]) on n vertices with valency Δ via Theorem 13. It has parameters

$$BCS(n, \Delta, n, 1-\delta, \frac{1-\delta}{\delta}\frac{4}{\Delta}).$$

The weakly biased array are derived from Theorem 10, an ϵ_0-biased $(n_0, \Delta)_p-$ array, where $n_0 = (\Delta/\epsilon_0)^{(e+1)/e}$. Choose $\epsilon_0 = \epsilon/2$ and Δ such that $\frac{1-\delta}{\delta}\frac{4}{\Delta} \leq \epsilon/2$. We obtain an $\epsilon-$biased $(N, K)_p-$ array, where $N = O(K/\epsilon^{2(e+1)/e})$.

We reach the promised conclusion: every exponent $b > 2$ in Problem 2 is reachable with an effective construction.

8 Constructions of weakly dependent arrays

We use Theorem 2 with the weakly biased arrays constructed in Section 6 and in Subsection 7.1 as one ingredient. Primitive BCH-codes have parameters $[p^i - 1, p^i - 1 - it, t + 1]_p$. The following is obtained:

Theorem 16. *Let p be a prime. For every natural number t and $a > 1$ there are efficient constructions for infinite families of t-wise $\epsilon-$ biased $(n, N)_p-$arrays, where asymptotically*

$$n \leq (\frac{tlog_p(N)}{\epsilon})^a.$$

For every natural number t and $b > 2$ there are efficient constructions for infinite families of t-wise $\epsilon-$ biased $(n, N)_p-$arrays, where asymptotically

$$n = O(tlog_p(N)/\epsilon^b).$$

Theorem 6 shows that we can replace the term *t-wise $\epsilon-$ biased* in Theorem 16 by *t-wise $\epsilon-$ dependent.*

8.1 A problem in coding theory

As one of the ingredients for the derivation of Theorem 16 we used primitive BCH-codes. An attempt to optimize the code contribution leads to the following highly nontrivial problem in coding theory.

Definition 10. *Fix q and t. Denote by $n_{t,q}(k)$ the maximal length n of a code $[n, n - k, t + 1]_q$, equivalently the largest length of a linear orthogonal array of dimension k and strength t. Define*

$$l(t, q) = lim\ inf_{k\to\infty} \frac{k}{log_q(n_{t,q}(k))}$$

The primitive $BCH-$codes show $l(t, q) \leq t$. This is what we used in the derivation of Theorem 16. The Gilbert-Varshamov bound seems to give the same bound $l(t, q) \leq t$ as the BCH-codes. The function $l(t, q)$ has been studied by Dumer in [15]. The sphere packing bound implies $l(t, q) \geq \lfloor t/2 \rfloor$. We have

$$\lfloor t/2 \rfloor \leq l(t, q) \leq t,$$

showing that asymptotically we cannot do better than replace the factors t in the statement of Theorem 16 by $\lfloor t/2 \rfloor$. Still, this is an interesting coding problem.

Let $t = 1$. Clearly $n_{1,q}(k) = q^k - 1$, hence $l(1, q) = 1$. Clearly we have $l(t, q) \geq l(t-1, q)$ in general. When $t = 2$ the maximum N such that a code $[N, N-k, 3]_p$ exists is of course $N = (p^k - 1)/(p-1)$. This yields $l(2, q) = 1$.

Theorem 17. *If there is an $\epsilon-$biased $(n, k)_p-$array, then there is a 2-wise $\epsilon-$biased $(n, \frac{p^k-1}{p-1})_p-$array. In particular for every $a > 1$ there are efficient constructions for infinite families of 2-wise $\epsilon-$ biased $(n, N)_p-$arrays, where asymptotically*

$$n \leq (\frac{log_p(N) + 1}{\epsilon})^a.$$

Also, in the binary case we have $l(t, 2) = \lfloor t/2 \rfloor$. To see this just use primitive $BCH-$codes with defining interval $[1, t/2]$ if t is even and $[0, (t-1)/2]$ if t is odd. In case $t = 3$ we obtain a question on geometrical objects called caps. It is known that $l(3, q) \leq 3/2$, and this is the best known upper bound for general q. It is an open general question on caps if $l(3, q) = 1$. The ternary case is special, too. It follows from a construction [11] based on the Hill cap that $l(3, 3) < 1/log_3(2.21)$.

9 Authentication codes

In this application we interpret the rows of our array as keys, the columns as source states in an authentication scheme. An $(N, m)_p-$array is $(\delta, t)-$**almost strongly universal** (short $(\delta, t)-$ASU) if for every set $U = U_0 \cup \{u\}$ of t columns and every $a' \in \mathbb{F}_p^{t-1}, x \in \mathbb{F}_p$ the frequencies $\nu_{U_0}(a')$ and $\nu_U(a', x)$ satisfy

$$\mid \nu_U(a', x)/\nu_{U_0}(a') \mid \leq \delta.$$

This means that when we authenticate t messages using the same key, the opponent's conditional probability of guessing the authenticator for message number t based on his knowledge of the authenticators of the preceding $t-1$ messages is still bounded by δ. The following obvious fact was proved in [28]:

Theorem 18. *A t-wise $\epsilon-$ dependent array is $(\delta, t)-ASU$, where*

$$\delta = \frac{p^{-t} + \epsilon}{p^{-(t-1)} - \epsilon}.$$

Because of Theorem 5 we see that t-wise $\epsilon-$biased arrays yield authentication codes $(\delta, t)-$ASU. As an example start from $\epsilon-$biased arrays constructed out of Hermitian codes via Theorem 7. Consider the Hermitian code of dimension $k \sim \frac{1}{2}p^m$ defined over $\mathbb{F}_{p^m}$, where m is even (see Section 11). The parameters of the code are $[p^{3m/2}, \frac{1}{2}p^m, p^{3m/2} - p^m]$. Theorem 7 yields an $\epsilon-$biased $(p^{5m/2}, \frac{1}{2}p^m)_p-$array, where $\epsilon = p^{-m/2}$. Apply Theorem 2 with codes $[p^{2i}, p^{2i} - 3i, 4]_p$ (recall $l(3,p) \leq 3/2$ from Subsection 8.1) Here $k \sim \frac{1}{2}p^m \sim 3i$. The result is a

$$\text{3-wise } p^{-m/2} - \text{ biased } (p^{5m/2}, p^{\frac{1}{3}p^m})_p - \text{array.}$$

In case $m = 6$ we obtain a $(p^{15}, p^{\frac{1}{3}p^6})_p-$array, which is $(\delta, 3)-$ASU, where $\delta \sim 2/p$. This can be used to authenticate a source with $\frac{1}{3}p^6 log(p)$ bits with $15 log(p)$ key bits, where the same authenticator is used for 3 consecutive messages. Choose for instance $p \sim 2^{41}$. The opponent's probability of success is limited by 2^{-40} in this case. We can authenticate a source of up to 2^{249} bits, using 615 key bits. The method of [28] would require more than 750 bits of key in this situation. Further improvements can be achieved by using Theorem 11 and more sophisticated AG-codes.

10 Covering arrays

An $(n, N, t)_p-$**covering array** is an $(n, N)_p-$array with the property that every projection to a set of t columns contains every t-tuple from $\mathbb{F}_p^t$ at least once. Covering arrays are also known as $t-$**independent sets** and as **universal sets.**

It follows from Theorem 6 that a $t-$wise $p^{-t}-$biased array is necessarily an $(n, N, t)_p-$ covering array. Theorem 16 shows that there are efficient constructions for such covering arrays, where $n = O(t log_p(N) \cdot p^{tb})$ for every $b > 2$.

Definition 11. *Consider $(n, N, t)_p-$covering arrays. Define*

$$c_p(t) = lim\ inf_N \frac{n}{log_p(N)} \text{ where } N \longrightarrow \infty.$$

We have seen that there are effective constructions of such covering arrays showing $c_p(t) = O(tp^{bt})$ for every $b > 2$, and non-constructively $c_p(t) = O(tp^{2t})$. A non-constructive upper bound is in [22]. In case of strength $t = 2$

the mechanism of **Sperner capacity** solves the asymptotic problem completely, for all q. The result in [21] is

$$c_q(2) = \frac{qlog_2q}{2} \text{ for all } q.$$

The most general construction principle for covering arrays involves perfect hash families.

Definition 12. *An $(N, n)_q-$array is* **perfectly $t-$hashing** *if for every set of t columns there is a row such that the entries in this set of columns are different. We use the notation $PHF(N; n, q, t)$. The most suggestive interpretation is obtained when we interpret the array as a family of N mappings from an n-set to a $q-$set. The defining property says then that for every set of t elements in the domain we can find a function whose restriction to these t points is injective.*

Theorem 19. *If a set of n_1 functions from an N-set to an $m-$set is perfectly $t-$hashing and a set of n_2 functions from the $m-$set to a q-set is t-covering, then the composition is a set of n_1n_2 functions from an N-set to a $q-$set, which is again t-covering, hence an $(n_1n_2, N, t)_q-$covering array.*

This has been formulated in the language of mappings, where the statement is indeed obvious. In order to apply Theorem 19 we need perfect hash families and small covering arrays. The only effective construction of perfect hash families seems to follow from the following remark of Alon [1], which makes use of codes again.

Lemma 5 (Alon). *Let $\mathcal{C}$ be a q-ary code of length N with minimum distance d and n codewords. Write the codewords as columns of an array. Then a $PHF(N; n, q, t)$ is obtained if t is chosen such that*

$$\frac{d}{N} > 1 - 1/\binom{t}{2}.$$

The main construction in [14] is an application of Theorem 19, our friends the AG-codes following from Theorem 8 furnishing the perfect hash families. **Intersecting codes** are used in order to construct small covering arrays to which to apply the construction. The main result of [14] is an efficient construction for families of binary covering arrays showing $c_2(t) = O(t^3 2^{2t})$. This is rather close to what our mechanism yields in the binary case.

11 Hermitian codes for the user

An important ingredient in our constructions are AG-codes. The effeciency of these constructions hinge upon the effectiveness of the description of the AG-codes used. The first layer of the construction in [20] are the Hermitian codes. We describe how to obtain generator matrices for the Hermitian codes. A comparison with the Reed-Solomon codes shows that Hermitian codes are just as effectively described.

Consider the field extension $\mathbb{F}_{q^2} \mid \mathbb{F}_q$ and the corresponding trace tr and norm N, where $tr(x) = x + x^q, N(x) = x^{q+1}$. Our codes are defined over $\mathbb{F}_{q^2}$ and have length q^3.

The coordinates are parametrized by the pairs (α, β), where $N(\alpha) = tr(\beta)$. So we need to calculate traces and norms of all elements in the field and to list all these pairs in some order. There are q^3 such pairs.

The general build-up: We construct a $(q^3 - g, q^3)-$ matrix G with entries from $\mathbb{F}_{q^2}$. Here $g = \binom{q}{2}$. The first k rows of G generate the $k-$dimensional Hermitian code. It has parameters

$$[q^3, k, q^3 - k + 1 - g]_{q^2}.$$

The pole-order test: For $n = 0, 1, 2, \ldots$ we have to decide if n is a **pole-order** or not. If n is a pole-order we determine its **coordinate vector** (i, j). This is done as follows: Let r be the remainder of n mod q, where $0 \leq r \leq q - 1$ and $-s$ the (negative) remainder of n mod $q + 1$, where $0 \leq s \leq q$. Then n is a pole-order if and only if

$$x = \frac{n - r}{q} \geq \frac{n + s}{q + 1} = y.$$

If $n \geq 2g$, then the pole-order test does not need to be performed. Every such number is a pole-order. If n is a pole-order, then $n = (q+1)i + qj$, where $i = (x - y)q + r, j = s$. The coordinate vector of n is (i, j).

Constructing the rows of G : Let $u_1 = 0, u_2 = q, u_3 = q + 1 \ldots$ be the first pole-orders. If u_k has coordinate-vector (i, j), then the entry of row k of G in coordinate (α, β) is $\beta^i \alpha^j$.

We conclude that the use of Hermitian codes requires the usual field arithmetic, just as Reed-Solomon codes.

References

[1] N.Alon: *Explicit construction of exponential sized families of k-independent sets,* Discrete Mathematics **58**(1986),191-193.

[2] N.Alon,J.Bruck,J,Naor,M.Naor,R.Roth: *Construction of asymptotically good low-rate error-correcting codes through pseudo-random graphs,* IEEE Transaction on Information Theory **38**(1992),509-516.

[3] N.Alon,U.Feige,A.Wigderson,D.Zuckerman: *Derandomized graph products,* Computational Complexity(1995),60-75.

[4] N.Alon,O.Goldreich,J.Håstad,R.Peralta: *Simple constructions of almost k-wise independent random variables,* Random Structures in Algorithms **3**(1992), prelim. version Symposium 31st FOCS 1990,544-553.

[5] N.Alon and V.D.Milman: λ_1, *isoperimetric inequalities for graphs, and superconcentrators,* JCTB **38**(1985),73-88.

[6] Y Azar, R. Motwani and J. Naor: *Approximating probability distributions using small sample spaces,* Combinatorics **18**(1998), 151-171.

[7] J.Boyar, G.Brassard and R.Peralta: *Subquadratic zero-knowledge,* JACM **42**(1995),1169-1193.

[8] C.H.Bennett,G.Brassard and J-M.Roberts: *Privacy amplification by public discussion,* SIAM J.Comput. 17,2(1988),210-229.

[9] G.Brassard, C.Crépeau and M.Santha: *Oblivious transfers and intersecting codes,* IEEE Transactions on Information Theory **42**(1996),1769-1780.

[10] J.Bierbrauer: *Bounds on orthogonal arrays and resilient functions,* Journal of Combinatorial Designs **3** (1995),179-183.

[11] A.R.Calderbank and P.C.Fishburn: *Maximal three-independent subsets of* $\{0,1,2\}^n$, Designs, Codes and Cryptography **4** (1994),203-211.

[12] J.L.Carter and M.N.Wegman: *Universal Classes of Hash Functions,* J.Computer and System Sci. 18(1979), 143-154.

[13] B.Chor,O.Goldreich,J.Håstad,J.Friedman, S.Rudich and R.Smolensky: *The bit extraction problem or t-resilient functions,* Proc. 26th Symp. on Foundations of Computer Science (1985),396-407.

[14] G.D.Cohen and G.Zémor: *Intersecting codes and independent families,* IEEE Transaction on Information Theory **40**(1994),1872-1881.

[15] Ilya Dumer: *Nonbinary double-error-correcting codes designed by means of algebraic verieties,* IEEE Transactions on Information Theory **41**(1995),1657-1666.

[16] Y.Edel and J.Bierbrauer: *Twisted BCH-codes,* Journal of Combinatorial Designs **5** (1997), 377-389.

[17] Yves Edel and J.Bierbrauer: *Quantum twisted codes,* Journal of Combinatorial Designs, to appear.

[18] O.Gabber and Z.Galil: *Explicit construction of linear sized superconcentrators,* Journal on Computer and System Science **22**(1981),407-420.

[19] A.Gal: *A characterization of span program size and improved lower bounds for monotone span programs,* Proceedings 13^{th} Symposium of the Theory of Computing(1998),429-437.

[20] A.Garcia and H.Stichtenoth: *A tower of Artin-Schreier extensions of function fields attaining the Drinfeld-Vladut bound,* Inventiones Mathematicae **121**(1995),211-222.

[21] L. Gargano, J. Körner and U. Vaccaro: *Sperner Capacities,* Graphs and Combinatorics **9**(1993),31-46.

[22] A.Godbole, D.E.Skipper and R.A.Sunley: *t-Covering arrays: upper bounds and Poisson approximations,* Combinatorics, Probability and Computing **5**(1996),105-118.

[23] O.Goldreich and A.Wigderson: *Tiny families of functions with random properties: a quality-size trade-off for Hashing,* Proceedings STOC 1994, 574-583.

[24] K.Gopalakrishnan and D.R.Stinson: *A simple analysis of the error probability of two-point based sampling,* Information Processing Letters **60**(1996),91-96.

[25] M.Hattori,R.J.McEliece and G.Solomon: *Subspace subcodes of Reed-Solomon codes, IEEE Transactions on Information Theory* **44**(1998),1861-1880.

[26] N.Kahale : *Better expansion for Ramanujan graphs, Proc. 33rd FOCS 1992,* 398-404.

[27] N.Kahale : *Eigenvalues and expansion of regular graphs, Journal of the ACM* **42** (1995), 1091-1106.

[28] K.Kurosawa, T.Johansson and D.Stinson: *Almost k-wise independent sample spaces and their cryptologic applications, Lecture Notes in Computer Science* **1233**(1997),409-421 (Advances in Cryptology, Eurocrypt 97).

[29] A.Lubotzky, R.Phillips and P.Sarnak: *Explicit expanders and the Ramanujan conjectures, Proceedings 18th ACM Symposium Theory of Computing 1986*, 240-246.

[30] J.Naor and M.Naor: *Small-bias probability spaces: efficient constructions and applications, SIAM Journal on Computing* **22**(1993),838-856, preliminary version: Proceedings STOC 1990, 213-223.

[31] M.Naor and O.Reingold: *On the construction of pseudo-random permutations:Luby-Rackoff revisited,* Proceedings STOC 29(1997),189-199,
complete version under
http://theory.lcs.mit.edu/ tcryptol/homepage.html

[32] N. Nisan: *Extracting randomness: How and Why - A Survey, Complexity 96.*

[33] M.A.Tsfasman, S.G.Vladut and T.Zink: *Modular curves, Shimura curves and Goppa Codes better than the Varshamov-Gilbert bound, Mathematische Nachrichten* **109**(1982),21-28.

[34] V.Vazirani: *Randomness, adversaries and computation,* Ph.D. Thesis, University of California, Berkeley (1986).

[35] M.N.Wegman and J.L.Carter: *New Hash Functions and Their Use in Authentication and Set Equality,* J.Computer and System Sci. 22(1981),265-279.