Title

Pseudorandomness via the Discrete Fourier Transform

Journal

SIAM Journal on Computing, 47(6)

ISSN

0097-5397

Authors

Gopalan, Parikshit
Kane, Daniel M
Meka, Raghu

Publication Date

2018

DOI

10.1137/16m1062132

Peer reviewed

Pseudorandomness via the discrete Fourier transform

Parikshit Gopalan
Microsoft Research,
Mountain View, California, USA.
Email: parik@microsoft.com

Daniel Kane
UCSD,
San Diego, California, USA.
Email: dakane@ucsd.edu

Raghu Meka
UCLA,
Los Angeles, California, USA.
Email: raghum@cs.ucla.edu

Abstract

We present a new approach to constructing unconditional pseudorandom generators against classes of functions that involve computing a linear function of the inputs. We give an explicit construction of a pseudorandom generator that fools the *discrete Fourier transforms* of linear functions with seed-length that is nearly logarithmic (up to polyloglog factors) in the input size and the desired error parameter. Our result gives a single pseudorandom generator that fools several important classes of tests computable in logspace that have been considered in the literature, including halfspaces (over general domains), modular tests and combinatorial shapes. For all these classes, our generator is the first that achieves near logarithmic seed-length in both the input length and the error parameter. Getting such a seed-length is a natural challenge in its own right, which needs to be overcome in order to derandomize RL — a central question in complexity theory.

Our construction combines ideas from a large body of prior work, ranging from a classical construction of [1] to the recent gradually increasing independence paradigm of [2]–[4], while also introducing some novel analytic machinery which might find other applications.

Keywords

pseudorandomness; Fourier transform; probability; halfspaces; limited independence; Chernoff bound

I. Introduction

A central goal of computational complexity is to understand the power that randomness adds to efficient computation. The main questions in this area are whether $\mathsf{BPP} = \mathsf{P}$ and $\mathsf{RL} = \mathsf{L}$, which respectively assert that randomness can be eliminated from efficient computation, at the price of a polynomial slowdown in time, and a constant blowup in space. It is known that proving $\mathsf{BPP} = \mathsf{P}$ will imply strong circuit lower bounds that seem out of reach of current techniques. In contrast, proving $\mathsf{RL} = \mathsf{L}$, could well be within reach. Indeed, *bounded-space algorithms* are a natural computational model for which we know how to construct strong *pseudo-random generators*, PRGs, unconditionally.

Let RL denote the class of randomized algorithms with $O(\log n)$ work space which can access the random bits in a read-once pre-specified order. Nisan [5] devised a PRG of seed length $O(\log^2(n/\varepsilon))$ that fools RL with error ε. This generator was subsequently used by Nisan [6] to show that $\mathsf{RL} \subseteq \mathsf{SC}$ and by Saks and Zhou [7] to prove that RL can be simulated in space $O(\log^{3/2} n)$. Constructing PRGs with the optimal $O(\log(n/\varepsilon))$ seed length for this class and showing that $\mathsf{RL} = \mathsf{L}$ is arguably the outstanding open problem in derandomization (which might not require a breakthrough in lower bounds). Despite much progress in this area [4], [8]–[16], there are few cases where we can improve on Nisan's twenty year old bound of $O(\log^2(n/\varepsilon))$ [5].

A. Fourier shapes

A conceptual contribution of this work is to propose a class of functions in RL which we call *Fourier shapes* that unify and generalize the problem of fooling many natural classes of test functions that are computable in logspace and involve computing linear combinations of (functions of) the input variables. In the following, let $\mathbb{C}_1 = \{z : |z| \leq 1\}$ be the unit-disk in the complex plane.

Definition 1. *A (m, n)-*Fourier shape *$f : [m]^n \to \mathbb{C}_1$ is a function of the form $f(x_1, \ldots, x_n) = \prod_{j=1}^n f_j(x_j)$ where each $f_j : [m] \to \mathbb{C}_1$. We refer to m and n as the alphabet size and the dimension of the Fourier shape respectively.*

Clearly, (m, n)-Fourier shapes can be computed with $O(\log n)$ workspace, as long as the bit-complexity of $\log(f_j)$ is logarithmic for each j; a condition that can be enforced without loss of generality. Since our goal is to fool functions $f : \{0,1\}^n \to \{0,1\}$, it might be unclear why we should consider complex-valued functions (or larger domains). The answer comes from the discrete Fourier transform which maps integer random variables to $\mathbb{C}_1$. Concretely consider a Boolean function $f : \{0,1\}^n \to \{0,1\}$ of the form $f(x) = g(\sum_j w_j x_j)$ where $x \in \{0,1\}^n$, $w_j \in \mathbb{Z}$, and $g : \mathbb{Z} \to \{0,1\}$ is a *simple* function like a threshold or a mod function. To fool such a function f, it suffices to *fool* the linear function

$w(x) = \sum_j w_j x_j$. A natural way to establish the closeness of distributions on the integers is via the discrete Fourier transform. The discrete Fourier transform of $w(x)$ at $\alpha \in [0,1]$ is given by

$$\phi_\alpha(w(x)) = \exp(2\pi i \alpha \cdot w(x)) = \prod_{j=1}^{n} \exp(2\pi i \alpha w_j x_j)$$

which is a Fourier shape.

Allowing a non-binary alphabet m not only allows us to capture more general classes of functions (such as combinatorial shapes), it makes the class more robust. For instance, given a Fourier shape $f : \{0,1\}^n \to \mathbb{C}$, if we consider inputs bits in blocks of length $\log(m)$, then the resulting function is still a Fourier shape over a larger input domain $[m]$ (in dimension $n/\log(m)$). This allows certain compositions of PRGs and simplifies our construction even for the case $m = 2$.

1) PRGs for Fourier shapes and their applications.: A PRG is a function $\mathcal{G} : \{0,1\}^r \to [m]^n$. We refer to r as the seed-length of the generator. We say $\mathcal{G}$ is *explicit* if the output of $\mathcal{G}$ can be computed in time $\mathrm{poly}(n)$.[1]

Definition 2. *A PRG $\mathcal{G} : \{0,1\}^r \to [m]^n$ fools a class of functions $\mathcal{F} = \{f : [m]^n \to \mathbb{C}\}$ with error ε (or ε-fools $\mathcal{F}$) if for every $f \in \mathcal{F}$,*

$$\left| \mathop{\mathbb{E}}_{x \in_u [m]^n} [f(x)] - \mathop{\mathbb{E}}_{y \in_u \{0,1\}^r} [f(\mathcal{G}(y))] \right| < \varepsilon.$$

We motivate the problem of constructing PRGs for Fourier shapes by discussing how they capture a variety of well-studied classes like halfspaces (over general domains), combinatorial rectangles, modular tests and combinatorial shapes.

PRG*s for halfspaces.:* Halfspaces are functions $h : \{0,1\}^n \to \{0,1\}$ that can be represented as

$$h(x) = \mathbb{1}^+(\langle w, x \rangle - \theta)$$

for some *weight* vector $w \in \mathbb{Z}^n$ and *threshold* $\theta \in \mathbb{Z}$ where $\mathbb{1}^+(a) = 1$ if $a \geq 0$ and 0 otherwise. Halfspaces are of central importance in computational complexity, learning theory and social choice. Lower bounds for halfspaces are trivial, whereas the problem of proving lower bounds against depth-2 TC_0 or halfspaces of halfspaces is a frontier open problem in computational complexity. The problem of constructing explicit PRGs that can fool halfspaces is a natural challenge that has seen a lot of exciting progress recently [17]–[21]. The best known PRG construction for halfspaces is that of Meka and Zuckerman [18] who gave a PRG with seed-length $O(\log n + \log^2(1/\varepsilon))$, which is $O(\log^2(n))$ for polynomially small error. They also showed that PRGs against RL with inverse polynomial error can be used to fool halfspaces, and thus constructing better PRGs for halfspaces is a necessary step towards progress for bounded-space algorithms. However, even for special cases of halfspaces (such as derandomizing the Chernoff bound), beating seed-length $O(\log^2(n))$ has proved difficult.

We show that a PRG for $(2,n)$-Fourier shapes with error ε/n^2 also fools halfspaces with error ε. In particular, PRGs fooling Fourier shapes with polynomially small error also fool halfspaces with small error.

PRG*s for generalized halfspaces.:* PRGs for (m,n)-Fourier shapes give us PRGs for halfspaces not just for the uniform distribution over the hypercube, but for a large class of distributions that have been studied in the literature. We can derive these results in a unified manner by considering the class of *generalized halfspaces*.

Definition 3. *A generalized halfspace over $[m]^n$ is a function $g : [m]^n \to \{0,1\}$ that can be represented as*

$$g(x) = \mathbb{1}^+ \left(\sum_{j=1}^{n} g_j(x_j) - \theta \right).$$

where $g_j : [m] \to \mathbb{R}$ are arbitrary functions for $j \in [n]$ and $\theta \in \mathbb{R}$.

PRGs for (m,n)-Fourier shapes imply PRGs for generalized halfspaces. This in turn captures settings of fooling halfspaces with respect to the Gaussian distribution and the uniform distribution on the sphere [18], [20]–[22], and a large class of product distributions over $\mathbb{R}^n$ [23].

Derandomizing the Chernoff-Hoeffding bound.: A consequence of fooling generalized halfspaces is to *derandomize* Chernoff-Hoeffding type bounds for sums of independent random variables which are ubiquitous in the analysis of randomized algorithms. We state our result in the language of "randomness-efficient samplers" (cf. [24]). Let $X_1, \ldots, X_n$ be independent

[1]Throughout, for a multi-set S, $x \in_u S$ denotes a uniformly random element of S.

random variables over a domain $[m]$ and let $g_1, \ldots, g_n : [m] \to [-1, 1]$ be arbitrary bounded functions. The classical Chernoff-Hoeffding bounds [25] say that

$$\Pr\left[\left|\sum_{i=1}^{n} g_i(X_i) - \sum_{i=1}^{n} \mathbb{E}[g_i(X_i)]\right| \geq t\right] \leq 2\exp(-t^2/4n).$$

There has been a long line of work on showing sharp tail bounds for pseudorandom sequences starting from [26] who showed that similar tail bounds hold under limited independence. But all previous constructions for the polynomial small error regime required seed-length $O(\log^2(n))$. PRGs for generalized halfspaces give Chernoff-Hoeffding tail bounds with polynomially small error, with seed-length $O(\log(n))$.

PRG*s for modular tests.:* An important class of functions in L is that of *modular tests*, i.e., functions of the form $g : \{0,1\}^n \to \{0,1\}$, where $g(x) = \mathbb{1}(\sum_i a_i x_i \bmod m \in S)$, for $m \leq M$, coefficients $a_i \in \mathbb{Z}_m$ and $S \subseteq \mathbb{Z}_m$. Such a test is computable in L as long as $M \leq \mathrm{poly}(n)$. The case when $m = 2$ corresponds to small-bias spaces, for which optimal constructions were first given in the seminal work of Naor and Naor [1]. The case of arbitrary m was considered by [27] (see also [28]), their generator gives seed-length $\tilde{O}(\log(n/\varepsilon) + \log^2(M))$. Thus for $M = \mathrm{poly}(n)$, their generator does not improve on Nisan's generator even for constant error ε. PRGs fooling $(2, n)$-Fourier shapes with polynomially small error fools modular tests.

PRG*s for combinatorial shapes.: Combinatorial shapes* were introduced in the work of [29] as a generalization of combinatorial rectangles and to address fooling linear sums in statistical distance. These are functions $f : [m]^n \to \{0,1\}$ of the form

$$f(x) = h\left(\sum_{i=1}^{n} g_i(x_i)\right)$$

for functions $g_i : [m] \to \{0,1\}$ and a function $h : \{0, \ldots, n\} \to \{0,1\}$. The best previous generators of [29] and [30] for combinatorial shapes achieve a seed-length of $O(\log(mn) + \log^2(1/\varepsilon))$, $O(\log m + \log(n/\varepsilon)^{3/2})$; in particular, the best previous seed-length for polynomially small error was $O(\log^{3/2}(n))$. PRGs for (m, n)-Fourier shapes with error ε/n imply PRGs for combinatorial shapes.

Combinatorial rectangles are a well-studied subset of combinatorial shapes [31]–[34]. They are functions that can be written as $f(x) = \prod_j \mathbb{1}(x_j \in A_j)$ for some arbitrary subsets $A_j \subseteq [m]$. The best known PRG due to [4], [35] gives a seed-length of $O(\log(mn/\varepsilon)\log\log(mn/\varepsilon))$. Combinatorial rectangles are special cases of Fourier shapes so our PRG for (m, n)-Fourier shapes also fools combinatorial rectangles, but requires a slightly longer seed. The *alphabet-reduction* step in our construction is inspired by the generator of [4], [35].

2) Achieving optimal error dependence via Fourier shapes.: We note that having generators for Fourier shapes with seed-length $\tilde{O}(\log(n))$ even when ε is polynomially small is essential in our reductions: we sometimes need error $\varepsilon/\mathrm{poly}(n)$ for Fourier shapes in order to get ε error for our target class of functions. Once we have this, starting with ε a sufficiently small polynomial results in polynomially small error for the target class of functions.

We briefly explain why previous techniques based on *limit theorems* were unable to achieve polynomially small error with optimal seed-length, by considering the setting of halfspaces under the uniform distribution on $\{0,1\}^n$. Fooling halfspaces is equivalent to fooling all linear functions $L(x) = \sum_i w_i x_i$ in Kolmogorov or cdf distance. Previous work on fooling halfspaces [17], [18] relies on the Berry-Esséen theorem, a quantiative form of the central limit theorem, to show that the cdf of *regular* linear functions is close to that of the Gaussian distribution, both under the uniform distribution and under the pseudorandom distribution. However, even for the majority function (which is the most regular linear function), the discreteness of $\sum_i x_i$ means that the Kolmogorov distance from the Gaussian distribution is $1/\sqrt{n}$, even when x is uniformly random. Approaches that show closeness in cdf distance by comparison to the Gaussian distribution seem unlikely to give polynomially small error with optimal seed-length.

We depart from the *derandomized limit theorem* approach taken by several previous works [17], [18], [23], [29], [36], [37] and work directly with the Fourier transform. A crucial insight (that is formalized in Lemma IX.1) is that fooling the Fourier transform of linear forms to within polynomially small error implies polynomially small Kolmogorov distance.

B. Our results

Our main result is the following:

Theorem I.1. *There is an explicit generator $\mathcal{G} : \{0,1\}^r \to [m]^n$ that fools all (m, n)-Fourier shapes with error ε, and has seed-length $r = O(\log(mn/\varepsilon) \cdot (\log\log(mn/\varepsilon))^2)$.*

We now state various corollaries of our main result starting with fooling halfspaces.

Corollary I.2. *There is an explicit generator $\mathcal{G} : \{0,1\}^r \to \{0,1\}^n$ that fools halfspaces over $\{0,1\}^n$ under the uniform distribution with error ε, and has seed-length $r = O(\log(n/\varepsilon)(\log\log(n/\varepsilon))^2)$.*

The best previous generator due to [18] had a seed-length of $O(\log n + \log^2(1/\varepsilon))$, which is $O(\log^2 n)$ for polynomially small error ε.

We also get a PRG with similar parameters for generalized halfspaces.

Corollary I.3. *There is an explicit generator $\mathcal{G} : \{0,1\}^r \to [m]^n$ that ε-fools* generalized halfspaces *over $[m]^n$, and has seed-length $r = O(\log(mn/\varepsilon) \cdot (\log\log(mn/\varepsilon))^2)$.*

From this we can derive PRGs with seed-length $O(\log(n/\varepsilon)(\log\log(n/\varepsilon))^2)$ for fooling halfspaces with error ε under the Gaussian distribution and the uniform distribution on the sphere. Indeed, we get the following bound for arbitrary product distributions over $\mathbb{R}^n$, which depends on the 4^{th} moment of each co-ordinate.

Corollary I.4. *Let X be a product distribution on $\mathbb{R}^n$ such that for all $i \in [n]$,*

$$\mathbb{E}[X_i] = 0, \mathbb{E}[X_i^2] = 1, \mathbb{E}[X_i^4] \leq C.$$

There exists an explicit generator $\mathcal{G} : \{0,1\}^r \to \mathbb{R}^n$ such that if $Y = \mathcal{G}(z)$, then for every halfspace $h : \mathbb{R}^n \to \{0,1\}$,

$$|\mathbb{E}[h(X)] - \mathbb{E}[h(Y)]| \leq \varepsilon.$$

The generator G has seed-length $r = O(\log(nC/\varepsilon)(\log\log(nC/\varepsilon))^2)$.

This improves on the result of [23] who obtained seedlength $O(\log(nC/\varepsilon)\log(C/\varepsilon)$ for this setting via a suitable modification of the generator from [18].

The next corollary is a near-optimal derandomization of the Chernoff-Hoeffding bounds. To get a similar guarantee, the best known seed-length that follows from previous work [18], [23], [26] was $O(\log(mn) + \log^2(1/\varepsilon))$.

Corollary I.5. *Let $X_1, \ldots, X_n$ be independent random variables over the domain $[m]$. Let $g_1, \ldots, g_n : [m] \to [-1,1]$ be arbitrary bounded functions. There exists an explicit generator $G : \{0,1\}^r \to [m]^n$ such that if $(Y_1, \ldots, Y_n) = G(z)$ where $z \in_u \{0,1\}^r$, then Y_i is distributed identically to X_i and*

$$\Pr\left[\left|\sum_{i=1}^n g_i(Y_i) - \sum_{i=1}^n \mathbb{E}[g_i(Y_i)]\right| \geq t\right] \leq 2\exp(-t^2/2n) + \varepsilon.$$

G has seed-length $r = O(\log(mn/\varepsilon)(\log\log(mn/\varepsilon))^2)$.

We get the first generator for fooling modular tests whose dependence on the modulus M is near-logarithmic. The best previous generator from [27] had a seed-length of $\tilde{O}(\log(n/\varepsilon) + \log^2(M))$, which is $\tilde{O}(\log^2 n)$ for $M = \mathrm{poly}(n)$.

Corollary I.6. *There is an explicit generator $\mathcal{G} : \{0,1\}^r \to \{0,1\}^n$ that fools all linear tests modulo m for all $m \leq M$ with error ε, and has seed-length $r = O(\log(Mn/\varepsilon) \cdot (\log\log(Mn/\varepsilon))^2)$.*

Finally, we get a generator with near-logarithmic seedlength for fooling combinatorial shapes. [29] gave a PRG for combinatorial shapes with a seed-length of $O(\log(mn) + \log^2(1/\varepsilon))$. This was improved recently by De [30] who gave a PRG with seed-length $O(\log m + \log(n/\varepsilon)^{3/2})$; in particular, the best previous seed-length for polynomially small error was $O((\log(n)^{3/2})$.

Corollary I.7. *There is an explicit generator $\mathcal{G} : \{0,1\}^r \to [m]^n$ that fools (m,n)-combinatorial shapes to error ε and has seed-length $r = O(\log(mn/\varepsilon)(\log\log(mn/\varepsilon))^2)$.*

C. Other related work

Starting with the work of Diakonikolas et al. [17], there has been a lot of interest in constructing PRGs for halfspaces and related classes such as intersections of halfspaces and polynomial threshold functions over the domain $\{\pm 1\}^n$ [18]–[20], [23], [36]–[38]. Rabani and Shpilka [39] construct optimal hitting set generators for halfspaces over $\{\pm 1\}^n$; hitting set generators are weaker than PRGs.

Another line of work gives PRGs for halfspaces for the uniform distribution over the sphere (*spherical caps*) or the Gaussian distribution. For spherical caps, Karnin, Rabani and Shpilka [22] gave a PRG with a seed-length of $O(\log n + \log^2(1/\varepsilon))$. For the Gaussian distribution, [20] gave a PRG which achieves a seed-length of $O(\log n + \log^{3/2}(1/\varepsilon))$. Recently, [21] gave the first PRGs for these settings with seedlength $O((\log(n/\varepsilon))(\log\log(n/\varepsilon)))$. Fooling halfspaces over the hypercube is known to

be *harder* than the Gaussian setting or the uniform distribution on the sphere; hence our result gives a construction with similar parameters up to a $O(\log \log n)$ factor. At a high level, [21] also uses a iterative dimension reduction approach like in [2]–[4]; however, the final construction and its analysis are significantly different from ours. Gopalan *et al.* [23] gave a generator fooling halfspaces under product distributions with bounded fourth moments, whose seed-length is $O(\log(n/\varepsilon)\log(1/\varepsilon))$.

The present work completely subsumes a manuscript of the authors which essentially solved the special-case of derandomizing Chernoff bounds and a special class of halfspaces [40]. Some proofs are omitted from these conference proceedings and can be found in the full version [41].

II. Proof overview

We describe our PRG for Fourier shapes as in Theorem I.1. The various corollaries are derived from this Theorem using properties of the discrete Fourier transform of integer-valued random variables.

Let us first consider a very simple PRG: $O(1)$-wise independent distributions over $[m]^n$. At a glance, it appears to do very poorly as it is easy to express the parity of a subset of bits as a Fourier shape and parities are not fooled even by $(n-1)$-wise independence. The starting point for our construction is that bounded independence does fool a special but important class of Fourier shapes, namely those with polynomially small *total variance*.

For a complex valued random variable Z, define the variance of Z as

$$\sigma^2(Z) = \mathbb{E}\left[|Z - \mathbb{E}[Z]|^2\right] = \mathbb{E}[|Z|^2] - |\,\mathbb{E}[Z]|^2.$$

It is easy to verify that

$$\sigma^2(Z) + |\,\mathbb{E}[Z]|^2 = E[|Z|^2],$$

so that if Z takes values in $\mathbb{C}_1$, then

$$\sigma^2(Z) + |\,\mathbb{E}[Z]|^2 \leq 1.$$

The *total-variance* of a (m,n)-Fourier shape $f : [m]^n \to \mathbb{C}_1$ with $f(x) = \prod_{j=1}^n f_j(x_j)$ is defined as

$$\mathsf{Tvar}(f) = \sum_j \sigma^2(f_j(x_j)).$$

To gain some intuition for why this is a natural quantity, note that $\mathsf{Tvar}(f)$ gives an easy upper bound on the expectation of a Fourier shape:

$$\left|\mathop{\mathbb{E}}_{x\in[m]^n}[f(x)]\right| = \prod_j |\mathbb{E}[f_j(x_j)]| \leq \prod_j \sqrt{1-\sigma^2(f_j(x_j))} \leq \exp(-\mathsf{Tvar}(f)/2). \tag{1}$$

This inequality suggests a natural dichotomy for the task of fooling Fourier shapes. It suggests that high variance shapes where $\mathsf{Tvar}(f) \gg \log(1/\varepsilon)$ are *easy* in the sense that $\mathbb{E}[f] \ll \varepsilon$ is small for such Fourier shapes. So a PRG for such shapes only needs to ensure that $\mathbb{E}[f]$ is also sufficiently small under the pseudorandom output.

To complement the above, we show that if the total-variance $\mathsf{Tvar}(f)$ is very small, then generators based on limited independence do fairly well. Concretely, our main technical lemma says that limited independence fools products of bounded (complex-valued) random variables, provided that the sum of their variances is small. The lemma is a generalization to the unit complex disc of a result proved for real-valued random variables bounded in the range $[-1,1]$ in [35].

Lemma II.1. *Let $Y_1, \ldots, Y_n$ be k-wise independent random variables taking values in $\mathbb{C}_1$. Then,*

$$\left|\mathbb{E}[Y_1 \cdots Y_n] - \prod_{j=1}^n \mathbb{E}[Y_j]\right| \leq \exp(O(k)) \left(\frac{\sum_{j=1}^n \sigma^2(Y_j)}{\sqrt{k}}\right)^{\Omega(k)}.$$

We defer discussion of the proof to Section II-D, and continue the description of our PRG construction. Recall that we are trying to fool a (m,n)-Fourier shape $f : [m]^n \to \mathbb{C}_1$ with $\mathsf{Tvar}(f) \leq O(\log(1/\varepsilon)$ to error $\varepsilon = \mathrm{poly}(1/nm)$. It is helpful to think of the desired error ε as being fixed at the beginning and staying unchanged through our iterations, while m and n change during the iterations. Generating k-wise independent distributions over $[m]^n$ takes $O(k\log(mn))$ random bits. Thus if we use $k = O(\log(1/\varepsilon))$-wise independence, we would achieve error ε, but with seed-length $O(\log(1/\varepsilon)\log(mn))$ rather than $O(\log(1/\varepsilon))$.

On the other hand, if $\mathsf{Tvar}(f) \leq 1/(mn)^c$ for a fixed constant c, then choosing $k = O(\log(1/\varepsilon)/(\log mn))$-wise independence is enough to get error ε while also achieving seed-length $O(k\log(mn)) = O(\log(1/\varepsilon))$ as desired. We exploit this observation by combining the use of limited independence with the recent *iterative-dimension-reduction* paradigm of

[2]–[4]. Our construction reduces the problem of fooling Fourier shapes with $\mathsf{Tvar}(f) \leq O(\log(1/\varepsilon))$ through a sequence of iterations to fooling Fourier shapes where the total variance is polynomially small in m, n in each iteration and then uses limited independence in each iteration.

To conclude our high-level description, our generator consists of three modular parts. The first is a generator for Fourier shapes with high total variance: $\mathsf{Tvar}(f) \geq \mathrm{poly}(\log(1/\varepsilon))$. We then give two reductions to handle low variance Fourier shapes: an alphabet-reduction step reduces the alphabet m down to $\sqrt{m}$ and leaves n unchanged, and a dimension-reduction step that reduces the dimension from n to $\sqrt{n}$ while possibly blowing up the alphabet to $\mathrm{poly}(1/\varepsilon)$. We describe each of these parts in more detail below.

A. Fooling high-variance Fourier shapes

We construct a PRG with seed-length $O(\log(mn/\varepsilon)\log\log(1/\varepsilon))$ which ε-fools (m,n)-Fourier shapes f when $\mathsf{Tvar}(f) \geq (\log(1/\varepsilon))^C$ for some sufficiently large constant C. We build the generator in two steps.

In the first step, we build a PRG with seed-length $O(\log(mn))$ which achieves constant error for (m,n)-Fourier shapes f with $\mathsf{Tvar}(f) \geq 1$. In the second step, we drive the error down to ε as follows. We hash the coordinates into roughly $(\log(1/\varepsilon))^{O(1)}$ buckets, so that for at least $\Omega(\log(1/\varepsilon))$ buckets, f restricted to the coordinates within the bucket has total-variance at least 1. We use the PRG with constant error within each bucket, while the seeds across buckets are recycled using a PRG for small-space algorithms. This construction is inspired by the construction of small-bias spaces due to Naor and Naor [1]; the difference being that we use generators for space bounded algorithms for amplification, as opposed to expander random walks as done in [1].

B. Alphabet-reduction

The next building block in our construction is *alphabet-reduction* which helps us assume without loss of generality that the alphabet-size m is polynomially bounded in terms of the dimension n. This is motivated by the construction of [4].

Concretely, we show that constructing an ε-PRG for (m,n)-Fourier shapes can be reduced to that of constructing an ε'-PRG for (n^4,n)-Fourier shapes for $\varepsilon' \approx \varepsilon/(\log m)$. The alphabet-reduction step consists of $(\log\log m)$ steps where in each step we reduce fooling (m,n)-Fourier shapes for $m > n^4$, to that of fooling $(\sqrt{m},n)$-Fourier shapes, at the cost of $O(\log(m/\varepsilon))$ random bits.

We now describe a single step that reduces the alphabet from m to $\sqrt{m}$. Consider the following procedure for generating a uniformly random element in $[m]^n$:

- For $D \approx \sqrt{m}$, sample uniformly random subsets
$$S_1 = \{X[1,1], X[1,2], \ldots, X[D,1]\}, \ldots, S_n = \{X[1,n], X[2,n], \ldots, X[D,n]\} \subseteq [m].$$
- Sample $Y = (Y_1, \ldots, Y_n)$ uniformly at random from $[D]^n$.
- Output $(Z_1, \ldots, Z_n)$, where $Z_j = X[Y_j, j]$.

Our goal is to derandomize this procedure. The key observation is that once the subsets $S_1, \ldots, S_n$ are chosen, we are left with a (D,n)-Fourier shape as a function of Y. So the choice of Y can be derandomized using a PRG for Fourier shapes with alphabet $[D]$, and it suffices to derandomize the choice of the X's. A calculation shows that (because the Y's are uniformly random), derandomizing the choice of the X's reduces to that of fooling a Fourier shape of total-variance $1/m^{\Omega(1)}$. Lemma II.1 implies that this can be done with limited independence.

C. Dimension-reduction for low-variance Fourier shapes

We show that constructing an ε-PRG for (n^4,n)-Fourier shapes f with $\mathsf{Tvar}(f) \leq \mathrm{poly}(\log(mn/\varepsilon))$ can be reduced to that of ε'-fooling $(\mathrm{poly}(n/\varepsilon), \sqrt{n})$-Fourier shapes for $\varepsilon' \approx \varepsilon/\log n$. Note that here we decreased the dimension at the expense of increasing the alphabet-size. However, this can be fixed by employing another iteration of alphabet-reduction. This is the reason why considering (m,n)-Fourier shapes for arbitrary m helps us even if we were only trying to fool $(2,n)$-Fourier shapes. The dimension-reduction proceeds as follows:

1) We first hash the coordinates into roughly $\sqrt{n}$ buckets using a k-wise independent hash function $h \in_u \mathcal{H} = \{h : [n] \leftarrow [\sqrt{n}]\}$ for $k \approx O(\log(n/\varepsilon)/\log n)$. Note that this only requires $O(\log(n/\varepsilon))$ random bits.
2) For the coordinates within each bucket we use a k'-wise independent string in $[m]^n$ for $k' \approx O(\log(n/\varepsilon)/\log n)$. We use true independence across buckets. Note that this requires $\sqrt{n}$ independent seeds of length $r = O(\log(n/\varepsilon))$.

While the above process requires too many random bits by itself, it is easy to analyze. We then reduce the seed-length by observing that if we fix the hash function h, then what we are left with as a function of the seeds used for generating the symbols in each bucket is a $(2^r \leq \mathrm{poly}(n/\varepsilon), \sqrt{n})$-Fourier shape. So rather than using independent seeds, we can use the output of a generator for such Fourier shapes.

The analysis of the above construction again relies on Lemma II.1. The intuition is that since $\mathsf{Tvar}(f) \leq \mathrm{poly}(\log(n/\varepsilon))$, and we are hashing into $\sqrt{n}$ buckets, for most hash functions h the Fourier shape restricted to each bucket has variance $O(1/n^c)$ for some fixed constant $c > 0$. By Lemma II.1, limited independence fools such Fourier shapes.

D. Main Technical Lemma

The lemma is a generalization to the unit complex disc of a result proved for real-valued random variables bounded in the range $[-1, 1]$ in [35]. The generalization is substantial and seems to require different proof techniques.

We first consider the case where the Y_j's not only have small total-variance, but also have small absolute deviation from their means. Concretely, let $Y_j = \mu_j(1 + Z_j)$ where $\mathbb{E}[Z_j] = 0$ and $|Z_j| \leq 1/2$. In this case, we do a variable change $W_j = \log(1 + Z_j)$ (taking the principal branch of the algorithm) to rewrite

$$\prod_j Y_j = \prod_j \mu_j(1 + Z_j) = \prod_j \mu_j \cdot \exp\left(\sum_j W_j\right).$$

We then argue that $\exp(\sum_j W_j)$ can be approximated by a polynomial $P(W_1, \ldots, W_n)$ of degree less than k with small expected error. The polynomial P is obtained by truncating the Taylor series expansion of the $\exp(\ )$ function. Once, we have such a low-degree polynomial approximator, the claim follows as limited independence fools low-degree polynomials.

To handle the general case where Z_j's are not necessarily bounded, we use an inclusion-exclusion argument and exploit the fact that with high probability, not many of the Z_j's (say more than $k/2$) will deviate too much from their expectation. We leave the details to the actual proof.

III. Preliminaries

We set up some notation. For $v \in \mathbb{R}^n$ and a hash function $h : [n] \to [m]$, define

$$h(v) = \sum_{j=1}^{m} \|v_{|h^{-1}(j)}\|_2^4. \tag{2}$$

Let $\mathbb{C}_1 = \{z : z \in \mathbb{C}, |z| \leq 1\}$ be the unit disk in the complex plane. For a complex valued random variable Z,

$$Var(Z) \equiv \sigma^2(Z) \equiv \mathbb{E}\left[|Z - \mathbb{E}[Z]|^2\right].$$

Unless otherwise stated c, C denote universal constants. Throughout we assume that n is sufficiently large and that $\delta, \varepsilon > 0$ are sufficiently small. For positive functions f, g, h we write $f = g + O(h)$ when $|f - g| = O(h)$. For a integer-valued random variable Z, its Fourier transform is given as follows: for $\alpha \in [0, 1]$, $\hat{Z}(\alpha) = \mathbb{E}[\exp(2\pi i \alpha Z)]$. Further, given the Fourier coefficients $\hat{Z}(\alpha)$, one can compute the probability density function of Z as follows: for any integer j,

$$\Pr[Z = j] = \int_0^1 \exp(2\pi i j \alpha) \hat{Z}(\alpha)\, d\alpha.$$

Definition. *For $n, m, \delta > 0$ we say that a family of hash functions $\mathcal{H} = \{h : [n] \to [m]\}$ is δ-biased if for any $r \leq n$ distinct indices $i_1, i_2, \ldots, i_r \in [n]$ and $j_1, \ldots, j_r \in [m]$,*

$$\Pr_{h \in_u \mathcal{H}}\left[h(i_1) = j_1 \wedge h(i_2) = j_2 \wedge \cdots \wedge h(i_r) = j_r\right] = \frac{1}{m^r} \pm \delta.$$

We say that such a family is k-wise independent if the above holds with $\delta = 0$ for all $r \leq k$.

We say that a distribution over $\{\pm 1\}^n$ is δ-biased or k-wise independent if the corresponding family of functions $h : [n] \to [2]$ is.

Such families of functions can be generated efficiently using small seeds.

Fact III.1. *For $n, m, k, \delta > 0$, there exist explicit δ-biased families of hash functions $\mathcal{H} = \{h : [n] \to [m]\}$ that can be generated efficiently from a seed of length $s = O(\log(n/\delta))$. There are also, explicit k-wise independent families that can be generated efficiently from a seed of length $s = O(k \log(nm))$.*

Taking the pointwise sum of such generators modulo m gives a family of hash functions that is both δ-biased and k-wise independent generated from a seed of length $s = O(\log(n/\delta) + k \log(nm))$.

A. Basic Results

We start with the simple observation that to δ-fool an (m,n)-Fourier shape f, we can assume the functions in f have bit-precision $2\log_2(n/\delta)$. This observation will be useful when we use PRGs for small-space machines to fool Fourier shapes in certain parameter regimes.

Lemma III.2. *If a* PRG $\mathcal{G} : \{0,1\}^r \to [m]^n$ *δ-fools (m,n)-Fourier shapes $f = \prod_j f_j$ when $\log(f_j)$'s have bit precision $2\log_2(n/\delta)$, then $\mathcal{G}$ fools all (m,n)-Fourier shapes with error at most 2δ.*

Proof: Consider an arbitrary (m,n)-Fourier shape $f : [m]^n \to \mathbb{C}_1$ with $f = \prod_j f_j$. Let $\tilde{f}_j : [m] \to \mathbb{C}_1$ be obtained by truncating the $\log(f_j)$'s to $2\log_2(n/\delta)$ bits. Then, $|f_j(x_j) - \tilde{f}_j(x_j)| \le \delta/n$ for all $x_j \in [m]$. Therefore, if we define $\tilde{f} = \prod_j \tilde{f}_j$, then for any $x \in [m]^n$, (as the f_j's and $\tilde{f}_j$'s are in $\mathbb{C}_1$)

$$\left|f(x) - \tilde{f}(x)\right| = \left|\prod_j f_j(x_j) - \prod_j \tilde{f}(x_j)\right| \le \sum_j \left|f_j(x_j) - \tilde{f}(x_j)\right| \le \delta.$$

The claim now follows as the above inequality holds point-wise and by assumption, $\mathcal{G}$ δ-fools $\tilde{f}$. ∎

We collect some known results about pseudorandomness and prove some other technical results that will be used later.

We shall use PRGs for small-space machines or read-once branching programs (ROBP) of Nisan [5], [9] and Impagliazzo, Nisan and Wigderson [8]. We extend the usual definitions of read-once branching programs to compute complex-valued functions; the results of [5], [9], [8] apply to this extended model readily[2].

Definition 4 ((S,D,T)-ROBP)**.** *An (S,D,T)-ROBP M is a layered directed graph with $T+1$ layers and 2^S vertices per layer with the following properties.*

- *The first layer has a single* start *node and the vertices in the last layer are labeled by complex numbers from $\mathbb{C}_1$.*
- *A vertex v in layer i, $0 \le i < T$ has 2^D edges to layer $i+1$ each labeled with an element of $\{0,1\}^D$.*

A graph M as above naturally defines a function $M : \left(\{0,1\}^D\right)^T \to \mathbb{C}_1$ where on input $(z_1,\ldots,z_T) \in \left(\{0,1\}^D\right)^T$ one traverses the edges of the graph according to the labels $z_1,\ldots,z_T$ and outputs the label of the final vertex reached.

Theorem III.3 ([5], [8])**.** *There exists an explicit* PRG $\mathcal{G}^{INW} : \{0,1\}^r \to \left(\{0,1\}^D\right)^T$ *which ε-fools (S,D,T)-branching programs and has seed-length $r = O(D + S\log T + \log(T/\delta)\cdot(\log T))$.*

Theorem III.4 ([9])**.** *For all $C > 1$ and $0 < c < 1$, there exists an explicit* PRG $\mathcal{G}^{NZ} : \{0,1\}^r \to \left(\{0,1\}^D\right)^T$ *which ε-fools (S,S,S^C)-branching programs for $\varepsilon = 2^{-\log^{1-c} S}$ and has seed-length $r = O(S)$.*

The next two lemmas quantify load-balancing properties of δ-biased hash functions in terms of the ℓ_p-norms of vectors. Proofs can be found in the full version.

Lemma III.5. *Let $p \ge 2$ be an integer. Let $v \in \mathbb{R}^n$ and $\mathcal{H} = \{h : [n] \to [m]\}$ be either a δ-biased hash family for $\delta > 0$ or a p-wise independent family for $\delta = 0$. Then*

$$\mathbb{E}[h(v)^p] \le O(p)^{2p}\left(\frac{\|v\|_2^4}{m}\right)^p + O(p)^{2p}\|v\|_4^{4p} + m^p\|v\|_2^{4p}\delta.$$

Lemma III.6. *For all $v \in \mathbb{R}_+^n$, let $p \ge 2$ be even and $\mathcal{H} = \{h : [n] \to [m]\}$ a p-wise independent family, and $j \in [m]$,*

$$\Pr\left[\left|\left\|v_{|h^{-1}(j)}\right\|_1 - \|v\|_1/m\right| \ge t\right] \le \frac{O(p)^{p/2}\,\|v\|_2^p}{t^p}.$$

IV. Fooling products of low-variance random variables

We show that products of complex-valued random variables are fooled by limited independence if the sum of variances of the random variables is small. The lemma is essentially equivalent to saying that limited independence fools low-variance Fourier shapes.

Lemma IV.1. *Let $Y_1,\ldots,Y_n$ be k-wise independent random variables taking values in $\mathbb{C}_1$. Then,*

$$\left|\mathbb{E}[Y_1\cdots Y_n] - \prod_{j=1}^n \mathbb{E}[Y_j]\right| \le \exp(O(k))\cdot\left(\frac{\sum_j \sigma^2(Y_j)}{k}\right)^{\Omega(k)}.$$

Due to space constraints, the proof is deferred to the full version.

[2]This is because these results in fact give guarantees in terms of statistical distance.

V. A GENERATOR FOR HIGH-VARIANCE FOURIER SHAPES

In this section, we construct a generator that fools Fourier shapes with high variance.

Theorem V.1. *There exists a constant $C > 0$, such that for all $\delta > 0$, there exists an explicit generator $\mathcal{G}_\ell : \{0,1\}^{r_\ell} \to [m]^n$ with seed-length $r_\ell = O(\log(mn/\delta)\log\log(1/\delta))$ such that for all Fourier shapes $f : [m]^n \to \mathbb{C}_1$ with $\mathsf{Tvar}(f) \geq C\log^5(1/\delta)$, we have*

$$\left| \mathop{\mathbb{E}}_{z\sim\{0,1\}^{r_\ell}}[f(\mathcal{G}_\ell(z))] - \mathop{\mathbb{E}}_{X\in_u[m]^n}[f(X)] \right| < \delta.$$

We start with the simple but crucial observation that Fourier shapes with large variance have small expectation.

Lemma V.2. *For any Fourier shape $f : [m]^n \to \mathbb{C}_1$, we have*

$$\left| \mathop{\mathbb{E}}_{X\in_u[m]^n}[f(X)] \right| \leq \exp(-\mathsf{Tvar}(f)/2). \tag{3}$$

Proof: Let $f(x) = \prod_j f_j(x_j)$. Since $f_j(x) \in \mathbb{C}_1$, we have $|f_j(x)| \leq 1$. Let $\mu_j = \mathbb{E}_{X_j\in[m]}[f_j(X_j)]$. For $X \in_u [m]^n$,

$$\sigma_j^2 = \mathbb{E}[|f_j(X_j) - \mu_j|^2] = \mathbb{E}[|f_j(X_j)|^2] - |\mu_j|^2 \leq 1 - |\mu_j|^2.$$

Hence

$$\begin{aligned}\left|\mathop{\mathbb{E}}_{X}[f(X)]\right| = \prod_{j=1}^n |\mu_j| &\leq \prod_{j=1}^n (1-\sigma_j^2)^{1/2} \\ &\leq \exp(-\sum_{j=1}^n \sigma_j^2/2) \leq \exp(-\mathsf{Tvar}(f)/2).\end{aligned}$$

■

We build the generator in two steps. We first build a generator with seed-length $O(\log n)$ which achieves constant error for all f with $\mathsf{Tvar}(f) \geq 1$. In the second step, we reduce the error down to δ. This construction is inspired by a construction of Naor and Naor [1] of small-bias spaces.

A. A generator with constant error

Our goal in this subsection is get a generator with constant error for Fourier shapes where $\mathsf{Tvar}(f) = \Omega(1)$. We start by showing that when $\mathsf{Tvar}(f) = \Theta(1)$ (instead of just $\Omega(1)$), $O(1)$-wise independence is enough to fool f.

Lemma V.3. *For all constants $0 < c_1 < c_2$, there exist $p \in \mathbb{Z}_+$ and $0 < c' < 1$ such that the following holds. For any (m,n)-Fourier shape, f with $\mathsf{Tvar}(f) \in [c_1, c_2]$, and $Z \sim [m]^n$ $2p$-wise independent,*

$$\left|\mathop{\mathbb{E}}_{Z}[f(Z)]\right| < c'.$$

Proof: Let $f = \prod_j f_j$, $X \in_u [m]^n$. Now, by Lemma IV.1 applied to $Y_j = f_j(Z_j)$, we have,

$$|\mathbb{E}[f(Z)] - \mathbb{E}[f(X)]| \leq \exp(O(p))(\mathsf{Tvar}(f)/\sqrt{p})^{\Omega(p)} = \exp(O(p))(c_2/\sqrt{p})^{\Omega(p)}.$$

Note that by taking p to be a sufficiently large constant compared to c_2, we can make the last bound arbitrary small.

On the other hand, by Equation (3),

$$|\mathbb{E}[f(X)]| \leq \exp(-\mathsf{Tvar}(f)/2) \leq \exp(-c_1/2).$$

Therefore,

$$|\mathbb{E}[f(Z)]| \leq \exp(-c_1/2) + \exp(O(p))(c_2/\sqrt{p})^{\Omega(p)} < c'$$

for p sufficiently large constant and some constant $0 < c' < 1$. ■

We reduce the general case of $\mathsf{Tvar}(f) \in [1,n]$ to the case above where $\mathsf{Tvar}(f) = \Theta(1)$ by using the Valiant-Vazirani technique of sub-sampling. For $B \subseteq [n]$ let $\mathsf{Tvar}(f_B) = \sum_{i\in B}\sigma_i^2$. If we sample a random subset $B \subseteq [n]$ with $|B| \approx n/\mathsf{Tvar}(f)$ in a pairwise independent manner, we will get $\mathsf{Tvar}(f_B) = \Theta(1)$ with $\Omega(1)$ probability. Since we do not know $\mathsf{Tvar}(f)$, we sample $\log(n)$ subsets whose cardinalities are geometrically increasing; one of them is likely to satisfy the desired bound.

We set up some notation that will be used in the remainder of this section.

- Assume n is a power of 2, and set $T = \log_2(n) - 1$. Let $\Pi \subseteq \mathbb{S}_n$ be a family of pairwise independent permutations so that $\pi \in_u \Pi$ can be sampled efficiently with $O(\log n)$ random bits. For $0 \le j \le T$, let $B_j = \{\pi(i) : i \in \{2^j, \ldots, 2^{j+1} - 1\}\}$ be the 2^j co-ordinates that land in the j^{th} bucket.
- For $v \in \mathbb{R}^n$, let $v^j = v_{B_j}$ denote the projection of v onto coordinates in bucket j. Similarly, for $x \in [m]^n$, let x^j denote the projection of x to the co-ordinates in B_j.
- Fix an (m, n)-Fourier shape $f : [m]^n \to \mathbb{C}_1$ with $f(x) = \prod_i f_i(x_i)$. Define $f^j : [m]^{B_j} \to \mathbb{C}_1$ as $f^j(x^j) = \prod_{i \in B_j} f_i(x_i)$.

Lemma V.4. *Let $v \in \mathbb{R}^n$ with $\|v\|_2^2 \in [1, n]$, $\|v\|_\infty \le 1$ and $t \in [\log_2 n]$ be such that $n/2^{t+1} \le \|v\|_2^2 \le n/2^t$. Then,*

$$\Pr_{\pi \in_u \Pi}\left[\left\|v^t\right\|_2^2 \in [1/6, 4/3]\right] \ge 7/16.$$

The proof of this lemma is standard (see [41]).

This naturally suggests using an $O(1)$-wise independent distribution within each bucket. But using independent strings across the $\log(n)$ buckets would require a seed of length $O(\log(mn) \cdot (\log n))$. We analyze our generator assuming independence across distinct buckets, but then recycle the seeds using PRGs for space bounded computation to keep the seed-length down to $O(\log(mn))$ (rather than $O(\log^2(n))$).

We now prove the main claim of this subsection.

Lemma V.5. *There exists an explicit generator $\mathcal{G}_1 : \{0,1\}^r \to [m]^n$ with $r = O(\log(mn))$ such that for all Fourier shapes $f : [m]^n \to \mathbb{C}_1$ with $\mathsf{Tvar}(f) \ge 1$, we have*

$$\left|\mathop{\mathbb{E}}_{z \sim \{0,1\}^r}[f(\mathcal{G}_1(z))]\right| \le c.$$

for some constant $0 < c < 1$.

Proof: Let $\pi \in_u \Pi$ and let $Z^j \sim [m]^{2^j}$ be an independent p-wise independent string for a parameter $p = O(1)$ to be chosen later. Define

$$\mathcal{G}_1'(\pi, Z^0, \ldots, Z^T) = Y, \quad \text{where } Y_{B_j} = Z^j \text{ for } j \in \{0, \ldots, T\}.$$

In other words, the generator applies the string Z^j to the coordinates in bucket B_j.

Observe that $f(Y) = \prod_{j=0}^{\log(n)-1} f^j(Z^j)$. Since the Z^j's are independent of each other

$$|\mathbb{E}[f(Y)]| = \left|\prod_{j=0}^{\log(n)-1} \mathbb{E}[f^j(Z^j)]\right| \le \left|\mathbb{E}[f^t(Z^t)]\right|,$$

for any $t \le T$. Applying Lemma V.3 to $v = (\sigma_1(f_1), \ldots, \sigma_n(f_j))$, we get that for some $t \le T$, $\mathsf{Tvar}(f^t) = \|v^t\|_2^2 \in [1/6, 4/3]$ with probability at least $7/16$. Conditioned on this event, Lemma V.3 implies that for p a sufficiently large constant, there exists a constant $c' < 1$ so that $|\mathbb{E}[f^t(Z^t)]| < c'$. Therefore, overall we get

$$|\mathbb{E}[f(Y)]| \le \left|\mathbb{E}[f^t(Z^t)]\right| \le \frac{9}{16} + \frac{7c'}{16} = c'' < 1.$$

We next improve the seed-length of $\mathcal{G}_1'$ using the PRG for ROBPs of Theorem III.4. To this end, note that by Lemma III.2 we can assume that every $\log(f_i(x_i))$, and hence every $\log(f^j(x^j))$, has bit precision at most $O(\log n)$ bits (since our goal is to get error $\delta = O(1)$). Further, each Z^j can be generated efficiently with $O(\log(mn))$ random bits.

Thus, for a fixed permutation π, the computation of $f(\mathcal{G}'(\pi, Z^1, \ldots, Z^T))$ can be done by a (S, D, T)-ROBP where S, T are $O(\log n)$ and $D = O(\log(mn))$: for $j \in \{1, \ldots, T\}$, the ROBP computes $f^j(Z^j)$ and multiplies it to the product computed so far, which can be done using $O(\log n)$ bits of space. Let $\mathcal{G}^{NZ} : \{0,1\}^r \to \left(\{0,1\}^D\right)^T$ be the generator in Theorem III.4 fooling (S, D, T)-ROBPs as above with error $\delta < (1 - c'')/2$. $\mathcal{G}^{NZ}$ has seedlength $O(\log(mn))$. Let

$$\mathcal{G}_1(\pi, z) = \mathcal{G}_1'(\pi, \mathcal{G}^{NZ}(z)).$$

It follows that $|\mathbb{E}[f(\mathcal{G}_1(\pi, z))]| < c$ for some constant $c < 1$. Finally, the seed-length of $\mathcal{G}_1$ is $O(\log(mn))$ as π can be sampled with $O(\log n)$ random bits and the seed-length of $\mathcal{G}^{NZ}$ is $O(\log(mn))$. The lemma is now proved. ∎

B. Reducing the error

We now amplify the error to prove Theorem V.1. The starting point for the construction is the observation that for $X \in_u [m]^n$, $|\mathbb{E}[f(X)]| \leq \exp(-\mathsf{Tvar}(f/2)) \leq \delta$ once $\mathsf{Tvar}(f) \gg \log(1/\delta)$. Therefore, it suffices to design a generator so that $\mathbb{E}[f] \ll \delta$, when $\mathsf{Tvar}(f)$ is sufficiently large.

Our generator will partition $[n]$ into $m = O((\log(1/\delta))^5)$ buckets $B_1, \ldots, B_m$, using a family of hash functions with the following *spreading* property:

Definition 5. *A family of hash functions $\mathcal{H} = \{h : [n] \to [m]\}$ is said to be (B, ℓ, δ)-spreading if for all $v \in [0,1]^n$ with $\|v\|_2^2 \geq B$,*

$$\Pr_{h \in_u \mathcal{H}}[|\{j \in [m] : \left\|v_{h^{-1}(j)}\right\|_2^2 \geq B/2m\}| \geq \ell] \geq 1 - \delta.$$

Using the notation from the last subsection, we write $f(x) = \prod_{j=1}^m f^j(x^j)$ where $f^j(x^j) = \prod_{i \in B_j} f_i(x_i)$. If $\mathsf{Tvar}(f)$ is sufficiently large, then the spreading property guarantees that for at least $\Omega(\log(1/\delta))$ of the buckets B_j, $\mathsf{Tvar}(f^j) \geq 1$. If we now generate $X \in [m]^n$ by setting X_{B_j} to be an independent instantiation of the generator $\mathcal{G}_1$ from Lemma V.3, then we get $\mathbb{E}[f(X)] \ll \delta$. As in the proof of Lemma V.5, we keep the seed-length down to $\tilde{O}(\log(n/\delta))$ by recycling the seeds for the buckets using a PRG for small-space machines.

We start by showing that the desired hash functions can be generated from a small-bias family of hash functions. We show that it satisfies the conditions of the lemma by standard moment bounds (see [41]).

Lemma V.6. *For all constants C_1, there exist constants C_2, C_3 such that following holds. For all $\delta \geq 0$, there exists an explicit hash family $\mathcal{H} = \{h : [n] \to [T]\}$, where $T = C_2 \log^5(1/\delta))$ which is $(C_3 \log^5(1/\delta), C_1 \log(1/\delta), \delta)$-spreading and $h \in_u \mathcal{H}$ can be sampled efficiently with $O(\log(n/\delta))$ bits.*

We are now ready to prove Theorem V.1.

Proof of Theorem V.1: Let $\ell = C \log(1/\delta)$ for some constant to be chosen later and let $\mathcal{H} = \{h : [n] \to [T]\}$ be a (B, ℓ, δ)-spreading family as in Lemma V.6 above for $B = \Theta(\log^5(1/\delta))$ and $T = \Theta(\log^5(1/\delta))$. Let $\mathcal{G}_1 : \{0,1\}^{r_1} \to [m]^n$ be the generator in Lemma V.3. Define a new generator $\mathcal{G}'_\ell : \mathcal{H} \times (\{0,1\}^{r'})^T \to [m]^n$ as:

$$\mathcal{G}'_\ell(h, z^1, \ldots, z^T) = X, \quad \text{where } X_{h^{-1}(j)} = \mathcal{G}_1(z^j) \text{ for } j \in [T].$$

Let $f : [m]^n \to \mathbb{C}_1$ with $\mathsf{Tvar}(f) \geq \max(2T, B)$. For $h \in \mathcal{H}$, let $I = \{j : \mathsf{Tvar} f^j \geq 1\}$. For any fixed $h \in \mathcal{H}$, as the z^j's are independent of each other,

$$|\mathbb{E}[f(X)]| = \prod_{j=1}^m \left|\mathbb{E}[f^j(\mathcal{G}_1(z^j))]\right| \leq \prod_{j \in I} \left|\mathbb{E}[f^j(\mathcal{G}_1(z^j))]\right| \leq c^{|I|},$$

where $c < 1$ is the constant from Lemma V.3. By the spreading property of $\mathcal{H}$, with probability at least $1-\delta$, $|I| \geq C \log(1/\delta)$. Therefore, for C sufficiently large,

$$|\mathbb{E}[f(X)]| \leq \delta + c^{C \log(1/\delta)} < 2\delta.$$

As in Lemma V.5, we recycle the seeds for the various buckets using the PRGs for ROBPs. By Lemma III.2, we may assume that f^j has bit precision at most $O(\log(n/\delta))$ bits. Further note that

$$f(\mathcal{G}'_\ell(h, z^1, \ldots, z^m)) = \prod_{j=1}^m f^j(\mathcal{G}_1(z^j)).$$

For a fixed hash function $h \in \mathcal{H}$, this can be computed by a (S, D, T)-ROBP where $S = O(\log(n/\delta))$ and $D = O(\log(mn))$, corresponding to the various possible seeds for $\mathcal{G}_1$. Let $\mathcal{G}^{INW} : \{0,1\}^r \to \left(\{0,1\}^D\right)^T$ be a generator fooling (S, D, T)-ROBPs as in Theorem III.3 with error δ and define

$$\mathcal{G}_\ell(h, z) = \mathcal{G}'_\ell(h, \mathcal{G}^{INW}(z)).$$

The seed-length is dominated by the seed-length of $\mathcal{G}^{INW}$, which is

$$O(\log(mn/\delta) \log T) = O(\log(mn/\delta) \log\log(1/\delta)).$$

It follows that $|\mathbb{E}[f(\mathcal{G}_\ell(h, z))]| < 3\delta$, whereas for a truly random $Y \in_u [m]^n$,

$$|\mathbb{E}[f(Y)]| \leq \exp(-\mathsf{Tvar}(f)/2) < \delta.$$

The theorem now follows. ∎

VI. Alphabet reduction for Fourier shapes

In this section, we describe our alphabet-reduction procedure, which reduces the general problem of constructing an ε-PRG for (m,n)-Fourier shapes where m could be much larger than n, to that of constructing an $\varepsilon/\log(m)$-PRG for (n^4,n)-Fourier shapes. This reduction is composed of $O(\log\log m)$ steps where in each step we reduce fooling (m,n)-Fourier shapes to fooling $(\sqrt{m},n)$-Fourier shapes. Each of these steps in turn will cost $O(\log(m/\varepsilon))$ random bits, so that the overall cost is $O(\log(m/\varepsilon)\cdot(\log\log m))$. Concretely, we show the following:

Theorem VI.1. *Let $n,\delta>0$ and suppose that for some $r'=r'(n,\delta')$, for all $m'\le n^4$ there exists an explicit generator $\mathcal{G}_{m'}:\{0,1\}^{r_1}\to[m']^n$ which δ'-fools (m',n)-Fourier shapes. For all m, there exists an explicit generator $\mathcal{G}_m:\{0,1\}^r\to[m]^n$ which $(\delta'+\delta)$-fools (m,n)-Fourier shapes with seed-length $r=r'+O(\log(m/\delta)\log\log(m))$.*

Proof: We prove the claim by showing that for $m>n^4$, we can reduce $(\delta+\delta')$-fooling (m,n)-Fourier shapes to that of δ'-fooling $(\sqrt{m},n)$-Fourier shapes with $O(\log(m/\delta))$ additional random bits. The theorem follows by applying the claim $\log\log(m)$ until the alphabet size drops below n^4 when we can use $\mathcal{G}_{m'}$. This costs a total of $r'+O(\log(m/\delta)\log\log(m))$ random bits, and gives error $\delta'+\log\log(m)\delta$. The claim follows by replacing δ with $\delta/\log\log(m)$.

Thus, suppose that $m>n^4$ and for $D=\lfloor\sqrt{m}\rfloor$, we have a generator $\mathcal{G}_D:\{0,1\}^{r_D}\to[D]^n$ which δ'-fools (D,n)-Fourier shapes. The generator $\mathcal{G}_m$ works as follows:

1) Generate a matrix $X\in[m]^{D\times n}$ where
   - Each column of X is from a pairwise independent distribution over $[m]^D$.
   - The different columns are k-wise independent for $k=C\log(1/\delta)/\log(m)$ for some sufficiently large constant C.
2) Generate $Y=(Y_1,\ldots,Y_n)=\mathcal{G}_D(z)\in[D]^n$ for $z\in_u\{0,1\}^{r_D}$.
3) $\mathcal{G}_m$ outputs $Z=(Z_1,\ldots,Z_n)\in[m]^n$ where $Z_j=X[Y_j,j]$ for $j\in[n]$.

Each column of X can be generated using a seed of length $2\log m$. By using seeds for various columns that are k-wise independent, generating X requires seedlength $O(k\log m)=O(\log(1/\delta))$ (as $m>n^2$), while the number of bits needed to generate Z is $r_D+O(\log(1/\delta))$.

Fix an (m,n)-Fourier shape $f:[m]^n\to\mathbb{C}_1$, $f(z)=\prod_j f_j(z_j)$. For $x\in[m]^{D\times n}$, define a (D,n)-Fourier shape $f^x:[D]^n\to\mathbb{C}_1$ by:

$$f^x(y_1,\ldots,y_n)=\prod_{j=1}^n f_j(x[y_j,j]).$$

Note that $f(Z)=f^X(Y)$.

Let X',Y' be random variables distributed uniformly over $[m]^{D\times n}$ and $[D]^n$ respectively. Let $Z'_j=X'[Y'_j,j]$ for $j\in[n]$, so that Z' is uniform over $[m]^n$ and $f(Z')=f^{X'}(Y')$. Our goal is to show that $f(Z')$ and $f(Z)$ are close in expectation. We do this by replacing X' and Y' by X and Y respectively.

That we can replace Y' with Y follows from the pseudorandomness of $\mathcal{G}_D$. For any fixed $x\in[m]^n$, as $\mathcal{G}_D$ fools (D,n)-Fourier shapes,

$$\left|\mathop{\mathbb{E}}_{Y=\mathcal{G}_D(z)}[f^x(Y)]-\mathop{\mathbb{E}}_{Y'\in_u[D]^n}[f^x(Y')]\right|\le\delta'. \tag{4}$$

We now show that for truly random Y', one can replace X by X'. Note that

$$\mathop{\mathbb{E}}_{Y'\in_u[D]^n}[f^x(Y')]=\prod_{j=1}^n\left(\frac{1}{D}\cdot\left(\sum_{\ell=1}^D f_j(x[\ell,j])\right)\right)\equiv B_f(x). \tag{5}$$

where we define the *bias-function* $B_f:[m]^{D\times n}\to\mathbb{C}_1$ as above. We claim that X fools B_f:

$$|\mathbb{E}[B_f(X)]-\mathbb{E}[B_f(X')]|\le\delta. \tag{6}$$

For $j\in[n]$, let

$$A_j=\frac{1}{D}\left(\sum_{\ell=1}^D f_j(X[\ell,j])\right),\ A'_j=\frac{1}{D}\left(\sum_{\ell=1}^D f_j(X'[\ell,j])\right)$$

so that

$$B_f(X)=\prod_{j=1}^n A_j,\ B_f(X')=\prod_{j=1}^n A'_j.$$

Since $f_j(X[\ell, j]) \in \mathbb{C}_1$ for $\ell \in [D]$, it follows that $A_j, A'_j \in \mathbb{C}_1$. Since the $f_j(X[\ell, j])$s are pairwise independent variables,

$$\mathbb{E}[A_j] = \mathbb{E}[A'_j],\ \mathrm{Var}[A_j] = \mathrm{Var}[A'_j].$$

Note that

$$\mathbb{E}[B_f(X')] = \mathbb{E}[\prod_{i=1}^{n} A'_j] = \prod_{j=1}^{n} \mathbb{E}[A'_j] = \prod_{j=1}^{n} \mathbb{E}[A_j],\ \mathbb{E}[B_f(X)] = \mathbb{E}[A_1 \cdots A_n]. \tag{7}$$

The random variables $A_1, \ldots, A_n$ are k-wise independent. Further, we have

$$\mathrm{Var}(A_j) = \frac{1}{D^2} \sum_{\ell=1}^{D} \mathrm{Var}(f_j(X[\ell, j])) = \frac{\sigma^2(f_j)}{D} \leq \frac{1}{D}.$$

Therefore, by Lemma IV.1,

$$\left| \mathbb{E}[A_1 \cdots A_n] - \prod_{j=1}^{n} \mathbb{E}[A_j] \right| \leq \left(\frac{n}{D}\right)^{\Omega(k)} \leq m^{-\Omega(k)} \leq \delta \tag{8}$$

where the second to last inequality follows becase $n \leq m^{1/4}$ and $D \geq \sqrt{m}/2$, and the last holds for $k = C \log(1/\delta)/\log(m)$ for a sufficiently big constant C. Equation 6 now follows from Equations (8) and (7).

Finally,

$$\begin{aligned}
|\mathbb{E}[f(Z)] - \mathbb{E}[f(Z')]| &= \left|\mathbb{E}[f^X(Y)] - \mathbb{E}[f^{X'}(Y')]\right| \\
&= \left|\mathbb{E}[f^X(Y')] - \mathbb{E}[f^{X'}(Y')]\right| + \delta' && \text{Equation (4)} \\
&= |\mathbb{E}[B_f(X)] - \mathbb{E}[B_f(X')]| + \delta' && \text{Equation (5)} \\
&\leq \delta + \delta'. && \text{Equation (6)}
\end{aligned}$$

Hence the theorem is proved. ∎

VII. Dimension reduction for low-variance Fourier shapes

We next describe our *dimension reduction* step for low-variance Fourier shapes. We start with an (m, n)-Fourier shape where $m \leq n^4$ and $\mathsf{Tvar}(f) \leq \log(n/\delta)^c$. We show how one can reduce the dimension to $t = \sqrt{n}$, at a price of a blowup in the alphabet size m' which now becomes $(n/\delta)^c$ for some (large) constant c.

Theorem VII.1. *Let $\delta > 0$, $n > 0$ and $t = \lceil \sqrt{n} \rceil$. There is a constant c and $m' \leq (n/\delta)^c$ such that the following holds: if there exists an explicit PRG $\mathcal{G}' : \{0,1\}^{r'} \to [m']^t$ with seed-length $r' = r'(n, \delta')$ which δ'-fools (m', t)-Fourier shapes, then there exists an explicit generator $\mathcal{G} : \{0,1\}^r \to [m]^n$ with seed-length $r = r' + O(\log(n/\delta))$ which $(\delta + \delta')$-fools (m, n)-Fourier shapes f with $m \leq n^4$ and $\mathsf{Tvar}(f) \leq n^{1/9}$.*

We first set up some notation. Assume that we have fixed a hash function $h : [n] \to [t]$. For $x \in [m]^n$ and $j \in [t]$, let x^j denote the projection of x onto co-ordinates in $h^{-1}(j)$. For an (m, n)-Fourier shape $f : [m]^n \to \mathbb{C}_1$ with $f = \prod_{i=1}^{n} f_i$, let

$$f^j(x^j) = \prod_{i:h(i)=j} f_i(x_i)$$

$$\text{so that } \ f(x) = \prod_{j=1}^{t} f^j(x^j).$$

We start by constructing an easy to analyze generator G_1 which hashes co-ordinates into buckets using k-wise independence and then uses independent k-wise independent strings within a bucket. Let

$$k = C \frac{\log(n/\delta)}{\log(n)} \tag{9}$$

where C will is a sufficiently large constant. Let $\mathcal{H} : \{[n] \to t\}$ be a k-wise independent family of hash functions. Let $G_0 : \{0,1\}^{r_0} \to [m]^n$ be a k-wise independent generator over $[m]^n$. Define a new generator $G_1 : \mathcal{H} \times (\{0,1\}^{r_0})^t \to [m]^n$ as:

$$G_1(h, z_1, \ldots, z_t) = Z, \text{where } Z^j = G_0(z_j)\ \forall\ j \in [t]. \tag{10}$$

We argue that G_1 fools (m,n)-Fourier shapes with small total variance as in the theorem. Our analysis proceeds as follows:

- With high probability over $h \in_u \mathcal{H}$, each of the f^j's has low variance except for a few heavy co-ordinates (roughly $\mathsf{Tvar}(f)/t$ after dropping $k/2$ heavy coordinates).
- Within each bin we have k-wise independence, whereas the distributions across bins are independent. So even conditioned on the heavy co-ordinates in a bin, the remaining distribution in the bin is $k/2$-wise independent. Hence each f^j is fooled by Lemma IV.1.

However, the seed-length of G_1 is prohibitively large: since we use independent seeds across the various buckets, the resulting seed-length is $O(\sqrt{n}\log(n/\delta))$. The crucial observation is that we can *recycle* the seeds for various buckets using a generator that fools (m',t)-Fourier shapes with $m' = 2^{r_0} = \mathrm{poly}(n/\delta)$ and $t = O(\sqrt{n})$. Given such a generator $\mathcal{G}' : \{0,1\}^{r'} \to [m']^t$ which δ-fools (m',t)-Fourier shapes, our final generator for small-variance Fourier shapes is $\mathcal{G}_s : \mathcal{H} \times \{0,1\}^{r'} \to [m]^n$ is defined as

$$\mathcal{G}(h,w) = G_1(h, \mathcal{G}'(w)). \tag{11}$$

It is worth mentioning that even though the original Fourier shape $f : [m]^n \to \mathbb{C}_1$ has low total variance, the generator $\mathcal{G}'$ needs to fool all (m',t)-Fourier shapes, not just those with low variance.

A. Analysis of the dimension-reduction step

For $\alpha > 0$, to be chosen later, let $L = \{j \in [n] : \sigma^2(f_j) \geq \alpha\}$ denote the α-large indices and $S = [n] \setminus L$ denote the small indices. We call a hash function $h \in \mathcal{H}$ (α,β)-*good* if the following two conditions hold for every bin $h^{-1}(j)$ where $j \in [t]$:

1) The bin does not have too many large indices: $|h^{-1}(j) \cap L| \leq k/2$.
2) The small indices in the bin have small total variance:

$$\sum_{\ell \notin L : h(\ell) = j} \sigma^2(f_\ell) \leq \beta.$$

Using standard moment bounds for k-wise independent hash functions one can show that $h \in_u \mathcal{H}$ is (α,β)-good with probability at least $1 - n^{-\Omega(k)}$ for $\alpha = n^{-\Omega(1)}$ and $\beta = n^{-\Omega(1)}$. We defer the proof of the following Lemma to [41].

Lemma VII.2. *Let $\mathsf{Tvar}(f) \leq n^{1/9}$ and let $\mathcal{H} = \{h : [n] \to [t]\}$ be a k-wise independent family of hash functions for $t = \Theta(\sqrt{n})$. Then $h \in \mathcal{H}$ is $(n^{-1/3}, n^{-1/36})$-good with probability $1 - O(k)^{k/2} n^{-\Omega(k)}$.*

We next argue that if $h \in \mathcal{H}$ is (α,β)-good then, k-wise independence is sufficient to fool f^j for each $j \in [t]$.

Lemma VII.3. *Let $h \in \mathcal{H}$ be (α,β)-good, and let $j \in [t]$. For $Z' \sim [m]^n$ k-wise independent, and $Z'' \in_u [m]^n$,*

$$\left|\mathbb{E}[f^j(Z')] - \mathbb{E}[f^j(Z'')]\right| \leq \exp(O(k)) \cdot \beta^{\Omega(k)}.$$

Proof: Fix $j \in [t]$. By relabelling coordinates, let us assume that $h^{-1}(j) = \{1,\ldots,n_j\}$ and $L \cap h^{-1}(j) = \{1,\ldots,r\}$, where $r \leq k/2$. As Z' is k-wise independent, $(Z'_1,\ldots,Z'_r)$ is uniformly distributed over $[m]^r$. We couple Z' and Z'' by taking $Z'_i = Z''_i$ for $i \leq r$. Even after conditioning on these values, $Z'_{r+1},\ldots,Z'_{n_j}$ are $k/2$-wise independent.

Let $Y_\ell = f_\ell(Z'_\ell)$ for $\ell \in \{r+1,\ldots,n_j\}$. As h is (α,β)-good,

$$\sum_{\ell=r+1}^{n/t} \sigma^2(Y_\ell) \leq \beta.$$

Therefore, by Lemma IV.1,

$$\left| \mathbb{E}\left[\prod_{\ell=r+1}^{n_j} Y_\ell\right] - \prod_{\ell=r+1}^{n_j} \mathbb{E}[Y_\ell] \right| \leq \exp(O(k)) \cdot \beta^{\Omega(k)}. \tag{12}$$

But since $Z'_\ell = Z''_\ell$ for $\ell \le r$, we have

$$\begin{aligned}
\mathbb{E}[f^j(Z')] &= \prod_{\ell=1}^{r} \mathbb{E}[f^\ell(Z'_\ell)]\, \mathbb{E}[\prod_{\ell=r+1}^{n} Y_\ell], \\
\mathbb{E}[f^j(Z'')] &= \prod_{\ell=1}^{r} \mathbb{E}[f^\ell(Z'_\ell)] \prod_{\ell=r+1}^{n_j} \mathbb{E}[Y_\ell], \\
\left|\mathbb{E}[f^j(Z')] - \mathbb{E}[f^j(Z'')]\right| &= \left|\prod_{\ell=1}^{r} \mathbb{E}[f^\ell(Z'_\ell)]\, \mathbb{E}[\prod_{\ell=r+1}^{n} Y_\ell] - \prod_{\ell=1}^{r} \mathbb{E}[f^\ell(Z'_\ell)] \prod_{\ell=r+1}^{n_j} \mathbb{E}[Y_\ell]\right| \\
&\le \left|\mathbb{E}[\prod_{\ell=r+1}^{n_j} Y_\ell] - \prod_{\ell=r+1}^{n_j} \mathbb{E}[Y_\ell]\right| && \text{Since } |f^\ell(Z'_\ell)| \le 1 \\
&\le \exp(O(k)) \cdot \beta^{\Omega(k)}. && \text{Equation (12)}
\end{aligned}$$

■

We use these lemmas to prove Theorem VII.1.

Proof of Theorem VII.1:

Let $f : [m]^n \to \mathbb{C}_1$ be a Fourier shape with $\mathsf{Tvar}(f) \le n^{1/9}$. Let G_1 be the generator in Equation (10) with parameters as above. We condition on $h \in_u \mathcal{H}$ being $(n^{-1/3}, n^{-1/36})$-good; by Lemma VII.2 this only adds an additional $O(k)^{k/2} n^{-\Omega(k)}$ to the error. We fix such a good hash function h.

Recall that $G_1(h, z_1, \ldots, z_t) = Z$ where $Z^j = G_0(z_j)$ for $j \in [t]$. Since the z_js are independent, so are the Z^j's. Hence,

$$\mathbb{E}[f(G_1(h, z^1, \ldots, z^t))] = \prod_{j=1}^{t} \mathop{\mathbb{E}}_{h} \left[f^j(Z^j)\right].$$

By Lemma VII.3, for $(n^{-1/3}, n^{-1/36})$-good h, if $Y \in_u [m]^n$, then

$$\begin{aligned}
&\left|\prod_{j=1}^{t} \mathbb{E}\left[f^j(Z^j)\right] - \prod_{j=1}^{t} \mathbb{E}\left[f^j(Y^j)\right]\right| \\
&\le \sum_{r=0}^{t-1} \left|\prod_{j=1}^{r} \mathbb{E}\left[f^j(Y^j)\right] \prod_{j=r+1}^{t} \mathbb{E}\left[f^j(Z^j)\right] - \prod_{j=1}^{r+1} \mathbb{E}\left[f^j(Y^j)\right] \prod_{j=r+2}^{t} \mathbb{E}\left[f^j(Z^j)\right]\right| \\
&\le \sum_{r=0}^{t-1} \left|\mathbb{E}\left[f^{r+1}(Z^{r+1})\right] - \left[f^{r+1}(Y^{r+1})\right]\right| \\
&\le \exp(O(k)) \cdot O(tn^{-k/36}).
\end{aligned}$$

Combining the above equations we get that for $Y \in_u [m]^n$,

$$|\mathbb{E}[f(G_1(h, z_1, \ldots, z_t))] - \mathbb{E}[f(Y)]| \le O(k)^{k/2} n^{-\Omega(k)} + \exp(O(k)) \cdot O(tn^{-k/36}) \le \delta \tag{13}$$

where the last inequality holds by taking C in Equation (9) to be a sufficiently large constant.

We next derandomize the choice of the z^j's by using a PRG for appropriate Fourier shapes. Let r_0 be the seed-length of the generator G_0 obtained by setting $k = C \log(n/\delta)/(\log n)$ as above, and let c be such that $r_0 \le c \log(n/\delta)$. Let

$$m' = 2^{r_0} \le \left(\frac{n}{\delta}\right)^c$$

and identify $[m']$ with $\{0,1\}^{r_0}$. Given a hash function $h \in \mathcal{H}$, let us define $\bar{f}^j : [m'] \to \mathbb{C}_1$ for $j \in [t]$ and $\bar{f} : [m']^n \to \mathbb{C}_1$ as

$$\bar{f}^j(z_j) = f^j(G_0(z_j)),\ \bar{f}(z) = \prod_{i=1}^{t} \bar{f}^j(z_j)$$

respectively. Observe that $\bar{f}$ is a Fourier shape, and

$$f(G_1(h, z_1, \ldots, z_t)) = \prod_{j=1}^{t} f^j(G_0(z_j)) = \bar{f}(z).$$

By assumption, we have an explicit generator $\mathcal{G}' : \{0,1\}^{r'} \to [m']^t$ which δ'-fools (m', t)-Fourier shapes. We claim that $\mathcal{G} : \mathcal{H} \times \{0,1\}^{r'} \to [m]^n$ defined as

$$\mathcal{G}_s(h, w) = G_1(h, \mathcal{G}'(w))$$

$(\delta' + \delta)$ fools small-variance (m, n)-Fourier shapes.

Since $\mathcal{G}'$ fools (m', t)-Fourier shapes,

$$|\mathbb{E}[f(\mathcal{G}(h, w))] - \mathbb{E}[f(G_1(h, z_1, \ldots, z_t))]| \leq \delta'.$$

By Equation (13), whenever $\mathsf{Tvar}(f) \leq \log(n/\delta)^C$,

$$|\mathbb{E}[f(G_1(h, z_1, \ldots, z_t))] - \mathbb{E}[f(Z')]| \leq \delta.$$

Combining these equations,

$$|\mathbb{E}[f(\mathcal{G}(h, w))] - \mathbb{E}[f(Z')]| \leq \delta' + \delta.$$

The seed-length required for $\mathcal{G}_s$ is $O(\log(n/\delta))$ for h and r' for w. ∎

VIII. Putting things together

We put the pieces together and prove our main theorem, Theorem I.1. We show the following lemma which allows simultaneous reduction in both the alphabet and the dimension, going from fooling (m, n)-Fourier shapes to fooling $(n^2, \lceil\sqrt{n}\rceil)$-Fourier shapes.

Lemma VIII.1. *Let $\delta > 0$, $n > \log^C(1/\delta)$ for some sufficiently large constant C, and $t = \lceil\sqrt{n}\rceil$. If there exists an explicit* PRG *$\mathcal{G}'' : \{0,1\}^{r''} \to [m'']^t$ with seed-length $r'' = r''(n, \delta)$ which δ-fools (m'', t)-Fourier shapes for all $m'' \leq n^2$, then there exists an explicit generator $\mathcal{G} : \{0,1\}^r \to [m]^n$ with seed-length $r = r'' + O(\log(mn/\delta)\log\log(mn))$ which 4δ-fools (m, n)-Fourier shapes.*[3]

Proof: Let r'' be the seed-length required for $\mathcal{G}''$ to have error δ. Let $m' \leq (n/\delta)^c$ be as in the statement of VII.1. Applying Theorem VI.1 to $\mathcal{G}''$, we get a generator $\mathcal{G}'$ with seedlength $r'' + O(\log(n/\delta)\log\log(n/\delta))$ that $\delta' = 2\delta$-fools $(m', \sqrt{n})$ Fourier shapes. Invoking Theorem VII.1 with $\mathcal{G}'$, we get an explicit generator $\mathcal{G}_s : \{0,1\}^{r_s} \to [m]^n$ which 3δ fools (m, n)-Fourier shapes $f : [m]^n \to \mathbb{C}_1$ with $\mathsf{Tvar}(f) \leq n^{1/9}$ and $m \leq n^4$, with seed-length

$$r_s = r'' + O(\log(n/\delta)\log\log(n/\delta)).$$

For $m \leq n^4$, let $\mathcal{G}_\ell : \{0,1\}^{r_\ell} \to [m]^n$ be a generator for large Fourier shapes as in Theorem V.1, which δ-fools (m, n)-Fourier shapes $f : [m]^n \to \mathbb{C}_1$ with $\mathsf{Tvar}(f) \geq C\log^5(1/\delta)$. Since $m \leq n^4$, this generator requires seed-length

$$r_\ell = O(\log(n/\delta)\log\log(1/\delta)).$$

Define the generator

$$\mathcal{G}_{\ell\oplus s}(w_1, w_2) = \mathcal{G}_\ell(w_\ell) \oplus \mathcal{G}_s(w_s)$$

where the seeds $w_\ell \in \{0,1\}^{r_\ell}$ and $w_s \in \{0,1\}^{r_s}$ are chosen independently and $\oplus$ is interpreted as the sum mod m. Note that the total seed-length is

$$r_\ell + r_s = r'' + O(\log(n/\delta)\log\log(n/\delta)).$$

We now analyze $\mathcal{G}_{\ell\oplus s}$. Let $Y = \mathcal{G}_\ell(w_1)$ and $Z = \mathcal{G}_\ell(w_2)$ and let $X \in_u [m]^n$. Fix an (m, n)-Fourier shape $f : [m]^n \to \mathbb{C}_1$. We consider two cases based on $\mathsf{Tvar}(f)$:

Case 1: $\mathsf{Tvar}(f) \geq C\log(1/\delta)^5$*.:* For any $z \in [m]^n$, define a new Fourier shape $f_z(y) = f(y \oplus z)$. Then, for any fixed z, Y δ-fools f_z as $\mathsf{Tvar}(f_z) = \mathsf{Tvar}(f) \geq C\log(1/\delta)^5$. Therefore,

$$|\mathbb{E}[f(Y \oplus Z)] - \mathbb{E}[f(X)]| \leq \mathop{\mathbb{E}}_{Z} |\mathbb{E}[f_Z(Y)] - \mathbb{E}[f(X)]| \leq \delta.$$

[3]Comparing this to Theorem VII.1, the main difference is that we do not assume that $\mathsf{Tvar}(f)$ is small. Further, the generator $\mathcal{G}''$ for small dimensions requires $m'' \leq n^2$, and our goal is to fool Fourier shapes in n dimensions with arbitrary alphabet size m.

Case 2: $\mathsf{Tvar}(f) \leq n^{1/9}$.*:* Consider a fixing y of Y and define $f_y(Z) = f(y \oplus Z)$. Then, for any fixed y, Z 3δ-fools f_y as $\mathsf{Tvar}(f_y) \leq n^{1/9}$. Therefore,

$$|\mathbb{E}[f(Y \oplus Z)] - \mathbb{E}[f(X)]| \leq \mathop{\mathbb{E}}_{Y} |\mathbb{E}[f_Y(Z)] - \mathbb{E}[f(X)]| \leq 3\delta.$$

In either case, we have

$$|\mathbb{E}[f(Y \oplus Z)] - \mathbb{E}[f(X)]| \leq 3\delta.$$

Finally, for arbitrary m, by applying Theorem VI.1 to $\mathcal{G}_{\ell \oplus s}$, we get a generator $\mathcal{G} : \{0,1\}^r \to [m]^n$ that 4δ fools (m,n)-Fourier shapes with seed-length

$$O(\log(m/\delta)\log\log(m)) + r_\ell + r_s = r'' + O(\log(nm/\delta)\log\log(nm/\delta)).$$

■

We prove Theorem I.1 by repeated applications of this lemma.

Proof of Theorem I.1: Assume that the final error desired is δ'. Let $\delta = \delta'/4\log\log(n)$. Applying Lemma VIII.1, by using $O(\log(mn/\delta')\log\log(mn/\delta'))$ random bits we reduce fooling (m,n)-Fourier shapes to fooling $(m', \lceil\sqrt{n}\rceil)$-Fourier shapes for $m' \leq n^2$.

We now apply the lemma $O(\log\log n)$ times to reduce to the case of fooling $(\log^C(1/\delta), \log^C(1/\delta))$-Fourier shapes. This can be done by noting that by Lemma III.2 it suffices to fool Fourier shapes with $\log(f_i)$ having $O(\log(1/\delta))$ bits of precision. Such Fourier shapes can be computed by width-$O(\log(1/\delta))$ ROBPs, and thus using the generator from Theorem III.3, we can fool this case with seed length $O(\log(1/\delta)\log\log(1/\delta))$ bits. Since each step requires $O(\log(n/\delta)\log\log(n/\delta)$ random bits, the overall seedlength is bounded by

$$O(\log(mn/\delta)\log\log(mn/\delta) + O(\log(n/\delta)(\log\log(n/\delta))^2).$$

■

IX. Applications of PRGs for Fourier shapes

In this Section, we show how Theorem I.1 implies near optimal PRGs for halfspaces, modular tests and combinatorial shapes. We first prove two technical lemmas relating closeness between Fourier transforms of integer valued random variables to closeness under other metrics. We define the Fourier distance, statistical distance and Kolmogorov distance between two integer-valued random variables respectively as

$$d_{FT}(Z_1, Z_2) = \max_{\alpha \in [0,1]} |\mathbb{E}[\exp(2\pi i \alpha Z_1)] - \mathbb{E}[\exp(2\pi i \alpha Z_2)]|, \tag{14}$$

$$d_{TV}(Z_1, Z_2) = \frac{1}{2}\sum_{j \in \mathbb{Z}} |\Pr(Z_1 = j) - \Pr(Z_2 = j)|, \tag{15}$$

$$d_K(Z_1, Z_2) = \max_{k \in \mathbb{Z}}(|\Pr(Z_1 \leq k) - \Pr(Z_2 \leq k)|) \tag{16}$$

A standard argument shows that for two integer-valued random variables Z_1, Z_2 supported on $[0, N]$, $d_{TV}(Z_1, Z_2) \leq O(\sqrt{N}) \cdot d_{FT}(Z_1, Z_2)$ (see the full version for proof). We next relate Kolmogorov distance to Fourier distance. The key is that unlike the bound on the statistical distance the dependence on N is logarithmic. This difference is crucial to fooling halfspaces with polynomially small error (since N can exponential in the dimension n in this setting).

Lemma IX.1. *Let Z_1, Z_2 be two integer-valued random variables supported on $[-N, N]$. Then,*

$$d_K(Z_1, Z_2) \leq O(\log(N) \cdot d_{FT}(Z_1, Z_2)).$$

Proof: By definition, we have $d_K(Z_1, Z_2) = \max_{-N \leq k \leq N}(|\Pr(Z_1 \leq k) - \Pr(Z_2 \leq k)|)$. We note that

$$\begin{aligned}
\Pr(Z_i \leq k) &= \sum_{j=-N}^{k} \Pr(Z_i = j) \\
&= \sum_{j=-N}^{k} \int_0^1 \exp(-2\pi i j \alpha)\, \mathbb{E}[\exp(2\pi i \alpha Z_i)] d\alpha \\
&= \int_0^1 s(k, N, \alpha)\, \mathbb{E}[\exp(2\pi i \alpha Z_i)] d\alpha
\end{aligned}$$

where

$$s(k,N,\alpha) = \sum_{j=-N}^{k} \exp(-2\pi i j\alpha).$$

It is clear that $|s(k,N,\alpha)| \le 2N$. Further,

$$|s(k,N,\alpha)| = \left|\frac{\exp(-2\pi ik\alpha)(\exp(2\pi i(N+k+1)\alpha)-1)}{\exp(2\pi i\alpha)-1}\right| \le \frac{1}{|\exp(2\pi i\alpha)-1|} \le O(\frac{1}{[\alpha]})$$

where $[\alpha]$ is the distance between α and the nearest integer. Therefore, we have

$$\begin{aligned}
|\Pr(Z_1 \le k) - \Pr(Z_2 \le k)| &\le \int_0^1 |s(k,N,\alpha)|\,|\mathbb{E}[\exp(2\pi i\alpha Z_1)] - \mathbb{E}[\exp(2\pi i\alpha Z_1)]|\,d\alpha \\
&\le \int_0^1 O\left(\min\left(N, \frac{1}{[\alpha]}\right)\right) d_{FT}(Z_1,Z_2)d\alpha \\
&= O(d_{FT}(Z_1,Z_2))\left(\int_0^{1/N} N d\alpha + \int_{1/N}^{1/2} \frac{d\alpha}{\alpha} + \int_{1/2}^{1-1/N} \frac{d\alpha}{1-\alpha} + \int_{1-1/N}^{1} N d\alpha\right) \\
&= O(d_{FT}(Z_1,Z_2)\log(N)).
\end{aligned}$$

■

A. Corollaries of the main result

We combine Lemma IX.1 with Theorem I.1 to derive Corollary I.2, which gives PRGs for halfspaces with polynomially small error from PRGs for $(2,n)$-Fourier shapes.

Proof of Corollary I.2: Let $\mathcal{G} : \{0,1\}^r \to \{\pm 1\}^n$ be a PRG which δ-fools $(2,n)$-Fourier shapes (here we identify [2] with $\{\pm 1\}$ arbitrarily). We claim that $\mathcal{G}$ also fools all halfspaces with error at most $\varepsilon = O(n\log(n)\delta)$.

Let $h : \{\pm 1\}^n \to \{\pm 1\}$ be a halfspace given by $h(x) = \mathbb{1}^+(\langle w,x\rangle - \theta)$. It is well known that we can assume the weights and the threshold θ to be integers bounded in the range $[-N,N]$ for $N = 2^{O(n\log n)}$ (cf. [42]). Let $X \in_u \{\pm 1\}^n$ and $Y = \mathcal{G}(y)$ for $y \in_u \{0,1\}^r$ and $Z_1 = \langle w,X\rangle$, $Z_2 = \langle w,Y\rangle$. Note that Z_1, Z_2 are bounded in the range $[-n\cdot N, n\cdot N]$.

We first claim that

$$d_{FT}(Z_1,Z_2) \le \delta.$$

For $\alpha \in [0,1]$, we define $f_\alpha : \{\pm 1\}^n \to \mathbb{C}_1$ as

$$f_\alpha(x) = \exp(2\pi i\alpha\langle w,x\rangle) = \prod_{j=1}^{n} \exp(2\pi i\alpha w_j x_j) \tag{17}$$

then f_α is a $(2,n)$-Fourier shape. Hence,

$$|\mathbb{E}[f_\alpha(X)] - \mathbb{E}[f_\alpha(Y)]| \le \delta.$$

That $d_{FT}(Z_1,Z_2) \le \delta$ now follows from the definition of Fourier distance, and the fact that $\mathbb{E}[f_\alpha(X)]$ and $\mathbb{E}[f_\alpha(Y)]$ are the Fourier transforms of X and Y at α respectively.

Therefore, by Lemma IX.1 applied to Z_1, Z_2, $d_K(Z_1,Z_2) \le O(n\log n)\delta$. Finally, note that

$$|\mathbb{E}[h(X)] - \mathbb{E}[h(Y)]| \le d_K(\langle w,X\rangle, \langle w,Y\rangle) = d_K(Z_1,Z_2) \le O(n\log n)\delta.$$

The corollary now follows by picking a generator as in Theorem I.1 for $m = 2$ with error $\delta = \varepsilon/(Cn\log n)$ for sufficiently big C. ■

To prove Corollary I.3, we need the following lemma about generalized halfspaces.

Lemma IX.2. *In Definition 3, we may assume that each $g_i(j)$ is an integer of absolute value $(mn)^{O(mn)}$.*

Proof: Let $g : [m]^n \to \{0,1\}$ be a generalized halfspace where the g_is are arbitrary. Embed $[m]^n$ into $\{0,1\}^{mn}$ by sending each $x_i \in [m]$ to $(y_{i,1},\ldots,y_{i,m})$ where $y_{i,j} = 1$ if $x_i = j$ and $y_{i,j} = 0$ otherwise. Note that

$$\sum_{i=1}^{n} g_i(x_i) = \sum_{i=1}^{n}\sum_{j=1}^{m} g_i(j)y_{i,j}$$

However, the halfspace

$$\sum_{i=1}^{n}\sum_{j=1}^{m} g_i(j)y_{i,j} \geq \theta$$

over the domain $\{0,1\}^{mn}$ has a representation where the weights $g_i'(j)$ and θ' are integers of size at most $(mn)^{O(mn)}$. Hence we can replace each $g_i(j)$ in the defintion of g with $g_i'(j)$ without changing its value at any point in $[m]^n$. ∎

We now prove Corollary I.3 giving PRGs for generalized halfspaces over $[m]^n$.

Proof of Corollary I.3: Letting $X \in_u [m]^n$ and letting X' be obtained from a PRG for (m,n)-Fourier shapes with error at most ε, we let $Z_1 = \sum_i g_i(X_i)$ and $Z_2 = \sum_i g_i(X_i')$. By Lemma IX.1 that $d_K(Z_1, Z_2) \leq O(\varepsilon nm \log(nm))$. Picking ε sufficiently small gives our generator for generalized halfspaces. ∎

The proofs of the other corollaries are similar in spirit and are deferred to the full version.

REFERENCES

[1] J. Naor and M. Naor, "Small-bias probability spaces: Efficient constructions and applications," *SIAM J. on Comput.*, vol. 22, no. 4, pp. 838–856, 1993.

[2] D. M. Kane, R. Meka, and J. Nelson, "Almost optimal explicit johnson-lindenstrauss families," in *RANDOM 2011*, 2011, pp. 628–639.

[3] L. E. Celis, O. Reingold, G. Segev, and U. Wieder, "Balls and bins: Smaller hash families and faster evaluation," *SIAM J. Comput.*, vol. 42, no. 3, pp. 1030–1050, 2013.

[4] P. Gopalan, R. Meka, O. Reingold, L. Trevisan, and S. P. Vadhan, "Better pseudorandom generators from milder pseudorandom restrictions," in *53rd Ann. IEEE Symp. on Foundations of Computer Science, FOCS 2012*, 2012, pp. 120–129.

[5] N. Nisan, "Pseudorandom generators for space-bounded computation," *Combinatorica*, vol. 12, no. 4, pp. 449–461, 1992.

[6] ——, "RL $\subseteq$ SC," *Computational Complexity*, vol. 4, no. 1, pp. 1–11, 1994.

[7] M. E. Saks and S. Zhou, "$BPSPACE(S) \subseteq DSPACE(S^{3/2})$," *J. Comput. Syst. Sci.*, vol. 58, no. 2, pp. 376–403, 1999.

[8] R. Impagliazzo, N. Nisan, and A. Wigderson, "Pseudorandomness for network algorithms," in *Proceedings of the Twenty-Sixth Annual ACM Symposium on Theory of Computing, 23-25 May 1994, Montréal, Québec, Canada*, 1994, pp. 356–364.

[9] N. Nisan and D. Zuckerman, "Randomness is linear in space," *J. Comput. System Sci.*, vol. 52, no. 1, pp. 43–52, 1996.

[10] R. Raz and O. Reingold, "On recycling the randomness of states in space bounded computation," in *Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing, May 1-4, 1999, Atlanta, Georgia, USA*, 1999, pp. 159–168.

[11] O. Reingold, "Undirected connectivity in log-space," *J. ACM*, vol. 55, no. 4, 2008.

[12] O. Reingold, L. Trevisan, and S. P. Vadhan, "Pseudorandom walks on regular digraphs and the RL vs. L problem," in *Proceedings of the 38th Annual ACM Symposium on Theory of Computing, Seattle, WA, USA, May 21-23, 2006*, 2006, pp. 457–466.

[13] M. Braverman, A. Rao, R. Raz, and A. Yehudayoff, "Pseudorandom generators for regular branching programs," *SIAM J. Comput.*, vol. 43, no. 3, pp. 973–986, 2014.

[14] J. Brody and E. Verbin, "The coin problem and pseudorandomness for branching programs," in *51th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2010, October 23-26, 2010, Las Vegas, Nevada, USA*, 2010, pp. 30–39.

[15] M. Koucký, P. Nimbhorkar, and P. Pudlák, "Pseudorandom generators for group products: extended abstract," in *Proceedings of the 43rd ACM Symposium on Theory of Computing, STOC 2011, San Jose, CA, USA, 6-8 June 2011*, 2011, pp. 263–272.

[16] A. De, "Pseudorandomness for permutation and regular branching programs," in *Proc. 26th Ann. IEEE Conference on Computational Complexity, CCC 2011*, 2011, pp. 221–231.

[17] I. Diakonikolas, P. Gopalan, R. Jaiswal, R. A. Servedio, and E. Viola, "Bounded independence fools halfspaces," in *Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science (FOCS '09)*, 2009.

[18] R. Meka and D. Zuckerman, "Pseudorandom generators for polynomial threshold functions," *SIAM J. Comput.*, vol. 42, no. 3, pp. 1275–1301, 2013.

[19] D. M. Kane, "A small PRG for polynomial threshold functions of Gaussians," in *IEEE 52nd Annual Symposium on Foundations of Computer Science, FOCS 2011, Palm Springs, CA, USA, October 22-25, 2011*, 2011, pp. 257–266.

[20] ——, "A pseudorandom generator for polynomial threshold functions of Gaussians with subpolynomial seed length," in *IEEE 29th Conference on Computational Complexity, CCC 2014, Vancouver, BC, Canada, June 11-13, 2014*, 2014, pp. 217–228.

[21] P. Kothari and R. Meka, "Almost-optimal pseudorandom generators for spherical caps," in *STOC 2015*, 2015, to appear in STOC 2015.

[22] Z. S. Karnin, Y. Rabani, and A. Shpilka, "Explicit dimension reduction and its applications," *SIAM J. Comput.*, vol. 41, no. 1, pp. 219–249, 2012.

[23] P. Gopalan, R. O'Donnell, Y. Wu, and D. Zuckerman, "Fooling functions of halfspaces under product distributions," in *25th Annual IEEE Conference on Computational Complexity*, 2010, pp. 223–234.

[24] D. Zuckerman, "Randomness-optimal oblivious sampling," *Random Struct. Algorithms*, vol. 11, no. 4, pp. 345–367, 1997.

[25] W. Hoeffding, "Probability inequalities for sums of bounded random variables," *Journal of the American Statistical Association*, vol. 58, no. 301, 1963.

[26] J. P. Schmidt, A. Siegel, and A. Srinivasan, "Chernoff-hoeffding bounds for applications with limited independence," *SIAM J. Discrete Math.*, vol. 8, no. 2, pp. 223–250, 1995.

[27] S. Lovett, O. Reingold, L. Trevisan, and S. P. Vadhan, "Pseudorandom bit generators that fool modular sums," in *RANDOM 2009*, 2009, pp. 615–630.

[28] R. Meka and D. Zuckerman, "Small-bias spaces for group products," in *RANDOM 2009*, 2009, pp. 658–672.

[29] P. Gopalan, R. Meka, O. Reingold, and D. Zuckerman, "Pseudorandom generators for combinatorial shapes," *SIAM J. Comput.*, vol. 42, no. 3, pp. 1051–1076, 2013.

[30] A. De, "Beyond the central limit theorem: asymptotic expansions and pseudorandomness for combinatorial sums," in *IEEE Annual Symposium on Foundations of Computer Science, FOCS 2015, to appear*, 2015.

[31] G. Even, O. Goldreich, M. Luby, N. Nisan, and B. Velickovic, "Efficient approximation of product distributions," *Random Struct. Algorithms*, vol. 13, no. 1, pp. 1–16, 1998.

[32] R. Armoni, M. E. Saks, A. Wigderson, and S. Zhou, "Discrepancy sets and pseudorandom generators for combinatorial rectangles," in *37th Ann. Symp. Foundations of Computer Science, FOCS '96*, 1996, pp. 412–421.

[33] N. Linial, M. Luby, M. E. Saks, and D. Zuckerman, "Efficient construction of a small hitting set for combinatorial rectangles in high dimension," *Combinatorica*, vol. 17, no. 2, pp. 215–234, 1997. [Online]. Available: http://dx.doi.org/10.1007/BF01200907

[34] C. Lu, "Improved pseudorandom generators for combinatorial rectangles," *Combinatorica*, vol. 22, no. 3, pp. 417–434, 2002.

[35] P. Gopalan and A. Yehudayoff, "Inequalities and tail bounds for elementary symmetric polynomials with applications," 2014. [Online]. Available: http://arxiv.org/abs/1402.3543

[36] I. Diakonikolas, D. Kane, and J. Nelson, "Bounded independence fools degree-2 threshold functions," in *Proceedings of the 51st Annual IEEE Symposium on Foundations of Computer Science (FOCS '10)*, 2010.

[37] P. Harsha, A. Klivans, and R. Meka, "An invariance principle for polytopes," *J. ACM*, vol. 59, no. 6, p. 29, 2012.

[38] D. M. Kane, "k-independent Gaussians fool polynomial threshold functions," in *Proceedings of the 26th Annual IEEE Conference on Computational Complexity, CCC 2011, San Jose, California, June 8-10, 2011*, 2011, pp. 252–261.

[39] Y. Rabani and A. Shpilka, "Explicit construction of a small epsilon-net for linear threshold functions," *SIAM J. Comput.*, vol. 39, no. 8, pp. 3501–3520, 2010.

[40] P. Gopalan, D. Kane, and R. Meka, "Pseudorandomness for conecntration bounds and signed majorities," 2014, available on the ArXiv. [Online]. Available: http://arxiv.org/abs/1411.4584

[41] ——, "Pseudorandomness via the discrete fourier transform," 2015, full version of this paper. [Online]. Available: http://arxiv.org/abs/1411.4584

[42] P. M. Lewis and C. L. Coates, *Threshold Logic*. John Wiley, New York, 1967.