Electronic Colloquium on Computational Complexity, Report No. 71 (2025)

Pseudorandom bits for non-commutative programs

Chin Ho Lee[*]
North Carolina State University

Emanuele Viola[†]
Northeastern University

June 2, 2025

Abstract

We obtain new explicit pseudorandom generators for several computational models involving groups. Our main results are as follows:

1. We consider read-once group-products over a finite group G, i.e., tests of the form $\prod_{i=1}^{n} g_i^{x_i}$ where $g_i \in G$, a special case of read-once permutation branching programs. We give generators with optimal seed length $c_G \log(n/\varepsilon)$ over any p-group. The proof uses the small-bias plus noise paradigm, but derandomizes the noise to avoid the recursion in previous work. Our generator works when the bits are read in any order. Previously for any non-commutative group the best seed length was $\geq \log n \log(1/\varepsilon)$, even for a fixed order.
2. We give a reduction that "lifts" suitable generators for group products over G to a generator that fools *width-w block products*, i.e., tests of the form $\prod g_i^{f_i}$ where the f_i are arbitrary functions on disjoint blocks of w bits. Block products generalize several previously studied classes. The reduction applies to groups that are mixing in a representation-theoretic sense that we identify.
3. Combining (2) with (1) and other works we obtain new generators for block products over the quaternions or over any commutative group, with nearly optimal seed length. In particular, we obtain generators for read-once polynomials modulo any fixed m with nearly optimal seed length. Previously this was known only for $m = 2$.
4. We give a new generator for products over "mixing groups." The construction departs from previous work and uses representation theory. For constant error, we obtain optimal seed length, improving on previous work (which applied to any group).

This paper identifies a challenge in the area that is reminiscent of a roadblock in circuit complexity – handling composite moduli – and points to several classes of groups to be attacked next.

[*] Work done in part at Harvard University, supported by Madhu Sudan's and Salil Vadhan's Simons Investigator Awards.

[†] Supported by NSF grants CCF-2114116 and CCF-2430026.

ISSN 1433-8092

1 Introduction

The construction of explicit pseudorandom generators is a fundamental research goal that has applications in many areas of theoretical computer science. For background we refer to the recent survey [HH23]. We first define pseudorandom generators, incorporating the variants of *any order* (reflected in the permutation π) and *non-Boolean* tests (reflected in the range set R).

Definition 1 (Pseudorandom generators (PRGs))**.** An explicit function $P\colon \{0,1\}^s \to \{0,1\}^n$ is a *pseudorandom generator (PRG)* with seed length s and error ε for a class of functions F mapping $\{0,1\}^n$ to a set R if for every $f \in F$ the statistical distance between $f(P(U_s))$ and $f(U_n)$ is $\leq \varepsilon$, where U_s denotes the uniform distribution over $\{0,1\}^s$. We say P fools F *in any order* if $\pi(P)$ fools F for any permutation π of the positions of the n input bits. A PRG is *explicit* if it is computable in time n^c.

PRGs for branching programs, and group programs. A main agenda is obtaining explicit pseudorandom generators for read-once branching programs (ROBPs), with an ultimate goal of proving BPL = L. However, even for *constant-width, permutation* ROBPs, the best known seed length is $\geq \log n \log(1/\varepsilon)$. This is $\geq \log^2 n$ when $\varepsilon = 1/n$, and thus falls short of the optimal seed length $c \log(n/\varepsilon)$. For permutation ROBPs of width w, seed length $c_w \log(n/\varepsilon) \log(\varepsilon^{-1} \log n)$ follows from instantiating the "Polarizing Random Walks" [CHHL19] with a bound from [RSV13, LPV22]. These generators work in *any order*; thus they essentially match the seed length $c_w \log(n/\varepsilon) \log(1/\varepsilon)$ that was already available for *fixed-order* in a sequence of exciting works culminating in [Ste12].

The class of permutation ROBPs is equivalent to *group programs* (see e.g. [KNP11]):

Definition 2. A *program* (or *product*) p of length n over a group G is a tuple $(g_1, g_2, \ldots, g_n) \in G^n$. The program computes the function $f_p\colon \{0,1\}^n \ni x \mapsto \prod_{i\in[n]} g_i^{x_i} \in G$.

No generator with seed length less than $\log n \log(1/\varepsilon)$ was available for any non-commutative group. While optimal seed length $c \log(n/\varepsilon)$ was known for $\mathbb{Z}_2$ since [NN90], it took nearly 20 years and different techniques to have the same seed length over $\mathbb{Z}_3$ [LRTV09b, MZ09a], and remarkably that seed length is still not available even for $\mathbb{Z}_6$ (see [GKM18] for the best known construction).

PRGs for read-once polynomials. Another model that has received significant attention is *read-once polynomials*. Intuitively, this model can serve as a bridge between permutation and non-permutation ROBPs. The available generators for non-permutation ROBPs have significantly worse seed length than for permutation programs, see e.g. [MRT19a] and the discussion there.

A sequence of works [LV20a, MRT19a, DHH20] culminated in PRGs with seed length $c \log n + \log(1/\varepsilon) \log^c \log(1/\varepsilon)$ for read-once polynomials over $\mathbb{Z}_2$. But for other domains such as $\mathbb{Z}_3$ such good seed lengths were not known.

PRGs for block-products. A more general model that generalizes and unifies the previous ones is what we call block-products of width w over a group G. Here, the input bits are arbitrarily partitioned in blocks of w bits, arbitrary Boolean functions are then applied to each block, and finally the outputs are used as exponents to group elements. For our results, we will need to allow one block to be larger; we call this *spill* and incorporate it in the following definition.

Definition 3 (Block-product with spill)**.** A function $f\colon \{0,1\}^n \to G$ is computable by a w-block product with ℓ terms and a spill of q bits, written as (ℓ, w, q)*-product,* over a group G if there exist $\ell + 1$ disjoint subsets $I_0, I_1, \ldots, I_\ell \subseteq [n]$, where $|I_0| \leq q$ and $|I_i| \leq w$ for each $i \in [\ell]$ such that

$$f(x) = \prod_{i=1}^{\ell} g_i^{f_i(x_{I_i})}$$

for some group elements $g_i \in G$, functions $f_i\colon \{0,1\}^{I_i} \to \{0,1\}$. Here x_{I_i} are the $|I_i|$ bits of x indexed by I_i.

Note that block products are unordered by definition. They are a generalization of several function classes that have been studied, including *modular sums* [LRTV09a, MZ09b, GKM18] (when G is a cyclic group and $w = 1$), product tests with outputs in $\{-1,1\}$ (a.k.a. combinatorial checkerboards) [Wat13, HLV18, LV18, LV20b, Lee19] (when $G = \mathbb{Z}_2$), themselves a generalization of *combinatorial rectangles* [ASWZ96, Lu02, GMR$^+$12], and unordered *combinatorial shapes* [GMRZ13, GKM18] (when $G = \mathbb{Z}_{m+1}$). Block products also generalize read-once polynomials because one can show (for the uniform and typically also pseudorandom distributions) that monomials of degree $\geq \log(n/\varepsilon)$ do not affect the result significantly, and so one can simulate these polynomials with blocks of size $\log(n/\varepsilon)$.

In terms of generators, a series of works culminating in [Lee19] gives nearly-optimal seed length (i.e., $w + \log(\ell/\varepsilon)$ up to lower-order factors) over $\mathbb{Z}_2$. But such a result was not known over other groups such as $\mathbb{Z}_3$ or any non-commutative group.

1.1 Our results

In this work we bring new techniques, notably from group theory, to bear on these problems, and use them to obtain new pseudorandom generators.

First, we obtain optimal seed length for products over p-groups.

Definition 4. A finite p-group is a group of order p^k for an integer k and a prime p.

Equivalently, the order of every element is a power of p. (The latter definition makes sense for infinite groups, but we only consider finite groups.) The class of p-groups is rich and has been studied in various areas of theory of computation. For example, p-groups remain a candidate for good group-theoretic algorithms for matrix multiplication [BCC$^+$17]; the isomorphism testing for a subclass of p-groups has been identified as a barrier to faster group isomorphism algorithms [Sun23]; p-groups (specifically, unitriangular groups) are used for cryptography in NC^0 [AIK06] (see [Vio09a] for an exposition emphasizing these groups); finally, p-groups (specifically, quaternions) are used in computer graphics to express 3D rotations [Kui02].

We now give a few examples of such groups, all of which are non-commutative.

- The *quaternion group* $\mathbb{Q}_8$ of order 8 is a 2-group.
- *Unitriangular groups* over $\mathbb{F}_p$ are p-groups. They consist of upper-triangular matrices (of some fixed dimension), with 1 on the diagonal and entries in $\mathbb{F}_p$.
- *Wreath products* give natural examples of p-groups. For example, the wreath product $\mathbb{Z}_p \wr \mathbb{Z}_p$ is a group of order p^{p+1}, hence a p-group. This group is the direct product $\mathbb{Z}_p^p$ with another element in $\mathbb{Z}_p$ acting on the tuple by shifting the coordinates. For concreteness, the case $p = 2$ can be presented as $(a, b; z)$ where $a, b, z \in \mathbb{Z}_2$, $(a, b; 0)(a', b'; z') = (a + a', b + b'; z')$, and $(a, b; 1)(a', b'; z') = (a + b', b + a'; 1 + z')$. Wreath product constructions (not necessarily p-groups) have been studied in a variety of contexts ranging from group-theoretic algorithms for matrix multiplication [CKSU05], to construction of expander graphs [ALW01, RSW06], to mixing in non-quasirandom groups [GV22].
- The *dihedral group* $\mathbb{D}_n$ is the group of order $2n$ of symmetries of a regular polygon with n sides. When $n = 2^t$, $\mathbb{D}_n$ is a 2-group.

We give pseudorandom generators for programs over p-groups, with optimal seed length. Throughout this paper, we use c_x to denote a constant that depends on the variable x.

Theorem 5. *Let G be a p-group. There is an explicit pseudorandom generator that fools programs of length n over G in any order, with seed length $c_G \log(n/\varepsilon)$.*

In fact, the same result holds even for block-products over p-groups with constant block length w.

Polynomials and block-products. We give a general reduction that "lifts" a PRG P for group products over G to a PRG P' for block-products (and read-once polynomials) over G. The reduction applies to any group G that is *mixing*:

Definition 6 (Mixing groups)**.** A group G is *mixing* if it has a complete set of unitary irreducible representations where every non-identity matrix does not have 1 as an eigenvalue.

Remark 7. Our results for mixing groups (Theorems 10 and 12) apply more generally to fooling *words* over a mixing subset H of a (not necessarily mixing) group G. The property we need is that Definition 6 holds for every element in H. There are many examples of mixing subsets of non-mixing groups which generate the entire group G. For example, for $\mathbb{S}_3 = \mathbb{D}_3$, it suffices to exclude the "flip" element, i.e. the non-identity element r, where $r^2 = 1$. Moreover, one can have natural examples for infinite groups. However for simplicity we focus on finite mixing groups.

We note that mixing groups are exactly the class of *Dedekind groups*.

Definition 8. (Finite) *Dedekind groups* are groups of the form $\mathbb{Q}_8 \times \mathbb{Z}_2^t \times D$ for any integer t and commutative group D of odd order. A non-commutative Dedekind group is also called a Hamiltonian group.

Lemma 9 (Mixing characterization of Dedekind groups)**.** *A finite group is mixing if and only if it is Dedekind.*

A proof of Lemma 9 is in Section 9.

We can now state our reduction:

Theorem 10. *Let G be a mixing group. Suppose there is a PRG P_1 with seed length s_1 that ε-fools $(\ell, 1, 3\log(1/\varepsilon))$-products over G. Then there is a PRG that $\big(c_G \log(w+\log(\ell/\varepsilon))\cdot\varepsilon\big)$-fools $(\ell, w, \log(1/\varepsilon))$-products over G, with seed length*

$$c_G\big(s_1 + \log(\ell/\varepsilon) + w\big) \cdot \log^c\big(w + \log(n\ell/\varepsilon)\big).$$

Note that if P_1 has nearly optimal seed length (i.e., $\log(\ell/\varepsilon)$ times lower-order terms) then also the final PRG has nearly optimal seed length (i.e., $w + \log(\ell/\varepsilon)$, times lower-order terms).

Applying the reduction (Theorem 10) we obtain near-optimal PRGs for block products over commutative or Dedekind 2-groups (in particular, the quaternions).

Corollary 11. *Let G be either a commutative group, or a Dedekind 2-group, that is, $G = \mathbb{Q}_8 \times \mathbb{Z}_2^t$ for some t. There is an explicit PRG that ε-fools $(\ell, w, 0)$-block products over G with seed length $c_G(w + \log(\ell/\varepsilon))\log^c(w + \log(\ell n/\varepsilon))$.*

Proof. We use the reduction (Theorem 10). For commutative groups we use the PRG in [GKM15] for P_1; for Dedekind 2-groups we use our Theorem 5 for P_1. Actually, in both cases the generators were only stated for group products while we need to handle the spill. The simple modification is in Section 8. □

As remarked earlier, as a consequence of Corollary 11, we obtain PRGs for read-once polynomials over n variables over any finite field $\mathbb{F}$ with near-optimal seed length $c_{\mathbb{F}} \log(n/\varepsilon)\log^c \log(n/\varepsilon)$. Again, this was not known even for $\mathbb{F}_3$.

This result is also a step towards handling group programs over more general groups, for example *nilpotent groups,* which are direct products of p-groups (for different p). Jumping ahead, our techniques imply that generators for such groups follow from generators for (non-read-once) polynomials over composites.

Finally, we give a new generator for products over mixing groups.

Theorem 12. *Let G be a mixing group. There is an explicit PRG P that ε-fools length-n programs over G with seed length $c_G \log(n/\varepsilon)\log(1/\varepsilon)$, in any order.*

The parameter improvement over previous work appears tiny: As remarked earlier, [CHHL19] gives seed length $c_G \log(n/\varepsilon)\log(\varepsilon^{-1}\log n)$, and moreover for any G. Still, for constant error we obtain optimal seed length which was known only in the fixed-order case (cf. [Ste12]). Also note that mixing groups of the form $\mathbb{Q}_8 \times \mathbb{Z}_2^t$ (i.e., $m = 1$) are 2-groups, for which we gave optimal seed length in Theorem 5. But the techniques there do not even apply to the commutative (mixing) group $\mathbb{Z}_2 \times \mathbb{Z}_3$.

Our main interest in this result is that its proof is different from previous work: it showcases how we can use information on the representation theory to improve the parameters, pointing to several open problems.

1.2 Future directions and open problems

This work suggests that the difficulty of handling more general classes of groups lies in composite moduli. For example, we do not have new generators for $\mathbb{D}_3 = \mathbb{S}_3$, a group of order 6, even though we have optimal seed length for $\mathbb{D}_n$ when n is a power of two. Thus, a challenge emerging from this work is to improve the seed length over any non-commutative group of composite order. Again, $\mathbb{S}_3$ is an obvious candidate, which is equivalent to fooling width-3 permutation ROBPs. But other groups could be easier to handle, for example Dedekind groups or the direct product of a p-group and a q-group where $p \neq q$ are primes.

Also, the techniques in this paper point to several other questions. Can we extend our reduction to block products where instead of g^f for Boolean f we more generally have g^f replaced by a function with range G? For what other groups can we exploit representation theory to obtain better PRGs?

2 Proof of Theorem 5

We use the fact that programs over p-groups can be written as polynomials. Elements in a group of order p^k will be written as k-tuples over $\mathbb{F}_p$.

Lemma 13. *Let G be a group of order p^k, and k an integer. There is a 1-1 correspondence between G and $\mathbb{F}_p^k$ and a polynomial map $f = (f_1, \ldots, f_k)\colon (\mathbb{F}_p^k)^n \to \mathbb{F}_p^k$ over $\mathbb{F}_p$ where the $f_i\colon (\mathbb{F}_p^k)^n \to \mathbb{F}_p$ have degree c_G such that for any $\overline{g} := (g_1, g_1, \ldots, g_n) \in G^n$ and $x \in \{0,1\}^n$, we have*

$$\prod_{i=1}^{n} g_i^{x_i} = \big(f_1(\overline{g}), f_2(\overline{g}), \ldots, f_k(\overline{g})\big)(x_1, \ldots, x_n).$$

This lemma is essentially in the previous work [PT01]. However the statement there is for nilpotent groups and cannot be immediately used. Also, the proof relies on previous work and is somewhat indirect. So we give a direct proof of the result we need (i.e., Lemma 13).

Before the proof we illustrate it via an example.

Example 14. Let $G := \mathbb{Z}_2 \wr \mathbb{Z}_2$ from the introduction. Consider a product $\prod_i (a_i, b_i; z_i)$. Via a polynomial map we can rewrite this product into a normal form where all the z_i are in one element only:

$$\Big(\prod_i (a_i', b_i'; 0)\Big)(0, 0; z').$$

Computing this product is then immediate, via a linear map. The key observation is that $a_i' = a_i$ if the sum of the z_j with $j < i$ is even, and $a_i' = b_i$ otherwise, and that this computation is a quadratic polynomial (in the input bits a_i, b_i, z_i).

Proof of Lemma 13. We proceed by induction on k. If $k = 1$, then G is cyclic. We can take a generator $a \in G$ and define the 1-1 mapping $G \ni a^z \leftrightarrow z \in \mathbb{F}_p$. So $\prod_{i=1}^n g_i^{x_i} = \prod_{i=1}^n a^{z_i x_i}$ can be written as the degree-1 polynomial $\sum_{i=1}^n z_i x_i$.

Otherwise, G has a normal subgroup H of order p^{k-1} [DF04, Chapter 6, Theorem 1.(3)]. The corresponding quotient group $Q = G/H$ has order p and is therefore cyclic. So we can write $g_i \in G$ as

$$g_i = a^{e_i} h_i$$

where $h_i \in H$ and a is a generator of Q. Applying the induction hypothesis on H, we can identify each element $g_i = a^{e_i} h_i$ with a k-tuple $(e_i, e_i') \in \mathbb{F}_p^k$, where $e_i' \in \mathbb{F}_p^{k-1}$ corresponds to $h_i \in H$.

Now we apply the conjugation trick as in [Bar89], and use induction. That is, let $b_i := a^{\sum_{j \le i} e_j x_j}$ and write

$$\prod_{i=1}^{n} g_i^{x_i} = \Big(\prod_{i=1}^{n} \big(b_i h_i^{x_i} b_i^{-1}\big)\Big) b_n.$$

Note that b_i and b_i^{-1} can be computed by some degree-1 polynomials over $\mathbb{F}_p$, and $h_i^{x_i}$ can be (trivially) computed by a degree-c_H polynomial over $\mathbb{F}_p$.

Therefore, each term $b_i h_i^{x_i} b_i^{-1}$ can be computed by some degree-c_G polynomial map $f^{H,i} = (f_1, \ldots, f_k)$ over $\mathbb{F}_p$. Moreover, these terms lie in H because H is a normal subgroup of G. Hence we have reduced to a product over H, which by induction hypothesis, can be computed by some degree-c_H polynomial, and the result follows. □

Given Lemma 13, it suffices to construct a *bit*-generator that fools low-degree read-once polynomials over $\mathbb{F}_p$.

The case $p = 2$. For this case, we can simply combine Lemma 13 with known generators for polynomials over $\mathbb{F}_2$ [BV10, Lov09, Vio09b]. In fact, we obtain results for non-read-once programs as well, and of any length. (Indeed, such polynomials are equivalent to low-degree polynomials over $\mathbb{F}_2$.)

The case $p > 2$. Here we need additional ideas because *bit*-generators that output bits and fool polynomials over $\mathbb{F}_q$ with $q \neq 2$ are not known. However, the works [BV10, Lov09, Vio09b] do give generators that output *field elements* that fool such polynomials.

Lemma 15 ([Vio09b]). *There are distributions Y over $\mathbb{F}_p^n$ that can be explicitly sampled from a uniform seed of $c_p(2^d \log(1/\varepsilon) + \log n)$ bits such that for any degree-d polynomial f in n variables over $\mathbb{F}_p$, we have $\Delta(f(Y), f(U)) \le \varepsilon$.*

However, we need distributions over $\{0,1\}^n$. This distinction is critical and arises in a number of previous works. Currently, for domain $\{0,1\}^n$ only weaker results with seed length $\ge \log^2 n$ are known [LMS10].

Still, as pointed out in [LRTV09b, MZ09a], Lemma 15 implies results over the domain $\mathbb{F}_2^n$ for *biased bits:*

Definition 16. We denote by N_p a vector of n i.i.d. bits coming up 1 with probability $1/p$.

Corollary 17 ([LRTV09b, MZ09a]). *There are distributions X over $\{0,1\}^n$ that can be explicitly sampled from a uniform seed of $c_p(2^d \log(1/\varepsilon) + \log n)$ bits such that for any degree-d polynomial f in n variables over $\mathbb{F}_p$ we have $\Delta(f(X), f(N_p)) \le \varepsilon$.*

Proof. Let $Y = (Y_1, Y_2, \ldots, Y_n)$ be the distribution from Lemma 15, for degree $d(p-1)$. Define $X := (Y_1^{p-1}, Y_2^{p-1}, \ldots, Y_n^{p-1})$. Note X is over $\{0,1\}^n$. Also, if U is uniform in $\mathbb{F}_p$ then $U^{p-1} = N_p$. The result follows. □

We will show how to use biased bits. For this we use that the program is read-once.

Lemma 18. *Let X fool degree-1 polynomials over $\mathbb{F}_2$ with error $\varepsilon^{c_{G,p}}$. Then $X + N_p$ fools programs of length n over G with error ε.*

Proof. This follows from Lemma 7.2 in [FK18] combined with the Fourier bound in [RSV13, LPV22]. The proof in [FK18] is for the fixed noise parameter $p = 4$, but the generalization to any p is immediate (replace $1/2$ with $1 - 2/p$ in the last two lines of the proof). □

We now have all the ingredients.

Proof of Theorem 5. Use Lemma 18. Averaging over X, it suffices to derandomize N_p. By Lemma 13 it suffices to do this for low-degree polynomials. This follows from Corollary 17. □

3 Representation theory and matrix analysis

In this section, we present the fragment of representation theory and matrix analysis that we need. The books by Serre [Ser77], Diaconis [Dia88], and Terras [Ter99] are good references for representation theory and non-commutative Fourier analysis. The Barbados notes [Wig10], [Gow17, Section 13], [GV22], or [DLV24] provide briefer introductions.

Matrices. Let M be a square complex matrix. We denote by $\mathrm{tr}(M)$ the trace of M, by $\overline{M}$ the conjugate of M, by M^T the transpose of M, and by M^* the conjugate transpose $\overline{M}^T$ (aka adjoint, Hermitian conjugate, etc.). The matrix M is *unitary* if the rows and the columns are orthonormal; equivalently $M^{-1} = M^*$.

The *Frobenius norm*, (a.k.a. Schatten 2-norm, Hilbert–Schmidt operator) of a square matrix M, denoted $\|M\|_{\mathsf{F}}$, is $\sum_{i,j} |M_{i,j}|^2 = \mathrm{tr}(MM^*)$.

The *operator norm* of a matrix M, denoted $\|M\|_{\mathsf{op}}$, is the square root of the largest eigenvalue of the matrix MM^*. In particular, if M is a normal matrix, i.e. $MM^* = M^*M$, then $\|M\|_{\mathsf{op}}$ equals its largest eigenvalue in magnitude.

Fact 19. $\|AB\|_{\mathsf{op}} \leq \|A\|_{\mathsf{op}} \|B\|_{\mathsf{op}}$.

Fact 20. *For a $d \times d$ matrix M with eigenvalues $\lambda_1, \ldots, \lambda_d$, we have $\|M\|_{\mathsf{F}}^2 = \sum_{i=1}^{d} |\lambda_i|^2 \leq d \|M\|_{\mathsf{op}}^2$.*

Representation theory. Let G be a group. A *representation* ρ of G with dimension d maps elements of G to $d \times d$ unitary, complex matrices so that $\rho(xy) = \rho(x)\rho(y)$. Thus, ρ is a homomorphism from G to the group of linear transformations of the vector space $\mathbb{C}^d$. We denote by d_ρ the dimension of ρ.

If there is a non-trivial subspace W of $\mathbb{C}^d$ that is invariant under ρ, that is, $\rho(x)W \subseteq W$ for every $x \in G$, then ρ is *reducible*; otherwise it is *irreducible.* Irreducible representations are abbreviated *irreps* and play a critical role in Fourier analysis. We denote by $\widehat{G}$ a complete set of inequivalent irreducible representations of G.

Let $\widehat{G}$ be the set of irreducible representations of G (i.e. the dual group of G). We have

$$\sum_{\rho\in\widehat{G}} d_\rho^2 = |G|. \tag{1}$$

For a random variable Z we also use Z to denote its probability mass function. For an irrep $\rho \in \widehat{G}$, the ρ-th Fourier coefficient of Z is

$$\widehat{Z}(\rho) := \sum_{g\in G} Z(g)\overline{\rho(g)} = \mathbf{E}\Big[\overline{\rho(Z)}\Big].$$

The Fourier expansion of $Z\colon G \to \mathbb{R}$ is

$$Z(g) = \frac{1}{|G|}\sum_{\rho\in\widehat{G}} d_\rho \operatorname{tr}\big(\widehat{Z}(\rho)\rho(g)\big).$$

Parseval's identity gives

$$\sum_{g\in G} Z(g)^2 = \frac{1}{|G|}\sum_{\rho\in\widehat{G}} d_\rho \|\widehat{Z}(\rho)\|_{\mathsf{F}}^2.$$

Claim 21. *Suppose X and Y are two random variables over G such that for every irreducible representation ρ of G, we have $\|\mathbf{E}[\rho(X)] - \mathbf{E}[\rho(Y)]\|_{\mathsf{op}} \le \varepsilon$. Then X and Y are $(\sqrt{|G|}\cdot\varepsilon)$-close in statistical distance.*

Proof.

$$\begin{aligned}
\frac{1}{2}\sum_{g\in G}\big|X(g) - Y(g)\big| &\le \frac{\sqrt{|G|}}{2}\Big(\sum_{g\in G}\big(X(g) - Y(g)\big)^2\Big)^{1/2} && \text{(Cauchy–Schwarz)}\\
&= \frac{\sqrt{|G|}}{2}\Big(\frac{1}{|G|}\sum_{\rho\in\widehat{G}} d_\rho \big\|\widehat{X}(\rho) - \widehat{Y}(\rho)\big\|_{\mathsf{F}}^2\Big)^{1/2} && \text{(Parseval)}\\
&= \frac{1}{2}\Big(\sum_{\rho\in\widehat{G}} d_\rho\cdot(d_\rho\varepsilon^2)\Big)^{1/2} && \text{(Fact 20)}\\
&= \frac{\varepsilon}{2}\cdot\Big(\sum_{\rho\in\widehat{G}} d_\rho^2\Big)^{1/2} = \sqrt{|G|}\cdot\varepsilon/2. && \text{(Equation (1))}
\end{aligned}$$

□

4 Proof of Theorem 12

Again, besides the parameter improvement, our main point here is to illustrate how we use representation theory to obtain pseudorandom generators. These ideas will then be generalized to the more general and complicated setting of block products in the next section.

Let ρ be an irreducible representation of a mixing group (Definition 6). By definition of mixing, if ρ is a non-identity matrix then it does not have 1 as its eigenvalues. A main observation is that if there are many non-identity matrices $\rho(g_i)$ in the program, then the bias $\|\mathbf{E}[\prod_{i=1}^n \rho(g_i)^{U_i}]\|_{\mathsf{op}}$ is small. This is proved in the next two claims.

Claim 22. *Let M be a unitary matrix with eigenvalues $e^{i\theta_j}$ for some $\theta_j \in [-\pi, \pi]$ on the unit circle. Suppose $|\theta_j| \geq \theta$ for every j. Then $\|(I+M)/2\|_{\mathsf{op}} \leq 1 - \theta^2/8$.*

Proof. As M is unitary, we can write $M = Q^* D Q$, where D is a diagonal matrix with M's eigenvalues on its diagonal and Q is unitary. The eigenvalues of $(I+M)/2 = Q^*(\frac{I+D}{2})Q$ are $\frac{1+e^{i\theta_j}}{2} = e^{i\theta_j/2} \cdot \frac{e^{-i\theta_j/2}+e^{i\theta_j/2}}{2} = e^{i\theta_j/2}\cos(\theta_j/2)$, which have magnitudes at most $|\cos(\theta_j/2)| \leq 1 - \theta^2/8$. □

Claim 23. *Let G be a mixing group. Let ρ be an irreducible representation of G of dimension d_ρ. Let $f_\rho(x) = \prod_{i=1}^n \rho(g_i)^{x_i}$ be the representation of a group program. Suppose $\rho(g_i) \neq I_{d_\rho}$ for $t \geq c_G \log(1/\varepsilon)$ many i's. Then $\|\mathbf{E}[f_\rho(U)]\|_{\mathsf{op}} \leq \varepsilon$.*

Proof. Let T be the t coordinates j where $\rho(g_j) \neq I_{d_\rho}$. For every fixing of the other coordinates, we can write $f_\rho(U)$ as

$$B \prod_{j \in T} \rho(g_j)^{U_j} B_j$$

for some unitary matrices B and B_j's. So

$$\|f_\rho(U)\|_{\mathsf{op}} \leq \|B\|_{\mathsf{op}} \prod_{j \in T} \|\mathbf{E}[\rho(g_j)^{U_j}]\|_{\mathsf{op}} \|B_j\|_{\mathsf{op}} \leq (1 - c_G)^t \leq \varepsilon.$$ □

We now proceed with the proof of the main result. The proof extends to handle the spill, but for simplicity we do not discuss it here. We fool each irreducible representation of G separately and then appeal to Claim 21. Fix a representation ρ and consider the product

$$f_\rho(x) := \prod_{j=1}^{\ell} \rho(g_j)^{x_j}.$$

Let t be the number of non-identity elements $\rho(g_j) : j \in [n]$ and S be their coordinates.

Let us sketch the construction. First, XORing with an almost $2c_G \log(1/\varepsilon)$-wise uniform distribution takes care of the case $t \leq c_G \log(1/\varepsilon)$, so we may assume that t is larger. In this case, by Claim 23, we have that the bias $\|\mathbf{E}[f_\rho(U)]\|_{\mathsf{op}}$ is small under the uniform distribution. Our goal is to set ct bits in S to uniform and apply Claim 23 again.

Let $\ell := c_G \log(1/\varepsilon)$. Let M be a $(\log n) \times 10\ell$ matrix filled with uniform bits.

We will make $\log n$ guesses of t. For each guess $v = 2^i \cdot \ell : i \in \{0, \ldots, \log n - 1\}$ of t, we select a subset of size ℓ of the input positions using a hash function h_i, and then hash these ℓ positions to row i of M using another hash function h, and assign input bits correspondingly. The final generator is obtained by trying all guesses, using the same seed for each guess h_i, and XORing together the bits.

In more detail, for each $i \in \{0, \ldots, \log n - 1\}$, let $h_i \colon [n] \to \{0,1\}$ be a 10ℓ-wise independent hash family with $\mathbf{Pr}_{h_i}[h_i(j) = 1] = 2^{-i}$ for each $j \in [n]$. Let $h \colon [n] \to [10\ell]$ be another 5ℓ-wise uniform hash family. The output of our generator is

$$D := D^{(0)} \oplus \cdots \oplus D^{(\log n - 1)},$$

where the j-th bit of $D^{(i)}$ is

$$h_i(j) \cdot M_{i,h(j)}.$$

We use the same seed to sample $h_0, \ldots, h_{\log n-1}$, which costs at most $O_G(\log n \log(1/\varepsilon))$ bits [Vad12, Corollary 3.34]. Sampling h uses another $O_G(\log(n/\varepsilon)\log(1/\varepsilon))$ bits. This uses a total of $O_G(\log(n/\varepsilon)\log(1/\varepsilon))$ bits.

We now show that $\|\mathbf{E}[f_\rho(D)]\|_{\mathsf{op}} \le O(\varepsilon)$. Suppose $t \in [2^i\ell, 2^{i+1}\ell]$. Recall that S are the coordinates corresponding to the non-identity matrices in the product. Let $J := h_i^{-1}\{1\} \cap S$. As $\mathbf{Pr}[h_i(j) = 1] = 2^{-i}$, we have $\mathbf{E}[|J|] \in [\ell, 2\ell]$. Applying tail bounds for bounded independence (see Lemma 36), we have $|J| \in [\ell/2, 3\ell]$ except with probability ε. Conditioned on this event, as $|J| \le 3\ell$ and h is 5ℓ-wise uniform, we can think of h as a random function from J to $[10\ell]$. Hence, for each $j \in [10\ell]$, we have

$$\mathbf{Pr}\big[|J \cap h^{-1}(j)| = 1\big] = |J| \cdot 1/(10\ell) \cdot (1 - 1/(10\ell))^{|J|-1} \ge (\ell/2) \cdot (1/10\ell) \cdot (1/2) \ge 1/40.$$

By a Chernoff bound, we have that except with probability at most ε, the number of j such that $|J \cap h^{-1}(j)| = 1$ is at least $\ell/10$.

Let T be these coordinates. Fixing all the bits in M except the ones in row i that are fed into T, we can write the conditional expectation of $f_\rho(G)$ over the bits in T as

$$B \prod_{j \in T} \mathop{\mathbf{E}}_{x_j}[A_j^{x_j}] B_j,$$

for some unitary matrices B, A_j's and B_j's, and in particular, A_j has its eigenvalues bounded away from 1 on the complex unit circle. Therefore, by Claim 22,

$$\Big\| B \prod_{j \in T} \mathop{\mathbf{E}}_{x_j}[A_j^{x_j}] B_j \Big\|_{\mathsf{op}} \le \prod_{j \in T} \Big\| \frac{(I + A_j)}{2} \Big\|_{\mathsf{op}} \le \varepsilon.$$

5 Proof of Theorem 10

In this section we prove Theorem 10. This type of reductions goes back to the work of [GMR$^+$12] on read-once CNFs (itself building on [AW89]), and have been refined in several subsequent works. The work [LV20a] extended the techniques to read-once polynomials. It exploited the observation that when the number of monomials is significantly larger than its degree, the bias of the polynomial is small, and therefore the bias of the restricted function remains small. Building on this observation, [MRT19b] showed that one can aggressively restrict most of the coordinates, while keeping the bias of the restricted function small. In addition, a typical restricted product is a low-degree polynomial (plus a spill), for which we have optimal generators [BV10, Lov09, Vio09b].

However, [MRT19b] reduces to non-linear polynomials (degree 16). As discussed earlier, bit-generators with good seed lengths are only known over $\mathbb{Z}_2$. We give a refined reduction that reduces to polynomials of degree one, for which we have generators over $\mathbb{Z}_m$ for any m [LRTV09b, MZ09a, GKM18].

At the same time, we show that the reduction can be carried over any mixing group, by working with representations of the group.

Definition 24. Let $\mathcal{U}_\theta(d)$ be the set of $d \times d$ unitary matrices with eigenvalues $e^{2\pi i\theta_j}$ where $|\theta_j| \ge \theta$.

Definition 25. A group G is *θ-mixing* if it has a complete set of unitary irreducible representations where each non-identity matrix lies in $\mathcal{U}_\theta(d)$ for some d.

The following theorem will serve as the basis of our iterative construction of the PRG.

Theorem 26. *Let $w \geq \log\log(1/\varepsilon) + \log m$. Suppose there is a PRG P with seed length s that ε-fools $(m^5 2^{30w}, 2w, 2\log(1/\varepsilon))$-products over G. Let P_1 be a PRG with seed length s_1 that ε-fools $(\ell, 1, 3\log(1/\varepsilon))$-products over a group G of order m that is $(1/m)$-mixing. Then there is a PRG that $O(\varepsilon)$-fools $(m^5 2^{45w}, 3w, 2\log(1/\varepsilon))$-products over G with seed length*

$$s + \Big(s_1 + O_m((\log(1/\varepsilon) + w)\log w + \log\log n)\Big).$$

We first show how to apply Theorem 26 iteratively to obtain Theorem 10.

Proof of Theorem 10. We iterate Theorem 26 repeatedly for some t times to reduce the problem to fooling an $O(\log(m/\varepsilon))$-junta which can be done using an almost bounded uniform distribution.

Given an $(\ell, w, \log(1/\varepsilon))$-product f, let $w' = \max\{w, \log\ell, \log m\}$ so that we can view f as an $(m^5 2^{45w'}, 3w', 2\log(1/\varepsilon))$-product. We first apply Theorem 26 for $t_1 = O(\log w')$ times until we have a

$$\big((m \cdot \log(1/\varepsilon))^C, \log\log(1/\varepsilon) + \log m, 2\log(1/\varepsilon)\big)\text{-product},$$

for some constant C.

Let $b := \frac{\log(1/\varepsilon)+\log m}{\log\log(1/\varepsilon)+\log m}$. We will apply the following repeatedly for some $r = O_m(1)$ steps. We divide the $f_i : i \geq 1$ into groups of b functions and view the product of functions in each group as a single function, this way we can think of the above product as a

$$\Big(\tfrac{(m\cdot\log(1/\varepsilon))^C}{b}, \log(1/\varepsilon) + \log m, 2\log(1/\varepsilon)\Big)\text{-product.}$$

So we can continue applying Theorem 26 for $t_2 = O(\log(\log(1/\varepsilon) + \log m)) \leq O_m(\log\log(1/\varepsilon))$ times and the restricted function becomes a

$$\Big(\tfrac{(m\cdot\log(1/\varepsilon))^C}{b}, \log\log(1/\varepsilon) + \log m, 2\log(1/\varepsilon)\Big)\text{-product.}$$

Repeating this process for

$$r = \log_b\Big(\big(m \cdot \log(1/\varepsilon)\big)^C\Big) \leq \frac{2C\big(\log m + \log\log(1/\varepsilon)\big)}{\log\log(1/\varepsilon)} = O_m(1)$$

times, we are left with a

$$\big(O(1), \log\log(1/\varepsilon) + \log m, 2\log(1/\varepsilon)\big)\text{-product}$$

which can be fooled by an ε-almost $O(\log(m/\varepsilon))$-wise uniform distribution that can be sampled using $s' = O(\log(m/\varepsilon) + \log\log n)$ bits [NN93, AGHP92]. Therefore, in total we apply Theorem 26 for

$$t := t_1 + r \cdot t_2 \leq O(\log w') + O_m(\log\log(1/\varepsilon)) = O_m\big(\log(w + \log(\ell/\varepsilon))\big)$$

times, each with a seed of

$$s = s_1 + O_m((\log(\ell/\varepsilon) + w)\log w + \log\log n).$$

bits. Hence in total it uses

$$s \cdot t + s' \le O_m\big(s_1 + \log(\ell/\varepsilon) + w\big) \cdot \operatorname{polylog}\big(w, \log \ell, \log n, \log(1/\varepsilon)\big)$$

bits. □

5.1 Analysis of one iteration: Proof of Theorem 26

We now prove Theorem 26. Given an $(m^5 \cdot 2^{45w}, 3w, 2\log(1/\varepsilon))$-product $f = \prod_{i=0}^{\ell} f_i$ over G of order m that is $(1/m)$-mixing, let ℓ be the number of non-constant f_i. We say f is a *long product* if $\ell \ge m^5 \cdot 2^{30w}$, otherwise f is *short*. At a high-level, we apply Theorem 30 to P_1 to obtain a PRG that fools long products in *one shot*, and use Lemma 27 and 29 below to reduce fooling a short product to fooling a product of smaller width w.

Lemma 27. *Let $w \ge \log m$ and C be a sufficiently large constant. Define*

$$\begin{aligned} k &:= C(w + \log(\ell/\varepsilon)) \\ \delta &:= (m \cdot w)^{-k} \\ p &:= 2^{-C}. \end{aligned}$$

There exist two δ-almost k-wise independent distributions D and T with $\mathbf{E}[D_i] = 1/2$ and $\mathbf{E}[T_i] = p$ for every $i \in [n]$, such that for every $(\ell, w, 0)$-product f over G of order m, we have $|\mathbf{E}_{D,T}[f(D + T \wedge U)] - \mathbf{E}[f(U)]| \le \varepsilon$.

Moreover, D and T can be efficiently sampled with a seed of length $O_m((\log(\ell/\varepsilon) + w)\log w + \log\log n)$.

Lemma 27 follows from the following lemma, which can be obtained from applying a variant of a result of Forbes and Kelley [FK18] to the Fourier bounds on functions computable by block products over groups, which was established in [LPV22]. (Block products are called generalized group products in [LPV22].)

Lemma 28 ([FK18, LPV22])**.** *Let $f\colon \{0,1\}^n \to \{0,1\}$ be computable by an $(\ell, w, 0)$-block product over a group G. Let D and T be two independent δ-almost $2(k+w)$-wise independent distributions on $\{0,1\}^n$ with $\mathbf{E}[D_i] = 1/2$ and $\mathbf{E}[T_i] = p$, and U be the uniform distribution on $\{0,1\}^n$. Then*

$$\big|\mathbf{E}[f(D + T \wedge U)] - \mathbf{E}[f(U)]\big| \le \ell \cdot \big(\sqrt{\delta} \cdot (w \cdot |G|)^{k+w} + (1 - 2p)^{k/2} + \sqrt{\gamma}\big).$$

We remark that for every constant p, one can show that $n^{-\omega(1)}$-bias plus noise N_p is necessary to fool programs over groups of order $\operatorname{poly}(n)$ with any subconstant error ε. This follows from [DILV24], where it shows that there exists such a distribution which puts 2ε more probability mass on strings whose Hamming weight is greater than $n/2 + O_p(\sqrt{kn})$ than the uniform distribution.

Lemma 29 (Width reduction for short products)**.** *Let G be any group of order m. Let D and T be the two distributions defined in Lemma 27. Let $w \geq \log m$. Let f be an $(\ell, 3w, 2\log(1/\varepsilon))$-product over G, where $\ell \leq m^5 \cdot 2^{30w}$. Then with probability at least $1-\varepsilon$ over D and T, the restricted function $f_{D,T}$ is an $(\ell, 2w, 2\log(1/\varepsilon))$-product over G.*

We prove Lemma 29 in Section 7.

Theorem 30. *Let $w \geq \log\log(1/\varepsilon) + 2\log(1/\theta)$. Suppose there is a PRG P_1 with seed length s_1 that ε-fools $(\ell, 1, 3\log(1/\varepsilon))$-product $f := \prod_{i=0}^{\ell} f_i$ over a matrix group $\mathcal{M}$ supported on $\mathcal{U}_\theta(d)$, where $\ell \geq 2^{3w}\theta^{-2}$ and each f_i is non-constant. Then there is a PRG that fools $(\ell, 3w, 2\log(1/\varepsilon))$-products $f = \prod_{i=0}^{\ell} f_i$ over $\mathcal{M}$, where $\ell \in [2^{30w}\theta^{-5}, 2^{45w}\theta^{-5}]$ and every f_i is non-constant, with seed length $s = s_1 + O_\theta\big(\log(1/\varepsilon) + w + \log\log n\big)$.*

Corollary 31. *Theorem 30 applies to products over any θ-mixing group G with ε replaced by $\varepsilon/\sqrt{|G|}$.*

Proof. By definition, all its irreps ρ belong to $\mathcal{U}_\theta(d_\rho)$. It follows from Claim 21 that it suffices to fool all its irreps. □

We prove Theorem 30 in Section 6. We now show how Theorem 26 follows from Lemma 27 and 29 and Theorem 30.

Proof of Theorem 26. Let P_1 be the PRG that ε-fools $(\ell, 1, 3\log(1/\varepsilon))$-products with seed length s_1. Applying Theorem 30 with P_1, we obtain a PRG P_{long} that ε-fools every $(\ell, 3w, 2\log(1/\varepsilon))$-product $f = \prod_{i=0}^{\ell} f_i$, where $\ell \in [m^5 2^{30w}, m^5 2^{45w}]$ and every f_i is non-constant, with seed length $s_{\mathsf{long}} = s_1 + O_m(\log(1/\varepsilon) + w + \log\log n)$. We now sample the distributions D, T in Lemma 27, and output

$$(D + T \wedge P(U)) + P_{\mathsf{long}}.$$

Using Lemma 27 and $\ell \leq 2^{O_m(w)}$, sampling D and T uses $s_{\mathsf{short}} = O_m((\log(1/\varepsilon) + w)\log w + \log\log n)$ bits. So altogether this takes $s + s_{\mathsf{long}} + s_{\mathsf{short}} = s + s_1 + O_m((\log(1/\varepsilon) + w)\log w + \log\log n)$ bits.

Let f be an $(m^5 2^{45w}, 3w, 2\log(1/\varepsilon))$-product with ℓ many non-constant f_i's. If $\ell \geq m^5 2^{30w}$, then P_{long} ε-fools it. Otherwise, $\ell \leq m^5 2^{30w}$ and so f is an $(m^5 2^{30w}, 3w, 2\log(1/\varepsilon))$-product. So by Lemma 29, with probability at least $1-\varepsilon$ over the choices of D and T, the function $f_{D,T}$ is an $(m^5 2^{30w}, 2w, 2\log(1/\varepsilon))$-product, and therefore can be ε-fooled using the generator P given by the assumption. The total error is $O(\varepsilon)$. □

6 Width reduction for long products: Proof of Theorem 30

In this section, we prove Theorem 30. Let $f = \prod_{i=0}^{\ell} f_i$ be an $(\ell, 3w, 2\log(1/\varepsilon))$-product over a matrix group $\mathcal{M} \subseteq \mathcal{U}_\theta(d)$, where $\ell \in [2^{30w}\theta^{-5}, 2^{45w}\theta^{-5}]$ and each f_i is non-constant. Note that when a product f has this many non-constant functions, the "bias" $\|\mathbf{E}[f(U)]\|_{\mathsf{op}}$ of f is *doubly exponentially small* in w, i.e. at most $\exp(-2^{2w})$ (see Claim 33, which is at most ε

whenever $w \geq \log\log(1/\varepsilon)$). Following [MRT19b], we will pseudorandomly restrict most of the coordinates of f and show that the bias of a typical restricted product remains bounded by ε. More importantly, we will show that this restricted product has *width 1* (with a small spill). Therefore, it suffices to construct a PRG for width-1 products (with a small spill).

We remark that previous works showed that a typical restricted product has degree at most 16, as opposed to 1. This difference is already crucial in fooling products over $\mathbb{Z}_m$ for composites m with good seed lengths, as we do not have (bit)-PRGs even for degree-2 polynomials over $\mathbb{Z}_6$.

6.1 The reduction

We will use the following standard construction of δ-almost k-wise independent distributions with marginals p.

Claim 32. *There exists an explicit δ-almost k-wise independent distribution T on $\{0,1\}^n$ with $\mathbf{E}[T_i] = 2^{-b}$ for every $i \in [n]$ which can be sampled using $O(b + k + \log(1/\delta) + \log\log n)$ bits.*

Proof. We sample an (δ, kb)-biased distribution D on $\{0,1\}^{nb}$ and b uniform bits U_b. By standard construction [NN93, AGHP92], D can be sampled using $O(b + \log(k/\varepsilon) + \log\log n)$ bits. Write $D = (D_1, \ldots, D_n)$ where each $D_i \in \{0,1\}^b$. We output $T \in \{0,1\}^n$, where $T_i = \mathsf{AND}_b(D \oplus U_b)$, where AND_b is the AND function on b bits. We have $\mathbf{E}[T_i] = 2^{-b}$ because U_b is uniform. By [MRT19b, Claim 3.7], T is $(\varepsilon \cdot 2^k)$-almost k-wise uniform. Setting $\varepsilon = 2^{-k} \cdot \delta$ proves the claim. □

Let C be a sufficiently large constant. Let

$$\begin{aligned} k &= C(\log(1/\varepsilon) + w) \\ \delta &= \theta^k \\ p &= 2^{-23w}\theta^3. \end{aligned} \tag{2}$$

Let D and T be two δ-almost k-wise independent distributions, with $\mathbf{E}[D_i] = 1/2$ and $\mathbf{E}[T_i] = p$ for every $i \in [n]$, and let P_1 be the PRG given by the theorem. The generator is

$$P := D + T \wedge P_1.$$

By Claim 32, this uses $s_1 + O_\theta(\log(1/\varepsilon) + w + \log\log n)$ bits.

6.2 Analysis

We first state a claim showing that if the number of non-constant f_i's ℓ in a block product f is much greater than its width w, then the bias $\|\mathbf{E}[f(U)]\|_{\mathsf{op}}$ is small. We defer its proof to Section 6.2.1.

Claim 33. *For integers w and q, let $f = \prod_{i=0}^{\ell} f_i$ be an (ℓ, w, q)-product over some matrix group $\mathcal{M}$ supported on $\mathcal{U}_\theta(d)$ for some $\ell \geq 2^{2w+2}\theta^2 \log(1/\varepsilon)$, where each f_i is non-constant. Then $\|\mathbf{E}[f(U)]\|_{\mathsf{op}} \leq \varepsilon$.*

Recall that $\ell \in [2^{30w}\theta^{-5}, 2^{45w}\theta^{-5}]$. Given D, T, let $f_{D,T} \colon \{0,1\}^T \to \mathcal{M}$ be the restricted product

$$f_{D,T}(x) := f(D + T \wedge x) = \prod_{i=0}^{\ell} f_i(D + T \wedge x).$$

We use $f_{D,T,i}(x)$ to denote $f_i(D + T \wedge x)$.

The following lemma shows that with high probability over D and T, the function $f_{D,T}$ is an $(\ell, 1, 3\log(1/\varepsilon))$-product, that is, a group program with a small spill. Note that this lemma is true for products over any group.

Lemma 34. *Let D and T be two distributions on $\{0,1\}^n$ defined in (2). Let $w \geq \log\log(1/\varepsilon) + 2\log(1/\theta)$ and $f = \prod_i f_i$ be an $(\ell, 3w, 2\log(1/\varepsilon))$-product, where $\ell \in [2^{30w}\theta^{-5}, 2^{45w}\theta^{-5}]$ and each f_i is non-constant. Then with probability $1 - \varepsilon$ over D and T, the function $f_{D,T}$ is an $(\ell, 1, 3\log(1/\varepsilon))$-product, where $\ell \geq 2^{3w}\theta^{-2}$ and each $f_{D,T,i}$ is non-constant.*

Theorem 30 follows from Claims 33 and 42 and Lemma 34.

Proof of Theorem 30. As $\ell \geq 2^{2w+2}\theta^{-2}\log(1/\varepsilon)$, by Claim 33, we have $\|\mathbf{E}[f(U)]\|_{\mathsf{op}} \leq \varepsilon$. By Lemma 34, with probability $1 - \varepsilon$ over D and T, the restricted function $f_{D,T} = \prod_i f_{D,T,i}$ is an $(\ell, 1, 3\log(1/\varepsilon))$-product, where $\ell \geq 2^{3w}\theta^{-2}$ and each $f_{D,T,i}$ is non-constant. As $w \geq \log\log(1/\varepsilon)$, again by Claim 33, we have $\|\mathbf{E}[f_{D,T}(U)]\|_{\mathsf{op}} \leq \varepsilon$. By our assumption, we have $\|\mathbf{E}[f_{D,T}(P_1)]\|_{\mathsf{op}} \leq \|\mathbf{E}[f_{D,T}(U)]\|_{\mathsf{op}} + \varepsilon \leq 2\varepsilon$. So altogether we have $\|\mathbf{E}[f(U)] - \mathbf{E}[f(G(U))]\|_{\mathsf{op}} \leq O(\varepsilon)$. The seed length follows from the construction. □

Proof of Lemma 34. To get some intuition, think of θ as a constant. Recall that the number of functions ℓ is roughly between 2^{30w} and 2^{45w}, and T is keeping each bit free with probability $p = 2^{-25w}$. Therefore, under a typical restriction, we expect for most functions in the product, only 1 bit is set to free, and very few functions have 2 free bits.

We first need to lower-bound the probability that a non-constant function remains non-constant under a random restriction.

Claim 35. *Let g be a non-constant function on w bits. For $p \in [0,1]$, let T be the distribution on $\{0,1\}^w$, where the coordinates T_i's are independent and $\mathbf{E}[T_i] = p$ for each $i \in [w]$. With probability at least $p \cdot ((1-p)/2)^{w-1}$ the function $g_{U,T}(x) := g(U + T \wedge x)$ is a non-constant function on 1 bit.*

Proof. Since g is non-constant, there is an $x \in \{0,1\}^w$ and a coordinate $j \in [w]$ such that $g(x + e_j) \neq g(x)$. The probability that only the coordinate T_j is 1 (and the rest are 0), and U agrees with x on the rest of the $w - 1$ coordinates is

$$\begin{aligned}\mathbf{Pr}\Big[(T = \{j\}) \wedge \bigwedge_{i \neq j} U_i = x_i\Big] &= \mathbf{Pr}\Big[T = \{j\}\Big] \cdot \mathbf{Pr}\Big[\bigwedge_{i \neq j} U_i = x_i\Big] \\ &= p \cdot (1-p)^{w-1} \cdot 2^{-(w-1)} = p \cdot \left(\frac{1-p}{2}\right)^{w-1}. \end{aligned}$$ □

We will use the following standard tail bound for almost k-wise independent random variables.

Lemma 36 (Lemma 8.1 in [LV20a]). *Let $X_1, \ldots, X_\ell$ be γ-almost t-wise independent random variables supported on $[0,1]$. Let $X := \sum_i X_i$, and $\mu := \mathbf{E}[X]$. We have*

$$\mathbf{Pr}\Big[|X - \mu| \geq \mu/2\Big] \leq O\left(\frac{t}{\mu}\right)^t + O\left(\frac{\ell}{\mu}\right)^t \gamma.$$

Proof of Lemma 34. We will show that for most choices of D and T, at least $2^{3w}\theta^{-2}$ of the $f_{D,T,i}$ depend on only 1 coordinate, and the ones that depend on at least 2 coordinates together form a $\log(1/\varepsilon)$-junta.

We first consider the set of functions $f_{D,T,i}$ that are restricted to 1-bit non-constant functions. Let

$$J_1 := \{i \in [\ell] : |T \cap I_i| = 1 \text{ and } f_{T,D,i} \text{ is non-constant}\}.$$

If D and T were *exactly* independent instead of almost-independent, then applying Claim 35 with our choice of $p \geq 2^{-23w}\theta^3$, we would have

$$\mathop{\mathbf{E}}_{D,T}\big[|J_1|\big] \geq \ell \cdot p \cdot \left(\frac{1-p}{2}\right)^{3w-1} \geq (2^{30w}\theta^{-5}) \cdot (2^{-23w}\theta^3) \cdot 2^{-3w} \geq 2^{4w}\theta^{-2}.$$

As (D,T) is δ-almost k-wise independent and $|I_i| \leq 3w$ for $i \in [\ell]$, the indicators $\mathbb{1}(i \in J_1) : i \in [\ell]$ are δ-almost $\lfloor k/(3w) \rfloor$-wise independent. So applying Lemma 36 with $t = \frac{C(\log(1/\varepsilon)+w)}{300w} \leq \lfloor k/(3w) \rfloor$ and $\gamma = \delta = \theta^k$, and recalling $k = C(\log(1/\varepsilon)+w)$, $\ell \leq 2^{45w}\theta^{-5}$, and $w \geq \log\log(1/\varepsilon) + \log(1/\theta)$, we have

$$\begin{aligned}\mathop{\mathbf{Pr}}_{D,T}\big[|J_1| \leq 2^{3w}\theta^{-2}\big] &\leq O\left(\frac{t}{2^{4w}\theta^{-2}}\right)^t + O\left(2^{41w}\theta^{-3}\right)^t \cdot \theta^k \\ &\leq 2^{-\Omega\left(w \cdot \frac{C(\log(1/\varepsilon)+w)}{300w}\right)} + \theta^{k/2} \\ &\leq \varepsilon. \end{aligned} \tag{3}$$

We now consider the $f_{D,T,i}$'s that depend on at least two coordinates. We will show that these functions altogether depend on at most $\log(1/\varepsilon)$ coordinates. As a result, we can think of these functions as a single $\log(1/\varepsilon)$-junta.

Let $J_{\geq 2} := \{i \in [\ell] : |I_i \cap T| \geq 2\}$ be the set of functions $f_{D,T,i}$'s that depend on at least 2 coordinates, and $Q := \bigcup_{i \in J_{\geq 2}} I_i \cap T$ be the collection of coordinates these functions depend on. Suppose $|Q| \geq \log(1/\varepsilon)$. Then as $|I_i \cap T| \geq 2$ for $i \in J_{\geq 2}$, it must be the case that some $u \leq \lceil \frac{\log(1/\varepsilon)}{2} \rceil$ of the subsets $I_i \cap T : i \in J_{\geq 2}$ together contain at least $2u$ many free coordinates. The probability of the latter event is at most

$$\binom{\ell}{u} \cdot \binom{u \cdot 3w}{2u} \cdot \Big(p^{2u} + \delta\Big).$$

Setting $u = \frac{\log(1/\varepsilon)}{2w} + 1$, and recalling $\ell \leq 2^{45w}\theta^{-5}$, $p = 2^{-23w}\theta^3$ and $\delta = \theta^k \leq p^{2u}$, the above is at most

$$\begin{aligned}\ell^u \cdot (6w)^{2u} \cdot 2p^{2u} &\leq (2^{45wu}\theta^{5u}) \cdot 2^{3u\log w} \cdot (2 \cdot 2^{-46w}\theta^6) \\ &\leq (2^{-2w}\theta)^u \leq \varepsilon. \end{aligned} \tag{4}$$

Let $I_0' := (T \cap I_0) \cup Q$. By (3) and (4), with probability $1 - 2\varepsilon$ over (D, T), we have $|J_1| \geq 2^{3w}\theta^{-2}$ and $|I_0'| \leq 3\log(1/\varepsilon)$. In this case, the function $f_{D,T}$ is a product of at least $2^{3w}\theta^{-2}$ non-constant 1-bit functions and a $(3\log(1/\varepsilon))$-junta. In other words, $f_{D,T} = \prod_i f_{D,T,i}$ is a $(\ell, 1, 3\log(1/\varepsilon))$-product, where $\ell \geq 2^{3w}\theta^{-2}$, and each $f_{D,T,i} : i \in [\ell]$ is non-constant. □

6.2.1 Long products have small bias: Proof of Claim 33

In this section, we prove Claim 33. We start with bounding the bias of a single arbitrary non-constant function on w bits.

Claim 37. *Let $\mathcal{M}$ be a group of matrices supported on $\mathcal{U}_\theta(d)$. We have $\|\mathbf{E}[g(U)]\|_{\mathsf{op}} \leq 1 - 2^{-(2w+2)}\theta^2$ for every non-constant function $g\colon \{0,1\}^w \to \mathcal{M}$.*

Proof. Let T be the uniform distribution on $\{0,1\}^n$. Note that $\mathbf{E}[g(U)] = \mathbf{E}_{U,T}[\mathbf{E}_{U'}[g(U + T \wedge U')]]$. Applying Claim 35 with $p = 1/2$, with probability at least $2^{-(2w-1)}$, the function $g_{U,T}(x) := g(U + T \wedge x)$ is a non-constant 1-bit function. Suppose $M_1 =: g_{U,T}(1) \neq g_{U,T}(0) := M_0$. By Claim 22, we have $\|(M_1 + M_0)/2\|_{\mathsf{op}} = \|M_1(I + M_1^{-1}M_0)/2\|_{\mathsf{op}} \leq 1 - \theta^2/8$. So

$$\begin{aligned}\big\|\mathbf{E}[g(U)]\big\|_{\mathsf{op}} &\leq (1 - 2^{-(2w-1)}) \cdot 1 + 2^{-(2w-1)} \cdot \|(M_1 + M_0)/2\|_{\mathsf{op}} \\ &\leq 1 - 2^{-(2w-1)} \cdot \big(1 - (1 - \theta^2/8)\big) \\ &= 1 - 2^{-(2w+2)}\theta^2.\end{aligned}$$

□

Proof of Claim 33. By Claim 37, we have $\|\mathbf{E}[f_i(U)]\|_{\mathsf{op}} \leq 1 - 2^{-(2w+2)}\theta^2$ for each $i \in [\ell]$. Hence, for $\ell \geq 2^{2w+2}\theta^2 \log(1/\varepsilon)$, we have

$$\big\|\mathbf{E}[f(U)]\big\|_{\mathsf{op}} \leq \prod_{i\in[\ell]} \big\|\mathbf{E}[f_i(U)]\big\|_{\mathsf{op}} \leq \big(1 - 2^{-(2w+2)}\theta^2\big)^\ell \leq \exp\big(-\ell \cdot 2^{-(2w+2)}\theta^2\big) \leq \varepsilon.$$

□

7 Width reduction of short products: Proof of Lemma 29

In this section, we prove Lemma 29. Recall that D, T are δ-almost k-wise independent distributions with $\mathbf{E}[D_i] = 1/2$ and $\mathbf{E}[T_i] = p$, where

$$\begin{aligned}k &= C\big(w + \log(m/\varepsilon)\big) \\ \delta &= (m \cdot w)^{-k} \\ p &= 2^{-C}.\end{aligned}$$

for a sufficiently large constant C.

Given T, say a coordinate $i \in [n]$ is *fixed* if $T_i = 0$ and is *free* if $T_i = 1$. Let $f(x) := \prod_{i=0}^{\ell} f_i(x_{I_i})$ be a $(\ell, 3w, 2\log(1/\varepsilon))$-product, where $\ell = m^5 \cdot 2^{30w}$. We will show that with high probability over T, (1) at most $\log(1/\varepsilon)$ of the $2\log(1/\varepsilon)$ coordinates in I_0 are free; (2) for most of the $I_i : i \geq 1$, at most $2w$ coordinates in each of them are free, and (3) in the remaining I_i's, there are at most $\log(1/\varepsilon)$ many free coordinates in total.

To proceed, let

$$J := \{i \in [\ell] : |T \cap I_i| \geq 2w\} \quad \text{and} \quad Q := \bigcup_{j\in J} I_j \cap T.$$

It suffices to show the following two claims.

Claim 38. $|Q| \leq \log(1/\varepsilon)$ *with probability* $1-\varepsilon$ *over* T.

Claim 39. $|T \cap I_0| \leq \log(1/\varepsilon)$ *with probability* $1-\varepsilon$ *over* T.

Proof of Lemma 29. Let $I_0' = (T \cap I_0) \cup Q$. By Claims 38 and 39, with probability $1-2\varepsilon$ over T, we have $|I_0'| \leq 2\log(1/\varepsilon)$, and for every $i \in [\ell] \setminus J$, we have $|T \cap I_i| \leq 2w$. Therefore, the function $f_{D,T}$ is a $(\ell, 2w, 2\log(1/\varepsilon))$-product. □

Proof of Claim 38. Suppose $|Q| \geq \log(1/\varepsilon)$. Then as $|T \cap I_j| \geq 2w$ for $j \in J$, it must be the case that some $u \leq \lceil \frac{\log(1/\varepsilon)}{2w} \rceil$ subsets $T \cap I_j : j \in J$ altogether contain $2w \cdot u$ many free coordinates. This happens with probability at most

$$\binom{\ell}{u} \cdot \binom{3w \cdot u}{2w \cdot u} \cdot \Big(p^{2w \cdot u} + \delta\Big).$$

Setting $u = \frac{\log(1/\varepsilon)}{3w} + 1$, and recalling $\ell \leq m^5 2^{30w}$ and $\delta = (m \cdot w)^{-k} \leq p^{2w \cdot u}$, the above is at most

$$\begin{aligned}
\ell^u \cdot 2^{3w \cdot u} \cdot 2p^{2w \cdot u} &\leq \left(m^5 2^{30w}\right)^u \cdot 2^{3w \cdot u} \cdot 2^{-C \cdot 2w \cdot u + 1} \\
&\leq 2^{u\left(33w + 5\log m - C \cdot 2w\right)} \\
&\leq 2^{-u \cdot 3w} \leq \varepsilon
\end{aligned}$$

where in the second last inequality we used $w \geq \log m$. □

Proof of Claim 39. Recall that $|I_0| \leq 2\log(1/\varepsilon)$, and $\delta = (m \cdot w)^{-k} \leq p^{\log(1/\varepsilon)}$. So

$$\begin{aligned}
\mathbf{Pr}\Big[|T \cap I_0| \geq \log(1/\varepsilon)\Big] &\leq \binom{|I_0|}{\log(1/\varepsilon)}\Big(p^{\log(1/\varepsilon)} + \delta\Big) \\
&\leq 2^{2\log(1/\varepsilon)} \cdot 2^{-C\log(1/\varepsilon)+1} \\
&\leq \varepsilon/2.
\end{aligned}$$

□

8 Fooling $(1, w, 3\log(1/\varepsilon))$-products over groups

In this section, we show how to extend the PRGs for $(\ell, 1, 0)$-products over p-groups (Theorem 5) and commutative groups [GKM15] to fool $(\ell, 1, 3\log(1/\varepsilon))$-products.

8.1 p-groups

We use the fact that our generator in Theorem 5 is simply the XOR of independent copies of small-bias distributions. The following claim shows that conditioning on a small number of bits of a small-bias distribution remains small bias.

Claim 40. *Let* D *be an* ε*-biased distribution on* $\{0,1\}^n$*. For any set* S *and* $y \in \{0,1\}^S$*, the distribution of* D *conditioned on* $D_S = y$ *is* $(2^{|S|+1}\varepsilon)$*-biased on* $\{0,1\}^{[n]\setminus S}$.

Proof. We may assume $\varepsilon \leq 2^{-(|S|+1)}$, for otherwise the claim is vacuous. For a subset $T \subseteq [n]$, let $\chi_T(x) := (-1)^{\sum_{i\in T} x_i}$ be any parity test. Let T be any nonempty subset of $[n] \setminus S$. First observe that

$$\mathbb{1}(D_S = y) = \prod_{i \in S} \frac{1 - \chi_{\{i\}}(D)}{2} = 2^{-|S|} \sum_{S' \subseteq S} (-1)^{|S'|} \chi_{S'}(D).$$

Taking expectations on both sides and applying the triangle inequality, we have $\mathbf{Pr}[D_S = y] \geq 2^{-|S|} - \varepsilon \geq 2^{-(|S|+1)}$. Note that

$$\begin{aligned}\mathbf{E}[\chi_T(D) \mid D_S = y]\,\mathbf{Pr}[D_S = y] &= \mathbf{E}[\chi_T(D) \cdot \mathbb{1}(D_S = y)] \\ &= \mathbf{E}\Big[\chi_T(D) \cdot \prod_{i \in S} \frac{1 - \chi_{\{i\}}(D)}{2}\Big] \\ &= 2^{-|S|} \sum_{S' \subseteq S} (-1)^{|S'|}\, \mathbf{E}\big[\chi_{T \cup S'}(D)\big].\end{aligned}$$

So its magnitude is bounded by ε. Therefore, $\big|\mathbf{E}[\chi_T(D) \mid D_S = y]\big| \leq \mathbf{Pr}[D_S = y]^{-1}\varepsilon \leq 2^{|S|+1}\varepsilon$. □

Corollary 41. *There is a PRG that ε-fools $(n, 1, 3\log(1/\varepsilon))$-products over any p-groups of order m with seed length $O_m(\log(n/\varepsilon))$.*

Proof. Recall our generator for p-groups in Theorem 5 is simply the XOR of independent copies of $(\varepsilon/n)^c$-biased distributions. By Claim 40, for any fixing of the input bits of f_0 in each copy, each distribution remains $(2/\varepsilon)(\varepsilon/n)^c$-biased. □

8.2 Commutative groups

We now show that the Gopalan–Kane–Meka PRG fools $(\ell, 1, 3\log(1/\varepsilon))$-products. We will use the following PRG by Gopalan, Kane, and Meka [GKM18] that fools $(\ell, 1, 0)$-products over $\mathbb{C}$. A simple argument shows that the same PRG also fools $(\ell, 1, 3\log(1/\varepsilon))$-products.

Claim 42. *There is an explicit PRG P that ε-fools $(\ell, 1, 3\log(1/\varepsilon))$-products over commutative groups of order m with seed length $O_m(\log(\ell/\varepsilon)(\log\log(\ell/\varepsilon)^2)$.*

Lemma 43 (Theorem 1.1 and Lemma 9.1 in [GKM18])**.** *There is an explicit $P_{\mathsf{GKM}} \colon \{0,1\}^s \to \{0,1\}^n$ where $s = O(\log(\ell/\varepsilon))(\log\log(\ell/\varepsilon))^2$ such that the following holds. If $w \in \mathbb{R}^n$ satisfies $\sum_i |w_i| \leq W$, then*

$$\mathrm{dist}_{TV}\big(\langle w, U\rangle, \langle w, P_{\mathsf{GKM}}(U)\rangle\big) \leq O(\sqrt{W}) \cdot \varepsilon.$$

Proof of Claim 42. By Claim 21, it suffices to fool the product over each irreducible representation of G with error $\varepsilon/\sqrt{m}$. Since G is commutative, all its irreps are 1-dimensional. Moreover, they are supported on subsets of $\mathbb{C}_m := \{z \in \mathbb{C} : |z|^m = 1\}$. Let $f = \prod_{i=0}^{\ell} f_i$ be an $(\ell, 1, 3\log(1/\varepsilon))$-product over $\mathbb{C}_m$.

Let $\omega := e^{i\frac{2\pi}{m}}$. Note that for any 1-bit function $g \colon \{0,1\} \to \mathbb{C}_m$ we can write g as

$$g(y) = \omega^{a_1 y + a_0(1-y)} = \omega^{a_0} \cdot \omega^{(a_1 - a_0)y}$$

for some $a_0, a_1 \in \{0, \ldots, m-1\}$. We can also write $f_0(x_{I_0})$ as $\omega^{h(x_{I_0})}$, for some $h \colon \{0,1\}^{I_0} \to \{0, \ldots, m-1\}$. Therefore, f has the form of

$$f(x) = \omega^b \cdot \omega^{\sum_{j \in J} a_j x_j + h(x_{I_0})},$$

for some coefficients b and a_j's taking values from $\{0, \ldots, m-1\}$, and J and I_0 are disjoint subsets of $[n]$. Write $I_0 = \{r_1 < \cdots < r_{|I_0|}\}$ for some $r_j \in [n]$. Consider the integer-valued function $F \colon \{0,1\}^\ell \to \mathbb{Z}$ defined by

$$F(x) := \sum_{j \in J} a_j x_j + 2^{\lceil \log(m\ell) \rceil} \sum_{j=1}^{|I_0|} 2^{j-1} x_{r_j}.$$

So the first $\lceil \log m\ell \rceil$ bits of $F(x)$ encode the first sum, and the last $|I_0|$ bits are the decimal encoding of the binary string x_{I_0}. Note that we can compute $f(x)$ given $F(x)$. Moreover, $F(x) = \langle w, x \rangle$ for some $w \in \mathbb{R}^n$ with $\sum_i |w_i| \le O(m\ell/\varepsilon^3)$. Therefore, if we let P be the PRG in Lemma 43 with error $O(\frac{\varepsilon^3}{m\ell})$, which uses a seed of $O(\log(m\ell/\varepsilon))(\log\log(m\ell/\varepsilon))^2$ bits, then it follows that $\mathrm{dist}_{TV}(F(U), F(P_{\mathsf{GKM}}(U)) \le \varepsilon/\sqrt{m}$. □

9 Proof of mixing characterization of Dedekind groups, Lemma 9

This proof is provided by Yves de Cornulier on https://mathoverflow.net/a/482286/8271.

Lemma 44. *A finite group is mixing if and only if it is Dedekind.*

We first prove the following equivalent condition of mixing.

Claim 45. *For any group element g and representation ρ of a group G, $\rho(g)$ has no eigenvalue 1 if and only if $\ker(\rho(g) - I)$ is a subrepresentation.*

Proof. Consider the subspace $W_g := \{v : \rho(g)v = v\}$. W_g is a subrepresentation if and only if $\rho(g')W_g \subseteq W_g$ for every $g' \in G$. By the definition of W_g, this is equivalent to $\rho(g)(\rho(g')v) = \rho(g')v$ for every $v \in W_g$, which means $\rho(g')v$ is an eigenvector of $\rho(g)$ with eigenvalue 1, unless $v = 0$. □

The proof of Lemma 9 follows from the following two claims. Here we use the equivalence that a group G is Dedekind if and only if every subgroup of G is normal.

Claim 46. *If G is Dedekind, then $\ker(\rho(g) - I)$ is a subrepresentation for every g and ρ.*

Proof. Take an element $g \in G$. By definition of Dedekind, every subgroup in G is normal. In particular $\langle g \rangle$ is also normal.

Take $v \in W_g := \{v : \rho(g)v = v\}$. To show that W_g is a subrepresentation, we need to show that $\rho(g)(\rho(h)v) = \rho(h)v$ for every h. But this is equivalent to showing

$$\rho(h)^{-1}\rho(g)\rho(h)v = \rho(h^{-1}gh)v = v.$$

Since $\langle g \rangle$ is normal, we have $h^{-1}gh = g^i$ for some i. It is clear that $\rho(g^i)v = \rho(g)^i v = \rho(g)^{i-1}(\rho(g)v) = \rho(g)^{i-1}v = \cdots = v$. So indeed W_g is a subrepresentation. □

Claim 47. *If* $\ker(\rho(g)-I)$ *is a subrepresentation for every* g *and* ρ *of* G*, then* G *is Dedekind.*

Proof. As in the previous claim, we consider $W_g = \ker(\rho(g)-I) = \{v : \rho(g)v = v\}$. Note that for $v \in W_g$, we have $\rho(g^i)v = \rho(g)^i v = v$. Suppose W_g is a subrepresentation. Take $h \in G$ and $v \in W_g$. Since $\rho(h)W_g \subseteq W_g$, we have $\rho(g)\rho(h)v = \rho(h)v$. That means $\rho(h^{-1}gh)v = v$ for every $v \in W_g$. This implies $\langle h^{-1}gh\rangle = \langle g\rangle$, meaning $h^{-1}gh \in \langle g\rangle$ and thus $\langle g\rangle$ is normal. □

References

[AGHP92] Noga Alon, Oded Goldreich, Johan Håstad, and René Peralta. Simple constructions of almost k-wise independent random variables. *Random Structures & Algorithms*, 3(3):289–304, 1992. 5, 6.1

[AIK06] Benny Applebaum, Yuval Ishai, and Eyal Kushilevitz. Cryptography in NC^0. *SIAM J. on Computing*, 36(4):845–888, 2006. 1.1

[ALW01] Noga Alon, Alexander Lubotzky, and Avi Wigderson. Semi-direct product in groups and zig-zag product in graphs: Connections and applications. In *IEEE Symp. on Foundations of Computer Science (FOCS)*, pages 630–637, 2001. 1.1

[ASWZ96] Roy Armoni, Michael E. Saks, Avi Wigderson, and Shiyu Zhou. Discrepancy sets and pseudorandom generators for combinatorial rectangles. In *37th IEEE Symp. on Foundations of Computer Science (FOCS)*, pages 412–421, 1996. 1

[AW89] Miklos Ajtai and Avi Wigderson. Deterministic simulation of probabilistic constant-depth circuits. *Advances in Computing Research - Randomness and Computation*, 5:199–223, 1989. 5

[Bar89] David A. Mix Barrington. Bounded-width polynomial-size branching programs recognize exactly those languages in NC^1. *J. of Computer and System Sciences*, 38(1):150–164, 1989. 2

[BCC$^+$17] Jonah Blasiak, Thomas Church, Henry Cohn, Joshua A. Grochow, and Chris Umans. Which groups are amenable to proving exponent two for matrix multiplication? *CoRR*, abs/1712.02302, 2017. 1.1

[BV10] Andrej Bogdanov and Emanuele Viola. Pseudorandom bits for polynomials. *SIAM J. on Computing*, 39(6):2464–2486, 2010. 2, 2, 5

[CHHL19] Eshan Chattopadhyay, Pooya Hatami, Kaave Hosseini, and Shachar Lovett. Pseudorandom generators from polarizing random walks. *Theory Comput.*, 15:1–26, 2019. 1, 1.1

[CKSU05] Henry Cohn, Robert D. Kleinberg, Balázs Szegedy, and Christopher Umans. Group-theoretic algorithms for matrix multiplication. In *IEEE Symp. on Foundations of Computer Science (FOCS)*, pages 379–388, 2005. 1.1

[DF04] David S. Dummit and Richard M. Foote. *Abstract Algebra*. Wiley, 3rd edition, 2004. 2

[DHH20] Dean Doron, Pooya Hatami, and William M. Hoza. Log-seed pseudorandom generators via iterated restrictions. In Shubhangi Saraf, editor, *35th Computational Complexity Conference, CCC 2020, July 28-31, 2020, Saarbrücken, Germany (Virtual Conference)*, volume 169 of *LIPIcs*, pages 6:1–6:36. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. 1

[Dia88] Persi Diaconis. *Group representations in probability and statistics*, volume 11 of *Institute of Mathematical Statistics Lecture Notes—Monograph Series*. Institute of Mathematical Statistics, Hayward, CA, 1988. 3

[DILV24] Harm Derksen, Peter Ivanov, Chin Ho Lee, and Emanuele Viola. Pseudorandomness, symmetry, smoothing: Ii, 2024. 5.1

[DLV24] Harm Derksen, Chin Ho Lee, and Emanuele Viola. Boosting uniformity in quasirandom groups: fast and simple. In *IEEE Symp. on Foundations of Computer Science (FOCS)*, 2024. 3

[FK18] Michael A. Forbes and Zander Kelley. Pseudorandom generators for read-once branching programs, in any order. In *IEEE Symp. on Foundations of Computer Science (FOCS)*, 2018. 2, 2, 5.1, 28

[GKM15] Parikshit Gopalan, Daniel Kane, and Raghu Meka. Pseudorandomness via the discrete fourier transform. In *IEEE Symp. on Foundations of Computer Science (FOCS)*, pages 903–922, 2015. 1.1, 8

[GKM18] Parikshit Gopalan, Daniel M. Kane, and Raghu Meka. Pseudorandomness via the discrete Fourier transform. *SIAM J. Comput.*, 47(6):2451–2487, 2018. 1, 1, 1, 5, 8.2, 43

[GMR$^+$12] Parikshit Gopalan, Raghu Meka, Omer Reingold, Luca Trevisan, and Salil Vadhan. Better pseudorandom generators from milder pseudorandom restrictions. In *IEEE Symp. on Foundations of Computer Science (FOCS)*, 2012. 1, 5

[GMRZ13] Parikshit Gopalan, Raghu Meka, Omer Reingold, and David Zuckerman. Pseudorandom generators for combinatorial shapes. *SIAM J. Comput.*, 42(3):1051–1076, 2013. 1

[Gow17] W. T. Gowers. Generalizations of Fourier analysis, and how to apply them. *Bull. Amer. Math. Soc. (N.S.)*, 54(1):1–44, 2017. 3

[GV22] W. T. Gowers and Emanuele Viola. Mixing in non-quasirandom groups. In *ACM Innovations in Theoretical Computer Science conf. (ITCS)*, 2022. 1.1, 3

[HH23] Pooya Hatami and William Hoza. Theory of unconditional pseudorandom generators. *Electron. Colloquium Comput. Complex.*, TR23-019, 2023. 1

[HLV18] Elad Haramaty, Chin Ho Lee, and Emanuele Viola. Bounded independence plus noise fools products. *SIAM J. Comput.*, 47(2):493–523, 2018. 1

[KNP11] Michal Koucký, Prajakta Nimbhorkar, and Pavel Pudlák. Pseudorandom generators for group products: extended abstract. In *STOC*, pages 263–272. ACM, 2011. 1

[Kui02] Jack B. Kuipers. Quaternions in computer graphics and robotics. In *SIGGRAPH 2002 Course Notes*, San Antonio, TX, 2002. ACM SIGGRAPH. 1.1

[Lee19] Chin Ho Lee. Fourier bounds and pseudorandom generators for product tests. In *34th Computational Complexity Conference*, volume 137. 2019. 1, 1

[LMS10] Shachar Lovett, Partha Mukhopadhyay, and Amir Shpilka. Pseudorandom generators for $CC^0[p]$ and the Fourier spectrum of low-degree polynomials over finite fields. In *51th IEEE Symp. on Foundations of Computer Science (FOCS)*. IEEE, 2010. 2

[Lov09] Shachar Lovett. Unconditional pseudorandom generators for low degree polynomials. *Theory of Computing*, 5(1):69–82, 2009. 2, 2, 5

[LPV22] Chin Ho Lee, Edward Pyne, and Salil Vadhan. Fourier growth of regular branching programs. In *Approximation, randomization, and combinatorial optimization. Algorithms and techniques*, volume 245 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages Art. No. 2, 21. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2022. 1, 2, 5.1, 5.1, 28

[LRTV09a] Shachar Lovett, Omer Reingold, Luca Trevisan, and Salil Vadhan. Pseudorandom bit generators that fool modular sums. In *Approximation, randomization, and combinatorial optimization*, volume 5687 of *Lecture Notes in Comput. Sci.*, pages 615–630. Springer, Berlin, 2009. 1

[LRTV09b] Shachar Lovett, Omer Reingold, Luca Trevisan, and Salil P. Vadhan. Pseudorandom bit generators that fool modular sums. In *13th Workshop on Randomization and Computation (RANDOM)*, volume 5687 of *Lecture Notes in Computer Science*, pages 615–630. Springer, 2009. 1, 2, 17, 5

[Lu02] Chi-Jen Lu. Improved pseudorandom generators for combinatorial rectangles. *Combinatorica*, 22(3):417–433, 2002. 1

[LV18] Chin Ho Lee and Emanuele Viola. The coin problem for product tests. *ACM Trans. Comput. Theory*, 10(3):Art. 14, 10, 2018. 1

[LV20a] Chin Ho Lee and Emanuele Viola. More on bounded independence plus noise: Pseudorandom generators for read-once polynomials. *Theory of Computing*, 16:1–50, 2020. Available at http://www.ccs.neu.edu/home/viola/. 1, 5, 36

[LV20b] Chin Ho Lee and Emanuele Viola. More on bounded independence plus noise: pseudorandom generators for read-once polynomials. *Theory Comput.*, 16:Paper No. 7, 50, 2020. 1

[MRT19a] Raghu Meka, Omer Reingold, and Avishay Tal. Pseudorandom generators for width-3 branching programs. In Moses Charikar and Edith Cohen, editors, *ACM Symp. on the Theory of Computing (STOC)*, pages 626–637. ACM, 2019. 1, 1

[MRT19b] Raghu Meka, Omer Reingold, and Avishay Tal. Pseudorandom generators for width-3 branching programs. In *STOC'19—Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 626–637. ACM, New York, 2019. 5, 5, 6, 6.1

[MZ09a] Raghu Meka and David Zuckerman. Small-bias spaces for group products. In *13th Workshop on Randomization and Computation (RANDOM)*, volume 5687 of *Lecture Notes in Computer Science*, pages 658–672. Springer, 2009. 1, 2, 17, 5

[MZ09b] Raghu Meka and David Zuckerman. Small-bias spaces for group products. In *Approximation, randomization, and combinatorial optimization*, volume 5687 of *Lecture Notes in Comput. Sci.*, pages 658–672. Springer, Berlin, 2009. 1

[NN90] J. Naor and M. Naor. Small-bias probability spaces: efficient constructions and applications. In *22nd ACM Symp. on the Theory of Computing (STOC)*, pages 213–223. ACM, 1990. 1

[NN93] Joseph Naor and Moni Naor. Small-bias probability spaces: efficient constructions and applications. *SIAM J. on Computing*, 22(4):838–856, 1993. 5, 6.1

[PT01] Pierre Péladeau and Denis Thérien. On the languages recognized by nilpotent groups (a translation of "sur les langages reconnus par des groupes nilpotents"). *Electron. Colloquium Comput. Complex.*, TR01-040, 2001. 2

[RSV13] Omer Reingold, Thomas Steinke, and Salil P. Vadhan. Pseudorandomness for regular branching programs via Fourier analysis. In *Workshop on Randomization and Computation (RANDOM)*, pages 655–670, 2013. 1, 2

[RSW06] Eyal Rozenman, Aner Shalev, and Avi Wigderson. Iterative construction of cayley expander graphs. *Theory Comput.*, 2(5):91–120, 2006. 1.1

[Ser77] Jean Pierre Serre. *Linear Representations of Finite Groups.* Springer, 1977. 3

[Ste12] Thomas Steinke. Pseudorandomness for permutation branching programs without the group theory. *Electron. Colloquium Comput. Complex.*, TR12-083, 2012. 1, 1.1

[Sun23] Xiaorui Sun. Faster isomorphism for p-groups of class 2 and exponent p. In *STOC*, pages 433–440. ACM, 2023. 1.1

[Ter99] Audrey Terras. *Fourier analysis on finite groups and applications*, volume 43 of *London Mathematical Society Student Texts*. Cambridge University Press, Cambridge, 1999. 3

[Vad12] Salil P. Vadhan. Pseudorandomness. *Foundations and Trends in Theoretical Computer Science*, 7(1-3):1–336, 2012. 4

[Vio09a] Emanuele Viola. On the power of small-depth computation. *Foundations and Trends in Theoretical Computer Science*, 5(1):1–72, 2009. 1.1

[Vio09b] Emanuele Viola. The sum of d small-bias generators fools polynomials of degree d. *Computational Complexity*, 18(2):209–217, 2009. 2, 2, 15, 5

[Wat13] Thomas Watson. Pseudorandom generators for combinatorial checkerboards. *Computational Complexity*, 22(4):727–769, 2013. 1

[Wig10] Avi Wigderson. Representation theory of finite groups, and applications. Available at http://www.math.ias.edu/∼avi/TALKS/Green_Wigderson_lecture.pdf, 2010. 3

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il