

Pseudorandom Generators for Width-3 Branching Programs

Raghu Meka*
UCLA

Omer Reingold†
Stanford University

Avishay Tal‡
Stanford University

June 4, 2018

Abstract

We construct pseudorandom generators of seed length $\tilde{O}(\log(n) \cdot \log(1/\varepsilon))$ that ε -fool *ordered* read-once branching programs (ROBPs) of width 3 and length n . For *unordered* ROBPs, we construct pseudorandom generators with seed length $\tilde{O}(\log(n) \cdot \text{poly}(1/\varepsilon))$. This is the first improvement for pseudorandom generators fooling width 3 ROBPs since the work of Nisan [Nis92].

Our constructions are based on the “iterated milder restrictions” approach of Gopalan et al. [GMR⁺12] (which further extends the Ajtai-Wigderson framework [AW85]), combined with the INW-generator [INW94] at the last step (as analyzed by Braverman et al. [BRRY14]). For the unordered case we combine iterated milder restrictions with the generator of Chattopadhyay et al. [CHHL18].

Our main technical contributions are:

1. A relabeling technique allowing us to analyze a relabeled version of the given branching program, which turns out to be much easier.
2. Treating the number of colliding layers in a branching program as a progress measure and showing that it reduces significantly under pseudorandom restrictions.

In addition, we achieve nearly optimal seed-length $\tilde{O}(\log(n/\varepsilon))$ for the classes of: (1) read-once polynomials on n variables, (2) locally-monotone ROBPs of length n and width 3 (generalizing read-once CNFs and DNFs), and (3) constant-width ROBPs of length n having a layer of width 2 in every consecutive $\text{poly} \log(n)$ layers.

*raghum@cs.ucla.edu. Supported by NSF grant CCF-1553605.

†reingold@stanford.edu. Supported in part by NSF grant CCF-1763299.

‡avishay.tal@gmail.com. Supported by a Motwani Postdoctoral Fellowship and by NSF grant CCF-1763299.

Contents

1	Introduction	1
1.1	The Ajtai-Wigderson framework	2
2	Proof Overview	3
2.1	Pseudorandomly assigning half of the coordinates	3
2.1.1	Reducing width-3 ROBPs to the XOR of short width-3 ROBPs	4
2.1.2	Pseudorandom restrictions for the XOR of short width-3 ROBPs	5
2.2	Simplification under pseudorandom restrictions	6
2.3	The proof of Theorem 3	7
2.4	Organization	7
3	Preliminaries	8
3.1	Restrictions	8
3.2	Fourier analysis of Boolean functions	8
3.3	Small-biased distributions	9
3.4	Standard tail bounds for k -wise independence	9
3.5	Branching programs	9
3.6	Helpful lemmas	11
4	From width-3 ROBPs to the XOR of short ROBPs	12
5	Pseudorandom restrictions for the XOR of short ROBPs	16
5.1	Low-Variance Case	17
5.2	High-Variance Case	22
6	Assigning all the variables: a pseudorandom generator for the XOR of short ROBPs	25
6.1	PRG for the XOR of many functions with block-length b	26
6.2	A thought experiment	29
6.3	The actual generator	31
6.4	Pseudorandom generator for read-once polynomials	33
7	Pseudorandom generators for width-3 ROBPs	33
7.1	Proof overview	34
7.2	Reducing the length of $(3, \ell, m)$ -ROBPs	35
7.3	PRGs for ROBPs with few colliding layers	37
7.4	Proof of Theorem 1	40
7.5	Proof of Claim 7.14	44
7.6	Pseudorandom generator for unordered 3ROBPs	44
7.7	Pseudorandom generator for locally-monotone width-3 ROBPs	45
A	Appendices	48
A.1	Proof of Theorem 3.10	48
A.2	Restatement of XOR-lemma for functions fooled by small-biased spaces	49
A.3	Pseudorandom restrictions for the composition of 3ROBPs	50

1 Introduction

A central challenge in complexity theory is to understand the trade-off between *space* and *randomness* as resources and in particular, whether $\text{BPL} = \text{L}$. One of the main techniques we have for approaching this question is to design pseudorandom generators that fool tests computable in small space. The latter question can be elegantly captured in the language of designing pseudorandom generators for **read-once branching programs**; we define these objects next.

Definition 1.1. For $w, n \in \mathbb{N}$, a **read-once branching program (ROBP)** of width w and length n is a layered directed graph B with $n + 1$ layers where all but the first layer have at most w nodes, the first layer has a single vertex designated the **start vertex**, and the vertices in the last layer are either labeled **accept** or **reject**. Each vertex in the first n layers has exactly two outgoing edges to vertices in the next layer with one labeled 1 and the other labeled -1 .

Given a ROBP as above, it defines a function $B : \{\pm 1\}^n \rightarrow \{\pm 1\}$ naturally where on input $x \in \{\pm 1\}^n$ starting from the start vertex, you follow the edges labeled by x_i for $1 \leq i \leq n$ and output -1 if the last vertex reached is accepting and 1 otherwise.

Derandomizing space-bounded computations is fundamentally related to designing pseudorandom generators (and *hitting set generators*) for ROBPs as above.

Definition 1.2. Given a class of functions $\mathcal{F} = \{f : \{\pm 1\}^n \rightarrow \mathbb{R}\}$, a function $G : \{\pm 1\}^r \rightarrow \{\pm 1\}^n$ is a **pseudorandom generator (PRG)** with error ε (or ε -fools) \mathcal{F} if for every $f \in \mathcal{F}$,

$$\left| \Pr_{x \in_u \{\pm 1\}^n} [f(x)] - \Pr_{y \in_u \{\pm 1\}^r} [f(G(y))] \right| \leq \varepsilon.$$

We say the generator is **log-space explicit** if G can be computed in space logarithmic in the output length n and refer to r as the **seed-length** of the generator.

It is well-known by now that if there exists a log-space explicit PRG (or even a hitting set generator) with constant error that fools ROBPs of width w and length n with seed-length $O(\log n)$, then $\text{BPL} = \text{L}$. In this vein, a seminal result of Nisan [Nis92] gave a log-space explicit PRG that ε -fools ROBPs of width w and length n with seed-length $r = O((\log n) \cdot \log(wn/\varepsilon))$. Despite significant attention, improving Nisan's PRG has been a fundamental bottleneck in pseudorandomness. For width $w = 2$, it is known that **small-bias spaces** fool width two ROBPs ([SZ95, BDVY13]), leading to a PRG with seed-length $O(\log(n/\varepsilon))$. However, even for the case of ε a constant and width $w = 3$, the best provable PRG had seed-length $O(\log^2 n)$ —no better than what Nisan's PRG gives for polynomial width ROBPs.¹ In this work, we obtain the first improvement over Nisan's PRG for width-3 ROBPs:

Theorem 1 (Main Theorem). For any $\varepsilon > 0$, there exists a log-space explicit PRG that ε -fools width-3 ROBPs with seed-length² $\tilde{O}(\log(n/\varepsilon)) + O(\log(1/\varepsilon) \cdot \log(n))$.

¹Nearly optimal hitting-sets generators for width-3 ROBPs were given in [SZ11, GMR⁺12].

²Henceforth, $\tilde{O}(t)$ is used to denote $O(t \cdot \text{poly} \log(t))$.

We in fact also obtain PRG’s with nearly optimal dependence for constant error for the bigger class of **unordered** width-3 ROBPs, which are functions computable by ROBPs under some unknown permutation (see Section 3.5 for the formal definition).

Theorem 2. *For any $\varepsilon > 0$, there exists a log-space explicit PRG that ε -fools unordered width-3 ROBPs with seed-length $\tilde{O}(\log(n/\varepsilon)) + O(\text{poly}(1/\varepsilon) \cdot \log(n))$.*

A special class of unordered width-4 ROBPs that have received recent attention are **read-once polynomials** (see [Tre10, LV17]) for which we give a PRG with nearly optimal seed-length both in terms of the error and input length (up to $\text{poly}(\log \log)$ factors):

Theorem 3. *There exists a log-space explicit ε -PRG for the class of read-once polynomials on n variables with seed-length $\tilde{O}(\log(n/\varepsilon))$.*

In comparison, the best previous PRG for read-once polynomials had seed-length $\tilde{O}(\log(n/\varepsilon)) \cdot \log(1/\varepsilon)$, thus in particular needed $\tilde{O}(\log^2 n)$ seed-length to fool read-once polynomials with polynomially small error.

Our results rely on several new conceptual ideas as well as technical ingredients, including PRGs fooling other interesting intermediate classes of ROBPs, that we believe could be useful for other applications especially in the context of obtaining PRGs for constant-width ROBPs. Our results rely on the framework of *iterative mild random restrictions* introduced in [GMR⁺12] and further developed in [GY14, GKM15, HLV17], the latter of which also present an elegant alternate view of the technique as *bounded-independence plus noise*. We describe this framework, our proof techniques next.

1.1 The Ajtai-Wigderson framework

The Ajtai-Wigderson [AW85] framework, that was revived and refined in the work of Gopalan et al. [GMR⁺12], provides a “recipe” for constructing PRGs for classes of functions that simplify under (pseudo)random restrictions. Roughly speaking, in order to fool a class of functions \mathcal{C} it suffices to fool \mathcal{C} under pseudo-random restrictions keeping each variable alive with probability p . Equivalently, it suffices to pseudorandomly assign p -fraction of the coordinates while approximately preserving the acceptance probability (on average) of every function $f \in \mathcal{C}$. Suppose we have such a pseudorandom partial assignment, and assume that the class of functions \mathcal{C} is closed under restrictions. Then, iteratively applying a pseudorandom partial assignment on the remaining coordinates until we assigned all of them gives us a pseudorandom generator for \mathcal{C} . We expect to assign all the coordinates after $O(p^{-1} \cdot \log n)$ iterations, thus if each iteration requires at most s random bits, we get a PRG with seed-length $O(s \cdot p^{-1} \cdot \log n)$. Naively, it seems impossible to achieve nearly-logarithmic seed length using this approach, however this was obtained in the work of [GMR⁺12] as explained next.

Achieving Near-Logarithmic Seed-Length. In the work of Gopalan et al. [GMR⁺12] the Ajtai-Wigderson approach was used to construct ε -PRGs for read-once CNFs (and read-once DNFs) with seed length $\tilde{O}(\log(n/\varepsilon))$. In order to achieve nearly-logarithmic seed-length Gopalan et al. showed that one can assign a constant fraction of the coordinates while preserving the acceptance probability up to error $\text{poly}(\varepsilon/n)$ using only $s = \tilde{O}(\log(n/\varepsilon))$

bits of randomness. Plugging into the estimates above would give naively seed-length $\tilde{O}(\log(n/\varepsilon) \cdot \log(n))$. In order to avoid the additional factor of $\log(n)$, they prove that after pseudorandomly assigning all but $1/\text{poly} \log(n)$ of the coordinates, the function simplifies significantly so that it can be fooled using additional $O(\log(n/\varepsilon))$ -random bits.

We describe the approach more precisely. A p -pseudorandom restriction against a class of functions \mathcal{C} specifies a set $T \subseteq [n]$ of roughly $p \cdot n$ of the coordinates, and an assignment $x \in \{\pm 1\}^T$ to these coordinates, such that for any $f \in \mathcal{C}$:

$$\mathbf{E}_{T,x} \mathbf{E}_{y \in \{\pm 1\}^{[n] \setminus T}} [f(x \circ y)] = \mathbf{E}_{z \in \{\pm 1\}^n} [f(z)] \pm \varepsilon$$

where $(x \circ y)$ denotes the string whose T -coordinates are taken from x and other coordinates are taken from y . The main observation of Gopalan et al. [GMR⁺12] is that given T , it suffices that x would fool the Bias-function, defined as

$$\text{Bias}_T f(x) \triangleq \mathbf{E}_{y \in \{\pm 1\}^{[n] \setminus T}} [f(x \circ y)].$$

This is due to the fact that

$$\left| \mathbf{E}_{z \in \{\pm 1\}^n} [f(z)] - \mathbf{E}_x \mathbf{E}_{y \in \{\pm 1\}^{[n] \setminus T}} [f(x \circ y)] \right| = \left| \mathbf{E}_{z \in \{\pm 1\}^T} [\text{Bias}_T f(z)] - \mathbf{E}_x [\text{Bias}_T f(x)] \right|.$$

The observation that it suffices to fool the bias-function instead of just fooling the restricted functions, enabled Gopalan [GMR⁺12] to use “mild” restrictions with $p = \Omega(1)$ for the class of CNFs/DNFs. They show that in this case, the average of the restricted functions (i.e., the bias-function) is much easier to fool than a typical restricted function.

2 Proof Overview

Similarly to [GMR⁺12], in order to achieve a PRG with nearly-logarithmic seed-length fooling width-3 ROBPs, we show that:

1. We can pseudorandomly assign half the input coordinates while preserving the acceptance probability (on average) of every width-3 ROBP up to error ε , using seed-length $\tilde{O}(\log(n/\varepsilon))$.
2. After pseudorandomly assigning all but $1/\text{poly} \log(n)$ of the coordinates any width-3 ROBP simplifies enough so that it can be fooled using additional $\tilde{O}(\log(n) \log(1/\varepsilon))$ random bits.

Both steps are involved and explained in greater detail in the next two sections.

2.1 Pseudorandomly assigning half of the coordinates

In Sections 4 and 5 we prove the following theorem showing that we can pseudorandomly assign $1/\text{poly} \log \log(n/\varepsilon)$ of the coordinates while changing the acceptance probability by at most ε .

Theorem 4. *Let $n \in \mathbb{N}, \varepsilon > 0$. There exists a log-space explicit pseudorandom restriction assigning $p = 1/O(\log \log(n/\varepsilon))^6$ fraction of the variables using $O(\log(n/\varepsilon) \log \log(n/\varepsilon))$ random bits, that maintains the acceptance probability of any unordered width-3 length- n ROBP up to error ε .*

Given Theorem 4, we can assign half of the coordinates by iteratively applying the pseudorandom restriction $O(1/p)$ times. This ultimately uses $O(\log(n/\varepsilon)(\log \log(n/\varepsilon))^7) = \tilde{O}(\log(n/\varepsilon))$ random bits to assign half of the coordinates, as promised.

We describe the techniques that go into the proof of Theorem 4. The proof proceeds in two steps. The first step (described in Section 4) reduces the task of generating a pseudorandom restriction for width-3 ROBPs to the task of generating a pseudorandom restriction for the XOR of short (logarithmic-length) width-3 ROBPs. The second step (described in Section 5) is a pseudorandom restriction for the latter class of Boolean functions.

2.1.1 Reducing width-3 ROBPs to the XOR of short width-3 ROBPs

Next, we explain how we reduce fooling width-3 ROBPs to fooling the XOR of short width-3 ROBPs. Let B be a ROBP of length- n and width-3. We pick a set $T_0 \subseteq [n]$ of size $\approx n/2$ using a small-biased distribution (as in [NN93]). We wish to show that for most choices of T_0 , we can pseudorandomly assign pn of the coordinates in T_0 , while fooling the Bias-function $\text{Bias}_{T_0}B$. Our *main observation* is that for most choices for T_0 , the bias-function $\text{Bias}_{T_0}B$ is the average of simpler width-3 ROBPs.

Recall that every layer of edges in a ROBP contains two sets of edges, one corresponding to the transition made when the input bit equals 1 and similarly one corresponding to the input bit equaling -1 . Observe that if the two sets of edges are the same, then the layer is **redundant** and the value of the input bit does not affect whether the ROBP accept or not. We thus assume without loss of generality that there are no redundant layers. We say that a layer of edges is a **colliding layer** if there are two edges marked by the same label (i.e. both labeled 1 or both labeled -1) that enter the same vertex in the next layer.

First, suppose (ideally) that all layers in a width-3 ROBP are colliding. Then, under the pseudorandom restriction, with high probability, in every $O(\log(n/\varepsilon))$ consecutive layers we will have a layer of edges whose corresponding variable is fixed to a value for which the edges in the layer collide, leaving at most 2 vertices reachable in the next layer of vertices. Using a result of Bogdanov et al. [BDVY13] such restricted ROBPs can be written as linear combinations of functions of the following form: XOR of width-3 ROBPs of length $O(\log(n/\varepsilon))$ defined over disjoint sets of variables. It thus suffices to fool these XOR of short width-3 ROBPs in order to fool the restricted ROBP, as we do in Section 5.

The assumption that all layers in a width-3 ROBP are colliding is not necessarily true. In fact, it can be the case that in every layer of edges both the 1-edges and the (-1) -edges form a permutation on the state space with no collisions. Indeed, such ROBPs are known in the literature as permutation-ROBPs. (For example, the $\text{MOD}_3(x_1, \dots, x_n)$ function indicating whether $(\sum_i x_i \equiv 0 \pmod{3})$ can be computed by width-3 permutation ROBP.) Nonetheless, as mentioned earlier, it suffices to fool the bias-function and this task is easier than fooling each restricted function.

Relabeling Under The Bias Function: In the following, we consider relabeling of a ROBP. Recall that in a ROBP every vertex has a pair of outgoing edges: one labeled 1 and the other labeled -1 . A **relabeling** of a ROBP B is any ROBP B' that can be achieved from B by swapping the labels for some of these pairs of edges.

Our *key observation* is that the bias function $\text{Bias}_T B$ of a program B does not depend on the labels of the edges associated with the variables outside T . This is due to the fact that the value of $\text{Bias}_T B(x)$ on a given partial input $x \in \{\pm 1\}^T$ is the probability of acceptance of B on a random assignment to the variables in $[n] \setminus T$, and this value remains the same under any relabeling of the edges associated with the variables in $[n] \setminus T$. Moreover, a simple fact shows that any **non-redundant** layer of edges can be relabeled so that it is **colliding**. Thus, for any ROBP B and any fixed T , we can relabel the edges associated with variables with $[n] \setminus T$ so that they are colliding, yielding another width-3 ROBP, denoted B^T . We get that $\text{Bias}_T B = \text{Bias}_T B^T$, and B^T is a ROBP in which all layers in $[n] \setminus T$ are colliding. We can thus apply the previous argument and conclude that $\text{Bias}_T B^T$ is the average of width-3 ROBPs whose vast majority have a layer of vertices of width-2 in every $O(\log(n/\varepsilon))$ consecutive layers. These ROBPs are then fooled by the pseudorandom partial assignment described in the next section.

To sum up, since the bias-function is the average over all restricted functions of B , it also equals the average over all restricted functions of B^T , and these restricted functions are simple enough for us to fool.

Relabeling was previously used in [BV10, Ste13, CGR14] to show that the best ROBPs distinguishing between certain distributions and the uniform distribution must be “locally-monotone” (see Section 3.5 for the formal definition). In the general case, it is not clear that locally-monotone programs are the hardest to fool. We rather show that under the bias-function, one might as well treat the restricted layers as if they were locally-monotone.

2.1.2 Pseudorandom restrictions for the XOR of short width-3 ROBPs

Our main result in Section 5 is the following:

Theorem 5. *Let $n, w, b \in \mathbb{N}$, $\varepsilon > 0$. There exists a log-space explicit pseudorandom restriction assigning $p = 1/O(\log(b \cdot \log(n/\varepsilon)))^{2w}$ fraction of n variables using $O(w \cdot \log(n/\varepsilon) \cdot (\log \log(n/\varepsilon) + \log(b)))$ random bits, that maintains the acceptance probability of any XOR of ROBPs of width- w and length- b up to error ε .*

Recall that in the previous section, we reduced the case of width-3 ROBPs to this case with $w = 3$ and $b = O(\log(n/\varepsilon))$. Our proof for Theorem 5 follows previous strategies by [GMR⁺12, GY14, GKM15, LV17]. Indeed, the functions we are trying to fool are a special case of **product-functions** that were recently studied in [HLV17, LV17]. Product-functions are functions of the form $f(x) = f_1(x) \cdot f_2(x) \cdots f_m(x)$ where each f_i depends on a set B_i of at most b variables, and $\{B_1, \dots, B_m\}$ are pairwise-disjoint.

PRGs for product-functions were constructed in previous work, however none achieve the parameters we need. Haramaty, Lee and Viola [HLV17] and Lee and Viola [LV17] constructed PRGs with seed length $\tilde{O}(b + \sqrt{mb \log(1/\varepsilon)})$ and $\tilde{O}((b + \log(m/\varepsilon)) \cdot \log(1/\varepsilon))$ respectively for such functions. While the latter is nearly optimal for constant ε , we require ε to be smaller

than $1/m$, since the reduction in the previous section from [BDVY13] incurs a multiplicative factor of m on the error. Gopalan, Meka and Kane [GKM15] achieve nearly optimal seed-length $\tilde{O}(\log(n/\varepsilon))$ but only for the case where the blocks B_1, \dots, B_m are known.

The main reason we are able to achieve better seed-length is due to the fact that we further assume that the functions f_1, \dots, f_m are computed by constant-width ROBPs. We rely on the previous work of Chattopadhyay et al. [CHRT17]. They constructed PRGs for constant-width length- n ROBPs with seed-length $\text{poly} \log(n)$. We observe that under an unusual setting of parameters, namely when applying this result to constant-width ROBPs of length $\text{poly} \log(n)$, one gets seed-length $\tilde{O}(\log(n/\varepsilon))$. This enables us to fool the XOR of any subset of $\text{poly} \log(n)$ of the functions f_1, \dots, f_m using nearly-logarithmic seed-length. Relying on the proof strategy laid by Gopalan and Yehudayoff [GY14], we bootstrap this into a pseudorandom restriction fooling the XOR of f_1, \dots, f_m .

2.2 Simplification under pseudorandom restrictions

Recall that our proof strategy was similar to that of Gopalan et al. [GMR⁺12]:

1. For $i = 0, \dots, O(\log \log n)$: assign half of the remaining coordinates pseudorandomly using $\tilde{O}(\log(n/\varepsilon))$ random bits, while changing the acceptance probability by at most ε .
2. Pseudorandomly assign the remaining coordinates using $\tilde{O}(\log(n/\varepsilon))$ random bits.

The first step was overviewed in the previous section. In order to carry on the second step, we wish to find some progress measure, that would decrease in each iteration of the first step. For the case of CNFs the CNF-width (i.e., the maximal number of literals in a clause) was a good progress measure for [GMR⁺12]. They showed that without loss of generality the CNF-width is $O(\log(n/\varepsilon))$ initially, and that it decreases by a constant-factor in each iteration of Step 1.

Our analogous progress measure is the number of colliding layers. We recall that in a ROBP, some layers of edges form permutations on the state space, while others are colliding.

We show that after the first application of step 1, with high probability the restricted ROBP can be written as a composition of m subprograms D_1, \dots, D_m where each D_i has at most 2 vertices in the first and last layers and at most $\ell_0 = O(\log(n/\varepsilon))$ colliding layers. Intuitively, this happens since every colliding layer reduces the width to 2 with constant probability and thus with high probability in any $O(\log(n/\varepsilon))$ consecutive colliding layers at least one would be set to the value that reduces the width to 2. This motivates the following definition.

Definition 2.1. *We call a ROBP B a (w, ℓ, m) -ROBP if B can be written as $D_1 \circ \dots \circ D_m$, with each D_i being a width w ROBP with the first and last layers having at most two vertices and each D_i having at most ℓ colliding layers.*

We wish to show that the parameter ℓ (that bounds the maximal number of colliding layer in a subprogram D_i with width-2 in the first and last layers) reduces by a constant factor under any iteration of step 1. That is, to show that after iteration i of step 1 we get with high probability a $(3, \ell_i, m_i)$ -ROBP where $\ell_i = \ell_0/c^i$ for some constant $c < 1$. As long as $m_i \leq \exp(O(\ell_i))$, an inductive argument works since the colliding layers in each individual

D_j reduces by a factor c with probability $1 - \exp(-\Omega(\ell_i))$ and we can afford a union bound over all m_i subprograms. However, we cannot afford such a union bound if $m_i \gg \exp(\ell_i)$. To handle this, we prove the following structural result: any $(3, \ell_i, m_i)$ -ROBP can be well-approximated by $(3, \ell_i, C^{\ell_i})$ -ROBPs for some constant C . Furthermore, we show that the error indicator of the approximator can be written as the AND of C^{ℓ_i} many $(3, \ell_i, 1)$ -ROBPs, and that its expectation under the uniform distribution is doubly-exponentially small in ℓ_i . This allows us to show that the error indicator is small under the pseudo-random assignments as well, and we can safely replace a $(3, \ell_i, m_i)$ -ROBP with its $(3, \ell_i, C^{\ell_i})$ -ROBP approximator.

Applying the restriction and the structure result $O(\log \log n)$ times, we end up with a $(3, \ell', C^{\ell'})$ ROBP where $\ell' = O(\log(1/\varepsilon))$. As a last step, we show that $(3, \ell', C^{\ell'})$ -ROBPs are fooled by the INW generator [INW94] with seed-length $\tilde{O}(\log(n) \log(1/\varepsilon))$. This follows from the results of [BRRY14]. For the unordered case, we use the generator from the recent work of [CHHL18] for the last step, with seed-length $\tilde{O}(\log(n) \cdot \text{poly}(1/\varepsilon))$ (using a structural result by [SVW14]).

2.3 The proof of Theorem 3

Theorem 3 is a special case of the following theorem

Theorem 6. *Let $n, w, b \in \mathbb{N}$, $\varepsilon > 0$. There exists a log-space explicit pseudorandom generator that ε -fools any XOR of ROBPs of width- w and length- b (defined on disjoint sets of variables), using seed-length $O(\log(b) + \log \log(n/\varepsilon))^{2w+2} \cdot \log(n/\varepsilon)$.*

We consider b as the progress measure, and wish to show that this parameter reduces under pseudorandom restrictions. This is analogous to the the number of colliding layers ℓ in the previous section. However, here, in some cases, we cannot guarantee that the application of the pseudorandom restriction from Theorem 5 would decrease b . The problematic cases are when we have the XOR of more than $\exp(b)$ functions on b variables each. We show that in such cases, an “aggressive” pseudorandom restriction, assigning $1 - \exp(-b)$ fraction of the variables, simplifies the function significantly, while maintaining its acceptance probability. Combining applications of mild-restrictions and aggressive-restrictions in a “decision tree of random restrictions” results in an assignment that fools the function. However, this does not give a PRG as the decisions made along the tree depend adaptively on the function we try to fool, and PRGs cannot depend on the function they try to fool. We fix this by taking the XOR of several pseudorandom assignments, one per each path in this decision tree in order to construct a PRG that fools this class of functions. (For a longer overview, see Section 6.)

2.4 Organization

In Section 3, we state useful definitions and results from previous work to be used throughout the paper. The rest of paper is organized such that each section starts with an overview that highlights one or two main results proved in it. Section 4 proves the reduction from 3ROBPs to the XOR of short 3ROBPs. In Section 5, we show how to pseudorandomly assign $1/\text{poly} \log \log(n)$ of the input coordinates while preserving the acceptance probability of the XOR of short ROBPs of constant-width (Theorem 5). In Section 6, we construct

pseudorandom generators (assigning all the inputs) for the XOR of short ROBPs of constant-width. As an application, we prove Theorem 3. Then, in Section 7, we prove Theorems 1 and 2. We remark that a reader interested only in the proof of Theorem 3 may skip Sections 4 and 7. Similarly, a reader interested only in the proof of Theorems 1 and 2 may skip Section 6.

3 Preliminaries

Denote by U_n the uniform distribution over $\{\pm 1\}^n$, and by U_S for $S \subseteq [n]$ the uniform distribution over $\{\pm 1\}^S$. Denote by \log the logarithm in base 2. For any function $f : \{\pm 1\}^n \rightarrow \mathbb{R}$, we shorthand by $\mathbf{E}[f] = \mathbf{E}_{x \sim U_n}[f(x)]$ and by $\mathbf{Var}[f] = \mathbf{E}[f^2] - \mathbf{E}[f]^2$. For an event E we denote by $\mathbb{1}_E$ its indicator function.

3.1 Restrictions

For a set $T \subseteq [n]$ and two strings $x \in \{\pm 1\}^T$, $y \in \{\pm 1\}^{[n] \setminus T}$ we denote by $\text{Sel}_T(x, y)$ the string with

$$\text{Sel}_T(x, y)_i = \begin{cases} x_i, & i \in T \\ y_i, & \text{otherwise.} \end{cases}$$

Definition 3.1 (Restriction). *Let $f : \{\pm 1\}^n \rightarrow \mathbb{R}$ be a function. A restriction is a pair (T, y) where $T \subseteq [n]$ and $y \in \{\pm 1\}^{[n] \setminus T}$. We denote by $f_{T|y} : \{\pm 1\}^n \rightarrow \mathbb{R}$ the function f restricted according to (T, y) , defined by $f_{T|y}(x) = f(\text{Sel}_T(x, y))$.*

Definition 3.2 (Random Valued Restriction). *Let $n \in \mathbb{N}$. A random variable (T, y) , distributed over restrictions of $\{\pm 1\}^n$ is called random-valued if conditioned on T , the variable y is uniformly distributed over $\{\pm 1\}^{[n] \setminus T}$.*

Definition 3.3 (p -Random Restriction). *A p -random restriction is a random-valued restriction over pairs (T, y) sampled in the following way: For every $i \in [n]$, independently, pick i to T with probability p ; Sample y uniformly from $\{\pm 1\}^{[n] \setminus T}$. We denote this distribution of restrictions by \mathcal{R}_p .*

Definition 3.4 (The Bias-Function). *Let $f : \{\pm 1\}^n \rightarrow \mathbb{R}$. Let $T \subseteq [n]$. We denote by $\text{Bias}_T(f) : \{\pm 1\}^n \rightarrow \mathbb{R}$ the function defined by $(\text{Bias}_T(f))(x) = \mathbf{E}_{y \sim U_{[n] \setminus T}}[f_{T|y}(x)]$. When T is clear from the context, we shorthand $\text{Bias}_T(f)$ as \tilde{f} .*

3.2 Fourier analysis of Boolean functions

Any function $f : \{\pm 1\}^n \rightarrow \mathbb{R}$ has a unique Fourier representation:

$$f(x) = \sum_{S \subseteq [n]} \hat{f}(S) \cdot \prod_{i \in S} x_i,$$

where the coefficients $\hat{f}(S) \in \mathbb{R}$ are given by $\hat{f}(S) = \mathbf{E}_{x \sim U_n}[f(x) \cdot \prod_{i \in S} x_i]$. We have $\mathbf{Var}[f] = \sum_{\emptyset \neq S \subseteq [n]} \hat{f}(S)^2$. We denote the spectral-norm of f by $L_1(f) \triangleq \sum_{S \subseteq [n]} |\hat{f}(S)|$.

For any functions $f, g : \{\pm 1\}^n \rightarrow \mathbb{R}$ it holds that $L_1(f \cdot g) \leq L_1(f) \cdot L_1(g)$ where equality holds if f and g depends on disjoint sets of variables. Additionally, $L_1(f+g) \leq L_1(f) + L_1(g)$. The following fact relates the Fourier coefficients of a Boolean function and its bias-function.

Fact 3.5 ([O'D14, Proposition 4.17]). *Let $f : \{\pm 1\}^n \rightarrow \mathbb{R}$ and $S, T \subseteq [n]$. Then, $(\widehat{\text{Bias}_T f})(S) = \widehat{f}(S) \cdot \mathbb{1}_{\{S \subseteq T\}}$*

3.3 Small-biased distributions

We say that a distribution \mathcal{D} over $\{\pm 1\}^n$ is δ -biased if for any non-empty $S \subseteq [n]$ it holds that $|\mathbf{E}_{x \sim \mathcal{D}}[\prod_{i \in S} x_i]| \leq \delta$. [NN93, AGHP92, ABN⁺92, BT13, Ta-17] show that δ -biased distributions can be sampled using $O(\log(n/\delta))$ random bits.

Let $p \in (0, 1]$. We say that a distribution \mathcal{D}_p over subsets of $[n]$ is δ -biased with marginals p if for any non-empty $S \subseteq [n]$ it holds that $\Pr_{T \sim \mathcal{D}_p}[S \subseteq T] = p^{|S|} \pm \delta$.

Claim 3.6. *Let $p = 2^{-a}$ for some integer $a > 0$, let \mathcal{D} be an ε -biased distribution over $\{\pm 1\}^{na}$. Define \mathcal{D}_p to be a distribution over subsets of $[n]$ as follows: Sample $x \sim \mathcal{D}$. Output $T = \{i \in [n] : \bigwedge_{j \in [a]} (x_{(i-1)a+j} = 1)\}$. Then \mathcal{D}_p is ε -biased with marginals p .*

Proof. For any fixed S , the probability that $S \subseteq T$ is exactly the probability that $\bigwedge_{i \in S, j \in [a]} (x_{(i-1)a+j} = 1)$. In an ε -biased distribution, the latter event happens with probability $2^{-a \cdot |S|} \pm \varepsilon$ (See [AGHP92]). \square

Claim 3.7. *If \mathcal{D}_p is δ -biased with marginals p , then for any disjoint $S, S' \subseteq [n]$ it holds that $\Pr_{T \sim \mathcal{D}_p}[S \cap T = \emptyset, S' \subseteq T] = (1-p)^{|S|} \cdot p^{|S'|} \pm \delta \cdot 2^{|S|}$.*

Proof. By inclusion-exclusion

$$\begin{aligned} \Pr_{T \sim \mathcal{D}_p}[S \cap T = \emptyset, S' \subseteq T] &= \sum_{R \subseteq S} (-1)^{|R|} \cdot \Pr_{T \sim \mathcal{D}_p}[R \cup S' \subseteq T] \\ &= \sum_{R \subseteq S} (-1)^{|R|} \cdot (\Pr_{T \sim \mathcal{U}}[R \cup S' \subseteq T] \pm \delta) \\ &= \Pr_{T \sim \mathcal{U}}[S \cap T = \emptyset, S' \subseteq T] \pm 2^{|S|} \cdot \delta. \end{aligned} \quad \square$$

3.4 Standard tail bounds for k -wise independence

Lemma 3.8 ([SSS95, Thm. 4, restated]). *Let ℓ be an even positive integer. Let X_1, \dots, X_m be some ℓ -wise independent random variables bounded in $[-1, 1]$ with expectation 0. Let $V = \sum_{i=1}^m \text{Var}[X_i]$. Then, $\mathbf{E}[(X_1 + \dots + X_m)^\ell] \leq \max\{\ell^\ell, (\ell V)^{\ell/2}\}$.*

3.5 Branching programs

A read-once branching program (ROBP) B of length n and width w is a directed layered graph with $n + 1$ layers of vertices denoted V_1, \dots, V_{n+1} . Each V_i consists of $w_i \leq w$ vertices $\{v_{i,1}, \dots, v_{i,w_i}\}$, and between every two consecutive layers V_i and V_{i+1} there exists a set of directed edges (from V_i to V_{i+1}), denoted E_i , such that any vertex in V_i has precisely two

out-going edges in E_i , one marked by 1 and one marked by -1 . The vertices in V_{n+1} are marked with either ‘accept’ and ‘reject’.

A branching program B and an input $x \in \{\pm 1\}^n$ naturally describes a **computation path** in the layered graph: we start at node $v_1 = v_{1,1}$ in V_1 . For $i = 1, \dots, n$, we traverse the edge going out from v_i marked by x_i to get to a node $v_{i+1} \in V_{i+1}$. The resulting computation path is $v_1 \rightarrow v_2 \rightarrow \dots \rightarrow v_{n+1}$. We say that B accepts x iff the computation path defined by B and x reaches an accepting node. Naturally B describes a Boolean function $B : \{\pm 1\}^n \rightarrow \{\pm 1\}$ whose value is -1 on input x iff B accepts x .

Unordered branching programs are defined similarly, expect that there exists a permutation $\pi \in S_n$ such that in step i the computation path follows the edge marked by x_{π_i} , for $i \in [n]$. We also consider unordered branching programs on $[n]$ of shorter length $n' \leq n$. In such case, the program stops after reading n' input bits.³

For two programs B_1 and B_2 defined over disjoint sets of variables and having the end width of B_1 equal the start width of B_2 , we denote by $B_1 \circ B_2$ the concatenation of B_1 and B_2 , defined in the natural way.

Locally Monotone Branching Programs. Let B be a width- w length- n ROBP. For any vertex v in the ROBP, denote by β_v the probability to accept a uniformly random input starting from the vertex v . Since renaming the vertices in each layer does not affect the functionality of B , we may assume without loss of generality that the vertices in V_i are ordered according to β_v . That is, for every $i \in [n+1]$ we have $\beta_{v_{i,1}} \leq \beta_{v_{i,2}} \leq \dots \leq \beta_{v_{i,w_i}}$. In case of equalities, we break ties arbitrarily but commit to a strict ordering of the nodes in each layer. B is called **locally monotone** if for any vertex v in B the vertex reached from v using the 1-edge has larger or equal index than the vertex reached from v using the (-1) -edge.

For $i \in [n]$, denote by $E_{i,1}$ the set of edges in E_i marked by 1 and similarly define $E_{i,-1}$. We say that E_i is a **identity layer** if $E_{i,1} = E_{i,-1}$ (in which case x_i does not affect the output of B). We say that E_i is a **permutation layer** if both $E_{i,1}$ and $E_{i,-1}$ form a matching between V_i and V_{i+1} (i.e., $|V_i| = |V_{i+1}|$ and for $b \in \{-1, 1\}$ no two edges in $E_{i,b}$ enter the same vertex in V_{i+1}). The following is a key lemma from the work of [BV10].

Lemma 3.9 (Collision Lemma [BV10]). *In a locally monotone branching program, every permutation layer is an identity layer.*

To see it, note that if we think of the vertices in each layer $\{v_{i,1}, \dots, v_{i,w_i}\}$ as written from top to bottom according to β_v , then in a locally monotone program for any vertex v the 1-edge leads to the same vertex or to a vertex below the one that follows the (-1) -edge. Thus, assuming both $E_{i,-1}$ and $E_{i,1}$ form a matching, the only way this could happen is if they both form the same matching.

The following is a restatement of a result from [CHRT17]. We give its proof for completeness in Appendix A.1.

Theorem 3.10. *Let B be an unordered oblivious read-once branching programs with width- w and length- n . Let $\varepsilon > 0$, $p \leq 1/O(\log n)^w$, $k = O(\log(n/\varepsilon))$, and \mathcal{D}_p be a δ_T -biased*

³Note that in the unordered case, the set of bits being read could be an arbitrary subset of $[n]$ of size n' .

distribution over subsets of $[n]$ with marginals p , for some $\delta_T \leq p^{2k}$. Then, with probability at least $1 - \varepsilon$ over $T \sim \mathcal{D}_p$,

$$L_1(\tilde{B}) = \sum_{S \subseteq T} |\widehat{B}(S)| \leq O((nw)^3/\varepsilon).$$

Theorem 3.11 (Implied by [CHRT17, Thm. 2] and [SVW14, Thm. 4.1]). *Let \mathcal{C} be the class of all unordered oblivious read-once branching programs on $[n]$ of length at most n' and width at most w . Then, there exists a log-space explicit pseudorandom generator*

$$\text{CHRT} : \{\pm 1\}^{s_{n,n',w,\varepsilon}} \rightarrow \{\pm 1\}^n$$

that ε -fools \mathcal{C} , where $s_{n,n',w,\varepsilon} = O(\log(n')^{w+1} \log \log(n') \log(n/\varepsilon))$.

3.6 Helpful lemmas

Lemma 3.12. *Let $a, b > 0$. If X is a real-valued random variable bounded in $[-b, a]$ with mean 0, then $\mathbf{Var}[X] \leq ab$.*

Proof. $\mathbf{Var}[X] = \mathbf{E}[X^2]$ since $\mathbf{E}[X] = 0$. As x^2 is convex and X domain is bounded, the maximal value that $\mathbf{E}[X^2]$ can get is if all of X 's probability mass is on the boundary. Denote by $p = \mathbf{Pr}[X = a]$. Since $\mathbf{E}[X] = 0$ we get $0 = p \cdot a + (1 - p) \cdot (-b)$, i.e., $p = b/(a + b)$, thus

$$\mathbf{Var}[X] = \mathbf{E}[X^2] \leq a^2p + (1 - p)b^2 = \frac{a^2b}{a + b} + \frac{ab^2}{a + b} = ab. \quad \square$$

Theorem 3.13 (Hyper-contractivity of Variance). *Let $f : \{\pm 1\}^k \rightarrow \{\pm 1\}$ be a Boolean function. Then, $\mathbf{E}_{T \sim \mathcal{R}_p}[\mathbf{Var}[\tilde{f}]] \leq p \cdot \mathbf{Var}[f]$. Furthermore, if $p \leq 1/3$, then $\mathbf{E}_{T \sim \mathcal{R}_p}[\mathbf{Var}[\tilde{f}]] \leq \mathbf{Var}[f]^{3/2}$.*

Proof. First, observe that using Fact 3.5 and $\mathbf{Var}[g] = \sum_{S \neq \emptyset} \widehat{g}(S)^2$, we have

$$\mathbf{E}_{T \sim \mathcal{R}_p}[\mathbf{Var}[\tilde{f}]] = \mathbf{E}_{T \sim \mathcal{R}_p} \left[\sum_{S \neq \emptyset} (\widehat{\text{Bias}_T f})(S)^2 \right] = \sum_{S \neq \emptyset} \widehat{f}(S)^2 \cdot \mathbf{Pr}_{T \sim \mathcal{R}_p}[S \subseteq T] = \sum_{S \neq \emptyset} p^{|S|} \cdot \widehat{f}(S)^2.$$

For the first item, we get $\mathbf{E}_{T \sim \mathcal{R}_p}[\mathbf{Var}[\tilde{f}]] = \sum_{S \neq \emptyset} p^{|S|} \cdot \widehat{f}(S)^2 \leq p \cdot \sum_{S \neq \emptyset} \widehat{f}(S)^2 = p \cdot \mathbf{Var}[f]$.

For the second item, we use the Hyper-contractivity Theorem [Bon70] (cf. [O'D14, Ch. 9]) stating that $\|N_\rho g\|_2 \leq \|g\|_{1+\rho^2}$ for any function $g : \{\pm 1\}^n \rightarrow \mathbb{R}$ (where N_ρ is the noise operator that satisfies $\widehat{N_\rho g}(S) = \rho^{|S|} \cdot \widehat{g}(S)$ for all $S \subseteq [n]$). Take $g = f - \mathbf{E}[f]$ and $\rho = \sqrt{p}$. Then,

$$\mathbf{E}_{T \sim \mathcal{R}_p}[\mathbf{Var}[\tilde{f}]] = \sum_{S \neq \emptyset} p^{|S|} \cdot \widehat{f}(S)^2 = \|N_{\sqrt{p}} g\|_2^2 \leq \|g\|_{1+p}^2 = \mathbf{E}_{x \sim U_k} [|g(x)|^{1+p}]^{2/(1+p)}$$

We analyze the RHS. Let $\beta = \mathbf{E}[f]$. Then, $\beta \in [-1, 1]$, $\mathbf{Var}[f] = 1 - \beta^2$, and under the uniform distribution $|g(x)|$ gets value $|1 - \beta| = 1 - \beta$ with probability $(1 + \beta)/2$ and value $|-1 - \beta| = 1 + \beta$ with probability $(1 - \beta)/2$. We get

$$\mathbf{E}_{x \sim U_k} [|g(x)|^{1+p}] = \frac{1 + \beta}{2} \cdot (1 - \beta)^{1+p} + \frac{1 - \beta}{2} \cdot (1 + \beta)^{1+p}$$

$$= (1 - \beta^2) \cdot \left(\frac{1}{2}(1 - \beta)^p + \frac{1}{2}(1 + \beta)^p\right) \leq 1 - \beta^2 = \mathbf{Var}[f]$$

where the last inequality follows by concavity of $x \mapsto x^p$. Overall if $p \leq 1/3$, then $\mathbf{E}_{T \sim \mathcal{R}_p}[\mathbf{Var}[\tilde{f}]] \leq \mathbf{Var}[f]^{2/(1+p)} \leq \mathbf{Var}[f]^{3/2}$. \square

Lemma 3.14. *Suppose \mathcal{D}_p is δ_T -biased distribution with marginals p . Let $\ell \in \mathbb{N}$. Let $f_1, \dots, f_\ell : \{\pm 1\}^n \rightarrow \mathbb{R}$ be real valued functions, not necessarily distinct. Then,*

$$\left| \mathbf{E}_{T \sim \mathcal{D}_p} \left[\prod_{i=1}^{\ell} \mathbf{Var}[\tilde{f}_i] \right] - \mathbf{E}_{T \sim \mathcal{R}_p} \left[\prod_{i=1}^{\ell} \mathbf{Var}[\tilde{f}_i] \right] \right| \leq \delta_T \cdot \prod_{i=1}^{\ell} \mathbf{Var}[f_i].$$

Proof. Using Fact 3.5, for any fixed T , we have

$$\prod_{i=1}^{\ell} \mathbf{Var}[\tilde{f}_i] = \prod_{i=1}^{\ell} \sum_{S_i \neq \emptyset} \hat{f}_i(S_i)^2 \cdot \mathbb{1}_{\{S_i \subseteq T\}} = \sum_{S_1, \dots, S_\ell \neq \emptyset} \hat{f}_1(S_1)^2 \cdots \hat{f}_\ell(S_\ell)^2 \cdot \mathbb{1}_{\{S_1 \cup \dots \cup S_\ell \subseteq T\}}.$$

Thus,

$$\mathbf{E}_{T \sim \mathcal{D}_p} \left[\prod_{i=1}^{\ell} \mathbf{Var}[\tilde{f}_i] \right] = \sum_{S_1, \dots, S_\ell \neq \emptyset} \hat{f}_1(S_1)^2 \cdots \hat{f}_\ell(S_\ell)^2 \cdot (p^{|S_1 \cup \dots \cup S_\ell|} \pm \delta_T)$$

and

$$\mathbf{E}_{T \sim \mathcal{R}_p} \left[\prod_{i=1}^{\ell} \mathbf{Var}[\tilde{f}_i] \right] = \sum_{S_1, \dots, S_\ell \neq \emptyset} \hat{f}_1(S_1)^2 \cdots \hat{f}_\ell(S_\ell)^2 \cdot p^{|S_1 \cup \dots \cup S_\ell|}.$$

The difference between the two is at most

$$\left| \mathbf{E}_{T \sim \mathcal{D}_p} \left[\prod_{i=1}^{\ell} \mathbf{Var}[\tilde{f}_i] \right] - \mathbf{E}_{T \sim \mathcal{R}_p} \left[\prod_{i=1}^{\ell} \mathbf{Var}[\tilde{f}_i] \right] \right| \leq \delta_T \cdot \sum_{S_1, \dots, S_\ell \neq \emptyset} \hat{f}_1(S_1)^2 \cdots \hat{f}_\ell(S_\ell)^2 = \delta_T \cdot \prod_{i=1}^{\ell} \mathbf{Var}[f_i],$$

which completes the proof. \square

4 From width-3 ROBPs to the XOR of short ROBPs

In Section 5, we prove the following theorem.

Theorem 5. *Let $n, w, b \in \mathbb{N}$, $\varepsilon > 0$. There exists a log-space explicit pseudorandom restriction assigning $p = 1/O(\log(b \cdot \log(n/\varepsilon)))^{2w}$ fraction of n variables using $O(w \cdot \log(n/\varepsilon) \cdot (\log \log(n/\varepsilon) + \log(b)))$ random bits, that maintains the acceptance probability of any XOR of ROBPs of width- w and length- b up to error ε .*

The pseudorandom restriction assigns p fraction of the variables as follows:

1. Choose a set of coordinates $T \subseteq [n]$ according to a δ_T -biased distribution with marginals p , for $\delta_T := p^{O(\log(n/\varepsilon))}$.

2. Assign the variables in T according to a δ_x -biased distribution, for $\delta_x := (\varepsilon/n)^{O(\log b)}$.

Known constructions of small-biased distributions [NN93, AGHP92, ABN⁺92, BT13, Ta-17] show that it suffices to use $O(\log(n/\delta_T) + \log(n/\delta_x)) \leq O(w \cdot \log(n/\varepsilon) \cdot (\log \log(n/\varepsilon) + \log(b)))$ random bits to sample the restriction.

In this section, we show how to design pseudorandom restrictions for unordered width-3 ROBPs from pseudorandom restrictions to the XOR of many width-3 ROBPs of length $O(\log(n/\varepsilon))$. We get the following theorem.

Theorem 4. *Let $n \in \mathbb{N}, \varepsilon > 0$. There exists a log-space explicit pseudorandom restriction assigning $p = 1/O(\log \log(n/\varepsilon))^6$ fraction of the variables using $O(\log(n/\varepsilon) \log \log(n/\varepsilon))$ random bits, that maintains the acceptance probability of any unordered width-3 length- n ROBP up to error ε .*

Proof Sketch. In this section, we shall show that under pseudorandom restrictions leaving each variable alive with probability $1/2$, with high probability, the bias function of a ROBP B can be written as a linear combination (up to a small error) over functions of the form $f_1 \cdot f_2 \cdot \dots \cdot f_m$ where each f_i is a short subprogram of the original program of length $O(\log(n/\varepsilon))$, and each f_i is defined on a disjoint set of coordinates. Each function g in the linear combination will have a weight $\alpha_g \in [-1, 1]$, and the sum of absolute values of weights over all functions participating in the linear combination will be at most n . This will show that any generator that ε/n -fools the XOR of short width-3 ROBPs also ε -fools width-3 length- n ROBPs under random restrictions.

The reduction will first establish that with high probability (over the choice of the set of coordinates that are left alive) the bias function of a ROBP B can be written as the average of width-3 length- n ROBPs, whose vast majority have at most $O(\log(n/\varepsilon))$ layers between every two layers with width-2. Then, we use a result of Bogdanov et al. [BDVY13] that reduces branching programs with many width-2 layers to the XOR of short ROBPs.

We focus on the first part of the reduction. First, consider the case when B is locally-monotone. In this case, every layer of edges is either the identity layer or a colliding layer (Lemma 3.9). Assume without loss of generality that there are no identity layers. Then, under a pseudorandom restriction, with high probability, in every $O(\log(n/\varepsilon))$ consecutive layers we will have a layer of edges whose corresponding variable is fixed to the value on which the edges in the layer collide, leaving at most 2 vertices reachable in the next layer of vertices. Removing unreachable vertices, we get that with high probability under the random restriction, in every $O(\log(n/\varepsilon))$ consecutive layers there is a layer of vertices with width-2.

However, in the case that B is not locally-monotone (e.g., when B is a permutation ROBP) it could be the case that the widths of all layers of vertices remain 3 under the random restriction. Our main observation is that since the bias function takes the average over all assignments to the restricted variables, the bias function of B does not depend on the labels of edges marked by the restricted variables. More formally, for any $T \subseteq [n]$, if B and C are two ROBPs with the same graph structure that only differ on the labels on the edges in layers $[n] \setminus T$, then $\text{Bias}_T(B) = \text{Bias}_T(C)$. Thus, once T is fixed we may relabel the layers in $[n] \setminus T$ so that they are locally-monotone, yielding a new ROBP B' , and then apply the

bias function. Using the analysis of the locally monotone case, we get that the bias function of B' (and thus the bias function of B) is the average of B' over all restrictions fixing the coordinates in $[n] \setminus T$, and we know that most of these restricted ROBPs have width-2 in every $O(\log(n/\varepsilon))$ consecutive layers.

Essentially, the bias function allows us to imagine as if we are taking the average over restrictions of B' rather than restrictions of B , and restrictions of B' are “simpler” to fool than restrictions of B since they have many layers with width-2.

The formal argument follows.

Theorem 4.1 (From width-3 to almost width-2). *Let B be a ROBP of width-3 and length- n . Let $\varepsilon > 0$. Let $\mathcal{D}_{1/2}$ be a $(\varepsilon/n)^{10}$ -biased distribution over subsets of $[n]$ with marginals $1/2$. Let $T \sim \mathcal{D}_{1/2}$ be a random variable. Let B^T be the branching program B where the layers in $[n] \setminus T$ are relabeled so that they are locally monotone. Then,*

$$\text{Bias}_T(B)(x) = \text{Bias}_T(B^T)(x) = \mathbf{E}_{y \sim U_{[n] \setminus T}} [(B_{T|y}^T(x))]$$

and with probability at least $1 - \varepsilon$ over the choice of T and y , $B_{T|y}^T$ can be computed by a ROBP of the form $D_1 \circ \dots \circ D_m$ where $\{D_i\}_{i=1}^m$ are defined over disjoint sets of at most $b = (3 \log(n/\varepsilon))$ variables, and each D_i is a width-3 ROBP with at most 2 vertices on the first and last layers.

Proof. We first observe that $\text{Bias}_T(B)(x) = \text{Bias}_T(B^T)(x)$. Indeed, for any fixed x , $\text{Bias}_T(B)(x)$ equals the probability that the following random-path in B accepts:

Initiate v_1 to be the start node of B . For $i = 1, \dots, n$ if $i \in T$, take the edge exiting v_i marked by x_i , otherwise (i.e., if $i \in [n] \setminus T$) pick a random edge out of the two edges exiting v_i . Denote by v_{i+1} the node at the end of the edge taken in the i -th step. Accept if and only if v_{n+1} is an accepting node.

Observe that the following random process is oblivious to the labels of edges in layers $[n] \setminus T$, thus it would yield the same probability for $\text{Bias}_T(B)$ and for $\text{Bias}_T(B^T)$. Overall, we got that $\text{Bias}_T(B)$ and $\text{Bias}_T(B^T)$ are equal as functions.

In the remainder of the proof, we analyze $\text{Bias}_T(B^T)$. Let $E_{i,1}$ and $E_{i,-1}$ denote the set of edges in the i -layer of B marked by 1 and -1 respectively. We assume without loss of generality that in all layers of edges $E_{i,1} \neq E_{i,-1}$, as otherwise the i -th layer is redundant and may be eliminated. (Observe that under any relabeling of B this property is preserved.) By the collision lemma of Brody-Verbin [BV10] (Lemma 3.9), for any $i \in [n] \setminus T$, layer i in B^T has the following property: either $E_{i,1}$ or $E_{i,-1}$ has at most 2 end-vertices.

Next, we consider the program $B_{T|y}^T$ for a pseudorandom T and a random $y \in \{\pm 1\}^{[n] \setminus T}$. For $i = 1, \dots, n$ we say that the i -th layer of edges is “good” under the choice of T and y , if $i \in [n] \setminus T$ and layer E_{i,y_i} of B^T has at most 2 end-vertices. Let $b = 3 \log(n/\varepsilon)$. For $i = 1, \dots, n - b + 1$ let \mathcal{E}_i be the event that none of layers $\{i, i + 1, \dots, i + (b - 1)\}$ is good. Since T is sampled from a $(\varepsilon/n)^{10}$ -biased distribution with marginals $1/2$, we have that T is $(\varepsilon/2n)$ -almost b -wise independent. Thus, up to an error of $\varepsilon/2n$ we may analyze the event \mathcal{E}_i under uniform choice of a subset $T \subseteq [n]$. Indeed, under a uniform choice for T and y each

layer is good with probability at least $1/4$, and all b layers are not good with probability at most $(3/4)^b$. Overall, we get $\Pr[\mathcal{E}_i] \leq (3/4)^b + (\varepsilon/2n) \leq \varepsilon/n$. By the union bound,

$$\Pr[\mathcal{E}_1 \vee \mathcal{E}_2 \vee \dots \vee \mathcal{E}_{n-b+1}] \leq (n-b+1) \cdot (\varepsilon/n) \leq \varepsilon.$$

Under the event that all \mathcal{E}_i are false, we get that $B_{T|y}^T$ has width 2 in every b layers. In such a case, we may write the restricted function $B_{T|y}^T$ as $D_1 \circ \dots \circ D_m$ where each D_i is a width-3 and length at most b ROBP with at most 2 vertices on the first and last layer. \square

Theorem 4.2 (from almost width-2 to the XOR of short ROBPs - restatement of [BDVY13, Thm. 2.1]). *Let B be a ROBP of the form $D_1 \circ \dots \circ D_m$ where $\{D_i\}_{i=1}^m$ are defined over disjoint sets of variables, and each D_i is a width-3 ROBP with at most 2 vertices on the first and last layers. Then, (as a real-valued function) B can be written as a linear combination of $\sum_{\alpha \in \{0,1\}^m} c_\alpha \cdot \prod_{i=1}^m D_{i,\alpha_i}$ where $D_{i,0}, D_{i,1}$ are subprograms of D_i and $\sum_{\alpha \in \{0,1\}^m} |c_\alpha| \leq m$.*

Proof of Theorem 4. We prove that the following pseudorandom restriction maintains the acceptance probability of ROBPs of width-3 and length- n up to error ε . Let $\varepsilon_1 := \varepsilon/2$, $\varepsilon_2 := \varepsilon/2n$.

1. Pick $T_0 \subseteq [n]$ using a $(\varepsilon_1/n)^{10}$ -biased distribution with marginals $1/2$.
2. (a) Pick $T \subseteq T_0$ using a δ_T -biased distribution with marginals $p = 1/O(\log \log(n/\varepsilon_2))^6$.
(b) Assign the coordinates in T using a $(\varepsilon_2/n)^{O(\log \log(n/\varepsilon_2))}$ -biased distribution \mathcal{D}_x .

Equivalently, we prove that the following distribution ε -fools ROBPs of width-3 and length- n .

1. Pick $T_0 \subseteq [n]$ using a $(\varepsilon_1/n)^{10}$ -biased distribution with marginals $1/2$.
2. Assign the coordinates in $[n] \setminus T_0$ uniformly at random.
3. (a) Pick $T \subseteq T_0$ using a δ_T -biased distribution with marginals $p = 1/O(\log \log(n/\varepsilon_2))^6$.
(b) Assign the coordinates in $T_0 \setminus T$ uniformly at random.
(c) Assign the coordinates in T using a $(\varepsilon_2/n)^{O(\log \log(n/\varepsilon_2))}$ -biased distribution \mathcal{D}_x .

Let $y \sim U_{[n] \setminus T_0}$. Let \mathcal{G} be the event that $B_{T_0|y}^{T_0}$ can be computed by a ROBP of the form $D_1 \circ \dots \circ D_m$ where $\{D_i\}_{i=1}^m$ are defined over disjoint sets of at most $b = 3 \log(n/\varepsilon_1)$ variables, and each D_i is a width-3 ROBP with at most 2 vertices on the first and last layers. By Theorem 4.1 $\Pr(\mathcal{G}) \geq 1 - \varepsilon_1$. Assuming that \mathcal{G} happened, then by Theorem 4.2, $B_{T_0|y}^{T_0}$ can be written as $\sum_{\alpha \in \{0,1\}^m} c_\alpha \cdot \prod_{i=1}^m D_{i,\alpha_i}$ where D_{i,α_i} are subprograms of D_i and $\sum_{\alpha \in \{0,1\}^m} |c_\alpha| \leq m$. For each $\alpha \in \{0,1\}^m$, using Theorem 5 we have that

$$\left| \mathbf{E}_{z \sim U_{T_0}} \left[\prod_{i=1}^m D_{i,\alpha_i}(z) \right] - \mathbf{E}_T \mathbf{E}_{x \sim \mathcal{D}_x} \mathbf{E}_{y' \sim U_{T_0 \setminus T}} \left[\prod_{i=1}^m D_{i,\alpha_i}(\text{Sel}_T(x, y')) \right] \right| \leq \varepsilon_2.$$

By linearity of expectation and the triangle inequality

$$\left| \mathbf{E}_{z \sim U_{T_0}} \left[\sum_{\alpha} c_\alpha \cdot \prod_{i=1}^m D_{i,\alpha_i}(z) \right] - \mathbf{E}_T \mathbf{E}_{x \sim \mathcal{D}_x} \mathbf{E}_{y' \sim U_{T_0 \setminus T}} \left[\sum_{\alpha} c_\alpha \cdot \prod_{i=1}^m D_{i,\alpha_i}(\text{Sel}_T(x, y')) \right] \right|$$

$$\leq \sum_{\alpha} |c_{\alpha}| \cdot \varepsilon_2 \leq m \cdot \varepsilon_2 \leq \varepsilon/2$$

Overall, we get

$$\begin{aligned} & \left| \mathbf{E}_{z \sim U_n} [B(z)] - \mathbf{E}_{\substack{T_0, \\ y \in U_{\overline{T_0}}}} \mathbf{E}_{\substack{T, x \sim \mathcal{D}_x \\ y' \sim U_{T_0 \setminus T}}} [B(\text{Sel}_{T_0}(\text{Sel}_T(x, y'), y))] \right| = \\ & \left| \mathbf{E}_{z \sim U_n} [B(z)] - \mathbf{E}_{\substack{T_0, \\ y \in U_{[n] \setminus T_0}}} \mathbf{E}_{\substack{T, x \sim \mathcal{D}_x \\ y' \sim U_{T_0 \setminus T}}} [B^{T_0}(\text{Sel}_{T_0}(\text{Sel}_T(x, y'), y))] \right| = \\ & \left| \mathbf{E}_{\substack{T_0, \\ y \in U_{[n] \setminus T_0}}} \mathbf{E}_{\substack{T, z \sim U_T \\ y' \sim U_{T_0 \setminus T}}} [B^{T_0}(\text{Sel}_{T_0}(\text{Sel}_T(z, y'), y))] - \mathbf{E}_{\substack{T_0, \\ y \in U_{[n] \setminus T_0}}} \mathbf{E}_{\substack{T, x \sim \mathcal{D}_x \\ y' \sim U_{T_0 \setminus T}}} [B^{T_0}(\text{Sel}_{T_0}(\text{Sel}_T(x, y'), y))] \right| \end{aligned} \quad (1)$$

where the last equality is due to the fact for any T, T_0 the distribution of $\text{Sel}_{T_0}(\text{Sel}_T(z, y'), y)$ is the uniform distribution over $\{\pm 1\}^n$. We bound Expression (1) by

$$\begin{aligned} & \mathbf{E}_{T_0, y \in U_{[n] \setminus T_0}} \left[\left| \mathbf{E}_T \mathbf{E}_{y' \sim U_{T_0 \setminus T}} \left(\mathbf{E}_{z \sim U_T} [B_{T_0|y}^{T_0}(\text{Sel}_T(z, y'))] - \mathbf{E}_{x \sim \mathcal{D}_x} [B_{T_0|y}^{T_0}(\text{Sel}_T(x, y'))] \right) \right| \right] \\ & \leq \Pr[-\mathcal{G}] + \mathbf{E}_{T_0, y \in U_{[n] \setminus T_0}} \left[\left| \mathbf{E}_{T, y' \sim U_{T_0 \setminus T}} \left(\mathbf{E}_{z \sim U_T} [B_{T_0|y}^{T_0}(\text{Sel}_T(z, y'))] - \mathbf{E}_{x \sim \mathcal{D}_x} [B_{T_0|y}^{T_0}(\text{Sel}_T(x, y'))] \right) \right| \middle| \mathcal{G} \right] \\ & \leq \varepsilon/2 + \varepsilon/2 \end{aligned}$$

where the second summand is bounded by $\varepsilon/2$ according to the above discussion using Theorem 4.2 and Theorem 5. \square

5 Pseudorandom restrictions for the XOR of short ROBPs

In this section, we prove Theorem 5. Let B_1, \dots, B_m be pairwise disjoint subsets of $[n]$, each of size at most b . For $i = 1, \dots, m$ let $f_i : \{\pm 1\}^{B_i} \rightarrow \{\pm 1\}$ be a width w ROBP. We construct a pseudorandom generator that ε -fools $f = \prod_{i=1}^m f_i$. We recall the statement of Theorem 5 and the construction.

Theorem 5. *Let $n, w, b \in \mathbb{N}$, $\varepsilon > 0$. There exists a log-space explicit pseudorandom restriction assigning $p = 1/O(\log(b \cdot \log(n/\varepsilon)))^{2w}$ fraction of n variables using $O(w \cdot \log(n/\varepsilon) \cdot (\log \log(n/\varepsilon) + \log(b)))$ random bits, that maintains the acceptance probability of any XOR of ROBPs of width- w and length- b up to error ε .*

Recall that the pseudorandom restriction assigns p fraction of the variables as follows:

1. Choose a set of coordinates $T \subseteq [n]$ according to a δ_T -biased distribution with marginals p , for $\delta_T := p^{O(\log(n/\varepsilon))}$.
2. Assign the variables in T according to a δ_x -biased distribution, for $\delta_x := (\varepsilon/n)^{O(\log b)}$.

Analysis. We shall assume without loss of generality that for all $i = 1, \dots, m$ it holds that $\mathbf{E}[f_i] \geq 0$. We shall also assume without loss of generality that for all $i = 1, \dots, m$ it holds that $\mathbf{Var}[f_i] > 0$ (i.e., that the functions are non-constant). Since the functions f_i are Boolean and depend on at most b bits, we have $\mathbf{Var}[f_i] = \mathbf{Pr}[f_i = 1] \cdot \mathbf{Pr}[f_i = -1] \geq 2^{-b} \cdot (1 - 2^{-b}) \geq 2^{-1-b}$.

We partition the functions into $O(\log b)$ buckets according to their variance. Let $\sigma_0 = 1$, for every $j \in \{1, \dots, \log_{1.1}(b+1)\}$, let $\sigma_j = 2^{-1.1^j}$ and $I_j = \{i \in [m] : \mathbf{Var}[f_i] \in (\sigma_j, \sigma_{j-1}]\}$. Let $C > 0$ be a sufficiently large constant. We consider two cases in our analysis:

Low-Variance Case: For every $j \in \{1, \dots, \log_{1.1}(b+1)\}$ we have

$$\sum_{i \in I_j} \mathbf{Var}[f_i] \leq C \cdot \log^2(n/\varepsilon) / (\sigma_{j-1})^{0.1} .$$

High-Variance Case: There exists a $j \in \{1, \dots, \log_{1.1}(b+1)\}$ with

$$\sum_{i \in I_j} \mathbf{Var}[f_i] > C \cdot \log^2(n/\varepsilon) / (\sigma_{j-1})^{0.1} .$$

Setting Up Parameters: Let $C' > 1$ be a sufficiently large constant. Set

$$\delta_T \triangleq p^{2C' \cdot \log(n/\varepsilon)}, \quad (2)$$

$$\delta \triangleq (\varepsilon/n)^{10C'}, \quad (3)$$

$$\delta'_x \triangleq (\varepsilon/n)^{100C'}, \quad (4)$$

$$\delta_x \triangleq (\delta'_x)^{\log_{1.1}(b+1)}. \quad (5)$$

5.1 Low-Variance Case

For $j = 1, \dots, \log_{1.1}(b+1)$, let $F_j(x) = \prod_{i \in I_j} f_j(x)$. Thus, $f = \prod_j F_j$. Let \mathcal{D}_p be any δ_T -biased distribution with marginals p . For $j \in \{1, \dots, \log_{1.1}(b+1)\}$, we shall show that with probability at least $1 - \varepsilon/2n$ over the choice of $T \sim \mathcal{D}_p$, it holds that

$$\left| \mathbf{E}_{x \sim \mathcal{D}'_x} [\widetilde{F}_j(x)] - \mathbf{E}_{z \sim U_T} [\widetilde{F}_j(z)] \right| \leq \varepsilon/n^{40},^4 \quad (6)$$

for any δ'_x -biased distribution \mathcal{D}'_x over $\{\pm 1\}^n$. Thus, by union bound Eq. (6) holds for all $j \in \{1, \dots, \log_{1.1}(b+1)\}$ simultaneously with probability at least $1 - \varepsilon/2$ over $T \sim \mathcal{D}_p$. Using the following XOR lemma for small-biased distributions from [GMR⁺12] we get that any $(\delta'_x)^{\log_{1.1}(b+1)}$ -biased distribution, fools $\widetilde{f}(x) = \prod_{j=1}^{\log_{1.1}(b+1)} \widetilde{F}_j(x)$ with error at most $16^{\log_{1.1}(b+1)} \cdot 2(\varepsilon/n^{40}) \leq \varepsilon/2$ (using $b \leq n$).

Lemma 5.1 ([GMR⁺12, Thm. 4.1], restated). *Let $0 < \varepsilon < \delta \leq 1$. Let $F_1, \dots, F_k : \{\pm 1\}^n \rightarrow [-1, 1]$ be functions on disjoint input variables such that each F_i is δ -fooled by any ε -biased distribution. Let $H : [-1, 1]^k \rightarrow [-1, 1]$ be a multilinear function in its inputs. Then $H(F_1(x), \dots, F_k(x))$ is $(16^k \cdot 2\delta)$ -fooled by any ε^k -biased distribution.*

⁴recall that we denote by $\widetilde{g} = \text{Bias}_T(g)$ for any function g .

In Appendix A.2, we show how to derive Lemma 5.1 from [GMR⁺12, Thm. 4.1].

In the remainder of this section, we focus on fooling a single F_j , that is, fooling the product (i.e., XOR) of functions $\{f_i\}_{i \in I_j}$ for which $\mathbf{Var}[f_i] \in (\sigma_j, \sigma_{j-1}]$. We note that since we are in the “Low-Variance Case”, then

$$|I_j| \leq C \cdot \sigma_j^{-1} \cdot \sigma_{j-1}^{-0.1} \cdot \log^2(n/\varepsilon). \quad (7)$$

We handle two cases depending on whether σ_{j-1} is big or not.

The case of $\sigma_{j-1} \geq 1/(C \cdot \log(n/\varepsilon))^{20}$: In this case there are at most $O(\sigma_{j-1}^{-1.2} \cdot \log^2(n/\varepsilon)) \leq \text{poly log}(n/\varepsilon)$ functions in I_j , each computed by a width- w ROBP on at most b bits. Thus, $F_j := \prod_{i \in I_j} f_i$ can be computed by a ROBP of length at most $n' = b \cdot \text{poly log}(n/\varepsilon)$ and width at most $2w$. Using Theorem 3.10 on F_j (which has length n' and width $2w$), with probability at least $1 - \delta$ the spectral-norm of \widetilde{F}_j is at most $O((n'w)^3/\delta)$, thus any δ'_x -biased distribution $O(\delta'_x \cdot (n'w)^3/\delta)$ -fools $\widetilde{F}_j = \prod_{i \in I_j} \widetilde{f}_i(x)$. For a large enough choice for C' , $O(\delta'_x \cdot (n'w)^3/\delta) \leq \varepsilon/n^{40}$ and we are done.

The case of $\sigma_{j-1} < 1/(C \cdot \log(n/\varepsilon))^{20}$: In this case all variances in I_j are certainly smaller than 0.5, and hence for all $i \in I_j$, we have $\mathbf{E}[f_i]^2 = \mathbf{E}[f_i^2] - \mathbf{Var}[f_i] = 1 - \mathbf{Var}[f_i] \in [0.5, 1]$. Let

$$\mu_i = \mathbf{E}[f_i] \quad \text{and} \quad g_i(x) \triangleq \frac{f_i(x)}{\mu_i} - 1.$$

Then,

$$\prod_i f_i(x) = \prod_i \mu_i \cdot (1 + g_i(x)).$$

We have $\mathbf{E}[g_i] = 0$ and $\mathbf{Var}[g_i] = \mathbf{Var}[f_i]/\mu_i^2 \in [\mathbf{Var}[f_i], \mathbf{Var}[f_i] \cdot 2]$. We will show that with high probability over T , any δ'_x -biased distribution fools $\prod_i \mu_i \cdot \prod_i (1 + \widetilde{g}_i(x))$.

For ease of notation, in this case we think of I_j as $[m]$ and denote by $\sigma = \sigma_{j-1}$. The proof strategy for this part follows the work of Gopalan and Yehudayoff [GY14]. We note that

$$\prod_{i=1}^m (1 + \widetilde{g}_i(x)) = 1 + \sum_{k=1}^m S_k(\widetilde{g}_1(x), \widetilde{g}_2(x), \dots, \widetilde{g}_m(x)),$$

where S_k is the k -symmetric polynomial given by $S_k(y_1, \dots, y_m) = \prod_{R \subseteq [m], |R|=k} \prod_{i \in R} y_i$. We show that x and T fool the low-degree symmetric polynomials. Then, the following theorem by Gopalan and Yehudayoff [GY14] bootstraps this to show that x and T also fool the sum of all high-degree symmetric polynomials.

Theorem 5.2 (Gopalan-Yehudayoff Tail Inequalities [GY14]). *Let $y_1, \dots, y_m \in \mathbb{R}$. Suppose $|S_\ell(y_1, \dots, y_m)| \leq \frac{t^\ell}{\sqrt{\ell!}}$ and $|S_{\ell+1}(y_1, \dots, y_m)| \leq \frac{t^{\ell+1}}{\sqrt{(\ell+1)!}}$ for some t and ℓ . Then, for every $k \in \{\ell, \dots, m\}$ it holds that $|S_k(y_1, \dots, y_m)| \leq (6et)^k \cdot (\ell/k)^{k/2}$. Furthermore, if $6et \leq 1/2$, then*

$$\sum_{k=\ell}^m |S_k(y_1, \dots, y_m)| \leq 2 \cdot (6et)^\ell.$$

Analyzing the Symmetric Polynomials

From Eq. (7) and our assumption that $\sigma < 1/(C \cdot \log(n/\varepsilon))^{20}$ we get that $m \leq \sigma^{-1.3}$. Recall that C' is a sufficiently large constant and recall the definition of $\delta, \delta'_x, \delta_T$ from Eqs. (2), (3) and (4). We set

$$\ell \triangleq C' \cdot \log(n/\varepsilon) / \log(1/\sigma) \quad (8)$$

In the following, we shall use the facts that $\sigma^{-\ell}, m^\ell \ll 1/\delta$ and $\delta'_x \ll \delta$.

Claim 5.3. *Let $T \sim \mathcal{D}_p$. Let $R \subseteq [m]$ be a set of size at most ℓ . Then, with probability at least $1 - O(b\ell w)^3 \cdot \delta$ over the choice of T , $\prod_{i \in R} \tilde{f}_i(x)$ has spectral-norm at most $1/\delta$.*

Proof. Note that $\prod_{i \in R} f_i(x)$ can be computed by a ROBP with length $b \cdot \ell \leq O(b \cdot \log(n/\varepsilon))$ and width $2w$ (as in the case where σ_j is big). Apply Theorem 3.10 to $\prod_{i \in R} f_i(x)$. \square

We say that $T \subseteq [n]$ is a **good** set if for all sets $R \subseteq [m]$ of size at most ℓ , the spectral-norm of $\prod_{i \in R} \tilde{f}_i$ is at most $1/\delta$. We observe that by Claim 5.3, the probability that T is good is at least $1 - (m+1)^\ell \cdot O(b\ell w)^3 \cdot \delta \geq 1 - \varepsilon/10n$ (using Eq. (7) and (3)).

Claim 5.4. *If T is good, then for any $k \leq \ell + 1$, $S_k(\tilde{g}_1, \tilde{g}_2, \dots, \tilde{g}_m)$ has spectral-norm at most $\delta^{-1} \cdot (4m)^k \leq \delta^{-2}$.*

Proof. We expand the k -symmetric polynomial: $S_k(\tilde{g}_1(x), \dots, \tilde{g}_m(x)) = \sum_{R \subseteq [m], |R|=k} \prod_{r \in R} \tilde{g}_r(x)$. Since T is good, each summand has spectral-norm

$$L_1\left(\prod_{r \in R} \tilde{g}_r(x)\right) = L_1\left(\prod_{r \in R} \left(\frac{\tilde{f}_r(x)}{\mathbf{E}[f_r]} - 1\right)\right) \leq L_1\left(\sum_{Q \subseteq R} (-1)^{|R|-|Q|} \prod_{r \in Q} \frac{\tilde{f}_r(x)}{\mathbf{E}[f_r]}\right) \leq 2^k \cdot \delta^{-1} \cdot 2^k,$$

(using $\mathbf{E}[f_r] \geq 1/2$). Summing over all $\binom{m}{k} \leq m^k$ summands completes the proof. \square

We wish to show that with high probability the total variance under restrictions $\sum_i \mathbf{Var}[\tilde{f}_i]$ is small. Towards this goal, we prove a bound on the ℓ -th moment of the total variance.

Claim 5.5. $\mathbf{E}_{T \sim \mathcal{D}_p}[(\sum_{i=1}^m \mathbf{Var}[\tilde{f}_i])^\ell] \leq 2 \cdot (2\sigma^{0.2})^\ell$

Proof. Fix $(i_1, \dots, i_\ell) \in [m]^\ell$, not necessarily distinct indices. By Lemma 3.14

$$\mathbf{E}_{T \sim \mathcal{D}_p} \left[\prod_{j=1}^{\ell} \mathbf{Var}[\tilde{f}_{i_j}] \right] \leq \mathbf{E}_{T \sim \mathcal{R}_p} \left[\prod_{j=1}^{\ell} \mathbf{Var}[\tilde{f}_{i_j}] \right] + \delta_T \cdot \prod_{j=1}^{\ell} \mathbf{Var}[f_{i_j}],$$

from which we deduce

$$\mathbf{E}_{T \sim \mathcal{D}_p} \left[\left(\sum_{i=1}^m \mathbf{Var}[\tilde{f}_i(z)] \right)^\ell \right] \leq \mathbf{E}_{T \sim \mathcal{R}_p} \left[\left(\sum_{i=1}^m \mathbf{Var}[\tilde{f}_i(z)] \right)^\ell \right] + \delta_T \cdot m^\ell \sigma^\ell.$$

We are left to bound $\mathbf{E}_{T \sim \mathcal{R}_p}[(\sum_{i=1}^m \mathbf{Var}[\tilde{f}_i])^\ell]$. By Fact 3.5, for any $i \in [m]$, the random variable $X_i = \mathbf{Var}[\tilde{f}_i]/\mathbf{Var}[f_i]$ (whose value depends on the choice of $T \sim \mathcal{R}_p$) is bounded in

$[0, 1]$. By Theorem 3.13, its expected value is at most $\mathbf{Var}[f_i]^{0.5} \leq \sigma^{0.5}$. Taking $X = \sum_{i=1}^m X_i$, we get that X is the sum of m independent random variables bounded in $[0, 1]$. Using $m \leq \sigma^{-1.3}$, we have that $\mathbf{E}[X] \leq \sigma^{0.5} \cdot m \leq \sigma^{-0.8}$. Thus, by Chernoff's bounds, with probability at least $1 - \exp(-\Omega(\sigma^{-0.8}))$ we have $X \leq 2 \cdot \sigma^{-0.8}$. In such a case $\sum_i \mathbf{Var}[\tilde{f}_i] \leq 2 \cdot \sigma^{-0.8} \cdot \sigma \leq 2\sigma^{0.2}$. We get $\mathbf{E}_{T \sim \mathcal{R}_p}[(\sum_{i=1}^m \mathbf{Var}[\tilde{f}_i])^\ell] \leq \exp(-\Omega(\sigma^{-0.8})) \cdot (\sigma m)^\ell + (2\sigma^{0.2})^\ell$, which gives

$$\mathbf{E}_{T \sim \mathcal{D}_p} \left[\left(\sum_{i=1}^m \mathbf{Var}[\tilde{f}_i] \right)^\ell \right] \leq \delta_T \cdot m^\ell \sigma^\ell + \exp(-\Omega(\sigma^{-0.8})) \cdot (\sigma m)^\ell + (2\sigma^{0.2})^\ell \leq 2 \cdot (2\sigma^{0.2})^\ell. \quad \square$$

We say that a set $T \subseteq [n]$ is **excellent** if T is good and $\sum_i \mathbf{Var}[\tilde{g}_i] \leq \sigma^{0.1}$.

Claim 5.6. $\Pr_{T \sim \mathcal{D}_p}[T \text{ is not excellent}] \leq \varepsilon/10n + O(\sigma)^{0.1\ell} \leq \varepsilon/2n$

Proof. Note that $\sum_i \mathbf{Var}[\tilde{g}_i] \leq 2 \sum_i \mathbf{Var}[\tilde{f}_i]$ and apply Markov's inequality on $(2 \sum_i \mathbf{Var}[\tilde{f}_i])^\ell$ using Claim 5.5. \square

Claim 5.7. *Let T be an excellent set. Let \mathcal{D}'_x be any δ'_x -biased distributions. Then, for $k = 1, \dots, \ell + 1$ we have*

$$\mathbf{E}_{x \sim \mathcal{D}'_x} [S_k^2(\tilde{g}_1(x), \dots, \tilde{g}_m(x))] \leq \frac{2}{k!} \cdot \sigma^{0.1k}$$

and

$$\left| \mathbf{E}_{x \sim \mathcal{D}'_x} [S_k(\tilde{g}_1(x), \dots, \tilde{g}_m(x))] \right| \leq (\varepsilon/n)^{C'}.$$

Proof. Recall that $\delta = (\varepsilon/n)^{10C'}$ and $\delta'_x = (\varepsilon/n)^{100C'}$. The first claim relies on the following:

1. S_k^2 has small spectral-norm (using Claim 5.4, since T is good) and hence is fooled by \mathcal{D}'_x . In details, its spectral-norm is at most $L_1(S_k)^2 \leq \delta^{-4}$ and \mathcal{D}'_x is δ'_x -biased. Thus

$$\left| \mathbf{E}_{x \sim U_n} [S_k^2(\tilde{g}_1(x), \dots, \tilde{g}_m(x))] - \mathbf{E}_{x \sim \mathcal{D}'_x} [S_k^2(\tilde{g}_1(x), \dots, \tilde{g}_m(x))] \right| \leq \delta^{-4} \cdot \delta'_x \leq \delta \ll \frac{1}{k!} \cdot \sigma^{0.1k}.$$

2. The expectation of $S_k^2(\tilde{g}_1(x), \dots, \tilde{g}_m(x))$ on a uniformly chosen x is at most

$$\begin{aligned} \mathbf{E}_{x \sim U_n} [S_k^2(\tilde{g}_1(x), \dots, \tilde{g}_m(x))] &= \sum_{T, T' \subseteq [m], |T|=|T'|=k} \mathbf{E}_{x \sim U_n} \left[\prod_{i \in T} \tilde{g}_i(x) \prod_{i' \in T'} \tilde{g}_{i'}(x) \right] \\ &= \sum_{T \subseteq [m], |T|=k} \mathbf{E}_{x \sim U_n} \left[\prod_{i \in T} (\tilde{g}_i(x))^2 \right] \quad (\text{Since } \mathbf{E}[\tilde{g}_i] = 0) \\ &= \sum_{T \subseteq [m], |T|=k} \prod_{i \in T} \mathbf{Var}[g_i] \leq \frac{1}{k!} \cdot \left(\sum_{i=1}^m \mathbf{Var}[\tilde{g}_i] \right)^k \leq \frac{1}{k!} \cdot \sigma^{0.1k} \\ &\hspace{15em} (\text{Maclaurin's inequality}) \end{aligned}$$

The second claim relies on the following:

1. S_k has small spectral-norm (using Claim 5.4, since T is good) and hence is fooled by \mathcal{D}'_x . In details, its spectral-norm is at most δ^{-2} and \mathcal{D}'_x is δ'_x -biased. Thus

$$\left| \mathbf{E}_{x \sim U_n} [S_k(\tilde{g}_1(x), \dots, \tilde{g}_m(x))] - \mathbf{E}_{x \sim \mathcal{D}'_x} [S_k(\tilde{g}_1(x), \dots, \tilde{g}_m(x))] \right| \leq \delta^{-2} \cdot \delta'_x \leq \delta \leq (\varepsilon/n)^{C'}.$$

2. The expectation of $S_k(\tilde{g}_1(x), \dots, \tilde{g}_m(x))$ on a uniformly chosen x is 0. \square

The next lemma combined with Claim 5.7 concludes the low-variance case, since it shows that with high probability, T is excellent, and then \mathcal{D}'_x is an (ε/n^{40}) -PRG for $\prod_{i=1}^m \tilde{f}_i$ (for a sufficiently large choice of C').

Lemma 5.8. *If T is excellent, then $\mathbf{E}_{x \sim \mathcal{D}'_x} [\prod_{i=1}^m \tilde{f}_i] = (\prod_{i=1}^m \mu_i) \pm (\varepsilon/n)^{\Omega(C')}$.*

Proof. Let $x \sim \mathcal{D}'_x$, and let E be the event that $|S_\ell(\tilde{g}_1(x), \dots, \tilde{g}_m(x))| \leq \frac{t^\ell}{\sqrt{\ell!}}$ and $|S_{\ell+1}(\tilde{g}_1(x), \dots, \tilde{g}_m(x))| \leq \frac{t^{\ell+1}}{\sqrt{(\ell+1)!}}$. Picking $t = \sigma^{0.01}$, and using Claim 5.7 the event E happens with probability at least $1 - \sigma^{\Omega(\ell)} \geq 1 - (\varepsilon/n)^{\Omega(C')}$. Assuming E occurs, Theorem 5.2 gives

$$\sum_{k=\ell}^m |S_k(\tilde{g}_1(x), \dots, \tilde{g}_m(x))| \leq 2 \cdot (6et)^\ell \leq \sigma^{\Omega(\ell)} \leq (\varepsilon/n)^{\Omega(C')}.$$

Furthermore, for sets of smaller cardinality, i.e., for $k \in \{1, \dots, \ell - 1\}$, Claim 5.7 gives

$$\left| \mathbf{E}_{x \sim \mathcal{D}'_x} [S_k(\tilde{g}_1(x), \dots, \tilde{g}_m(x))] \right| \leq (\varepsilon/n)^{C'} \quad \text{and} \quad \left| \mathbf{E}_{x \sim \mathcal{D}'_x} [S_k^2(\tilde{g}_1(x), \dots, \tilde{g}_m(x))] \right| \leq 1.$$

We would like to bound $|\mathbf{E}_{x \sim \mathcal{D}'_x} [S_k(\tilde{g}_1(x), \dots, \tilde{g}_m(x)) \cdot \mathbb{1}_E]|$ for $k \in \{1, \dots, \ell - 1\}$. Towards this end, we consider the expectation of $S_k(\tilde{g}_1(x), \dots, \tilde{g}_m(x))$ by partitioning into the two cases depending on whether the event E occurred or not.

$$\begin{aligned} & \mathbf{E}_{x \sim \mathcal{D}'_x} [S_k(\tilde{g}_1(x), \dots, \tilde{g}_m(x))] \\ &= \mathbf{E}_{x \sim \mathcal{D}'_x} [S_k(\tilde{g}_1(x), \dots, \tilde{g}_m(x)) \cdot \mathbb{1}_E] + \mathbf{E}_{x \sim \mathcal{D}'_x} [S_k(\tilde{g}_1(x), \dots, \tilde{g}_m(x)) \cdot \mathbb{1}_{\bar{E}}] \\ &= \mathbf{E}_{x \sim \mathcal{D}'_x} [S_k(\tilde{g}_1(x), \dots, \tilde{g}_m(x)) \cdot \mathbb{1}_E] \pm \sqrt{\mathbf{E}_{x \sim \mathcal{D}'_x} [S_k^2(\tilde{g}_1(x), \dots, \tilde{g}_m(x))] \cdot \Pr[\bar{E}]} \\ & \hspace{15em} \text{(Cauchy-Schwarz)} \\ &= \mathbf{E}_{x \sim \mathcal{D}'_x} [S_k(\tilde{g}_1(x), \dots, \tilde{g}_m(x)) \cdot \mathbb{1}_E] \pm \sqrt{\Pr[\bar{E}]} \\ &= \mathbf{E}_{x \sim \mathcal{D}'_x} [S_k(\tilde{g}_1(x), \dots, \tilde{g}_m(x)) \cdot \mathbb{1}_E] \pm (\varepsilon/n)^{\Omega(C')} \end{aligned}$$

Thus, $|\mathbf{E}_{x \sim \mathcal{D}'_x} [S_k(\tilde{g}_1(x), \dots, \tilde{g}_m(x)) \cdot \mathbb{1}_E]| \leq (\varepsilon/n)^{\Omega(C')}$ and we get

$$\mathbf{E}_{x \sim \mathcal{D}'_x} \left[\prod_{i=1}^m \tilde{g}_i(x) \cdot \mathbb{1}_E \right] = \mathbf{E}_{x \sim \mathcal{D}'_x} [\mathbb{1}_E] \pm \sum_{k=1}^m |\mathbf{E}_{x \sim \mathcal{D}'_x} [S_k(\tilde{g}_1(x), \dots, \tilde{g}_m(x)) \cdot \mathbb{1}_E]| = 1 \pm (\varepsilon/n)^{\Omega(C')}.$$

Since the \tilde{f}_i 's and μ_i 's are bounded in $[-1, 1]$, we get

$$\begin{aligned} \mathbf{E}_x \left[\prod_i \tilde{f}_i \right] &= \mathbf{E}_x \left[\prod_i \tilde{f}_i \cdot \mathbb{1}_E \right] + \mathbf{E}_x \left[\prod_i \tilde{f}_i \cdot \mathbb{1}_{\neg E} \right] \\ &= \left(\prod_i \mu_i \cdot \mathbf{E}_{x \sim \mathcal{D}'_x} \left[\prod_{i=1}^m \tilde{g}_i(x) \cdot \mathbb{1}_E \right] \right) \pm \mathbf{Pr}[\neg E] \\ &= \prod_i \mu_i \cdot \left(1 \pm (\varepsilon/n)^{\Omega(C')} \right) \pm (\varepsilon/n)^{\Omega(C')} = \left(\prod_i \mu_i \right) \pm (\varepsilon/n)^{\Omega(C')}. \quad \square \end{aligned}$$

5.2 High-Variance Case

In the high-variance case, there exists a $\sigma \in (0, 1]$ and an interval $I_\sigma = \{i : \mathbf{Var}[f_i] \in (0.4 \cdot \sigma^{1.1}, \sigma]\}$ (the constant 0.4 handles the case $\sigma = 1$) satisfying:

$$\sum_{i \in I_\sigma} \mathbf{Var}[f_i] > C \cdot \sigma^{-0.1} \cdot \log^2(n/\varepsilon).$$

In this case, the expected value of $\prod_{i=1}^m f_i$ under the uniform distribution is rather small:

$$\left| \mathbf{E} \left[\prod_{i=1}^m f_i \right] \right| = \prod_{i=1}^m |\mathbf{E}[f_i]| = \prod_{i=1}^m \sqrt{1 - \mathbf{Var}[f_i]} \leq e^{-\sum_{i=1}^m \mathbf{Var}[f_i]/2} \leq e^{-C \cdot \log^2(n/\varepsilon)/2} \leq \varepsilon/2.$$

Recall that the pseudorandom restriction samples a set T according to some δ_T -biased distribution \mathcal{D}_p with marginals p , and a partial assignment to the bits in T according to some δ_x -biased distribution \mathcal{D}_x . In the high variance case, it suffices to show that $\left| \mathbf{E}_{T \sim \mathcal{D}_p, x \sim \mathcal{D}_x} \left[\prod_{i=1}^m \tilde{f}_i(x) \right] \right| \leq \varepsilon/2$. Fix T, x . Denote by $f_{i,x}^T(y) = (f_i)_{T|y}(x)$. Similarly to the calculation in the case of the uniform distribution, we have

$$\left| \prod_{i=1}^m \tilde{f}_i(x) \right| = \left| \prod_{i=1}^m \mathbf{E}_{y \sim U_{[n] \setminus T}} [f_{i,x}^T(y)] \right| \leq e^{-\sum_{i=1}^m \mathbf{Var}[f_{i,x}^T]/2}$$

Thus, it suffices to show that for most $T \sim \mathcal{D}_p, x \sim \mathcal{D}_x$ we have $\sum_{i=1}^m \mathbf{Var}[f_{i,x}^T] \geq 10 \cdot \log(1/\varepsilon)$.

Theorem 5.9 (Theorem 5 - High Variance Case). *With probability $1 - \varepsilon/4$ over $T \sim \mathcal{D}_p$ and $x \sim \mathcal{D}_x$, it holds that $\sum_{i \in I_\sigma} \mathbf{Var}[f_{i,x}^T] \geq 10 \cdot \log(1/\varepsilon)$.*

Proof. Denote by $\mathbf{Tvar} := \sum_{i \in I_\sigma} \mathbf{Var}[f_i]$. By our assumption, $\mathbf{Tvar} \geq C \cdot \log^2(n/\varepsilon) \cdot \sigma^{-0.1} \geq 5\sigma^{-0.1}$. Since all functions in I_σ have variance at least $0.4 \cdot \sigma^{1.1}$ we have

$$|I_\sigma| \leq \mathbf{Tvar} \cdot \frac{1}{0.4} \cdot \sigma^{-1.1} \leq \mathbf{Tvar}^{12} \quad (9)$$

We remark that in this case, unlike the low-variance case, we do not know how to handle large σ easily, so for the rest of the proof σ can be anything between 2^{-1-b} and 1.

Fix T and x . We expand $\mathbf{Var}[f_{i,x}^T]$

$$\mathbf{Var}[f_{i,x}^T] = \mathbf{E}_{y \sim U_{[n] \setminus T}} [f_{i,x}^T(y)^2] - \mathbf{E}_{y \sim U_{[n] \setminus T}} [f_{i,x}^T(y)]^2 = 1 - \tilde{f}_i(x)^2.$$

For any fixed T , using $\mathbf{E}[f_i] = \mathbf{E}[\tilde{f}_i]$ gives

$$\mathbf{E}_{z \sim U_T} [\mathbf{Var}[f_{i,z}^T]] = 1 - \mathbf{E}[(\tilde{f}_i)^2] = (1 - \mathbf{E}[f_i^2]) - (\mathbf{E}[(\tilde{f}_i)^2] - \mathbf{E}[\tilde{f}_i^2]) = \mathbf{Var}[f_i] - \mathbf{Var}[\tilde{f}_i]$$

Claim 5.10 (Most T 's preserve variance in expectation). *With probability at least $1 - \varepsilon/16$ over the choice of $T \sim \mathcal{D}_p$, it holds that $\mathbf{E}_{z \sim U_T} [\sum_{i \in I_\sigma} \mathbf{Var}[f_{i,z}^T]] \geq \mathbf{Tvar}/2$.*

Proof. Since $\mathbf{E}_{z \sim U_T} [\mathbf{Var}[f_{i,z}^T]] = \mathbf{Var}[f_i] - \mathbf{Var}[\tilde{f}_i]$, it suffices to show that with probability $1 - \varepsilon/16$ over the choice of $T \sim \mathcal{D}_p$ we have $\sum_i \mathbf{Var}[\tilde{f}_i] \leq \sum_i \mathbf{Var}[f_i]/2$. To show that $\sum_i \mathbf{Var}[\tilde{f}_i]$ is well-concentrated we analyze its k -th moment for $k = C' \log(1/\varepsilon)$ where C' is a sufficiently large constant.

$$\mathbf{E}_{T \sim \mathcal{D}_p} \left[\left(\sum_{i \in I_\sigma} \mathbf{Var}[\tilde{f}_i] \right)^k \right] = \sum_{i_1, i_2, \dots, i_k \in I_\sigma} \mathbf{E}_T \left[\prod_{j=1}^k \mathbf{Var}[\tilde{f}_{i_j}] \right].$$

Fix $i_1, \dots, i_k \in I_\sigma$, (not necessarily distinct), then by Lemma 3.14

$$\mathbf{E}_{T \sim \mathcal{D}_p} \left[\prod_{j=1}^k \mathbf{Var}[\tilde{f}_{i_j}] \right] \leq \mathbf{E}_{T \sim \mathcal{R}_p} \left[\prod_{j=1}^k \mathbf{Var}[\tilde{f}_{i_j}] \right] + \delta_T \cdot \prod_{j=1}^k \mathbf{Var}[f_{i_j}]$$

Overall, we get

$$\mathbf{E}_{T \sim \mathcal{D}_p} \left[\left(\sum_{i \in I_\sigma} \mathbf{Var}[\tilde{f}_i] \right)^k \right] \leq \mathbf{E}_{T \sim \mathcal{R}_p} \left[\left(\sum_{i \in I_\sigma} \mathbf{Var}[\tilde{f}_i] \right)^k \right] + \delta_T \cdot \mathbf{Tvar}^k.$$

To bound $\mathbf{E}_{T \sim \mathcal{R}_p} [(\sum_{i=1}^m \mathbf{Var}[\tilde{f}_i])^k]$ we use the fact that by Theorem 3.13

$$\mathbf{E}_{T \sim \mathcal{R}_p} [\mathbf{Var}[\tilde{f}_i]] \leq p \cdot \mathbf{Var}[f_i] \leq 0.1 \cdot \mathbf{Var}[f_i]$$

and then by Chernoff's bound $\sum_{i \in I_\sigma} \mathbf{Var}[\tilde{f}_i] \leq 0.2 \cdot \mathbf{Tvar}$ with probability at least $1 - \exp(-\Omega(\mathbf{Tvar}))$. Since $\sum_i \mathbf{Var}[f_i]$ is always upper bounded by \mathbf{Tvar} , the k -moment of the sum is at most

$$(0.2 \cdot \mathbf{Tvar})^k + (\mathbf{Tvar})^k \cdot \exp(-\Omega(\mathbf{Tvar})) \leq 2(0.2 \cdot \mathbf{Tvar})^k$$

We get that $\mathbf{E}_{T \sim \mathcal{D}_p} [(\sum_{i \in I_\sigma} \mathbf{Var}[\tilde{f}_i])^k] \leq 2(0.2 \cdot \mathbf{Tvar})^k + \delta_T \cdot \mathbf{Tvar}^k$. Since $\delta_T \ll 2^{-4k}$ this is at most $3 \cdot (0.2 \cdot \mathbf{Tvar})^k$. Thus, using Markov's inequality, the probability that $\sum_{i \in I_\sigma} \mathbf{Var}[\tilde{f}_i] \geq 0.5 \cdot \mathbf{Tvar}$ is at most $3 \cdot (0.2/0.5)^k \leq \varepsilon/16$ which completes the proof. \square

Let

$$\ell \triangleq C' \cdot \log(n/\varepsilon) / \log(|I_\sigma|) \tag{10}$$

where C' is a sufficiently large constant declared before Eq. (2). Assume that ℓ is an even integer. Recall that $\delta = (\varepsilon/n)^{-10C'} = |I_\sigma|^{-10\ell}$. We again define T to be a **good** set if $\prod_{i \in R} \tilde{f}_i$ has spectral-norm at most $1/\delta$ for all sets $R \subseteq I_\sigma$ of size at most ℓ . As in Claim 5.3 the probability that T is good is at least $1 - (|I_\sigma| + 1)^\ell \cdot O(\ell b w)^3 \cdot \delta \geq 1 - \varepsilon/16$. We define T to be an **excellent** set if T is good and Claim 5.10 holds for T . Then, $\Pr[T \text{ is excellent}] \geq 1 - \varepsilon/8$.

Claim 5.11. *If T is a good set, then at most ℓ of the \tilde{f}_i 's have $L_1(\tilde{f}_i) \geq \delta^{-1/\ell}$.*

Proof. If $\tilde{f}_{i_1}, \dots, \tilde{f}_{i_\ell}$ have $L_1(\tilde{f}_{i_j}) \geq \delta^{-1/\ell}$, then their product has spectral-norm at least δ^{-1} , since $L_1(\prod_{j=1}^{\ell} \tilde{f}_{i_j}) = \prod_{j=1}^{\ell} L_1(\tilde{f}_{i_j})$ for functions defined on disjoint variables. \square

Fix an excellent set T . Let G be the of indices $i \in I_\sigma$ with $L_1(\tilde{f}_i) \leq \delta^{-1/\ell}$. We show that with high probability over x , $\sum_{i \in G} \mathbf{Var}[f_{i,x}^T] \geq 0.1 \cdot \mathbf{Tvar}$. We denote by

$$\mathbf{err}_i(x) := \mathbf{Var}[f_{i,x}^T] - \mathbf{E}_{z \sim U}[\mathbf{Var}[f_{i,z}^T]] = \mathbf{Var}[f_{i,x}^T] - (\mathbf{Var}[f_i] - \mathbf{Var}[\tilde{f}_i]).$$

Obviously $\mathbf{E}_{z \sim U}[\mathbf{err}_i(z)] = 0$ and \mathbf{err}_i is bounded in $[-\sigma, 1]$. Furthermore, we have that

$$\mathbf{err}_i(x) = (1 - \tilde{f}_i(x)^2) - (1 - \mathbf{E}_{z \sim U}[\tilde{f}_i(z)^2]) = \mathbf{E}_{z \sim U}[\tilde{f}_i(z)^2] - \tilde{f}_i(x)^2$$

Thus, the error term have small spectral-norm since $L_1(\mathbf{err}_i) \leq L_1(\tilde{f}_i)^2$. We use this fact to bound $\mathbf{E}_{x \sim \mathcal{D}_x}[(\sum_{i \in G} \mathbf{err}_i(x))^\ell]$. (recall that ℓ is an even integer.)

Claim 5.12.

$$\mathbf{E}_{x \sim \mathcal{D}_x} \left[\left(\sum_{i \in G} \mathbf{err}_i(x) \right)^\ell \right] \leq 2 \cdot (\ell \cdot \mathbf{Tvar})^{\ell/2}.$$

Proof. The spectral-norm of $(\sum_{i \in G} \mathbf{err}_i(x))^\ell$ is at most $(|G| \cdot \delta^{-2/\ell})^\ell = |G|^\ell \cdot \delta^{-2}$. Thus, any δ_x -biased distribution fools $(\sum_{i \in G} \mathbf{err}_i(x))^\ell$ with error at most $\delta_x \cdot |G|^\ell \cdot \delta^{-2}$ and we get

$$\mathbf{E}_{x \sim \mathcal{D}_x} \left[\left(\sum_{i \in G} \mathbf{err}_i(x) \right)^\ell \right] \leq \mathbf{E}_{z \sim U} \left[\left(\sum_{i \in G} \mathbf{err}_i(z) \right)^\ell \right] + \delta_x \cdot |G|^\ell \cdot \delta^{-2}.$$

To bound $\mathbf{E}_{z \sim U}[(\sum_{i \in G} \mathbf{err}_i(z))^\ell]$ we use Lemma 3.8. We observe that $\{\mathbf{err}_i(z)\}_{i \in G}$ are independent random variables, where each $\mathbf{err}_i(z)$ is bounded in $[-\mathbf{Var}[f_i], 1]$ with mean zero, and hence $\mathbf{Var}[\mathbf{err}_i] \leq \mathbf{Var}[f_i]$ (See Lemma 3.12). Applying Lemma 3.8 gives

$$\mathbf{E}_{z \sim U} \left[\left(\sum_{i \in G} \mathbf{err}_i(z) \right)^\ell \right] \leq \max\{\ell^\ell, (\ell \cdot \mathbf{Tvar})^{\ell/2}\}.$$

Since $\ell \leq \sqrt{\ell \cdot \mathbf{Tvar}}$, the upper bound on $\mathbf{E}_{z \sim U}[(\sum_{i \in G} \mathbf{err}_i(z))^\ell]$ is at most $(\ell \cdot \mathbf{Tvar})^{\ell/2}$. Finally, the upper bound with respect to $x \sim \mathcal{D}_x$ is at most

$$\mathbf{E}_{x \sim \mathcal{D}_x} \left[\left(\sum_{i \in G} \mathbf{err}_i(x) \right)^\ell \right] \leq (\ell \cdot \mathbf{Tvar})^{\ell/2} + \delta_x \cdot |G|^\ell \cdot \delta^{-2} \leq 2 \cdot (\ell \cdot \mathbf{Tvar})^{\ell/2}. \quad \square$$

Using Markov's Inequality and Claim 5.12 gives

$$\Pr_{x \sim \mathcal{D}_x} \left[\left| \sum_{i \in G} \mathbf{err}_i(x) \right| \geq \mathbf{Tvar}/4 \right] \leq 2 \cdot \left(\frac{\sqrt{\ell \cdot \mathbf{Tvar}}}{\mathbf{Tvar}/4} \right)^\ell \leq O(\sqrt{\ell/\mathbf{Tvar}})^\ell \leq O(1/\mathbf{Tvar})^{\ell/4}.$$

using $\mathbf{Tvar} \geq \Omega(\log^2(n/\varepsilon))$ and $\ell \leq O(\log(n/\varepsilon))$ in the last inequality. Furthermore, using Eqs. (9) and (10): $O(1/\mathbf{Tvar})^{\ell/4} \leq |I_\sigma|^{-\Omega(\ell)} \leq (\varepsilon/n)^{\Omega(C')} \leq \varepsilon/8$. In the complement event,

$$\sum_{i \in G} \mathbf{Var}[f_{i,x}^T] = \sum_{i \in G} (\mathbf{E}_z[\mathbf{Var}[f_{i,z}^T]] + \mathbf{err}_i(x)) \geq \mathbf{Tvar}/2 - \ell - \mathbf{Tvar}/4 \geq 0.1 \cdot \mathbf{Tvar}.$$

Since $\mathbf{Tvar} \geq \Omega(\log^2(n/\varepsilon))$, we get that with probability at least $1 - \varepsilon/4$ over $T \sim \mathcal{D}_p$ and $x \sim \mathcal{D}_x$, $\sum_i \mathbf{Var}[f_{i,x}^T] \geq 10 \log(1/\varepsilon)$. (End of Proof of Theorem 5.9) \square

6 Assigning all the variables: a pseudorandom generator for the XOR of short ROBPs

In Theorem 5, we proved that we can pseudorandomly assign p -fraction of the coordinates of $f(x) = \prod_{i=1}^m f_i(x)$, while maintaining its acceptance probability up to an additive error of ε , using $\tilde{O}(\log(n/\varepsilon) \cdot \log(b))$ random bits. In this section, we will construct a pseudorandom generator ε -fooling f , by applying Theorem 5 poly $\log \log(n/\varepsilon)$ times, combined with Lovett's [Lov08] or Viola's [Vio08] pseudorandom generator for low-degree polynomials, and CHRT's pseudorandom generator for constant-width ROBPs [CHRT17]. Our main result is:

Theorem 6. *Let $n, w, b \in \mathbb{N}$, $\varepsilon > 0$. There exists a log-space explicit pseudorandom generator that ε -fools any XOR of ROBPs of width- w and length- b (defined on disjoint sets of variables), using seed-length $O(\log(b) + \log \log(n/\varepsilon))^{2w+2} \cdot \log(n/\varepsilon)$.*

Assigning 0.9999-fraction of the variables. The first step is rather standard. By making t recursive calls to Theorem 5 we can assign all but $(1-p)^t \leq e^{-pt}$ fraction of the coordinates while maintaining the acceptance probability. Note that we rely on the fact that under restrictions, the restricted function is still of the form $\prod_{i=1}^m g_i(x)$ where each g_i is a ROPB of width w that depends on at most b bits (in other words, the class of functions we are trying to fool is closed under restrictions). Setting $t = O(1/p)$, we can assign 0.9999 fraction of the inputs bits while changing the acceptance probability by at most ε/n .

Claim 6.1. *Let $f = \prod_{i=1}^m f_i(x)$, with block-length b . Then, there is a pseudorandom restriction $\rho = (J, y)$ using at most $O(\log(b) + \log \log(n/\varepsilon))^{2w+1} \cdot \log(n/\varepsilon)$ random bits, and changing the acceptance probability by at most (ε/n) . Furthermore, J is $(\varepsilon/n)^{\omega(1)}$ biased with marginals 0.0001, and y is $(\varepsilon/n)^{\omega(1)}$ biased.*

Proof Sketch. Apply Theorem 5 with error ε/n^2 for $t = \log(0.0001)/\log(1-p) = O(1/p)$ times recursively, with independent random bits per each iteration. Denoting by $J_0 = [n]$, this generates t -pseudorandom restrictions $(J_1, y_1), (J_2, y_2), \dots, (J_t, y_t)$ where $J_i \subseteq J_{i-1}$ and $y_i \in \{\pm 1\}^{J_{i-1} \setminus J_i}$ for all $i \in [t]$. To be more precise, for each i , $T_i = (J_{i-1} \setminus J_i)$ is the δ_T -biased subset with marginals p in the description of the generator in Theorem 5 and $y_i \in \{\pm 1\}^{T_i}$ is its assignment sampled from a δ_x -biased distribution. We take $J = J_t$ and $y \in \{\pm 1\}^{[n] \setminus J}$ to be the concatenation of y_1, \dots, y_t . By the hybrid argument,

$$\left| \mathbf{E}_{z \sim U_n} [f] - \mathbf{E}_{J, y} \mathbf{E}_{x \sim U_J} [f(\text{Sel}_J(x, y))] \right| \leq (\varepsilon/n^2) \cdot t \leq \varepsilon/n.$$

The amount of random bits used to sample the restriction is

$$O(p^{-1} \cdot w \log(n/\varepsilon)(\log \log(n/\varepsilon) + \log(b)) \leq O(\log(b) + \log \log(n/\varepsilon))^{2w+1} \cdot \log(n/\varepsilon).$$

Next, we claim that J is a $(\varepsilon/n)^{\omega(1)}$ -biased with marginals 0.0001. Recall that $J_{i-1} \setminus J_i$ are δ_T -biased with marginals p , and that $\delta_T = (\varepsilon/n)^{\omega(1)}$. By Claim 3.7, for any subset $S \subseteq [n]$ of size at most $\log(\delta_T)/10$ we have

$$\Pr[S \subseteq J] = \Pr[S \subseteq J_0] \cdot \Pr[S \subseteq J_1 | S \subseteq J_0] \cdots \Pr[S \subseteq J_t | S \subseteq J_{t-1}]$$

$$= ((1 - p)^{|S|} \pm 2^{|S|} \delta_T)^t = (0.0001)^{|S|} \pm (\varepsilon/n)^{\omega(1)}$$

which implies by monotonicity that for larger subsets S , we have $\Pr[S \subseteq J] \leq (\varepsilon/n)^{\omega(1)}$.

Finally, we claim that conditioned on J and in fact for any choice of J_1, \dots, J_t , y is $(\varepsilon/n)^{\omega(1)}$ -biased. This is due to the fact that y is the concatenation of y_1, \dots, y_t where each y_i is $\delta_x = (\varepsilon/n)^{\omega(1)}$ -biased. \square

We would like to claim that $f = \prod_{i=1}^m f_i$ simplifies after assigning 0.9999 of the coordinates. For a particular function f_i , with high probability, at least $1 - 1/32^b$, the block length decreases under a random restriction by a factor of 2. This is due to the fact that on expectation at most $0.0001 \cdot b$ of the variables will survive, and we can apply Chernoff's bound. Now, if $m \leq 16^b$, we can apply a union bound and get that with high probability the block-length decreases by a factor of 2 in all functions f_1, \dots, f_m simultaneously. We seem to have been making progress, going from block-length b to block-length $b/2$, and we might hope that $\log(b)$ iterations of Claim 6.1 are enough to get a function that depends on $O(1)$ many variables (which is easy to fool). But, in order to carry the argument, even in the second step, we need to be able to afford the union bound on all functions. Ideally, the number of functions that are still alive also decreases from at most 16^b to at most $16^{b/2}$, and a similar union bound works replacing b by $b/2$. We can continue similarly as long as in each iteration the block-length decreases by half and the number of functions by a square root.

We run into trouble if at some iteration we have more than $16^{b'}$ functions of block-length b' . The first observation is that in this case the total variance of the functions is extremely high, exponential in b' . Recall that the expected value of the product is exponentially small in the total variance. This means that the expected value of the product is doubly-exponentially small in b' . The second observation is that under $(1 - \alpha)$ -random restrictions, on average, the total variance decreases by a factor of α . Hence, we aim to apply a pseudorandom restriction assigning $(1 - \exp(-b'))$ fraction of the variables alive, while keeping the total variance higher than $\log(n/\varepsilon)$. This restriction is extremely aggressive, keeping only a polynomial fraction of the remaining variables alive (compared to say a constant fraction in Claim 6.1). However, we claim that in this case, such a restriction maintains the total variance high and thus the expected value of $\prod_{i=1}^m f_i(x)$ small (at most $\text{poly}(\varepsilon/n)$) in absolute value.

The nice thing about these “aggressive pseudorandom restrictions” is that they keep variables alive with such small probability that with high probability each function f_i will depend on at most $O(1)$ variables after the restriction, except for a small number of functions covering at most $O(\log(n/\varepsilon))$ “bad variables”. This will allow us to fool the restricted function using Lovett's [Lov08] or Viola's [Vio08] pseudorandom generator for low-degree polynomials. In the next section, we explain how to handle this case in more details. Then, in Section 6.2 we describe as a thought experiment a “fake PRG”: an adaptive process that fools the XOR of short ROBPs, but depends on the function being fooled. In Section 6.3 we show how to eliminate the adaptiveness and construct a true PRG for this class of functions.

6.1 PRG for the XOR of many functions with block-length b

Let $\mathcal{F}_{b,n,t}$ be the class of functions of the form $f(x) = f_0(x) \cdot \prod_{i=1}^m f_i(x)$ where f_0, \dots, f_m are Boolean functions on disjoint sets of variables, f_0 (the ‘junta’) depends on at most t variables, f_1, \dots, f_m are **non-constant** and depend on at most b variables and $16^b \leq m \leq 2 \cdot 16^{2b}$.

Lemma 6.2. *There exists a constant $C > 0$ such that the following holds. For all n, b, t such that $C \cdot \log \log(n/\varepsilon) \leq b \leq \log(n)$, there exists a log-space explicit pseudorandom generator $\mathbf{G}_{\oplus \text{Many}}(b, n, t, \varepsilon) : \{\pm 1\}^{O(t + \log n/\varepsilon)} \rightarrow \{\pm 1\}^n$ that ε -fools $\mathcal{F}_{b, n, t}$.*

Algorithm 1 The Pseudorandom Generator $\mathbf{G}_{\oplus \text{Many}}(b, n, t, \varepsilon)$

Input: A block-length b , the output length n , a junta-size t , an error parameter $\varepsilon \in (0, 1)$

- 1: Set $x := 1^n$
- 2: Pick $T \subseteq [n]$ using a $(\varepsilon/n)^{10C}$ -biased distribution with marginals 2^{-b} .
- 3: Assign coordinates of x in $[n] \setminus T$ using a $(\varepsilon/n)^{10C}$ -biased distribution.
- 4: Assign coordinates of x in T using Viola's generator with error $\frac{\varepsilon}{4} \cdot (\frac{\varepsilon}{n})^C \cdot 2^{-t}$ and degree 16.
- 5: **return** x .

Lemma 6.3. *Let $C > 0$ be a sufficiently large constant. Let $C \cdot \log \log(n/\varepsilon) \leq b \leq \log(n)$ be some integer. Let f_1, \dots, f_m be non-constant Boolean functions that depend on disjoint sets of at most b variables each. Assume $m \geq 16^b$. Suppose T is $(\varepsilon/n)^{10C}$ -biased distribution with marginals 2^{-b} . Suppose x is sampled from a $(\varepsilon/n)^{10C}$ -biased distribution. Then, with probability at least $1 - (\varepsilon/n)^{C/4}$, at least 4^b of the functions $(f_i)_{T|x}$ will be non-constant.*

Proof. Without loss of generality $m = 16^b$. Let $k = C \log(n/\varepsilon)/b$, and note that $k \leq 2^b$ since $b \geq C \cdot \log \log(n/\varepsilon)$ for a sufficiently large constant $C > 0$.

Let $B_1, \dots, B_m \subseteq [n]$ be the disjoint sets of variables on which f_1, \dots, f_m depend respectively. For any function $f_i : \{\pm 1\}^{B_i} \rightarrow \{\pm 1\}$, there exists a sensitive pair of inputs $(\alpha^{(i)}, \beta^{(i)}) \in \{\pm 1\}^{B_i}$ such that $\alpha^{(i)}$ and $\beta^{(i)}$ differ in exactly one coordinate j_i and such that $f_i(\alpha^{(i)}) \neq f_i(\beta^{(i)})$. We say that the sensitive pair “survives” the random restriction defined by (T, x) if both $\alpha^{(i)}$ and $\beta^{(i)}$ are consistent with the partial assignment defined by the restriction (i.e., if they agree with x on $B_i \setminus T$). For each function f_1, \dots, f_m fix one sensitive pair $(\alpha^{(1)}, \beta^{(1)}), \dots, (\alpha^{(m)}, \beta^{(m)})$ and denote by $\mathcal{E}_1, \dots, \mathcal{E}_m$ the events that these sensitive pairs survive. Next, we claim that $\mathcal{E}_1, \dots, \mathcal{E}_m$ are almost k -wise independent. We compare them to the events $\mathcal{E}'_1, \dots, \mathcal{E}'_m$ that indicate whether the sensitive pairs survive under a truly random restriction sampled from $\mathcal{R}_{2^{-b}}$. Denote by $p_i = \Pr(\mathcal{E}'_i)$. Observe that $p_i \geq 2^{1-2b}$ since in order for the pair to survive it is enough that the sensitive coordinate remains alive (happens with probability 2^{-b}) and that the partial assignment on the remaining coordinates agrees with $\alpha^{(i)}$ (happens with probability at least 2^{1-b}). Then,

$$\mathbf{E} \left[\left(\sum_{i=1}^m (\mathbb{1}_{\mathcal{E}_i} - p_i) \right)^k \right] \leq \mathbf{E} \left[\left(\sum_{i=1}^m (\mathbb{1}_{\mathcal{E}'_i} - p_i) \right)^k \right] + (2m)^k \cdot \max_{K \subseteq [m]: |K| \leq k} \left| \mathbf{E} \left[\prod_{i \in K} \mathbb{1}_{\mathcal{E}_i} \right] - \mathbf{E} \left[\prod_{i \in K} \mathbb{1}_{\mathcal{E}'_i} \right] \right|.$$

We upper bound the first and second summands separately. By Lemma 3.8 and Lemma 3.12, the first summand is upper bounded by $\max\{k^k, (Vk)^{k/2}\}$ where

$$V := \sum_{i=1}^m p_i.$$

Since $V \geq m \cdot 2^{1-2b} = 2 \cdot 4^b \geq k$, the first summand is upper bounded by $(Vk)^{k/2}$.

Next, we upper bound the second summand. By Vazirani's XOR lemma, since x is $(\varepsilon/n)^{10C}$ -biased, we have that the marginal distribution of any set of at most $k \cdot b$ bits in x is $(\varepsilon/n)^{10C} \cdot 2^{kb/2}$ -close to uniform in statistical distance. Since T is $(\varepsilon/n)^{10C}$ -biased with marginals 2^{-b} , using Claim 3.7 we have that the marginal distribution on any set of at most k coordinates in T is $(\varepsilon/n)^{10C} \cdot 4^k$ -close in statistical distance to the distribution sampled according to $\mathcal{R}_{2^{-b}}$. Thus, $|\mathbf{E}[\prod_{i \in K} \mathbb{1}_{\mathcal{E}_i}] - \mathbf{E}[\prod_{i \in K} \mathbb{1}_{\mathcal{E}'_i}]| \leq (2^{kb/2} + 4^k) \cdot (\varepsilon/n)^{10C} \leq 2^{kb/2+1} \cdot (\varepsilon/n)^{10C}$ and we get $(2m)^k \cdot 2^{kb/2+1} \cdot (\varepsilon/n)^{10C} \leq 1$.

Combining the bounds on both summands we get

$$\mathbf{E} \left[\left(\sum_{i=1}^m (\mathbb{1}_{\mathcal{E}_i} - p_i) \right)^k \right] \leq (Vk)^{k/2} + 1 \leq 2 \cdot (Vk)^{k/2}.$$

Using $V = \sum_{i=1}^m p_i$ we get

$$\Pr \left[\sum_{i=1}^m \mathbb{1}_{\mathcal{E}_i} \leq V/2 \right] \leq \Pr \left[\left(\sum_{i=1}^m (\mathbb{1}_{\mathcal{E}_i} - p_i) \right)^k \geq (V/2)^k \right] \leq 2(Vk)^{k/2} \cdot (V/2)^{-k}$$

Using $V \geq 2 \cdot 4^b$ and $k \leq 2^b$ we get $\Pr [\sum_{i=1}^m \mathbb{1}_{\mathcal{E}_i} \leq V/2] \leq 2(4k/V)^{k/2} \leq 2 \cdot (2/2^b)^{k/2} \leq (\varepsilon/n)^{C/4}$. In the complement event, at least $V/2 \geq 4^b$ of the functions $(f_1)_{T|x}, \dots, (f_m)_{T|x}$ are non-constant. \square

Lemma 6.4. *Let $f : \mathbb{F}_2^n \rightarrow \{\pm 1\}$. Suppose $f(x) = h(x) \cdot (-1)^{g(x)}$ where h is a k -junta and g is a polynomial of degree- d over \mathbb{F}_2 . If \mathcal{D} fools degree- d polynomials over \mathbb{F}_2 with error ε , then \mathcal{D} fools f with error $\varepsilon \cdot 2^{k/2}$.*

Proof. Let J be the set of variables on which h depends. Using the Fourier transform of h : $h(x) = \sum_{S \subseteq J} \widehat{h}(S) \cdot (-1)^{\sum_{i \in S} x_i}$ we write f as $f(x) = \sum_{S \subseteq J} \widehat{h}(S) \cdot (-1)^{\sum_{i \in S} x_i + g(x)}$. Note that $\sum_{i \in S} x_i + g(x)$ is a polynomial of degree- d over \mathbb{F}_2 as well, thus we get

$$\begin{aligned} \left| \mathbf{E}_{x \sim U}[f(x)] - \mathbf{E}_{x \sim \mathcal{D}}[f(x)] \right| &\leq \sum_S |\widehat{h}(S)| \cdot \left| \mathbf{E}_{x \sim U}[(-1)^{\sum_{i \in S} x_i + g(x)}] - \mathbf{E}_{x \sim \mathcal{D}}[(-1)^{\sum_{i \in S} x_i + g(x)}] \right| \\ &\leq L_1(h) \cdot \varepsilon \leq 2^{k/2} \cdot \varepsilon. \end{aligned} \quad \square$$

Proof of Lemma 6.2. First note that f has very small expectation under the uniform distribution

$$\left| \mathbf{E}_{z \sim U} \left[f_0(z) \cdot \prod_{j=1}^m f_j(x) \right] \right| \leq (1 - 2^{-b})^{16b} \ll \frac{\varepsilon}{4}.$$

using the assumption $b \geq C \log \log(n/\varepsilon)$. Thus, we need to maintain low-expectancy under the pseudorandom assignment. By Lemma 6.3, with probability at least $1 - (\varepsilon/n)^{C/4} \geq 1 - \frac{\varepsilon}{100}$ after the aggressive random restriction at least 4^b of the functions f_1, \dots, f_m remain non-constant. Since $b \geq C \log \log(n/\varepsilon)$ we maintained the low-expectancy under aggressive random restrictions. That is, whenever 4^b of the functions f_1, \dots, f_m remain non-constant under restriction, the expected value of the restricted function under the uniform distribution is at most $(1 - 2^{-b})^{4^b} \ll \varepsilon/4$ in absolute value.

Furthermore, we wish to show that with high probability, except for a set of at most $C \log(n/\varepsilon)$ “bad variables” all functions have block-length at most 16. Recall that there are at most $2 \cdot 16^{2b}$ functions. The probability that any particular k variables survive is at most $2^{-bk} + (\varepsilon/n)^{10C}$. Pick $k = C \log(n/\varepsilon)/b \leq C \log(n/\varepsilon)$. The probability that at least k variables in at most ℓ functions survive is

$$\binom{2 \cdot 16^{2b}}{\ell} \cdot \binom{\ell \cdot b}{k} \cdot (2^{-bk} + (\varepsilon/n)^{10C}) \leq 2 \cdot 2^{\ell+9b\ell-bk} \leq 2^{10b\ell-bk}$$

If $\ell \leq k/16$, then this probability is at most $2^{-6bk/16} = (\varepsilon/n)^{6C/16} \ll \frac{\varepsilon}{100}$. This means that, with high probability, there are less than k variables from all functions with more than 16 effective variables remaining. Otherwise, there would have been $\ell \leq k/16$ functions accountable to a total number of more than k variables that remained alive and effective, under the restriction.

Overall, with probability at least $1 - \varepsilon/50$ we are left with the XOR of a small-junta, on at most $t + C \log(n/\varepsilon)$ variables, and an XOR of at least 4^b non-constant functions on at most 16 variables (i.e., a degree 16 polynomial). Moreover, the restricted function has expected value at most $\varepsilon/4$ in absolute value under the uniform distribution. Using Claim 6.4 we get that Viola’s [Vio08] or Lovett’s [Lov08] PRG for low-degree polynomials $\varepsilon/4$ -fools the remaining function. Combining all estimates we get that the expected value of the restricted function under our distribution is at most $3\varepsilon/4$ in absolute value which completes the proof. \square

6.2 A thought experiment

We are ready to describe the pseudo-random restriction process in full detail. We start by describing a process that iteratively “looks” at the restricted functions in order to decide which pseudorandom restriction to apply next: the one described in Lemma 6.1 or the one from Lemma 6.3. This ultimately defines a decision tree of random restrictions. We then show in Section 6.3 how to transform the adaptive process into a non-adaptive pseudorandom generator that (by definition) does not depend on the function it tries to fool. Namely, we would generate a pseudorandom string that fools the function no matter what path was taken in the decision tree.

We start with $m \leq n$ blocks of length b . We assume that $m \leq 16^b$ (if not set $b = \log_{16}(m)$).

Algorithm 2 an “adaptive pseudorandom generator”

- 1: **for** $i = 0, 1, \dots$ **do**
 - 2: Let $b_i = b/2^i$.
 - 3: **if** $b_i \leq C \log \log(n/\varepsilon)$ **then** apply CHRT’s PRG on the remaining coordinates, and Halt!
 - 4: **if** more than 16^{b_i} of the restricted functions are non-constant and depend on at most b_i variables **then** apply $\mathbf{G}_{\oplus \text{Many}}(b_i, 10 \log(n/\varepsilon), n, \varepsilon/2)$ from Lemma 6.2 on the remaining variables, and Halt!
 - 5: **else** apply the pseudorandom restriction from Lemma 6.1 on the remaining variables.
-

Next, we show that the process yields a pseudorandom string fooling $f = \prod_{i=1}^m f_i(x)$. First, note that the process either stops at Step 3 or at Step 4. In both cases we assign

all the variables according to some pseudorandom generator, hence all the variables will be assigned by the end of the process.

For $i = 0, 1, \dots$. Let T_i be the set of coordinates that remain alive at the beginning of the i -th iteration. Denote by $f_j^{(i)}$ the j -th function under the restriction at the beginning of the i -th iteration. Define $\text{Var}[f_j^{(i)}]$ to be the set of variables that affect the output of $f_j^{(i)}$. For example if $f_j^{(i)}$ is a constant function, then $\text{Var}[f_j^{(i)}] = \emptyset$.

Let $\text{Good}_i = \{j : 1 \leq |\text{Var}[f_j^{(i)}]| \leq b_i\}$ be the set of functions that depend on some but not more than b_i variables, $\text{Bad}_i = \{j : |\text{Var}[f_j^{(i)}]| > b_i\}$ be the set of functions that depend on more than b_i variables and $\text{VarBad}_i = \bigcup_{j \in \text{Bad}_i} \text{Var}[f_j^{(i)}]$.

Claim 6.5. *Let $b_i > C \log \log(n/\varepsilon)$. Suppose $|\text{VarBad}_i| \leq 10 \log(n/\varepsilon)$ and $|\text{Good}_i| \leq 16^{b_i}$. Then, with probability at least $1 - (\varepsilon/n)$ we have $|\text{VarBad}_{(i+1)}| \leq 10 \log(n/\varepsilon)$.*

Proof. Under the assumptions we reach Step 5 in Algorithm 2. We show that:

1. With probability at least $1 - \frac{1}{2}(\varepsilon/n)$, at most $5 \log(n/\varepsilon)$ of the variables in VarBad_i remain alive in Step 5.
2. With probability at least $1 - \frac{1}{2}(\varepsilon/n)$, at most $5 \log(n/\varepsilon)$ new variables are added to $\text{VarBad}_{(i+1)}$.

Both claims rely on the fact that any set of $k \leq 5 \log(n/\varepsilon)$ variables remain alive under the pseudorandom restriction in Lemma 6.1 with probability at most $2 \cdot 0.0001^k$.

This first item follows since the probability that more than $5 \log(n/\varepsilon)$ variables in VarBad_i survive is at most

$$\binom{10 \log(n/\varepsilon)}{5 \log(n/\varepsilon)} \cdot 2 \cdot 0.0001^{5 \log(n/\varepsilon)} \leq \frac{1}{2}(\varepsilon/n).$$

As for the second item, we start with the case where $b_i \leq 2 \log(n/\varepsilon)$. Assume that more than $5 \log(n/\varepsilon)$ new variables were added to $\text{VarBad}_{(i+1)}$. This implies that there is a set of $k = \lceil 5 \log(n/\varepsilon)/(b_i/2) \rceil$ good functions in step i that are accountable to at least $5 \log(n/\varepsilon)$ bad variables in step $i + 1$. The latter event happens with probability at most

$$\binom{16^{b_i}}{k} \cdot \binom{kb_i}{5 \log(n/\varepsilon)} \cdot 2 \cdot 0.0001^{5 \log(n/\varepsilon)} \leq 32^{b_i k} \cdot 2 \cdot 0.0001^{5 \log(n/\varepsilon)} \leq \frac{1}{2}(\varepsilon/n)$$

(where we used $kb_i \leq b_i + 10 \log(n/\varepsilon) \leq 12 \log(n/\varepsilon)$) which finishes the case $b_i \leq 2 \log(n/\varepsilon)$.

In the case where $b_i > 2 \log(n/\varepsilon)$, we show that with high probability all good functions remain good. For each individual function, using Markov's inequality

$$\begin{aligned} \Pr[|\text{Var}[f_j^{(i+1)}]| \geq b_i/2] &\leq \frac{\mathbf{E} \left[\binom{|\text{Var}[f_j^{(i+1)}]|}{\log(n/\varepsilon)} \right]}{\binom{b_i/2}{\log(n/\varepsilon)}} \leq 2 \cdot 0.0001^{\log n/\varepsilon} \cdot \frac{\binom{b_i}{\log(n/\varepsilon)}}{\binom{b_i/2}{\log(n/\varepsilon)}} \\ &\leq 2 \cdot 0.0001^{\log n/\varepsilon} \cdot \frac{(e \cdot b_i / \log(n/\varepsilon))^{\log(n/\varepsilon)}}{((b_i/2) / \log(n/\varepsilon))^{\log(n/\varepsilon)}} \\ &= 2 \cdot (0.0001 \cdot e \cdot 2)^{\log(n/\varepsilon)} \leq \frac{1}{2}(\varepsilon/n^2). \end{aligned}$$

Thus, we can apply a union bound and show that all good functions remain good with probability at least $1 - \frac{1}{2}(\varepsilon/n)$. \square

Say the process finished. We shall assume that $|\text{VarBad}_i| \leq 10 \log(n/\varepsilon)$ for every iteration i until the process stopped. By Claim 6.5 this happens with probability at least $1 - \log(b) \cdot (\varepsilon/n) \geq 1 - \varepsilon/2$ by applying a union bound on the at most $\log(b)$ iterations. We wish to show that we constructed a pseudorandom string fooling f . We consider two cases:

1. We stopped on Step 3 at some iteration i . If $i = 0$ then $m \leq 16^{b_0} \leq \text{poly log}(n/\varepsilon)$ and at most $\text{poly log}(n/\varepsilon)$ variables remain that affect the functions $f_j^{(i)}$. Otherwise, since $|\text{Good}_{(i-1)}| \leq 16^{2b_i} \leq \text{poly log}(n/\varepsilon)$ and $|\text{VarBad}_{(i-1)}| \leq 10 \log(n/\varepsilon)$, at most $\text{poly log}(n/\varepsilon)$ variables remain that affect the functions $f_j^{(i-1)}$, and thus at most $\text{poly log}(n/\varepsilon)$ variables remain that affect the functions $f_j^{(i)}$. Thus, we can write $\prod_{j=1}^m f_j^{(i)}$ as a ROBP of width $2w$ and length $\text{poly log}(n/\varepsilon)$, which is $(\varepsilon/2)$ -fooled by the pseudorandom generator from Theorem 3.11 using $\tilde{O}(\log(n/\varepsilon))$ random bits.
2. We stopped at Step 4 at some iteration i . Certainly, $|\text{Good}_i| \leq |\text{Good}_{(i-1)}| + |\text{Bad}_{(i-1)}| \leq 16^{2b_i} + 10 \log(n/\varepsilon) \leq 2 \cdot 16^{2b_i}$. Thus, we are in the case that was handled in Section 6.1, with $t \leq 10 \log(n/\varepsilon)$. Indeed, Lemma 6.2 guarantees that $\mathbf{G}_{\oplus \text{Many}}(b_i, 10 \log(n/\varepsilon), n, \varepsilon/2)$ fools the remaining function with error at most $\varepsilon/2$ using $O(\log(n/\varepsilon))$ random bits.

6.3 The actual generator

Algorithm 2 described the pseudo-random generator as if we knew whether or not the condition in step 3 holds. However, a pseudorandom generator cannot depend on the function it tries to fool. To overcome this issue, we use the following general observation regarding pseudorandom generators.

Claim 6.6. *Say there are two families of functions \mathcal{F}_1 and \mathcal{F}_2 that are both closed under shifts (i.e., closed under XORing a constant string to the input). Say that \mathcal{D}_1 is an ε -PRG for \mathcal{F}_1 and \mathcal{D}_2 is an ε -PRG for \mathcal{F}_2 then $\mathcal{D}_1 \oplus \mathcal{D}_2$ is an ε -PRG for $\mathcal{F}_1 \cup \mathcal{F}_2$.*

Proof. Let $f \in \mathcal{F}_1 \cup \mathcal{F}_2$, we show that $\mathcal{D}_1 \oplus \mathcal{D}_2$ fools f . By symmetry assume $f \in \mathcal{F}_1$.

$$\begin{aligned} \left| \mathbf{E}_{\substack{x_1 \sim \mathcal{D}_1 \\ x_2 \sim \mathcal{D}_2}} [f(x_1 \oplus x_2)] - \mathbf{E}_{z \sim U} [f(z)] \right| &= \left| \mathbf{E}_{\substack{x_1 \sim \mathcal{D}_1 \\ x_2 \sim \mathcal{D}_2}} [f(x_1 \oplus x_2)] - \mathbf{E}_{\substack{x_1 \sim U \\ x_2 \sim \mathcal{D}_2}} [f(x_1 \oplus x_2)] \right| \\ &\leq \mathbf{E}_{x_2 \sim \mathcal{D}_2} \left| \mathbf{E}_{x_1 \sim \mathcal{D}_1} [f(x_1 \oplus x_2)] - \mathbf{E}_{x_1 \sim U} [f(x_1 \oplus x_2)] \right| \\ &= \mathbf{E}_{x_2 \sim \mathcal{D}_2} \left| \mathbf{E}_{x_1 \sim \mathcal{D}_1} [f_{x_2}(x_1)] - \mathbf{E}_{x_1 \sim U} [f_{x_2}(x_1)] \right| \end{aligned}$$

where $f_y(x) := f(x \oplus y)$. Since \mathcal{F}_1 is closed under shifts, we have that $f_{x_2} \in \mathcal{F}_1$ thus \mathcal{D}_1 ε -fools f_{x_2} and we get $\mathbf{E}_{x_2 \sim \mathcal{D}_2} |\mathbf{E}_{x_1 \sim \mathcal{D}_1} [f_{x_2}(x_1)] - \mathbf{E}_{x_1 \sim U} [f_{x_2}(x_1)]| \leq \varepsilon$. \square

The actual generator would proceed as follows.

Algorithm 3 The Pseudorandom Generator $\mathbf{GXOR}(T, w, b, \varepsilon)$

Input: a set $T \subseteq [n]$ of the “live” coordinates, a width w , an integer b , a parameter $\varepsilon \in (0, 1)$.

- 1: **if** $b \leq C \log \log(n/\varepsilon)$ **then return** $\mathbf{CHRT}(n, n', 2w, \varepsilon)|_T$ for $n' = 2 \cdot 16^{2b} \cdot b + 10 \log(n/\varepsilon)$
 - 2: Let $x := \mathbf{G}_{\oplus \text{Many}}(b, t, n, \varepsilon)|_T$ for $t = 10 \log(n/\varepsilon)$.
 - 3: Pick $T' \subseteq T$, $y \in \{\pm 1\}^{T \setminus T'}$ according to Claim 6.1
 - 4: Let $z := \mathbf{GXOR}(T', w, b/2, \varepsilon/2)$.
 - 5: **return** $x \oplus \text{Sel}_{T'}(z, y)$.
-

Claim 6.7 (Proof of Correctness). *Let $T \subseteq [n]$. Suppose f_1, \dots, f_m are functions on disjoint sets of T . Suppose each function depends on at most b variables except for a total of at most $10 \log(n/\varepsilon)$ variables, and the number of non-constant functions is at most $2 \cdot 16^{2b}$. Then, $\mathbf{GXOR}(T, w, b, \varepsilon)$ fools $f = \prod_{i=1}^m f_i$ with error ε .*

Proof. We prove the claim by induction on b . If $b \leq C \log \log(n/\varepsilon)$ then Theorem 3.11 implies correctness. If $b > C \log \log(n/\varepsilon)$ then we consider the following two cases:

1. If there are more than 16^b good functions, then $x = \mathbf{G}_{\oplus \text{Many}}(b, t, n, \varepsilon)|_T$ fools $\prod_{i=1}^m f_i$ with error ε .
2. Otherwise, there are at most 16^b good functions and we apply Step 3. According to Claim 6.1, the average acceptance probability of $f_{T'|y}$ is $\varepsilon/4$ close to that of f . Furthermore, with probability at least $1 - \varepsilon/4$ all functions $(f_1)_{T'|y}, \dots, (f_m)_{T'|y}$ depend on at most $b/2$ variables except for at most $10 \log(n/\varepsilon)$ variables (by Claim 6.5). In such a case, the number of non-constant functions among $(f_1)_{T'|y}, \dots, (f_m)_{T'|y}$ is at most $16^b + 10 \log(n/\varepsilon) \leq 2 \cdot 16^b$. Using induction, $z = \mathbf{GXOR}(T', w, b/2, \varepsilon/2)$ fools $f_{T'|y}$ with error $\varepsilon/2$, and we get that $\text{Sel}_{T'}(z, y)$ fools f with error ε .

Since we have a pseudorandom generator fooling the function in each case, Claim 6.6 shows that $x \oplus \text{Sel}_{T'}(z, y)$ fools f with error ε . \square

Claim 6.8 (Seed Length). *The amount of random bits used to calculate $\mathbf{GXOR}([n], w, b, \varepsilon)$ is at most $O(\log(b) + \log \log(n/\varepsilon))^{2w+2} \cdot \log(n/\varepsilon)$.*

Proof. Unwrapping the recursive calls in the evaluation of $\mathbf{GXOR}([n], w, b, \varepsilon)$ we see that there are at most $\log(b)$ recursive calls to the procedure and that the error parameters are at least $\varepsilon/2^{\log(b)} \geq \varepsilon/n$ in all of them.

We apply the generator from Theorem 3.11 only once during these recursive calls, on a ROBP of width- w and length $\text{poly} \log(n/\varepsilon)$. Thus, the application of Theorem 3.11 uses at most $O(\log \log(n/\varepsilon)^{w+2} \log(n/\varepsilon))$ random bits.

The partial assignment from Claim 6.1 uses at most $O(\log(b) + \log \log(n/\varepsilon))^{2w+1} \cdot \log(n/\varepsilon)$ each time we invoke it, and we invoke it at most $\log(b)$ times.

The generator $\mathbf{G}_{\oplus \text{Many}}$ uses $O(\log(n/\varepsilon))$ random bits each time we invoke it, and we invoke it at most $\log(b)$ times. \square

Claims 6.7 and 6.8 complete the proof Theorem 6 with $\mathbf{GXOR}([n], w, b, \varepsilon)$ as the generator.

6.4 Pseudorandom generator for read-once polynomials

Next, we restate and prove Theorem 3.

Theorem 3. *There exists a log-space explicit ε -PRG for the class of read-once polynomials on n variables with seed-length $\tilde{O}(\log(n/\varepsilon))$.*

Proof. We show that $\mathbf{GXOR}([n], 2, \log(8n/\varepsilon), \varepsilon/8n)$ fools any read-once polynomial with error at most ε . Its seed length is $O((\log \log(n/\varepsilon))^6 \cdot \log(n/\varepsilon))$.

A read-once polynomial can be written as the XOR of AND functions on disjoint variables, i.e., as the XOR of width-2 ROBPs on disjoint variables. It remains to show that these ROBPs are short. Rather, we show that any PRG that $(\varepsilon/8n)$ -fools read-once polynomials of degree at most $b = \log(8n/\varepsilon)$ also ε -fools all read-once polynomials. Let

$$f(x) = \sum_{i=1}^m \prod_{j \in B_i} x_j$$

be a read-once polynomial over \mathbb{F}_2 , where B_1, \dots, B_m are disjoint subsets of $[n]$. Without loss of generality let B_1, \dots, B_ℓ be the blocks of length bigger than b . Let

$$f'(x) = \sum_{i=\ell+1}^m \prod_{j \in B_i} x_j,$$

be the sum over monomials of degree at most b of f . Let $\mathcal{D} = \mathbf{GXOR}([n], 2, \log(8n/\varepsilon), \varepsilon/8n)$. By triangle inequality

$$\begin{aligned} \left| \mathbf{E}_{x \sim \mathcal{D}}[f(x)] - \mathbf{E}_{x \sim U_n}[f(x)] \right| &\leq \mathbf{Pr}_{x \sim \mathcal{D}}[f(x) \neq f'(x)] + \mathbf{Pr}_{x \sim U_n}[f(x) \neq f'(x)] + \left| \mathbf{E}_{x \sim \mathcal{D}}[f'(x)] - \mathbf{E}_{x \sim U_n}[f'(x)] \right| \\ &\leq \sum_{i=1}^{\ell} \mathbf{Pr}_{x \sim \mathcal{D}}[\wedge_{j \in B_i}(x_j = 1)] + \sum_{i=1}^{\ell} \mathbf{Pr}_{x \sim U_n}[\wedge_{j \in B_i}(x_j = 1)] + \varepsilon/8n \end{aligned} \quad (11)$$

For $i \in \{1, \dots, \ell\}$, since $|B_i| \geq b$, we have $\mathbf{Pr}_{x \sim U_n}[\wedge_{j \in B_i}(x_j = 1)] \leq 2^{-b} \leq \varepsilon/8n$. As for the distribution \mathcal{D} , by monotonicity

$$\mathbf{Pr}_{x \sim \mathcal{D}}[\wedge_{j \in B_i}(x_j = 1)] \leq \mathbf{Pr}_{x \sim \mathcal{D}}[\wedge_{j \in B'_i}(x_j = 1)]$$

where B'_i is any arbitrary subset of exactly b variables from B_i . Since \mathcal{D} fools degree- b read-once polynomials with error at most $\varepsilon/8n$, and $\wedge_{j \in B'_i}(x_j = 1)$ is such a polynomial, we get that $\mathbf{Pr}_{x \sim \mathcal{D}}[\wedge_{j \in B'_i}(x_j = 1)]$ is at most $2^{-b} + \varepsilon/8n \leq \varepsilon/4n$. Plugging both bounds into Eq. (11) we get $|\mathbf{E}_{x \sim \mathcal{D}}[f(x)] - \mathbf{E}_{x \sim U_n}[f(x)]| \leq (\varepsilon/4n) \cdot \ell + (\varepsilon/8n) \cdot \ell + \varepsilon/8n \leq \varepsilon$. \square

7 Pseudorandom generators for width-3 ROBPs

In this section, we construct pseudorandom generators fooling width-3 ROBPs (3ROBPs, in short) with seed-length $\tilde{O}(\log n)$. For ordered width-3 ROBPs we can guarantee error $1/\text{poly} \log(n)$ using seed-length $\tilde{O}(\log n)$:

Theorem 1 (Main Theorem). *For any $\varepsilon > 0$, there exists a log-space explicit PRG that ε -fools width-3 ROBPs with seed-length⁵ $\tilde{O}(\log(n/\varepsilon)) + O(\log(1/\varepsilon) \cdot \log(n))$.*

Note that in comparison, even for constant $\varepsilon > 0$, the best previous generators had seed-length $O(\log^2 n)$ for ordered 3ROBPs. We also get similar improvements for unordered 3ROBPs but with worse dependence on the error ε .

Theorem 2. *For any $\varepsilon > 0$, there exists a log-space explicit PRG that ε -fools unordered width-3 ROBPs with seed-length $\tilde{O}(\log(n/\varepsilon)) + O(\text{poly}(1/\varepsilon) \cdot \log(n))$.*

7.1 Proof overview

We heavily rely on the pseudorandom restriction from Theorem 4 that assigns $p = 1/\text{poly} \log \log(n)$ of the variables while changing the acceptance probability by at most $1/\text{poly}(n)$. As a first step we assign a constant fraction of the coordinates.

Assigning most of the coordinates. The first step is rather simple: we apply iteratively $O(1/p)$ times the pseudorandom restriction from Theorem 4 to get the following analog result to Claim 6.1. The proof is the same as that of Claim 6.1 and is omitted.

Claim 7.1. *Let $\delta > 0$. For all constants $\alpha \in (0, 1)$, there is a pseudorandom restriction $\rho = (T, y)$ using $\tilde{O}(\log(n/\delta))$ random bits, changing the acceptance probability of 3ROBPs by at most δ . Furthermore, T is $(\varepsilon/n)^{\omega(1)}$ biased with marginals α and y is $(\varepsilon/n)^{\omega(1)}$ biased.*

Let B be a 3ROBP of length- n . First, we claim that after applying the pseudorandom restriction ρ in Claim 7.1, with high probability (at least $1 - \text{poly}(\varepsilon/n)$), $B|_\rho$ has a simpler structure in that between any two width-2 layers the subprogram has at most $O(\log(n/\varepsilon))$ colliding layers. Concretely, we use the following definitions.

Definition 7.2. *Given a ROBP B , we call a layer of edges colliding if either the edges marked by -1 and the edges marked by 1 collide.*

Definition 7.3. *We call a ROBP B a (w, ℓ, m) -ROBP if B can be written as $D_1 \circ \dots \circ D_m$, with each D_i being a width w ROBP with the first and last layers having at most two vertices and each D_i having at most ℓ colliding layers.*

We show that after applying the pseudorandom restriction ρ in Claim 7.1, with high probability the restricting ROBP $B|_\rho$ is a $(3, O(\log(n/\varepsilon)), m)$ -ROBPs. Now, similar to Section 6, we wish to iteratively apply Claim 7.1, making the ROBP simpler in each step. We will have one progress measures on the restricted ROBP: the maximal number of colliding layers in a subprogram (denoted ℓ). We show that the number of colliding layers reduces by a constant-factor in each iteration. To do so, we prove a structural result on $(3, \ell, m)$ -ROBPs, showing that such ROBPs can be well-approximated by $(3, \ell, C^\ell)$ -ROBPs for some constant C . This allows us to not worry about the number of sub-programs and use the number of colliding layers as a progress measure. Applying the restriction and the structure result $O(\log \log n)$ times, we end up with a ROBP where $\ell = O(\log(1/\varepsilon))$. We also show that ROBPs with few colliding layers are fooled by the INW generator. This follows from the results of [BRRY14].

⁵Henceforth, $\tilde{O}(t)$ is used to denote $O(t \cdot \text{poly} \log(t))$.

7.2 Reducing the length of $(3, \ell, m)$ -ROBPs

Here, we show that $(3, \ell, m)$ -ROBPs can be approximated by $(3, \ell, C^\ell)$ -ROBPs for some constant C . A crucial point in the analysis is that we need the approximation to hold not just under the uniform distribution but also under the pseudo-random distribution. Fortunately, we are able to do so by arguing that the *error* function detecting when our approximation is wrong is itself computable by a conjunction of negations of width 3-ROBPs with few colliding layers.

Lemma 7.4 (Main Structural Result). *For any $C \geq 1$ the following holds. Any $(3, \ell, m)$ -ROBP B can be written as $B' + E$ where B' is a $(3, \ell, C^\ell)$ -ROBP and either $E \equiv 0$ or for any x , $|E(x)| \leq F(x) = \bigwedge_{i=1}^{C^\ell} (\neg F_i(x))$ where F_i are non-zero events that can be computed by $(3, \ell, 1)$ -ROBPs on disjoint variables.*

We shall also show (in the next claim) that any non-zero event F_i that can be computed by $(3, \ell, 1)$ -ROBP, happens with probability at least $4^{-(\ell+1)}$ under the uniform distribution. Thus, $\Pr_{x \sim U_n}[\bigwedge_{i=1}^{C^\ell} (\neg F_i(x))] \leq (1 - 4^{-(\ell+1)})^{C^\ell} \leq \exp(4^{-(\ell+1)} \cdot C^\ell)$ which is doubly-exponentially small in ℓ provided that C is a large enough constant.

For any vertex v in a ROBP, we denote by p_v the probability to reach v under a uniform random assignment to the inputs.

Claim 7.5. *In a ROBP with width w and at most ℓ colliding layers, every vertex whose $p_v > 0$ has $p_v \geq 2^{-(\ell+1) \cdot (w-1)}$.*

We remark that this bound is sharp.

Proof. We prove by induction (on the length of the program) that any program with width at most w , exactly ℓ colliding layers and exactly t reachable states in the last layer, has $p_v \geq 2^{-\ell \cdot (w-1) - (t-1)}$ for any reachable vertex v . Without loss of generality all nodes in the program are reachable (otherwise, we remove vertices that aren't reachable).

Consider a program B of length n with parameters (t, ℓ, w) . Removing the last layer gives a program B' of length $n - 1$ with parameters (t', ℓ', w) . By the induction hypothesis for any v' in the last layer of B' we have $p_{v'} \geq \delta$ for $\delta := 2^{-\ell' \cdot (w-1) - (t'-1)}$.

We perform a case analysis. The following simple bound will be used in all cases. Let v be a vertex in the last layer of B . Assume that e edges enter v from vertices in the second to last layer. Then, $p_v \geq \frac{1}{2} \cdot \delta \cdot e$. In particular, since we assumed all vertices are reachable, any vertex in the last layer have $p_v \geq \delta/2$.

If $\ell' = \ell$ and $t' = t$, then the last layer of edges in B is regular, i.e., any node in the last layer in B has exactly two ingoing edges. In this case any vertex v in the last layer has $p_v \geq \frac{1}{2} \cdot \delta \cdot 2 = \delta = 2^{-\ell \cdot (w-1) - (t-1)}$.

If $\ell' = \ell$, then $t' \leq t$, since there are no collisions in the last layer of edges. Since we already handled the case $t' = t$, we may assume $t' \leq t - 1$. For any vertex v in the last layer we have $p_v \geq \delta/2 \geq \frac{1}{2} \cdot 2^{-\ell' \cdot (w-1) - (t'-1)} \geq \frac{1}{2} \cdot 2^{-\ell \cdot (w-1) - (t-2)} = 2^{-\ell \cdot (w-1) - (t-1)}$.

If $\ell' < \ell$, then we consider two sub-cases: if $t = 1$ then only one vertex is reachable in the last layer and its p_v equals 1. Otherwise, $t \geq 2$ and $t' \leq w$ thus $t' \leq t + (w - 2)$ and for any vertex v in the last layer we have $p_v \geq \delta/2 \geq \frac{1}{2} \cdot 2^{-\ell' \cdot (w-1) - (t'-1)} \geq \frac{1}{2} \cdot 2^{-(\ell-1) \cdot (w-1) - (t+(w-2)-1)} = 2^{-\ell \cdot (w-1) - (t-1)}$. \square

We say that two vertices v and v' in a ROBP are **locally-equivalent** if the 1-edges exiting v and v' reach the same vertex and the (-1) -edges exiting v and v' reach the same vertex. We say that a ROBP has **no-redundant vertices** if any vertex in the program is reachable, and there are no locally-equivalent vertices. In the following, without loss of generality we can assume that ROBPs have no-redundant vertices, because we can eliminate unreachable vertices and merge locally-equivalent vertices.

Claim 7.6 (Colliding Layers \implies Colliding). *Let B be a 3ROBP with width-2 at the start and finish, at least one colliding layer and no-redundant vertices. Let $v_{1,1}$ and $v_{1,2}$ be the two start nodes. Then, there exists a string on which the two paths from $v_{1,1}$ and $v_{1,2}$ collide.*

Proof. First consider the case that B has width 2. Then, there exists a layer i and a value $b \in \{\pm 1\}$ such that the two edges marked by b in the i -th layer collide. Any string whose i -th bit equals b results in colliding paths.

For the rest of the proof assume that B has a layer with width 3. Let V_1, \dots, V_{n+1} be the layers of vertices in B . Let i denote the index of the last layer in B with width 3. Since B has width-2 at the end, $i < n + 1$.

There are six edges between V_i and V_{i+1} : three edges marked with $x_i = -1$ and three edges marked with $x_i = 1$. Since $|V_{i+1}| = 2$, by the Pigeon-hole principle, there are two edges marked with $x_i = -1$ going to some vertex $v \in V_{i+1}$, and two edges marked with $x_i = 1$ going to some vertex $v' \in V_{i+1}$ (v' is not necessarily different from v). These two pairs of edges cannot be starting from the same two nodes in V_i since then the two nodes will be locally-equivalent. By renaming the nodes in V_i , we can assume that the two edges from $v_{i,1}, v_{i,2} \in V_i$ marked with -1 go to $v \in V_{i+1}$ and the two edges from $v_{i,2}, v_{i,3} \in V_i$ marked with 1 go to $v' \in V_{i+1}$.

Since $v_{i,2}$ is reachable, there is an input (x_1, \dots, x_{i-1}) that leads from $v_{1,1}$ or $v_{1,2}$ to $v_{i,2}$. Without loss of generality, we assume that $v_{i,2}$ is reachable from $v_{1,1}$. Let $\tilde{v} \in V_i$ be the vertex reached by following the same input (x_1, \dots, x_{i-1}) starting from the other start vertex $v_{1,2}$. If $\tilde{v} = v_{i,2}$, then we already found a collision. If $\tilde{v} = v_{i,1}$ then for the choice $x_i = -1$ the two paths defined by (x_1, \dots, x_i) starting from $v_{1,1}$ and $v_{1,2}$ collide on $v \in V_{i+1}$. Similarly, if $\tilde{v} = v_{i,3}$, then for the choice $x_i = 1$ the two paths collide on $v' \in V_{i+1}$. \square

Claim 7.7 (“First Collisions” can be detected by 3ROBPs). *Let B be a 3ROBP with 2 vertices at the first layer, denoted $v_{1,1}, v_{1,2}$. Suppose there are at most ℓ colliding layers in B and that there exists a string on which the two paths from $v_{1,1}$ and $v_{1,2}$ collide. Let u be the first vertex on which a collision can occur, and let E be the event that a collision happened on u . Then, E can be computed by another width-3 ROBP with at most ℓ -colliding layers.*

Proof. To simulate whether the paths starting from $v_{1,1}$ and $v_{1,2}$ collide at u , we consider the 3ROBP that keeps the **unordered** pair corresponding to the states of the two paths during the computation. In each layer until u , we have only states corresponding to $\{0, 1\}$, $\{0, 2\}$ or $\{1, 2\}$. When we reach the layer of u we have two states: “accept” (corresponding to a collision on u) and “reject” (corresponding to anything else). Observe that any non-colliding layer in the original program defines a non-colliding layer in the new branching program (as a permutation over a finite set also defines a permutation over unordered pairs from this set). Thus, there are at most ℓ colliding layers in the 3ROBP computing E . \square

We are now ready to prove the main structural lemma – Lemma 7.4. In the following, we consider branching programs with two initial nodes $v_{1,1}, v_{1,2}$. We interpret the value of the program on input x as its average value on the two paths starting from $v_{1,1}$ and $v_{1,2}$. That is, the program can get value 1, 0 or -1 depending on whether the two paths from $v_{1,1}$ and $v_{1,2}$ accept or not.

Throughout this section we think of the error terms as $\{0, 1\}$ -indicators (instead of the usual $\{\pm 1\}$ -notation for other Boolean functions). We shall use $A \wedge B$ and \overline{A} to denote the standard AND and negation of these Boolean values.

Lemma 7.8. *Let $B = D_1 \circ \dots \circ D_m$ be a ROBP where each D_i is a width-3 ROBP with at most 2 vertices on the first and last layers. Then, for any $j \in \{2, \dots, m\}$ we can write $B(x)$ as the sum of $(D_j \circ \dots \circ D_m)(x)$ and an error term $E(x)$, that is bounded in absolute value by $\overline{\text{FCol}_j(x)} \wedge \dots \wedge \overline{\text{FCol}_m(x)}$ where $\text{FCol}_i(x)$ denotes the event that the two paths in D_i collide on input x at the first vertex on which it is possible to collide in D_i .*

Proof. Assume without loss of generality that no layer of vertices has width-1 except for maybe the first. For $j = 2, \dots, m$, let $v_{j,1}$ and $v_{j,2}$ be the two nodes at the first layer of the subprogram D_j . If D_1 has two nodes at the first layer, then denote them by $v_{1,1}$ and $v_{1,2}$, otherwise denote the single node by $v_{1,1}$. Let x be an input to the branching program B . If the two paths defined by x from $\{v_{j,1}, v_{j,2}\}$ collide at some point, then the value of $B(x)$ equals the value of $(D_j \circ \dots \circ D_m)(x)$. If the two paths do not collide, then $(D_j \circ \dots \circ D_m)(x) = 0$, since it is the average of two paths with different outcomes, thus $E(x) = B(x) - (D_j \circ \dots \circ D_m)(x)$ is at most 1 in absolute value. Furthermore, in such a case, for all $i \in \{j, \dots, m\}$ it holds that both paths in the subprogram D_i starting from $v_{i,1}$ and $v_{i,2}$ on input x do not collide, i.e., $\text{FCol}_i(x) = 0$. Overall, we got that $B(x) = E(x) + (D_j \circ \dots \circ D_m)(x)$, and $E(x) \neq 0$, it holds that $\overline{\text{FCol}_j(x)} \wedge \dots \wedge \overline{\text{FCol}_m(x)} = 1$ (i.e., $|E(x)| \leq \overline{\text{FCol}_j(x)} \wedge \dots \wedge \overline{\text{FCol}_m(x)}$). \square

Proof of Lemma 7.4. Let B be a $(3, \ell, m)$ -ROBP $B = D_1 \circ \dots \circ D_m$. If B has no colliding layers, then there is nothing to prove since B itself is a $(3, \ell, 1)$ -ROBP. If B has colliding layers, then without loss of generality each D_i has at least one colliding layer (since otherwise we can merge subprograms with no colliding layers with their successors or predecessors). If $m \leq C^\ell$, there is nothing to prove and we can take $B' = B$ and $E = 0$. Suppose that $m > C^\ell$. Let $j = m - C^\ell + 1 > 1$. Let $B' = D_j \circ \dots \circ D_m$ and let $F(x) = \overline{\text{FCol}_j(x)} \wedge \dots \wedge \overline{\text{FCol}_m(x)}$ where $\text{FCol}_i(x)$ denotes the event that the two paths in D_i collide on input x at the first vertex on which it is possible to collide in D_i . Then, by the previous claim, we can write $B = B' + E$ where for any input x , $|E(x)| \leq F(x)$. We argue that this gives the desired decomposition. Indeed, By Claim 7.7, for $i \in \{j, \dots, m\}$ the event $\text{FCol}_i(x)$ can be computed by a $(3, \ell, 1)$ -ROBP. Further, by Claim 7.6 each D_i has a possible collision, and thus each FCol_i is a non-zero event. \square

7.3 PRGs for RBPs with few colliding layers

In this section we show that we can ε -fool ordered RBPs with at most ℓ -colliding layers with $\tilde{O}(\log(\ell/\varepsilon) \cdot \log(n))$ seed-length.

Theorem 7.9. *For any $\varepsilon > 0$, there is a log-space explicit PRG that ε -fools ordered width w -ROBPs with length n and at most ℓ colliding layers using seed length*

$$O((\log \log n + \log(1/\varepsilon) + \log(\ell) + w) \cdot \log n).$$

The above relies on the PRGs for regular branching programs and generalizations of them due to Braverman, Rao, Raz, and Yehudayoff [BRRY14]. In the following, we say that a read-once branching program B is δ -reachable if for all reachable vertices v in B we have $p_v(B) \geq \delta$, where

$$p_v(B) := \Pr_{x \sim U_n} [\text{reaching } v \text{ on the walk on } B \text{ defined by } x].$$

We start by quoting a result by Braverman et al. [BRRY14].

Theorem 7.10 ([BRRY14]). *There is a log-space explicit PRG that ε -fools all δ -reachable ROBPs of length- n and width- w using seed length*

$$O(\log \log n + \log(1/\varepsilon) + \log(1/\delta) + \log(w)) \cdot \log n.$$

Next, we reduce the task of fooling ROBPs with at most ℓ -colliding layers to the task of fooling δ -reachable ROBPs. The reduction is similar to that in [CHRT17]. The main difference is that we simulate a ROBP with width w by a δ -reachable ROBP of width $w + 1$ by adding a new sink state that should be thought of as “immediate stop”. This change seems essential in our case, and the reduction from [CHRT17] does not seem to satisfy the necessary properties here.

Lemma 7.11. *Let $\delta \leq 2^{-(w-1)}$. Let \mathcal{D} be a distribution on $\{\pm 1\}^n$ that ε -fools all δ -reachable ROBPs of length n and width $w + 1$. Then, \mathcal{D} also fools width- w ROBPs with at most ℓ colliding layers with error at most $(\ell w + 1) \cdot \varepsilon + (2^w w \ell) \cdot \delta$.*

Proof. Let \mathcal{D} be a distribution on $\{\pm 1\}^n$ that ε -fools all δ -reachable ROBPs of length- n and width- w . The first observation is that \mathcal{D} also fools prefixes of these programs. This reason is simple: to simulate the prefix of length- k of a δ -reachable ROBP B , one can just reroute the last $n - k$ layers of edges in B so that they would “do nothing”, i.e. that they would be the identity transformation regardless of the values of x_{k+1}, \dots, x_n .

Let B be a length n width- w ROBP with at most ℓ colliding layers. Next, we introduce B' , a δ -reachable ROBP of length- n and width- $(w + 1)$, that would help bound the difference between

$$B(U_n) := \Pr_{x \sim U_n} [B(x) = 1] \quad \text{and} \quad B(\mathcal{D}) := \Pr_{x \sim \mathcal{D}} [B(x) = 1],$$

where U_n is the uniform distribution over $\{\pm 1\}^n$. Let B' be the the following modified version of B . To construct B' we consider a sequence of $\ell + 1$ branching programs B_0, \dots, B_ℓ where $B_0 = B$ and $B' = B_\ell$. Let i_1, \dots, i_ℓ be the colliding layers in B . For $j = 1, \dots, \ell$ we take B_j to be B_{j-1} except we may reroute some of the edges in the i_j -th layer. We explain the rerouting procedure. For $j = 1, \dots, \ell$ we calculate the probability to reach vertices in layer V_{i_j} of B_{j-1} . If some vertex v in the i_j -th layer has probability smaller than $2^{w-1} \cdot \delta$, then we reroute the two edges going from the vertex v to go to “immediate stop”. We denote by V_{small} the set of vertices for which we rerouted the outgoing edges from them.

First, we claim that any reachable vertex v in B_ℓ has $p_v \geq \delta$. Let $i_{\ell+1} = n + 1$ for convenience. We apply induction and show that for $j = 0, 1, \dots, \ell$ any vertex reachable by B_j in layers $1, \dots, i_{j+1}$ has $p_v \geq \delta$. The base case holds because up to layer i_1 the branching program has no colliding layers and we may apply Claim 7.5 to get that $p_v \geq 2^{-(w-1)} \geq \delta$. To apply induction assume the claim holds for B_{j-1} and show that it holds for B_j . The claim obviously holds for all vertices in layers $1, \dots, i_j$ in B_j since we didn't change any edge in those layers going from B_{j-1} to B_j . Let v be a reachable vertex in layer i where $i_j < i \leq i_{j+1}$ in B_j . It means that there is a vertex in v' with $p_{v'}(B_j) \geq 2^{w-1} \cdot \delta$ in the i_j -th layer of B_j (and also in B_{j-1}) and a path going from v' to v . Looking at the subprogram from v' to v we note that this is a subprogram with no colliding edges (only the first layer has the potential to be colliding, but in a ROBP the first layer can never be colliding as there is only one edge marked by (-1) and only one edge marked by -1). By Claim 7.5 the probability to get from v' to v is at least $2^{-(w-1)}$. Thus, the probability to reach v is at least $p_{v'}(B_j) \cdot \Pr[\text{reach } v | \text{reached } v'] \geq 2^{w-1} \cdot \delta \cdot 2^{-(w-1)} = \delta$.

Next, we bound $|B(U_n) - B(\mathcal{D})|$ by using the triangle inequality

$$|B(U_n) - B(\mathcal{D})| \leq |B(U_n) - B'(U_n)| + |B'(U_n) - B'(\mathcal{D})| + |B'(\mathcal{D}) - B(\mathcal{D})| \quad (12)$$

and bounding each of the three terms separately.

1. The first term is bounded by the probability of reaching one of the nodes in V_{small} in B' when taking a uniform random walk. This follows since if the path defined by x didn't pass through V_{small} then we would end up with the same node in both B and B' (since no rerouting affected the path). Each vertex v in V_{small} has $p_v(B') < 2^{w-1} \cdot \delta$. By union bound, the probability to pass through V_{small} is at most $|V_{\text{small}}| \cdot 2^{w-1} \cdot \delta$.
2. The second term is at most ε since the program B' is δ -reachable.
3. Similarly to the first term, the third term is bounded by the probability of reaching one of the nodes in V_{small} in B' when taking a walk sampled by \mathcal{D} .

$$\begin{aligned} |B'(\mathcal{D}) - B(\mathcal{D})| &\leq \Pr_{x \sim \mathcal{D}}[\text{reaching } V_{\text{small}} \text{ on the walk on } B' \text{ defined by } x] \\ &\leq \sum_{v \in V_{\text{small}}} \Pr_{x \sim \mathcal{D}}[\text{reaching } v \text{ on the walk on } B' \text{ defined by } x] \end{aligned}$$

However since \mathcal{D} is pseudorandom for prefixes of B' , for each $v \in V_{\text{small}}$ the probability of reaching v when walking according to \mathcal{D} is ε -close to the probability of reaching v when walking according to U_n .

$$\begin{aligned} |B'(\mathcal{D}) - B(\mathcal{D})| &\leq \sum_{v \in V_{\text{small}}} (\varepsilon + \Pr_{x \sim U_n}[\text{reaching } v \text{ on the walk on } B' \text{ defined by } x]) \\ &= \sum_{v \in V_{\text{small}}} (\varepsilon + p_v(B')) \leq |V_{\text{small}}| \cdot (\varepsilon + 2^{w-1} \delta) \end{aligned}$$

Summing the upper bound on the three terms in Eq. (12) gives:

$$|B(U_n) - B(\mathcal{D})| \leq |V_{\text{small}}| \cdot (\varepsilon + 2^w \delta) + \varepsilon \leq \ell w \cdot (\varepsilon + 2^w \delta) + \varepsilon. \quad \square$$

Proof of Theorem 7.9. Take $\varepsilon' = \varepsilon/(2(\ell w + 1))$ and $\delta = \varepsilon'/2^w$. Take the generator from Theorem 7.10 with parameters δ and ε' . Applying Lemma 7.11, the error of this generator on the class of ROBPs with width w length n and at most ℓ colliding layers is at most $(\ell w + 1) \cdot \varepsilon' + (2^w \cdot w \cdot \ell) \cdot \delta \leq \varepsilon/2 + \varepsilon/2 = \varepsilon$. By Theorem 7.10, its seed length is

$$O(\log \log n + \log(1/\varepsilon') + \log(1/\delta) + \log(w)) \cdot \log(n)$$

which is at most $O(\log \log n + \log(1/\varepsilon) + \log(\ell) + w) \cdot \log(n)$. \square

7.4 Proof of Theorem 1

We are now ready to prove our main result on fooling 3ROBPs. Our generator is obtained by applying Claim 7.1 iteratively $O(\log \log n)$ times and then using a PRG fooling 3ROBPs with at most $O(\text{poly}(1/\varepsilon))$ colliding layers as in Theorem 7.9. The intuition is as follows.

Let B be a 3ROBP and let ρ_0 be a pseudorandom restriction as in Claim 7.1. We first show that with probability at least $1 - \varepsilon/n$ over ρ_0 , $B^0 = B|_{\rho_0}$ is a $(3, \ell_0, m)$ -ROBP for $\ell_0 = O(\log(n/\varepsilon))$. Let $B^0 = D_1^0 \circ \dots \circ D_m^0$ where each D_i^0 has at most ℓ_0 colliding layers and begins and ends with width two layers. Let ρ_1 be an independent pseudo-random restriction as in Claim 7.1. Then $B^1 \equiv B^0|_{\rho_1} = D_1^0|_{\rho_1} \circ \dots \circ D_m^0|_{\rho_1}$ and it is easy to check that with probability at least $1 - 2^{-\Omega(\ell_0)}$, each $D_i^0|_{\rho_1}$ has at most $\ell_0/2$ colliding layers. Ideally, we would like to apply a union bound over the different D_i^0 and conclude that B^1 is a $(3, \ell_0/2, m)$ -ROBP. In the first step, this approach works since $m \leq C^{\ell_0}$ for a large enough constant C (by the definition on ℓ_0), and we can afford a union bound. We get that with probability at least $1 - 2^{-\Omega(\ell_0)}$, B^1 is a $(3, \ell_0/2, m_1)$ -ROBP (for some $m_1 \leq m$). Continuing this process by induction, at step i we have that B^i is a $(3, \ell_0/2^i, m_i)$ -ROBP. To carry the union bound in the i -th step we need $m_i \leq C^{\ell_0/2^i}$, however m_i could be much larger than that. Nevertheless, we know that we can always approximate B^i with a $(3, \ell_0/2^i, C^{\ell_0/2^i})$ -ROBP by Lemma 7.4. This approximation allows us to apply the union bound and conclude that the number of colliding layers in each block decreases by a factor of 2. We iterate this approach until the maximal number of colliding layers in a subprogram is at most $O(\log(1/\varepsilon))$, and then use the PRG from Theorem 7.9.

To carry the induction forward as outlined above, we need the following lemma that shows that the error terms simplify as well under the pseudorandom restrictions.

Lemma 7.12. *For any constant $C \geq 20$, there exists $\alpha \in (0, 1)$ such that the following holds. Let $\ell, n \in \mathbb{N}$ be sufficiently large and $m = C^\ell \leq n$. Let $F = \overline{\text{FCol}}_1 \wedge \dots \wedge \overline{\text{FCol}}_m$ where $\text{FCol}_i(x)$ are non-zero events on disjoint variables computed by $(3, \ell, 1)$ -ROBPs. Let ρ be a pseudorandom restriction as in Claim 7.1 with parameter α and error parameter $\delta \leq 1/n^5$. Then, with probability at least $1 - 2C^{-\ell/2}$, we have $F|_\rho \leq \overline{\text{FCol}}'_i \wedge \dots \wedge \overline{\text{FCol}}'_{\sqrt{m}}$ where $\text{FCol}'_i(x)$ are non-zero events on disjoint variables computed by $(3, \ell/2, 1)$ -ROBPs.*

Proof. First, we show that with high probability, each FCol_i has at most $\ell/2$ colliding layers under the pseudo-random restriction. To see it, note that any colliding layer that is restricted can be either:

- Assigned to a value that reduces the width of the original program to 2, and thus the width of FCol_i to 1, in which case any previous layer in FCol_i is not affecting its value.

- Assigned to a value that applies a permutation on the states of the program, thus reducing the number of colliding layers.

In either case, if k colliding layers are unassigned, then $\text{FCol}_i|_\rho$ can be computed by a 3ROBP with at most k colliding layers. By Claim 7.1 the probability that less than $\ell/2$ colliding layers are unassigned is at least $1 - \binom{\ell}{\ell/2} \cdot \alpha^{\ell/2} - (\varepsilon/n)^{\omega(1)} \geq 1 - 2^\ell \alpha^{\ell/2}$. Taking a union bound over the C^ℓ functions $\{\text{FCol}_i\}_{i=1}^m$ we get that with probability at least $1 - (2C)^\ell \cdot \alpha^{\ell/2} \geq 1 - C^{-\ell/2}$ (for a suitable choice of α) all functions $\{\text{FCol}_i|_\rho\}_{i=1}^m$ can be computed by 3ROBPs with at most $\ell/2$ colliding layers.

We move to show that with high probability at least $C^{\ell/2}$ of the functions $\text{FCol}_i|_\rho$ are non-zero. We apply the second moment method. Denote by $p_i = \Pr_{z \sim U}[\text{FCol}_i(z)]$ for $i \in [m]$. Let A_1, \dots, A_m be the events that $\{(\text{FCol}_i|_\rho)(y) = 1\}_{i=1}^m$ respectively, where ρ is the pseudorandom restriction from Claim 7.1 and y is uniformly distributed. By Claim 7.1

$$\Pr[A_i] = \Pr_{\rho, y \sim U}[(\text{FCol}_i|_\rho)(y) = 1] = \mathbf{E}_{z \sim U}[\text{FCol}_i(z)] \pm \delta = p_i \pm \delta,$$

and by the next lemma, whose proof is deferred to Appendix A.3, we get

$$\begin{aligned} \Pr[A_i \wedge A_j] &= \Pr_{\rho, y}[(\text{FCol}_i|_\rho)(y) \wedge (\text{FCol}_j|_\rho)(y) = 1] \\ &= \mathbf{E}_{z \sim U}[\text{FCol}_i(z) \wedge \text{FCol}_j(z)] \pm \delta \cdot (n+1)^2 = p_i p_j \pm \delta \cdot (n+1)^2. \end{aligned}$$

Lemma 7.13. *Let f_1, \dots, f_k be 3ROBPs on disjoint sets of variables of $[n]$. Let $H : \{\pm 1\}^k \rightarrow \{\pm 1\}$ be any Boolean function. Then, $f = H(f_1, f_2, \dots, f_k)$ is $\delta \cdot (n+1)^k$ -fooled by the pseudorandom restriction in Claim 7.1.*

Thus, the covariance of the two events A_i and A_j is at most $\delta' := \delta((n+1)^2 + 3)$. Denote by $M = \sum_{i=1}^m p_i$. By Claim 7.5 we have that $M \geq 4^{-(\ell+1)} \cdot m \geq 8C^{\ell/2}$ (since $m = C^\ell$, $C \geq 20$ and ℓ is sufficiently large). Let $Z = \sum_{i=1}^m \mathbb{1}_{A_i}$. Then, $\mathbf{E}[Z] \geq M - \delta m$ and by Chebyshev's inequality

$$\Pr[Z < M/2] \leq \Pr[|Z - \mathbf{E}[Z]| \geq M/2 - \delta m] \leq \frac{\mathbf{Var}[Z]}{(M/2 - \delta m)^2}.$$

We bound

$$\mathbf{Var}[Z] = \sum_i \mathbf{Var}[A_i] + \sum_{i \neq j} \mathbf{Cov}[A_i, A_j] \leq \sum_{i=1}^m (p_i + \delta) + \sum_{i \neq j} \delta' \leq M + m^2 \cdot \delta',$$

which gives $\Pr[Z < M/2] \leq (M + \delta' m^2) / (M/2 - \delta m)^2 \leq (M+1) / (M/2 - 1)^2 \leq 8/M \leq C^{-\ell/2}$ using $\delta \leq 1/n^5$. In the complement event, at least $M/2 \geq C^{\ell/2}$ of the events A_1, \dots, A_m occur, and in particular at least $C^{\ell/2}$ of the restricted functions $\{\text{FCol}_i|_\rho\}_{i=1}^m$ are non-zero.

Suppose that at least $C^{\ell/2}$ of the restricted functions $\{\text{FCol}_i|_\rho\}_{i=1}^m$ are non-zero, and that all restricted functions has at most $\ell/2$ colliding layers. By the above analysis this happens with probability at least $1 - 2C^{-\ell/2}$. Under this assumption, we can reduce the number of functions to be exactly $\sqrt{m} = C^{\ell/2}$, resulting in an upper bound on $F|_\rho$ which we denote by $\overline{\text{FCol}}'_1(x) \wedge \dots \wedge \overline{\text{FCol}}'_{\sqrt{m}}(x)$. \square

We are now ready to prove the main theorem, Theorem 1.

Proof of Theorem 1. Let $C \geq 20$. Let $\alpha \in (0, 1)$ be a constant to be chosen later. Let $\ell_0 = O(\log(n/\varepsilon))$. Let k be a parameter to be chosen later and let $\ell_i = \ell_0/2^i$ for $1 \leq i \leq k$.

Our generator is as follows. First choose $\rho_0, \rho_1, \dots, \rho_k$ independent pseudo-random restrictions as in Claim 7.1 with parameter α and $\delta = \varepsilon/n^{10}$. After iteratively applying the restrictions $\rho_0, \rho_1, \dots, \rho_k$, we set the remaining bits using the generator from Theorem 7.9 for a parameter $\ell = \ell_k \cdot C^{\ell_k}$ and error parameter ε' to be chosen later. Let Y be the output distribution of the generator.

Let $B^0 = B|_{\rho_0}$. We first claim that B^0 is a $(3, \ell_0, m)$ -ROBP with high probability. In the following let X be uniformly random over $\{\pm 1\}^n$.

Claim 7.14. *With probability at least $1 - \varepsilon/n$, $B|_{\rho_0}$ is a $(3, \ell_0, m)$ -ROBP and $\mathbf{E}_{\rho_0, X}[B|_{\rho_0}(X)] = \mathbf{E}_X[B(X)] \pm \delta$.*

For $0 \leq i \leq k$, let $\rho^i \triangleq \rho_0 \circ \dots \circ \rho_i$. We will show the following claim by induction on i .

Claim 7.15. *For $0 \leq i \leq k$, with probability at least $1 - \frac{\varepsilon}{n} - 4i \cdot C^{-\ell_i}$, $B|_{\rho^i}$ can be written as $B^i + E^0 + E^1 + \dots + E^i$ where B^i is a $(3, \ell_i, C^{\ell_i})$ -ROBP and the error terms E^j for $0 \leq j \leq i$ satisfy: Either $E_j \equiv 0$ or $|E^j(x)| \leq F^j(x)$ with $F^j(x) = \bigwedge_{h=1}^{m_j} (\neg F_h^j(x))$ where F_h^j are non-zero events computed by $(3, \ell_i, 1)$ -ROBPs on disjoint sets of variables and $m_j = C^{\ell_i}$.*

Furthermore, $\mathbf{E}_{\rho^i, X}[B|_{\rho^i}(X)] = \mathbf{E}_X[B(X)] \pm (i+1)\delta$.

A crucial point in the above is that the functions F^0, \dots, F^i bounding the error terms are conjunctions of negations of $(3, \ell_i, 1)$ -ROBPs and there exactly C^{ℓ_i} in each of them.

Proof. For $i = 0$, the claim follows immediately by applying Lemma 7.4 to $B|_{\rho_0}$. Now, suppose the claim is true for i . Suppose, we can write $B|_{\rho^i} = B^i + \mathcal{E}^i$, where $\mathcal{E}^i = E^0 + E^1 + \dots + E^i$ as in the claim. By the induction hypothesis, this happens with probability at least $1 - \frac{\varepsilon}{n} - 4i \cdot C^{-\ell_i}$.

Clearly, $B|_{\rho^{i+1}} = B^i|_{\rho^{i+1}} + \mathcal{E}^i|_{\rho^{i+1}}$. Let $B^i = D_1 \circ \dots \circ D_{m'}$ be a decomposition where each D_j has at most ℓ_i colliding layers, starts and ends with width-2 layers and $m' \leq C^{\ell_i}$.

Now, observe that as each D_j has at most ℓ_i colliding layers, the probability that at least $\ell_i/2$ of these colliding layers are unfixed under ρ_{i+1} is at most $\binom{\ell_i}{\ell_i/2} \cdot (\alpha^{\ell_i/2} + (\varepsilon/n)^{\omega(1)}) \leq 2^{\ell_i} \alpha^{\ell_i/2}$ by Claim 7.1. Thus, by a union bound over $1 \leq j \leq m'$, with probability at least $1 - 2^{\ell_i} \alpha^{\ell_i/2} \cdot C^{\ell_i} \geq 1 - C^{-\ell_i}$ (for a suitable choice of α), over ρ_{i+1} , $B^i|_{\rho_{i+1}}$ is a $(3, \ell_i/2, C^{\ell_i})$ -ROBP. Now, conditioning on this event, by Lemma 7.4, we can write $B^i|_{\rho_{i+1}}$ as $B^{i+1} + E^{i+1}$, where B^{i+1} is a $(3, \ell_{i+1}, C^{\ell_{i+1}})$ -ROBP and E^{i+1} satisfies the conditions of the claim. Thus, with probability at least $1 - \frac{\varepsilon}{n} - 4i \cdot C^{-\ell_i} - C^{-\ell_i}$,

$$\begin{aligned} B|_{\rho^{i+1}} &= B^i|_{\rho_{i+1}} + \mathcal{E}^i|_{\rho_{i+1}} \\ &= B^{i+1} + \mathcal{E}^i|_{\rho_{i+1}} + E^{i+1}, \end{aligned}$$

where B^{i+1} , and E^{i+1} satisfy the conditions of the claim.

We just need to argue that $\mathcal{E}^i|_{\rho_{i+1}}$ can be written in the requisite form. To this end, note that for $0 \leq j \leq i$, $|E^j|_{\rho_{i+1}} \leq F^j|_{\rho_{i+1}}$. By the induction hypothesis, we either have $E^j \equiv 0$ or we can write $|E^j| \leq F^j = \bigwedge_{h=1}^{m_j} (\neg F_h^j(x))$ where F_h^j are $(3, \ell_i, 1)$ -ROBPs on disjoint sets of variables and $m_j = C^{\ell_i}$. We can now apply Lemma 7.12 to conclude that with probability

at least $1 - 2C^{-\ell_i/2}$, we can write $F^j|_{\rho_{i+1}} = \bigwedge_{h=1}^{m'_j} (\neg H_h^j(x))$ where H_h^j are non-zero events computed by $(3, \ell_i/2, 1)$ -ROBPs on disjoint sets of variables and $m'_j = C^{\ell_i/2}$. This satisfies the constraints of the claim.

Adding up the failure probabilities over the choice of ρ_{i+1} , we get the desired decomposition for $i + 1$ with probability at least

$$1 - \frac{\varepsilon}{n} - 4i \cdot C^{-\ell_i} - C^{-\ell_i} - (i + 1) \cdot 2C^{-\ell_i/2} \geq 1 - \frac{\varepsilon}{n} - 4(i + 1)C^{-\ell_{i+1}}.$$

(since $2C^{-\ell_i} \leq C^{-\ell_{i+1}}$). The furthermore part follows immediately from Claim 7.1. The claim now follows by induction. \square

We are now ready to prove the theorem. By the above claim, we have that with probability at least $1 - \frac{\varepsilon}{n} - 4kC^{-\ell_k}$ over the choice of $\rho_0, \rho_1, \dots, \rho_k$, we can write

$$B|_{\rho^k} = B^k + E^0 + \dots + E^k,$$

where B^k is a $(3, \ell_k, C^{\ell_k})$ -ROBP and E^0, \dots, E^k can be bounded by functions F^0, \dots, F^k that are conjunctions of negations of C^{ℓ_k} non-zero events computed by $(3, \ell_k, 1)$ -ROBPs.

Note that each such F^j can be written as a width-4 ROBP, say H^j , by adding an additional layer to compute the conjunction and that the number of collisions in the width 4 ROBP is at most $\ell_k \cdot C^{\ell_k}$. Therefore, if we let Y be the output distribution of the generator from Theorem 7.9 with $\ell = \ell_k \cdot C^{\ell_k}$ and error parameter ε' , we get that for all $0 \leq j \leq k$, and X uniformly random over $\{\pm 1\}^n$,

$$\mathbf{E}[B^k(X)] = \mathbf{E}[B^k(Y)] \pm \varepsilon'$$

$$\mathbf{E}[|E^j(Y)|] \leq \mathbf{E}[H^j(Y)] \leq \mathbf{E}[H^j(X)] + \varepsilon' \leq (1 - 4^{-(\ell_k+1)})^{C^{\ell_k}} + \varepsilon'$$

where we used Claim 7.5 to bound $\mathbf{E}[H^j(X)]$. Since $C \geq 20$, $\mathbf{E}[|E^j(Y)|] \leq \exp(-2^{\ell_k}) + \varepsilon'$.

Combining the above inequalities we get that with probability at least $1 - \frac{\varepsilon}{n} - 4kC^{-\ell_k}$ over the choice of $\rho_0, \rho_1, \dots, \rho_k$

$$\left| \mathbf{E}_X[B|_{\rho^k}(X)] - \mathbf{E}_Y[B|_{\rho^k}(Y)] \right| \leq \varepsilon' + (k + 1) \cdot (\exp(-2^{\ell_k}) + \varepsilon').$$

Finally, as we also have that

$$\left| \mathbf{E}_{\rho_0, \dots, \rho_k} [B|_{\rho^k}(X)] - \mathbf{E}[B(X)] \right| \leq (k + 1) \cdot \delta,$$

we get

$$\left| \mathbf{E}_{\rho_0, \dots, \rho_k} [B|_{\rho^k}(Y)] - \mathbf{E}[B(X)] \right| \leq ((k + 1) \cdot \delta) + \left(\frac{\varepsilon}{n} + 4kC^{-\ell_k} \right) + (\varepsilon' + (k + 1) \cdot (\exp(-2^{\ell_k}) + \varepsilon')).$$

To get $\ell_k = \log \log \log(n) + \log(1/\varepsilon)$ we set $k = \log(\ell_0/\ell_k) = O(\log \log n)$. Furthermore, setting $\delta = \varepsilon/n^{10}$ and $\varepsilon' = \varepsilon/4k$, the above error bound becomes

$$\left| \mathbf{E}_{\rho_0, \dots, \rho_k} [B|_{\rho^k}(Y)] - \mathbf{E}[B(X)] \right| \leq \varepsilon.$$

Finally, we estimate the seed-length of our generator. Choosing the random restrictions takes $\tilde{O}(\log(n/\delta)) = \tilde{O}(\log(n/\varepsilon))$ random bits. Sampling Y requires seed-length

$$O(\log \log n + \log(1/\varepsilon') + \log(\ell_k \cdot C^{\ell_k}) + 4) \cdot \log n = O(\log \log n + \log(1/\varepsilon)) \cdot \log n.$$

Thus, the final seed-length is $\tilde{O}(\log(n/\varepsilon)) + O(\log(1/\varepsilon)(\log n))$. The theorem follows. \square

7.5 Proof of Claim 7.14

Claim. *With probability at least $1 - \varepsilon/n$, $B|_{\rho_0}$ is a $(3, \ell_0, m)$ -ROBP and $\mathbf{E}_{\rho_0, X}[B|_{\rho_0}(X)] = \mathbf{E}_X[B(X)] \pm \delta$.*

Proof. The second part follows from Claim 7.1. We are left to prove the first part.

Let $\rho_0 = (T, y)$ be the pseudorandom restriction, where $T \subseteq [n]$ and $y \in \{\pm 1\}^{[n] \setminus T}$. Assume there are L colliding layers in B and let i_1, i_2, \dots, i_L be their indices. For $j \in [L]$, call a layer i_j “good” under the choice of (T, y) if $i_j \in [n] \setminus T$ and the edges in the i_j -layer of B marked by y_{i_j} collide.

For $j \in \{1, \dots, L - \ell_0 + 1\}$ let \mathcal{E}_j be the event that none of layers $\{i_j, i_{j+1}, \dots, i_{j+(\ell_0-1)}\}$ is good. Recall that T is sampled from a $(\varepsilon/n)^{\omega(1)}$ -biased distribution with marginals α , and y is sampled from a $(\varepsilon/n)^{\omega(1)}$ -biased distribution. For \mathcal{E}_j to happen, we must have a partition $S_1 \cup S_2 = \{j, j+1, \dots, j+\ell_0-1\}$ such that all layers $i_{j'}$ for $j' \in S_1$ are in T and all layers $i_{j''}$ for $j'' \in S_2$ are in $[n] \setminus T$ but the edges marked by $y_{i_{j''}}$ in the $i_{j''}$ -th layer do not collide. For any fixed j and fixed partition $S_1 \cup S_2 = \{j, j+1, \dots, j+\ell_0-1\}$, the above event happens with probability at most

$$(\alpha^{|S_1|} + (\varepsilon/n)^{\omega(1)}) \cdot (2^{-|S_2|} + (\varepsilon/n)^{\omega(1)}) \leq 2 \cdot \alpha^{|S_1|} \cdot 2^{-|S_2|} = 2 \cdot \alpha^{|S_1|} \cdot 2^{-(\ell_0 - |S_1|)}$$

(using $|S_1| + |S_2| = \ell_0 = O(\log(n/\varepsilon))$). Overall,

$$\Pr[\mathcal{E}_j] \leq \sum_{S_1 \subseteq \{j, \dots, j+\ell_0-1\}} (2 \cdot \alpha^{|S_1|} \cdot 2^{-(\ell_0 - |S_1|)}) = 2 \cdot \left(\frac{1}{2} + \alpha\right)^{\ell_0} \leq \varepsilon/n^2$$

assuming $\alpha > 0$ is a sufficiently small constant and $\ell_0 = c \log(n/\varepsilon)$ for a sufficiently large constant $c > 0$. By the union bound,

$$\Pr[\mathcal{E}_1 \vee \mathcal{E}_2 \vee \dots \vee \mathcal{E}_{L-\ell_0+1}] \leq (L - \ell_0 + 1) \cdot \varepsilon/n^2 \leq \varepsilon/n.$$

Under the event that all \mathcal{E}_j are false, we get that $B|_{\rho}$ can be written as $D_1 \circ \dots \circ D_m$ where each D_i is a width-3 ROBP with at most ℓ_0 colliding layers and at most 2 vertices on the first and last layer. \square

7.6 Pseudorandom generator for unordered 3ROBPs

In this section, using the recent generator of Chattopadhyay et al. [CHHL18], and a Fourier bound by Steinke, Vadhan and Wan [SVW14], we show that we can also handle unordered 3ROBPs, thus proving Theorem 2.

Lemma 7.16 (Lemma 3.14 [SVW14]). *Let $\ell \in \mathbb{N}$ and let B be a width- w ROBP with at most ℓ colliding layers. Then, for all $k = 1, \dots, n$ it holds that $L_{1,k}(f) \leq O(w^3 \cdot \ell)^k$.*

Theorem 7.17 (Theorem 4.5 [CHHL18]). *Let \mathcal{F} be a family of n -variate Boolean functions closed under restrictions. Assume that for all $f \in \mathcal{F}$ for all $k = 1, \dots, n$, $L_{1,k}(f) \leq a \cdot b^k$. Then, for any $\varepsilon > 0$, there exists a log-space explicit PRG which fools \mathcal{F} with error ε , whose seed length is $O(\log(n/\varepsilon) \cdot (\log \log(n) + \log(a/\varepsilon)) \cdot b^2)$.*

Corollary 7.18. *There is a log-space explicit PRG that ε -fools unordered ROBPs with width w , length n and at most ℓ colliding layers using seed length*

$$O(\log(n/\varepsilon) \cdot (\log \log(n) + \log(1/\varepsilon)) \cdot w^6 \ell^2)$$

Proof of Theorem 2. The proof is essentially the same as that of Theorem 1, where instead of using the generator from Theorem 7.9 to set the bits after the pseudorandom restrictions, we use the generator from the above corollary. The final seed-length has a worse dependence on ε as we need to set $\ell = C^{\log(1/\varepsilon) + \log \log \log(n)} = \text{poly}(1/\varepsilon) \cdot \text{poly} \log \log(n)$ in Cor. 7.18. \square

7.7 Pseudorandom generator for locally-monotone width-3 ROBPs

In this section, we construct pseudorandom generators that ε -fool unordered locally-monotone 3ROBPs with seed-length $\tilde{O}(\log(n/\varepsilon))$. Our dependency on ε is much better than in Section 7, and we get nearly logarithmic (in n) seed-length even for error $\varepsilon = 1/\text{poly}(n)$.

We remark that read-once CNFs and read-once DNFs are special cases of locally-monotone 3ROBPs, hence our result extends the result of GMRTV [GMR⁺12] that constructs an ε -PRG for the former classes of functions using seed length $\tilde{O}(\log(n/\varepsilon))$.

Theorem 7. *For any $\varepsilon > 0$, there exists a log-space explicit PRG that ε -fools unordered locally-monotone 3ROBPs with seed-length $\tilde{O}(\log(n/\varepsilon))$.*

Proof. The pseudorandom generator samples a string in $\{\pm 1\}^n$ as follows:

1. Apply the pseudorandom restriction $\rho_0 = (T, y)$ from Claim 7.1 with $\alpha = \frac{1}{10}$ and $\delta = \frac{\varepsilon}{3}$.
2. Assign the coordinates in T using the pseudorandom generator $\mathbf{GXOR}(T, 3, \ell_0, \varepsilon/3n)$ with $\ell_0 = O(\log(n/\varepsilon))$.

It is clear that the seed-length is $\tilde{O}(\log(n/\varepsilon))$ by Claims 7.1 and 6.8.

Next, we show that the generator ε -fools unordered locally monotone 3ROBPs. By Claim 7.14, with probability at least $1 - \varepsilon/n$, $B|_{\rho_0}$ is a $(3, \ell_0, m)$ -ROBP for $\ell_0 = O(\log(n/\varepsilon))$. Furthermore, $\mathbf{E}_{\rho_0, x \sim U} [B|_{\rho_0}(x)] = \mathbf{E}[B] \pm \varepsilon/3$. Now, observe that since B is locally monotone, for each layer i either $E_{i,1}$ or $E_{i,-1}$ has at most 2 end-vertices (Lemma 3.9). By the proof of Claim 7.14 we get that with probability at least $1 - \varepsilon/n$, $B|_{\rho_0}$ is of the form $D_1 \circ \dots \circ D_m$ where each D_i has at most ℓ_0 layers, and at most 2 vertices on the first and last layers. In other words, in this case not only does each subprogram have a few colliding layers, it actually has a few layers! Whenever $B|_{\rho_0}$ is of the above form we say that it **simplified** under the restriction.

Thm. 4.2 states that whenever $B|_{\rho_0}$ simplified, it can be written as $\sum_{\alpha \in \{0,1\}^m} c_\alpha \prod_{i=1}^m D_{i,\alpha_i}$ where D_{i,α_i} are subprograms of D_i , and $\sum_{\alpha \in \{0,1\}^m} |c_\alpha| \leq m \leq n$. By Claim 6.7, the distribution $\mathcal{D} = \mathbf{GXOR}(T, 3, \ell_0, \varepsilon/3n)$ fools $\prod_{i=1}^m D_{i,\alpha_i}$ with error at most $\varepsilon/3n$, for all $\alpha \in \{0,1\}^m$. Thus, it fools $B|_{\rho_0}$ with error at most $(\varepsilon/3n) \cdot \sum_{\alpha} |c_\alpha| \leq \varepsilon/3$. Overall, we get

$$\begin{aligned} \left| \mathbf{E}_{\rho_0=(T,y), x \sim \mathcal{D}} [B(\text{Sel}_T(x, y))] - \mathbf{E}[B] \right| &\leq \left| \mathbf{E}_{\substack{\rho_0=(T,y), \\ x \sim U_T}} [B|_{\rho_0}(x)] - \mathbf{E}[B] \right| \\ &\quad + \mathbf{Pr}[B|_{\rho_0} \text{ did not simplify}] + \varepsilon/3 \\ &\leq \varepsilon/3 + \varepsilon/n + \varepsilon/3 \leq \varepsilon. \end{aligned} \quad \square$$

References

- [ABN⁺92] N. Alon, J. Bruck, J. Naor, M. Naor, and R. M. Roth. Construction of asymptotically good low-rate error-correcting codes through pseudo-random graphs. *IEEE Trans. Information Theory*, 38(2):509–516, 1992.
- [AGHP92] N. Alon, O. Goldreich, J. Håstad, and R. Peralta. Simple construction of almost k -wise independent random variables. *Random Structures and Algorithms*, 3(3):289–304, 1992.
- [AW85] M. Ajtai and A. Wigderson. Deterministic simulation of probabilistic constant depth circuits. In *FOCS*, pages 11–19, 1985.
- [BDVY13] A. Bogdanov, Z. Dvir, E. Verbin, and A. Yehudayoff. Pseudorandomness for width-2 branching programs. *Theory of Computing*, 9:283–293, 2013.
- [Bon70] A. Bonami. Étude des coefficients de fourier des fonctions de l_p (g). 1970.
- [BRRY14] M. Braverman, A. Rao, R. Raz, and A. Yehudayoff. Pseudorandom generators for regular branching programs. *SIAM J. Comput.*, 43(3):973–986, 2014.
- [BT13] A. Ben-Aroya and A. Ta-Shma. Constructing small-bias sets from algebraic-geometric codes. *Theory of Computing*, 9:253–272, 2013.
- [BV10] J. Brody and E. Verbin. The coin problem and pseudorandomness for branching programs. In *Proceedings of the 51st annual FOCS*, pages 30–39, 2010.
- [CGR14] G. Cohen, A. Ganor, and R. Raz. Two sides of the coin problem. In *APPROX-RANDOM*, pages 618–629, 2014.
- [CHHL18] E. Chattopadhyay, P. Hatami, K. Hosseini, and S. Lovett. Pseudorandom generators from polarizing random walks. *Electronic Colloquium on Computational Complexity (ECCC)*, 25:15, 2018.
- [CHRT17] E. Chattopadhyay, P. Hatami, O. Reingold, and A. Tal. Improved pseudorandomness for unordered branching programs through local monotonicity. *Electronic Colloquium on Computational Complexity (ECCC)*, 24:171, 2017.
- [DETT10] A. De, O. Etesami, L. Trevisan, and M. Tulsiani. Improved pseudorandom generators for depth 2 circuits. In *APPROX-RANDOM*, pages 504–517, 2010.
- [GKM15] P. Gopalan, D. M. Kane, and R. Meka. Pseudorandomness via the discrete fourier transform. In *FOCS*, pages 903–922, 2015.
- [GMR⁺12] P. Gopalan, R. Meka, O. Reingold, L. Trevisan, and S. P. Vadhan. Better pseudorandom generators from milder pseudorandom restrictions. In *FOCS*, pages 120–129, 2012.
- [GY14] P. Gopalan and A. Yehudayoff. Inequalities and tail bounds for elementary symmetric polynomial. *CoRR*, abs/1402.3543, 2014.

- [HLV17] E. Haramaty, C. H. Lee, and E. Viola. Bounded independence plus noise fools products. In *32nd Computational Complexity Conference, CCC 2017*, pages 14:1–14:30, 2017.
- [INW94] R. Impagliazzo, N. Nisan, and A. Wigderson. Pseudorandomness for network algorithms. In *Proceedings of the 26th annual STOC*, pages 356–364, 1994.
- [Lov08] S. Lovett. Unconditional pseudorandom generators for low degree polynomials. In *40th Annual STOC*, pages 557–562, 2008.
- [LV17] C. H. Lee and E. Viola. More on bounded independence plus noise: Pseudorandom generators for read-once polynomials. *Electronic Colloquium on Computational Complexity (ECCC)*, 24:167, 2017.
- [Nis92] N. Nisan. Pseudorandom generators for space-bounded computation. *Combinatorica*, 12(4):449–461, 1992.
- [NN93] J. Naor and M. Naor. Small-bias probability spaces: Efficient constructions and applications. *SIAM J. on Computing*, 22(4):838–856, 1993.
- [O’D14] R. O’Donnell. *Analysis of boolean functions*. Cambridge University Press, 2014.
- [SSS95] J. P. Schmidt, A. Siegel, and A. Srinivasan. Chernoff-hoeffding bounds for applications with limited independence. *SIAM J. Discrete Math.*, 8(2):223–250, 1995.
- [Ste13] J. P. Steinberger. The distinguishability of product distributions by read-once branching programs. In *Proceedings of the 28th Conference on Computational Complexity, CCC 2013*, pages 248–254, 2013.
- [SVW14] T. Steinke, S. Vadhan, and A. Wan. Pseudorandomness and fourier growth bounds for width-3 branching programs. *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, page 885, 2014.
- [SZ95] M. Saks and D. Zuckerman. Personal Communication, 1995.
- [SZ11] J. Síma and S. Zák. Almost k -wise independent sets establish hitting sets for width-3 1-branching programs. In *CSR*, pages 120–133, 2011.
- [Ta-17] A. Ta-Shma. Explicit, almost optimal, epsilon-balanced codes. In *Proceedings of the 49th Annual ACM SIGACT STOC 2017*, pages 238–251, 2017.
- [Tre10] L. Trevisan. Open problems in unconditional derandomization. *Presentation at China Theory Week*, 2010.
- [Vio08] E. Viola. The sum of d small-bias generators fools polynomials of degree d . In *Proceedings of the 23rd Annual IEEE Conference on Computational Complexity (CCC)*, pages 124–127, 2008.

A Appendices

A.1 Proof of Theorem 3.10

In this section, we view the Boolean functions computed by branching programs as functions $B : \{\pm 1\}^n \rightarrow \{0, 1\}$. For any set $T \subseteq [n]$, this changes the sum $\sum_{S \subseteq T} |\widehat{B}(S)|$ by a factor of 2, which we can afford.

Let B be a ROBP of length n and width w . Recall that V_1, \dots, V_{n+1} denote the layers of vertices in B . For a vertex $v \in V_i$ in the branching program we denote by $B_{\rightarrow v}$ the sub-branching program ending in the i -th layer and having v the only accepting state. We denote by $B_{v \rightarrow}$ the sub-branching program starting at v and ending at V_{n+1} . Observe that we may express the function computed by the branching program B as a sum of products of these sub-programs, namely

$$\forall i \in [n] : \forall x \in \{\pm 1\}^n : B(x) = \sum_{v \in V_i} B_{\rightarrow v}(x) \cdot B_{v \rightarrow}(x). \quad (13)$$

The main technical result from [CHRT17] is the following theorem:

Theorem A.1 ([CHRT17, Thm. 2]). *Let B be an ordered read-once, oblivious branching program of length n and width w . Then,*

$$\forall k \in [n] : \sum_{s:|s|=k} |\widehat{B}(s)| \leq O(\log n)^{wk}.$$

We are ready to prove a corollary of this theorem, namely Theorem 3.10.

Theorem A.2 (Thm. 3.10, restated). *Let B be a width- w length- n ROBP. Let $\varepsilon > 0$, $p \leq 1/O(\log n)^w$, $k = O(\log(n/\varepsilon))$, and \mathcal{D} be a δ_T -biased distribution over $[n]$ with marginals p , where $\delta_T \leq p^{2k}$. Then, with probability at least $1 - \varepsilon$ over $T \sim \mathcal{D}$,*

$$L_1(\widetilde{B}) = \sum_{S \subseteq T} |\widehat{B}(S)| \leq O((nw)^3/\varepsilon).$$

Claim A.3. *For all $\beta > 0$, the following holds with probability at least $1 - \frac{w^2 \cdot n^3}{\beta}$ over T : for all v_0 and v and $1 \leq j \leq \min\{2k, n\}$:*

$$\sum_{s \subseteq T, |s|=j} |\widehat{B_{v_0 \rightarrow v}}(s)| \leq \frac{\beta}{2^j}. \quad (14)$$

Proof. Fix v_0 and v . Letting M denote the branching program $B_{v_0 \rightarrow v}$ we get $\sum_{s:|s|=j} |\widehat{M}(s)| \leq O(\log n)^{wj}$ from Theorem A.1. Thus,

$$\mathbf{E}_T \left[\sum_{s:|s|=j} |\widehat{M}(s)| \cdot \mathbb{1}_{\{s \subseteq T\}} \right] = \sum_{s:|s|=j} |\widehat{M}(s)| \cdot \mathbf{Pr}_T[s \subseteq T] \leq O(\log n)^{wj} \cdot (p^j + \delta) \leq \frac{1}{2^j}.$$

Finally, we conclude by applying the Markov inequality and a union bound, as there is a total of at most $w^2 \cdot n^2$ branching programs $B_{v_0 \rightarrow v}$ and at most n choices for j . \square

Theorem 3.10 follows from the next claim which uses Claim A.3 with $\beta = (nw)^3/\varepsilon$ and $k = O(\log(n/\varepsilon))$ that ensure $\frac{w^2 \cdot n^3}{\beta} \leq \varepsilon$ and $\frac{\beta}{2^k} \leq \frac{\varepsilon}{nw}$. Indeed, with probability at least $1 - \varepsilon$, the spectral-norm of \tilde{B} is at most $1 + \sum_{j=1}^k \frac{\beta}{2^j} + (n - k) \cdot \frac{\varepsilon}{nw} \leq 2 + \beta$.

Claim A.4. *Suppose that T is such that the events in Claim A.3 hold for β, k such that $\beta/2^k \leq \varepsilon/(nw)$. Then for every j such that $k \leq j \leq n$,*

$$\sum_{s \subseteq T, |s|=j} |\widehat{B}(s)| \leq \frac{\varepsilon}{nw}. \quad (15)$$

Proof. We prove by induction on j that Eq. (15) holds for all $B_{\rightarrow v}$, for any $\ell \in [n + 1]$ and $v \in V_\ell$. Note that B itself is of the form $B_{\rightarrow v}$ for v being the accept node in the final layer (w.l.o.g. there exists only one such node). The case $k \leq j \leq 2k$ is handled by Claim A.3, since $\sum_{s \subseteq T: |s|=j} |\widehat{B_{\rightarrow v}}(s)| \leq \frac{\beta}{2^j} \leq \frac{\beta}{2^k} \leq \frac{\varepsilon}{(nw)^2}$. For $j > 2k$ we have:

$$\begin{aligned} \sum_{s \subseteq T: |s|=j} |\widehat{B_{\rightarrow v}}(s)| &\leq \sum_{i \in T \cap [\ell]} \sum_{v_0 \in V_i} \sum_{\substack{s_0 \subseteq T \cap \{1, \dots, i-1\}: \\ |s_0|=j-k}} \sum_{\substack{s_1 \subseteq T \cap \{i, \dots, \ell\}: \\ |s_1|=k, i \in s_1}} |\widehat{B_{\rightarrow v_0}}(s_0) \cdot \widehat{B_{v_0 \rightarrow v}}(s_1)| \\ &\hspace{20em} \text{(by Eq. (13))} \\ &\leq \sum_{i \in T \cap [\ell]} \sum_{v_0 \in V_i} \left(\sum_{\substack{s_0 \subseteq T \cap \{1, \dots, i-1\}: \\ |s_0|=j-k}} |\widehat{B_{\rightarrow v_0}}(s_0)| \right) \cdot \left(\sum_{\substack{s_1 \subseteq T \cap \{i, \dots, \ell\}: \\ |s_1|=k, i \in s_1}} |\widehat{B_{v_0 \rightarrow v}}(s_1)| \right) \\ &\leq \sum_{i \in T \cap [\ell]} \sum_{v_0 \in V_i} \frac{\varepsilon}{nw} \cdot \frac{\varepsilon}{nw} \leq \frac{\varepsilon}{nw} \hspace{10em} \text{(induction and Claim A.3)} \end{aligned}$$

This completes the induction, and hence the claim follows. \square

A.2 Restatement of XOR-lemma for functions fooled by small-biased spaces

In this section we show how Lemma 5.1 is a restatement of Thm 4.1 in [GMR⁺12]. We recall the following equivalence between having sandwiching approximations with small spectral-norm and being fooled by every small-biased distribution.

Lemma A.5 ([DETT10]). *Let $f : \{\pm 1\}^n \rightarrow \mathbb{R}$ be a function. Then, the following hold for every $0 < \varepsilon < \delta$:*

- *If f has δ -sandwiching approximations of spectral-norm at most δ/ε , then for every ε -biased distribution D on $\{\pm 1\}^n$, $|\mathbf{E}_{x \sim D}[f(x)] - \mathbf{E}[f]| \leq \delta$.*
- *If for every ε -biased distribution D on $\{\pm 1\}^n$, $|\mathbf{E}_{x \sim D}[f(x)] - \mathbf{E}[f]| \leq \delta$, then f has (2δ) -sandwiching approximations of spectral-norm at most $1 + \delta/\varepsilon$.*

We recall [GMR⁺12, Thm. 4.1].

Theorem A.6 ([GMR⁺12, Thm. 4.1]). *Let $F_1, \dots, F_k : \{\pm 1\}^n \rightarrow [0, 1]$ be functions on disjoint input variables such that each F_i has δ -sandwiching approximation of spectral-norm at most t . Let $H : [0, 1]^k \rightarrow [0, 1]$ be a multilinear function in its inputs. Let $h : \{\pm 1\}^n \rightarrow [0, 1]$ be defined as $h(x) = H(F_1(x), \dots, F_k(x))$. Then h has $(16^k \delta)$ -sandwiching approximations of spectral-norm at most $4^k(t+1)^k$.*

We translate the domain $[0, 1]$ to $[-1, 1]$ to get a restatement of the previous theorem.

Theorem A.7 ([GMR⁺12, Thm. 4.1], ± 1 -version). *Let $F_1, \dots, F_k : \{\pm 1\}^n \rightarrow [-1, 1]$ be functions on disjoint input variables such that each F_i has δ -sandwiching approximation of spectral-norm at most t . Let $H : [-1, 1]^k \rightarrow [-1, 1]$ be a multilinear function in its inputs. Let $h : \{\pm 1\}^n \rightarrow [-1, 1]$ be defined as $h(x) = H(F_1(x), \dots, F_k(x))$. Then h has $(16^k \delta)$ -sandwiching approximations of spectral-norm at most $2^{k+1}(t+4)^k$.*

Proof. We take F'_1, \dots, F'_k to be $\frac{F_1+1}{2}, \dots, \frac{F_k+1}{2}$ respectively. We get that F'_i has $\delta/2$ -sandwiching approximations of spectral-norm at most $(t+1)/2$, for all $i \in \{1, \dots, k\}$. We take $H' : [0, 1]^k \rightarrow [0, 1]$ to be $H'(y_1, \dots, y_k) = \frac{1+H(2y_1-1, \dots, 2y_k-1)}{2}$. Since H is multilinear, so is H' . By Theorem A.6, we get that $H'(F'_1, \dots, F'_k)$ has $(16^k \cdot \delta/2)$ -sandwiching approximations of spectral-norm at most $4^k(\frac{t+1}{2}+1)^k$. Since $H(F_1, \dots, F_k) = 2 \cdot H'(F'_1, \dots, F'_k) - 1$ we got that H as a $(16^k \cdot \delta)$ -sandwiching approximations of spectral-norm at most $1 + 2 \cdot 4^k(\frac{t+1}{2} + 1)^k = 1 + 2 \cdot 2^k(t+3)^k \leq 2^{k+1} \cdot (t+4)^k$. \square

Finally, we restate Lemma 5.1 and prove it.

Lemma A.8. *Let $0 < \varepsilon < \delta \leq 1$. Let $F_1, \dots, F_k : \{\pm 1\}^n \rightarrow [-1, 1]$ be functions on disjoint input variables such that each F_i is δ -fooled by any ε -biased distribution. Let $H : [-1, 1]^k \rightarrow [-1, 1]$ be a multilinear function in its inputs. Then $H(F_1(x), \dots, F_k(x))$ is $(16^k \cdot 2\delta)$ -fooled by any ε^k -biased distribution.*

Proof. Using the second item in Lemma A.5, since F_1, \dots, F_k are δ -fooled by any ε -biased distribution, we have that there exist 2δ -sandwiching approximations of spectral-norm at most $1 + \delta/\varepsilon$. Thus by Thm. A.7, $H(F_1, \dots, F_k)$ has $(16^k \cdot 2\delta)$ -sandwiching approximations of spectral-norm at most $2^{k+1} \cdot (\delta/\varepsilon + 5)^k$. Set $\delta' := 16^k \cdot 2\delta$ and $\varepsilon' := \delta' / (2^{k+1} \cdot (\delta/\varepsilon + 5)^k)$. Then, $H(F_1, \dots, F_k)$ has δ' -sandwiching approximations of spectral-norm at most δ'/ε' . Using the first item in Lemma A.5 (noting that $\varepsilon' < \delta'$), any ε' -biased distribution δ' -fools $H(F_1, \dots, F_k)$. A small calculation shows that $\varepsilon' \geq \varepsilon^k$, hence any ε^k -biased distribution also δ' -fools $H(F_1, \dots, F_k)$. \square

A.3 Pseudorandom restrictions for the composition of 3ROBPs

We restate and prove Lemma 7.13.

Lemma A.9. *Let f_1, \dots, f_k be 3ROBPs on disjoint sets of variables of $[n]$. Let $H : \{\pm 1\}^k \rightarrow \{\pm 1\}$ be any Boolean function. Then, $f = H(f_1, f_2, \dots, f_k)$ is $(\delta \cdot (n+1)^k)$ -fooled by the pseudorandom partial assignment in Claim 7.1 with parameter δ .*

Proof. Claim 7.1 applies Theorem 4 iteratively $t < n$ times with error parameter δ/n . Thus, it suffices to show that under each application of pseudorandom restriction from Theorem 4 the acceptance probability of $H(f_1, f_2, \dots, f_k)$ changes by at most $(\delta/n) \cdot (n+1)^k$.

Let $\varepsilon := \delta/n$. Let $V(f_1), \dots, V(f_k)$ be the sets of variables on which f_1, \dots, f_k depend. We write H in the Fourier basis: $H(y_1, \dots, y_k) = \sum_{S \subseteq [k]} \widehat{H}(S) \cdot \prod_{i \in S} y_i$. Thus, $H(f_1(x), \dots, f_k(x)) = \sum_{S \subseteq [k]} \widehat{H}(S) \cdot \prod_{i \in S} f_i(x)$. Recall that the pseudorandom assignment in Theorem 4 is composed of two stages: Let $\varepsilon_1 = \varepsilon/2$ and $\varepsilon_2 = \varepsilon/2n$.

1. Pick $T_0 \subseteq [n]$ using a $(\varepsilon_1/n)^{10}$ -biased distribution with marginals $1/2$.
2. Assign the coordinates in $[n] \setminus T_0$ uniformly at random.
3. (a) Pick $T \subseteq T_0$ using a δ_T -biased distribution with marginals $p = 1/O(\log \log(n/\varepsilon_2))^6$.
 (b) Assign the coordinates in $T_0 \setminus T$ uniformly at random.
 (c) Assign the coordinates in T using a $(\varepsilon_2/n)^{O(\log \log(n/\varepsilon_2))}$ -biased distribution \mathcal{D}_x .

Recall that for a fixed T_0 , the bias-function of any program f_i behaves the same under any relabeling of the layers in $[n] \setminus T_0$. We imagine as if these layers are relabeled so that a collision is possible, and denote this relabeled program by $f_i^{T_0}$. We have $\text{Bias}_{T_0}(f_i)(x) = \mathbf{E}_{y \sim U_{[n] \setminus T_0}}[(f_i^{T_0})_{T_0|y}(x)]$ and similarly since the sets $V(f_1), \dots, V(f_k)$ are disjoint $\text{Bias}_{T_0}(H(f_1, \dots, f_k))(x) = \mathbf{E}_{y \sim U_{[n] \setminus T_0}}[H((f_1^{T_0})_{T_0|y}(x), \dots, (f_k^{T_0})_{T_0|y}(x))]$. By Theorems 4.1 and 4.2, with probability at least $1 - \varepsilon_1 \cdot k$ the choice of T_0 and y , we can write each $(f_i^{T_0})_{T_0|y}(x)$ for $i = 1, \dots, k$ as a linear combination of $\prod_{j \in [m_i]} [f_{i,j}]$ where the sum of coefficients in absolute value is at most the number of variables in f_i (i.e., $|V(f_i)|$), and each $f_{i,j}$ is a ROBP on at most $O(\log(n/\varepsilon))$ bits. Overall with high probability over T_0, y the product $\prod_{i \in S} (f_i^{T_0})_{T_0|y}$ can be written as a linear combination of the functions $\prod_{i \in S} \prod_{j \in [m_i]} [f_{i,j}]$ where the sum of coefficients in absolute values in the linear combination is at most $\prod_{i \in S} |V(f_i)|$. Thus, $H((f_1^{T_0})_{T_0|y}, \dots, (f_k^{T_0})_{T_0|y})$ can be written as a linear combination of XOR of $O(\log(n/\varepsilon))$ -length width-3 ROBPs where the sum of coefficients is at most

$$\sum_{S \subseteq [k]} |\widehat{H}(S)| \cdot \prod_{i \in S} |V(f_i)| \leq \sum_{S \subseteq [k]} 1 \cdot \prod_{i \in S} |V(f_i)| = \prod_{i=1}^k (1 + |V(f_i)|) \leq (n+1)^k$$

By Theorem 5, each XOR of $O(\log(n/\varepsilon))$ -length width-3 ROBPs is ε_2 -fooled by the pseudorandom assignment defined by Step 3 above, thus the overall error is at most $\varepsilon_1 \cdot k + \varepsilon_2 \cdot (n+1)^k \leq \varepsilon \cdot (n+1)^k = (\delta/n) \cdot (n+1)^k$. \square